Advances in Data Analysis and Classification (2019) 13:905–932
https://doi.org/10.1007/s11634-019-00356-9

REGULAR ARTICLE

Robust and sparse k-means clustering for high-dimensional data

Šárka Brodinová[1] · Peter Filzmoser[1] · Thomas Ortner[1] · Christian Breiteneder[2] · Maia Rohm[2]

Received: 17 August 2017 / Revised: 24 February 2019 / Accepted: 6 March 2019 /
Published online: 19 March 2019

Abstract
In real-world application scenarios, the identification of groups poses a significant challenge due to possibly occurring outliers and existing noise variables. Therefore, there is a need for a clustering method which is capable of revealing the group structure in data containing both outliers and noise variables without any pre-knowledge. In this paper, we propose a k-means-based algorithm incorporating a weighting function which leads to an automatic weight assignment for each observation. In order to cope with noise variables, a lasso-type penalty is used in an objective function adjusted by observation weights. We finally introduce a framework for selecting both the number of clusters and variables based on a modified gap statistic. The conducted experiments on simulated and real-world data demonstrate the advantage of the method to identify groups, outliers, and informative variables simultaneously.

Keywords Clusters · Outliers · Noise variables · High-dimensions · Gap statistic

Mathematics Subject Classification 62H30

1 Introduction

The identification of groups in real-world high-dimensional datasets reveals challenges due to several aspects: (1) the presence of outliers; (2) the presence of noise variables; (3) the selection of proper parameters for the clustering procedure, e.g. the number of clusters. Whereas we have found a lot of work addressing the three aspects separately, a much smaller number of studies is available in case all three aspects are treated simultaneously. Indeed, in any large and high-dimensional complex dataset, not only

✉ Šárka Brodinová
sarka.brodinova@tuwien.ac.at

[1] Institute of Statistics and Mathematical Methods in Economics, TU Wien, Vienna, Austria

[2] Institute of Software Technology and Interactive Systems, TU Wien, Vienna, Austria

outliers but also noise variables are very likely to appear. Hence, a clustering method needs to be designed in such a way that both aspects are taken into account, no matter if outliers are considered as highly interesting observations due to their typically different content or just as noise. The data complexity in terms of the number of groups and the proportion of outliers as well as the number of noise variables very much depends on the dataset itself. Therefore, a clustering procedure should ideally be data-independent. In other words, no information about the data complexity should be assumed. The goal of this paper is to introduce a clustering method designed for such an application scenario.

Considering the task of revealing the group structure in contaminated data, i.e. data with outliers, a natural step is to first apply an outlier detection procedure to exclude deviating observations for the following cluster analysis. However, coping with outliers in such a way might be complicated due to the parameter specification, which is commonly required by most existing clustering (e.g. the number of clusters) as well as outlier detection methods (Aggarwal 2016). Better strategies can be to incorporate a measure of outlyingness through data clustering (see e.g. Campello et al. 2015), to use forward search techniques (see e.g. Cerioli et al. 2018; Atkinson et al. 2018), or trimming-based clustering approaches (see e.g. Neykov et al. 2007; Gallegos and Ritter 2009; García-Escudero et al. 2008, 2010; Coretto and Hennig 2016). The general idea of trimming-based approaches is to exclude observations which usually do not fit to an assumed model. In order to apply a trimming concept, not only the number of clusters but also the trimming level, i.e. the proportion of observations supposed to be discarded, need to be specified in advance. Several techniques have been proposed to solve the problem of pre-specifiyng this parameter (see e.g. García-Escudero et al. 2011; Dotto et al. 2018). Although all these approaches perform well on contaminated data sets, they can be easily affected by a large number of variables holding no information for cluster separation.

The problem of data clustering in the presence of noise variables is usually addressed by sparse- and variable selection-based clustering approaches (see e.g. Witten and Tibshirani 2010; Raftery and Dean 2006). The methods generally aim at removing noise variables that can easily mask a group structure (Gordon 1999). An overview of such methods can be found in the study by Galimberti et al. (2018) with a special focus on model-based clustering. Although the number of clusters in model-based clustering is commonly estimated based on the Bayesian information criterion, some methods usually assume that the size of a group is typically larger than the dimensionality of the data space where a group is located. Therefore, such approaches might have troubles to sufficiently discover high-dimensional low sample size groups. A suitable method for such a situation is introduced by Witten and Tibshirani (2010). The method imposes a lasso-type penalty on incorporated variable weights in the objective function of k-means leading to the sparse k-means algorithm. In order to apply the sparse k-means, the number of clusters needs to be determined in advance, which is hardly possible for most real-world application scenarios.

The task of identifying groups becomes even more problematic when both outliers and noise variables are present. For this situation, Kondo et al. (2016) introduce the robust and sparse k-means (RSKC) that robustifies the sparse k-means by Witten and Tibshirani (2010) by incorporating a trimming concept. However, the approach

assumes prior knowledge about the number of clusters, the degree of sparsity, and the trimming level in order to correctly detect clusters. Furthermore, the method has been tested only in terms of clustering and no evaluation has been performed regarding the detection of outliers. Such observations may additionally provide useful information about the analyzed datasets since they usually differ from the main group structure.

In contrast to RSKC, we introduce a robust and sparse k-means-based procedure that is capable of finding the true underlying structure in very complex data, i.e. data containing clusters, outliers, and noise variables simultaneously. The presented k-means-based algorithm incorporates a weighting function employing a measure of outlyingness in order to automatically assign a weight to each observation. While a high weight indicates that an observation is part of a cluster, a low weight refers to a potential outlier. The advantage of using a weighting function is that we do not have to pre-specify any trimming level as for trimming-based approaches. To exclude noise variables, we use a lasso-type penalty imposed on the variable weights in an objective function adjusted by observation weights. In order to correctly detect groups, we eventually propose a framework aiming at the determination of the optimal parameters, such as the degree of sparsity and the number of clusters.

The rest of this paper is organized as follows. Sect. 2 briefly reviews k-means-based clustering approaches and motivates the proposed clustering procedure which is described in detail in Sect. 3. The parameter selection is presented in Sect. 4. Section 5 describes the evaluation setup. The algorithm is thoroughly tested on simulated data sets in Sect. 6 and eventually compared to other k-means-based clustering methods on a real-world dataset in Sect. 7. Section 8 concludes the paper.

2 *k*-means-based algorithms

Despite the large number of developed clustering procedures, k-means remains one of the most popular and simplest partition algorithms (Jain 2010). Given a data matrix $\mathbf{X} = \{x_{ij}\}, i = 1, \ldots, n,\ j = 1, \ldots, p$, with n observations described by p variables, the task of finding k clusters based on k-means was originally established using the within-cluster sum of squares W^k for the given number of clusters k as

$$\begin{aligned} W_j^k &= \sum_{r=1}^{k} \sum_{i \in K_r} (x_{ij} - m_{jr})^2, \\ W^k &= \sum_{j=1}^{p} W_j^k \rightarrow \min_{K_1, \ldots, K_k}, \end{aligned} \tag{1}$$

where W_j^k corresponds to the within-cluster sum of squares in the jth variable and the set K_r contains the indices of the observations assigned to the rth cluster, for $r = 1, \ldots, k$. Note that such an optimization problem can also be reformulated with respect to the between-cluster sum of squares B^k (Witten and Tibshirani 2010) as

$$B_j^k = \sum_{i=1}^{n}(x_{ij} - m_j)^2 - \sum_{r=1}^{k}\sum_{i\in K_r}(x_{ij} - m_{jr})^2,$$
$$B^k = \sum_{j=1}^{p} B_j^k \rightarrow \max_{K_1,\ldots,K_k}, \tag{2}$$

where B_j^k denotes B^k in the jth variable, m_j is the jth coordinate of the overall data center, and m_{jr} denotes the center of the rth cluster in the jth variable.

Although k-means is very popular, it has several disadvantages that need to be taken into account when developing a clustering procedure. The first drawback of k-means is the random initialization of cluster centers, which may lead to non-optimal solutions. This can be overcome by using an appropriate initialization method; an overview of such approaches can be found in a study by Celebi et al. (2013). For our method, we incorporate the ROBIN (ROBust INitialization) approach by Mohammad et al. (2009). The method is able to find optimal centers in a small number of runs unlike the original k-means. ROBIN seeks for k initial centers that are located in the most dense region and are simultaneously far away from each other in order to avoid the selection of outliers as initial centers. In order to identify the observations in highly dense regions, ROBIN uses LOF (Local Outlier Factor) proposed by Breunig et al. (2000). LOF was primarily introduced to measure a degree of outlyingness of an observation with respect to its q nearest observations, i.e. neighbors, in data where observations tend to form groups. The outlyingness of an observation $\mathbf{x}_i$ is defined as

$$lof_q(\mathbf{x}_i) = \frac{1}{N_q(\mathbf{x}_i)} \sum_{\mathbf{x}\in N_q(\mathbf{x}_i)} \frac{lrd_q(\mathbf{x})}{lrd_q(\mathbf{x}_i)}, \tag{3}$$

where $N_q(\mathbf{x}_i)$ refers to the local neighborhood of $\mathbf{x}_i$ spanned by its q neighbors and $lrd_q(\mathbf{x}_i)$ denotes the so-called local (reachability) density of $\mathbf{x}_i$ reflecting how far $\mathbf{x}_i$ is from its q neighbors on average. The resulting outlyingness, $lof_q(\mathbf{x}_i)$, of an observation $\mathbf{x}_i$ close to 1 indicates that $\mathbf{x}_i$ is potentially part of a cluster and, therefore, a candidate for an initial cluster center, as proposed by ROBIN. In contrast, $lof_q(\mathbf{x}_i) \gg 1$ suggests that $\mathbf{x}_i$ is a possible outlier and thus $\mathbf{x}_i$ should not be considered as an initial center. ROBIN searches for the clusters centers subsequently in the set of observations for which $lof_q(\mathbf{x}_i) > 1.1$. The first center is selected randomly and the next centers are those with the largest distances to all previously chosen.

The second limitation of k-means is the employed sample mean that suffers from a lack of robustness. As a result, k-means is also not resistant against outliers and even a single deviating observation can affect the final clustering solution (Garcia-Escudero and Gordaliza 1999). In order to robustify k-means, Cuesta-Albertos et al. (1997) proposed a trimmed version defined as

$$
\begin{aligned}
{}^{t}B_j^k &= \sum_{i\in L}(x_{ij}-m_j)^2 - \sum_{r=1}^{k}\sum_{i\in K_r\cap L}(x_{ij}-m_{jr})^2,\\
{}^{t}B^k &= \sum_{j=1}^{p} {}^{t}B_j^k \rightarrow \max_{K_1,\ldots,K_k,L},
\end{aligned}
\tag{4}
$$

where ${}^{t}B^k = \sum_{j=1}^{p} {}^{t}B_j^k$ represents the between-cluster sum of squares calculated on the untrimmed observations, L denotes the set containing indices of $[n(1-\alpha)]$ (untrimmed) observations that have the smallest distance to their closest cluster center. The symbol $[\cdot]$ stands for the integer part of a real number and α is the trimming level. The maximization (4) can also be reformulated by incorporating binary observation weights ${}^{t}v_i$ as

$$
{}^{t}B^k = \sum_{j=1}^{p}\left\{\sum_{i=1}^{n} {}^{t}v_i(x_{ij}-m_j)^2 - \sum_{r=1}^{k}\sum_{i\in K_r} {}^{t}v_i(x_{ij}-m_{jr})^2\right\} \rightarrow \max_{K_1,\ldots,K_k}, \tag{5}
$$

where $m_j = \frac{1}{\sum_{i=1}^{n} {}^{t}v_i}\sum_{i=1}^{n} {}^{t}v_i x_{ij}$, m_{jr} are calculated analogously for the rth cluster, and the binary observation weights ${}^{t}v_i$ are defined as

$$
{}^{t}v_i = \begin{cases} 0, & \text{if } d_i \geq d_{([n(1-\alpha)])} \\ 1, & \text{if } d_i < d_{([n(1-\alpha)])}, \end{cases} \tag{6}
$$

where d_i is the distance between x_i and its nearest cluster center, $d_{(l)}$ are the order statistics, i.e, $d_{(1)} \leq d_{(2)} \leq \cdots \leq d_{(n)}$, α is the pre-specified trimming level. While the observations with ${}^{t}v_i = 1$ are considered as untrimmed, the observations with ${}^{t}v_i = 0$ are marked as trimmed and discarded from the calculation of ${}^{t}B^k$. Such a robustification excludes the α fraction of observations, i.e. potential outliers, for calculating the cluster centers in order to achieve an accurate clustering solution if α is chosen correctly according to the true outlier proportion. Determining α may however be problematic for real-world data. In order to avoid the parameter-dependent robust k-means, we propose to incorporate a measurement of outlyingness which leads to a clear decision on determining outliers. Such a concept was introduced by Filzmoser et al. (2008) in case of a one-group data structure resulting in a more sophisticated choice of observation weights. The weights reflect how much an observation is outlying on the [0, 1]-scale with a low weight indicating a potential outlier. We incorporate the concept of such weights in k-means in order to robustify the method in such a way that no parameter pre-specification is required.

The last disadvantage of k-means occurs when a group structure is detectable only in a small subset of variables. In order to find such variables, Witten and Tibshirani (2010) introduced a framework for sparse k-means based on a lasso-type penalty leading to the problem of maximizing the weighted B^k for a given k and a sparsity parameter s as

$$B^{sk} = \sum_{j=1}^{p} w_j B_j^k \rightarrow \max_{K_1,\ldots,K_k,\mathbf{w}}, \tag{7}$$

subject to $||\mathbf{w}||^2 \leq 1$, $||\mathbf{w}||_1 \leq s$ for $\mathbf{w} = \{w_j \geq 0\}\, \forall j$ and $s \in (1, \sqrt{p}\,]$, which can be solved in an iterative way as proposed by Witten and Tibshirani (2010). The parameter s controls the degree of sparsity in the variable weight vector, i.e. the values of w_j. The more important (informative) the jth variable, the higher the value of w_j. Our method also uses a lasso-type penalty in the objective function, but the value of B^{sk} is additionally adjusted by observation weights in order to achieve robustness. Although the proposed method is similar to RSKC by Kondo et al. (2016), our procedure can be seen as a better alternative since no trimming level is required. In addition, we aim at analyzing the data structure more thoroughly, i.e. discovering clusters, outliers, and informative variables simultaneously.

3 The proposed algorithm

The introduced method is an iterative three-step approach. In the first step, k-means employing a weighting function is applied on the data space spanned by the variables with some contribution to a cluster separation [i.e. with the variables having $w_j > 0$, see Eq. (7)]. The incorporated weighting function robustifies k-means and results in observation weights reflecting the outlyingness. The second step aims at updating the variable weights with respect to both clusters and observation weights from the first step. The two steps are iteratively repeated until the variable weights stabilize. In the third step, the observations are clustered with respect to the identified informative variables and the observations with small weights are classified as outliers. The detailed description of the algorithm is given in the following subsections.

3.1 Step 1: Downweighting outlying observations

The aim of the first step is to robustify k-means by incorporating a weighing function in order to downweight the influence of potential outliers. Assuming that the number of clusters k is known, we apply ROBIN with a default of 10 nearest neighbors (Mohammad et al. 2009), i.e. $q = 10$ in Eq. (8), on weighted data, ${}_w\mathbf{X} = \{{}_w\mathbf{x}_i\} = \{w_j\, x_{ij}\}, \forall i, \forall j$, where $\mathbf{w} = \{w_j = 1/\sqrt{p}\}, \forall j$. Note that the initial values $w_j = 1/\sqrt{p}$ are considered only in the first iteration as recommended by Witten and Tibshirani (2010), but in the next iteration w_j will be already different and will better reflect the contribution to a cluster separation.

After applying ROBIN, each observation is assigned to its closest cluster center leading to the corresponding cluster membership $K_1, \ldots, K_k$. We then propose to apply a weighting function on the detected clusters to reveal outliers. The weighting function should be a monotonic decreasing function using an outlyingness measure as an argument in order to obtain observation weights that range between 0 and 1, with a low weight indicating a potential outlier. Hence, it is essential to choose both a suitable outlyingness measure and an appropriate weighting function.

A naive approach is to calculate the Euclidean distance of an observation to its closest cluster center. However, using the Euclidean distance provides the information about how far an observation is from its closest center rather than how much an observation deviates or to what degree it is an outlier. In fact, such information can be easily obtained by applying LOF on each detected cluster as $lof({}_w\mathbf{x}_i) := lof_q({}_w\mathbf{x}_i), i \in K_r, \forall r$, with $q = 10$. The choice and role of q is discussed later on in Sect. 4. The LOF scores calculated according to Eq. (8) are then standardized as

$$lof_i^* = \frac{lof({}_w\mathbf{x}_i) - \text{mean}(lof({}_w\mathbf{x}_i), i \in K_r)}{\text{sd}(lof({}_w\mathbf{x}_i), i \in K_r)} \tag{8}$$

to be suitable for the weighting function with the mentioned properties. Preliminary studies indicated good empirical results when the observation weights, denoted as $v_i^{(1)}$, were obtained using the translated bi-weight function (Rocke 1996) as follows

$$v_i^{(1)} = \begin{cases} 0, & lof_i^* \geq c \\ \left(1 - \left(\frac{lof_i^* - M}{c - M}\right)^2\right)^2, & M < lof_i^* < c, \\ 1, & lof_i^* \leq M \end{cases} \tag{9}$$

for $i \in K_r, r = 1, \ldots, k$, and $c > M$. Based on the preliminary studies, the values of the parameters are taken as $c = 2$ and $M = \text{med}(lof_i^*, i \in K_r) + \text{MAD}(lof_i^*, i \in K_r)$, where med stands for the median and MAD refers to the median absolute deviation. The obtained weights correspond to the measure of outlyingness values in [0, 1]. While a value close to 1 indicates that an observation is part of a cluster, $v_i^{(1)} \approx 0$ suggests that $\mathbf{x}_i$ is an outlier with respect to the detected cluster. The weights based on LOF are smoother and better express the degree of deviation than e.g. using a simple Euclidean distance of an observation to the closest cluster as employed in the trimmed k-means [see Eq. (6)] or RSKC. In addition, the weights should be more robust against elliptically-shaped clusters due to the properties of LOF; see Breunig et al. (2000). If the shape of a cluster is slightly elliptical, RSKC might exclude observations which are further away from the cluster center but still part of a cluster.

After assigning weights to observations from each detected cluster according to (8) and (9), we plug the weights $v_i^{(1)}$ into the weighted between-cluster sum of squares for a given $\mathbf{w}$, and optimize the cluster assignment as

$$\begin{aligned} {}^{v^{(1)}}B_j^k = &\sum_{i=1}^{n} v_i^{(1)} \left(x_{ij} - \frac{1}{\sum_{i=1}^{n} v_i^{(1)}} \sum_{i=1}^{n} v_i^{(1)} x_{ij}\right)^2 \\ &- \sum_{r=1}^{k} \sum_{i \in K_r} v_i^{(1)} \left(x_{ij} - \frac{1}{\sum_{i \in K_r} v_i^{(1)}} \sum_{i \in K_r} v_i^{(1)} x_{ij}\right)^2 \end{aligned} \tag{10}$$

$$\sum_{j=1}^{p} w_j \, {}^{v^{(1)}}B_j^k \rightarrow \max_{K_1, \ldots, K_k} \tag{11}$$

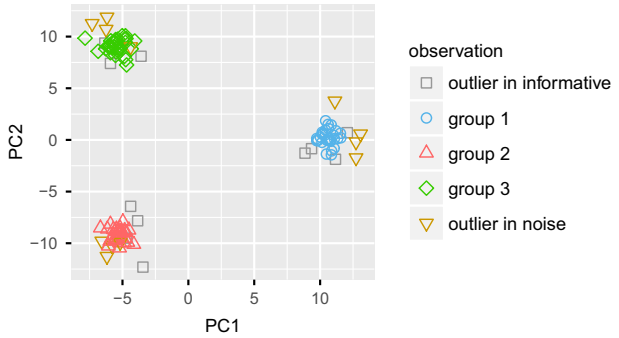

Fig. 1 The generated dataset shown in the principal component space. The observations from 3 groups, outliers placed in informative and noise variables are displayed in different colors and symbols (color figure online)

in order to robustify k-means. We can clearly see from (10) that if an observation is a potential outlier, i.e $v_i^{(1)} \approx 0$, the distance of such an observation to its closest cluster center is downweighted by the corresponding value of $v_i^{(1)}$. In contrast, an observation with $v_i^{(1)} \approx 1$ highly contributes to the maximization. The observation weights are also used to determine the next cluster centers, i.e. $\frac{1}{\sum_{i \in K_r} v_i^{(1)}} \sum_{i \in K_r} v_i^{(1)} x_{ij}, \forall r$, in a robust way by using the weighted mean of observations in each coordinate. The cluster centers with the corresponding cluster assignment are iteratively updated until a local optimum is reached during a certain number of iterations in the sense of maximization of (11). In our experiments the method is allowed to search for the local optimum during 15 iterations, but also a higher number can be considered. Note that the local optimum is achieved on the weighted data, i.e. in a data space spanned by the variable vector with $w_j > 0$ adjusted by the values of w_j.

We illustrate the efficiency of the weighting function on an example dataset that consists of three groups with the same sizes of 40 observations. The group structure is described by 50 variables leading to high-dimensional low sample size groups. We add 750 noise variables and contaminate 10% of the observations from each group in the informative variables and in 75 noise variables; a detailed description of the data setup is provided in Sect. 6 and corresponds to the first simulation study. Figure 1 visualizes the generated dataset in the space spanned by the first two principal components; the group membership and outliers are differentiated by colors and symbols. The final weights, obtained during two iterations given the initial cluster centers by ROBIN, are shown in Fig. 2 in decreasing order to visualize the shape of the weighting function. Importantly, the observation weights are calculated in the data space defined by 50 informative variables. In other words, we now assume that $\mathbf{w}$ is known beforehand in order to demonstrate the concept of the weighting function. We can see in Fig. 2 that all observations from group 3 are correctly assigned to cluster 1 because no observations from group 3 are visible in the following plots. The plot particularly indicates that the weighting function works properly since all non-outliers obtain a weight around 1. In contrast, outliers placed in informative variables receive a weight around 0 and can thus be easily identified. A similar conclusion can be made in case of the other

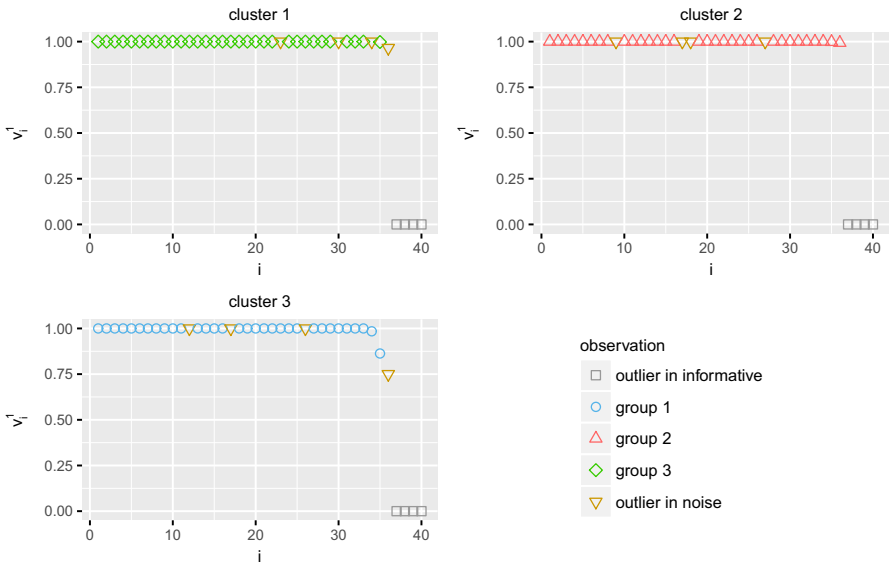

Fig. 2 Illustration of incorporating the weighting function in k-means in order to reveal outliers as observations with low $v_i^{(1)}$ and to detect 3 clusters on the weighted data

two clusters. The plots may suggest that the non-outliers with a weight smaller than 1 could be located on the edge of a cluster or slightly further from the other clustered observations. However, we cannot reveal outliers placed in noise variables as indicated by their weights equal to 1, since the noise variables are not involved in the clustering due to their zero weights.

In order to identify outliers in noise variables as well, we additionally apply the proposed weighting function on unweighted data clusters, consisting of the data matrices $\mathbf{X}_r = \{\mathbf{x}_i\}, i \in K_r, r = 1, \ldots, k$, leading to the second observation weights $v_i^{(2)}$. Note that $\mathbf{X}_r = \{w_j x_{ij}\}$, where $w_j = 1, \forall j$, and $i \in K_r, r = 1, \ldots, k$. Figure 3 shows the second resulting observation weights obtained on the data example shown in Fig. 1. The three plots clearly indicate that all outliers placed in noise variables receive considerably lower weights in contrast to both non-outliers and outliers present in informative variables.

As a consequence of applying the weighting function for the second time, each observation has two weights, $v_i^{(1)}$ and $v_i^{(2)}$, which are finally combined in a single weight

$$v_i = \min \left\{ v_i^{(1)}, v_i^{(2)} \right\}. \tag{12}$$

Determining v_i in this way ensures that all outliers receive low weights and that we can easily identify whether or not an observation is an outlier as indicated by zero weights for all outliers in Fig. 4. The obtained weights v_i are used in the next step aiming at selecting the variables which are informative for the cluster separation.

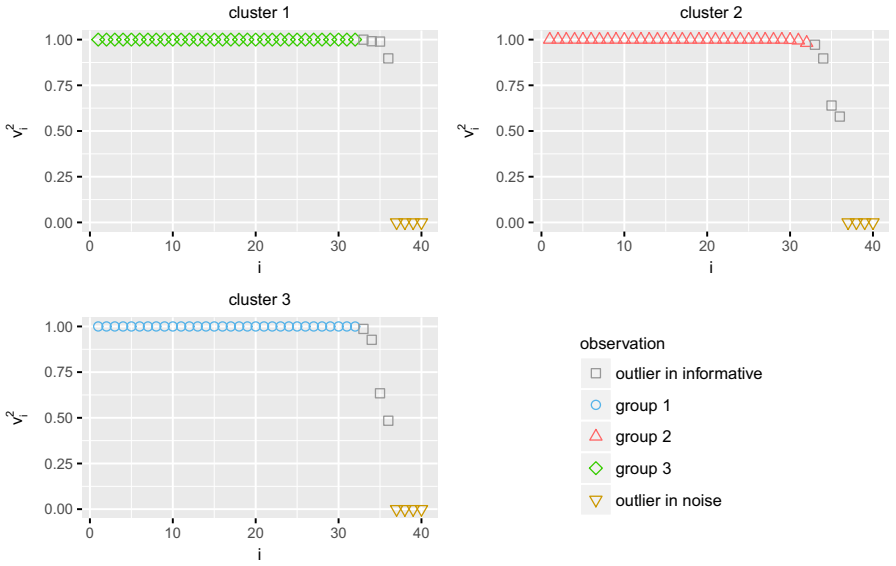

Fig. 3 Illustration of applying the weighting function on the 3 unweighted data clusters in order to reveal outliers in noise variables as observations with low $v_i^{(2)}$

3.2 Step 2: Variable selection

The purpose of the second step is to update w_j according to the maximization of (7) for a given sparsity parameter s and the observation weights v_i from Eq. (9). Incorporating v_i assures that the variable selection is not affected by outliers. Indeed, the presence of an outlier apparent even in one variable can considerably increase the between-cluster sum of squares. As a result, such a variable receives a high weight although the variable does not contribute to the cluster separation but rather to the separation between an outlier and a cluster (Kondo et al. 2016).

Therefore, for the obtained cluster assignment $K_1,\ldots, K_k$, the observation weights v_i from the first step, and for a given s, we update the weights w_j according to

$$^{v}B^{ks} = \sum_{j=1}^{p} w_j \, {}^{v}B_j^k \rightarrow \max_{||\mathbf{w}||^2 \leq 1, ||\mathbf{w}||_1 \leq s}, \tag{13}$$

where ${}^{v}B_j^k$ corresponds to Eq. (10) with v_i from Eq. (9) instead. In order to optimize (13) with respect to $\mathbf{w}$ for a given tuning parameter s, we follow the procedure suggested by Witten and Tibshirani (2010). Whereas small s leads to high sparsity, i.e. $w_j = 0$ for most variables, a high value of s results in almost no sparsity corresponding to $w_j > 0$ for most variables. High w_j suggests that the jth variable is informative and, thus, it contributes to the maximization of (13). In contrast, $w_j = 0$ indicates that the jth variable is not informative for the cluster separation and it is thus excluded in (13).

Once the variable weights $\mathbf{w}$ are updated, the first iteration is completed and the algorithm continues with the first step with respect to updated weights w_j. This means

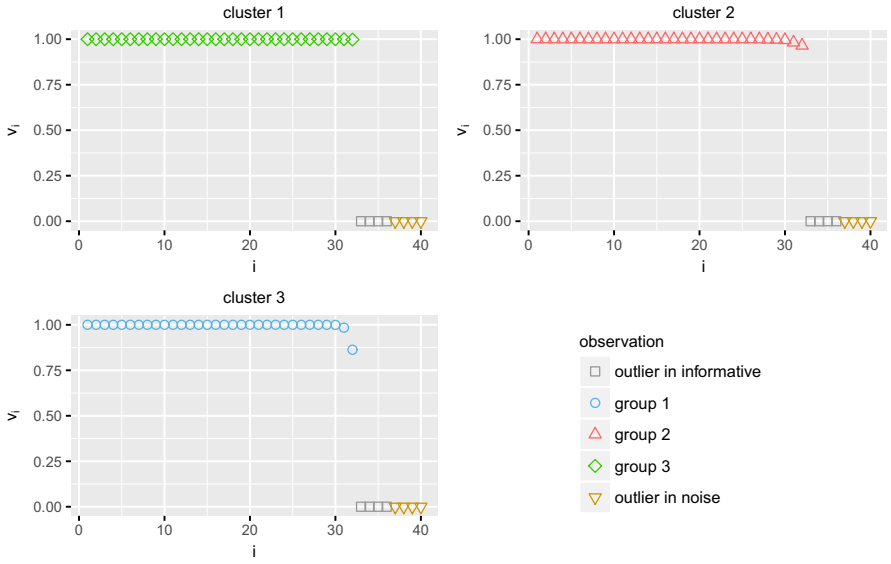

Fig. 4 Illustration of combining the two observations weights leading to the final observation weights v_i calculated on 3 clusters

that the ROBIN approach is again applied on ${}_w\mathbf{X} = \{{}_w\mathbf{x}_i\}$ with updated $\mathbf{w}$ in order to find the next cluster centers. The reason for the re-initialization is that ROBIN is not primarily designed to deal with a large number of noise variables. Therefore, the selection of the first cluster centers is very likely to be affected by noise variables due to $\mathbf{w} = \{w_j = 1/\sqrt{p}\}, \forall j$ in the first step. After obtaining the next centers, the method continues as described. The two steps of the proposed approach are iteratively repeated until there are no considerable changes in ${}^v B^{ks}$. Such a stopping criterion is closely related to the criterion employed by Witten and Tibshirani (2010).

3.3 Step 3: Detection of groups and outliers

The last step aims at determining the cluster membership $K_1, \ldots, K_k$ by assigning observations to their closest cluster center in the data space spanned by variables with $w_j > 0$, adjusted by their corresponding final weights. We estimate the final observation weights v_i as described in Sect. 3.1 in order to classify observations with low weights as outliers. This classification can be made based on visualization of the resulting observation weights against the corresponding observation index, as shown in Fig. 4, and the following search for a cut-off value which clearly separates low weights from high weights. Nevertheless, we recommend to use $v_i < 0.5$ for the identification of outliers as we observed good empirical results for such a choice.

3.4 Summary of the algorithm

In order to better understand the proposed algorithm, a brief summary is provided below.

Phase 1: Calculation of observation weights v_i.

Step 1: Identify cluster membership with respect to a given number k of cluster centers determined by the ROBIN approach on weighted data ${}_w\mathbf{X}$. Keep the cluster membership for Phase 2.
Step 2: Calculate observation weights $v_i^{(1)}$ according to Eq. (9) on weighted data clusters ${}_w\mathbf{x}_i$, $i \in K_r$.
Step 3: Calculate observation weights $v_i^{(2)}$ according to Eq. (9) on unweighted data clusters $\mathbf{x}_i$, $i \in K_r$.
Step 4: Combine $v_i^{(1)}$ and $v_i^{(2)}$ as in Eq. (12) to obtain observation weights v_i. Keep v_i for Phase 2.

Phase 2: Calculation of variable weights w_j.

Step 1: Calculate ${}^vB_j^k$ according to Eq. (11) with respect to v_i, the cluster membership from Phase 1 and given sparsity parameter s.
Step 2: Calculate variable weights w_j according to Eq. (13) for a given s with ${}^vB_j^k$ calculated in Step 1.

Phase 1 and Phase 2 are repeated until convergence is achieved.

Phase 3: Identification of outliers informative variables, and clusters.

Step 1: Identify informative variables with final $w_j > 0$.
Step 2: Identify the final cluster membership in the space spanned by informative variables.
Step 2: Calculate final v_i as described in Phase 1 with respect to final cluster membership and declare outliers as observations with $v_i > 0.5$.

4 Selection of parameters

This section provides details about the selection of the parameters, i.e. k, s, q, and c. While the number of clusters k with the sparsity parameter s are required for clustering, the number of the nearest neighbors q and s are essential for the proposed weighting function in order to sufficiently detect outliers. In this section, we first present an automatic procedure to select k and s. Then, we discuss the fixed choices of q and c.

4.1 The number of clusters *k* and sparsity parameter *s*

We have so far assumed pre-knowledge about the number of clusters k as well as the tuning parameter s determining the variable weights w_j. Such information is usually not available beforehand for most real-world data and, therefore, there is a need for a systematic way of estimating both parameters. The problem of selecting the optimal k has been widely studied for data where the assumption is that all variables are involved in data clustering; an overview of such procedures can be found in the studies by Sugar and James (2003), Xu and Wunsch (2005). However, we have not found much work dedicated to the optimization of k in case that the sizes of groups are much lower than the dimensionality of the data space describing the group structure and at the same time the group structure is hidden in a large number of noise variables.

We discuss the effect when k is optimized with and without taking the contribution of variables into account using the gap statistic (Tibshirani et al. 2001). The gap statistic,

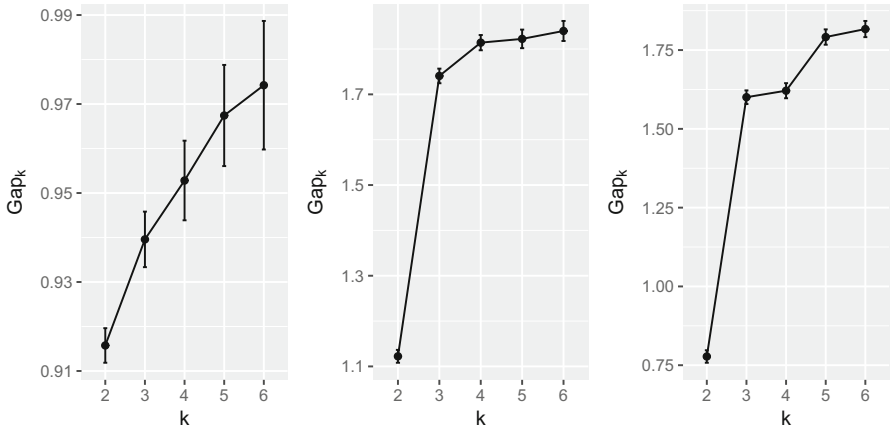

Fig. 5 The effect of noise variables and outliers when estimating the optimal number of clusters. The values of Gap_k applied on a data set with both noise variables and outliers (left); the resulting Gap_k from a dataset where the effect of noise variables is eliminated (middle); the obtained Gap_k when both noise variables and outliers are neglected (right)

Gap_k, is calculated for a clustering solution obtained by a clustering algorithm, e.g. k-means, for a given k and can be formulated as

$$Gap_k = \sum_{j=1}^{p} w_j \left(\frac{1}{A} \sum_{a=1}^{A} \log \left({}_a W_j^k \right) - \log \left(W_j^k \right) \right), \tag{14}$$

where $w_j = 1, \forall j$ since all variables are assumed to contribute equally, ${}_a W^k = \sum_j {}_a W_j^k$ corresponds to $W^k = \sum_j W_j^k$ calculated on the clustering solution obtained on the dataset with independently permuted observations in each variable (Witten and Tibshirani 2010), and A represents the number of permuted datasets. In our experiments we consider $A = 10$. Gap_k is generally calculated for a clustering solution with varying k and the optimal number of clusters is chosen as the smallest k for which $Gap_k \geq Gap_{k+1} - se_{k+1}$ is fulfilled (Tibshirani et al. 2001), where se_k denotes the standard error of $\log({}_a W^k)$. From (14) it is obvious that Gap_k does not only depend on k but also on $\mathbf{w}$ representing the contribution of each variable. Since all variables are assumed to be informative, Gap_k might be considerably affected if a dataset contains a large number of noise variables. Moreover, the presence of deviating observations can lead to an unreliable decision on k as well.

Figure 5 demonstrates the effect of noise variables and outliers on the choice of k based on the gap statistic. We consider the same data example as in Sect. 3.3 and apply k-means with ROBIN initialization for the numbers of clusters $k = 2, \ldots, 6$. The gap statistic is calculated for each clustering solution in order to select the optimal k as described above. Figure 5 (left) shows the values of Gap_k with the corresponding standard errors calculated on the data example with both outliers and noise variables. As expected, the presence of both disturbing factors leads to a wrong choice of the optimal k corresponding to 5 clusters. Moreover, even if only the 50 informative variables are taken into account, the choice of k is also influenced by outliers as

illustrated in Fig. 5 (middle) resulting in $k = 4$. In contrast, Fig. 5 (right) shows Gap_k when downweighting outlying observations and noise variables leading to a correct decision, i.e $k = 3$.

The example indicates that both disturbing factors have to be considered when selecting an optimal k for k-means. In the proposed k-means-based clustering approach, we directly downweight the effect of outliers by observation weights v_i. However, the impact of noise variables, which is reflected by their corresponding variable weights w_j, can be neglected only if the sparsity parameter s, see Eq. (13), is correctly selected. In order to optimize s, Witten and Tibshirani (2010) introduced the gap statistic Gap_s, which is defined for given k as

$$Gap_s = \log(B^{sk}) - \frac{1}{A}\sum_{a=1}^{A}\log({}_aB^{sk}), \tag{15}$$

where ${}_aB^{sk}$ denotes the weighted between-cluster sum of squares calculated, compare (7), with respect to a clustering solution obtained on a permuted dataset. Obviously, the calculation of Gap_s is impossible if the number of clusters k is unknown which is often the case for real data. Moreover, the presence of outliers might also influence the correct estimation of s. Therefore, we propose to adjust Gap_s by observation weights v_i in order to downweight the influence of outliers leading to the modified gap statistic ${}^vGap_{sk}$ calculated as

$${}^vGap_{sk} = \log({}^vB^{sk}) - \frac{1}{A}\sum_{a=1}^{A}\log({}^v_aB^{sk}), \tag{16}$$

where ${}^v_aB^{sk}$ represents ${}^vB^{sk}$ obtained on a permuted dataset. We calculate ${}^vGap_{sk}$ for a clustering solution not only with various s but also various k in order to first optimize the degree of sparsity s for each k. The value of ${}^vGap_{s^*k}$ for the optimal parameter s^* is compared with the largest ${}^vGap_{sk}$ such that ${}^vGap_{s^*k} \geq {}^vGap_{sk} - se_{sk}$, where se_k refers to the standard error of $\log({}^v_aB^{sk})$. The optimization of s leads to k values of ${}^vGap_{s^*k}$ for which the largest value corresponds to an optimal k.

Figure 6 depicts the gap statistic for both tuning parameters when applying the proposed method on the data example in Sect. 3.3 with $k = 2, \ldots, 6$. The value of s starts at 1.1 and increases in steps of 0.5 to such a value that leads to no sparsity in the variable weights, i.e $w_j \neq 0, \forall j$. We show the optimal s for each k by larger symbols in Fig. 6. As expected, the optimal degree of sparsity s differs almost for all k. We select the optimal parameter setting which leads to the largest ${}^vGap_{s^*k}$ resulting in $k = 3$ and $s = 6.6$. The plot additionally illustrates that a smaller choice of s, e.g. $s = 4.1$, results in an incorrect number of clusters when following the rule for optimizing k based on Gap_k according to Tibshirani et al. (2001). This supports the fact that both parameters need to be optimized at the same time in order to correctly identify groups.

The selected choices $k = 3$ and $s = 6.6$ correspond to the correct number of clusters as well as appropriate values of w_j leading to non-zero weights for all 50 informative

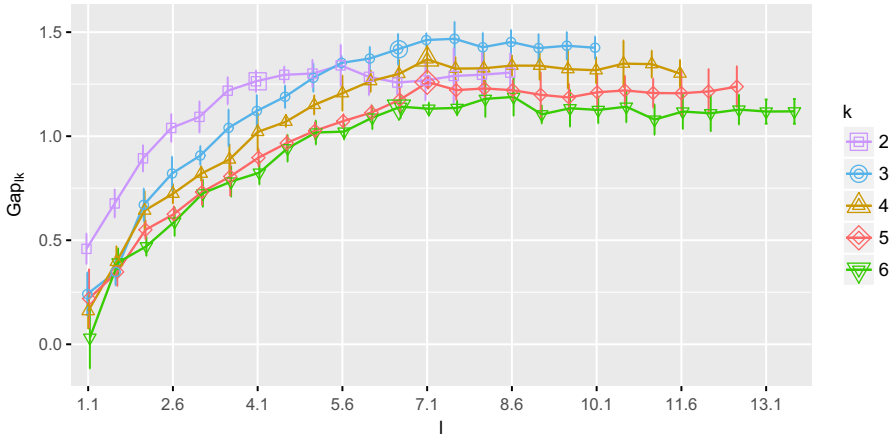

Fig. 6 Selection of the tuning parameter s and the number of clusters k based on ${}^{v}Gap_{sk}$

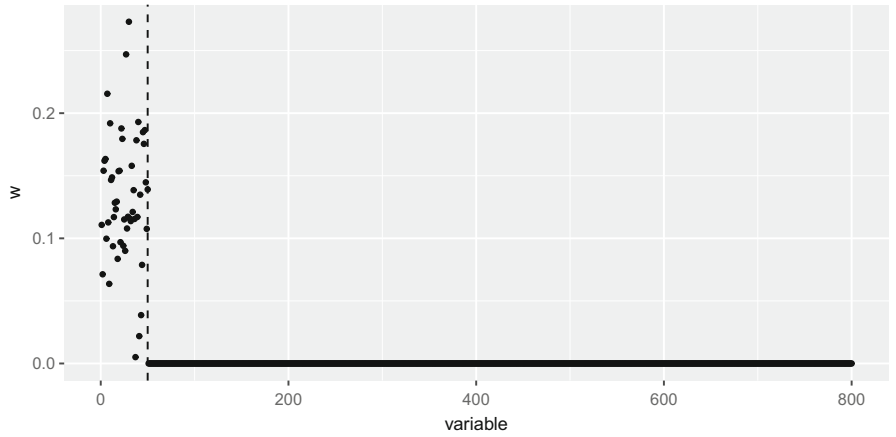

Fig. 7 The variable weights corresponding to the optimal $s = 6.6$ and $k = 3$, the first 50 informative variables are separated by dashed line from the rest 750 noise variables

variables as shown in Fig. 7. Considering higher values of s, more and more noise variables obtain non-zero weights. In contrast, lower values of s lead to less variables with non-zero weights.

4.2 The number of nearest neighbors *q* and the constant *c*

The choice of $q = 10$ and $c = 2$ has already been mentioned while presenting the proposed clustering method. Both values were selected based on our preliminary studies as well as on the properties of LOF (Breunig et al. 2000) employed in the weighting function, see (8) and (9). LOF is also employed in the ROBIN initialization method.

According to Breunig et al. (2000), the choice of q needs to be selected based on the sizes of clusters in order to keep the LOF properties, such that $lof(\mathbf{x}_i) >> 1$ indicates outliers whereas values around 1 pointing on clustered observations. Indeed, the value of q should not be smaller nor larger than the size of any cluster in a dataset. This heuristic guideline has been developed for a dataset where clusters are of different sizes. In our algorithm, LOF is applied on each cluster separately, which makes it easier to decide on q. In addition, Breunig et al. (2000) recommend to set q at least to 10 to avoid fluctuations in LOF scores. Based on these facts, we decided to set $q = 10$ in the weighting function. The same value is also taken for ROBIN. Such choice, however, implies that initial centers of very small clusters might not be correctly detected. Nevertheless, the observations of such small clusters could be considered and declared as outliers by the proposed method.

The role of c in the weighting function is to trim all observations with a standardized LOF score greater than c. Accordingly, such observations are downweighted to zero. Therefore, the parameter c plays a similar role as a trimming level. Nevertheless, using $c = 2$ discards the most outlying observations independently of specifying how many of them should be trimmed. In contrast, using a trimming level removes the pre-specified number of observations no matter if these observations actually deviate. In case c is slightly higher, there is a next boundary, i.e. M in Eq. (9), being data-derived from LOF. M ensures that observations with LOF scores lower than M are downweighted as well. This is very helpfully in case that c is too large. The opposite case, i.e. if c smaller 2, can be selected, but $c > M$ has to be fulfilled. However, we recommend to select $c = 2$ because it worked well in all our experiments.

5 Evaluation setup

We evaluate the performance of the proposed method in terms of the clustering solution, outlier detection, and the identification of informative variables. The clustering solution is evaluated based on the Classification Error Rate (CER), also used by Witten and Tibshirani (2010). CER measures to which extent clustering and group memberships disagree on any pair of observations. While CER = 0 refers to the best cluster solution, CER = 1 corresponds to the poorest performance. In order to evaluate the ability of our method to detect outliers, we report the mean value of observation weights v_i separately for the true non-outliers, i.e $\bar{v}^{nonout}$, and outliers, denoted as $\bar{v}^{out}$. The weights for outliers are supposed to be considerably lower than the weights for non-outliers. Since we recommend to use the final weights v_i for classifying outliers as the observations with $v_i < 0.5$, we calculate True Positive and False Positive Rates (TPR and FPR) ranging between 0 and 1. TPR is defined as the proportion of the number of correctly identified outliers and the actual number of outliers present in a given dataset. High TPR indicates a good ability to identify outliers while low TPR demonstrates poor performance. FPR is calculated as the ratio between the number of non-outliers wrongly declared as outliers and the number of the actual non-outliers in an analyzed dataset. Hence, low values of FPR are preferable over high values. The performance regarding the variable selection is evaluated by comparing the mean value of w_j for informative variables, $\bar{w}^{inf}$, with the mean value of w_j that are different from zero,

denoted as $\bar{w}^{non0}$. The higher and more similar the values, the better the ability to correctly select informative variables. We provide a similar evaluation for noise variables and calculate the mean of their weights, $\bar{w}^{noise}$, which is supposed to be close to zero.

Since the clustering procedure employs k-means, we compare the method with several existing k-means-based clustering algorithms, such as k-means (K),[1] trimmed k-means (TKC)[1] by Cuesta-Albertos et al. (1997), and sparse k-means (SKC)[2] by Witten and Tibshirani (2010). The proposed weighted robust and sparse k-means (WRSK) is also compared with trimmed and sparse k-means (RSKC)[2] by Kondo et al. (2016). Although our algorithm is designed in a similar way as RSKC, we avoid to pre-specify the trimming level by incorporating the proposed weighted function. Since no procedure for selecting the optimal k and s has been presented by Kondo et al. (2016) for RSKC in case that no information about data is available, we employ the modified gap statistic considering zero weights for trimmed observations and weights equal to one for untrimmed observations. Note that all trimming-based algorithms require for the pre-specification of a trimming level α, therefore, when applying these methods we consider α as the true percentage of outliers present in a simulated dataset and $\alpha = 0.10$ for real-world data as recommended by Kondo et al. (2016) being a suitable choice for most cases.

6 Simulation study

In this section, we explore the ability of the proposed clustering method to correctly reveal the complex data structure in three simulation studies. We first show the efficiency of the gap statistic to properly select s and k. Then, we test the method on the datasets containing various percentages of outliers. Finally, the proposed method is compared with several existing k-means-based approaches.

We now describe the general setting of the simulated datasets considered in the three studies. Each dataset consists of n observations described by the informative as well as uninformative part in terms of the group separation. The observations in the informative part form g groups of sizes $n_t, t = 1, \ldots, g$. The groups are described by p_{inf} variables and are generated following a Gaussian model with parameters $\boldsymbol{\mu}_t \in \mathbb{R}^{p_{inf}}$ and $\boldsymbol{\Sigma}_t \in \mathbb{R}^{p_{inf} \times p_{inf}}$. The elements of the mean vector $\boldsymbol{\mu}_t = (\mu_{t1}, \ldots, \mu_{tp_{inf}})$ are constructed as

$$\mu_{tj} = \begin{cases} \mu, & j = a_z, \\ 0, & \text{else} \end{cases} \tag{17}$$

where μ is randomly chosen from the uniform distribution in $[-6, -3] \cup [3, 6]$, i.e $U[-6, -3] \cup U[3, 6]$. a_z represents the arithmetic sequence defined as $a_{z+1} = a_z + g$, $a_1 = t$ meaning that the first nonzero element of $\boldsymbol{\mu}_t$ is placed on the tth position and the following nonzero elements, i.e. μ, are always on the position increased by g with respect to the previous index of the nonzero element. Considering, for example, 4 groups of 10 dimensions, the mean vectors of the first two clusters are constructed

[1] We employed the code implemented in the R package `RSKC` (Kondo et al. 2016).

[2] The used code for sparse k-means as available in the R package `sparcl` (Witten and Tibshirani 2013).

as $\boldsymbol{\mu}_1 = (\mu, 0, 0, 0, \mu, 0, 0, 0, \mu, 0, 0)$ and $\boldsymbol{\mu}_2 = (0, \mu, 0, 0, 0, \mu, 0, 0, 0, \mu, 0)$. The covariance matrix $\boldsymbol{\Sigma}_t$ is generated according to Campello et al. (2015) as

$$\boldsymbol{\Sigma}_t = \mathbf{Q} \begin{pmatrix} 1 & \rho_t & \dots & \rho_t \\ \rho_t & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \rho_t \\ \rho_t & \dots & \rho_t & 1 \end{pmatrix} \mathbf{Q}^\top, \tag{18}$$

where $\mathbf{Q}$ denotes a random rotation matrix satisfying $\mathbf{Q}^\top = \mathbf{Q}^{-1}$ and the off-diagonal elements ρ_t are random numbers from $U[0.1, 0.9]$ which are exclusively generated for each group. To the informative part, we also add p_{noise} noise variables that follow univariate standard normal distributions leading to a total dimensionality of $p = p_{inf} + p_{noise}$.

Such an obtained dataset is finally contaminated by replacing a certain percentage of observations in each group by outliers. We create two types of outliers in the informative variables. While uniformly distributed outliers are generated as random values from $U[-12, 6] \cup U[6, 12]$, the scattered outliers follow a Gaussian model with the same location as a group, i.e. $\boldsymbol{\mu}_t$, but a different covariance structure $\sigma\mathbf{I} \in \mathbb{R}^{p_{inf} \times p_{inf}}$. The parameter σ is randomly generated from an uniform distribution in $[3, 10]$. We also replace a certain proportion of observations from each group in the noise variables by uniformly distributed outliers, according to $U[-12, 6] \cup U[6, 12]$. Note that the observations contaminated in the informative variables differ from those in the noise variables. Furthermore, we always replace (contaminate) the first observations from each group in the informative variables, whereas observations in the noise variables are randomly selected for the following contamination.

6.1 Simulation 1: Selection of parameters

In the first study, we investigate the ability of the modified gap statistic to correctly select the number of clusters k and the sparsity parameter s when applying the introduced algorithm. We consider 100 datasets of 800 dimensions in which the first 50 variables describe the group structure. In order to explore the performance of the gap statistic, 3 situations with different numbers of groups are considered, i.e. $g = 3, 4, 5$. The sizes of the observations in the groups are randomly selected, ranging from 50 to 150. The contamination strategy corresponds to replacing the first 10% of observations from each group in all informative variables by scattered outliers. In contrast, the uniformly distributed outliers are placed in 75 randomly selected noise variables.

The proposed method is applied with $k = 2, \dots, 7$ and various s going from 1.1 up to $\sqrt{p}$ in steps of 0.5, in order to calculate the gap statistic and to select optimal parameters. The results are evaluated in terms of the estimated number of clusters and the evaluation measures described in Sect. 5. It should be noted that CER is calculated with respect to the group membership before contamination. Since each group is contaminated by scatter outliers, such outliers have the same location as a group and, therefore, they should be assigned to the corresponding group.

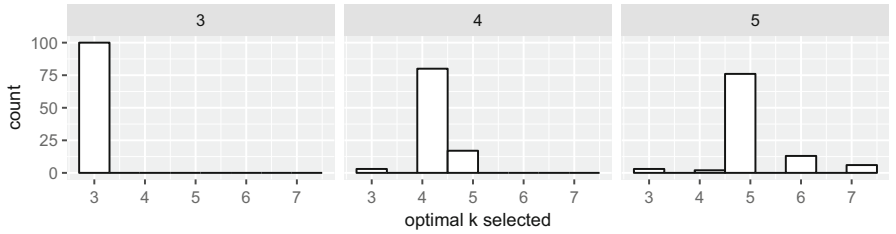

Fig. 8 Evaluation of the results in terms of the optimal k selected by the gap statistic, for different numbers of groups, i.e. $g = 3, 4, 5$. The reported values are based on 100 simulated datasets for each g

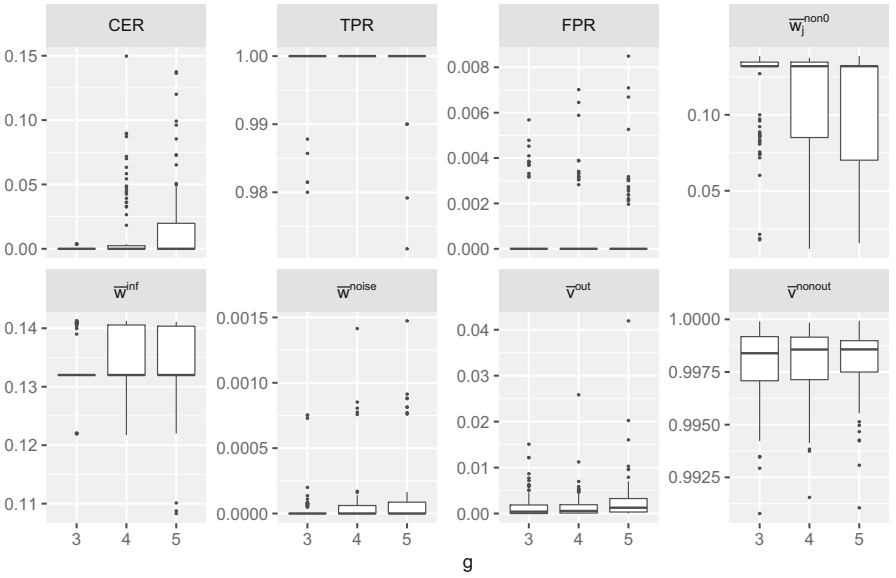

Fig. 9 Evaluation of the results based on the optimal parameter selection determined by the modified gap statistic, for different numbers of groups, i.e. $g = 3, 4, 5$. The reported values of the evaluation measures and the selected k represent all 100 simulated datasets. Note that the scale of the y-axes is not fixed in the plots in order to better show the performance of the methods in various settings

Figure 8 summarizes the resulting optimal k selected by the gap statistic as histograms, for the three different numbers of underlying groups ($g = 3, 4, 5$). While the gap statistic works perfectly in case of 3 groups, its performance gets slightly worse for a higher number of groups. Nevertheless, the last two histograms clearly indicate that the optimal k is correctly chosen in most cases.

Figure 9 summarizes the results based on the evaluation measures. In general, there is no clear dependence between the considered numbers of groups and the resulting values of evaluation measures. Overall, low CER indicate that the proposed procedure can correctly identify the group structure. In addition, high as well as similar values of $\bar{w}^{inf}$ and $\bar{w}^{non0}$ demonstrate the appropriate selection of s. Hence, it seems that most of the informative variables can be correctly identified. The high performance of variable selection is also supported by zero values of $\bar{w}^{noise}$ suggesting that the

method is able to discard all noise variables. We also evaluate the method regarding the detection of outliers. We can see that outliers receive on average considerably low weights, i. e. around 0, in contrast to non-outliers, i. e. around 1; compare $\bar{v}^{out}$ and $\bar{v}^{nonout}$. Therefore, classifying the observations with $v_i > 0.5$ as outliers seems to be a reasonable choice. Indeed, such a cut-off value leads to a great ability to identify outliers indicated by high TPR centered around 1 as well as low FPR centered around 0. Considering the values of the evaluation measures, we can conclude that the method as well as the parameter selection work efficiently.

6.2 Simulation 2: Resistance against outliers

The second simulation study aims at investigating how resistant the proposed method as well as the modified gap statistic are against various proportions of outliers. For this study, we generate 100 datasets that consist of 3 groups of different sizes ranging between 50 and 150 (randomly selected). The data space is defined by 170 informative variables and 830 noise variables, leading to 1000 dimensions in total. Overall, we consider 8 contamination strategies in terms of different percentages of outliers. The datasets in the first strategy are free of outliers. In contrast, the second strategy considers 5% of scatter outliers in all informative variables and no outliers in noise variables. The datasets in the remaining strategies are contaminated with 10%, 15%, 20%, 30%, and 40% scatter outliers, respectively, in the informative variables. In addition, the proportion of outliers in the 83 (10%) noise variables is always kept as 10%.

Again, the proposed algorithm is applied with the different numbers of clusters ($k = 2, 3, 4, 5, 6$) and various s. Subsequently, the gap statistic is employed to estimate the optimal parameter settings. The performance is finally evaluated by the measures described in Sect. 5 as well as the selected k. As in the previous study, we calculate CER by taking the true group membership before contamination into account.

Figure 10 shows the optimal number of clusters estimated by the proposed gap statistic for each contamination strategy. The histograms clearly indicate that the gap statistic allows to correctly select the number of clusters, i.e. $k = 3$, even if data sets contain 40% outliers in total. Even if the highest contamination level is considered, in most cases the correct k is selected. It should be noted that such a high contamination, i.e. 50% outliers, is very extreme and unrealistic in practice.

Figure 11 summarizes the performance in terms of evaluation measures and demonstrates a great ability to discover the group structure independently of the number of outliers, reflected by low CER. The low CER can also be observed in case of the highest contamination. This might indicate that even if the gap statistic estimates a higher number of clusters than the true underlying number of groups (see Fig. 10), the detected clusters seem to be to some extent still homogeneous. The great performance of the gap statistic is additionally supported by similar values of $\bar{w}^{non0}$ and $\bar{w}^{inf}$, indicating highly efficient variable selection. Furthermore, zero values of $\bar{w}^{noise}$ imply that most noise variables are discarded for data clustering. Therefore, we can assume that the sparsity parameter s is appropriately estimated. Regarding the detection of outliers, the method can identify most outliers indicated by TPR around 1. However, TPR is slightly below 1 for the extremely contaminated data sets (i.e. 40/10). Such

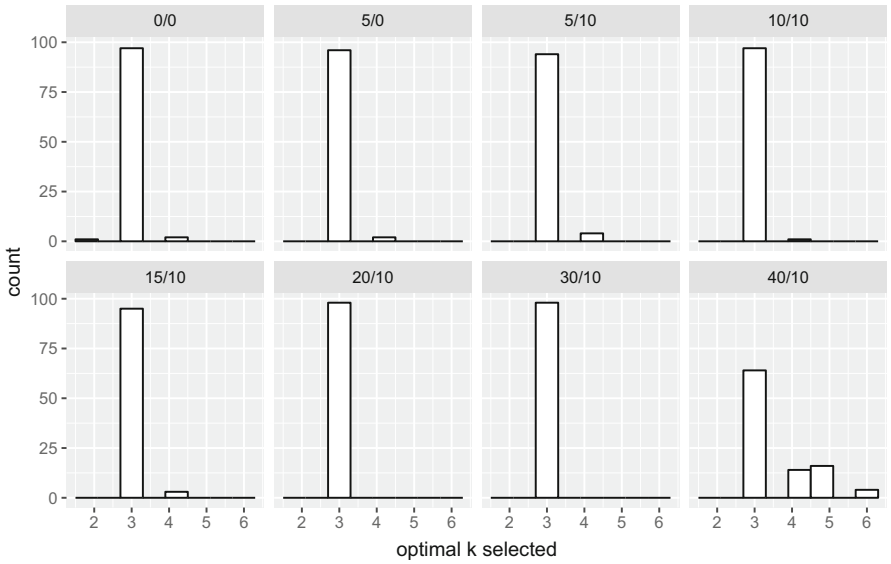

Fig. 10 Evaluation of the ability to correctly estimate k, considering different percentages of outliers in informative and in noise variables (x/x). The reported values of evaluation measures represent all 100 simulated datasets

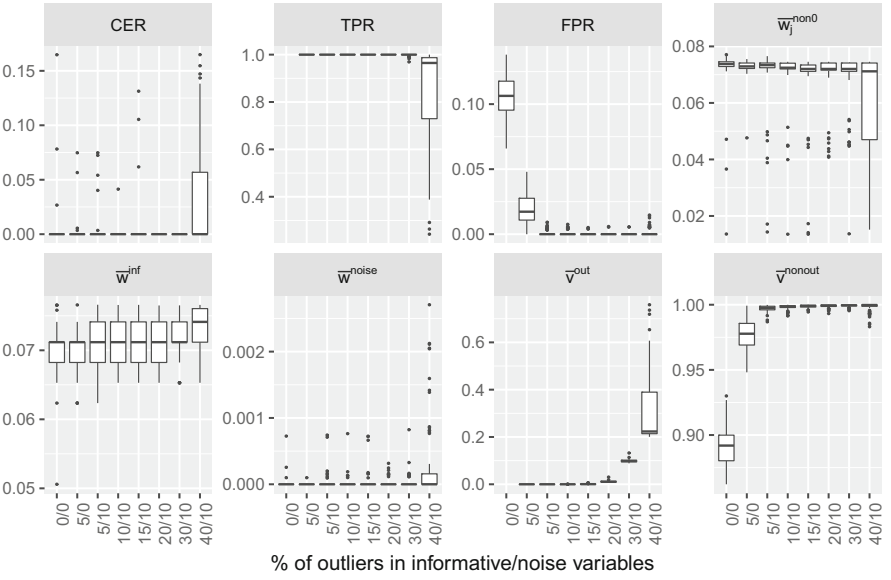

Fig. 11 Evaluation of the results considering different percentages of outliers in informative and in noise variables (x/x). The reported values of evaluation measures represent all 100 simulated datasets. Note that the scale of the y-axes is not fixed in the plots in order to better show the performance of the methods in various settings

low TPR can be a consequence of high observation weights for outliers, reflected by higher $\bar{v}^{out}$. This might indicate that the weights of some outliers are similar to the weights of non-outliers, or the cut-off value 0.5 needs to be increased in order to achieve perfect outlier detection for a large contamination level. Although the method misclassifies around 10% of normal observations in case of no contamination (0/0), it is able to correctly classify almost all non-outliers in contaminated datasets indicated by zero FPR. Based on the overall evaluation, the method demonstrates a great ability to identify a complex data structure in contaminated datasets.

6.3 Simulation 3: Comparison

In the last study, we compared the proposed weighted robust and sparse k-means (WRSK) algorithm with other k-means-based approaches, such as k-means (KC), trimmed k-means (TKC), sparse k-means (SKC) and its trimmed version (RSKC) on 30 simulated datasets. Each dataset is represented by 4 groups of various sizes ranging between 15 and 150. The generated observations are described by 4000 variables. Since the additional goal is to investigate the influence of different proportions of informative variables, three settings are considered, such as a percentage of 1%, 2%, and 5% of informative variables. Moreover, 20% of the observations are replaced by uniformly distributed outliers in the first 20% of the informative variables, and 10% of other observations are contaminated in 20% of randomly selected noise variables.

When applying the methods on the generated datasets, we assume prior knowledge of the number of clusters and optimize s in case of sparse-based algorithms. The trimming level for both TKC and RSKC corresponds to the total percentage of outliers, i.e. $\alpha = 0.30$. We evaluate the clustering solution by CER, and if appropriate, the performance regarding the outlier detection by TPR and FPR. Note that CER is again calculated with respect to the true group memberships before contamination. Since the outliers are placed only in the subset of informative variables, there is still some information about the group separation in non-contaminated variables.

Figure 12 summarizes the result based on 30 simulations. In general, in comparison to the remaining k-means-based methods, both the proposed method and RSKC seem to be resistant against the different percentages of informative variables. The clustering performance of KC, TKC, and SKC increases with an increasing proportion of informative variables, indicated by decreasing CER. In addition, CER shows that the proposed method outperforms the remaining methods in terms of identifying the underlying group structure reflected by the lowest CER for most simulated datasets. Although lower TPR demonstrate that our WRSK is not capable of identifying all outliers in comparison to the trimmed-based methods, the proposed method misclassifies fewer non-outliers indicated by the lowest FPR. Considering the performance, it seems that our method is able to sufficiently identify the group structure even if a large amount of noise variables is present in a data set.

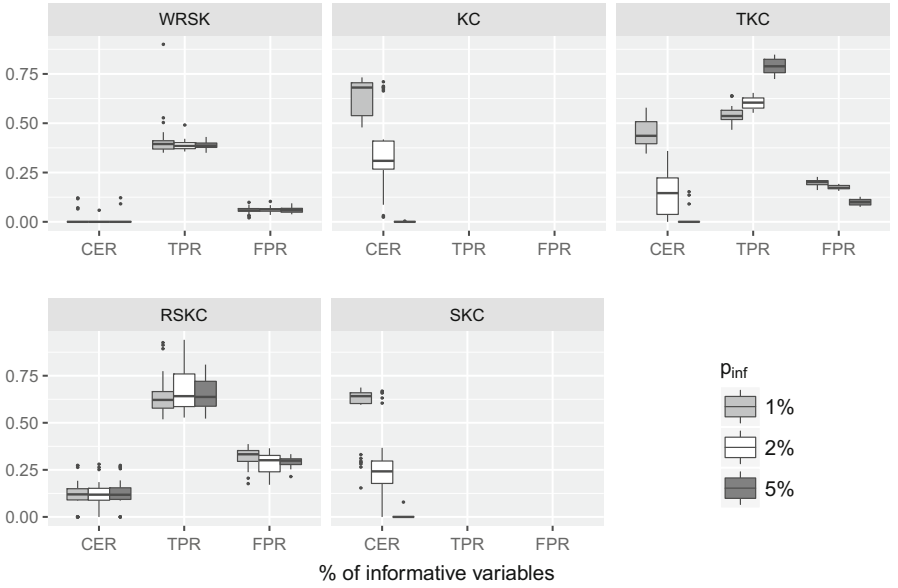

Fig. 12 Evaluation of various k-means based clustering methods, considering various proportions of informative variables (p_{inf}). The reported values of the evaluation measures correspond to the 30 simulated datasets

7 Analyzing the group structure of glass vessels

The proposed algorithm is particularly useful in the situation where a large number of variables is present as in the case of archaeological glass vessels from the 16th and 17th centuries, which were excavated in Antwerp being one of the most important historical centers of both glass manufacturing and trade. In 1997, chemical analysis was conducted in order to get better insight into the glass collection, including also the possible origin of the various glass samples. For this reason, the glass vessels were analyzed by an electron-probe X-ray micro-analysis (EXPMA) to measure spectra at different energy levels (Janssens et al. 1998). Consequently, traditional calibration methods were applied on spectra to extract major chemical elements resulting in the separation of four glass vessels groups, i.e. sodic, potasso-calcic, calcic, and potassic. The connection between element concentrations of glass vessels and their origin was discussed by Janssens et al. (1998). Lemberge et al. (2000) used their findings on an extended dataset consisting of 180 glass samples described by 1920 variables (different energy levels) in order to predict the same concentrations of the major elements as Janssens et al. (1998), using partial least squares. In this paper we employ the extended dataset consisting of 4 groups as well, as shown in Fig. 13. The plot additionally shows that the largest group (sodic) is split into two subgroups that are not clearly separated in the two-dimensional space of chemical concentrations. The two subgroups are caused by the installation of different detector efficiencies in the EXPMA. Detecting the subgroup of glass samples analyzed after the installation has been investigated e.g. by Serneels et al. (2005) and Filzmoser et al. (2008).

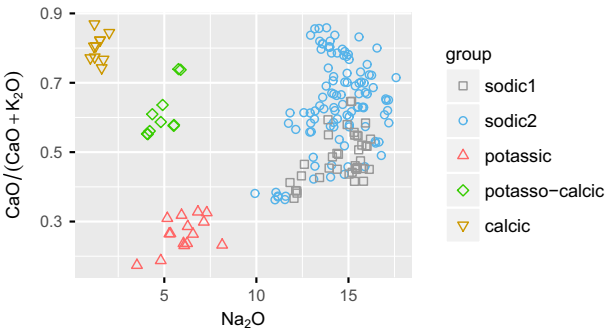

Fig. 13 Group membership of analyzed glass vessels, based on element concentrations (Lemberge et al. 2000)

Table 1 Evaluation of the clustering performance of k-means-based clustering methods

Method	WRSK	KC	TKC	RSKC	SKC
CER	0.039	0.183	0.166	0.191	0.167

Our focus is to detect an entire group structure, i.e. 5 groups, which might be hidden in the high-dimensional data space. Note that there is no pre-knowledge about the informative variables, neither of outliers in each group. In addition, the group membership based on the chemical concentrations does not necessarily have to reflect the group structure based on the origin of the glass samples. However, there exist some assumptions about the connection between the chemical elements - the glass manufacturing process - and the origin (Janssens et al. 1998). Therefore, we evaluate the performance in terms of CER with respect to the group membership shown in Fig. 13, and the cluster membership obtained by k-means-based algorithms with $k = 5$. Although it is not sure whether or not the dataset contains outliers, we set the trimming level to 0.10 for the trimming-based methods as suggested by Kondo et al. (2016). The optimal sparsity parameter for RSKC and WRSKC is selected from 1.5 to $\sqrt{p}$ in steps of 0.1 based on the gap statistic described in Sect. 4. The evaluation of the resulting clustering solution is presented in Table 1, which clearly shows that WRSK outperforms the remaining methods indicated by the lowest CER. Incorporating the trimming concept or sparsity seem to improve the performance of k-means (KC) as demonstrated by slightly larger CER for TKC or SKC. RSKC shows the worst performance. The reason might be that either important variables have been excluded, or that wrong observations have been trimmed, or a combination of both.

We also examine the final variable weights obtained by the sparse k-means-based algorithms. Figure 14 shows the final weights for each sparse method. The resulting values of the weights demonstrate that SKC completely fails in terms of achieving sparsity in the variable weight vector, as $w_j > 0$ for almost all variables. Nevertheless, there are several variables that receive a higher weight than in case of RSKC; see two peeks highlighted by dashed lines. This may indicate that there could be useful information about the group separation in the last energy levels of the measured spectra.

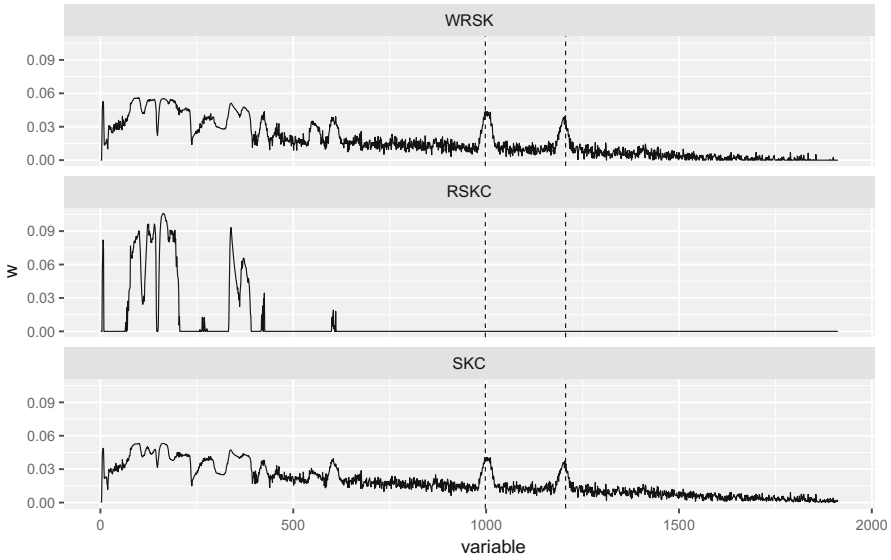

Fig. 14 The final variable weights obtained by sparse k-means-based clustering methods

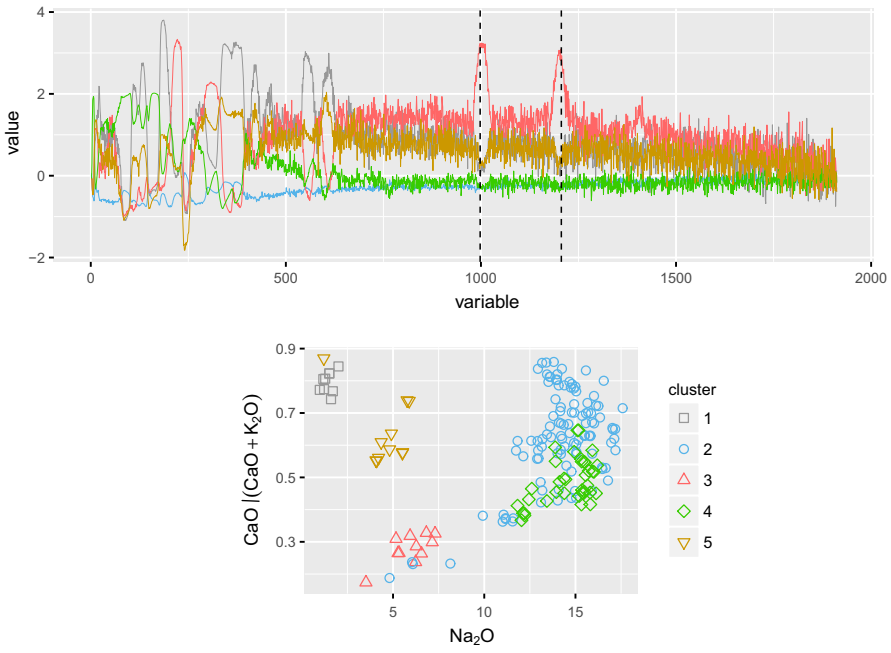

Fig. 15 Cluster centers calculated for each variable with respect to the final cluster membership obtained by WRSK (top) and the corresponding cluster membership (bottom)

A very similar conclusion can be made for the weights obtained by the proposed WRSK. In addition, WRSK results in a slightly sparse variable weight vector and at the same time can appropriately identify 5 groups as indicated by the lowest CER.

In order to investigate the final variable weights obtained by the proposed method in more detail, we examine how the centers of the detected clusters are distinguishable at each energy level of the spectra, i.e. for each variable. For this reason, we calculate the cluster centers as a weighted mean of the observations in each variable with the corresponding observation weights and the identified cluster membership. The resulting centers are displayed as spectra in Fig. 15 (top) and are distinguished by different colors based on the final cluster membership visualized in Fig. 15 (bottom). Figure 15 (top) particularly indicates that the centers appear to be well separated already at the low energy levels, i.e. in the first part of the variable vector. Furthermore, the center of cluster 3 appears to be well separated from other centers in the higher energy levels, highlighted by two dashed lines. In fact, the proposed WRSK is capable of identifying this informative part of the spectra; see Fig. 14. Although the proposed method does not lead to high sparsity in the variable weight vector, the final cluster membership visualized in Fig. 15 (bottom) indicates a great ability of WRSK to correctly identify informative variables since all 5 glass vessels groups are well recovered with only 5 misclassified observations. Whereas the misclassified calcic glass sample has an observation weight equal to 1, the remaining four misclassified potassic glass samples obtain weights considerably smaller than 1, i.e. 0.06, 0.00, 0.60, 0.76. This might indicate that although these observations are originally from the potassic group, their chemical structure seems to be different from the remaining observations of that group.

8 Conclusion

We propose a k-means-based clustering procedure that endeavors to simultaneously detect groups, outliers, and informative variables in high-dimensional data. The motivation behind our method is to improve the performance of the popular k-means method for real-world data that possibly contain both outliers and noise variables. Kondo et al. (2016) have addressed both issues in the robust (trimmed) and sparse k-means procedure, but our method goes even further. Firstly, our method aims to identify clusters, outliers, and noise variables at the same time. Secondly, the proposed procedure is designed in such a way that the required parameters are automatically estimated and, therefore, no pre-knowledge about the data is required. By incorporating the weighting function in k-means, each observation automatically receives a weight reflecting the degree of outlyingness based on which the outliers are identified. In order to correctly detect the informative variables, we employ a sparsity concept adjusted by observation weights. The proposed modified gap statistic is employed to optimize both the sparsity parameter and the number of clusters.

The introduced method together with the modified gap statistic has thoroughly been tested on a variety of simulated data sets as well as on a high-dimensional real data set. The conducted experiments indicated a great ability of the proposed procedure to discover the group structure. Two properties, the convergence and the stability of

the algorithm, have been investigated additionally. Although for reasons of space, the results are not presented here, our findings are briefly discussed. We observed that the algorithm, as applied in our simulation studies, typically convergences after around 6 iterations. The maximum number of iterations, considered as 15, was reached only occasionally. This can happen because the objective function is non-convex and rather difficult to optimize, but still the results as shown previously are convincing. Regarding the stability, the algorithm usually appeared to be insensitive to the order of observations or variables in terms of cluster membership as well as in terms of the identified outliers. The algorithm is implemented in the R Core Team (2016), freely available at https://github.com/brodsa/wrsk.

Future research includes extending the analysis of a data structure to identify the variables which are responsible for outliers. Such an idea is closely related to cell-wise outlier detection by Rousseeuw and Bossche (2018) for the situation of a single group data structure. A similar concept was introduced by Farcomeni (2014) in the context of clustering. The aim was to demonstrate that cell-wise contamination does not affect the introduced approach. However, the method has been tested in terms of clustering only, and no investigation has been conducted with respect to cell-wise outlier detection. Considering that outliers are commonly highly interesting observations due to their typically different content, it is even more important to find out which variables are behind this unusual behavior.

Acknowledgements Open access funding provided by TU Wien (TUW). This work has been partly funded by the Vienna Science and Technology Fund (WWTF) through Project ICT12-010 and by the K-project DEXHELPP through COMET—Competence Centers for Excellent Technologies, supported by BMVIT, BMWFW and the province Vienna. The COMET program is administrated by FFG and Österreichische Forschungsförderungsgesellschaft (Grand No. 843550).

References

Aggarwal CC (2016) Outlier analysis, 2nd edn. Springer, Berlin

Atkinson AC, Riani M, Cerioli A (2018) Cluster detection and clustering with random start forward searches. J Appl Stat 45(5):777–798

Breunig MM, Kriegel HP, Ng RT, Sander J (2000) LOF: identifying density-based local outliers. ACM Sigmod Rec 29:93–104

Campello RJ, Moulavi D, Zimek A, Sander J (2015) Hierarchical density estimates for data clustering, visualization, and outlier detection. ACM Trans Knowl Discov Data 10(1):5:1–5:51

Celebi ME, Kingravi HA, Vela PA (2013) A comparative study of efficient initialization methods for the k-means clustering algorithm. Expert Syst Appl 40(1):200–210

Cerioli A, Riani M, Atkinson AC, Corbellini A (2018) The power of monitoring: how to make the most of a contaminated multivariate sample. Stat Methods Appl 27(4):559–587

Coretto P, Hennig C (2016) Robust improper maximum likelihood: tuning, computation, and a comparison with other methods for robust gaussian clustering. J Am Stat Assoc 111(516):1648–1659

Cuesta-Albertos J, Gordaliza A, Matrán C (1997) Trimmed *k*-means: an attempt to robustify quantizers. Ann Stat 25(2):553–576

Dotto F, Farcomeni A, García-Escudero LA, Mayo-Iscar A (2018) A reweighting approach to robust clustering. Stat Comput 28(2):477–493
Farcomeni A (2014) Snipping for robust k-means clustering under component-wise contamination. Stat Comput 24(6):907–919
Filzmoser P, Maronna R, Werner M (2008) Outlier identification in high dimensions. Comput Stat Data Anal 52:1694–1711
Galimberti G, Manisi A, Soffritti G (2018) Modelling the role of variables in model-based cluster analysis. Stat Comput 18(1):145–169
Gallegos MT, Ritter G (2009) Trimming algorithms for clustering contaminated grouped data and their robustness. Adv Data Anal Classif 3(2):135–167
Garcia-Escudero LA, Gordaliza A (1999) Robustness properties of k-means and trimmed k-means. J Am Stat Assoc 94(447):956–969
García-Escudero LA, Gordaliza A, Matrán C, Mayo-Iscar A (2008) A general trimming approach to robust cluster analysis. Ann Stat 36(3):1324–1345
García-Escudero LA, Gordaliza A, Matrán C, Mayo-Iscar A (2010) A review of robust clustering methods. Adv Data Anal Classif 4(2–3):89–109
García-Escudero LA, Gordaliza A, Matrán C, Mayo-Iscar A (2011) Exploring the number of groups in robust model-based clustering. Stat Comput 21(4):585–599
Gordon AD (1999) Classification, 2nd edn. Chapman and Hall, London
Jain AK (2010) Data clustering: 50 years beyond k-means. Pattern Recognit Lett 31(8):651–666
Janssens KH, Deraedt I, Schalm O, Veeckman J (1998) Composition of 15–17th century archaeological glass vessels excavated in Antwerp, Belgium. Springer, Vienna, pp 253–267
Kondo Y, Salibian-Barrera M, Zamar R (2016) RSKC: an R package for a robust and sparse k-means clustering algorithm. J Stat Softw 72:1–26
Lemberge P, De Raedt I, Janssens KH, Wei F, Van Espen PJ (2000) Quantitative analysis of 16–17th century archaeological glass vessels using PLS regression of EPXMA and μ-XRF data. J Chemom. 14(5–6):751–763
Mohammad AH, Vineet C, Saeed S, Mohammed JZ (2009) Robust partitional clustering by outlier and density insensitive seeding. Pattern Recognit. Lett. 30(11):994–1002
Neykov N, Filzmoser P, Dimova R, Neytchev P (2007) Robust fitting of mixtures using the trimmed likelihood estimator. Comput. Stat. Data Anal. 52(1):299–308
R Core Team (2016) R: a language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/
Raftery AE, Dean N (2006) Variable selection for model-based clustering. J Am Stat Assoc 101(473):168–178
Rocke DM (1996) Robustness properties of S-estimators of multivariate location and shape in high dimension. Ann Stat 24(3):1327–1345
Rousseeuw PJ, Bossche WVd (2018) Detecting deviating data cells. Technometrics 60(2):135–145
Serneels S, Croux C, Filzmoser P, Van Espen PJ (2005) Partial robust M-regression. Chemom Intell Lab Syst 79(1):55–64
Sugar CA, James GM (2003) Finding the number of clusters in a dataset: an information-theoretic approach. J Am Stat Assoc 98(463):750–763
Tibshirani R, Walther G, Hastie T (2001) Estimating the number of clusters in a data set via the gap statistic. J R Stat Soc Ser B (Stat Methodol) 63(2):411–423
Witten DM, Tibshirani R (2010) A framework for feature selection in clustering. J Am Stat Assoc 105(490):713–726
Witten DM, Tibshirani R (2013) sparcl: Perform sparse hierarchical clustering and sparse k-means clustering. R package version 1.0.3
Xu R, Wunsch D (2005) Survey of clustering algorithms. Trans Neural Netw 16(3):645–678