Improved strategies for distance based clustering of objects on subsets of attributes in high-dimensional data

Kampert, M.M.D.

Chapter 5

Obtaining L Groups from COSA Distances

Although the main output of COSA are distances that we have labeled "cluster-happy", it is not straightforward to extract L groups from a COSA distance matrix. COSA on its own is not equipped to be compared with L-groups clustering algorithms. COSA does not provide the option of choosing L, instead, it circumvents the pre-specification of the number of clusters by working with N neighborhoods. Thus, an extra step is required when the main objective is to extract L clusters from the COSA distances.

The chapter outline is as follows. In the first section of this chapter we will discuss two popular methods that have been applied already by others to extract L clusters from the COSA-KNN distances. The first method is referred to as Partitioning Around Medoids (PAM; Kaufman & Rousseeuw, 1987), and we will show that it does not provide optimal results when it is applied. The second method is hierarchical clustering that is followed by cutting a dendrogram. We will show that cutting a dendrogram may actually be a good choice for COSA, but only if Ward's minimum variance method is used. However, Ward's method is not sensitive enough to detect small homogeneous groups that have overlapping subsets of attributes, and are in the presence of a large heterogeneous or residual group of noise objects. In these latter data settings, Ward's method splits the large heterogeneous cluster into smaller clusters, and merges the smaller homogeneous clusters, as will be demonstrated in Section 5.2.

To overcome this disadvantage of Ward's method, we present in Section 5.3 a new algorithm that uses a Minimum Variance strategy to Partition In Neighborhoods: MVPIN. This new algorithm applies a restricted form of Ward's method on the distances such that it is able to detect small homogeneous clusters that are similar to each other, in the presence of large heterogeneous clusters. In MVPIN, the merging of clusters by Ward's Minimum Variance method is restricted by the popular Jarvis and Patrick (1973) shared nearest neighbors similarity; that is, clusters are only allowed

the merge if a similarity measure based on the *shared nearest neighbors* between the two clusters is the highest among the similarities that pass a certain threshold.

The performance of MVPIN is demonstrated in Sections 5.4 and 5.5 on a synthetic data set and 11 benchmark data sets. In Section 5.4 the results for MVPIN are compared to Ward's method and to PAM when L, the number of clusters, is specified beforehand. Because the number of clusters is usually not known beforehand, we also show in Section 5.5 the performance of MVPIN when the number of clusters is decided by the Gap statistic procedure. Section 5.6 provides the conclusion and discussion of the chapter.

5.1 Obtaining L Groups from Distances

Among the most popular methods to cluster a distance matrix into L groups are medoid based methods such as Partitioning Around Medoids (PAM; Kaufman & Rousseeuw (1987) and hierarchical clustering methods (Wiwie, Baumbach & Röttger, 2015; de Souto et al. 2008; Kaufman & Rousseeuw, 1990; Jain & Dubes, 1988). To our knowledge, PAM and hierarchical clustering are the only two methods that have been used to cluster COSA distances into L groups: this was done in a study by Jing et al. (2007) and a study by Witten and Tibshirani (2010). Since the main purposes of these studies were not related to COSA, little attention was given to what method should have been used to extract L groups from the COSA distances. From Jing et al. (2007) one can only retrieve that a hierarchical clustering algorithm was used, but what type of hierarchical clustering and what selection procedure to extract L groups remains unclear. The procedures that were used in the study by Witten and Tibshirani (2010), however, can be replicated. These were

1. applying partitioning around medoids (PAM) to the COSA distances;
2. cutting an average linkage dendrogram for the COSA distances at a given height that leads to L groups.

These two choices to extract L groups from the COSA distances are suboptimal. Even for what we call a 'perfect' distance matrix, these choices can easily result in assigning the wrong group labels, as will be shown in the next sections.

5.1.1 Partitioning Around Medoids

The objective of PAM (Kaufman & Rousseeuw, 1987) is to minimize the distances between the medoid and the objects within each cluster. Here, a medoid within a cluster is the object that is designated to be most representative for the cluster. PAM can be applied directly on the COSA distances. Figure 5.1 displays the average linkage dendrogram and the MDS configuration of the COSA-λNN distances, obtained for a simulated data set from the COSA prototype model where there are two small clusters and one large remaining group. Thus, $L = 3$. The colors in Figure 5.1 represent the cluster labels that are the result of PAM when applied directly on the COSA distances.

As can be seen, PAM is able to find the two small homogeneous groups that cluster on overlapping subsets of attributes. However, some of the noise objects are also assigned to one of the two small clusters (colored red instead of grey). Apparently, the COSA distance between each of these wrongly classified noise objects and the medoid of the red cluster, is smaller than the distance between these 'red' noise objects and the medoid of the 'grey' noise objects. This undesirable result is inherent to the objective of PAM, i.e. minimizing the sum over the distances of each object with the medoid of the cluster it belongs to.

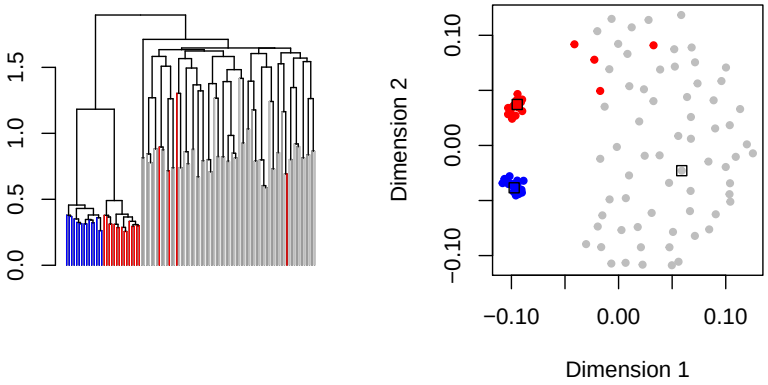

Figure 5.1: An average linkage dendrogram (left panel) and the MDS configuration (right panel) of the COSA distances. The colors of the objects are the labels that are the results of applying PAM to the COSA distances directly. The medoid of each PAM cluster is indicated with a square.

Although close, PAM does not result in an optimal partition for clusters when applied to a distance matrix that represents clusters of different densities. This effect can be easily detected when we compare the resulting clusters from PAM on their 'thightness' and 'separability', as can be done by the silhouette (Rousseeuw, 1987). For every object i we can compute its silhouette, silh(i), by combining two concepts. The first concept is the average distance of object i with all other objects within the cluster of object i, defined as

$$\text{AVW}_i = \sum_{j \neq i} c_{jl^*} D_{ij}[\mathbf{W}] \Big/ \sum_{j \neq i} c_{jl^*} \,, \tag{5.1}$$

where l^* is the cluster that object i is assigned to. The second concept is the minimum average distance of object i to the objects of any other cluster than l^*, defined as

$$\text{MINAVB}_i = \min_{l \,|\, l \neq l^*} \left\{ \sum_{j \neq i} c_{jl} D_{ij}[\mathbf{W}] \Big/ \sum_{j \neq i} c_{jl} \right\}. \tag{5.2}$$

Then, the silhouette of an object is expressed as

$$\text{silh}(i) = \frac{\text{MINAVB}_i - \text{AVW}_i}{\max\{\text{AVW}_i, \text{MINAVB}_i\}}. \tag{5.3}$$

The silhouette of object i is an indication of how well object i matches its cluster. The closer the value is to 1, the better object i matches its cluster. When the silhouette of object i becomes negative (with a minimum of -1), it is an indication that it would have been more natural to assign object i to its closest neighboring cluster. If an object is on its own a cluster of size 1, then its silhouette is set equal to 0, giving it a 'neutral' value.

When we use the graphical display of the silhouette (Rouseeuw, 1987), we can see four 'red' clustered objects of which three have obtained a negative silhouette and one has an 'outlying' low positive silhouettte, see Figure 5.2. These are the four wrongly clustered objects. Having many positive silhouettes within a cluster shows

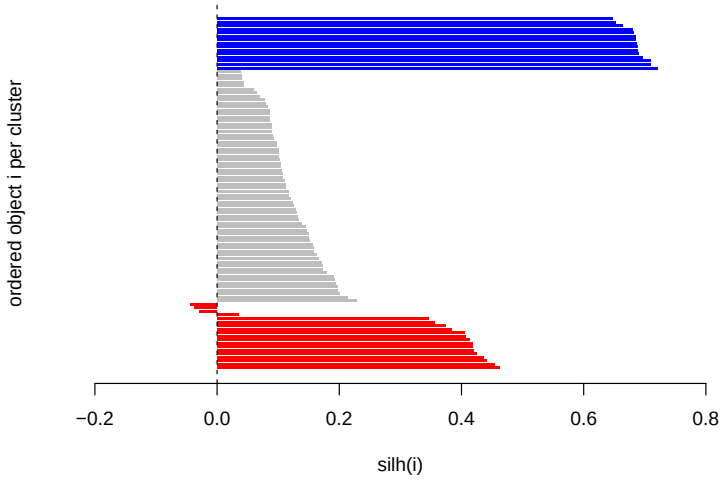

Figure 5.2: Graphical Display of silhouettes of the PAM clustering solution, each colored horizontal bar respresents an object. The length of the bar represents the object's silhouette.

that the cluster is well separable, and the higher the values of the positive silhouettes, the thighter the clustering. Rousseeuw (1987) proposed to use the average of the silhouettes within a cluster as a measure of validation of that cluster, and suggests that the average over all the silhouettes can be used as a measure of validation to evaluate the strength of the resulting partition of the set of objects, and therefore may serve for the selection of the optimal number of clusters in a data set. Although for low-dimensional data settings ($N < P$), in an extensive study about cluster validation measures by Arbelaitz et al. (2013) it is shown that the silhouette (still) outperforms more recent cluster measures for choosing the number of clusters.

A reason that PAM does not show an optimal performance on the COSA distances is that PAM is sensitive to violations of the triangular inequality. In a strict sense,

the COSA distances are non-metric dissimilarities, i.e., the distances may violate the triangular inequality. The higher the differences in the subsets of attribute weights in COSA, the more likely it is that the COSA distances will violate the triangle inequality property (see Chapter 2). Suppose two objects, o_1 and o_2, are dissimilar to each other, but similar to a third object o_3. Then, it may still be possible that when o_3 is a medoid, that o_1 and o_2 end up in the same cluster through a violation of the triangular inequality. The path from o_1 to o_2 via the candidate medoid o_3 can be shorter than the direct path from o_1 to o_2:

$$D_{12} > D_{13} + D_{23}. \tag{5.4}$$

Such triangular violations can complicate the discovery of homogeneous clusters in PAM. For example, when the medoid 3 is being updated to medoid 2, suddenly the same group would become more heterogeneous. The larger the number of distances that violate the triangular inequality, the more the medoids based algorithms are affected, and the more likely it becomes that the underlying clustering structure cannot be revealed. For a further explanation, see Baraty, Simovici, and Zara (2011). Thus, misrepresentations of within-cluster homogeneity can obscure the partitioning in the data.

We can avoid the violations of the triangle inequality by applying PAM directly to the Euclidean distances of a best-fitting MDS configuration of $N - 1$ dimensions. In De Leeuw and Groenen (1997) it is shown that the MDS configuration in $N - 1$ dimensions results in metric Euclidean distances for which the global minimum of the STRESS function is achieved (for the STRESS function, see Chapter 1, page 16, equation (1.12)). However, applying PAM to the best fitting Euclidean distances, or any of the Euclidean distances that correspond to lower-dimensional configurations, results in more misclassified objects for the particular data sets that are generated from the prototype model. Although the minimized STRESS function gives a configuration matrix of which the Euclidean distances between the objects with large COSA distances remain relatively large (as compared to classical scaling solutions), it may come at the cost of the distances between the objects of the two smaller clusters. Thus, fixing the triangular inequality violations with the use of MDS does not provide better results for PAM.

The worst performance of PAM in combination with an MDS configuration is when default settings of the implementations of `smacof()` and `pam()` are applied in `R`. Extracting the configuration matrix $\mathbf{X}$ in `R` using the default options in `smacof()`, and then feeding this configuration matrix into `pam()` with its default options, gives results as shown in Figure 5.3. Note that this is an easily made, but inconsistent combination of choices. When the input of `pam()` is a configuration matrix, it computes the Manhattan distances in $\mathbf{X}$, while the specific MDS configuration matrix is actually supposed to contain Euclidean distances. The results of using this detrimental strategy are shown in Figure 5.3. While in Figure 5.1 the two smaller clusters were detected, now the two smaller clusters are interpreted as one red cluster, and the noise objects are split into two groups. These results are consistent with findings in Van der Laan et al. (2003); PAM may experience difficulties in recognizing relatively small clusters.

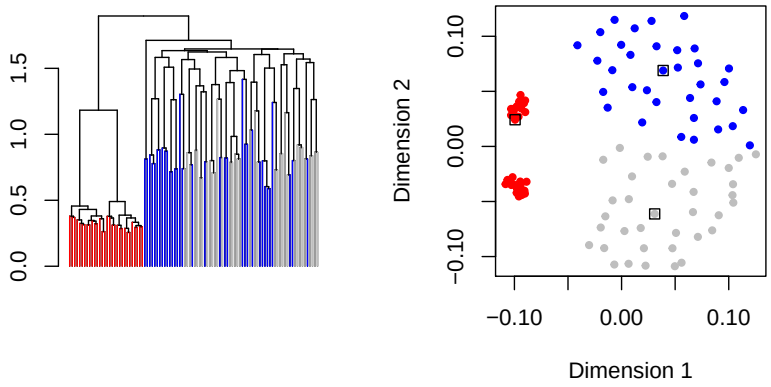

Figure 5.3: The resulting L groups of PAM on a two-dimensional MDS configuration of the COSA distances, shown in an average linkage dendrogram (left), and in the MDS configuration (right), in which the medoids are indicated as points within a square.

5.1.2 Cutting the Dendrogram

Another common approach to extract L clusters from a distance matrix is by cutting a dendrogram. Here, we will show with an example why cutting an average linkage dendrogram of a perfect COSA distance matrix is suboptimal. Since COSA can reveal clusters with different densities, it is common to see in average linkage dendrograms that all members in a group will meet each other at different heights. However, one of the standard rules in cutting a dendrogram, is to cut at one specific height only (Jain & Dubes 1988, Kaufman & Rousseeuw, 1990).

Suppose we wish to cut the average linkage dendrogram from the default COSA-λNN results on a simulated data set from the COSA prototype model, displayed in the left panel of Figure 5.4. The ground truth seems to be perfectly revealed: a clustering structure of two homogeneous separate groups that seem to nest together into one bigger group (due to sharing of a subset of attributes), and a remainder heterogeneous group, containing the noise objects (in grey). Although from a pure COSA perspective one could argue that the results of one cut only will be good enough, e.g., a cut that results in two groups of 15 objects and 70 singleton objects, it is not possible to extract the $L = 3$ groups in just one cut. Note that this may be undesirable when the noise objects could be interpreted as a heterogeneous group with disturbances in the (gen)omics system (Van Wieringen & Van der Vaart, 2015).

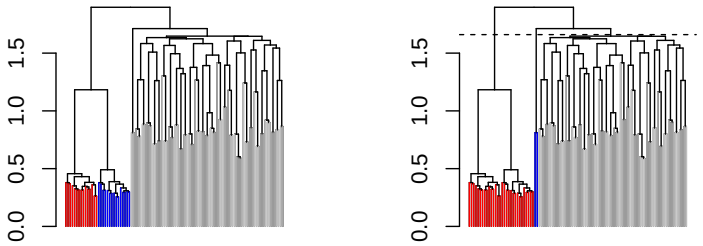

Figure 5.4: The average linkage dendrograms of the COSA-λNN distance obtained from the data set that was generated from the prototype mode. In the left panel each object is labeled with the color of its cluster group according to the truth structure, in the right panel the objects are colored according to their group label obtained by cutting the dendrogram.

If the only purpose of performing COSA would be to fit a dendrogram on the COSA distances that needs to be cut in L groups, than better results can be expected from a density based linkage strategy. While linking an out-of-cluster object to an already large group may not have a large impact on the average distance, it will have a large impact on the density of the distances within that group. Thus, linking according to a strategy of a minimum variance increase, such as Ward's minimum variance method (Ward, 1963; Wishart, 1969), would provide better results for the sole purpose of finding $L = 3$ groups; the two small homogeneous clusters and the residual group of noise objects (see Figure 5.5). This is a finding in line with Jain and

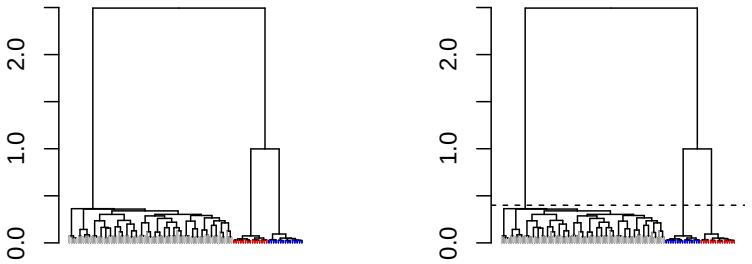

Figure 5.5: The Ward's method dendrogram for the COSA distances of the prototype model data set with each object labelled with a color of its cluster group according to the ground truth (on the left), and the same dendrogram with each of the objects colored according to the group label obtained from cutting the dendrogram (on the right).

Dubes (1988), in which an overview of several comparative studies was presented with the conclusion that for the purpose of dendrogram cutting at a specific height to find L groups, Ward's method outperforms other hierarchical clustering methods (among which single, average, and complete linkage). For the purpose of plotting the COSA distances, however, the dendrograms based on Ward's minimum variance method are suboptimal. The distances between the noise objects are supposed to be much larger than the distances of the objects within the homogeneous clusters.

5.2 A Closer Look at Ward's Method

Originally, Ward's method has been proposed as an agglomerative procedure for forming hierarchical groups of mutually exclusive clusters for rating values in a set of N objects and P attributes. Here, the N objects are progressively fused into hierarchical clusters by applying an algorithm to minimize a certain objective function that can "be any functional relation that an investigator selects to reflect the relative desirability of groupings" (Ward, 1963). As an example for an objective function that would represent loss of information, Ward (1963) used the 'error sum of squares'. Applying the algorithm of Ward's method based on the error sum of squares loss function is also referred to as Ward's minimum variance method.

Using Ward's method to find L groups "probably will yield a good solution, although it may be one that does not optimize the objective function for the specified number of groups" (Ward, 1963), since the solution is restricted to a hierarchical grouping only. Thus, even though the title of the study by Ward (1963) is 'Hierarchical grouping to optimize an objective function', the method does not globally optimize the criterion for given L. In Ward's method the rate of change of the criterion, e.g., as in (5.15), is minimized with each union of two clusters starting from $L = N$ singleton clusters towards L equal to the desired number of clusters. At each step, the unification of the two 'intermediate' selected clusters produces the least impairment of the optimal value of the criterion.

5.2.1 The Distance-based Version of Ward's Method

Wishart (1969) generalized Ward's minimum variance method by rewriting it in terms of the distance-based update formula of the Lance-Williams family of clustering algorithms (Lance & Williams, 1967). The Lance-Williams updating formula describes how to compute distances between clusters that are formed by merging two existing clusters, and the remaining clusters. As we have seen in Chapter 2, using *Huygens' principle*, the error sum of squares within clusters can be rewritten as the sum of squared Euclidean distances. Using the squared Euclidean distances, Wishart (1969) derived the Lance-Williams update formula for Ward's method. Here, we will use the specific update formula on the squared COSA distances in equation (5.7).

Let each object initially be regarded as a singleton cluster, and the 'Ward' distance between two singleton clusters l and l' be defined as the COSA distance of object l and l', i.e.

$$\delta(l, l') = D_{ll'}[\mathbf{W}], \tag{5.5}$$

where, initially, the cardinality of cluster l, and l', are

$$N_l = N_{l'} = 1. \tag{5.6}$$

Thus, at the start, the between-clusters distance matrix Δ would be of size $N \times N$, where each element denoted by $\delta(l, l')$, is a COSA distance. Having initialized the Ward distances, we can recursively reduce the number of clusters by one. We find those two (still singleton) clusters for which $\delta(l, l')$ is smallest, and then fuse l and

l' into a new cluster of two objects. The number of clusters is now reduced by one. Then, to compute Δ again, which is now reduced to an $(N-1)\times(N-1)$ matrix, the Ward distances between the new cluster and all other clusters are updated via:

$$\delta^2\left(l\cup l', l''\right) = \frac{N_l + N_{l''}}{N_l + N_{l'} + N_{l''}}\delta^2\left(l, l''\right) + \frac{N_{l'} + N_{l''}}{N_l + N_{l'} + N_{l''}}\delta^2\left(l', l''\right) - \frac{N_{l''}}{N_l + N_{l'} + N_{l''}}\delta^2\left(l, l'\right), \tag{5.7}$$

where $l \neq l'$, $l' \neq l''$, $l'' \neq l'''$. In the subsequent steps the update formula from equation (5.7) is recursively applied to those two clusters l and l' that have the smallest Ward between-cluster distance until L clusters are reached and Δ is of size $L \times L$.

5.2.2 A Drawback of Ward's method

Although we have seen in the previous sections that for $L = 3$, the cluster labels obtained with Ward's method (Figure 5.5) are prefered over those obtained with PAM (Figure 5.1), or cutting the average linkage dendrogram (Figure 5.4), Ward's method has a disadvantage that needs to be taken into account when applied to COSA distances. Suppose we have a clustering structure that consists of a large heterogeneous group of noise objects and two small homogeneous groups that cluster exactly the same way on a large overlapping subset of attributes, but they also have a small unique subset of attributes each. In such a situation the update formula from equation (5.7) would assure a minimum increase at a step where the two homogeneous clusters are merged, instead of merging subclusters or objects from the large heterogeneous cluster.

Suppose we modify the COSA prototype model (from Figure 1.2 in Chapter 1) into a data model with an extra large subset of overlapping attributes, as can be obtained in Figure 5.6. Instead of having an overlapping subset of 15 attributes, we now have an overlapping subset of 27 attributes and two unique subsets, consisting of only 8 attributes each. From a data set generated from the 'extra overlap' model, compared to a data set from the COSA prototype model, we see in Figure 5.7 that Ward's method breaks down for $L = 3$ clusters (also does PAM, and cutting the average linkage dendrogram).

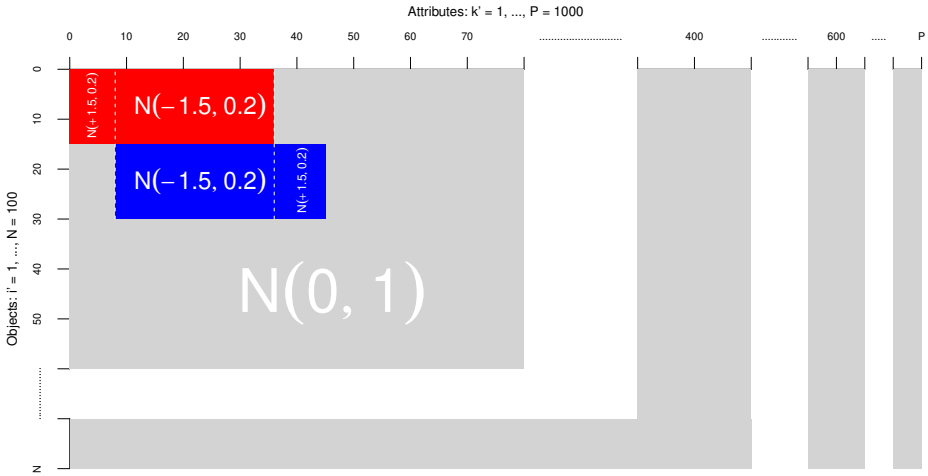

Figure 5.6: Extra Overlapping Attributes model for a data set with 100 objects and 1,000 attributes (of which only a limited number is shown). There are two groups of 15-objects (red and blue) each clustering on a subset of 35 attributes. On an overlapping subset of 27 attributes the clustering is exactly the same for the two groups. After generating a data set from this model, each attribute is scaled to unit variance and zero-mean.

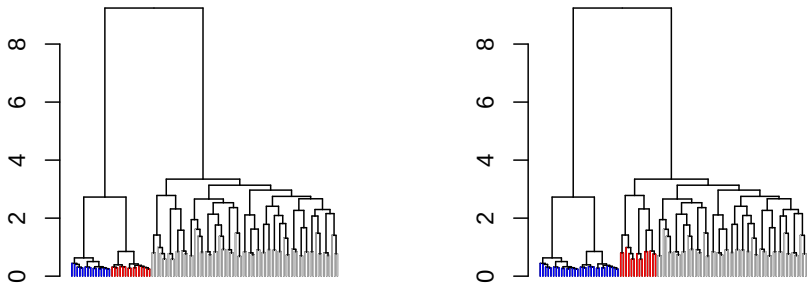

Figure 5.7: The Ward's method dendrograms fitted on the COSA distances of the data set from the 'extra overlap' model. In the left panel each object is colored according to the model in Figure 5.6, in the right panel each object is colored according to the group label obtained from cutting the dendrogram into three groups.

5.3 MVPIN

To obtain L groups of objects from COSA distances there are two properties that could be exploited. The first property is that the within cluster distances should be small. With respect to this first property we have seen that Ward's method performs well, by only merging clusters as long as the increment of the sum of the squared COSA distances is the smallest. The second property, that the within-cluster distances for COSA are based on neighborhoods of overlapping nearest neighbors, is not (yet) exploited. To be able to recover the small but stable homogeneous groups with a largely similar clustering pattern, we will also exploit this latter property.

We propose a new algorithm that can be implemented on a distance matrix, called MVPIN. While using Ward's Minimum Variance method, the algorithm Partitions In shared Neighborhoods. In MVPIN we combine Ward's method based on a modified version of the Jarvis and Patrick (1973) similarity measure that is based on nearest neighbors. In essence, MVPIN is an agglomerative clustering algorithm that applies Ward's minimum variance method as a linkage strategy on the COSA distances; however, the agglomeration process is steered differently. This different steering process renders MVPIN to be better able to recover unequally sized clusters, as long the number of shared nearest neighbors within each cluster is high.

5.3.1 Shared Nearest Neighbors

Instead of merging two clusters l and l' for which the Ward between-cluster distance is smallest, we pose an extra condition that the shared number of nearest neighbors between l and l should be the highest. Here, the shared number of nearest neighbors is based on a similarity measure between two objects as explained in Jarvis and Patrick (1973). Let K be the number of objects in a neighborhood, i.e. the neighborhood size. Then, the definition of the shared nearest neighbors similarity measure between objects i and j is

$$\widetilde{n}_{ij}(K) = \begin{cases} 1 + |\mathrm{KNN}(i) \cap \mathrm{KNN}(j)|, & \text{for } i \in \mathrm{KNN}(j) \text{ and } j \in \mathrm{KNN}(i) \\ 0, \text{ otherwise.} \end{cases} \tag{5.8}$$

Thus, $\widetilde{n}_{ij}(K)$ can be seen as the intersection of two neighborhoods of i and j given that i and j are in each others neighborhood. The neighborhoods $\mathrm{KNN}(i)$ and $\mathrm{KNN}(j)$ are each based on the COSA distances, as was defined in equation 2.22 of Chapter 2 (p. 30). According to Jarvis and Patrick (1973), the definition of the shared nearest neighbors similarity between two clusters l and l' is

$$\widetilde{N}_{ll'}(K) = \max_{\{i,j\}} \{\widetilde{n}_{ij}(K) \,|\, i \in C_l, j \in C_{l'}\}\,. \tag{5.9}$$

Note that the similarity in equation (5.9) can be seen as the inverse distance of the single linkage function between two clusters l and l'. Instead of linking the two clusters with the minimum cluster distance, we can link the two clusters with the maximum cluster similarity.

To assure the shared nearest neighbors similarity contributes to recovering a stable clustering structure in the data, a good value for K needs to be specified. The larger the value for K, the more object pairs will have a high number of shared nearest neighbors. Thus, by increasing K, eventually the between-cluster objects will have a similar number of shared nearest neighbors as within-cluster objects. Similarly, when the value for K is set to be too small, then, depending on the size of the cluster, the within-cluster objects may end up with too few shared nearest neighbors, or perhaps even none.

The idea of the similarity measure in MVPIN is that the value for K is chosen such that $K+1$ equals the size of the smallest 'clearly' detectable cluster in the data. Since the clustering structure in the data is unknown, thus we need a strategy to approximate K. We propose a strategy that approximates the value of K by searching for the smallest homogeneous group of objects that may be indicative for the smallest detectable cluster. Suppose that for each value of K the average number of shared nearest neighbors is expressed as a proportion of K, i.e.

$$p_K = \frac{1}{NK} \sum_{i=1}^{N} \sum_{j \in \mathrm{KNN}_{(i)}} \widetilde{n}_{ij}(K), \tag{5.10}$$

which is referred to as the proportion of shared nearest neighbors out of K nearest neighbors. Then, the idea is that the smallest detectable cluster in the data has size $K+1$, where the value for K corresponds to the first local maximum for p_K, where $p_{K-1} < p_K > p_{K+1}$. For an example, see Figure 5.8, where the input consisted of the COSA distances for a data set that was generated from the COSA model with extra overlapping attributes (Figure 5.6). In Figure 5.8 the value for K that corresponds to the first local maximum is equal to $K = 14$. Not surprisingly, $K+1$ corresponds to the size of the two smallest clusters, i.e. $\min_l(N_{\mathcal{C}_l}) = 15$. Here, the first local maximum exactly represents the total number of nearest neighbors that have the same cluster membership for an object of one of the two smallest clusters.

When the first local maximum at p_K is higher than the straight line between p_1 and p_{N-1}, then we set $\widehat{K}$ equal to the value for K that corresponds to this first local maximum. If, however, the first local maximum for K does not have a higher proportion of shared nearest neighbors than 'expected' with a linear increase from $K = 1$ to $K = N - 1$, then we set $\widehat{K}$ equal to $N - 1$, such that MVPIN reduces Ward's method (as will become clear in the next paragraph). Empirical evidence so far suggests that when the value for $\widehat{K}$ corresponds with a first local maximum for p_K that is lower than the linear 'expectation', then MVPIN becomes sensitive for small homogeneous clusters that are due to noise only.

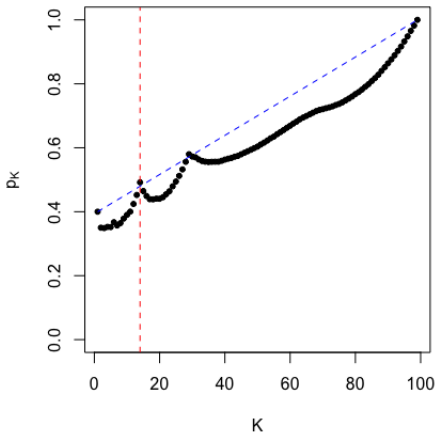

Figure 5.8: The proportion of the average of the shared nearest neighbors out of K, for $K \in \{1, \dots N-1\}$. The vertical red line indicates $\widehat{K}$, the first robust local maximum for p_K. The blue straight line connects $p_{K=1}$ with $p_{K=N-1}$

5.3.2 The MVPIN Update Formula

MVPIN is an algorithm based on the update formula for Ward's method on the COSA distances, however it is steered by the single linkage function $\widetilde{N}_{ll'}(K)$ from equation (5.9) based on the shared neighbors similarity $\widetilde{n}_{ij}(K)$ from equation (5.8). MVPIN only considers a fusion for clusters l^* and l'^* if

$$\{l^*, l'^*\} \in \underset{\{l,l'\}}{\operatorname{argmax}} \left(\widetilde{N}_{ll'}(K) \right), \tag{5.11}$$

where $l \neq l'$, and therefore $l^* \neq l'^*$. Thus, the update formula now becomes

$$\begin{aligned} \delta^2 \left(l^* \cup l'^*, l''^* \right) = & \frac{N_{l^*} + N_{l''^*}}{N_l + N_{l'^*} + N_{l''^*}} \delta^2 \left(l^*, l''^* \right) + \frac{N_{l'^*} + N_{l''^*}}{N_l + N_{l'^*} + N_{l''^*}} \delta^2 \left(l'^*, l''^* \right) \\ & - \frac{N_{l''^*}}{N_l + N_{l'^*} + N_{l''^*}} \delta^2 \left(l^*, l'^* \right), \end{aligned} \tag{5.12}$$

where $l^* \neq l'^*$, $l'^* \neq l''^*$, $l''^* \neq l'''^*$.

The first fusion step of the two singleton clusters l^* and l'^* are now based upon the minimized COSA distance for the maximized shared nearest neighbors similarity, i.e.

$$\min_{\{l^*,l'^*\}} \delta(l^*, l'^*) = \min_{\{l^*,l'^*\}} \left\{ D_{l^* l'^*}[\mathbf{W}] \;\middle|\; \{l^*, l'^*\} \in \underset{\{l,l'\}}{\operatorname{argmax}} \left(\widetilde{N}_{ll'}(K) \right) \right\}. \tag{5.13}$$

Note that MVPIN reduces to Ward's method when $\widehat{K} = N - 1$ since for all object pairs we will see that $\widetilde{n}_{ij}(\widehat{K}) = N - 1$.

The search for L clusters with MVPIN starts with finding $\widehat{K}$. Having set $\widehat{K}$, we compute the shared nearest neighbors similarity measures, $\{\widetilde{n}_{ij}\}$. Then, we apply the update formula in equation (5.12) to the COSA distances and the shared nearest neighbors similarity measures until the number of L clusters is reached. Thus, the algorithm can be described as

MVPIN
0 : Set: L;
1 : Compute: $\widehat{K}$, $\widetilde{n}_{ij}(\widehat{K}) \forall \{i, j\}$;
2 : Initialize: Δ, where each $\delta(i, j) = D_{ij}[\mathbf{W}]$;
3 : Loop {
Reduce size of Δ using equation (5.12) based on $\widetilde{N}_{ll'}(\widehat{K})$ (5.9)
Update $\mathbf{C}$
If (size of Δ is $L \times L$){Go to 4}
3 : }
4 : Output: $\mathbf{C}$.

(5.14)

5.4 MVPIN in Action

5.4.1 Results on Simulated Data

In Figure 5.9 three steps of the algorithm in (5.14) are shown for the synthetic data set of the COSA model with extra overlapping attributes from Figure 5.6. Objects that are merged into a group have been connected here with black lines. The connections are shown in a two-dimensional MDS configuration of the COSA-λNN distances for the data set that was also used in Figure 5.8. Thus, for this particular data set we have a neighborhood size of $\widehat{K} = 14$.

In the first (left) panel of 5.9, it shown which (sets of) objects are allowed to be linked with the update formula (5.12), based on the restriction in (5.11). Between the objects within the small homogeneous objects, edges are drawn indicating that these objects are allowed to be connected under the restriction. Then after some iterations, we see in the second (middle) panel in Figure 5.9 the intermediate results for MVPIN with a (maximum) shared nearest neighbors similarity of 9 or higher. Due to the shared nearest neighbors restriction in (5.11), we see that the small homogeneous clusters cannot be connected to each other, and that the noise objects start to merge into groups of objects, even though the noise objects have larger distances, than the distances between the two homogeneous clusters. The last panel (right) in Figure 5.9 displays the final results of MVPIN for $L = 3$. Already at a maximum similarity of five shared nearest neighbors, MVPIN detects two small homogeneous clusters and a large residual group of noise objects.

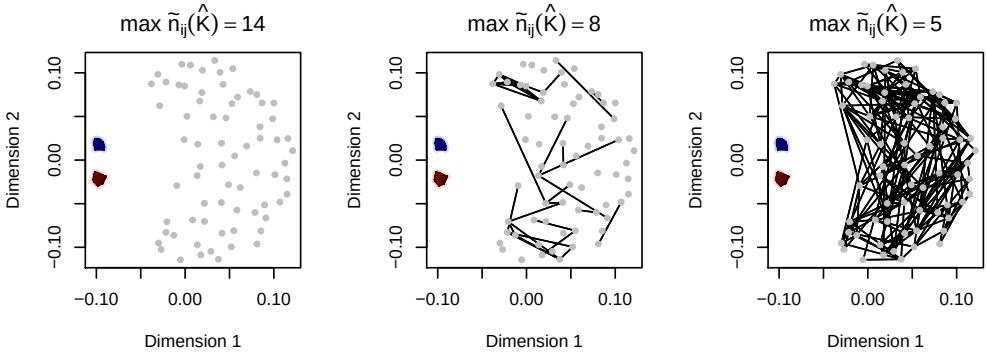

Figure 5.9: Searching for $L = 3$ when applying MVPIN to the COSA distances for an optimal value of K equal to 14, $\widehat{K} = 14$. It is of interest to note that the graphical display is shown on a two-dimensional MDS configuration of the COSA distances.

Repeating this particular simulation experiment on 100 simulated data sets, generated from the extra overlap model in Figure 5.6, resulted into a complete recovery of the clustering structure for 28 (out of 100) times for MVPIN, Ward's method was able to do so only for only 4 (out of the 100) data sets, and PAM only once. The clustering structure was never recovered by cutting an average linkage dendrogram at the specific height that would result into a partition of $L = 3$ groups.

5.4.2 Comparing MVPIN, WARD and PAM

Except the data from the extra overlapping attributes model, on all the earlier data examples no differences occurred between the performance of Ward's method and that of MVPIN on the optimized COSA distances for either COSA-KNN or COSA-λNN. For COSA distances of the simulated data sets from the previous chapters, MVPIN and Ward's method were able to recover the clustering structure perfectly, and performed equally well (or even better) than PAM, and cutting the average linkage dendrogram.

Real Data Sets

Results have been obtained for Ward and MVPIN applied to the optimized COSA distances for 12 oncological benchmark data sets. Apart from the ApoE3 data (Chapter 3) and the Breast cancer data (Chapter 4), the remaining oncological benchmark data sets have not been described yet. Note that the details for the data sets will be described in Chapter 6. For now, Table 5.1 shows the details of the 'true' clustering structure that was supposed to be recovered from the data sets.

Table 5.1: The benchmark data sets consist of a metabolomics data set (nr. 0) and microarray gene expression data sets (nrs. 1 - 10).

#	Data Name	L	$N(N_1 + \ldots + N_L)$	P	Source
0	ApoE3 Mice	2	38 (18 + 20)	1,550	Damien et al. (2004)
1	Brain	5	42 (10 + 10 + 10 + 4 + 8)	5,597	Pomeroy (02)
2	Breast	2	276 (183 + 93)	22,215	Wang et al. (05)
3	Colon	2	62 (22 + 40)	2,000	Alon et al. (99)
4	Leukemia	2	72 (47 + 25)	3,571	Golub et al. (99)
5	Lung1	2	181 (150 + 31)	12,533	Gordon et al. (02)
6	Lung2	2	203 (139 + 64)	12,600	Bhattacharjee et al. (01)
7	Lymphoma (DLBCL)	3	62 (42 + 9 + 11)	4,026	Alizadeh et al. (00)
8	Prostate	2	102 (50 + 52)	6,033	Singh et al. (02)
9	SRBCT	4	63 (23 + 8 + 12 + 20)	2,308	Kahn (01)
10	SuCancer	2	174 (83 + 91)	7,909	Su et al. (01)
11	SuCancer	2	174 (83 + 91)	7,909	Su et al. (01)
12	X-Perou	4	62 (8 + 7 + 11 + 36)	1,753	Perou et al.(00)

Rand Index Results

Instead of visualizing the assigned clustering structure in a dendrogram or an MDS configuration for each of the 12 benchmark data sets, we will use the Rand Index (Rand 1971) to show the performance of recovering of the clustering structure. A Rand Index (RI) of 1 indicates perfect retrieval of the ground truth clustering structure, and a Rand Index of 0 is an indication of anti-clustering, i.e., all pairs of objects that should belong to the same cluster are in different clusters and vice versa. The Rand Index results of the clustering algorithms for the 12 benchmark data sets are shown in Figure 5.14.

From the results on the benchmark data sets, there is no clear winner among MVPIN, Ward and PAM. The merit of MVPIN is mainly visible for the DLBCL data set with a positive difference of 0.246 on the Rand Index for cluster recovery. For all other data sets we see that MVPIN, Ward's method, and PAM have a very similar performance. It is of interest to note that PAM performs best with a Rand index difference 0.047 for the Colon data set, and worst for the DLBCL and Lung1 data sets.

We are aware that it is often argued that there are other indices in cluster validation research that make better use of the [0,1] interval to measure the similarity between clustering structures than the Rand Index (Romano et. al, 2016; Steinley, 2004), e.g. the adjusted Rand index (ARI; Hubert & Arabie, 1985), Normalized Mutual Information (NMI; Strehl & Gosh, 2002), and the adjusted Mutual Information (AMI; Vin, Eps, & Bailey, 2010). However, using either of the other measures (ARI, NMI, AMI) resulted in the same conclusions for the comparison of MVPIN, WARD and PAM. The reason we used the Rand Index will become clear in Chapter 6, where we compare the current chapter's results of MVPIN with those from other studies where the Rand Index was used (Arias-Castro & Pu, 2017; Deng et al., 2011).

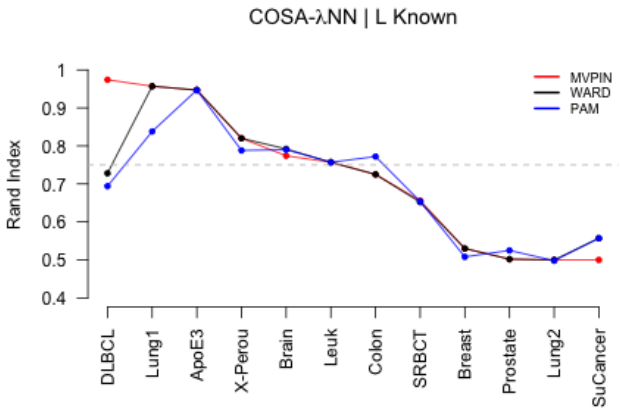

Figure 5.10: The Rand Index computed over the resulting partitions from MVPIN (red), Ward (black) and PAM (blue). The algorithms have been applied to the optimized COSA-λNN distances for each data set. The horizontal dashed grey line indicates a Rand Index of 0.75, and the data sets are ordered corresponding to the results of MVPIN.

5.5 Gap statistic: A strategy for when L is unknown

So far, we assumed that L, the optimal number of clusters in the data, was known. In practice this is seldomly the case. As will be shortly discussed in Chapter 6, strategies employed to choose the optimal value of L are elaborately explored, but remains an unsolved ongoing topic of research (Hancer et al., 2017; Arbelaitz et al. 2013; Lee & Olafsson, 2013; Tibshirani & Walther, 2005; Dudoit & Fridlyand, 2002). In this Section, we will consider L as a tuning parameter that needs to be optimized for PAM, Ward's methods and MVPIN. For PAM we will follow the example set in Kaufman and Rousseeuw (1990) to select the value of L that corresponds with the highest average silhouette over the objects, i.e. the average based on equation (5.3). For Ward's method and MVPIN we will rely on an adjusted version of the Gap statistic procedure (Tibshirani et al., 2001), described in Chapter 2.

Ward's method is motivated by the minimization of a criterion, i.e. any objective function that can evaluate the loss of information. By replacing the squared Euclidean distances with the squared COSA distance, the criterion of Ward's minimum variance method, referred to as the objective function in Ward (1963, p. 237), or Wishart (1969, p. 166) becomes

$$Q_{Ward}(L) = \frac{1}{2}\sum_{l=1}^{L}\sum_{i=1}^{N}\sum_{j=1}^{N} c_{il}c_{jl}D_{ij}^{2}[\mathbf{W}], \tag{5.15}$$

where $c_{il} = 1$ when $i \in C_l$. We find that due to the left-skewed density of the COSA distances, this criterion has a polynomial growth rate for decreasing L. For Ward's method we will apply the Gap statistic procedure to the criterion from equation (5.15) to determine the number of clusters. Our findings so far show that the Gap statistic

procedure applied to the criterion directly would always result in favoring the smallest number of groups, while the GAP statistic procedure applied to the logged criterion has the tendency to favor too large L. Best results were obtained when the GAP statistic was applied to the square root of the criterion, i.e.

$$\text{GAP}_{\surd} = E_{Q^\circ}\left[\sqrt{Q^\circ_{Ward}(L)}\right] - \sqrt{Q_{Ward}(L)}. \tag{5.16}$$

Here, E_{Q° denotes the expectation of the Ward criterion of COSA distances that were optimized for comparable data sets in which only noise was present. Last, to make sure all COSA distances of the permuted data sets as well as the real data set were comparable, we normalized the COSA distances to have sum of squares equal to N.

MVPIN is also an approximate heuristic directed by the rate of change of the criterion in(5.15), although with a restriction on the maximum number of shared nearest neighbors, resulting from (5.11). Therefore, the criterion for MVPIN is defined as the sum of rates of changes:

$$Q_{MVPIN}\left(L \,|\, \widehat{K}\right) = \frac{1}{N-L}\sum_{m=N}^{L+1} \min\left\{\delta^2\left(l^* \cup l'^*\,,\, l''^* \,|\, \Delta_{m\times m}\right)\right\}. \tag{5.17}$$

The criterion in (5.17), is the average of the minimum distances obtained from each of the $N-L$ restricted Ward distance matrices $(\{\Delta_{m\times m}\}_{m=N}^{L})$ that are recursively reduced from size $N\times N$ to $L\times L$, via the update formula in equation (5.12). Note that the procedure to obtain the estimate $\widehat{K}$ is based on equation (5.10) described in Section 5.3.1.

The Gap statistic procedure applied to the criterion for MVPIN, showed consistent results for the Gap, $\text{Gap}_{\log}$, and $\text{Gap}_{\surd}$, and for both the COSA-λNN or COSA-KNN distances applied to all data examples in this chapter. See Figure 5.11 for a visualization of the gap statistics procedure for MVPIN applied to the COSA distances of the simulation example from the extra overlapping attributes model. Applying the Gap statistic to the COSA distances of the simulated data example from Figure 5.6, and to the COSA distances of $B = 25$ permuted versions of this data set, resulted in the correct selection of L, i.e. $L = 3$ clusters.

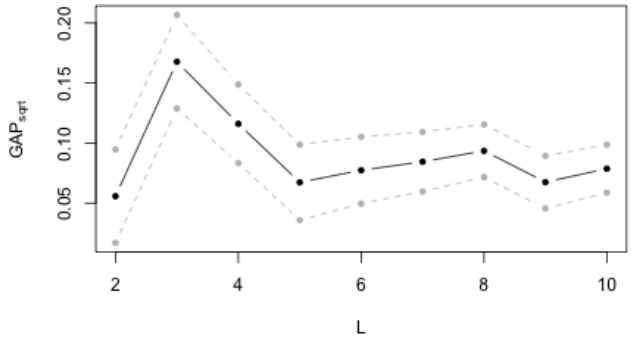

Figure 5.11: For MVPIN we obtain $L = 3$ for the value with the maximum gap statistic. The dashed grey lines are the 1 standard deviation upper and lower bounds of the Gap statistic.

5.5.1 Benchmark Data Sets Results

When applying the Gap statistic procedure in MVPIN and Ward, and the average silhouette for PAM, to determine the number of clusters L, the performances of the three methods on the COSA-λNN distances, are less similar. As can be seen in Figure 5.12, MVPIN is five (out of twelve) times a clear winner in the recovery of the ground truth clustering structure. For the results we used $B = 25$ permuted 'noise' data sets over which the average criterion could be computed to obtain an estimate of the expected criterion under noise only, i.e. $E_{Q^\circ}\left[\sqrt{Q^\circ_{Ward}(L)}\right]$.

Noteworthy is the bad performance of Ward's method on the Lung1 data set. Here, the max Gap statistic for the criterion for Ward's method occurs at $L = 5$ clusters, where the true clustering structure has $L = 2$. Apart from the Lung1 data set, we see that the performance of Ward's method and PAM on the COSA-λNN distances is very similar with a slight advantage for PAM.

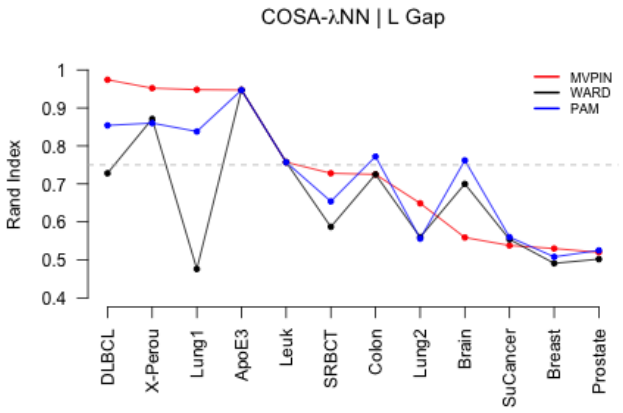

Figure 5.12: The Rand Index computed over the resulting partition from the gap-statistic optimized versions of MVPIN (red), Ward (black) and PAM (blue). The horizontal dashed grey line indicates a Rand Index of 0.75, and the data sets are ordered corresponding to the results of MVPIN, optimized for L.

5.6 Conclusion and Discussion

MVPIN is a clustering algorithm designed for using COSA distances to partition objects in a number of L groups. Compared to competing clustering algorithms such as PAM, hierarchical clustering with average linkage or Ward's method, we find that MVPIN is better at detecting small homogeneous clusters that are similar to each other, because of a similar clustering pattern on an overlapping subset of overlapping attributes. In a comparative study we have shown in a first examination that when the number of clusters is unknown, MVPIN in combination with the Gap statistic shows better results than those obtained by PAM and Ward's method on the (optimized) COSA-λNN distances for the 12 benchmark data sets. Assuming L to be known beforehand, the performances of the clustering algorithms are more similar to one another. Although not shown, similar, but weaker conclusions could be made for MVPIN, PAM and Ward's method on the optimized COSA-KNN distances (Section 5.7.1).

As of yet, we have only demonstrated our empirical experience of a first examination of MVPIN. Although we have provided an advice and a conceptual explanation for finding a good value of K, more research is needed to see under what conditions this advice holds or could break down. For future research it could be of interest to see how MVPIN would perform when applied directly to the neighborhoods that have been used for the computation of the attribute dispersions in either COSA-KNN or COSA-λNN. Moreover, it may be interesting to search for a data-driven lower threshold of the shared number of nearest neighbors from which the algorithmic steps in MVPIN perhaps already sooner could switch to the unrestricted versions of Ward's method.

There are more algorithms that can find a partition for a set of objects while using a distance matrix as input. For future research it could be interesting to compare the performance of MVPIN with the clustering algorithms PAMSIL (Van der Laan et al., 2003) and MiniDisconnect (Lee & Olafsson , 2011). We may think that PAMSIL could be a good competitor since it was designed to find small clusters in biological contexts. However, PAMSIL does not take into account density properties of the small clusters, e.g. a within-cluster shared nearest neighbors index, making it more sensitive to small groups of objects that are simply clusters on sampling fluctuations. MiniDisconnect, however, focuses on minimizing a concept of disconnectivity between clusters that is based on a mutual shared nearest neighbors measure as defined in Chidanda Gowda and Krishna (1978). Moreover, MiniDisconnect assumes L to be unknown, and learns by itself what the optimal value of L needs to be.

We did experiment with the implicit medoid based algorithm 'fast search and find of density peaks' by Laio and Rodriguez (2014). The algorithm did not seem to be capable to deal with the non-metric COSA dissimilarities, and showed a performance that was similar to that of cutting an average linkage dendrogram at a certain height to obtain L groups. Moreover, the 'cut-off' distance tuning parameter in the fast search for density peaks algorithm does not allow for a simple tuning strategy.

5.7 Appendix

5.7.1 MVPIN Results on the COSA-KNN Distances

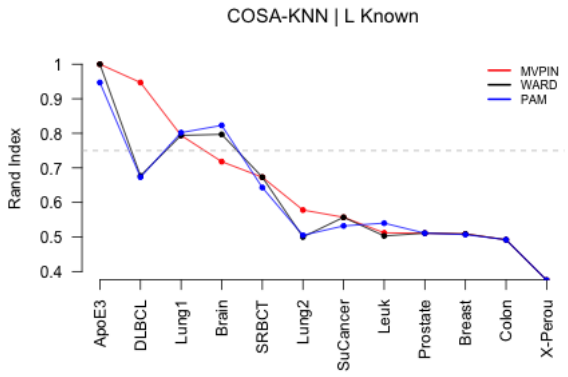

Figure 5.13: The Rand Index computed over the resulting partitions from MVPIN (red), Ward (black) and PAM (blue) applied to the optimized COSA-KNN distances for each data set. The dashed grey line indicates a Rand Index of 0.75.

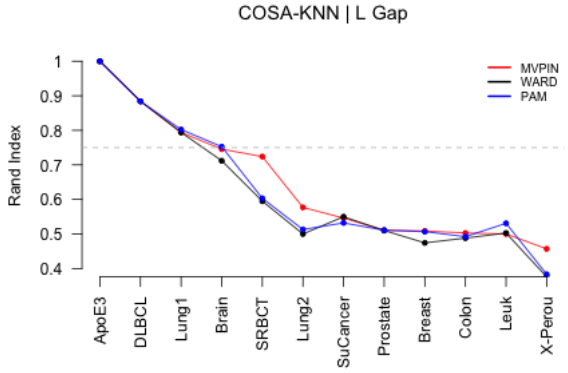

Figure 5.14: The Rand Index computed over the resulting partitions from MVPIN (red), Ward (black) and PAM (blue), with a Gap optimized L, applied to the optimized COSA-KNN distances. The dashed grey line indicates a Rand index of 0.75.