The handle http://hdl.handle.net/1887/74690 holds various files of this Leiden University dissertation.

Author: Kampert, M.M.D.
Title: Improved strategies for distance based clustering of objects on subsets of attributes in high-dimensional data
Issue Date: 2019-07-03
Chapter 3

A Robust and Tuned COSA-KNN

The implementation of the COSA-KNN algorithm might not have enough power with its default values of the tuning parameters $\lambda$ and $K$, being $\lambda = 0.2$ and $K = \sqrt{N}$. These values may be too far off from the optimal tuning parameter values that can retrieve an underlying the clustering structure. In such a situation, we need a search strategy that can provide good candidate values for the tuning parameters.

To increase the number of good candidate values, we propose in Section 3.1 to apply a robust version of COSA-KNN on a data set. In the algorithm of this robust version of COSA-KNN, a criterion is minimized that is based on median-based attribute dispersions, instead of mean-based attribute dispersions. The result is that in the robust version of COSA-KNN the attribute weights are also computed based on median-based attribute dispersions, instead of mean-based attribute dispersions, rendering the selection of the subset of attributes to be less sensitive to the noise in the data.

We propose a criterion-based search strategy that can find successful values of the tuning parameters. We describe in Section 3.2 the behavior of both the original COSA-KNN criterion, and the criterion for the robust COSA-KNN, as a function of the tuning parameters. Compared to the original criterion, the robust criterion values show a systematic 'zigzag' behavior between the odd and even values for $K$. This behavior diminishes for larger value of $K$, but is amplified by the value for $\lambda$. The higher $\lambda$, the more pronounced the peculiarity.

In section 3.3 we describe the criterion-based search strategy: the Gap statistic procedure, developed by Tibshirani, Walther, and Hastie (2001). The Gap statistic procedure is a permutation approach in which we select that particular combination of values for the tuning parameters that provides the largest gap between the value of the robust criterion obtained for a particular data set, and (an approximation of) the expected value obtained for a null-reference model, i.e. a comparable data set with no clustering structure. Two versions of the Gap statistic procedure will be described; the
original procedure where the natural logarithm is taken over the robust COSA-KNN criterion of each data set, and the version where the criterion is not transformed. Since the Gap statistic was originally developed to choose the number of clusters in $K$-means type algorithms, it is not self-evident that it would work for $\lambda$ and $K$ in COSA-KNN.

We will combine the Gap statistic procedure with the robust version of COSA-KNN and a pre-specified grid of values for $\lambda$ and $K$. In Section 3.4 we apply the Gap statistic to a real data example, and show that with the resulting optimal tuning parameters a sharper representation of the clustering structure is obtained, i.e., the structure exhibits a clearer separation between the different clusters. In this data example, the consequences of the zigzag peculiarity of the robust criterion will also be discussed. Section 3.4.2 shows the effectiveness of the Gap statistic on a simulated data example that consists of a subtle clustering structure that would not have been found with the default settings of (robust) COSA-KNN. Section 3.4.3 shows the satisfactory behavior of the Gap statistic procedure and COSA-KNN, when applied on a data example that represents noise only. The discussion of the chapter is provided in Section 3.5.

### 3.1 Robust versus Non-Robust Attribute Dispersions

The goal of COSA-KNN is to obtain distances between objects that can represent a hard-to-detect clustering structure present in a data set ($X$) of $N$ objects by $P$ attributes. The clustering structure is hard to detect when many of these $P$ attributes do not contribute to the clustering, and the small number of attributes that do contribute to the clustering, may be cluster-specific, i.e., different for each cluster. In Chapter 1 we described a typical data model of such a clustering structure, referred to as the prototype model. To facilitate reading, we also display this specific prototype model in Figure 3.1.

The COSA-KNN distances are defined as

$$D_{ij}[W] = \sum_{k=1}^{P} \max(w_{ik}, w_{jk})d_{ijk},$$  \hspace{1cm} (3.1)

where each $d_{ijk}$ is defined as

$$d_{ijk} = \frac{|x_{ik} - x_{jk}|}{IQR(\{x_{ik}\}_{i=1}^{N})/1.35},$$  \hspace{1cm} (3.2)

and the attribute weights are defined as

$$w_{ik} = \frac{\exp\left(-\frac{S_{ik}}{\lambda}\right)}{\sum_{k=1}^{P} \exp\left(-\frac{S_{ik}}{\lambda}\right)},$$  \hspace{1cm} (3.3)
Figure 3.1: A Monte Carlo model for 100 objects with 1,000 attributes (not all are shown due to \( P \gg N \)). There are two small 15-object groups (red and blue), clustering each on 30 attributes out of 1000 attributes, with partial overlap, and nested within an unclustered background of 70 objects (grey). After generating a data set from this model, each attribute is standardized to have zero mean and unit variance.

where \( S_{ik} \) is the arithmetic mean of the distances on attribute \( k \), between object \( i \) and its \( K \) nearest neighbors, i.e.,

\[
S_{ik} = \frac{\sum_{j=1}^{N} c_{ij} d_{ijk}}{K}, \tag{3.4}
\]

with \( c_{ij} \) equal to 1 when \( j \) belongs to the \( K \) nearest neighbors of \( i \), and 0 otherwise. Although it is clear that \( S_{ik} \) is influenced by \( K \) in equation (3.4), the value for \( S_{ik} \) is also implicitly influenced by the value of \( \lambda \), since the \( K \) nearest neighbors for object \( i \) are found based on the COSA distance \( D_{ij}[W] \) from an earlier iteration in the algorithm COSA-KNN, and \( D_{ij}[W] \) in (3.7) consists of the attribute weights that involve the value of \( \lambda \).

Instead of a mean of the distances on attribute \( k \), we can also use the median. Using medians as an alternative measure of central tendency eliminates effects of so-called outliers, making the overall procedure more robust as compared to the use of the mean-based version. In this chapter we will refer to median-based attribute dispersion as the robust attribute dispersions, and to the mean-based attribute dispersions as non-robust attribute dispersions. Thus, instead of the non-robust attribute dispersion, we can compute the robust attribute dispersion as

\[
\tilde{S}_{ik} = \text{median} \left( \{ d_{ijk} \}_{j \in KNN(i)} \right), \tag{3.5}
\]
such that the definition of a median-based (robust) attribute weight becomes

\[ \tilde{w}_{ik} = \exp \left( -\frac{\tilde{s}_{ik}}{\lambda} \right) \sum_{p=1}^{P} \exp \left( -\frac{\tilde{s}_{ip}}{\lambda} \right), \]  

(3.6)

which leads to the median-based (robust) COSA distance, i.e.,

\[ D_{ij}[\tilde{W}] = \sum_{k=1}^{P} \max(\tilde{w}_{ik}, \tilde{w}_{jk})d_{ijk}. \]  

(3.7)

Although Friedman and Meulman (2004) advised to use this robust version for the attribute dispersion, no explanation, nor any example of comparison was given to demonstrate its effectiveness.

### 3.1.1 Demonstrating a Prototype Data Example

When we apply COSA-KNN to a data set generated from the prototype model (Figure 3.1), then the results are less dependent on \( K \) and \( \lambda \) when the robust attribute dispersions (equation 3.5) are used, compared to the use of the mean-based attribute dispersions. We depicted in Figure 3.2 an example of dendrograms of both versions of the COSA distances for three different combinations of values for \( \lambda \) and \( K \). For this specific data example, we define a dendrogram to be successful when we can cut it into 70 singletons and two groups of each 15 objects.

Although the successful results of the non-robust COSA-KNN can show a sharper representation of the clustering structure, i.e., more distinct clusters, the robust version of COSA-KNN has more combinations for the values of \( K \) and \( \lambda \) that lead to successful results. In Figure 3.3, we show that for combinations of \( 1 \leq K \leq 50 \) and \( 0.01 \leq \lambda \leq 1 \), the true clustering was revealed successfully for 69 percent of the times in a dendrogram of the distances for the robust version of COSA-KNN, while for the non-robust version only 31 percent of the same combinations for \( K \) and \( \lambda \) produced successful dendrograms. Thus, when using the non-robust attribute dispersion in equation (3.4), it is more likely to miss the true clustering structure in the data with COSA-KNN.

Another important difference between both versions of the algorithm occurs in the first iterations. Empirical findings, displayed in Figure 3.4, show that in general the robust version of COSA-KNN needs fewer iterations to retrieve a good clustering structure. Both the robust and non-robust version of the attribute weights are computed based on neighborhoods. However, at the starting iterations we see that the neighborhoods of the clustered objects consist of a large number of ‘incorrect’ neighbors, where incorrect neighbors are defined as objects that are from a different cluster. We conjecture that the robust version of COSA-KNN is less affected by these incorrect neighbors, when compared to the non-robust version of COSA-KNN.
Figure 3.2: Average linkage dendrograms. The dendrograms in the first row are visualizations of the COSA-KNN distances using robust attribute dispersions. The dendrograms in the second row use non-robust attribute dispersions.
Figure 3.3: The grid of $K$ (vertical axis) versus $\lambda$ (horizontal axis) combinations for the robust and non-robust COSA-KNN: the grey points indicate unsuccessful dendrograms for both the robust and the non-robust versions, the yellow points indicate successful dendrograms for only the robust versions, dark red points indicate successful dendrograms for only the non-robust version, and the orange points indicate the successful dendrograms for both versions.
Figure 3.4: The dendrograms of the robust COSA-KNN distances converge faster towards a correct representation of the clustering structure than the non-robust COSA-KNN distances for $K = 14$ and $\lambda = 0.15$ at iterations 0, 1, and 2.

Attribute Distance Distributions

When an incorrect neighbor has a different clustering mechanism on the signal attributes in the particular neighborhood of an object, then it likely enlarges the dispersion for each of these signal attributes on which the incorrect neighbor clusters differently. Suppose we use the prototype model from Figure 3.1 then an attribute distance from equation (3.2) is folded-normally distributed (Leone, Nelson, & Not-
tingham, 1961):

\[
d_{ijk} \sim \begin{cases} 
1 \cdot \frac{1}{\sqrt{2\pi\sigma_{ijk}^2}} e^{-\frac{(d_{ijk} - \mu_{ijk})^2}{2\sigma_{ijk}^2}} + \\
1 \cdot \frac{1}{\sqrt{2\pi\sigma_{ijk}^2}} e^{-\frac{(d_{ijk} + \mu_{ijk})^2}{2\sigma_{ijk}^2}} \end{cases}.
\]

(3.8)

Here, \(\mu_{ijk}\) and \(\sigma_{ijk}^2\) are the mean and variance of the normal distribution that is to be folded. When objects \(i\) and \(j\) belong to the same cluster, then for a signal attribute from an unique subset we have \(\mu_{ijk} = 0\) and \(\sigma_{ijk}^2 \approx (2 \cdot 0.2^2)/(0.15 \cdot 0.2^2 + 0.85 \cdot 1^2)\). For the same attribute, the distance between a red cluster object and a noise object, we have \(\mu_{ijk} = 1.5/(0.15 \cdot 0.2^2 + 0.85 \cdot 1^2)\) and \(\sigma_{ijk}^2 = (1^2 + 0.2^2)/(0.15 \cdot 0.2^2 + 0.85 \cdot 1^2)\), resulting in a distribution with a higher mean and larger variance, as is depicted in Figure 3.5. Indeed, from Figure 3.5 we can also obtain that the attribute distance between an object from the red cluster with a noise-object (wrong neighbor) is most likely to result in the largest (outlying) attribute distance. While the non-robust attribute dispersion, \(S_{ik}\) in (3.4), is sensitive to these larger attribute distances, the robust attribute dispersion, \(\tilde{S}_{ik}\) in (3.5), is (to some extent) protected (\(S_{ik}\) in (3.4)) due to the properties of the median.

Figure 3.5: The theoretical probability density functions of the attribute distances between correct neighbors (red), correct and incorrect neighbors (dashed grey), and between noise objects (solid grey) for a signal attribute from a unique subset.

The Behavior of \(\tilde{S}_{ik}\) and \(S_{ik}\) in COSA-KNN

Shortly after the first iteration, the robust version of COSA-KNN often already performs better than the non-robust version. Both versions of the algorithm start with
equal attribute weights, and hence the same COSA distances. We will refer to these attribute weights and COSA distances as the results of ‘iteration 0’. The dendrograms in the top panels of Figure 3.4 visualize these distances. In these identical dendrograms at iteration 0, the objects from the same cluster (red or blue color) seem closer to each other than objects that do not belong to the same cluster. However, the complete clustering structure from this specific data set is not yet revealed. Based on these distances, we find a first estimate of the neighborhoods of each the objects and compute the attribute weights.

From the results of iteration 0 it is very likely that each object will have a neighborhood that does not yet contain the ‘true’ nearest neighbors. Therefore, the effect of outliers on either the robust and non-robust attribute dispersions already becomes visible for the different COSA-KNN versions in iteration 1. Figure 3.6 displays this difference at iteration 1 for the attribute dispersions of 15 red cluster objects from the data that was generated from the prototype model (Figure 3.1), i.e. \(i = i' \in \{1, \ldots, 15\}\). For these objects from the red cluster are known to have their own unique subset of attributes for \(k = k' \in \{1, \ldots, 15\}\) and a shared subset of attributes with the objects from the blue cluster for \(k = k' \in \{16, \ldots, 30\}\), all other attributes are considered as noise. Figure 3.6 depicts for each attribute the average over the 15 objects for either versions of the attribute weights (top panel), as well as the two versions of the attribute weights (lower panel). Figure 3.6 depicts higher averages for the dispersions of the unique subset of attributes, than for the shared subset attributes. Since the generating process of attribute values in the shared subset is the same for objects in the red and the blue cluster, the attribute dispersions can only be distorted by the remaining noise objects. However, the averages of the attribute dispersions on the unique subset, can be distorted by the objects from the blue cluster as well. The more objects there are to distort the neighborhood, the more the non-robust version of the attribute dispersions (and weights) are affected. The averages of the dispersions (and the weights) for the unique subset of attribute are higher (and lower), than those of the robust versions. Since it is to be expected that neighborhoods of the objects are more distorted in the beginning iterations, we see that the robust COSA-KNN converges faster and produces more often successful distances, when compared to the non-robust version.
Figure 3.6: The top panel depicts the robust (black) and non-robust (grey) average attribute weights of the red cluster objects for attribute $k' = 1 \ldots 75$. The lower panel depicts the corresponding robust and non-robust average attribute dispersions, the average is taken over the neighborhoods from each object in the red cluster.

### 3.1.2 Data Example 2: Empowered Results

The robust version of COSA-KNN may be the only option in situations where the clustering structure is more subtle. Figure 3.7 depicts an example of a generating model for such a data set, referred to as the subtle structure model. The clustering mechanism in this model is very similar to that of the prototype model. While $N$ remains 100 objects, the number of attributes is set to $P = 10,000$ (instead of a 1,000). We still have two clusters, but this time each cluster consists of 20 objects (instead of 15). The sizes of the subsets of signal attributes remain 30 attributes, and the overlap remains 15 attributes. The process for data generation remains the same to that of the prototype model for the noise attribute values, as well as for the signal attribute values in the unique and shared subsets. After the data is generated, each attribute is standardized to have a zero mean and unit variance.
Figure 3.7: A subtle structure data model for dataset $X$ with 100 objects and 10,000 attributes (of which only 60 are shown). There are two groups of 20-objects (red and blue) each clustering on a subset of 30 attributes. After generating a data set from this model, each attribute is standardized to have zero mean and unit variance.

Independent of the choice for $\lambda$ or $K$, the non-robust version of COSA-KNN is not able to pick up the clustering structure in the domain $1 \leq K \leq 50$ with any of the values for

$$\lambda \in \{0.01, 0.05, 0.075, 0.1, 0.125, 0.15, 0.2, 0.25, 0.3, 0.4, 0.5, ..., 1\}.$$  

However, provided we choose a suitable combination of $K$ and $\lambda$, the robust version can result in distances that yield a successful dendrogram, as is shown in Figure 3.8. While both versions of the COSA-KNN manage to distinguish the noise objects from the clustered objects, only the robust COSA-KNN version manages to distinguish the two clusters.
A reason why the non-robust COSA-KNN cannot retrieve the clustering structure is its sensitivity to nestedness of the clustering. Because the non-robust version is slower in finding the clustering structure in the first iterations, the chances are also higher that objects from different clusters with overlapping subsets of attributes, have the tendency to stay longer in each others neighborhoods during the iterations; see Figure 3.9. Since the attribute dispersion from equation (3.4) is based on the mean of all the attribute distances within the neighborhood, Figure 3.6 depicts that the dispersions become small (and the weights become high) for the shared subset of attributes. Since the mean-based attribute dispersions cannot be ‘robustly’ corrected for incorrect neighbors, the cluster-unique attribute dispersions within the neighborhood of an object remain overestimated. This overestimation is not only due to the distortion of the noise objects, but also due the overrepresentation of the objects from another ‘neighbor cluster’, rendering attribute weights with low values for the cluster-unique subset.

When there are too many attributes in the data set (or too many objects), this
Figure 3.9: Dendrograms for the robust distances converge faster towards a correct representation, as compared to the non-robust COSA-KNN distances for $K = 19$ and $\lambda = 0.10$. A particular distortion process may even affect the median-based attribute dispersions. As can be seen in Figure 3.9, for the specific data set from the subtle structure model, neither non-robust version of COSA-KNN manages to retrieve the clustering structure for the usual values of $\lambda = 0.2$ and $K = \sqrt{N}$. Thus, a strategy is needed to find optimal values for $\lambda$ and $K$.

3.2 The Criterion as a Function of $\lambda$ and $K$

To select the values of $\lambda$ and $K$, we propose to apply a permutation approach that is based on the criterion of COSA-KNN, and is referred to as the Gap statistic (Tibshirani et al., 2001). For a successful application of this approach, a good understanding
of the properties of the COSA-KNN criterion regarding its tuning parameters is required. The definition of the original COSA-KNN criterion is

\[
Q(\lambda, K) = \sum_{i=1}^{N} \left\{ \sum_{k=1}^{P} w_{ik} S_{ik} + \lambda \left( \sum_{k=1}^{P} w_{ik} \log(w_{ik}) + \log(P) \right) \right\}.
\] (3.9)

In this criterion the value for each \(d_{ijk}\) is fixed, such that the values for each \(w_{ik}\) in (3.3) and each \(S_{ik}\) in (3.4), follow from the values for \(\lambda\) and \(K\). Since \(Q(\lambda, K)\) is based on the non-robust attribute weights, it is referred to as the non-robust criterion. Figure 3.10 displays the behavior of the values of this non-robust criterion for different values of \(\lambda\) and \(K\), given the generated data from the prototype model.

\[\text{Figure 3.10: The non-robust COSA-KNN criterion values on a prototype data set sampled from the COSA model with } N = 100 \text{ objects and } P = 1000 \text{ attributes. We ran COSA-KNN on a grid with } K \in \{1, 2, 3, ..., 50\} \text{ and } \lambda \in \{0.01, 0.025, 0.05, 0.075, ..., 0.3, 0.35, 0.4, 0.5, ..., 1.0\}. \text{ The arrows at the axes in the plot indicate the direction in which either the value of } \lambda \text{ or the value of } K \text{ increases.} \]

Figure 3.10 displays that in general higher values of either \(\lambda\) or \(K\) result in an increased value for the criterion, which is a concave function. While keeping \(K\) fixed, a higher value for \(\lambda\) renders the values of the attribute weights within each neighborhood to become more equal, which results in a stronger influence of the larger attribute dispersions, resulting in an increase in the criterion value (the reverse is true for the smaller attribute dispersions). However, changing the value for \(\lambda\) may also render the attribute weights to indicate different subsets of attributes, resulting in different COSA distances from which different neighborhoods are extracted. Thus, the monotone pattern in the relationship between \(\lambda\) and \(Q(\lambda|K)\) can be interrupted.

While keeping the value for \(\lambda\) fixed, a similar monotone pattern exists between the criterion and \(K\). With a larger \(K\), each object \(i\) will obtain more nearest neighbors. Since the distance between object \(i\) and the \(K^{th}\) nearest neighbor is lower than (or equal to) the distance between object \(i\) and the \((K + 1)^{th}\) nearest neighbor, we may expect that the weighted sum of the attribute dispersions also increases for \(K + 1\) in
the criterion in (3.9). Note, however, when we add the \((K + 1)^{th}\) neighbor objects to each corresponding neighborhood, the distribution of the attribute dispersions within each neighborhood may change, which may change the distribution of the attribute weights. When this change is the difference between detecting a clustering structure or not, then it may happen that the criterion actually drops for a larger value of \(K\). Random fluctuations can also contribute to violations of monotonicity at a fixed low value for \(\lambda\) and the lower values of \(K\). The lower the value of \(\lambda\), the more the attribute weights diverge from equal weights, and the more sensitive they are for random fluctuations of the attribute dispersions.

### 3.2.1 The Robust Criterion and \(K\): a ZigZag Tendency

The robust version of COSA-KNN contains the robust attribute dispersions, \(\tilde{S}_{ik}\) in (3.5) and the robust attribute weights, \(\tilde{w}_{ik}\) in (3.6). The criterion is defined as

\[
\tilde{Q}(\lambda, K) = \sum_{i=1}^{N} \left\{ \sum_{k=1}^{P} \tilde{w}_{ik} \tilde{S}_{ik} + \lambda \left( \sum_{k=1}^{P} \log(\tilde{w}_{ik}) \tilde{w}_{ik} + \lambda \log(P) \right) \right\},
\]

(3.10)

The properties that were described for the non-robust criterion in equation (3.9), can also hold in general for this robust criterion, but only as long as the values for \(K\) in the criterion are only even, or only odd.

The robust criterion has a peculiar behavior over the value of \(K\). Since the peculiar behavior is directly related to the definition of the robust attribute dispersion in (3.5), it is useful to first clarify the direct relationship between \(\tilde{S}_{ik}\) and \(\tilde{Q}(\lambda, K)\), without the involvement of the robust attribute weights \(\tilde{w}_{ik}\) in (3.6).

Based on the definition of equation (3.6), we can rewrite the robust criterion in equation (3.10) as,

\[
\tilde{Q}(\lambda, K) = \sum_{i=1}^{N} -\lambda \log \left\{ \frac{1}{P} \sum_{k=1}^{P} \exp \left( -\frac{\tilde{S}_{ik}}{\lambda} \right) \right\},
\]

(3.11)

This rewritten formulation is a sum of \(N\) ‘softmin’ attribute dispersions, that can be continuously deformed over \(\lambda\) to a sum of mean attribute dispersions, or to a sum of the minimum attribute dispersions. For each neighborhood, the definition of the softmin over \(P\) attribute dispersions is

\[
\tilde{Q}(\lambda, K)_i = -\lambda \log \left\{ \frac{1}{P} \sum_{k=1}^{P} \exp \left( -\frac{\tilde{S}_{ik}}{\lambda} \right) \right\},
\]

(3.12)

For \(\lambda \to 0\), we can obtain the minimum over the \(P\) attribute dispersions, i.e.,

\[
\lim_{\lambda \to 0} \tilde{Q}(\lambda, K)_i = \min_k \left( \tilde{S}_{ik} \right).
\]

(3.13)

For \(\lambda \to \infty\), we obtain the arithmetic mean of the attribute dispersions, i.e.,

\[
\lim_{\lambda \to \infty} \tilde{Q}(\lambda, K)_i = \frac{1}{P} \sum_{k=1}^{P} \tilde{S}_{ik}.
\]

(3.14)
The higher $\lambda$, the stronger the influence of the large attribute dispersions (and vice versa). Therefore, the higher $\lambda$, the higher the value of the criterion. (Note that one can arrive at equations (3.11), (3.13), and (3.14), with derivations that are similar to those used in Sections 2.7.2 and 2.7.3 from the Appendix of Chapter 2)

In general, for higher values of $K$, we can also expect larger attribute dispersions, and thus a larger value of the robust criterion. However, as is displayed in Figure 3.11, the behavior of the criterion over the values of $K$ shows a particular zigzag behavior between the odd and even values for $K$. This particular zigzag pattern seems to be stronger present in the lower values of $K$, and is amplified for larger values of $\lambda$.

![Figure 3.11: The behavior of $\tilde{Q}(\lambda,K)$ as a function of $K$ for $\lambda$ equal to 0.01, 0.1, and 1. The blue triangles represent odd values for $K$, the red squares represent the even values for $K$. These results were obtained from a data set of $N = 100$ objects by $P = 1000$ attributes data that consisted of noise only: $i.i.d. \sim N(0,1)$. After generation of the data, the attributes were standardized to have zero-mean and unit-variance.](image)

While keeping $\lambda$ fixed, we not only see that the criterion value is lower for an odd value for $K$, compared to $K + 1$, but the criterion value for the odd value for $K$ is also lower than, or approximately equal to, the criterion value based on $K - 1$. This zigzag pattern can be formulated as

$$\tilde{Q}(\lambda,K - 1) \geq \tilde{Q}(\lambda,K) < \tilde{Q}(\lambda,K + 1) \geq \tilde{Q}(\lambda,K + 2) < \ldots,$$

for odd $K$.\hspace{1cm}(3.15)

Thus, for an odd number of neighbors $K$, the sum of the (inverse-exponential) means of the robust attribute dispersions is lower than the sum for $K + 1$ neighbors, but also lower than, or approximately equal to, when based on $K - 1$ neighbors.

From the perspective of a truncated mean (or trimmed mean), we can explain the difference between an odd value of $K$, and the even value $K - 1$, for a median-based attribute dispersion. We have seen in Section 3.1.1 (Figure 3.6) that the mean-based (non-robust) attribute dispersion is more sensitive to the higher attribute distances than the median-based (robust) attribute dispersion. This robustness property can be generalized to attribute dispersions that are defined by the truncated mean of the attribute distances, where first an equal amount at the high and low end of the attribute distances is discarded, and then the mean computed over the remaining
attribute distances. The higher the percentage of attribute distances that are discarded, the more robust the truncated mean is to outlying large attribute distances. Note that the attribute distance is bounded by zero, and that there are less (if any) small outlying distances to be expected.

The median-based attribute dispersion, $\tilde{S}_{ik}$ in (3.5), is a special case of the truncated mean-based attribute dispersion. In particular, for $K$ odd, the attribute dispersion $\tilde{S}_{ik}$ is the fully truncated mean of the attribute distances, i.e. only the middle attribute distance remains. For $K$ even, $\tilde{S}_{ik}$ is the truncated mean where $K/2 − 1$ of the smallest attribute distances, and $K/2 − 1$ of the largest attribute distances are discarded. While in general for values of $K$, a higher proportion of the attribute distances is discarded, the proportion for an even number of neighbors, defined as $K + 1$, is always smaller than the discarded proportion of attributes for odd $K$, resulting in another zigzag pattern, i.e.,

$$\frac{(K − 1) − 2}{(K − 1)} < \frac{K − 1}{K} > \frac{(K + 1) − 2}{(K + 1)} < \frac{(K + 2) − 1}{(K + 2)} > \ldots, \text{ for odd } K.$$  (3.16)

Although for large values of $K$ these inequalities may be negligible, the smaller the value of $K$ the larger these inequalities become. For odd $K$, the proportion of extreme attribute distances that is discarded, is larger than the proportion that is discarded for the $K − 1$ and $K + 1$ even numbers of neighbors. It is more likely that the robust attribute dispersion for odd $K$ is smaller, than the robust attribute dispersion based on $K − 1$, or $K + 1$, explaining the zigzag pattern as displayed in Figure 3.11.

What is left unexplained is that the zig-zag pattern is moderated by the values of $\lambda$. The higher the values of $\lambda$, the stronger the influence of the larger attribute dispersions, thus the larger the difference between the softmin attribute dispersions for odd and even $K$. Therefore we see for the highest value of $\lambda$, in the right panel of Figure 3.11, that the zigzag pattern is most strongly pronounced. Moreover, for $K = 2$ we even see that the value of the criterion is even higher than a whole range of both even and odd values values for $K \in \{1, 3, \ldots, 23\}$. This high value of the criterion for $K = 2$ should be related to the standard error of the attribute dispersions.

Since we cannot rely on the central limit theorem for particularly the region of low values for $K$, we conjecture that the attribute dispersions within each neighborhood have a distribution with positive skew. Then, attribute dispersions that have larger standard errors, are also more likely to have larger values. Since, for higher values of $\lambda$ the softmin over the robust attribute dispersions starts to resembles the mean, a stronger influence is to be expected from the larger attribute dispersions, amplified by the larger standard errors.

Even though the standard error of the attribute dispersions for an odd $K$ is larger (based on one attribute distance), than even $K + 1$ (based on two attribute distances), the larger standard errors can only exercise a strong influence when the robust attribute dispersion is also expected to be large. Since the robust dispersions for even $K$ have the tendency to be larger, as formulated in equation (3.15), we can expect that large standard errors have stronger influence on the robust attribute dispersions for even $K$. Thus, in the region of high values for $\lambda$, and for lower values of $K$, it is likely to obtain a higher value of the criterion (the sum of softmin attribute dispersions)
for an even value of $K$, e.g. $K = 2$, as compared to its higher values of $K$, which is
ascribed to the amplifying influence of the standard error.

However, as can be seen in the left panel of Figure 3.11, the opposite effect is
likely to occur for the value of the criterion in the region of the lower values of $\lambda$. For a low value of $\lambda$, the zigzag pattern over $K$ appears to be less pronounced, and
less consistent. The reason is that in the region of lower value of $K$, we see that a
lower $K$ and a lower $\lambda$, causes smaller standard errors for the softmin of the attribute
dispersions.

The influence of the larger attribute dispersions is mitigated by lower values of $\lambda$ in
the softmin estimate, resulting in a less pronounced zigzag pattern is less pronounced. Although counter-intuitively we would expect a larger standard error for a minimum,
than for the mean of the attribute dispersions, this is not necessarily the case in the
region of the lower values of $K$. In fact, the opposite may hold true. Due to the lower
bound of 0, and the positive skew of the attribute dispersions for lower $K$, there is
not much more variation left for the minimum of the robust attribute dispersions. However, the larger $K$, the more variation there is possible for the minimum of the
attribute dispersions, therefore we see that the zigzag pattern can break for the higher
values of $K$.

In this section we have demonstrated the behavior of the criterion for noise-only
data. When the data would consist of a clustering structure, then there are more ‘out-
lying’ attribute dispersions that are based on neighborhoods that consist of incorrect
and correct neighbors (see Figure 3.5 for the example distribution of an attribute dis-
tance between an object with an incorrect neighbor). These outlying robust attribute
dispersions also amplify the zigzag pattern in similar vein as is done by higher values
of $\lambda$. In conclusion, the COSA-KNN criterion that is based on the robust attribute
dispersions shows a zigzag pattern as a function of $K$, which is amplified by $\lambda$ or by a
clustering structure, but, when plotting the criterion as function of only even or odd
$K$, the behavior of the robust criterion in equation (3.10) is more similar to that of
the non-robust criterion in equation (3.9).

3.3 Tuning $\lambda$ and $K$ with the Gap statistic

The robust version of COSA-KNN showed to be faster (in Section 3.1.1) and better
(in Section 3.1.2) in capturing the clustering structure, as compared to its non-robust
version. However, we have seen that in the presence of a subtle structure in the data, a
method for choosing the values of $\lambda$ and $K$ is required to obtain a good performance
of COSA-KNN. In this section we will work with robust version of COSA-KNN,
and propose to find the values of the tuning parameters via a procedure that is
based on the criterion: the Gap statistic (Tibshirani et al., 2001). The Gap statistic
was originally developed for selecting the number of clusters in standard $K$-means
clustering algorithms. However, the procedure also got successfully implemented in
Witten and Tibshirani (2010), and Arias-Castro and Pu (2017) for closely related
tuning-problems.

The idea of the Gap statistic is to compare the value of $\log(\tilde{Q}(\lambda,K))$ on observed
data with the expected value of the criterion for an appropriate null reference model (Tibshirani et al., 2001). Define the Gap statistic as

$$\text{Gap}_{\log}(\lambda, K) = E_{\tilde{Q}} \left[ \log \left( \tilde{Q}(\lambda, K) \right) \right] - \log \left( \tilde{Q}(\lambda, K) \right), \quad (3.17)$$

where $E_{\tilde{Q}} \left[ \log \left( \tilde{Q}(\lambda, K) \right) \right]$ denotes the expectation of the criterion for data sets that originate from an appropriate null reference model. We estimate the expectation by taking the average from $B$ copies of $\log \left( \tilde{Q}(\lambda, K) \right)$, where each copy is computed based on a Monte Carlo sample drawn from the reference distribution.

We draw samples from the reference distribution for $\log \left( \tilde{Q}(\lambda, K) \right)$ by computing the COSA-KNN criterion on $B$ permuted data sets $X^\circ_1, \ldots, X^\circ_B$. Each permuted data set is generated by independent permutations of the observations within each attribute. This renders correlated attributes in the original data to become uncorrelated in the permuted data sets. Thus, the Gap statistic quantifies the strength of the clustering that is obtained on the real data set, compared to the clustering result that is obtained from data sets that do not contain any clustering structure. Eventually, we select the values of $\lambda$ and $K$ for which the Gap is largest, or the smallest Gap that is within one standard error (1SE) of the largest Gap, and corresponds to a higher value for $\lambda$ and $K$.

### 3.3.1 1SE Rule and the Simpler COSA-KNN Model

Although the one standard error (1SE) rule was originally proposed for a different tuning parameter problem (Breiman, Friedman, Stone, & Olshen, 1984), its purpose remains similar in the Gap statistic procedure. With the 1SE rule we wish to choose tuning parameter values of a simpler model that would still be comparable to the optimal model.

We define the COSA-KNN model to become simpler for higher values of $\lambda$ or larger values of $K$, which can be conceptually explained as follows. For higher values of $\lambda$, the more difficult it will become to find unique subsets of attributes for each cluster, thus the fewer ‘degrees of freedom’ for the candidate solutions of the subsets of attributes. Similarly, for larger values of $K$, the COSA distances will have more difficulty (less ‘degrees of freedom’) in capturing a clustering structure of smaller clusters. In the most extreme case where $\lambda \to \infty$ and $K \to N - 1$, the COSA distances will just become a ‘simple’ sum over the attribute distances, e.g., the ordinary Manhattan distances. Thus, we define the COSA model to be more ‘complex’, since it has more degrees of freedom for smaller $\lambda$ and smaller $K$ to represent the more more complex clustering structures – smaller clusters with each their own unique subset of important attributes.

Let $\lambda^*$ and $K^*$ be the values for $\lambda$ and $K$ that correspond with the largest Gap statistic, then, we select the smallest Gap statistic that corresponds with $\lambda \geq \lambda^*$ and $K \geq K^*$, and is within the range of 1SE of the largest Gap statistic. Note that in general for these higher values of $\lambda$ and $K$, the standard error of the criterion will be lower, and therefore the variance of the Gap statistic will also be lower (since these
are simpler COSA-KNN models). Thus, another way to look at the 1SE rule is that it acknowledges the error with which the maximum Gap statistic itself is estimated.

3.3.2 With or Without the Natural Logarithm

The original version of the Gap statistic, in (3.17), is based on the natural logarithm. Mohajer, Englmeier, and Schmid (2011) corroborated the findings by Dudoit and Fridlyand (2002) that this particular use of the logarithm renders the procedure to prefer more complex models (by overestimating the number of clusters), compared to the Gap statistic that is not based on the natural logarithm, i.e.

$$\text{Gap}(\lambda, K) = E_{\tilde{Q}} \left[ \tilde{Q}(\lambda, K) \right] - \tilde{Q}(\lambda, K).$$

(3.18)

Moreover, Mohajer et al. (2011) demonstrated a proof, for criteria that have a monotone relationship with their tuning parameters, that there are situations in which it becomes impossible to find the optimal Gap statistic in the original procedure. Their proof also shows that whenever the original Gap\text{log}(\lambda, K) results in a solution, this solution will always be a possible solution with Gap(\lambda, K) as well, but the reverse is not necessarily true. However, it is not clear whether the proof of Mohajer et al. (2011) also holds for \tilde{Q}(\lambda, K).

The motivation Tibshirani et al. (2001) provide for taking the logarithm, seems solely based on interpretation reasons from likelihood theory. For those cases where the Gap statistic is applied on results from a K-means clustering algorithm, the Gap statistics behaves as a likelihood-ratio statistic based on mixture models (e.g. Scott and Symons, 1971). However, this is not an advantage we know how to exploit for COSA, nor does the logarithm provides us with computational advantages. Nevertheless, we will compare both versions of the Gap statistic: with and without using the logarithm in equations (3.17) and (3.18), respectively.

3.3.3 The Algorithm for the Gap statistic procedure

For computing the Gap statistic with COSA-KNN, the following steps are required:

1. Compute the criterion obtained by performing COSA-KNN on the data $X$ for each candidate combination of the tuning parameter values $K$ and $\lambda$.

2. Obtain permuted datasets $X_1^\circ, \ldots, X_B^\circ$ by independently permuting the observations within each attribute.

   (a) For $b = 1, 2, \ldots, B$, compute $\log \left( \tilde{Q}_b^\circ(\lambda, K) \right)$, the criterion obtained by performing COSA-KNN with candidate tuning parameter values $\lambda$ and $K$ on the data $X_b^\circ$.

   (b) Compute

   $$\text{Gap}_{\text{log}}(\lambda, K) = \frac{1}{B} \sum_{b=1}^{B} \log \left( \tilde{Q}_b^\circ(\lambda, K) \right) - \log \left( \tilde{Q}(\lambda, K) \right).$$

   (3.19)
or
\[
\text{Gap}(\lambda, K) = \frac{1}{B} \sum_{b=1}^{B} \tilde{Q}^c_b(\lambda, K) - \tilde{Q}(\lambda, K). \tag{3.20}
\]

3. Choose $\lambda^*$ and $K^*$ corresponding to the largest value of $\text{Gap}(\lambda, K)$. Then, choose the simplest COSA-$K$NN model (smallest standard error) that is within range of one standard error of the value of $\text{Gap}(\lambda^*, K^*)$. We assure in the computation of the standard error that it additionally takes into account the simulation error, i.e.
\[
\text{se}_{\log}(\tilde{Q}^c(\lambda, K)) = \sqrt{\left(1 + \frac{1}{B}\right) \text{VAR}_b \left(\log \left(\tilde{Q}^c_b(\lambda, K)\right)\right)}, \tag{3.21}
\]
for $\text{Gap}_{\log}(\lambda, K)$, or as
\[
\text{se}_{\tilde{Q}^c}(\lambda, K) = \sqrt{\left(1 + \frac{1}{B}\right) \text{VAR}_b \left(\tilde{Q}^c_b(\lambda, K)\right)}, \tag{3.22}
\]
for $\text{Gap}(\lambda, K)$.

In the text that will follow, the ‘Gap statistic’ may have two meanings. Either the Gap statistic refers to the one that is computed in equation (3.20), or it will be clear from the context that we may refer to the Gap statistic in general, i.e., both versions of the Gap statistic. Moreover, we refer to the Gap statistic as computed in equation (3.19), as the $\text{Gap}_{\log}$ statistic.

### 3.4 Applications of the Gap Statistic

In this section we will apply the Gap Statistic procedure to three different data sets to demonstrate its use and effectiveness. Since we apply the Gap statistic to the robust criterion of COSA-$K$NN, we need to take into account the zigzag pattern over the odd and even values of $K$. Remember that the standard errors are higher for attribute dispersions of even $K$, compared to attribute dispersions on odd $K$. In a data set with a clustering structure we can even expect this inequality to become larger due to larger possible attribute distances in neighborhoods consisting of a mix of correct and incorrect neighbors. Assuming that the observed data contains a detectable clustering structure, the zigzag pattern is expected to be more strongly present in the criterion $\tilde{Q}(\lambda, K)$, than in $\tilde{Q}^c(\lambda, K)$. Therefore the zigzag pattern propagates in the Gap statistics as well, where larger Gap statistics are to be expected for even $K$, compared to odd $K$. More about the zigzag behavior will follow in a demonstration of the use of the Gap statistic on a real data example.

In Section 3.4.2 we give an example that shows how the Gap statistic procedure contributes in revealing a subtle clustering structure that would not have been revealed using COSA-$K$NN with default values for $\lambda$ and $K$. That the Gap statistic does not steer COSA-$K$NN to find a clustering at all costs, is shown with a noise-only data example in Section 3.4.3. We will also describe and compare the results for both versions Gap and $\text{Gap}_{\log}$ of the Gap statistic procedure.
3.4.1 ApoE3 Data Example: Mind the Gap ZigZag

To show the zigzag pattern with the Gap statistic procedure, we apply the robust version of COSA-KNN together with the Gap statistic on a small metabolomics data set from the study by Damian et al. (2007) about a Apolipoprotein variant ‘E3’. Therefore, we refer to this data set as ApoE3. The ApoE3 data set consists of $P = 1550$ LC-MS (liquid chromatography-mass spectrometry) measurements of plasma lipids on $N = 38$ mice, of which 18 were transgenic, and 20 were wild type mice.

Since the zigzag pattern in the robust criterion of COSA-KNN is more strongly visible for small $K$, we can expect to also see zigzag behavior over odd and even values for $K$ for the Gap Statistics. Still, even with the zigzag pattern we can see that the Gap statistic contributes towards a better separation of the clusters, although COSA-KNN also captures the clustering structure in this data set when the default tuning parameter values would have been used. Figure 3.12 depicts the default COSA-KNN results; using the default values of the tuning parameters we obtain a good separation of the wild type mice from the transgenic mice. Both the multidimensional scaling and the hierarchical clustering visualization show that the wild type mice are more homogeneous as a group than the transgenic mice.

![Figure 3.12: The visualization by multidimensional scaling (left panel) and an average linkage dendrogram (right panel) of the COSA-KNN distances for the default tuning parameters $\lambda = 0.2$ and $K = 6$. The red objects are the transgenic mice and the blue objects are the wild type mice.](image)

**Step 1: Set the Grid of Tuning Parameter Combinations**

We start by computing the (robust) criterion of COSA-KNN over a grid of $\lambda$ and $K$. Since the data set is small and not much computing time is required, we will use a dense grid consisting of all combinations for

$$\lambda \in \{0.01, 0.025, 0.05, 0.075, \ldots, 0.3, 0.35, 0.4\},$$

and
\( K \in \{1, 2, \ldots, 24\} \).

A visualization of the criterion \( \tilde{Q}(\lambda, K) \), as well as the log \( \log \tilde{Q}(\lambda, K) \), is shown in Figure 3.13. As expected, the value of the criterion is systematically higher for even \( K \) than for odd \( K \).

\[ \text{Figure 3.13: The robust criterion } \tilde{Q} \text{ (left panel) and the logarithm of the robust criterion } \log \tilde{Q} \text{ (right panel) depicted as a function } \lambda \text{ and } K. \]

**Step 2: Compute the Gap Statistics**

In the second step of the Gap statistic procedure we compute the estimates of the expectation of \( \tilde{Q}^o \) and \( \log \tilde{Q}^o \) for each combination of \( \lambda \) and \( K \) based on the \( B \) Monte Carlo reference data sets. The results can be found in Figure 3.14. Compared to the value of the criterion for the data, the expected value of the criterion of the null reference model, \( E_{\tilde{Q}_b^o} \left[ \tilde{Q}_b^o \right] \), is higher for most combinations of \( \lambda \) and \( K \), and it shows a slightly stronger ‘zigzag’ tendency between even and odd values for \( K \). A similar pattern is found for the logarithm of the value of the criterion for the data set and that of the expected value of the log criterion of the reference model.

\[ \text{Figure 3.14: The computed expectation of the robust criterion } \tilde{Q}^o \text{ (left panel), and its natural logarithm } \log \tilde{Q}^o \text{ (right panel), depicted as a function } \lambda \text{ and } K. \]
Since the zigzag tendency is more strongly pronounced in the values for the null reference criterion, both $\text{Gap}(\lambda, K)$ and $\text{Gap}_{\log}(\lambda, K)$ have higher values for even $K$ compared to odd $K$, as can be seen in Figure 3.15. When we visualize the Gap statistics as a function of odd or even values of $K$ only, then the zigzag pattern disappears, as can be seen in Figure 3.16. Apart from the zigzag pattern, the Gap statistic seems to be unimodal for $\lambda$ while keeping $K$ fixed, and unimodal for most $K$ while keeping $\lambda$ fixed.

Our findings for the optimal tuning parameters with $\text{Gap}$ and $\text{Gap}_{\log}$ are consistent with the results from Mohajer et. al (2011). We see that the $\text{Gap}_{\log}$ prefers the slightly more complex COSA-KNN models, i.e. lower values of $\lambda$ and smaller numbers of $K$ receive high values for $\text{Gap}_{\log}$, as compared to the values of $\text{Gap}$. Also, the maximum value for $\text{Gap}_{\log}$ is for a lower $\lambda$ and smaller $K$, than the maximum value of $\text{Gap}$.

![Figure 3.15: A heat map of the Gap statistics $\text{Gap}(\lambda, K)$ (left) and $\text{Gap}_{\log}(\lambda, K)$ (right) for the ApoE3 data. A dark red color corresponds to a high value of the Gap statistic, whereas the dark blue color corresponds to a low value of the Gap statistic.](image)
Figure 3.16: A heat map of the Gap statistics $\text{Gap}(\lambda, K)$ (left) and $\text{Gap}_{\log}(\lambda, K)$ (right) for the ApoE3 data. The top panels depict the Gap statistics for only odd values for $K$, the bottom panels depict only even values for $K$. A dark red color corresponds to a high value of the Gap statistic, whereas the dark blue color corresponds to a low value of the Gap statistic.
Step 3: Choose the Optimal Tuning Parameter Values

In the Tables 3.1 and 3.2, the eight highest Gap statistics for Gap and Gap\text{log}, respectively, are in ascending order. First, we will describe the results of the eight highest values for the Gap statistics.

Table 3.1 shows that there are only even numbers for $K$ selected as good candidates in the region of $10 \leq K < 16$. The maximum value for the Gap statistic, $\text{Gap}(K = 12, \lambda = 0.125) = 2.6091$, occurs at $K^* = 12$ and $\lambda^* = 0.125$. Using Breiman’s one standard error rule, we should select the values for $K$ and $\lambda$ of the simplest COSA-KNN model with a Gap statistic value higher than 2.5357. From the simpler models we can either select $\lambda = 0.150$ with $K = 10$, since it has the lowest standard error, or the model where $\lambda = 0.150$ and $K = 12$, since it is the model with the lowest standard error that has higher values on both the tuning parameters. Keeping in mind that the standard errors are also estimations, and that in general higher values for $K$ and $\lambda$ correspond with lower standard errors, we select the values $\lambda = 0.150$ and $K = 12$ to represent the simplest model within the range of one standard error.

Table 3.1: The eight highest Gap statistics for the ApoE3 data in ascending order: $\text{Gap}(\lambda^*, K^*) - \text{se}_{\tilde{Q}^\circ(\lambda^*, K^*)} = 2.5357$. The red Gap statistic value corresponds to $\lambda^*$ and $K^*$; the orange Gap statistic value corresponds to the chosen simplest model that is within the range of one standard error:

<table>
<thead>
<tr>
<th>$\lambda$</th>
<th>$K$</th>
<th>$\text{Gap}(\lambda, K)$</th>
<th>$Q(\lambda, K)$</th>
<th>$E_b Q^\circ_b(\lambda, K)$</th>
<th>$\text{se}_{\tilde{Q}^\circ(\lambda, K)}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.125</td>
<td>16</td>
<td>2.4760</td>
<td>20.6368</td>
<td>23.1129</td>
<td>0.0792</td>
</tr>
<tr>
<td>0.100</td>
<td>12</td>
<td>2.4801</td>
<td>17.6645</td>
<td>20.1446</td>
<td>0.0842</td>
</tr>
<tr>
<td>0.150</td>
<td>14</td>
<td>2.5114</td>
<td>21.8847</td>
<td>24.3961</td>
<td>0.0618</td>
</tr>
<tr>
<td>0.150</td>
<td>10</td>
<td>2.5384</td>
<td>20.8790</td>
<td>23.4174</td>
<td>0.0625</td>
</tr>
<tr>
<td>0.150</td>
<td>12</td>
<td><strong>2.5444</strong></td>
<td>21.4148</td>
<td>23.9592</td>
<td>0.0650</td>
</tr>
<tr>
<td>0.125</td>
<td>15</td>
<td>2.5480</td>
<td>20.2153</td>
<td>22.7634</td>
<td>0.0707</td>
</tr>
<tr>
<td>0.125</td>
<td>10</td>
<td>2.5932</td>
<td>19.2494</td>
<td>21.8427</td>
<td>0.0770</td>
</tr>
<tr>
<td><strong>0.125</strong></td>
<td><strong>12</strong></td>
<td><strong>2.6091</strong></td>
<td>19.7452</td>
<td>22.3543</td>
<td><strong>0.0735</strong></td>
</tr>
</tbody>
</table>

In Figure 3.17 we depict the COSA-KNN results for the tuning parameters that rendered the maximum Gap statistic, as well as the results for the values of $\lambda$ and $K$, based on the one standard error rule. Compared to the results obtained from the default tuning parameter settings of COSA-KNN, the clusters are now better separated, mainly due to a more homogeneous wild type mouse group (depicted in blue).
Figure 3.17: The multidimensional scaling and hierarchical clustering visualizations of the COSA-KNN results for $\lambda^*$ and $K^*$ (in the top row), and the values for $\lambda$ and $K$ of the simplest model within the range of one standard error (bottom row).

Compared to the Gap statistics, the values for $\text{Gap}_{\log}$ steer towards candidate values of $8 \leq K \leq 12$ and $0.075 \leq \lambda \leq 0.100$; more complex COSA-KNN models. In Figure 3.18 we depict the COSA-KNN results for $\lambda^*$ and $K^*$ based on the $\text{Gap}_{\log}$ values, as well as the results for the candidate values for $\lambda$ and $K$ based on the one standard error rule. In these visualizations, the wild type mice are more homogeneous than was shown in Figure 3.12. Instead of being a large group of dispersed objects, the transgenic mice group seems to get separated into subgroups, or at least one core subgroup. Next to these results that we demonstrated, similar results were obtained when the Gap statistic procedure was applied on a grid of only odd values for $K$ (Appendix 3.6.1).
Table 3.2: Eight highest Gap statistics of the log criteria of the ApoE3 data in ascending order: \( \text{Gap}_{\log} (\lambda^*, K^*) = 0.1301 \). The red Gap statistic value corresponds to \( \lambda^* \) and \( K^* \); the orange Gap statistic value corresponds to the chosen simplest model that is within the range of one standard error.

<table>
<thead>
<tr>
<th>( \lambda )</th>
<th>( K )</th>
<th>( \text{Gap}_{\log} (\lambda, K) )</th>
<th>( \log Q(\lambda, K) )</th>
<th>( E_b \log Q^b(\lambda, K) )</th>
<th>( \text{se}_{\log Q^b}(\lambda, K) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.050</td>
<td>10</td>
<td>0.1280</td>
<td>2.4226</td>
<td>2.5505</td>
<td>0.0074</td>
</tr>
<tr>
<td>0.075</td>
<td>6</td>
<td>0.1304</td>
<td>2.6141</td>
<td>2.7445</td>
<td>0.0055</td>
</tr>
<tr>
<td>0.075</td>
<td>12</td>
<td>0.1306</td>
<td>2.7109</td>
<td>2.8415</td>
<td>0.0054</td>
</tr>
<tr>
<td>\textbf{0.100}</td>
<td>12</td>
<td>\textbf{0.1314}</td>
<td>2.8716</td>
<td>3.0029</td>
<td>0.0042</td>
</tr>
<tr>
<td>0.100</td>
<td>12</td>
<td>0.1331</td>
<td>2.8154</td>
<td>2.9486</td>
<td>0.0045</td>
</tr>
<tr>
<td>0.100</td>
<td>10</td>
<td>0.1338</td>
<td>2.8452</td>
<td>2.9791</td>
<td>0.0038</td>
</tr>
<tr>
<td>0.075</td>
<td>8</td>
<td>0.1345</td>
<td>2.6526</td>
<td>2.7871</td>
<td>0.0051</td>
</tr>
<tr>
<td>\textbf{0.075}</td>
<td>10</td>
<td>\textbf{0.1357}</td>
<td>2.6822</td>
<td>2.8179</td>
<td>0.0055</td>
</tr>
</tbody>
</table>

Figure 3.18: The multidimensional scaling and hierarchical clustering visualizations of the COSA-KNN results based on \( \text{Gap}_{\log} \) for \( \lambda^* \) and \( K^* \) (in the top row), and the values for \( \lambda \) and \( K \) of the simplest model that is within the range of one standard error of the value for \( \lambda^* \) and \( K^* \) (bottom row).
Conclusion Gap Statistic on the ApoE3 Data

The application of the Gap statistic procedure on COSA-KNN results of the ApoE3 data, reveals a sharper clustering structure than the structure would have revealed with the usual choices of $\lambda = 0.2$ and $K = \sqrt{N}$. This holds for all four solutions based on either the maximum or the one standard error rule for Gap, and for $\text{Gap}_{\log}$. There are small differences between the solutions. The $\text{Gap}_{\log}$ statistics seem to prefer COSA-KNN models for higher $\lambda$, and higher $K$; the more complex models with higher standard errors. Since we don’t know the full truth for this data set, it remains inconclusive whether these models are better or worse than the ones proposed by the Gap statistic where the use of the logarithm is omitted.

Because we are working with the robust version of the COSA-KNN, the highest Gap statistic values will occur for even $K$ for this data example. When taking into account the Gap statistic for only odd $K$, the highest Gap statistics occur in the similar regions for $K$. Not being aware of this artifact could result in specifying a grid for $K$ and $\lambda$ which makes it more prone to end up in a local maximum for the Gap statistic. Especially in the case of small $N$ data sets that results in a grid with small values for $K$, and complex dependency structures in the data. E.g., for this data set, a grid for $K \in \{2, 13, 24\}$ results in $K = 2$. Since it is difficult to determine whether $N$ is large enough, and moreover, since cluster sizes for this particular unsupervised setting are unknown, we would advise to only work with a grid where the candidate values for $K$ are only even, or odd.

3.4.2 The Gap statistic on Subtle Structure Data

To demonstrate the power of the Gap statistic procedure, we show how it can steer COSA-KNN towards good candidate values of the tuning parameters for data generated from subtle structure model in Section 3.1.2 (see Figure 3.7). As was shown already, when we visualize the distances from the default settings of COSA-KNN, i.e. $\lambda = 0.2$ and $K = \sqrt{N} = 10$, we cannot retrieve the grouping structure by hierarchical clustering (with average, single, complete and ward linkage), nor after visualizations of results obtained by Multidimensional Scaling. See Figure 3.19 for the two types of visualizations of robust COSA-KNN results for the usual tuning parameter values.

Since we know the clustering, we could argue that the objects in the MDS visualization (in red and blue) are closer to each other than the noise objects (grey). However, without the coloring, this would not have been clear.
Applying the Gap Statistic Procedure

To find the subtle structure in the data set, we compute the COSA-KNN criterion value, and $B =$ null reference criteria for each combination of

$$\lambda \in \{0.01, 0.05, 0.075, 0.10, 0.15, 0.20, 0.25, 0.30, 0.40\}, \text{ and}$$

$$K \in \{6, 8, 12, 18, 26, 36, 48\}.$$  

In Figure 3.20 we show the heatmap of the resulting Gap statistic values for Gap and Gap$_{\text{log}}$, and in the Tables 3.3 and 3.4 the relevant information of the six highest Gap and Gap$_{\text{log}}$ statistic values is given. The largest value for the Gap statistic is obtained for $\lambda^* = 0.075$ and $K^* = 18$. There is only one simpler COSA-KNN model within the range of one standard error ($se \tilde{Q}(\lambda^*,K^*) = 0.1713$); the model with $\lambda = 0.075$ and $K = 26$ at a Gap of 2.3429, see Table 3.3. The highest value for the Gap$_{\text{log}}$ statistic is found for the combination $\lambda^* = 0.05$ and $K^* = 18$, with no simpler model in the range of one standard error.
Figure 3.20: A heatmap of the Gap values (left) and the $\text{Gap}_{\log}$ values for the Example 2 data as a function of the grid for $\lambda$ and $K$.

Table 3.3: Top 6 highest $\text{Gap}$ statistics in ascending order: $\text{Gap}(\lambda^*, K^*) - \text{se}_{\tilde{\text{Q}}}(\lambda^*, K^*) = 2.2144$. The red $\text{Gap}$ statistic value corresponds to $\lambda^*$ and $K^*$; the orange $\text{Gap}$ statistic value corresponds to the chosen simplest model that is within the range of one standard error.

<table>
<thead>
<tr>
<th>$\lambda$</th>
<th>$K$</th>
<th>Gap ($\lambda, K$)</th>
<th>$Q(\lambda, K)$</th>
<th>$E_b \tilde{Q}(\lambda, K)$</th>
<th>$\text{se}_{\tilde{\text{Q}}}(\lambda, K)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.100</td>
<td>26</td>
<td>1.6792</td>
<td>66.8569</td>
<td>68.5361</td>
<td>0.0386</td>
</tr>
<tr>
<td>0.050</td>
<td>12</td>
<td>1.9360</td>
<td>43.0336</td>
<td>44.9696</td>
<td>0.1848</td>
</tr>
<tr>
<td>0.075</td>
<td>36</td>
<td>2.1513</td>
<td>64.3225</td>
<td>66.4738</td>
<td>0.0823</td>
</tr>
<tr>
<td>0.050</td>
<td>18</td>
<td>2.2413</td>
<td>46.7598</td>
<td>49.0010</td>
<td>0.1633</td>
</tr>
<tr>
<td><strong>0.075</strong></td>
<td><strong>26</strong></td>
<td><strong>2.3429</strong></td>
<td>61.0269</td>
<td>63.3699</td>
<td>0.1095</td>
</tr>
<tr>
<td><strong>0.075</strong></td>
<td><strong>18</strong></td>
<td><strong>2.3858</strong></td>
<td>57.2769</td>
<td>59.6627</td>
<td>0.1713</td>
</tr>
</tbody>
</table>

Table 3.4: Top 6 highest $\text{Gap}_{\log}$ statistics in ascending order: $\text{Gap}_{\log}(\lambda^*, K^*) - \text{se}_{\log \tilde{\text{Q}}}(\lambda^*, K^*) = 0.0435$. The red $\text{Gap}_{\log}$ statistic value corresponds to $\lambda^*$ and $K^*$.

<table>
<thead>
<tr>
<th>$\lambda$</th>
<th>$K$</th>
<th>$\text{Gap}_{\log}(\lambda, K)$</th>
<th>$\log Q(\lambda, K)$</th>
<th>$E_b \log Q(\lambda, K)$</th>
<th>$\text{se}_{\log \tilde{\text{Q}}}(\lambda, K)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.010</td>
<td>8</td>
<td>0.0330</td>
<td>2.4676</td>
<td>2.5006</td>
<td>0.0190</td>
</tr>
<tr>
<td>0.010</td>
<td>12</td>
<td>0.0332</td>
<td>2.5307</td>
<td>2.5638</td>
<td>0.0199</td>
</tr>
<tr>
<td>0.075</td>
<td>26</td>
<td>0.0377</td>
<td>4.1113</td>
<td>4.1490</td>
<td>0.0017</td>
</tr>
<tr>
<td>0.075</td>
<td>18</td>
<td>0.0408</td>
<td>4.0479</td>
<td>4.0887</td>
<td>0.0029</td>
</tr>
<tr>
<td>0.050</td>
<td>12</td>
<td>0.0440</td>
<td>3.7620</td>
<td>3.8060</td>
<td>0.0041</td>
</tr>
<tr>
<td><strong>0.050</strong></td>
<td><strong>18</strong></td>
<td><strong>0.0468</strong></td>
<td>3.8450</td>
<td>3.8918</td>
<td>0.0033</td>
</tr>
</tbody>
</table>
The Results

From Figure 3.21 we can see that successful COSA-KNN results can be obtained for both combinations that are proposed with the Gap statistic; the values of $\lambda^*$ and $K^*$ that correspond to the maximal Gap statistic, and the one standard error rule values for $\lambda$ and $K$. Hardly any difference can be seen between the dendrograms or multidimensional scaling configurations for $\{\lambda = 0.075, K = 18\}$ and $\{\lambda = 0.075, K = 26\}$, which shows that with the one standard we still obtain successful results that can be interpreted as more stable (due to the lower standard error of the criterion).

![Hierarchical Clustering and Multidimensional Scaling](image1)

![Hierarchical Clustering and Multidimensional Scaling](image2)

Figure 3.21: The Multidimensional Scaling and Hierarchical Clustering visualizations of the COSA-KNN results based on Gap for $K^*$ and $\lambda^*$ (in the top row), and the values for $\lambda$ and $K$ of the simplest model that is within the range of one standard error of the value for $\lambda^*$ and $K^*$ (bottom row).

The COSA-KNN results for the values of $\{\lambda = 0.050, K = 18\}$, suggested by the Gap log procedure, show less desirable results, as can be seen in Figure 3.22. Although we still would be able to cut the dendrogram into the two groups and 60 singleton noise objects, the clustering structure is not optimally revealed, some of the noise objects
seem to be close to the clustered objects. While the results may be worse compared to the suggested values of the tuning parameters based on the Gap statistic that omits the logarithm, the results are still better when compared to those obtained with the usual tuning parameter values (Figure 3.19).

![Hierarchical Clustering](image1.png) ![Multidimensional Scaling](image2.png)

$\lambda = 0.05, K = 18$

*Figure 3.22: Visualization of the COSA distances with optimized tuning parameters $\lambda = 0.05$ and $K = 18$, selected by the Gap$_{log}$-statistic procedure.*

**Conclusion Gap Statistic on the Subtle Structure Data**

When we apply the Gap statistic procedure to the COSA-KNN criterion for a data set that is generated from the subtle structure model, we can find the values for $\lambda$ and $K$ with which we can reveal the clustering structure. The results are better for the optimized values of the tuning parameters based on the Gap statistic, than for those based on the Gap$_{log}$. In accordance with the findings of Mohajer et al. (2011), we see that with the Gap$_{log}$ statistic a lower value of $\lambda$ (and $K$) is preferred, that in general corresponds with higher standard errors of the criterion. Thus, a more complex model is preferred with the Gap$_{log}$ statistic compared to the Gap statistic.

**3.4.3 The Gap Statistic and Noise Only Data**

The Gap statistic renders COSA-KNN to be more powerful by steering towards good candidate values for $\lambda$ and $K$. One could argue that the use of Gap statistic, it also renders COSA-KNN to be more prone to a Type-I error, i.e. showing a clustering structure that is supported on only random fluctuations. In this section we will use an example to demonstrate the behavior of the Gap statistic in combination with COSA-KNN on a data set that is supposed to be noise only.

To obtain a data set that is not supposed to contain any systematic clustering structure, we generate a data set of $N = 100$ by $P = 10,000$ i.i.d. standard normal
values. On this data set we will apply COSA-KNN in combination with the Gap statistic, using the exact same settings as in section ?? . Thus, we apply the Gap statistics to the robust COSA-KNN criterion for all tuning parameter combinations on the grid

\[ \lambda \in \{0.01, 0.05, 0.075, 0.10, 0.15, 0.20, 0.25, 0.30, 0.40\}, \text{ and} \]

\[ K \in \{6, 8, 12, 18, 26, 36, 48\}. \]

While the criterion values values for \( \lambda \) and \( K \) seem fine in the left and middle panel of Figure 3.23, the surface of the Gap statistic values for \( \text{Gap}(\lambda, K) \) show more irregular behavior, compared to what we have seen so far. There does not seem to be a reasonable hill climbing path towards a (local) maximum value for the Gap statistic values. The same holds for the \( \text{Gap}_{\log} \) statistics, of which the heatmap is visualized in the right panel of Figure 3.24. This may be a first indication that we are dealing with a data set with no clustering structure. Similarly, from the heatmaps in Figure 3.24 no clear area can be obtained that indicates a ‘sweet heat spot’.

**Figure 3.23:** Surface of the estimated expectation of the log-criterion of the reference model (left) and the criterion (middle) of COSA-KNN for \( \lambda \) and \( K \). The right panel shows the Gap statistic.
A second indication of no clustering structure is when all Gap statistics have low (or even negative) values that are accompanied by large standard errors. In Table 3.5 we see for our specific data set that the values of the Gap statistics are lower when compared to a data that does contain a clustering structure, and has similar dimensions (e.g. the data in the previous section). Moreover, most of the highest Gap statistics have values that are within the range of one standard error from 0, indicating that the COSA-KNN with the optimized values for λ and K performs equally when compared to noise, which actually should be the case for our simulated noise data set.

Table 3.5: Top 8 highest Gap statistics in ascending order. According to the one standard error rule, we should choose the values for λ and K of the simplest possible COSA-KNN model with a Gap statistic value higher than \( \text{Gap}(\lambda^*, K^*) - \text{se}_{\hat{Q}^*}(\lambda^*, K^*) = 0.040 \). The red Gap statistic value corresponds to \( \lambda^* K^* \); the orange Gap statistic value corresponds to the chosen simplest model that is within the range of one standard error.

<table>
<thead>
<tr>
<th>( \lambda )</th>
<th>( K )</th>
<th>( \text{Gap}(\lambda, K) )</th>
<th>( Q(\lambda, K) )</th>
<th>( E_bQ_b^*(\lambda, K) )</th>
<th>( \text{se}_{\hat{Q}^*}(\lambda, K) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.050</td>
<td>12</td>
<td>-0.001</td>
<td>45.611</td>
<td>45.610</td>
<td>0.091</td>
</tr>
<tr>
<td>0.010</td>
<td>6</td>
<td>0.021</td>
<td>11.487</td>
<td>11.508</td>
<td>0.095</td>
</tr>
<tr>
<td>0.010</td>
<td>8</td>
<td>0.039</td>
<td>12.468</td>
<td>12.508</td>
<td>0.167</td>
</tr>
<tr>
<td>0.025</td>
<td>12</td>
<td>0.096</td>
<td>28.750</td>
<td>28.846</td>
<td>0.208</td>
</tr>
<tr>
<td><strong>0.050</strong></td>
<td><strong>48</strong></td>
<td><strong>0.097</strong></td>
<td><strong>63.466</strong></td>
<td><strong>63.563</strong></td>
<td><strong>0.197</strong></td>
</tr>
<tr>
<td>0.010</td>
<td>12</td>
<td>0.099</td>
<td>14.196</td>
<td>14.295</td>
<td>0.222</td>
</tr>
<tr>
<td>0.010</td>
<td>48</td>
<td>0.128</td>
<td>29.499</td>
<td>29.626</td>
<td>0.366</td>
</tr>
<tr>
<td><strong>0.025</strong></td>
<td><strong>18</strong></td>
<td><strong>0.247</strong></td>
<td><strong>31.581</strong></td>
<td><strong>31.828</strong></td>
<td><strong>0.207</strong></td>
</tr>
</tbody>
</table>

The optimal tuning parameters for \( \lambda \) and \( K \) are the same for both versions of the Gap statistic procedures (only the Gap statistic without the logarithm is shown here in Table 3.5 and Figure 3.25). Figure 3.25 depicts the visualizations belonging to the Gap optimized COSA-KNN results. Although, in general the dendrograms have
a stronger tendency to show a clustering structure, these structures seem to be very instable since the dendrogram for $\lambda = 0.025$ with $K = 18$ is very different from the results obtained with the 1 S.E. rule where $\lambda = 0.05$ with $K = 18$. Moreover, the visualizations of the multidimensional scaling configurations do not seem to hint at a clustering structure.

**Conclusion of the Use of the Gap Statistic on Noise Only**

We did not observe for our noise data example that the Gap statistic could steer COSA-KNN towards committing a Type-I error. Moreover, there are three indications by which we could foresee such an error. First, a resulting clustering structure is very likely to be based on sampling fluctuations only if the surface of the Gap statistic values is very irregular. Second, a large standard error for each Gap statistic also indicates that the Gap may just got larger than 0 based on chance. These are two strong indications from which we can derive whether we should trust our Gap...
statistic results for COSA-KNN. Last, instead of using the visualizations based on average linkage dendrograms, the visualizations of the COSA-KNN results by multidimensional scaling seem to provide more stable results.

3.5 Discussion

When a clustering structure is present in the data, the robust version of COSA-KNN is more successful than the non-robust version of COSA-KNN. Not only is the robust version faster in revealing the clustering structure, we have also seen an example where only the robust version can reveal the clustering structure. The number of successful candidate combinations of the tuning parameter values of $\lambda$ and $K$, is enlarged for robust COSA-KNN.

To automatically find successful candidate combinations of the tuning parameter, we demonstrated that the application of the Gap statistic on the criterion of robust COSA-KNN showed successful results. While the non-robust criterion seems to be a concave smooth surface over a grid of values for both $\lambda$ and $K$, the robust version of the criterion shows a zigzag pattern over the odd and even values for $K$. This particular zigzag pattern is also propagated in the Gap statistic values, leading towards the preference of even values for $K$, over odd values for $K$. We suggest to use a grid with preferably only even values for $K$.

The Gap statistic values can be computed over the robust COSA-KNN criterion directly, or on the natural logarithm of each criterion. Although for these two different procedures we did not find differences that would have resulted in different clustering structures, we did replicate earlier findings that the Gap$\log$ statistic procedure has the tendency to prefer the tuning parameter values corresponding to more complex models, i.e. lower values for $\lambda$ and lower values of $K$. Although it remains inconclusive which of the two versions of the Gap statistic procedure is better, we prefer the version that applies Occam’s Razor. Thus, we will be using the Gap statistic procedure that omits the logarithm in the succeeding chapters of this monograph for COSA.

While keeping $K$ fixed, the Gap statistic seems to be very close to a unimodal function for the values of $\lambda$. Thus, for each $K$ we may find candidates for the optimal $\lambda$ using a golden search algorithm (e.g., Brent, 1971) as was also proposed in Arias-Castro and Pu (2017) for the sparsity parameter of their algorithm. However, this extension deserves more investigation.
3.6 Appendix

3.6.1 Considering Only Odd Values $K$ for the ApoE3 Data

Table 3.6: The five highest $\text{Gap}_{\log}$ statistics for odd $K$ only and of the log criteria, of the ApoE3 data in ascending order: $\text{Gap}_{\log}(\lambda^*, K^*) - \text{se}_{\log Q_{\circ}(\lambda^*, K^*)} = 0.1152$. The red $\text{Gap}_{\log}$ statistic value corresponds to $\lambda^*$ and $K^*$; the orange $\text{Gap}$ statistic value corresponds to the chosen simplest model that is within the range of one standard error.

<table>
<thead>
<tr>
<th>$\lambda$</th>
<th>$K$</th>
<th>$\text{Gap}_{\log}(\lambda, K)$</th>
<th>$\log Q(\lambda, K)$</th>
<th>$E_b \log Q^\circ_b(\lambda, K)$</th>
<th>$\text{se}_{\log Q^\circ(\lambda, K)}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.050</td>
<td>9</td>
<td>0.1175</td>
<td>2.3919</td>
<td>2.5095</td>
<td>0.0067</td>
</tr>
<tr>
<td>0.075</td>
<td>9</td>
<td>0.1178</td>
<td>2.6517</td>
<td>2.7694</td>
<td>0.0047</td>
</tr>
<tr>
<td><strong>0.100</strong></td>
<td><strong>13</strong></td>
<td><strong>0.1178</strong></td>
<td><strong>2.8744</strong></td>
<td><strong>2.9922</strong></td>
<td><strong>0.0037</strong></td>
</tr>
<tr>
<td>0.100</td>
<td>11</td>
<td>0.1186</td>
<td>2.8466</td>
<td>2.9651</td>
<td>0.0037</td>
</tr>
<tr>
<td><strong>0.075</strong></td>
<td><strong>11</strong></td>
<td><strong>0.1200</strong></td>
<td><strong>2.6855</strong></td>
<td><strong>2.8055</strong></td>
<td><strong>0.0048</strong></td>
</tr>
</tbody>
</table>

Figure 3.26: A Visualization of the COSA-KNN results that are optimized based on the $\text{Gap}_{\log}$ statistic procedure. The top panels shows the multidimensional scaling configuration and the average linkage dendrogram for the tuning parameters values that correspond to the maximum $\text{Gap}_{\log}$ statistic, the bottom panels show the visualizations for the simplest model in the range of one standard error.
Table 3.7: The seven highest Gap statistics, for odd $K$ only, of the ApoE3 data in ascending order: $\text{Gap}(\lambda^*, K^*) - \tilde{Q}_{\lambda^*, K^*} = 2.2590$. The red Gap statistic value corresponds to $\lambda^*$ and $K^*$; the orange Gap statistic value corresponds to the chosen simplest model that is within the range of one standard error.

<table>
<thead>
<tr>
<th>$\lambda$</th>
<th>$K$</th>
<th>$\text{Gap} (\lambda, K)$</th>
<th>$Q (\lambda, K)$</th>
<th>$E_b \tilde{Q}_b (\lambda, K)$</th>
<th>$\text{se}_{\tilde{Q}_b (\lambda, K)}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.125</td>
<td>17</td>
<td>2.2211</td>
<td>20.7195</td>
<td>22.9406</td>
<td>0.0701</td>
</tr>
<tr>
<td>0.150</td>
<td>15</td>
<td><strong>2.2654</strong></td>
<td>21.9253</td>
<td>24.1907</td>
<td>0.0625</td>
</tr>
<tr>
<td>0.125</td>
<td>15</td>
<td>2.2848</td>
<td>20.2745</td>
<td>22.5593</td>
<td>0.0673</td>
</tr>
<tr>
<td>0.150</td>
<td>11</td>
<td>2.2903</td>
<td>20.8549</td>
<td>23.1451</td>
<td>0.0597</td>
</tr>
<tr>
<td>0.150</td>
<td>13</td>
<td>2.3032</td>
<td>21.4353</td>
<td>23.7385</td>
<td>0.0611</td>
</tr>
<tr>
<td>0.125</td>
<td>11</td>
<td>2.3123</td>
<td>19.2336</td>
<td>21.5459</td>
<td>0.0704</td>
</tr>
<tr>
<td><strong>0.125</strong></td>
<td><strong>13</strong></td>
<td><strong>2.3322</strong></td>
<td><strong>19.7785</strong></td>
<td><strong>22.1107</strong></td>
<td><strong>0.0732</strong></td>
</tr>
</tbody>
</table>

Figure 3.27: A Visualization of the COSA-KNN results that are optimized based on the Gap statistic procedure. The top panels shows the multidimensional scaling configuration and the average linkage dendrogram for the tuning parameters values that correspond to the maximum Gap statistic, the bottom panels show the visualizations for the simplest model in the range of one standard error.
### 3.6.2 Top 20 Gap Statistics for the ApoE3 Data

Table 3.8: The 20 highest Gap statistics for \( K \) on the criterion of the ApoE3 data in ascending order: \( \text{GAP}(\lambda^*, K^*) - \text{se} \tilde{Q}^b(\lambda^*, K^*) = 2.5357 \). Note that the first odd value for \( K \) is the 18\(^{th}\) highest Gap statistic.

<table>
<thead>
<tr>
<th>( \lambda )</th>
<th>( K )</th>
<th>( \text{Gap}(\lambda, K) )</th>
<th>( \text{Q}(\lambda, K) )</th>
<th>( E_b Q^b(\lambda, K) )</th>
<th>( \text{se} \tilde{Q}^b(\lambda, K) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1250</td>
<td>11</td>
<td>2.3123</td>
<td>19.2336</td>
<td>21.5459</td>
<td>0.0704</td>
</tr>
<tr>
<td>0.4000</td>
<td>2</td>
<td>2.3270</td>
<td>26.2375</td>
<td>28.5646</td>
<td>0.0562</td>
</tr>
<tr>
<td>0.1250</td>
<td>13</td>
<td>2.3322</td>
<td>19.7785</td>
<td>22.1107</td>
<td>0.0732</td>
</tr>
<tr>
<td>0.2000</td>
<td>12</td>
<td>2.3402</td>
<td>23.8172</td>
<td>26.1574</td>
<td>0.0532</td>
</tr>
<tr>
<td>0.1000</td>
<td>16</td>
<td>2.3432</td>
<td>18.5401</td>
<td>20.8833</td>
<td>0.0807</td>
</tr>
<tr>
<td>0.2000</td>
<td>10</td>
<td>2.3579</td>
<td>23.2857</td>
<td>25.6436</td>
<td>0.0557</td>
</tr>
<tr>
<td>0.1500</td>
<td>8</td>
<td>2.3696</td>
<td>20.3386</td>
<td>22.7082</td>
<td>0.0578</td>
</tr>
<tr>
<td>0.1000</td>
<td>8</td>
<td>2.3785</td>
<td>16.7005</td>
<td>19.0790</td>
<td>0.0850</td>
</tr>
<tr>
<td>0.1250</td>
<td>8</td>
<td>2.4038</td>
<td>18.7388</td>
<td>21.1426</td>
<td>0.0667</td>
</tr>
<tr>
<td>0.1000</td>
<td>14</td>
<td>2.4183</td>
<td>18.1167</td>
<td>20.5350</td>
<td>0.0869</td>
</tr>
<tr>
<td>0.1500</td>
<td>16</td>
<td>2.4302</td>
<td>22.3189</td>
<td>24.7491</td>
<td>0.0691</td>
</tr>
<tr>
<td>0.1000</td>
<td>10</td>
<td>2.4639</td>
<td>17.2057</td>
<td>19.6696</td>
<td>0.0752</td>
</tr>
<tr>
<td>0.1250</td>
<td>16</td>
<td>2.4760</td>
<td>20.6368</td>
<td>23.1129</td>
<td>0.0792</td>
</tr>
<tr>
<td>0.1000</td>
<td>12</td>
<td>2.4801</td>
<td>17.6645</td>
<td>20.1446</td>
<td>0.0842</td>
</tr>
<tr>
<td>0.1500</td>
<td>14</td>
<td>2.5114</td>
<td>21.8847</td>
<td>24.3961</td>
<td>0.0618</td>
</tr>
<tr>
<td>0.1500</td>
<td>10</td>
<td>2.5384</td>
<td>20.8790</td>
<td>23.4174</td>
<td>0.0625</td>
</tr>
<tr>
<td><strong>0.150</strong></td>
<td><strong>12</strong></td>
<td><strong>2.5444</strong></td>
<td><strong>21.4148</strong></td>
<td><strong>23.9592</strong></td>
<td><strong>0.0650</strong></td>
</tr>
<tr>
<td>0.125</td>
<td>14</td>
<td>2.5480</td>
<td>20.2153</td>
<td>22.7634</td>
<td>0.0707</td>
</tr>
<tr>
<td>0.125</td>
<td>10</td>
<td>2.5932</td>
<td>19.2494</td>
<td>21.8427</td>
<td>0.0770</td>
</tr>
<tr>
<td><strong>0.125</strong></td>
<td><strong>12</strong></td>
<td><strong>2.6091</strong></td>
<td><strong>19.7452</strong></td>
<td><strong>22.3543</strong></td>
<td><strong>0.0735</strong></td>
</tr>
</tbody>
</table>
### 3.6.3 Top 20 \( \text{Gap}_{\text{log}} \) Statistics for the ApoE3 Data

Table 3.9: The 20 highest \( \text{Gap}_{\text{log}} \) statistics for \( K \) based on the \( \text{log} \) criteria of the ApoE3 data in ascending order: \( \text{GAP}_{\text{log}}(\lambda^*, K^*) - \text{se}_{\text{log}} \hat{Q}^*(\lambda^*, K^*) = 0.1301 \). Note that the first odd value for \( K \) is the 17\(^{th} \) highest \( \text{Gap}_{\text{log}} \) statistic.

<table>
<thead>
<tr>
<th>( \lambda )</th>
<th>( K )</th>
<th>( \text{Gap}_{\text{log}}(\lambda, K) )</th>
<th>( \log Q(\lambda, K) )</th>
<th>( E_b \log Q_b^*(\lambda, K) )</th>
<th>( \text{se}_{\text{log}} \hat{Q}^*(\lambda, K) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.100</td>
<td>11</td>
<td>0.1186</td>
<td>2.8466</td>
<td>2.9651</td>
<td>0.0037</td>
</tr>
<tr>
<td>0.125</td>
<td>14</td>
<td>0.1187</td>
<td>3.0064</td>
<td>3.1251</td>
<td>0.0031</td>
</tr>
<tr>
<td>0.100</td>
<td>16</td>
<td>0.1190</td>
<td>2.9199</td>
<td>3.0389</td>
<td>0.0039</td>
</tr>
<tr>
<td>0.075</td>
<td>11</td>
<td>0.1200</td>
<td>2.6855</td>
<td>2.8055</td>
<td>0.0048</td>
</tr>
<tr>
<td>0.050</td>
<td>6</td>
<td>0.1203</td>
<td>2.3585</td>
<td>2.4788</td>
<td>0.0064</td>
</tr>
<tr>
<td>0.125</td>
<td>8</td>
<td>0.1207</td>
<td>2.9306</td>
<td>3.0513</td>
<td>0.0032</td>
</tr>
<tr>
<td>0.075</td>
<td>14</td>
<td>0.1214</td>
<td>2.7401</td>
<td>2.8615</td>
<td>0.0053</td>
</tr>
<tr>
<td>0.125</td>
<td>12</td>
<td>0.1241</td>
<td>2.9829</td>
<td>3.1070</td>
<td>0.0033</td>
</tr>
<tr>
<td>0.100</td>
<td>14</td>
<td>0.1253</td>
<td>2.8968</td>
<td>3.0221</td>
<td>0.0042</td>
</tr>
<tr>
<td>0.100</td>
<td>6</td>
<td>0.1254</td>
<td>2.7759</td>
<td>2.9013</td>
<td>0.0039</td>
</tr>
<tr>
<td>0.125</td>
<td>10</td>
<td>0.1264</td>
<td>2.9575</td>
<td>3.0839</td>
<td>0.0035</td>
</tr>
<tr>
<td>0.050</td>
<td>8</td>
<td>0.1268</td>
<td>2.3943</td>
<td>2.5211</td>
<td>0.0069</td>
</tr>
<tr>
<td>0.050</td>
<td>10</td>
<td>0.1280</td>
<td>2.4226</td>
<td>2.5505</td>
<td>0.0074</td>
</tr>
<tr>
<td>0.075</td>
<td>6</td>
<td>0.1304</td>
<td>2.6141</td>
<td>2.7445</td>
<td>0.0055</td>
</tr>
<tr>
<td>0.075</td>
<td>12</td>
<td>0.1306</td>
<td>2.7109</td>
<td>2.8415</td>
<td>0.0054</td>
</tr>
<tr>
<td><strong>0.100</strong></td>
<td><strong>12</strong></td>
<td><strong>0.1314</strong></td>
<td><strong>2.8716</strong></td>
<td><strong>3.0029</strong></td>
<td><strong>0.0042</strong></td>
</tr>
<tr>
<td>0.100</td>
<td>8</td>
<td>0.1331</td>
<td>2.8154</td>
<td>2.9486</td>
<td>0.0045</td>
</tr>
<tr>
<td>0.100</td>
<td>10</td>
<td>0.1338</td>
<td>2.8452</td>
<td>2.9791</td>
<td>0.0038</td>
</tr>
<tr>
<td>0.075</td>
<td>8</td>
<td>0.1345</td>
<td>2.6526</td>
<td>2.7871</td>
<td>0.0051</td>
</tr>
<tr>
<td><strong>0.075</strong></td>
<td><strong>10</strong></td>
<td><strong>0.1357</strong></td>
<td><strong>2.6822</strong></td>
<td><strong>2.8179</strong></td>
<td><strong>0.0055</strong></td>
</tr>
</tbody>
</table>
3.6.4 \( \text{Gap}_{\log} \) for the Noise Only Data Set

Table 3.10: Top 8 highest Gap statistics in ascending order. According to the one standard error rule, we should choose the values for \( \lambda \) and \( K \) of the simplest possible COSA-KNN model with a Gap statistic value higher than \( \text{Gap}_{\log}(\lambda^*, K^*) - \text{se}_{\log \tilde{Q}^\circ}(\lambda^*, K^*) = 0.0013 \). The red Gap statistic value corresponds to \( \lambda^* \) \( K^* \); the orange Gap statistic value corresponds to the chosen simplest model that is within the range of one standard error.

<table>
<thead>
<tr>
<th>( K )</th>
<th>( \lambda )</th>
<th>( \text{Gap}_{\log}(\lambda, K) )</th>
<th>( \log Q(\lambda, K) )</th>
<th>( E_\theta \log \tilde{Q}^\circ(\lambda, K) )</th>
<th>( \text{se}_{\log \tilde{Q}^\circ}(\lambda, K) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>0.050</td>
<td>-0.0000</td>
<td>3.8201</td>
<td>3.8201</td>
<td>0.0020</td>
</tr>
<tr>
<td>48</td>
<td><strong>0.050</strong></td>
<td><strong>0.0015</strong></td>
<td>4.1505</td>
<td>4.1520</td>
<td>0.0031</td>
</tr>
<tr>
<td>6</td>
<td>0.010</td>
<td>0.0018</td>
<td>2.4412</td>
<td>2.4430</td>
<td>0.0083</td>
</tr>
<tr>
<td>8</td>
<td>0.010</td>
<td>0.0031</td>
<td>2.5232</td>
<td>2.5263</td>
<td>0.0133</td>
</tr>
<tr>
<td>12</td>
<td>0.025</td>
<td>0.0033</td>
<td>3.3586</td>
<td>3.3619</td>
<td>0.0072</td>
</tr>
<tr>
<td>48</td>
<td>0.010</td>
<td>0.0043</td>
<td>3.3843</td>
<td>3.3886</td>
<td>0.0124</td>
</tr>
<tr>
<td>12</td>
<td>0.010</td>
<td>0.0068</td>
<td>2.6530</td>
<td>2.6598</td>
<td>0.0155</td>
</tr>
<tr>
<td><strong>18</strong></td>
<td><strong>0.025</strong></td>
<td><strong>0.0078</strong></td>
<td>3.4526</td>
<td>3.4603</td>
<td>0.0065</td>
</tr>
</tbody>
</table>

Figure 3.28: The multidimensional scaling and hierarchical clustering visualizations of the COSA-KNN results for 'noise only' data based on \( \text{Gap}_{\log} \) for \( K^* \) and \( \lambda^* \) (in the top row), and the values for \( \lambda \) and \( K \) of the simplest model that is within the range of one standard error of the value for \( K^* \) and \( \lambda^* \) (bottom row).