OXFORD

Data and text mining
Pooled variable scaling for cluster analysis

Jakob Raymaekers[1] and Ruben H. Zamar[2,*]

[1]Department of Mathematics, KU Leuven, Leuven 3001, Belgium and [2]Department of Statistics, University of British Columbia, Vancouver, BC V6T 1Z4, Canada

*To whom correspondence should be addressed.

Abstract

Motivation: Many popular clustering methods are not scale-invariant because they are based on Euclidean distances. Even methods using scale-invariant distances, such as the Mahalanobis distance, lose their scale invariance when combined with regularization and/or variable selection. Therefore, the results from these methods are very sensitive to the measurement units of the clustering variables. A simple way to achieve scale invariance is to scale the variables before clustering. However, scaling variables is a very delicate issue in cluster analysis: A bad choice of scaling can adversely affect the clustering results. On the other hand, reporting clustering results that depend on measurement units is not satisfactory. Hence, a safe and efficient scaling procedure is needed for applications in bioinformatics and medical sciences research.

Results: We propose a new approach for scaling prior to cluster analysis based on the concept of pooled variance. Unlike available scaling procedures, such as the SD and the range, our proposed scale avoids dampening the beneficial effect of informative clustering variables. We confirm through an extensive simulation study and applications to well-known real-data examples that the proposed scaling method is safe and generally useful. Finally, we use our approach to cluster a high-dimensional genomic dataset consisting of gene expression data for several specimens of breast cancer cells tissue obtained from human patients.

Availability and implementation: An R-implementation of the algorithms presented is available at https://wis.kuleuven.be/statdatascience/robust/software.

Contact: ruben@stat.ubc.ca

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Every time cluster analysis is used to find homogeneous groups in the data, we face the issue of how to scale the variables. The choice of scaling (including the option of not scaling at all) has practical implications because, in general, clustering methods are not scale-invariant. Even those clustering methods that are scale-invariant, become scale-dependent when they are combined with variable selection or regularization. As a consequence, we may get a different data partition if the variables in the data are rescaled. Unfortunately, there is not a generally accepted way to scale the variables for clustering (Jain, 2010; Jain and Dubes, 1988).

There are conflicting recommendations regarding the scaling of the data in the clustering literature. For example, Milligan and Cooper (1988)—often cited as the main benchmark study of this topic in the context of hierarchical clustering—recommend scaling the variables by their range. Steinley (2004) came to the same conclusion using *k*-means. On the other hand, Vesanto (2001) recommends the use of the SD for *k*-means clustering. Schaffer and Green (1996) studies clustering on real-data examples and argue against

scaling with the range or the SD. Stoddard (1979) also argued against the use of the SD in the analysis of laboratory procedures.

While scaling may not always be necessary, it seems that in general the most reliable approach is to use some sort of scaling for two main reasons. The first reason is that scaling gets rid of the measurement scales of the variables. These measurement scales may have a strong influence on the clustering results, to the extent that a single very large variable can solely determine the whole clustering outcome. Furthermore, it can be of practical importance to get rid of measurement scales, e.g. a variable measuring 'height' should have the same effect on the clustering procedure when measured in centimeters or in meters. The second reason is that scaling becomes practically mandatory in the context of high-dimensional clustering with variable selection. Variable selection is usually achieved through the addition of a penalization term to the objective function. This penalization is typically not scale-invariant and thus yields different outcomes depending on the variables' relative sizes. Therefore, without scaling, the penalty acts differently on each variable, which will often lead to ineffective variable selection. For these reasons, we consider feature scaling as a necessary pre-processing step in cluster analysis.

Though the arguments in favor of scaling before cluster analysis are clear, the issue of how to scale is delicate. The reason is that there are two types of variables: informative variables and noise variables. Informative variables help to separate clusters in the data whereas noise variables do not. If an informative variable is scaled with a very large scale, it will be compressed to a small size and thus there is a risk that much of its clustering power will disappear. In contrast, if a noise variable is scaled with a very small scale, it can potentially be blown up to where it solely determines the clustering outcome. Note that, this issue is even more pronounced in high-dimensional datasets as they typically contain many noise variables.

Interestingly, most of the commonly used scale estimators, such as the SD and the range have a tendency to yield large-scale estimates on potentially very informative variables. More precisely, these estimators may produce unduly large scales for a variable with a multimodal distribution because of the dispersion between the groups. Hence, a drawback of feature scaling by these estimates is the possible undesirable reduction of the relative clustering importance of features that clearly separate different groups. While these features are not guaranteed to reveal the whole clustering structure of the data, they show good potential and should be treated carefully.

The discussion so far suggests that there is a need for further study of both the effect of scaling and the best choice of scale estimator in cluster analysis. Despite the considerable potential impact of scaling on cluster outcomes, papers on this topic are scant, far apart and lack consensus.

In Section 2, we introduce two new scale estimators specially designed to scale variables before clustering: the *pooled standard deviation* (pSD) and the *pooled absolute deviation*. By using pooled scale estimators, we aim to scale variables without destroying their clustering power if they have any. If the marginal distribution of a feature shows evidence of clustering, our proposed scaling will enhance its clustering importance. If the marginal distribution of a feature X does not show evidence of clustering, but the joint distribution of some subset of features including X carries important clustering information, our proposed scaling approach will have a neutral effect on this variable. The proposed scaling approach can be used as pre-processing before applying any clustering method, including those with variable selection and subspace clustering, as illustrated with several examples. To calculate our proposed scale, we first run k-means clustering on each of the variables. In order to choose the number of clusters used to estimate the pooled scales, we use the Gap statistic (Tibshirani *et al.*, 2001) with a sped-up boot-strapping procedure that bypasses the otherwise unaffordable computational burden of this approach.

The rest of the article is organized as follows. In Section 3, we compare the new scale estimators with existing scaling procedures in an extensive simulation study. Moreover, we show an application of the pooled scale estimators prior to the hierarchical clustering of gene expressions of breast cancer sample tissues (additional real-data examples are presented in the Supplementary Material). Finally, Section 4 concludes.

2 Materials and methods

2.1 Pooled scale estimators
Our starting point is the univariate k-means clustering. Assume we have a univariate dataset $x_1, \ldots, x_n$. The well-known k-means clustering looks for the k cluster centers, which minimize the squared deviations for every point in the dataset to the closest cluster center. More precisely, the vector of cluster centers is defined as:

$$\boldsymbol{\mu} = (\mu_1, \ldots, \mu_k) = \underset{t=(t_1,\ldots,t_k)}{\arg\min} S_k(\boldsymbol{t}),$$

where

$$S_k(\boldsymbol{t}) = \sqrt{\frac{1}{n}\sum_{i=1}^{n} d_{2,i}(\boldsymbol{t})} \tag{1}$$

and $d_{2,i}(\boldsymbol{t}) = \min_{1 \leq j \leq k} ||x_i - t_j||_2^2$. Note that, $S_k(\boldsymbol{\mu})$ can be interpreted as an estimator of scale. In particular, if $k = 1$, then $\boldsymbol{\mu}$ is the classical sample mean and $S_1(\boldsymbol{\mu})$ reduces to the classical SD. If $k > 1$, the squared scale can be interpreted as a pooled variance of the points around their cluster centers. As an example, suppose that $k = 2$ and that the sets C_1 and C_2 contain the indices of the points in the two clusters. Moreover, let $|A|$ represents the number of elements in the set A, so that $|C_1| + |C_2| = n$. We then have $S_2^2(\boldsymbol{\mu}) = \frac{1}{n}\sum_{i=1}^{n} d_{2,i}(\boldsymbol{\mu}) = \frac{1}{n}\sum_{i \in C_1} ||x_i - \mu_1||_2^2 + \frac{1}{n}\sum_{i \in C_2} ||x_i - \mu_2||_2^2 = \frac{|C_1|}{n} \text{Var}(C_1) + \frac{|C_2|}{n} \text{Var}(C_2)$, where $\text{Var}(C_j)$ denotes the sample variance of the elements belonging to cluster j. We thus obtain a weighted mean of the within-cluster variances, with weights proportional to the number of observations in each cluster.

Our idea is to use S_k, with an appropriate (variable depending) value of k, for scaling the variables prior to the application of a clustering procedure. We will refer to this scale estimator as the *pSD*. The intuition for this scale estimator is the following. If a certain variable appears to not separate any clusters in the data, we consider the variability of this variable to be uninformative and we use the largest scale, S_1 (i.e. the classical SD), to scale it before clustering. However, when a variable does seem to separate, e.g. k clusters, then S_k will tend to be relatively small compared with S_1, and using S_k in that case will avoid dampening the variability (i.e. information) in this variable. This way, we hope to preserve as much of the clustering information in each variable as possible, while still making the variables unitless.

As an alternative to k-means, k-medians clustering can also be used for the scaling of the variables. Using the notation introduced above, the vector of cluster centers in k-medians is defined as:

$$\boldsymbol{\mu} = (\mu_1, \ldots, \mu_k) = \underset{t=(t_1,\ldots,t_k)}{\arg\min} M_k(\boldsymbol{t}),$$

where

$$M_k(\boldsymbol{t}) = \frac{1}{n}\sum_{i=1}^{n} d_{1,i}(\boldsymbol{t})$$

and $d_{1,i}(\boldsymbol{t}) = \min_{1 \leq j \leq k} |x_i - t_j|$. The cluster centers now correspond to the median of the observations in each cluster. Note that once again, M_k can be interpreted as an estimator of scale. If $k = 1$, then $\boldsymbol{\mu}$ is the classical sample median and $M_1(\boldsymbol{\mu})$ is the mean absolute deviation (mad) (from the median). If $k > 1$, M_k can be interpreted as the pooled mean absolute deviation (pmad) of every point around its cluster center, where the pooling is done by a weighted average with weights determined by the number of observations in each cluster. We will call M_k the *pmad*. The pmad is expected to be more robust against outliers compared with the pSD. This is consistent with the results of the simulation study of Section 4.

Remark 1. Since we are using univariate k-means and k-medians to obtain the pSD and pmad, it is worth noting that there is an algorithm, which guarantees the convergence to a global optimum instead of a local one. This algorithm uses dynamic programing, runs in $\mathcal{O}(kn^2)$ time and is implemented in the R-package Ckmeans.1d.dp, see Wang and Song (2011).

2.2 Determining k
An important question regarding our proposed pooled scale estimators is how to choose the appropriate number of clusters k^* for each variable in the dataset. There is a vast literature on how to choose the appropriate number of clusters in cluster analysis. Relevant comparative studies giving a good overview of the most common techniques are found in Milligan and Cooper (1985), Halkidi *et al.* (2001), Maulik and Bandyopadhyay (2002), Brun *et al.* (2007) and Arbelaitz *et al.* (2013). In the setting of pooled scale estimators, we need a criterion that is relatively fast to compute and not too

sensitive to spurious clusters. Most importantly, we need a criterion that can distinguish between the 'null-case' of one-cluster versus the alternative case of two or more clusters. This is an important point, since especially in high-dimensional datasets, we expect that many variables may not have interesting information for clustering the data and thus should be scaled by the largest scale, i.e. the SD or mad. Note that, this requirement makes many popular cluster validation indices unsuitable for us, since many of them do not yield a reasonable comparative assessment of the one-cluster-case.

A well-known criterion that satisfies our needs is the Gap statistic (Hastie *et al.*, 2009; Tibshirani *et al.*, 2001). Given a certain clustering algorithm and the resulting partition of the data, the gap statistic works as follows. For each value of the number of clusters under consideration, the 'tightness' of the clusters in the found partition is compared with the tightness of clusters obtained by clustering random datasets using the same algorithm. If this difference is large for a given number of clusters, it indicates 'stronger than random' clustering structure and the gap statistic will pick that number of clusters as the true number. A mathematical description of the gap statistic is as follows. Suppose, we have clustered the data into k clusters, $C_1, \ldots, C_k$, where C_j denotes the indices of the observations in cluster j. Let W_k be the sum of the within-cluster sums of squares around their corresponding cluster means, i.e. $W_k = \sum_{j=1}^{k} \sum_{i \in C_j} (x_i - \bar{x}_j)^2$ where $\bar{x}_j = \frac{1}{|C_j|} \sum_{i \in C_j} x_i$ is the mean of the observations in cluster j. In order to identify the number of clusters, the value of $\log(W_k)$ is compared to its expected value $E_n^*[\log(W_k)]$ under a uniform reference distribution on the range of the dataset. If this value deviates too much from its expected value under a uniform distribution, it indicates the existence of clusters in the data. The intuition for the comparison with the uniform distribution is that it is the distribution, which is most likely to generate spurious clusters (within the family of log-concave densities) and will thus on average provide the strongest evidence against the alternative hypothesis.

In principle, the reference distribution of $\log(W_k)$ is determined by generating bootstrap samples from the uniform distribution. As we would like scale every variable in the dataset, this appears to lead to a prohibitive computational cost, which would scale poorly with the number of variables. Fortunately, there is an efficient way to bypass this hindrance based on the results of Proposition 1 below.

In our case, the clusters are coming from the k-means (or k-medians) clustering algorithm. This means that for a given clustering $C_1, \ldots, C_k$, we have

$$W_k = \sum_{r=1}^{k} d_{2,i}(\mu_1, \ldots, \mu_k) = n S_k^2$$

in the computation of the pSD. For the pmad, we redefine W_k such that it corresponds to the pooled within-cluster sum of absolute deviations around the cluster medians:

$$W_k = \sum_{r=1}^{k} d_{1,i}(\mu_1, \ldots, \mu_k) = n M_k.$$

Therefore, in our setting, the gap statistic will be large when there is a 'significant difference' in the estimated pooled scale compared with the pooled-scaled estimate on data coming from a uniform distribution.

In order to estimate $E_n^*[\log(W_k)]$ and $\text{Var}_n^*[\log(W_k)]$, we apply k-means to B bootstrap samples of size n. The mean of these samples serves as the estimate for $E_n^*[\log(W_k)]$ and the appropriate scale, which accounts for the simulation error in $E_n^*[\log(W_k)]$, is then the SD of the bootstrap samples multiplied by $\sqrt{1 + 1/B}$.

We now turn to speeding up the bootstrapping procedure for scaling all the variables in a dataset. The speed-up is achieved by exploiting the fact that we are in the univariate setting and by using the properties of the proposed scaling methods established in the following proposition.

Proposition 1. *Let $x = x_1, \ldots, x_n$ be a sample of univariate observations and let $C_1, \ldots, C_k$ be a partition of x resulting from solving the k-means clustering problem. Denote the value of the objective function with S_k as in Equation (1). Let $s > 0$ and $t \in \mathbb{R}$ consider $z = z_1, \ldots, z_n$ where $z_i = (x_i - t)/s$ for $i = 1, \ldots, n$. We then have:*

1. *Shift and Scale invariance of k-means clusters:*
   $C_1, \ldots, C_k$ is a solution to the k-means clustering problem on z.

2. *Shift invariance and Scale equivariance of k-means objective function:*
   The value of the objective function of this clustering is S_k/s.

3. *Shift and Scale invariance of gap statistic:*
   The number of clusters selected by the gap statistic is the same for x and z.

The exact same result holds for clustering with k-medians. These results are intuitive and follow from the fact that Manhattan and Euclidean distances are scale equivariant. We give the proof in the Supplementary Material. Using this proposition, it becomes clear that we need to bootstrap the reference distribution of $\log(W_k)$ only once, instead of for every variable separately. The reason is that we can first rescale each variable in the dataset by the range, so that all of them have the same range of length one. Now the reference distribution of W_k is the same for each of these variables, allowing us to bootstrap once from the uniform distribution on $[0, 1]$ to obtain the reference distribution for all variables. Note that, it does not matter on which interval of length 1 the variable takes its values, since the whole procedure is shift invariant as well.

Remark 2. The uniform distribution is not the only possible reference distribution one can use. Another option briefly suggest by Tibshirani *et al.* (2001) is to use log-concave density estimation, which is possible for univariate distributions. This would yield a fit of a log-concave density to every variable in the data from which bootstrap samples can then be drawn. While this takes into account the individual distributions of every variable, the drawback is that in this case we have to take the slow approach of generating separate reference distributions for every variable.

Remark 3. Instead of the gap statistic, the jump statistic (Sugar and James, 2003) may be used as an alternative. In the univariate setting, the jump statistic considers the 'distortions' $\hat{d}_k = S_k^2$ for several numbers of clusters k. They then define the jumps as $J_k = \hat{d}_k^{-1/2} - \hat{d}_{k-1}^{-1/2}$, where $\hat{d}_0 \equiv 0$. Finally, the number of clusters is estimated by taking $K^* = \arg\max_k J_k$. One advantage of the jump statistic over the gap statistic is that it faster to compute, since we do not need to bootstrap a reference distribution. In our simulations however, the gap statistic performed better.

2.3 Algorithm

We are now ready to describe the procedure; we propose for scaling a dataset prior to clustering. For a p-variate dataset $X_1, \ldots, X_p$ with n observations per variable, we apply the following steps to scale all the variables:

1. Generate B bootstrap samples of size n from the uniform distribution on $[0, 1]$ and cluster each of them using k-means. Retain all the values $W_{k,b}^*$ for $k = 1, \ldots, k_{\max}$ and $b = 1, \ldots, B$. Denote with $m_k = (1/B) \sum_b \log(W_{k,b}^*)$, $\text{sd}_k = [(1/B) \sum_b \left(\log(W_{k,b}^*) - m_k\right)^2]^{1/2}$, the estimates for the location and scale of $\log(W_k)^*$, respectively. Finally, put $s_k = \sqrt{1 + 1/B} \, \text{sd}_k$.

2. For all variables X_j, $j = 1, \ldots, p$, do:
   a. Rescale the variable with its range to obtain $Z_j = X_j/r_j$, where $r_j = \text{range}(X_j)$.
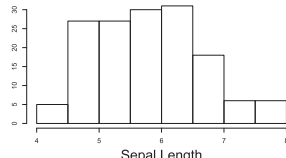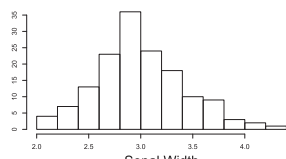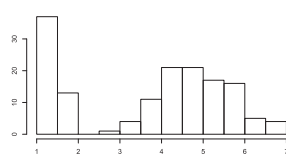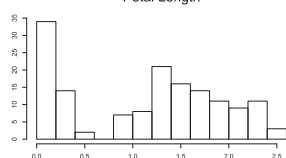
b. Cluster Z_j using k-means for $k = 1, \ldots, k_{max}$ and retain the values $W_{k,j} = n\, S_{k,j}^2$.

c. Calculate the values of the gap statistic: $\mathrm{Gap}_j(k) = m_k - \log(W_{k,j})$.

d. Choose the number of clusters k^*, by setting $k^* = $ smallest k such that $\mathrm{Gap}_j(k) \geq \mathrm{Gap}_j(k+1) - c\, s_{k+1}$.

e. Rescale the value of the objective function of the appropriate k, $r_j\, S_{k^*,j}$, and use this pSD to scale X_j.

The constant c in the above procedure controls the rejection of the null model. As c goes up, it becomes less likely to reject the null model of zero clusters. A default value is $c = 1$, which works well according to Breiman *et al.* (1984) and Tibshirani *et al.* (2001). For the number of bootstrap samples, a default of $B = 1000$ yields almost no variance in the resulting scale estimates in our experience. Replacing k-means with k-medians yields the scaling procedure with M_k, the pmad.

Example 1: Fisher's Iris data

As an illustrative example, we consider Fisher's well-known Iris dataset (Fisher, 1936), collected by Anderson (1935). This dataset contains 50 samples from each of 3 types of iris: Iris setosa, versicolor and virginica. Each flower is described by four variables, which describe the dimensions of its sepal and petal. Table 1 illustrates the effect of scaling with the proposed pSD on this data and compares with the SD and the range. For the first two variables, the pSD is equal to the SD, because these variables do not seem to clearly distinguish any groups in the data. However, variables 3 and 4 seem to both distinguish two clear groups, resulting in significantly lower pSDs. The last row of the table reports the adjusted rand index (ARI) (Hubert and Arabie, 1985) with respect to the true classification when performing k-means with $k = 3$ on the dataset after scaling. The ARI takes on values between -1 and 1, where an ARI

of 1 indicates perfect agreement between partitions and the lower the ARI, the higher the disagreement between the partitions. It is clear that scaling with the pSD gives better results than scaling with the SD or the range, since neither of these take into account the individual separative power of the variables. As a reference, we mention that k-means clustering without scaling gives an ARI of 0.73.

3 Results

3.1 Simulation study

The most well-known comparative study on the scaling of variables in clustering is arguably the one by Milligan and Cooper (1988), building on Milligan (1985). Recently, Qiu and Joe (2006) used the design of this simulation study as a basis for a new algorithm to generate clusters with a specified degree of separation. The R-package clusterGeneration (Qiu and Joe., 2015) contains an implementation of their algorithm and it will be the basis of our simulation study.

We compare the following types of scaling in our simulation study:

1. No scaling
2. The SD: $SD = \frac{1}{n-1}\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}$
3. The range: $\mathrm{range} = x_{(n)} - x_{(1)}$
4. The mad: $\mathrm{mad} = \frac{1}{n-1}\sum_{i=1}^{n}|x_i - \mathrm{median}_i x_i|$
5. The pSD: $\mathrm{psd} = S_{k^*}$
6. The pmad: $\mathrm{pmad} = M_{k^*}$

In order to get a complete picture of the different scaling methods, we perform an extensive simulation study. Each generated dataset has an even number of clean variables, which we vary between 2 and 10. The clusters are generated from the multivariate standard normal distribution. We consider equally sized clusters of size 100 each. The degree of separation between the clusters is either separated (0.21) or well-separated (0.34), see Qiu and Joe (2006). We then add a percentage of noise variables to the dataset varied between 0% and 2000% of the number of clean variables. The noise variables can either be multivariate standard normal or uniform over the range of the clean variables. The uniform-noise variables are generated by adding a small Gaussian perturbation to an equally spaced grid over the range of the signal variables to ensure that they do not have any separative power. We did the simulation both on clean data as well as contaminated data. For the contaminated data, 5% of the observations of each of the signal variables are replaced with points sampled randomly from the uniform distribution on $[\bar{X}_j - 4\, s(X_j), \bar{X}_j + 4\, s(X_j)]$, where $\bar{X}_j$ and $s(X_j)$ denote the mean and SD of the signal variable. Table 2 summarizes the factors of our simulation study and their levels.

All together this gives 1280 different settings and for each of these, we generate 100 datasets. Each generated dataset is scaled using the six scale estimators described above. Afterwards we perform the most popular methods of connectivity-based clustering and centroid-based clustering. More specifically, we use hierarchical clustering (Anderberg, 2014; Hartigan, 1975) on the Euclidean distances with single, average, complete and Ward's linkage functions (Ward, 1963) as well as k-means (Lloyd, 1982; MacQueen *et al.*, 1967) and partitioning around medoids using the Manhattan

Table 1. The effect of variable scaling on Fisher's Iris data

Variables		SD	Range	pSD
Sepal Length		0.83	3.6	0.83
Sepal Width		0.44	2.4	0.44
Petal Length		1.77	5.9	0.40
Petal Width		0.76	2.4	0.18
k-means ARI		0.62	0.72	0.89

Note: The pSD produces smaller scales for variables 3 and 4 than the classical SD and the range. This results in a higher ARI when clustering the data with the k-means algorithm after scaling.

Table 2. Design factors of the simulation study

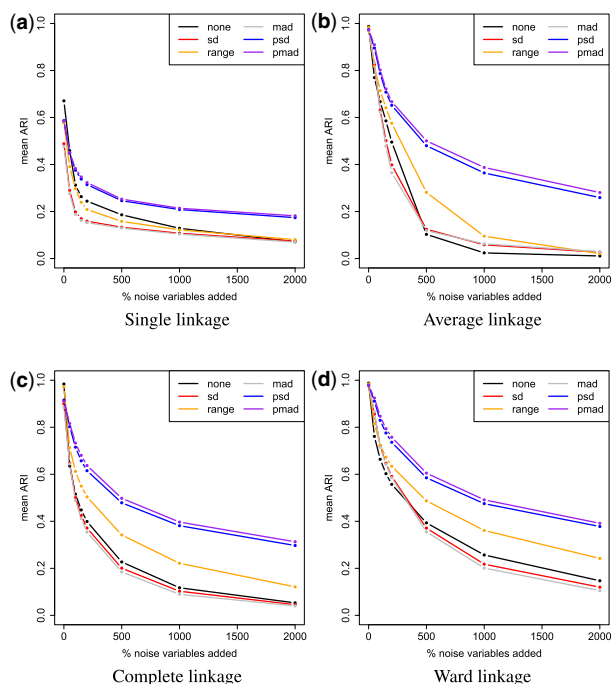| Factor	Levels	#
Number of clean variables	2, 4, 6, 8, 10	5
Number of clusters	2, 3, 4, 5	4
Degree of separation	separated, well-separated	2
% noise variables added	0, 50, 100, 150, 200, 500, 1000, 2000	8
Type of noise	Gaussian, uniform	2
% outliers	0, 5	2

Fig. 1. Simulation results for hierarchical clustering with single (**a**), average (**b**), complete (**c**) AQ10and Ward (**d**) linkage functions on outlier-free data. The pooled scale estimators are the least sensitive to an increasing number of noise variables
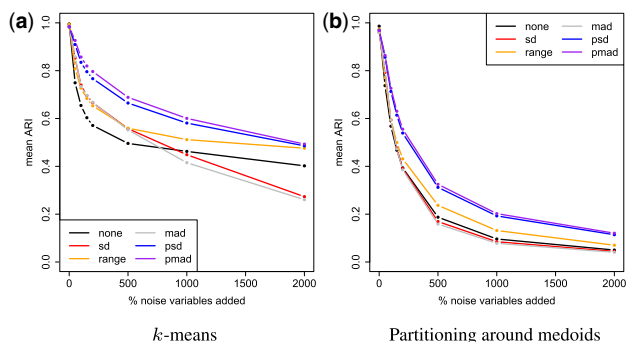

Fig. 2. Simulation results for *k*-means (**a**) and partitioning around medoids (**b**) on data without outliers. The pooled scale estimators are more resistant to the addition of noise variables to the data

distance (Kaufman and Rousseeuw, 2009). For *k*-means, the algorithm of Hartigan and Wong (1979) is used with 100 random starts and 100 maximum iterations for each starting value.

We compare the results using the ARI (Hubert and Arabie, 1985), which lies between −1 and 1 where 1 indicates a perfect clustering. We cluster each dataset for a variety of target clusters $k = 1, \ldots, 3 \times T$, where T denotes the true number of clusters. For *k*-means, this value is a direct input, whereas for hierarchical clustering, we cut the dendrogram at these various levels of k. We then pick the optimal value of the ARI over these different clusterings. The reason for this procedure is that we want to evaluate the effect of scaling on clustering without any distortion from the question of how to choose the optimal number of clusters. Furthermore, particularly in the case of hierarchical clustering, the clusterings resulting from a higher number of partitions are often more reflective of the real underlying structure than cutting the dendrogram at the true number of clusters, since single outlying observations can distort the dendrogram significantly.

Figure 1 shows the big picture of the simulation results for hierarchical clustering on data without outliers. For each type of

linkage, the graphs show the average ARI over all different settings with the increasing number of noise variables on the *x*-axis. Several interesting observations can be made from these plots. It is clear that as the number of noise variables increases, the clustering gets more difficult resulting in generally lower ARI values. However, scaling with the SD or the mad is clearly more sensitive to noise variables than the other methods. Scaling with the range does fairly well, which is in line with the findings of Milligan and Cooper (1988). The pooled scale estimators outperform all the other methods, especially when the number of noise variables is large. The difference between the pSD and the pmad seems very small and not significant in these plots. Finally, not scaling appears to perform similarly to scaling with either SD or mad. This is due to the particular simulation setup and one must always take into account that the performance of not scaling the variables can be completely destroyed by taking one noise variable, which has a variance which is much larger than the variance of the signal variables.

Figure 2 shows the results for *k*-means clustering and partitioning around medoids. The conclusions are largely the same as for the case of hierarchical clustering. When scaling with the SD or mad, the true clustering structure seems more difficult to retrieve. The range preserves more of the cluster structure and performs better than the SD and mad, which is in line with Steinley (2004). However, the pooled scale estimators again outperform the competitors.

The simulation results presented above only give a rough overview of the performance of the methods and do not show the performance in the presence of outliers. The most interesting insight from a more detailed analysis of the results is that the performance of the methods is highly dependent on the type of noise variables, which are added to the signal variables. Scaling with the range works well when the noise variables are more Gaussian but fails when the noise is more uniform. This can be explained by the fact that uniform noise variables have a large variance given their range. As a result, their impact on the clustering is large when scaling with the range. Scaling with the SD and mad works much better when the noise variables are more uniform than the case of Gaussian noise. This in turn can be explained by the fact that the uniform noise variables have a high variance for their range compared with Gaussian noise variables. Scaling them by their variance pushes the uniform noise more toward the center, which limits their influence.

The effect of outliers is shown in Supplementary Figures S1 and S2. These results expose the sensitivity of the range to the presence of outliers. Other than that, the relative performance of the different scaling methods is roughly the same as in the case of outlier-free data. An interesting note is that the pmad is clearly more robust to outliers than the pSD, which resembles the robustness of the mad versus that of the SD in classical scale estimation.

3.2 Gene expression example

In a seminal paper, Perou *et al.* (2000) analyzed gene expression patterns of 65 surgical specimens of human breast tumors. The data are publicly available at https://www.omicsdi.org/dataset. They identified 496 intrinsic genes, which had significantly larger variation between different tumors compared with the variation between paired samples from the same tumor. Using hierarchical clustering with Eisen linkage (Eisen *et al.*, 1998), they clustered the tumors into four different types: basal-like, Erb-B2+, normal-breast-like and luminal epithelial/ER+. We illustrate the effect of the pSD in hierarchical clustering with average, complete and Ward linkage on these 496 genes.

Figure 3 shows the resulting dendrograms when applying each of these clustering algorithms to the dataset after scaling it in various ways. For average linkage, the pSD clearly outperforms the other options. It only misclassifies two observations, whereas the other methods split the large group of luminal epithelial/ER+ tumors (in cyan) in two or more clusters and fail to identify the smallest group (in green), which contains the Erb-B2+ tumors. Complete linkage gives better results for all types of scaling. However, without scaling or when using the range, the largest group of tumors gets split up
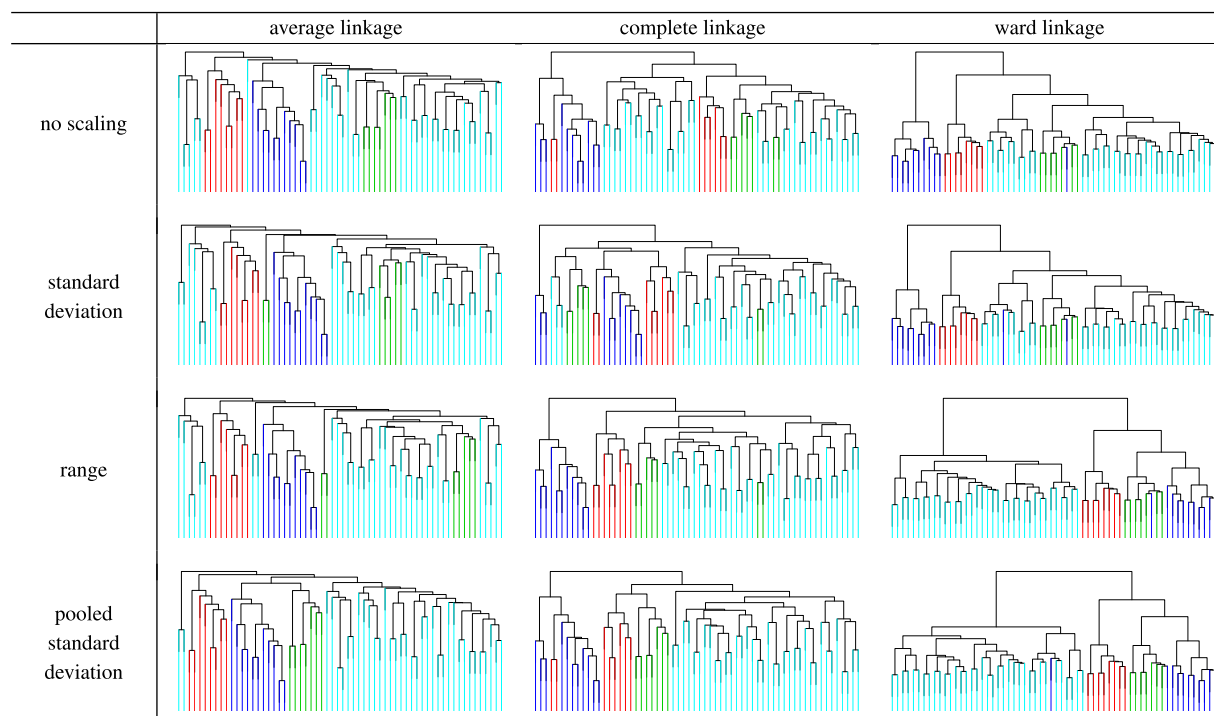
Fig. 3. The effect of variable scaling on the gene expression data. The colors correspond to the tumor type: basal-like in red, Erb-B2+ in green, normal-breast-like in dark blue and luminal epithelial/ER+ in cyan. The pSD generally yields superior recovery of the true clusters. (Color version of this figure is available at *Bioinformatics* online.)

Table 3. Genes for which the pSD is smaller than the SD

Variable	Description	SD/pSD
GF200: 96(8C12): 384(2F23)	ESTROGEN REGULATED LIV-1 PROTEIN (LIV-1) MRNA, PARTIAL CDS H29407 45	3.5
GF201: 96(88H2): 384(11O4)	GROWTH FACTOR RECEPTOR-BOUND PROTEIN 7 H53703 224	2.4
PEROU: 96(7F8): 384(20L16)	REGULATOR OF CHROMATIN, SUBFAMILY E, MEMBER 1 W63613 228	2.4
GF200: 96(13D9): 384(4G17)	CYTOCHROME P450, SUBFAMILY IIA (PHENOBARBITAL-INDUCIBLE), POLYPEPTIDE 7 T73031 61	2.2
PEROU: 96(8A1): 384(20B1)	68400 T57034 226	2.2
PEROU: 96(6A1): 384(20A2)	68400 T57034 227	2.2
GF200: 96(14D12): 384(4G24)	APOLIPOPROTEIN D H15842	2.2
PEROU: 96(9A9): 384(18B18)	IMMUNOGLOBULIN J CHAIN H24896 325	2.1

Note: The third column shows the ratios of these two scales.

into two groups. This does not happen when scaling with the SD, yet quite a few tumors are misclassified in this case. When scaling with the pSD, only two tumors are misclassified using complete linkage. Finally, with Ward's linkage, both the pSD and scaling with the range work well, with two and one misclassified tumor, respectively. Without scaling and with the SD however, the largest cyan cluster gets split up in two smaller clusters. In conclusion, the pSD yields better recovery of the true clusters than the other scaling methods.

In addition to improved clustering results, the pooled scaling procedure yields a diagnostic tool in the form of scale ratios. More precisely, we can compare the SDs of the variables with their pooled counterparts. Variables for which the ratio of these two scales is large typically show a clear grouping of the data, whereas variables for which this ratio is close to 1 do not distinguish clear groups. These ratios can thus be used as a fast and intuitive variable-screening procedure to identify potentially very informative variables, which is often a main research goal in this context.

In this example, only 8 of the 496 variables had a scale different from the SD. The information on those eight variables is presented in Table 3. Figure 4 shows the gene expressions for the four genes, which had the highest scale ratio. The colors again correspond to the four types of tumors. The top left panel shows the gene GF200:96(8C12):384(2F23), which clearly groups the red and blue points together and also shows high values for the majority of the cyan group. The top right and bottom left panels show the genes GF201:96(88H2):384(11O4) and PEROU:96(7F8):384(20L16), which distinctly separate the green tumors from the others. In the bottom right panel, the red tumors seem to have slightly lower values than the others, the blue tumors are grouped very tightly together and the cyan tumors appear to contain a sub-cluster with elevated values for this gene.

4 Discussion

We introduced a new approach to variable scaling prior to cluster analysis, which we call pooled scale estimators. The performance of pooled scaling is compared with the most common competitors on several popular clustering techniques, with a particular focus on the case of high-dimensional data with many uninformative noise variables. Scaling with the pooled scale estimators yields superior cluster recovery on the different clustering methods, in particular when the data contains a lot of noise variables. Since this is a common theme in the clustering of gene expression data, the performance was illustrated on breast cancer gene expressions in which it also outperformed the competition. The pooled scale estimates yield an additional diagnostic tool in the form of the ratio between pooled scales and
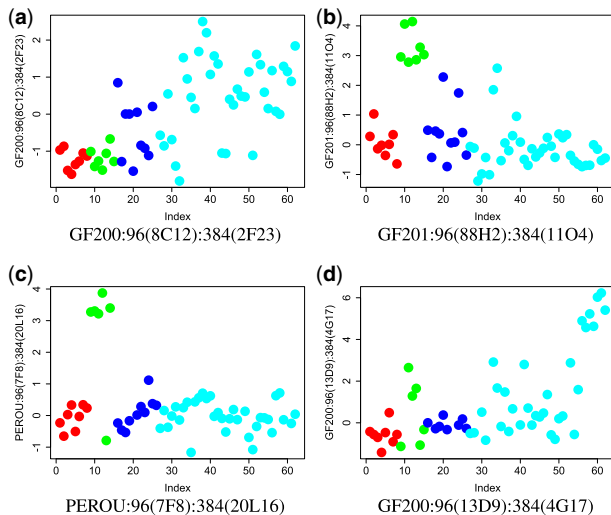
Fig. 4. Genes for which the pSD is smaller than the SD. These genes generally cluster the majority of the true groups together while sometimes identifying potentially interesting subclusters

default scales, which quantify the presence of cluster structure in the individual variables.

Acknowledgements

Funding

References

Anderberg,M.R. (2014) *Cluster Analysis for Applications: Probability and Mathematical Statistics: A Series of Monographs and Textbooks*. Vol. 19. Academic Press, Cambridge, Massachusetts.

Anderson,E. (1935) The irises of the Gaspé Peninsula. *Bull. Amer. Iris Soc.*, **59**, 2–5.

Arbelaitz,O. *et al.* (2013) An extensive comparative study of cluster validity indices. *Pattern Recognit.*, **46**, 243–256.

Breiman,L. *et al.* (1984) Classification and regression trees. Taylor & Francis, Monterey, CA.

Brun,M. *et al.* (2007) Model-based evaluation of clustering validation measures. *Pattern Recognit.*, **40**, 807–824.

Eisen,M.B. *et al.* (1998) Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. USA*, **95**, 14863–14868.

Fisher,R. (1936) The use of multiple measurements in taxonomic problems. *Ann. Eugen.*, **7**, 179–188.

Halkidi,M. *et al.* (2001) On clustering validation techniques. *J. Intell. Inf. Syst.*, **17**, 107–145.

Hartigan,J.A. (1975) *Clustering Algorithms*. John Wiley & Sons, Inc, Hoboken, New Jersey.

Hartigan,J.A. and Wong,M.A. (1979) Algorithm as 136: a k-means clustering algorithm. *J. R. Stat. Soc. Ser. C Appl. Stat.*, **28**, 100–108.

Hastie,T. *et al.* (2009) *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Science & Business Media, New York.

Hubert,L. and Arabie,P. (1985) Comparing partitions. *J. Classif.*, **2**, 193–218.

Jain,A.K. (2010) Data clustering: 50 years beyond k-means. *Pattern Recognit. Lett.*, **31**, 651–666.

Jain,A.K. and Dubes,R.C. (1988) *Algorithms for Clustering Data*. Prentice-Hall, Inc, New Jersey.

Kaufman,L. and Rousseeuw,P. (2009) *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley Series in Probability and Statistics. Wiley, Hoboken, New Jersey.

Lloyd,S. (1982) Least squares quantization in PCM. *IEEE Trans. Inf. Theory*, **28**, 129–137.

MacQueen,J. *et al.* (1967) Some methods for classification and analysis of multivariate observations. In: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, Vol. **1**, pp. 281–297. Oakland, CA, USA.

Maulik,U. and Bandyopadhyay,S. (2002) Performance evaluation of some clustering algorithms and validity indices. *IEEE Trans. Pattern Anal. Mach. Intell.*, **24**, 1650–1654.

Milligan,G. (1985) An algorithm for generating artificial test clusters. *Psychometrica*, **50**, 123–127.

Milligan,G. and Cooper,M. (1988) A study of standardization of variables in cluster analysis. *J. Classif.*, **5**, 181–204.

Milligan,G.W. and Cooper,M.C. (1985) An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, **50**, 159–179.

Perou,C.M. *et al.* (2000) Molecular portraits of human breast tumours. *Nature*, **406**, 747–752.

Qiu,W. and Joe,H. (2006) Generation of random clusters with specified degree of separation. *J. Classif.*, **23**, 315–334.

Qiu,W. and Joe,H. (2015) *clusterGeneration: Random Cluster Generation (with Specified Degree of Separation)*. R package version 1.3.4. https://cran.r-project.org/web/packages/clusterGeneration/index.html (April 2020, date last accessed).

Schaffer,C.M. and Green,P.E. (1996) An empirical comparison of variable standardization methods in cluster analysis. *Multivar. Behav. Res.*, **31**, 149–167.

Steinley,D. (2004) Standardizing variables in *K*-means clustering. In: Banks,D. *et al.* (eds) *Classification, Clustering, and Data Mining Applications. Studies in Classification, Data Analysis, and Knowledge Organisation*. Springer, Berlin, Heidelberg, pp. 53–60.

Stoddard,A.M. (1979) Standardization of measures prior to cluster analysis. *Biometrics*, **35**, 765–773.

Sugar,C.A. and James,G.M. (2003) Finding the number of clusters in a dataset. *J. Am. Stat. Assoc.*, **98**, 750–763.

Tibshirani,R. *et al.* (2001) Estimating the number of clusters in a data set via the gap statistic. *J. R. Stat. Soc. Series B Stat. Methodol.*, **63**, 411–423.

Vesanto,J. (2001) Importance of individual variables in the *k*-means algorithm. In: Cheung,D. *et al.* (eds) *Advances in Knowledge Discovery and Data Mining*. Springer, Berlin, Heidelberg, pp. 513–518.

Wang,H. and Song,M. (2011) Ckmeans.1d.dp: optimal *k*-means clustering in one dimension by dynamic programming. *R. J.*, **3**, 29–33.

Ward,J.H. Jr (1963) Hierarchical grouping to optimize an objective function. *J. Am. Stat. Assoc.*, **58**, 236–244.

OXFORD

Data and Text Mining

Pooled variable scaling for cluster analysis: supplementary material

Jakob Raymaekers [1] and Ruben H. Zamar [2]*

[1] Department of Mathematics, KU Leuven, Leuven, 3001, Belgium and
[2] Department of Statistics, University of British Columbia, Vancouver, V6T 1Z4, Canada.

*To whom correspondence should be addressed.

Abstract

This supplement contains the proof of proposition 1, the simulation results for the data with outliers and 4 additional real data examples.

1 Proof of Proposition 1

We prove the result in the case of k-means clustering. For k-medians clustering, the proof is entirely analogous.

Proof. **Part 1: effect of scaling on k-means clustering**
Let $\boldsymbol{x} = x_1, \ldots, x_n$ be a sample of univariate observations, $k > 0$ be a fixed natural number and suppose we have a solution to the k-means clustering problem for this value of k. Denote the centers of the clusters in this solution by $\boldsymbol{\mu} = \mu_1, \ldots, \mu_k$, the sets of indices of the clusters by $C_1, \ldots, C_k$ and the value of the objective function by $S_k(\boldsymbol{\mu}) = \sqrt{\frac{1}{n} \sum_{i=1}^n d_i(\boldsymbol{\mu})}$ where $d_i(\boldsymbol{\mu}) = \min_{j=1,\ldots,k} ||x_i - \mu_j||_2^2$. Let $s > 0$ be a positive real number, $t \in \mathbb{R}$ a real number and consider the sample $\boldsymbol{z} = z_1, \ldots, z_n$ where $z_i = (x_i - t)/s$.

We show that the clustering given by $C_j' = C_j$ and $\boldsymbol{\mu}' = \mu_1', \ldots, \mu_k'$ where $\mu_j' = (\mu_j - t)/s$ is a solution to the k-means problem on $\boldsymbol{z}$. Note first of all that if $C_j' = C_j$, we have $\mu_j' = (\mu_j - t)/s$ since the cluster centers are the sample means of the elements in the clusters and the sample mean is affine equivariant. Therefore, we also have that $s\, S_k'(\boldsymbol{\mu}') = S_k(\boldsymbol{\mu})$.

Suppose now that the clustering given by μ' does not solve the k-means problem on $z_1, \ldots, z_n$, i.e. there exists a clustering given by the centers $\boldsymbol{\theta} = \theta_1, \ldots, \theta_k$ such that $S_k'(\boldsymbol{\theta}) < S_k'(\boldsymbol{\mu}')$. Denote $d_i'(\boldsymbol{\theta}) = \min_{j=1,\ldots,k} ||z_i - \theta_j||_2^2$. Now consider the partition of the original dataset $x_1, \ldots, x_n$ given by the centers $s\boldsymbol{\theta} + t = s\,\theta_1 + t, \ldots, s\,\theta_k + t$. We then have $S_k(s\,\boldsymbol{\theta} + t) = \sqrt{\sum_{i=1}^n d_i(s\,\boldsymbol{\theta} + t)} = s\sqrt{\sum_{i=1}^n d_i'(\boldsymbol{\theta})} = s\, S_k'(\boldsymbol{\theta}) < s\, S_k'(\boldsymbol{\mu}') = S_k(\boldsymbol{\mu})$. This is a contradiction since $\boldsymbol{\mu}$ solves the k-means clustering of $x_1, \ldots, x_n$ and thus we cannot have a clustering

with a lower objective function.

Part 2: scale invariance of the gap analysis
Let k be fixed and consider the pooled within-cluster sum of squares $W_k(\boldsymbol{x}) = nS_k^2$ of the k-means clustering of $\boldsymbol{x}$. Let $\boldsymbol{z} = (\boldsymbol{x} - t)/s$ for a scale $s > 0$ and location $t \in \mathbb{R}$ as before. Then $W_k(\boldsymbol{z}) = W_k(\boldsymbol{x})/s^2$ and so

$$
\begin{aligned}
\mathrm{Gap}_{\boldsymbol{z}}(k) &= E_{U[z_{(1)}, z_{(n)}]}\left[\log\left(W_k\right)\right] - \log\left(W_k(\boldsymbol{z})\right) \\
&= E_{U[x_{(1)}, x_{(n)}]}\left[\log\left(W_k\right) - 2\log(s)\right] - \\
&\qquad \left(\log\left(W_k(\boldsymbol{x})\right) - 2\log(s)\right) \\
&= E_{U[x_{(1)}, x_{(n)}]}\left[\log\left(W_k\right)\right] - \log\left(W_k(\boldsymbol{x})\right) = \mathrm{Gap}_{\boldsymbol{x}}(k).
\end{aligned}
$$

Furthermore, we also have

$$
\begin{aligned}
\mathrm{Var}_{U[z_{(1)}, z_{(n)}]}\left[\log\left(W_k\right)\right] &= \mathrm{Var}_{U[x_{(1)}, x_{(n)}]}\left[\log\left(W_k\right) - 2\log(s)\right] \\
&= \mathrm{Var}_{U[x_{(1)}, x_{(n)}]}\left[\log\left(W_k\right)\right].
\end{aligned}
$$

So for every value of k, both the value of the gap statistic and the estimated variance of $\log(W_k)$ are invariant under the rescaling. Therefore, the optimal number of clusters resulting from the analysis based on the gap statistic is the same for $\boldsymbol{x}$ and $\boldsymbol{z}$.

2 Additional simulation results

(a) Single linkage

(b) Average linkage

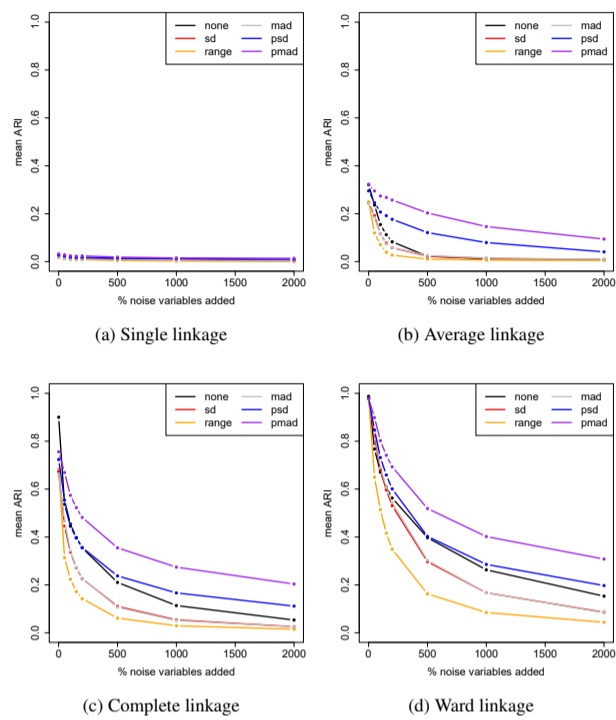(c) Complete linkage

(d) Ward linkage

Fig. 1: Simulation results for hierarchical clustering with single (a), average (b), complete (c) and Ward (d) linkage functions on data with 5 % outliers. The pooled scale estimators perform the best and the pmad is more robust to outliers than the psd.
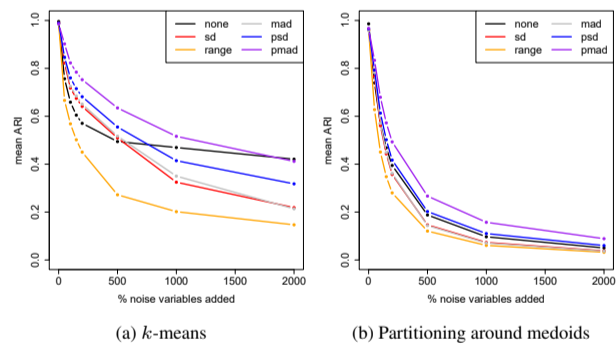

(a) k-means

(b) Partitioning around medoids

Fig. 2: Simulation results for k-means (a) and partitioning around medoids (b) on data with 5 % outliers. The pooled scale estimators perform better than the alternatives and the pmad is clearly more robust to outliers than the psd.

3 Additional real data examples

3.1 Lymphoma data

We analyze the lymphoma dataset first studied in Alizadeh *et al.* (2000) and publicly available in the R-package spls (Chung *et al.*, 2019). The dataset contains 4026 gene expression levels for 62 samples of 3 types of lymphoma. More precisely, there are 42 samples of diffuse large B-cell lymphoma, 9 samples of follicular lymphoma (FL), and 11 samples of chronic lymphocytic leukemia (CLL). The data was preprocessed as in Dettling and Bühlmann (2002).

We use the sparse hierarchical clustering with complete linkage of Witten and Tibshirani (2010) to cluster the data after scaling them with various estimators of scale. The dataset was also studied by Chung and Keles (2010) in the context of sparse PLS. The two best performing methods suggested that 50 or 197 features contain most of the information needed to find the groups in the data. We therefore cluster the dataset twice and fix the number of selected features in the sparse hierarchical clustering to be 50 and 197.

Figure 3 presents the result when choosing 50 features. Scaling with the pooled standard deviation yields 8 misclassified samples, the classical standard deviation gives 9 misclassified observations and the range 10. Without scaling, very little of the true clustering structure is recovered. When using 197 genes, the performance improves for the range and the pooled standard deviation, with 8 and 6 misclassified samples respectively (see Figure 4). The classical standard deviation fails to identify the smallest group of follicular lymphoma and not scaling gives mixed clusters. In summary, scaling with the pooled standard deviation yields the best cluster recovery.

(a) No scaling

(b) Standard deviation

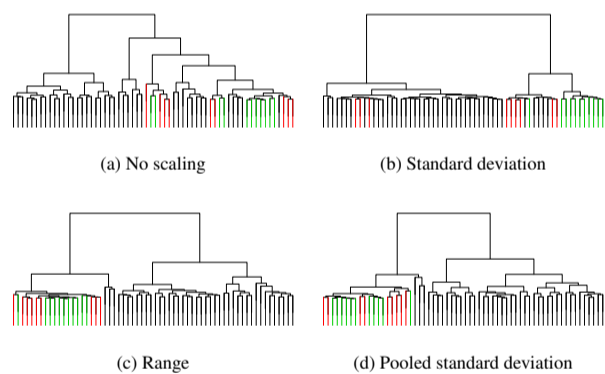(c) Range

(d) Pooled standard deviation

Fig. 3: Sparse hierarchical clustering with complete linkage on the lymphoma dataset using 50 features. Scaling with the pooled standard deviation yields the lowest number of misclassified observations.

3.2 Volatile Organic Compound Metabolites data

This dataset contains levels of volatile organic compounds (VOCs) in human urinary samples. The data was collected for the period 2015-2016 and is publicly available at the website of the National Health and Nutrition Examination Survey (NHANES), see NHANES (2019). It is known that long-term exposure to VOCs can lead to cancer and neurocognitive dysfunction. One of the most common causes of suspicious levels of VOCs is exposure to (tobacco) smoke. We therefore matched the patients in the dataset of VOCs with the available background information on their smoking behavior and defined two "true clusters": heavy smokers and non-smokers.

(a) No scaling

(b) Standard deviation

(c) Range
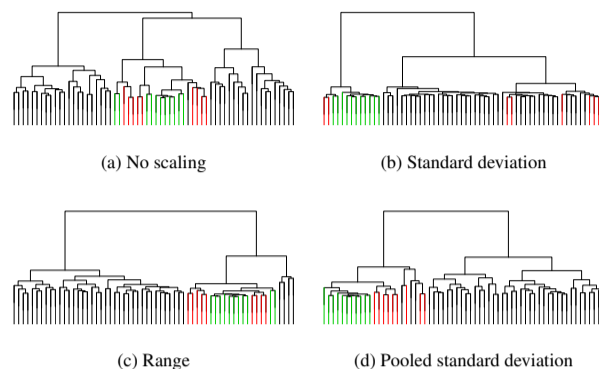
(d) Pooled standard deviation

Fig. 4: Sparse hierarchical clustering with complete linkage on the lymphoma dataset using 197 features. Again, scaling with the pooled standard deviation yields the lowest number of misclassified observations.

The plot of the first two score vectors from a principal component analysis shown in Figure 5 confirms that this grouping is in fact present in the data. Before clustering, we preprocessed the data by removing the observations which contained missing values as well as the variables with extremely low variance. We obtained a dataset of 20 levels of different VOCs for 522 patients. The names of the VOCs are listed in Table 1
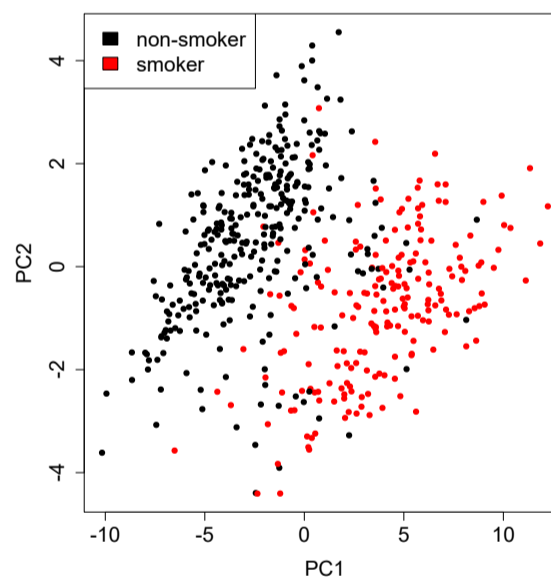

Fig. 5: Two first score vectors resulting from classical PCA on the VOC data. The two groups of smokers vs. non-smokers are clearly visible.

The clustering was done using regular k-means clustering with $k = 2$. Without scaling the data, the ARI value for the 2-means is 0.54. When scaling with the standard deviation, this drops down to 0.37. Scaling with the range yields an ARI of 0.47, while scaling with the pooled standard deviation has an ARI of 0.55. Surprisingly, there is only one variable for which the pooled standard deviation was smaller than the classical standard deviation. This variable is "URXCYM", which is the variable name for N-Acetyl-S-(2-cyanoethyl)-L-cysteine. It turns out that this is a well-known bio-marker for exposure to smoke, see e.g. Chen *et al.* (2019),

Variable Name	VOC name
URX2MH	2-Methylhippuric acid
URX34M	3- and 4-Methylhippuric acid
URXAAM	N-Acetyl-S-(2-carbamoylethyl)-L-cysteine
URXAMC	N-Acetyl-S-(N-methylcarbamoyl)-L-cysteine
URXATC	2-Aminothiazoline-4-carboxylic acid
URXBMA	N-Acetyl-S-(benzyl)-L-cysteine
URXBPM	N-Acetyl-S-(n-propyl)-L-cysteine
URXCEM	N-Acetyl-S-(2-carboxyethyl)-L-cysteine
URXCYM	N-Acetyl-S-(2-cyanoethyl)-L-cysteine
URXDHB	N-Acetyl-S-(3,4-dihydroxybutyl)-L-cysteine
URXHEM	N-Acetyl-S-(2-hydroxyethyl)-L-cysteine
URXHP2	N-Acetyl-S-(2-hydroxypropyl)-L-cysteine
URXHPM	N-Acetyl-S-(3-hydroxypropyl)-L-cysteine
URXIPM1	N-Acetyl- S- (4- hydroxy- 2- methyl- 2- butenyl)- L-cysteine
URXIPM3	N-Acetyl- S- (4- hydroxy- 2- methyl- 2- butenyl)-L-cysteine
URXMAD	Mandelic acid
URXMB3	N-Acetyl-S-(4-hydroxy-2-butenyl)-L-cysteine
URXPHE	N-Acetyl-S-(phenyl-2-hydroxyethyl)-L-cysteine
URXPHG	Phenylglyoxylic acid
URXPMM	N-Acetyl-S-(3-hydroxypropyl-1-methyl)-L-cysteine

Table 1. The volatile organic compounds in the VOC data.

since it typically results from the metabolization of acrylonitrile, a volatile liquid present in tobacco smoke. As can be seen in Figure 6, this variable indeed separates the majority of the smokers from the non-smokers. In conclusion, not only does scaling with the pooled standard deviation yield the best clustering results, it also flags a known important bio-marker for exposure to smoking.
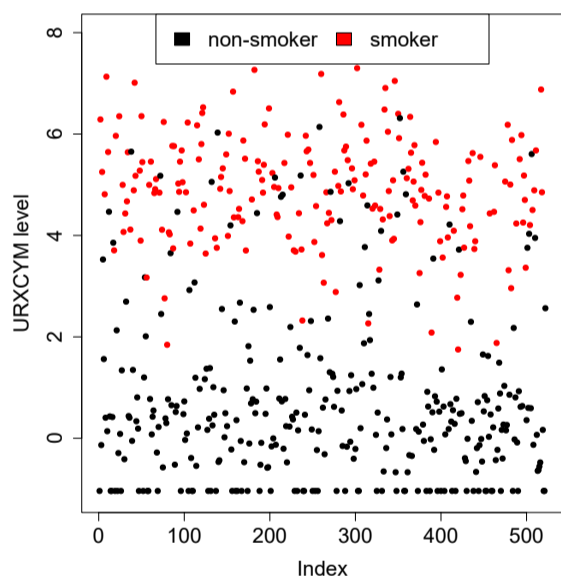

Fig. 6: The URXCYM variable for which the pooled standard deviation is clearly lower than its classical counterpart plays an important role in the clustering.

3.3 Leukemia data

We analyze the Leukemia data first studied by Golub *et al.* (1999). The data consists of the gene expression levels of 3051 genes for 38 patients, 27 with acute myeloid leukemia (AML) and 11 with acute lymphoblastic leukemia (ALL). The dataset is publicly available in the R-package

plsgenomics (Boulesteix *et al.*, 2018).

We apply hierarchical clustering to identify the two groups of patients with a different type of leukemia and again use four different scale estimators to scale the variables before clustering. Figure 7 shows the results of hierarchical clustering with Ward's linkage function. In this case, not scaling, scaling with the range and scaling with the pooled standard deviation perform equally well, misclassifying only 2 observations. Scaling with the standard deviation however splits the group of patients with AML in two and fails to recover the true clustering. Figure 8 shows the results of clustering the leukemia data using complete linkage. While complete linkage is clearly less suited to cluster these data, there are still noteworthy differences between the different scaling methods. No scaling gives 5 misclassified samples. Scaling with the range and pooled standard deviation gives 4 misclassified samples, whereas scaling with the standard deviation again messes up the clustering results. In summary, scaling with the pooled standard deviation or the range yields the best results for this dataset.

(a) No scaling　　　　　　(b) Standard deviation

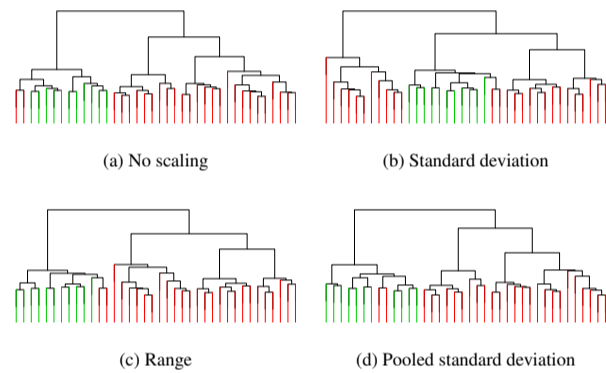(c) Range　　　　　　(d) Pooled standard deviation

Fig. 7: Hierarchical clustering with Ward linkage on the leukemia data. All methods work equally well, with the exception of the classical standard deviation.

(a) No scaling　　　　　　(b) Standard deviation
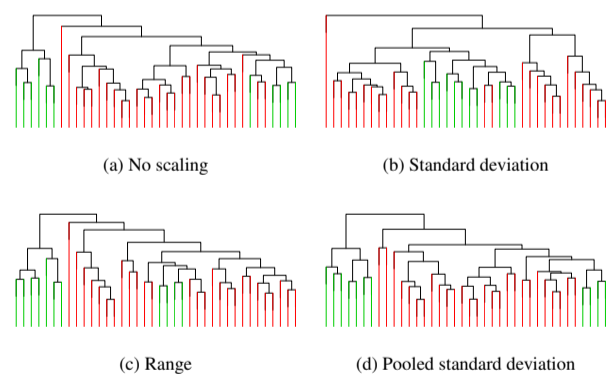
(c) Range　　　　　　(d) Pooled standard deviation

Fig. 8: Hierarchical clustering with complete linkage on the leukemia data. Scaling with the pooled standard deviation and the range gives the lowest number of misclassified patients.

3.4 Colon cancer data

We now analyze a dataset containing gene expressions from normal colon tissue samples as well as colon cancer samples. The data was collected from two datasets in the Gene Expression Omnibus (Edgar *et al.*, 2002) with IDs GSE8671 and GSE4183, and is publicly available in the R-package `antiProfilesData` (Bravo *et al.*, 2019). The probesets annotated to genes within blocks of hypomethylation in colon cancer defined in Hansen *et al.* (2011).

The data contains the expression levels of 5339 genes for 15 healthy patients and 23 patients with a tumor. We cluster the patients using hierarchical clustering with complete linkage. Figure 9 shows the resulting dendrograms, indicating that a perfect clustering is achieved without scaling or when scaling using the pooled standard deviation. When scaling with the standard deviation or range however, the true clusters are mixed and the recovery is quite poor.

(a) No scaling

(b) Standard deviation

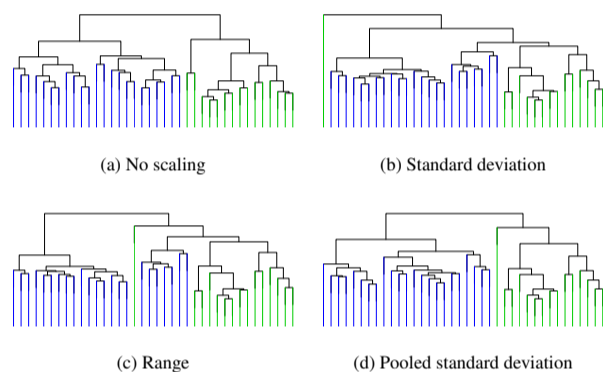(c) Range

(d) Pooled standard deviation

Fig. 9: Hierarchical clustering with complete linkage on the colon cancer data. Scaling with the pooled standard deviation and not scaling at all yields the best recovery of the true clusters.

As an added feature of computing the pooled standard deviations, we consider the 4 genes with the highest ratio of classical over pooled standard deviation. The Affy ID's of these genes are `207502_at`, `206134_at`, `207003_at` and `213921_at` and they are displayed in Figure 10. It is clear that all 4 of these genes clearly separate the patients with tumors from the healthy ones, indicating their biological importance.
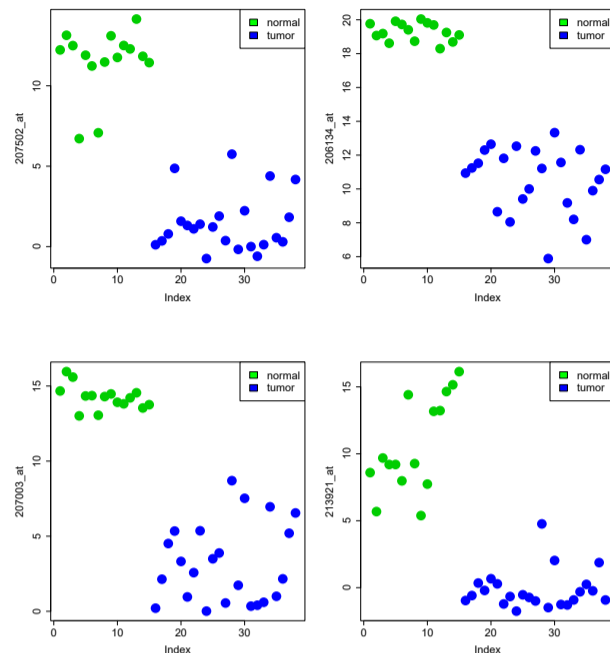
Fig. 10: The 4 genes in the colon data with the highest ratio of classical over pooled standard deviation. There is a clear separation between the healthy patients and those with a tumor.

References

Alizadeh, A. A., Eisen, M. B., Davis, R. E., Ma, C., Lossos, I. S., Rosenwald, A., Boldrick, J. C., Sabet, H., Tran, T., Yu, X., *et al.* (2000). Distinct types of diffuse large b-cell lymphoma identified by gene expression profiling. *Nature*, **403**(6769), 503–511.

Boulesteix, A.-L., Durif, G., Lambert-Lacroix, S., Peyre, J., and Strimmer., K. (2018). *plsgenomics: PLS Analyses for Genomics*. R package version 1.5-2.

Bravo, H. C., McCall, M., and Irizarry, R. A. (2019). *antiProfilesData: Normal colon and cancer preprocessed affy data for antiProfile building*. R package version 1.20.0.

Chen, M., Carmella, S. G., Sipe, C., Jensen, J., Luo, X., Le, C. T., Murphy, S. E., Benowitz, N. L., McClernon, F. J., Vandrey, R., Allen, S. S., Denlinger-Apte, R., Cinciripini, P. M., Strasser, A. A., al'Absi, M., Robinson, J. D., Donny, E. C., Hatsukami, D., and Hecht, S. S. (2019). Longitudinal stability in cigarette smokers of urinary biomarkers of exposure to the toxicants acrylonitrile and acrolein. *PLOS ONE*, **14**(1), 1–13.

Chung, D. and Keles, S. (2010). Sparse partial least squares classification for high dimensional data. *Statistical applications in genetics and molecular biology*, **9**(1).

Chung, D., Chun, H., and Keles, S. (2019). *spls: Sparse Partial Least Squares (SPLS) Regression and Classification*. R package version 2.2-3.

Dettling, M. and Bühlmann, P. (2002). Supervised clustering of genes. *Genome biology*, **3**(12), research0069–1.

Edgar, R., Domrachev, M., and Lash, A. E. (2002). Gene expression omnibus: Ncbi gene expression and hybridization array data repository. *Nucleic acids research*, **30**(1), 207–210.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J. R., Caligiuri, M. A., *et al.* (1999). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *science*, **286**(5439), 531–537.

Hansen, K. D., Timp, W., Bravo, H. C., Sabunciyan, S., Langmead, B., McDonald, O. G., Wen, B., Wu, H., Liu, Y., Diep, D., *et al.* (2011). Increased methylation variation in epigenetic domains across cancer types. *Nature genetics*, **43**(8), 768.

NHANES (2019). National Health and Nutrition Examination Survey Data. Hyattsville, MD: U.S. Department of Health and Human Services, Centers for Disease Control and Prevention. https://wwwn.cdc.gov/nchs/nhanes/.

Witten, D. M. and Tibshirani, R. (2010). A framework for feature selection in clustering. *Journal of the American Statistical Association*, **105**(490), 713–726.