Electronic Journal of Statistics
Vol. 13 (2019) 4849–4883
ISSN: 1935-7524
https://doi.org/10.1214/19-EJS1640

Quantile-based clustering

Christian Hennig[*], **Cinzia Viroli, and Laura Anderlucci**

Department of Statistical Sciences - University of Bologna
via Belle Arti 41
40126 Bologna, Italy
e-mail:
christian.hennig@unibo.it; cinzia.viroli@unibo.it; laura.anderlucci@unibo.it

Abstract: A new cluster analysis method, K-quantiles clustering, is introduced. K-quantiles clustering can be computed by a simple greedy algorithm in the style of the classical Lloyd's algorithm for K-means. It can be applied to large and high-dimensional datasets. It allows for within-cluster skewness and internal variable scaling based on within-cluster variation. Different versions allow for different levels of parsimony and computational efficiency. Although K-quantiles clustering is conceived as nonparametric, it can be connected to a fixed partition model of generalized asymmetric Laplace-distributions. The consistency of K-quantiles clustering is proved, and it is shown that K-quantiles clusters correspond to well separated mixture components in a nonparametric mixture. In a simulation, K-quantiles clustering is compared with a number of popular clustering methods with good results. A high-dimensional microarray dataset is clustered by K-quantiles.

Keywords and phrases: Fixed partition model, quantile discrepancy, high dimensional clustering, nonparametric mixture.

Received April 2019.

1. Introduction

In this paper we introduce a new clustering method, quantile-based or K-quantiles clustering. The method is fast and simple and can deal with large datasets. A special feature of the method is that it takes into account potential skewness of the within-cluster distributions.

The popular K-means method [19] represents all clusters by their centroids (cluster means) and assigns all points to the closest centroid. Quantile-based clustering represents the clusters by optimally chosen quantiles. Points are assigned to the closest quantile (or rather, in multidimensional data, distances to quantiles are summed up over the variables), but the distance measuring "closeness" treats points asymmetrically depending on which side of the quantile they are. This idea has been explored for supervised classification by [15], and here we present its application to clustering.

The algorithm for K-quantiles clustering works along the lines of Lloyd's classical K-means algorithm [25] and is in this way faster and simpler than

[*] Corresponding author.

many modern clustering methods, at the same time being more flexible than K-means.

There is some ambiguity in the literature about to what extent K-means is model-based. The K-means objective function can be motivated without reference to probability models; it formalizes optimal representation of all points in a cluster by the cluster centroid in the sense of least squares. It is therefore sometimes presented as assumption-free method. But K-means can also be derived as Maximum Likelihood (ML) estimator of a fixed partition model of spherical Gaussian clusters with equal within-cluster variances, which seems to be a quite severe assumption. Indeed K-means tends to produce spherical clusters, so it is hardly appropriate to call it assumption-free, although it is regularly applied to data that do not follow this model assumption. Whether this is appropriate does not depend so much on to what extent the model assumption is really fulfilled, but rather on whether the K-means characteristics matches the "shape" of clusters required in the application in hand. Different applications of cluster analysis ask for different kinds of clusters, and the user of cluster analysis needs to understand such characteristics of methods in order to choose an appropriate one for the application of interest [14].

In the same way, K-quantiles clustering can also be derived as ML estimator for a fixed partition model of generalized asymmetric Laplace distributions. This is helpful also for the construction of K-quantiles clustering, because it implies how to penalize variables against each other when using different quantiles for different variables. It also allows for an in-built scaling of variables that takes skewness into account. However, the main rationale of K-quantiles clustering is not the estimation of asymmetric Laplace distributions, but rather to define a general clustering principle that is almost as simple as K-means but more flexible by taking within-cluster skewness into account. Throughout the paper, the number of clusters K is treated as fixed; the estimation of K is left to future work.

We review the principle of K-means clustering in Section 2. In Section 3, quantile-based clustering is motivated and defined. First, we motivate it in a discrepancy-based nonparametric manner. Then we link it to a fixed partition model of asymmetric Laplace distributions. Some attention is paid to the penalty term introduced by ML-estimation in this model. A simple greedy algorithm is proposed, and various constrained versions of the quantile-based clustering are proposed, which allow for more parsimony and less computational effort. Section 4 is devoted to consistency theory. Quantile-based clustering is proved to be consistent in a nonparametric setting for the canonical clustering functional defined on a distribution, and another theorem shows that this functional will yield clusters that correspond to mixture components in mixtures with strongly separated nonparametric components. Section 5 presents a simulation study that includes high-dimensional setups, in which K-quantiles clustering is compared with some popular clustering methods. In Section 6, K-quantiles clustering is applied to a real microarray dataset with more than 3000 variables. Section 7 concludes the paper.

Supplementary material with some more detailed results and information is available on arxiv, [16].

2. *K*-means and distance-based probabilistic clustering

The aim of centroid-based clustering is to seek the best partition of n data vectors $\tilde{\mathbf{x}}_n = (\mathbf{x}_1, \dots, \mathbf{x}_n) \in (\mathbb{R}^p)^n$ into K disjoint subsets characterized by cluster prototypes (centroids) $\tilde{\boldsymbol{\xi}} = (\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_K)$ (the tilde denotes a collection of vectors rather than a single one).

In classical K-means [19] the 'best' partition $C = (C(1), \dots, C(n))$ is obtained by minimizing over $\tilde{\boldsymbol{\xi}}$ and C the variance function given by

$$V_{n,K}^{K-means}(\tilde{\boldsymbol{\xi}}, C, \tilde{\mathbf{x}}_n) = \sum_{i=1}^{n} \|\mathbf{x}_i - \boldsymbol{\xi}_{C(i)}\|^2, \tag{1}$$

where $\| \bullet \|$ denotes the L_2 or Euclidean distance, $C(i) \in \{1, \dots, K\}$ for $i = 1, \dots, n$. A classical estimation algorithm for minimizing $V_{n,K}^{K-means}$ consists of two steps sequentially iterated until convergence [25]. In the first step, for fixed $\boldsymbol{\xi}$ the best partition C is found by assigning each point to the nearest cluster center. Then in the second step, for fixed C, the centroids $\boldsymbol{\xi}_k$ $(k = 1, \dots, K)$ are estimated. Since the sum of squared Euclidean distances in (1) is minimized by the mean, the centroids $\boldsymbol{\xi}_k$ $(k = 1, \dots, K)$ are the within cluster means.

Although usually no probability assumption is mentioned when K-means is introduced, K-means can be derived as Maximum Likelihood (ML) estimator of a fixed partition model of spherical Gaussian clusters with equal within-cluster variances. According to such a model, $\mathbf{x}_1, \dots, \mathbf{x}_n$ are independently drawn from $\mathcal{N}(\boldsymbol{\xi}_{C(i)}, \sigma^2 \mathbf{I}_p)$, $i = 1, \dots, n$, where $C(i) \in \{K = 1, \dots, K\}$ are parameters giving the cluster memberships of the $\mathbf{x}_i$; as opposed to a mixture model, in a fixed partition model these are not modelled as random. The log-likelihood of such a model is

$$-\sum_{i=1}^{n} \frac{p}{2} \log \sigma^2 - \left\{ \frac{1}{2\sigma^2} \|\mathbf{x}_i - \boldsymbol{\xi}_{C(i)}\|^2 \right\},$$

which is maximized by the $\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_K, C(1), \dots, C(n)$ that minimize $V_{n,K}^{K-means}$, in other words, by K-means.

More generally, starting from an arbitrary distance from a prototype, denoted by $d(x, \xi)$, it is always possible to construct a probabilistic clustering model as proposed by [2] and [18]. The kernel of the distance-based density is the inverse of the exponential of the distance measure weighted by a positive concentration parameter λ:

$$f(x; \xi, \lambda) = \psi(\xi, \lambda) e^{-\lambda d(x, \xi)} \tag{2}$$

where $d(x, \xi)$ is a generic distance function from a location parameter ξ, $\lambda > 0$, and $\psi(\xi, \lambda)$ is a normalization constant such that $f(\mathbf{x}; \xi, \lambda)$ is a proper density function.

Distance-based models have been used by several authors [see 26, 7, 6] and adapted for classification in a mixture-based perspective by [28] for ranking data and by [1] for textual data.

Note that, when $d(x,\xi)$ is the L_2 (Euclidean) distance from the expected value of X, $\xi = E[X]$, the density (2) is the Gaussian distribution. When $d(x,\xi)$ is the L_1 distance, the density (2) coincides with the Laplace distribution. When $d(x,\xi)$ is the cosine distance and data are normalized to 1 according to the L_2 norm, (2) becomes the von Mises-Fisher distribution [1].

3. Quantile-based clustering

We now introduce a new clustering strategy based on the idea of assigning points to the closest quantile. Measuring "closeness" by the squared Euclidean distance is associated with the mean, in the sense that means optimize (1). Quantiles can also be characterized by minimizing a sum of discrepancies, although these discrepancies are not symmetric; they depend on which side of the quantile a point is. Using these discrepancies in "K-means style" leads to a simple clustering method that allows for within-cluster skewness.

3.1. Clustering based on the quantile discrepancy

Let X be a univariate random variable defined on $\mathbb{R}$ with probability cumulative function $F_X(x)$. Let $\theta \in [0,1]$ be a percentile and denote as $q(\theta)$ the corresponding quantile, such as $F_X^{-1}(\theta) = q(\theta) = \inf\{x : F_X(x) \geq \theta\}$.

The quantile $q(\theta)$ is the not necessarily unique value of ξ that minimizes the following variability measure:

$$\theta \int_{x>\xi} |x-\xi| dF_X(x) + (1-\theta) \int_{x<\xi} |x-\xi| dF_X(x) = \int \mathcal{Q}(x,\theta,\xi) dF_X(x), \quad (3)$$

where, for a single point x, we define the quantile discrepancy from ξ as a function $\mathcal{Q}: \mathbb{R} \times [0,1] \to [0,\infty)$:

$$\mathcal{Q}(x,\theta,\xi) = \left\{\theta + (1-2\theta)\mathbb{1}_{[x<\xi]}\right\} |x-\xi|. \quad (4)$$

For $\theta = 0.5$, this is the L_1 distance, but for $\theta \neq 0.5$ it is not symmetric and therefore not a distance. Not being based on squares, it shares with the L_1 distance its better resistance against outliers compared to the L_2 distance.

By definition the quantile discrepancy has a univariate nature. When X is a multivariate random variable on $\mathbb{R}^p$, the quantile discrepancy with respect to a generic vector of centroids $\boldsymbol{\xi}$ is defined as the sum of component-wise distances:

$$\mathcal{Q}^*(\mathbf{x},\boldsymbol{\theta},\boldsymbol{\xi}) = \sum_{j=1}^{p} \mathcal{Q}(x_j,\theta_j,\xi_j) = \sum_{j=1}^{p} \left\{\theta_j + (1-2\theta_j)\mathbb{1}_{[x_j<\xi_j]}\right\} |x_j - \xi_j|, \quad (5)$$

where θ_j can be variable-wise or a single common percentile for all variables.

The basic idea of quantile-based clustering is to use the quantile discrepancy instead of the squared L_2-distance in K-means, i.e., minimizing

$$V_{n,K}^{K-quantiles}(\boldsymbol{\theta},\tilde{\boldsymbol{\xi}},C,\tilde{\mathbf{x}}_n) = \sum_{i=1}^{n}\sum_{j=1}^{p} \mathcal{Q}(x_{ij},\theta_j,\xi_{C(i)j}), \quad (6)$$

where again $\tilde{\boldsymbol{\xi}} = (\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_K)$. $\boldsymbol{\theta}$ is assumed here to be the same for all clusters. $\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_K$ define the locations of the clusters. We call them "barycenters" from now on.

Proposition 1. *Let $\mathbf{x}_1, \dots, \mathbf{x}_n \in \mathbb{R}^p$, $\boldsymbol{\theta} \in (0,1)^p$ and $C(1), \dots, C(n) \in \{1, \dots, K\}$ so that $n_k = |\{\mathbf{x}_i : C(i) = k\}| > 0$. Then the empirical quantile vectors $\mathbf{q}_{nk}(\boldsymbol{\theta}) = \{q_{nk1}(\theta_1), \dots, q_{nkp}(\theta_p)\}$, $k = 1, \dots, K$, defined for $j = 1, \dots, p$ as*

$$q_{nkj}(\theta_j) = \inf \left\{ x_j : \frac{1}{n_k} \sum_{C(i)=k} \mathbb{1}_{[x_{ij} \leq x_j]} \geq \theta_j \right\}$$

satisfy

$$\sum_{i=1}^{n} \sum_{j=1}^{p} \mathcal{Q}(x_{ij}, \theta_j, q_{nkj}(\theta_j)) = \min_{\tilde{\boldsymbol{\xi}}} \sum_{i=1}^{n} \sum_{j=1}^{p} \mathcal{Q}(x_{ij}, \theta_j, \xi_{C(i)j}). \tag{7}$$

The proof of Proposition 1 is given in Appendix A.1.

(6) quantifies the discrepancy between the observations in a cluster and their barycenter given $\boldsymbol{\theta}$, and is therefore appropriate for finding the cluster barycenters and clustering the points, but it will not work well for finding $\boldsymbol{\theta}$. The problem of finding the optimal $\boldsymbol{\theta}$ will benefit from a model-based approach.

3.2. The fixed partition model and quantile-based clustering

Quantile-based clustering can be derived as ML estimator of a probabilistic model, similarly to K-means.

Consider the quantile discrepancy at $\xi = q(\theta)$, inserting $d(x, \xi) = \mathcal{Q}(x, q(\theta), \theta)$ in the distance-based density in (2) with $p = 1$ for the moment. The normalization constant is dependent on θ, and on ξ only through θ, so we can write $\psi(\theta, \xi, \lambda) = \psi(\theta, \lambda) = \lambda\theta(1-\theta)$. Therefore the quantile discrepancy based density is a spiky curve taking the general form:

$$f(x; \theta, \xi, \lambda) = \lambda\theta(1-\theta) e^{-\lambda\left\{\theta + (1-2\theta)\mathbb{1}_{[x<\xi]}\right\}|x-\xi|}, \tag{8}$$

where $\xi = q(\theta)$.

When $\theta = 0.5$, then $\xi = q(1/2)$ is the median, and the quantile-based density is the Laplace distribution. When $\theta \neq 0.5$ the quantile-based density is a special case of the asymmetric Laplace distribution [22] with expectation $E[X] = \xi + \frac{1-2\theta}{\lambda\theta(1-\theta)}$, variance $Var[X] = \frac{1-2\theta(1-\theta)}{(\lambda\theta(1-\theta))^2}$ and skewness $Skew[X] = \frac{(2(1-2\theta)(1-(1-\theta)\theta))}{(1-2(1-\theta)\theta)^{3/2}}$. Figure 1 shows some examples of its shape as θ varies.

For $\tilde{\mathbf{x}}_n = (\mathbf{x}_1, \dots, \mathbf{x}_n) \in (\mathbb{R}^p)^n$ we assume that the p variables are independent within clusters, and that the parameters θ and λ do not differ between clusters; the clusters are distinguished only by different barycenters $\boldsymbol{\xi}_k$, $k = 1, \dots, K$. We aim at finding a compromise here between flexibility on one side and parsimony and computational simplicity on the other side. In the K-means

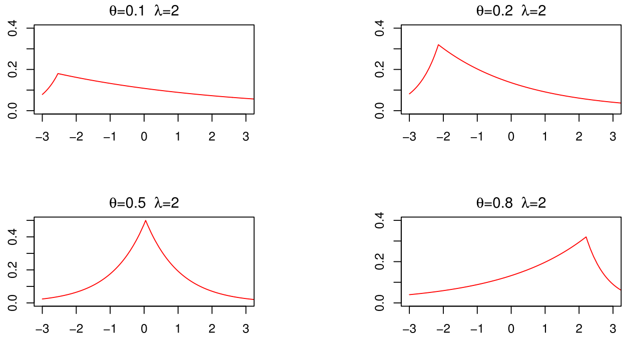

FIG 1. *Examples of quantile-based densities for different values of θ.*

model, variables are independent, all variables have the same within-cluster variances and clusters only differ regarding their centers. For quantile-based clustering, we define different levels of flexibility, see Section 3.5. For the moment we focus on the most general case of the models considered there, which allows both θ and λ to vary between variables, allowing for different distributional shapes and scales. Allowing them to differ between clusters as well, and incorporating within-cluster dependence would define a considerably more complex approach, both regarding the number of parameters and the computational burden. This is left for future research.

With parameter vector $\Theta = (\boldsymbol{\theta}, \tilde{\boldsymbol{\xi}}, \boldsymbol{\lambda}, C)$, the likelihood for a fixed partition asymmetric Laplace distribution model with independent variables is

$$f(\tilde{\mathbf{x}}_n; \Theta) = \prod_{i=1}^{n} \prod_{j=1}^{p} f(x_{ij}; \theta_j, \xi_{C(i)j}, \lambda_j).$$

Plugging in (8) and taking logs, the ML estimator is

$$\begin{aligned} T_{n,K}(\tilde{\mathbf{x}}_n) &= \arg\min_{\Theta} V_{n,K}(\Theta, \tilde{\mathbf{x}}_n), \\ V_{n,K}(\Theta, \tilde{\mathbf{x}}_n) &= \sum_{i=1}^{n} \sum_{j=1}^{p} \lambda_j \mathcal{Q}(x_{ij}, \theta_j, \xi_{C(i)j}) - n \sum_{j=1}^{p} \log \lambda_j \theta_j (1 - \theta_j), \end{aligned} \quad (9)$$

which for given $\boldsymbol{\theta}$ and $\boldsymbol{\lambda} = \mathbf{1}$ leads to the same clustering as (6). Proposition 1 enforces that $\xi_{11}, \ldots, \xi_{Kp}$ are the variable-wise within-cluster θ-quantiles, because the minimization with respect to $\tilde{\boldsymbol{\xi}}$ is independent of $\boldsymbol{\lambda}$. For given $(\boldsymbol{\theta}, \tilde{\boldsymbol{\xi}}, \boldsymbol{\lambda})$, the ML estimator of the clustering C is, for $i = 1, \ldots, n$:

$$C(i) = \arg\min_{k \in \{1,\ldots,K\}} \sum_{j=1}^{p} \lambda_j \mathcal{Q}(x_{ij}, \theta_j, \xi_{kj}) - n \sum_{j=1}^{p} \log \lambda_j \theta_j (1 - \theta_j). \quad (10)$$

We will therefore omit C in the parameter vector in the following. Here is the resulting definition.

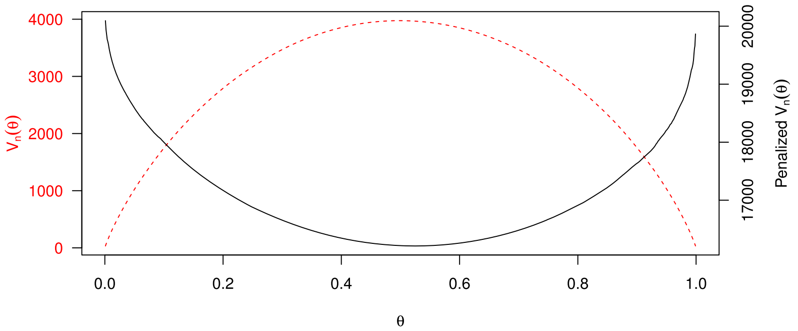

FIG 2. *Unpenalized (dashed red line) and penalized (black line) quantile dispersion for data generated from a Gaussian distribution.*

Definition 1. *Quantile-based (K-quantiles) clustering (with variable-wise θ and λ) is defined by*

$$T_{n,K}(\tilde{\mathbf{x}}_n) =$$

$$\arg\min_{\boldsymbol{\theta},\tilde{\boldsymbol{\xi}},\boldsymbol{\lambda}} \left(\sum_{i=1}^{n} \min_{k\in\{1,\ldots,K\}} \sum_{j=1}^{p} \lambda_j \mathcal{Q}(x_{ij},\theta_j,\xi_{kj}) - n\sum_{j=1}^{p} \log \lambda_j\theta_j(1-\theta_j) \right); \quad (11)$$

observations are clustered by (10) based on $(\boldsymbol{\theta},\tilde{\boldsymbol{\xi}},\boldsymbol{\lambda}) = T_{n,K}(\tilde{\mathbf{x}}_n)$.

3.3. Notes on penalization and scaling

Comparing (6) and (11) shows that the logarithmized normalization constant $-n\sum_{j=1}^{p}\log\lambda_j\theta_j(1-\theta_j)$ acts as a penalty term, penalizing θ_j too close to 0 or 1 and too small λ_j.

In order to illustrate why this is required (focusing on θ first), consider $K = 1$ and univariate data generated by a Gaussian distribution with some parameters μ and σ^2 and take $\xi = q_n(\theta)$, where $q_n(\theta)$ is the quantile computed on the sample of size n. The dashed red line of Figure 2 shows the shape of the dispersion $D_n(\theta) = \sum_{i=1}^{n}\mathcal{Q}(x_i,\theta,q_n(\theta))$ (w.l.o.g. $\lambda = 1$) on a large sample with $n = 10,000$ for a dense grid of values of the percentile between 0 and 1.

Since data have been generated by a symmetric distribution, the optimal value of θ should actually be $\frac{1}{2}$ corresponding to the median, and Figure 2 shows that the penalty is required to achieve this.

Figure 3 shows the penalized dispersion function for data generated by a symmetric distribution, by a positive skew distribution and by a negative skew distribution.

The parameters λ_j allow for implicit rescaling of the variables and scale equivariance. They need to be "penalized" because without the penalty term, $\boldsymbol{\lambda} \to 0$ would just enforce $V_{n,K} \to 0$.

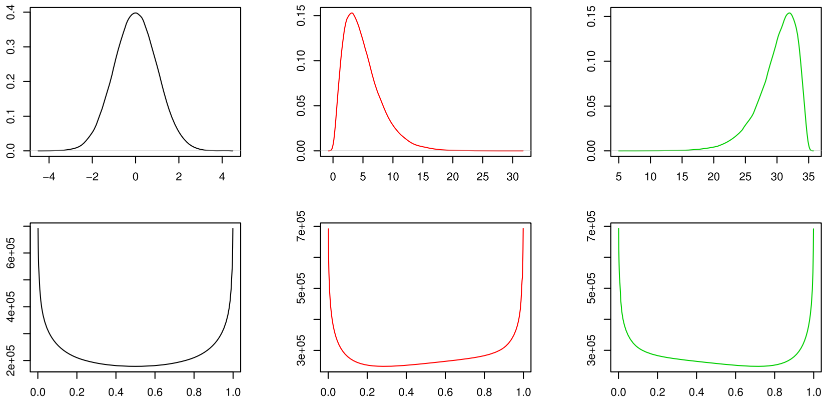

Fig 3. *In the second row of the panel the penalized dispersion function is plotted against θ for data generated according to the density functions depicted in the first row of the panel: a symmetric distribution, a positive skew distribution and a negative skew distribution.*

K-quantiles clustering is scale equivariant, which means that the clustering remains the same, and parameters change appropriately, if the variables in the data are multiplied by different constants.

Proposition 2. *For constants $\mathbf{c} = (c_1, \dots, c_p)^t$, $c_1, \dots, c_p \neq 0$, let $\tilde{\mathbf{x}}_N^* = (\mathbf{x}_1^*, \dots, \mathbf{x}_n^*)$ be defined by $\mathbf{x}_i^* = \mathbf{c}^t \mathbf{x}_i$, $i = 1, \dots, n$. Let*

$$T_{n,K}(\tilde{\mathbf{x}}_n) = \left(\boldsymbol{\theta}_{n,K}, \tilde{\boldsymbol{\xi}}_{n,K}, \boldsymbol{\lambda}_{n,K}\right), \ \mathbf{d} = \left(\frac{1}{c_1}, \dots, \frac{1}{c_p}\right)^t,$$

$\tilde{\boldsymbol{\xi}}_{n,K}^* = \left(\mathbf{c}^t \boldsymbol{\xi}_{n,K,1}, \dots, \mathbf{c}^t \boldsymbol{\xi}_{n,K,K}\right)$. Then,

$$T_{n,K}(\tilde{\mathbf{x}}_n^*) = \left(\boldsymbol{\theta}_{n,K}, \tilde{\boldsymbol{\xi}}_{n,K}^*, \mathbf{d}^t \boldsymbol{\lambda}_{n,K}\right),$$

and the corresponding clustering C is the same as for $T_{n,K}(\tilde{\mathbf{x}}_n)$.

The proof of Proposition 2 can be found in Appendix A.2.

Note that the parameters λ_j rescale the variables based on variation within clusters (only the discrepancy between x_{ij} and the cluster barycenter to which $\mathbf{x}_i$ is assigned are taken into account, see also Proposition 4 below). This is more appropriate than achieving scale equivariance by standardizing the variables beforehand based on the variance or some other dispersion measure, as is sometimes done for K-means, see [10]. Such methods will estimate a large dispersion if along a variable the separation between clusters is large, which may lead to downweighting of variables that are in fact very informative for clustering.

3.4. A greedy search algorithm

Lloyd's classical K-means algorithm [25] is a greedy algorithm, and K-quantiles clustering can also be computed using a fast greedy algorithm. This is based on the following two propositions, which show that $\boldsymbol{\theta}$ and $\boldsymbol{\lambda}$ minimizing $V_{n,K}$ can easily be found with all other parameters given. The propositions treat the case $p = 1$ w.l.o.g., because the variables can be treated separately for minimizing $V_{n,K}$ with respect to these parameters.

Proposition 3. *For one-dimensional $x_1, \dots, x_n$, given $\xi_1, \dots, \xi_K$, $C(1), \dots, C(n)$ and $\lambda > 0$, the solution to the problem*

$$\theta = \arg\min_{\theta^* \in (0,1)} \left(\sum_{i=1}^{n} \lambda \mathcal{Q}(x_i, \theta^*, \xi_{C(i)}) - n \log(\lambda\theta^*(1-\theta^*)) \right)$$

is given by one or both of the roots of the quadratic equation

$$\theta^2 \lambda \sum_{i=1}^{n} (x_i - \xi_{C(i)}) - \theta \left(2n + \lambda \sum_{i=1}^{n} (x_i - \xi_{C(i)}) \right) + n = 0.$$

Proposition 4. *For one-dimensional $x_1, \dots, x_n$, given $\xi_1, \dots, \xi_K$, $C(1), \dots, C(n)$ and $\theta \in (0,1)$, the solution to the problem*

$$\lambda = \arg\min_{\lambda^*} \left(\sum_{i=1}^{n} \lambda^* \mathcal{Q}(x_i, \theta, \xi_{C(i)}) - n \log(\lambda^*\theta(1-\theta)); \lambda > 0 \right)$$

is given by

$$\lambda = \frac{n}{\sum_{i=1}^{n} \mathcal{Q}(x_i, \theta_j, \xi_{C(i)})}.$$

Proofs of Propositions 3 and 4 are given in Appendix A.3 and A.4, respectively.

The greedy algorithm consists of an initialization step and a clustering step, which makes $V_{n,k}$ smaller in each step and is repeated until convergence. Because there are only finitely many possible clusterings, the algorithm will reach convergence after a finite number of steps (as does Lloyd's algorithm). For big datasets, if convergence takes too long, one could fix a maximum number of iterations. However, often convergence is reached very quickly; also the constrained methods proposed in Section 3.5 are faster. The scheme of the algorithm is the following:

1. *Initialization*: For each variable $j = 1, \dots, p$, choose randomly a value θ_j and K quantiles of equispaced probabilities as barycenters defined as $q_{nkj}(\theta^*_{kj})$, with $\theta^*_{kj} = (k-1)/2(K-1) + \theta_j/2$. Set $\lambda_j = 1$.
2. *Clustering step*: Repeat the following until $V_{n,K}(\boldsymbol{\theta}; \boldsymbol{\xi})$ stops changing:
    - (a) Compute the clustering $C(1), \dots, C(n)$ using (10).
    - (b) For $j = 1, \dots, p$ compute θ_j using Proposition 3.
    - (c) For $j = 1, \dots, p$ compute λ_j using Proposition 4.
    - (d) for $k = 1, \dots, K$, $j = 1, \dots, p$ compute the new barycenters $\xi_{kj} = q_{n_k kj}(\theta_j)$, where $n_k = \sum_{i=1}^{n} \mathbb{1}_{[C(i)=k]}$, and $q_{n_k kj}(\theta_j)$ denotes the quantile among the x_{ij} with $C(i) = k$.

Because of (10) and Propositions 3 and 4, $V_{n,K}$ is made smaller in every step, so the algorithm is guaranteed to converge. As usual, the algorithm only finds a local optimum of $V_{n,k}$. Therefore it is recommended to repeat the algorithms with a number of h different initializations (we use 30 as default), and the best solution is chosen according to the minimum value of $V(\boldsymbol{\theta}, \boldsymbol{\xi})$. The algorithm is implemented in the R package `QuClu` available on the CRAN Web page. Note that the proposed initialization of $\boldsymbol{\lambda}$ could be made scale equivariant by for example setting $\lambda_j = 1/s_j$ with s_j^2 being the sample variance of variable j, but this may come with the same issues as prior scaling of K-means, see Section 3.3.

3.5. Constrained versions

More parsimony and faster computation can be achieved by constraining the θ and λ-parameters. The algorithm described in Section 3.4 can easily be modified to accommodate these.

- Algorithm **CU**: **C**ommon θ and **U**nscaled variables.
  A common value of θ for all the variables is assumed, and variables are not implicitly scaled (the latter can make sense in applications in which the variables have comparable meanings and measurement units, if subject matter knowledge suggests that variable importance is proportional to variation). In this case we minimize the empirical loss function:

$$V_{n,K}(\theta, \boldsymbol{\xi}, \tilde{\mathbf{x}}_n) = \sum_{i=1}^{n} \min_{k \in \{1,\dots,K\}} \sum_{j=1}^{p} \mathcal{Q}(x_i, \theta, \xi_{kj}) - np \log(\theta(1-\theta)) \tag{12}$$

- Algorithm **CS**: **C**ommon θ and **S**caled variables through λ_j.
  A common value of θ is taken but variables are scaled through λ_j. Then the empirical loss function to be minimized is:

$$V_{n,K}(\theta, \boldsymbol{\xi}, \boldsymbol{\lambda}, \tilde{\mathbf{x}}_n) = \sum_{i=1}^{n} \min_{k \in \{1,\dots,K\}} \sum_{j=1}^{p} \lambda_j \mathcal{Q}(x_{ij}, \theta, \xi_{kj}) - n \sum_{j=1}^{p} \log(\lambda_j \theta(1-\theta))$$

- Algorithm **VU**: **V**ariable-wise θ_j and **U**nscaled variables.
  In this case we minimize

$$V_{n,K}(\boldsymbol{\theta}, \boldsymbol{\xi}, \tilde{\mathbf{x}}_n) = \sum_{i=1}^{n} \min_{k \in \{1,\dots,K\}} \sum_{j=1}^{p} \mathcal{Q}(x_{ij}, \theta_j, \xi_{kj}) - n \sum_{j=1}^{p} \log(\theta_j(1-\theta_j))$$

- Algorithm **VS**: **V**ariable-wise θ_j and **S**caled variables through λ_j. This is the most flexible method, with $V_{n,K}$ defined in (9).

Minimisation problems CS and VS are scale equivariant (Proposition 2 holds with the same proof for CS as well), whereas CU and VU are not.

4. Consistency theory

In this section we show that the parameters estimated by K-quantiles clustering from data are consistent estimators of the K-quantiles clustering functional, i.e., the version computed on an underlying distribution rather than on data. The first result of this kind for clustering seems to be the work of [31] on K-means clustering, and our Theorem 1 for K-quantiles clustering uses some of Pollard's ideas. Such consistency results have been shown for a number of clustering approaches, see, e.g., [33, 4, 5], but are still lacking for many methods.

It is well known in the case of K-means that consistency for the K-means functional does not imply that the estimated parameters (i.e., the K mean vectors) are consistent for the parameters of the Gaussian fixed partition model for which K-means is the ML estimator (see [3]), and in the same way the result presented here does not imply that K-quantiles clustering is consistent for estimating the parameters of a fixed partition model of asymmetric Laplace distributions as introduced in Section 3.2. Anyway, the consistency result given here is essentially nonparametric, for very general distributions, and it ensures the asymptotic stability of K-quantiles clustering, and the estimated parameters can be analyzed by considering the K-quantiles clustering functional. There is some literature on convergence rates and "performance guarantees" for K-means clustering (e.g., [24, 30]), but this relies on strong assumptions, and generalizing such results to K-quantiles clustering is beyond the scope of the present work. Instead, after the consistency result in Theorem 1, we show in Theorem 2 that the K-quantiles clustering functional defines clusters that are in line with "central sets" in a nonparametric mixture situation with sufficiently strong separation between mixture components. We are not aware of other results of this kind in the literature.

We consider the most flexible and general model defined above, with variable-wise θ and scaled variables, which is the most difficult one for proving consistency. Corresponding results for the less flexible models can be obtained more easily.

The proof relies heavily on showing that parameter estimators for large n do not leave a compact set, but (considering a single variable) $\lambda \to \infty$ and $\theta \to 0$ or $\theta \to 1$ may happen together without constraints on the parameter space (leading to the exponential distribution in the limit, which in practice could actually be integrated in the approach), causing trouble with uniform convergence arguments. This can be avoided by either constraining $\theta_j \in [r, 1-r]$, $r > 0$, or $\lambda_j \leq \lambda^+ < \infty$ for $j \in \{1, \dots, p\}$. We will impose the latter constraint here, so that results hold without further constraint for the unscaled case, i.e., $\boldsymbol{\lambda} = \mathbf{1}$.

The parameter space used here is

$$\begin{aligned}\mathcal{S} = \{(\boldsymbol{\theta}, \tilde{\boldsymbol{\xi}}, \boldsymbol{\lambda}):\ & \theta_j \in (0,1), \boldsymbol{\xi}_k \in \mathbb{R}^p, \lambda_j \in (0, \lambda^+], j \in \{1, \dots, p\},\\ & k \in \{1, \dots, K\}\}.\end{aligned}$$

We use the notation defined in (9) and (11); in case that the argmin is not unique, any solution can be taken. We modify (9) multiplying by $\frac{1}{n}$ in order to use stochastic convergence of means to expectations:

$$V_{n,K}(\Theta, \tilde{\mathbf{x}}_n) = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{p}\lambda_j \mathcal{Q}(x_{ij}, \theta_j, \xi_{C(i)j}) - \sum_{j=1}^{p}\log\lambda_j\theta_j(1-\theta_j).$$

For a given distribution P on $\mathbb{R}^p$ define

$$V_K(\boldsymbol{\theta}, \tilde{\boldsymbol{\xi}}, \boldsymbol{\lambda}, P) = \int \min_{k\in\{1,\dots,K\}}\sum_{j=1}^{p}\lambda_j\mathcal{Q}(x_j, \theta_j, \xi_{kj})dP(\mathbf{x}) - \sum_{j=1}^{p}\log\lambda_j\theta_j(1-\theta_j),$$

$$T_K(P) = (\boldsymbol{\theta}_K, \tilde{\boldsymbol{\xi}}_K, \boldsymbol{\lambda}_K) = \underset{(\boldsymbol{\theta},\tilde{\boldsymbol{\xi}},\boldsymbol{\lambda})\in\mathcal{S}}{\arg\min}\, V_K(\boldsymbol{\theta}, \tilde{\boldsymbol{\xi}}, \boldsymbol{\lambda}, P).$$

Let $S_{n,K} = V_{n,K}(T_{n,K}(\tilde{\mathbf{x}}_n), \tilde{\mathbf{x}}_n),\ S_K = V_K(T_K(P), P)$. In order to avoid issues due to label switching of the clusters, we consider consistency of lists $(\boldsymbol{\theta}, \tilde{\boldsymbol{\xi}}, \boldsymbol{\lambda})$, where $\tilde{\boldsymbol{\xi}}$ is the set of quantiles. Convergence and continuity are defined in terms of a distance d between two such lists $(\boldsymbol{\theta}_1, \tilde{\boldsymbol{\xi}}_1, \boldsymbol{\lambda}_1), (\boldsymbol{\theta}_2, \tilde{\boldsymbol{\xi}}_2, \boldsymbol{\lambda}_2)$ that is the maximum of $\|\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2\|, \|\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2\|$, and the maximum over the Euclidean distances between any element of $\tilde{\boldsymbol{\xi}}_i$ and its closest element of $\tilde{\boldsymbol{\xi}}_j$, $i \neq j, i, j = 1, 2$ (known as Hausdorff distance between $\tilde{\boldsymbol{\xi}}_1$ and $\tilde{\boldsymbol{\xi}}_2$).

The following assumptions will be required:

A1 $B = \int \|\mathbf{x}\|dP(\mathbf{x}) < \infty$.
A2 $T_k(P)$ is uniquely defined (up to cluster labelling) for $k = 1, \dots, K$.

A1 means that all involved integrals are finite; note that [31] requires $\int \|\mathbf{x}\|^2 dP(\mathbf{x}) < \infty$ for K-means. A2 enforces stability; as [31] noted for K-means, it implies that $S_K < S_{K-1} < \dots < S_1$ because if $S_k = S_{k-1}$ for some k, one could add any point to $\tilde{\boldsymbol{\xi}}_{k-1}$ to construct $\tilde{\boldsymbol{\xi}}_k$ that cannot have a worse value than S_k together with $\boldsymbol{\theta}_{k-1}, \boldsymbol{\lambda}_{k-1}$.

Theorem 1. *If $\mathbf{x}_1, \mathbf{x}_2, \dots \sim P$ i.i.d., and assumptions A1 and A2 hold, then, for $n \to \infty$: $T_{n,K}(\tilde{\mathbf{x}}_n) \to T_K(P),\ S_{n,K} \to S_K$ a.s.*

The proof of Theorem 1 is given in Appendix A.5.

The value of $T_K(P)$ for given P implies a clustering of $\mathbb{R}^p$ by

$$\gamma_{T_K(P)}(\mathbf{x}) = \arg\min_k \sum_{j=1}^{p}\lambda_{Kj}\mathcal{Q}(x_j, \theta_{Kj}, \xi_{Kkj})$$

for $\mathbf{x} = (x_1, \dots, x_p) \in \mathbb{R}^p$. The next result is about this implied clustering in case that $P_m,\ m \in \mathbb{N}$, is a sequence of mixture distributions with mixture components between which the separation becomes larger and larger with increasing m (we suspect that something like this is required to define clusters that can consistently be found by any method in such a nonparametric setting).

Here are some definitions and assumptions. Let $G_1, \dots, G_K$ be distribution functions on $\mathbb{R}^p$ defining distributions $Q_1, \dots, Q_K$ parameterized in such a way that 0 is their "center" in some sense; it could be the mode, the mean, the multivariate median or quantile; important is only that G_i is defined relative to 0. Let $\pi_1, \dots, \pi_K > 0$ mixture proportions with $\sum_{k=1}^{K}\pi_k = 1$. Assume

A3 For $m \in \mathbb{N},\ k \in \{1,\dots,K\}$ let $\boldsymbol{\rho}_{mk} \in \mathbb{R}^p$ sequences so that

$$\lim_{m\to\infty} \min_{k_1 \neq k_2 \in \{1,\dots,K\}} \|\boldsymbol{\rho}_{mk_1} - \boldsymbol{\rho}_{mk_2}\| = \infty.$$

A4 $\exists B_0 < \infty$ so that for all $k \in \{1,\dots,K\}\ :\ \int \|\mathbf{x}\| dG_k(\mathbf{x}) \leq B_0$.

Assumption A3 enforces the distance between central sets to become large enough for the statement to hold. A4 makes sure the involved expectations exist (note that a similar theorem could be proved for K-means, but this would require a bound on the mixture component-wise $E\|\mathbf{x}\|^2$).

Define a sequence of distributions P_m with distribution functions F_m on $\mathbb{R}^p$ by $F_m(\mathbf{x}) = \sum_{k=1}^K \pi_k G_k(\mathbf{x} - \boldsymbol{\rho}_{mk})$. Consider, for $\epsilon > 0$, the "central set" $\{\mathbf{x} :\ \|\mathbf{x}\| < \epsilon\}$ about 0. Then, by choosing $0 < \epsilon < \infty$ large enough,

$$\exists \delta > 0 :\ \forall k \in \{1,\dots,K\} :\ \pi_k Q_K\{\|\mathbf{x}\| < \epsilon\} \geq \delta. \tag{13}$$

The following theorem states that in this setup, when evaluating the K-quantiles clustering functional, eventually the different clusters include the full central sets of the different mixture components (and central sets can be of arbitrarily large though fixed radius), and in this sense the clustering corresponds to the mixture structure. The mixture components are allowed to overlap, although for $m \to \infty$ the overlap becomes arbitrarily small.

Theorem 2. *With the above definitions, assuming A3 and A4, for large enough m, the clusters of $T_K(P_m) = (\boldsymbol{\theta}_m, \tilde{\boldsymbol{\xi}}_m, \boldsymbol{\lambda}_m)$ can be numbered in such a way that for $k \in \{1,\dots,K\}$:*

$$\{\mathbf{x} :\ \|\mathbf{x} - \boldsymbol{\rho}_{mk}\| < \epsilon\} \subseteq \{\mathbf{x} :\ \gamma_{T_K(P_m)}(\mathbf{x}) = k\}.$$

The proof of Theorem 2 is given in Appendix A.6.

5. Simulation study

The performance of the K-quantiles clustering algorithm is evaluated in an extensive simulation study. We generate p vectors from K, $K = 2, 3, 5$, populations, $\mathbf{X}^{(K)}$, according to five different scenarios:

1. In the first scenario, we consider symmetric Student t-distributed variables W_j $(j = 1,\dots,p)$ with three degrees of freedom, and we simulate K location-shifted populations from W_j, each shift from the closest population being unitary [e.g. $X_j^{(1)} \sim W_j$, $X_j^{(2)} \sim (W_j + 1)$, $X_j^{(3)} \sim (W_j - 1)$, $\dots$].
2. In the second scenario, we test the behaviour of the clustering algorithm in highly skewed data by generating identically distributed vectors W_j $(j = 1,\dots,p)$ from a multivariate zero-centered Gaussian distribution, transforming them using the exponential function and shifting contiguous populations by 0.6 [e.g. $X_j^{(1)} \sim exp(W_j)$, $X_j^{(2)} \sim (exp(W_j) + 0.6)$, $X_j^{(3)} \sim (exp(W_j) - 0.6)$, $\dots$].

3. In the third scenario, we consider different distributions for the p variables. We first generate W_j from a multivariate zero-centered Gaussian distribution and then split p into five balanced blocks to which we apply different transformations:
   (i) a location shift [e.g. $X_j^{(1)} \sim W_j$, $X_j^{(2)} \sim (W_j + 0.7)$, $X_j^{(3)} \sim (W_j + 1.4)$, $\ldots$];
   (ii) an exponential transformation on the shifted populations at (i) [e.g. $X_j^{(1)} \sim exp(W_j)$, $X_j^{(2)} \sim exp(W_j + 0.7)$, $X_j^{(3)} \sim exp(W_j + 1.4)$, $\ldots$];
   (iii) a logarithmic transformation on the shifted populations at (i) [e.g. $X_j^{(1)} \sim log(|W_j|)$, $X_j^{(2)} \sim log(|W_j + 0.7|)$, $X_j^{(3)} \sim log(|W_j + 1.4|)$,$\ldots$];
   (iv) a quadratic transformation on the shifted populations at (i) [e.g. $X_j^{(1)} \sim W_j^2$, $X_j^{(2)} \sim (W_j + 0.7)^2$, $X_j^{(3)} \sim (W_j + 1.4)^2$, $\ldots$];
   (v) a square root transformation on the shifted populations at (i) [e.g. $X_j^{(1)} \sim \sqrt{|W_j|}$, $X_j^{(2)} \sim \sqrt{|W_j + 0.7|}$, $\mathbf{X}_j^{(3)} \sim \sqrt{|W_j + 1.4|}$, $\ldots$].
4. In the fourth scenario, we simulate different distributional shapes and levels of skewness even for different classes within each variable. Within each class, data are generated according to beta distributions, $X_j^{(k)} \sim Beta(a, b)$, $j = 1, \ldots, n$ and $k = 1, \ldots, K$, with parameters a and b in the interval $(1, 10)$ randomly generated for each class within each variable. The absolute difference between the class expected values for each variable is bounded from above by 0.2 (this is done in order to not make the clustering task too easy; as cluster differences are aggregated over many dimensions, simulated clusters may be so strongly separated that every method can find them easily).
5. The fifth scenario is similar to the fourth one. Within each class, data are generated according to beta distributions, $X_j^{(k)} \sim Beta(a, b)$, $j = 1, \ldots, n$ and $k = 1, \ldots, K$, with parameters a and b randomly chosen to be in the intervals: $(0, 1)$ and $(1, 5)$, or $(0, 1)$ and $(1, 10)$, $(1, 3)$ and $(5, 10)$, $(1, 3)$ and $(1, 3)$ so as to guarantee a higher level of skewness for some variables, for each class within each variable. The absolute difference between the class expected values for each variable is bounded from above by 0.1.

For each of the five scenarios and for each set of K populations, $K = 2, 3, 5$, we evaluate combinations of p = 50, 100, 500, n = 50, 100, 500, different percentages of relevant variables for grouping structure, i.e., 100%, 50% and 10%, independent or dependent variables (scenario 1-3 only), for a total of 648 different settings. In the "dependent variables" case, a dependence structure is introduced by generating variables $W_1, \ldots, W_p$ from either a t or a Gaussian distribution with random correlation matrix based on the method proposed by [20], so that the correlation matrices are uniformly distributed over the space of positive definite correlation matrices, with each correlation marginally distributed as $Beta(p/2, p/2)$ on the interval $(-1, 1)$. The irrelevant noise variables are generated independently of each other from the base distribution of each scenario. For each setting we simulate 100 datasets and we record the Adjusted

Rand Index (ARI) [17] of the yielded classification compared with the true cluster membership. The performance of the algorithm is also examined in the case of imbalanced clusters: for the settings with $K = 3$ and $n = 500$ the group sizes are set to $n_1 = 50$, $n_2 = 150$ and $n_3 = 300$.

We compare the K-quantiles clustering algorithms' capability of recovering the original cluster memberships with those of seven other clustering methods: two model-based clustering approaches (mixture of Gaussians, mixture of factor analyzers) [27], K-means algorithm [25], Partition Around Medoids [21], agglomerative hierarchical clustering with unweighted pair group method (UPGMA), spectral clustering [29] and affinity propagation [8]. The inclusion of irrelevant variables may prompt the idea that also clustering methods with variable selection should be tried out; however, variable selection is usually defined on top of an existing clustering method without variable selection, see, e.g., [9]. Such ideas can be applied to K-quantiles clustering as well as to the competing methods, which we leave for future research. Mixtures of Skew-t distributions were also considered; however, due to computational difficulties in high dimensions, solutions were available only 20% of times, so we do not present results.

Details about the implementation and parameter tuning of these methods are given in the Appendix B; detailed tables of simulation results are in the supplementary material.

More specifically, we evaluate the accuracy of each clustering method as its ARI minus the ARI of the Common Unscaled K-quantiles clustering algorithm, divided by the average ARI in the given setting (for computing this average, the three percentages of relevant variables are aggregated in order to avoid blowing up small differences between uniformly small ARI values for only 10% relevant variables by small denominators; where methods do not deliver a solution, the ARI has been set to zero). This is done for the sake of enabling a simpler display of the many results, because it allows to aggregate results for different K, n, p, dependence structure, and percentage of relevant variables by scaling all these results so that they become comparable. Raw ARI values are given in the supplementary material. The aggregated distributions of these rescaled results are displayed in the boxplots of Figure 4 for the balanced cluster settings. Results for imbalanced clusters are similar and can be found in the Appendix B.1.

For all methods, the capability of recovering the original cluster membership improves as the sample size increases. For the K-quantile clustering, this is particularly evident in the scenarios where the percentage of relevant variables is very low; in all the other cases, in fact, results are generally very good and no remarkable difference due a different sample size can be noticed. All the methods, for fixed sample size and percentage of relevant variables, seem to perform better as p increases in almost all of the settings. As could be expected, clustering performances worsen as the number of irrelevant variables increases.

The K-quantiles methods perform very well in most situations compared to other clustering approaches. In the scenarios with identical distributional shapes and symmetric variables, solutions from the quantile clustering are mostly

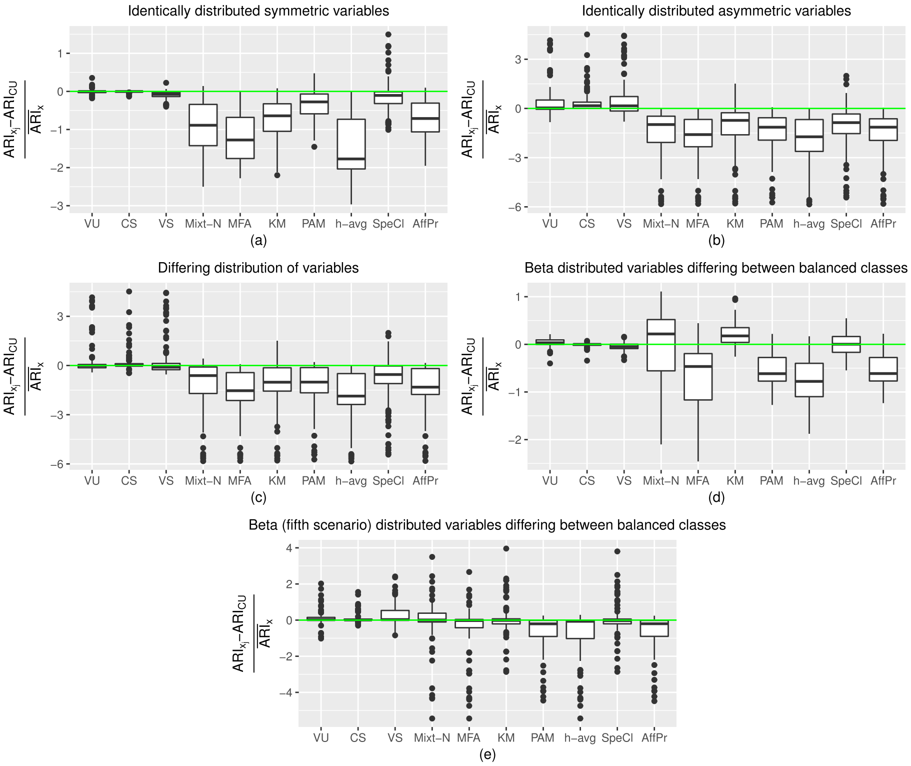

Fig 4. *Performance of different clustering algorithms relatively to the Common Unscaled K-quantiles clustering algorithm for balanced clusters. The labels along the horizontal axis refer to the different methods: CS, Common Scaled k-quantile; VU, Variable-wise Unscaled k-quantile; VS, Variable-wise Scaled K-quantile;* Mixt-N*, mixture of Normal distributions;* MFA*, mixture of factor analyzers;* KM*, k-means clustering;* PAM*, Partition Around Medoids algorithm;* h-avg*, hierarchical clustering with average linkage (UPGMA);* SpeCl*, spectral clustering;* AffPr*, affinity propagation. The five panels show the distribution of the Adjusted Rand Index for (a) identically distributed symmetric variables, (b) asymmetric variables, (c) different distributions of variables, (d) different distributions of classes within variables in balanced and unbalanced populations and (e) different (skew) distributions of classes within variables in balanced and unbalanced populations.*

preferable to those from any other method. Not surprisingly, common θ quantile procedures, i.e. CU and CS, slightly outperform those with variable-wise θ_js.

In the settings with identical distributional shapes and asymmetric variables, K-quantiles clustering methods outperform all other methods clearly and more or less uniformly; here, procedures with a variable-wise θ_j, i.e. VU and VS, seem to produce a slightly better clustering.

With different distributions of variables, the K-quantiles clustering methods again show very good global results. Only occasionally, in some situations with just 10% relevant variables, Gaussian mixtures, K-means and spectral clustering can improve on K-quantiles.

In the fourth scenario with beta distributions differing between variables and classes within variables, the K-quantiles clusterings do not always outperform the other methods: while they generally produce good results, they often fall behind the accuracy of the K-means algorithm, spectral clustering, and also Gaussian mixtures. The reason why this happens is that although these distributions are skew, their tails vanish outside the unit interval, and often the difference between means is the most distinct feature discriminating the clusters. Therefore a squared loss function is suitable for finding them. This contrasts with the fact that K-quantiles beats these methods for symmetric but t-distributed data in the first scenario, despite the fact that K-means and Gaussian mixtures implicitly assume symmetry, as opposed to K-quantiles; however, the squared loss function is more affected by outliers in these cases.

The results of the fourth scenario prompted us to set up the fifth one, with parameters of the beta random variables chosen from different intervals so that there is more extreme skewness, and information about clustering is rather connected to distributional features other than the means. In this situation, the K-quantiles VS algorithm is the best. K-means and spectral clustering still yield fairly good results, although worse than the K-quantiles algorithms. Gaussian mixtures also still do well, probably because flexible covariance matrices are still versatile here to adapt to these setups. Their median performance is about on a par with three of the four K-quantiles algorithms (results vary depending on whether p is rather large compared to n or not, see the supplementary material) but worse than the VS algorithm.

Generally, the capability of recovering the clustering memberships and the rankings of the methods do not change much with dependence, although performances are slightly better under independence. Similarly, the ranking of the methods does not strongly depend on the number of clusters, nor on the presence of imbalanced clusters.

The table in the Appendix C provides some information on computing times. Currently our implementation for running the K-quantiles is coded in R; faster implementations are certainly possible. However, our experiments show that the growth in computation time with n is much slower than for PAM and the mixture model-based methods, so that for the largest data format that we tried ($n = 50000, p = 100$) our K-quantiles implementation is substantially faster than all mixtures and PAM, beaten only by K-means (the hierarchical clustering does not deliver a solution). This demonstrates that K-quantiles have the potential to be used with very large datasets.

The number of clusters K is treated as fixed here, and estimating K is left to future work. However, Figure 5 shows the average behaviour (over 100 replications) of the quantile discrepancy function V_n for different numbers of clusters. Data come from $K = 3$ populations generated according to the second scenario, with an overall sample size $n = 500$ and 50 features. The discrepancy function decreases with increasing K; an elbow point is noticeable for $K = 3$, which is an indication that such curves may be of some use to choose K.

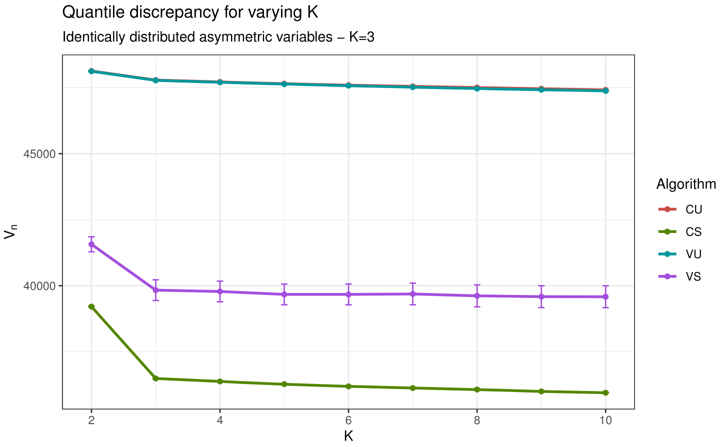

Fig 5. *Quantile discrepancy of the K-quantiles algorithm for different values of K, averaged over 100 replications. Error bars span ±1 standard error.*

The supplementary material gives some information about the distribution of estimated parameters in the third scenario and how this is related to skewness and variance of the cluster-wise distributions.

6. Application to gene expression data

For illustration we apply the K-quantiles clustering algorithms to gene expression data from the leukaemia microarray study of [11]. The dataset contains the expression levels of 3051 genes for 38 leukaemia patients, obtained from acute leukaemia patients at the time of diagnosis. The study reports that 27 subjects have Acute Lymphoblastic Leukaemia (ALL), while 11 have Acute Myeloid Leukaemia (AML). The objective is to group the set of 38 patients so as to reflect the corresponding leukaemia diagnosis by employing information coming from their gene expression levels. In general, for methods that are not scale invariant, results depend on the scale. We consider results for unscaled data and for data with all variables scaled to unit variance.

Data are taken from the R package `plsgenomics` and are analysed by the same clustering methods described in Section 5. The number of true clusters for all the methods is taken as known and set equal to 2. As different versions of Mclust delivered different results, we have tried out different initializations and we chose the one with largest likelihood. For K-means five random starts are run. The default settings of all the other algorithms are considered.

Results from all the other methods are shown in Table 1. As can be seen, the K-quantiles clustering algorithm with variable-wise θ_j and scaled variables via λ_j is able to perfectly recover the original clustering memberships. When using the unscaled version, the performance of VU is still pretty good and superior to

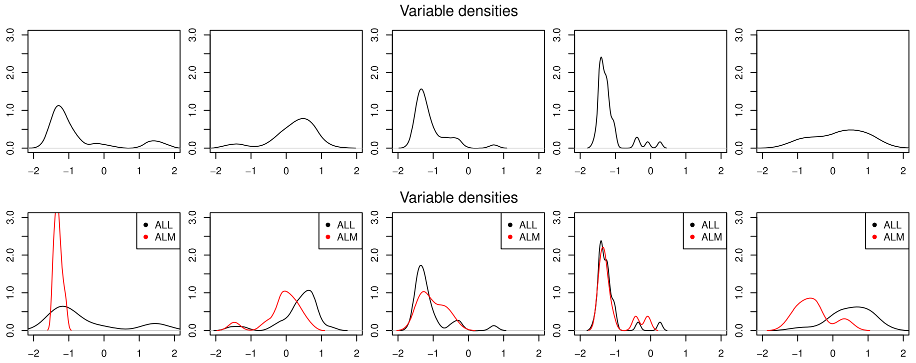

FIG 6. *Leukaemia dataset: densities of five randomly selected variables (gene expression levels; fitted by R's density function with default settings). First row: all observations together. Second row: by true cluster.*

the other solutions. Quantile methods with common θ, whether using the scaled or the unscaled version, are not able to detect the grouping structure identified by the diagnosis: this is probably due to the fact that the distribution of the expression levels is really different for different genes (see Figure 6, where the density of a random sample of gene expression levels is plotted).

Mixture of Factor Analyzers could not return any solution, as it has ended up with errors.

The mixture of Gaussians and K-means yield exactly the same clustering (up to label switching); their results are still good. As the number of variables is very large, `Mclust` could only estimate mixture of Gaussians with spherical or diagonal covariance matrices, reducing to 6 out of 14 possible parsimonious models, namely: `EII` (spherical, equal volume), `VII` (spherical, unequal volume), `EEI` (diagonal, equal volume and shape), `VEI` (diagonal, varying volume, equal shape), `EVI` (diagonal, equal volume, varying shape), `VVI` (diagonal, varying volume and shape).

Spectral clustering and affinity propagation provide overall good results but worse than the mixture of Gaussians.

The second column of Table 1 reports the ARI of the clustering methods after having standardized the variables. Apparently feature scaling has removed part of the cluster-separation signal from the data and the obtained clustering is worse for most methods. This shows that in order to find an appropriate scaling, global standardization does not always work well.

7. Conclusion

K-quantiles clustering is a new clustering method based on representing clusters by quantiles of the within-cluster distribution. It can be interpreted as Maximum Likelihood estimator for a fixed partition model of asymmetric Laplace distributions, but like K-means it is not in the first place meant to be as-

Table 1
Adjusted Rand Index multiplied by 100 for the leukaemia data set, respectively obtained on the original (unscaled) and on the scaled features. As shown earlier, CS and VS are scale invariant; any potential changes in their results are due to random initialisation and not to scaling.

Method	ARI unscaled data	ARI scaled data
CU	3.28	−4.87
VU	100.00	69.92
CS	−2.61	(scale invariant)
VS	89.13	(scale invariant)
Mixture of Normals	79.27	32.00
Mixture of FA	NA	NA
k-means	79.27	11.45
pam	61.20	61.20
h-avg	−3.06	−3.06
SpeCl	69.92	32.01
AffPr	61.20	61.20

sociated with a specific model assumption, but rather to provide an intuitive objective function that allows for within-cluster skewness and is easy to optimize locally using a Lloyd-type algorithm. In our simulations the method did well on a wide range of within-cluster models different from the asymmetric Laplace.

[13] encourages researchers to give potentially informal descriptions of what specific kinds of clusters a new clustering method is meant to find. The development of K-quantiles clustering was motivated in the first place by the potential of the quantile-based discrepancy to add flexibility to K-means, particularly regarding within-cluster skewness. The underlying model suggests that clusters can be distributions of which the marginals are unimodal and potentially skew; Theorem 2 shows that sufficiently well separated subpopulations will be K-quantiles clusters even if not unimodal (as long as K is fixed and there are more than K modes, it is hardly possible to have only unimodal clusters). Similarly to K-means, the K-quantiles objective function sums up information over the different variables. This does not necessarily mean that variables have to be independent within clusters, but information about dependence is not used. The discrepancy is just an aggregation of variable-wise information. Clusters will not be rotation invariant and information carried in the original variables will be lost when considering linear combinations such as principal components. The clusters are treated as of the same shape, although with enough separation this does not stop the method from finding clusters with different shapes, see Theorem 2 and the simulations. The advantage of this is that a parsimonious parametrization allows the handling of high-dimensional data. An obvious generalization would be to allow the parameters θ and λ to vary between clusters, but this is likely to require considerably more computational effort. The use of unsquared distances gives outliers less influence on the cluster barycenters than in K-means or ML estimators for Gaussian mixtures.

The number of clusters K is fixed here, and estimating K is left to future work. Many methods for estimating the number of clusters are based on computing a clustering for a range of values of K and then cluster validation indexes or stability assessments are used to pick the best K [12, 23]. Such an approach can be used for estimating K together with K-quantiles clustering in the same manner as with K-means or K-medoids. Similarly, principles for variable selection that exist for K-means and other clustering methods could be applied to K-quantiles.

The penalization of the quantile defining probability θ and the scaling parameter λ as derived here from a fixed partition model of asymmetric Laplace distributions may also be helpful for quantile-based supervised classification as introduced in [15].

Appendix A: Proofs of propositions and theorems

A.1. Proof of Proposition 1

The sum in (7) can be minimized for each component independently, see (5), and also separately for $k = 1, \dots, K$. Therefore consider w.l.o.g. $p = 1$ and $K = 1$.

Then the right side of (7) can be written as

$$\sum_{x_i \le \xi} (1-\theta)(\xi - x_i) + \sum_{x_i > \xi} \theta(x_i - \xi)$$

and the score function is

$$\sum_{x_i \le \xi} (1-\theta) - \sum_{x_i > \xi} \theta = \sum_{i=1}^{n} \mathbb{1}_{[x_i \le \xi]} - n\theta,$$

which is zero for $\theta = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{[x_i \le \xi]}$, so that $\xi = q_n(\theta)$ (any of the possible interval of quantiles).

A.2. Proof of Proposition 2

For all $\Theta = (\boldsymbol{\theta}, \tilde{\boldsymbol{\xi}}, \boldsymbol{\lambda}, C)$ from the parameter space:

$$V_{n,K}(\Theta, \tilde{\mathbf{x}}_n) = V_{n,K}(\boldsymbol{\theta}, \tilde{\boldsymbol{\xi}}^*, \mathbf{d}^t\boldsymbol{\lambda}, C, \tilde{\mathbf{x}}_n^*) + n\sum_{j=1}^{p} \log c_j,$$

where $\tilde{\boldsymbol{\xi}}^* = (\mathbf{c}^t\boldsymbol{\xi}_1, \dots, \mathbf{c}^t\boldsymbol{\xi}_K)$. As $n\sum_{j=1}^{p} \log c_j$ is a constant for a given dataset, minimizers of $V_{n,K}(\boldsymbol{\theta}, \tilde{\boldsymbol{\xi}}^*, \boldsymbol{\lambda}, C, \tilde{\mathbf{x}}_n^*)$ are obtained from minimizers of $V_{n,K}(\boldsymbol{\theta}, \tilde{\boldsymbol{\xi}}, \boldsymbol{\lambda}, C, \tilde{\mathbf{x}}_n)$ in the required way.

A.3. Proof of Proposition 3

The proof derives by taking the first derivative of $\sum_{i=1}^n \lambda \mathcal{Q}(x_i, \theta^*, \xi_{C(i)}) - n \log(\lambda\theta^*(1-\theta^*))$ with respect to θ^*, which gives:

$$\frac{\partial}{\partial\theta^*}\left(\lambda \sum_{i=1}^n \left\{\theta^* + (1-2\theta^*)\mathbb{1}_{[x_i<\xi_{C(i)}]}\right\} |x_i - \xi_{C(i)}| - n\log(\lambda\theta^*(1-\theta^*))\right) =$$
$$\lambda \sum_{i=1}^n (x_i - \xi_{C(i)}) - \frac{(1-2\theta^*)n}{\theta^*(1-\theta^*)}.$$

By equating the previous expression to zero and by multiplying by $-\theta^*(1-\theta^*)$ we get the quadratic solution for θ.

A.4. Proof of Proposition 4

Similarly to proposition 2, the proof derives by computing the score with respect to λ^*:

$$\frac{\partial}{\partial\lambda^*}\left(\lambda^* \sum_{i=1}^n \mathcal{Q}(x_i, \theta, \xi_{C(i)}) - n\log(\lambda^*\theta(1-\theta))\right) =$$
$$\sum_{i=1}^n \mathcal{Q}(x_i, \theta, \xi_{C(i)}) - \frac{n}{\lambda^*} = 0.$$

A.5. Proof of Theorem 1

The principle of the proof is to show that $T_{n,K}(\tilde{\mathbf{x}}_n)$ for large enough n has to lie in a compact set $\mathcal{C}$. In this compact set, by the uniform law of large numbers, $V_{n,K}(\boldsymbol{\theta}, \tilde{\boldsymbol{\xi}}, \boldsymbol{\lambda}, \tilde{\mathbf{x}}_n)$ will converge uniformly to $V_K(\boldsymbol{\theta}, \tilde{\boldsymbol{\xi}}, \boldsymbol{\lambda}, P)$, which in turn, together with continuity, will also enforce the minimizer to converge. $(\boldsymbol{\theta}, \boldsymbol{\lambda})$ optimizing $V_{n,K}$ are enforced to eventually lie in a compact set by the penalty term $-\log\lambda\theta(1-\theta)$. For $\tilde{\boldsymbol{\xi}}$, the argument is inductive, similar to what was done in [31]. It is first shown that at least one of the optimizing $\boldsymbol{\xi}_k$ must lie in a compact set, and then, assuming that this holds for $K-1$ clusters but not for K, the Kth cluster can be shown to have an asymptotically negligible additional contribution to S_K so that $S_K = S_{K-1}$ with contradiction against A2.

In order to show that of $T_{n,K}(\tilde{\mathbf{x}}_n) = (\boldsymbol{\theta}_n, \tilde{\boldsymbol{\xi}}_n, \boldsymbol{\lambda}_n)$ eventually $(\boldsymbol{\theta}_n, \boldsymbol{\lambda}_n)$ and at least one of the $\boldsymbol{\xi}_k$ must lie in a compact set, define $(\boldsymbol{\theta}_0, \tilde{\boldsymbol{\xi}}_0, \boldsymbol{\lambda}_0)$ as follows. For $j = 1, \dots, p,\ k = 1, \dots, K,\ \theta_{0j} = \frac{1}{2},\ \lambda_{0j} = 1,\ \xi_{0kj} = 0$. Then,

$$V_{n,K}(\boldsymbol{\theta}_0, \tilde{\boldsymbol{\xi}}_0, \boldsymbol{\lambda}_0, \tilde{\mathbf{x}}_n) = \frac{1}{n}\sum_{i=1}^n \sum_{j=1}^p \frac{1}{2}|x_{ij}| - p\log\frac{1}{4}.$$

The first part converges a.s. to $\frac{B_1}{2}$, where $B_1 = \int \sum_{j=1}^p |x_j| dP(\mathbf{x}) \leq \sqrt{p}B < \infty$ as defined in A1.

Suppose that (at least for a subsequence; apply this qualification also to further limits below, where necessary) $\lambda_{nj} \to 0$, $\theta_{nj} \to 0$, or $\theta_{nj} \to 1$. In this case $-\log \lambda_{nj}\theta_{nj}(1-\theta_{nj}) \to \infty$, and eventually

$$V_{n,K}(\boldsymbol{\theta}_n, \tilde{\boldsymbol{\xi}}_n, \boldsymbol{\lambda}_n, \tilde{\mathbf{x}}_n) > \frac{\sqrt{p}B}{2} - p\log\frac{1}{4},$$

for which reason $\lambda_{nj} \to 0$, $\theta_{nj} \to 0$, or $\theta_{nj} \to 1$ cannot happen when minimizing $V_{n,K}$. Therefore $\exists \theta^- > 0, \lambda^- > 0$ so that for large enough n, a.s., $\min(\theta_{n1}, \dots, \theta_{np}, 1-\theta_{n1}, \dots, 1-\theta_{np}) \geq \theta^-$, $\min(\lambda_{n1}, \dots, \lambda_{np}) \geq \lambda^-$.

Consider a compact set $M \subset \mathbb{R}^p$ with $0 \in M$, $P(M) > 0$, $|x_j| \leq m < \infty$ for $\mathbf{x} \in M$. Now suppose that there is no compact interval Ξ so that for large enough n, with suitable numbering of the clusters, at least for one $k \in \{1, \dots, K\}$: $\xi_{nk1}, \dots \xi_{nkp} \in \Xi$. Therefore, $\xi_n^- = \min_{k\in\{1,\dots,K\}} \max_{j\in\{1,\dots,p\}} |\xi_{nkj}| \to \infty$ and, for $\mathbf{x} \in M$:

$$\min_{k\in\{1,\dots,K\}} \sum_{j=1}^p \lambda_j \mathcal{Q}(x_j, \theta_j, \xi_{kj}) \geq \sum_{j=1}^p \lambda^- \theta^- (1-\theta^-)(\xi_n^- - m).$$

For large enough n this would make

$$V_{n,K}(\boldsymbol{\theta}_n, \tilde{\boldsymbol{\xi}}_n, \boldsymbol{\lambda}_n, \tilde{\mathbf{x}}_n) \geq P(M)p\lambda^-\theta^-(1-\theta^-)(\xi_n^- - m) - p\log\frac{1}{4} > \frac{\sqrt{p}B}{2} - p\log\frac{1}{4},$$

a.s., so $\xi_n^- \to \infty$ cannot happen.

Now assume (w.l.o.g.) that there is a compact set $\mathcal{C}$ so that for large enough n, a.s., $\boldsymbol{\xi}_{n1}, \dots, \boldsymbol{\xi}_{n(K-1)} \in \mathcal{C}$, but $\|\boldsymbol{\xi}_{nK}\| \to \infty$. Choose $\mathcal{C}$ large enough that it also contains all components of $\tilde{\boldsymbol{\xi}}_K$ (from the optimizer $T_K(P)$).

Consider the first term of $V_{n,K}(\boldsymbol{\theta}_n, \tilde{\boldsymbol{\xi}}_n, \boldsymbol{\lambda}_n, \tilde{\mathbf{x}}_n)$:

$$W^*_{n,K}(\boldsymbol{\theta}_n, \tilde{\boldsymbol{\xi}}_n, \boldsymbol{\lambda}_n, \tilde{\mathbf{x}}_n) = \frac{1}{n}\sum_{i=1}^n \min_{k\in\{1,\dots,K\}} \sum_{j=1}^p \lambda_j \mathcal{Q}(x_{ij}, \theta_j, \xi_{kj}).$$

Define, for any $\mathbf{x}$ and K,

$$C_{nK}(\mathbf{x}) = \operatorname*{arg\,min}_{k\in\{1,\dots,K\}} \sum_{j=1}^p \lambda_{nj} \mathcal{Q}(x_j, \theta_{nj}, \xi_{nkj}).$$

Then,

$$\begin{aligned}
W^*_{n,K}(\boldsymbol{\theta}_n, \tilde{\boldsymbol{\xi}}_n, \boldsymbol{\lambda}_n, \tilde{\mathbf{x}}_n) &= \frac{1}{n}\sum_{i=1}^n \sum_{j=1}^p \lambda_{nj} \mathcal{Q}(x_{ij}, \theta_{nj}, \xi_{nC_{nK}(\mathbf{x}_i)j}) \\
&= \frac{1}{n}\sum_{C_{nK}(\mathbf{x}_n)\neq K} \sum_{j=1}^p \lambda_{nj} \mathcal{Q}(x_{ij}, \theta_{nj}, \xi_{nC_{nK}(\mathbf{x}_i)j}) \\
&\quad + \frac{1}{n}\sum_{C_{nK}(\mathbf{x}_i)= K} \sum_{j=1}^p \lambda_{nj} \mathcal{Q}(x_{ij}, \theta_{nj}, \xi_{nKj}) \\
&\leq \frac{1}{n}\sum_{C_{nK}(\mathbf{x}_i)\neq K} \sum_{j=1}^p \lambda_{nj} \mathcal{Q}(x_{ij}, \theta_{nj}, \xi_{nC_{nK}(\mathbf{x}_i)j})
\end{aligned}$$

$$
\begin{aligned}
&\quad + \frac{1}{n}\sum_{C_{nK}(\mathbf{x}_i)=K} \min_{k\in\{1,\ldots,K-1\}} \sum_{j=1}^{p} \lambda_{nj}\,\mathcal{Q}(x_{ij},\theta_{nj},\xi_{nkj}).\\
&= \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{p} \lambda_{nj}\,\mathcal{Q}(x_{ij},\theta_{nj},\xi_{nC_{n(K-1)}(\mathbf{x}_i)j})\\
&= W^*_{n,K-1}(\boldsymbol{\theta}_n,\tilde{\boldsymbol{\xi}}^*_n,\boldsymbol{\lambda}_n,\tilde{\mathbf{x}}_n),
\end{aligned}
$$

where $\tilde{\boldsymbol{\xi}}^*_n = \{\boldsymbol{\xi}_{n1},\ldots,\boldsymbol{\xi}_{n(K-1)}\}$.

Consider any set $M = \{\|\mathbf{x}\| \leq m\}$ with $m < \infty$. Observe that, for large enough n, $M \cap \{\mathbf{x}:\ C_{nK}(\mathbf{x}) = K\} = \emptyset$. Furthermore,

$$
\frac{1}{n}\sum_{\mathbb{1}_{[C_{nK}(\mathbf{x}_i)=K]}} \min_{k\in\{1,\ldots,K-1\}} \sum_{j=1}^{p} \lambda_{nj}\,\mathcal{Q}(x_{ij},\theta_{nj},\xi_{nkj}) \leq \frac{1}{n}\sum_{\mathbb{1}_{[C_{nK}(\mathbf{x}_i)=K]}} \lambda^+ B_1.
$$

For large enough n this converges, a.s., to $P(\|\mathbf{x}\| > m)\lambda^+ B_1$, which can be made arbitrarily small by choosing m large enough.

Therefore, for arbitrarily small $\delta > 0$ and n large enough,

$$
\begin{aligned}
W^*_{n,K}(\boldsymbol{\theta}_n,\tilde{\boldsymbol{\xi}}_n,\boldsymbol{\lambda}_n,\tilde{\mathbf{x}}_n) &\leq W^*_{n,K-1}(\boldsymbol{\theta}_n,\tilde{\boldsymbol{\xi}}^*_n,\boldsymbol{\lambda}_n,\tilde{\mathbf{x}}_n)\\
&\leq \frac{1}{n}\sum_{C_{nK}(\mathbf{x}_i)\neq K}\sum_{j=1}^{p} \lambda_{nj}\,\mathcal{Q}(x_{ij},\theta_{nj},\xi_{nC_{nK}(\mathbf{x}_i)j}) + \delta\\
&\leq W^*_{n,K}(\boldsymbol{\theta}_n,\tilde{\boldsymbol{\xi}}_n,\boldsymbol{\lambda}_n,\tilde{\mathbf{x}}_n) + \delta. \qquad (14)
\end{aligned}
$$

In order to make use of this, a uniform convergence argument is needed. Recall $(\boldsymbol{\theta}_n,\tilde{\boldsymbol{\xi}}^*_n,\boldsymbol{\lambda}_n) \in \mathcal{C}$. According to [32], Example 19.8 (sometimes referred to as "uniform law of large numbers"), if $\mathcal{F} = \{f_\theta :\ \theta \in \Theta\}$ is a set of measurable functions with $\theta \mapsto f_\theta(x)$ continuous for all x, Θ compact, and $\exists F \geq |f_\theta| \forall \theta \in \Theta$, $\int F dP < \infty$, then

$$
\sup_{\theta\in\Theta}\left|\frac{1}{n}\sum_{i=1}^{n} f_\theta(\mathbf{x}_i) - \int f_\theta(\mathbf{x})dP(\mathbf{x})\right| \to 0 \text{ a.s.}
$$

For fixed x, $\mathcal{Q}(x,\theta,\xi) = \left\{\theta + (1-2\theta)\mathbb{1}_{[x<\xi]}\right\}|x-\xi|$ is continuous in (ξ,θ), because $\xi \to x \Rightarrow \mathcal{Q}(x,\theta,\xi) \to 0$ regardless of whether ξ comes from above or from below. Therefore, for fixed $\mathbf{x} \in \mathbb{R}^p$ and general K^*,

$$
U(\boldsymbol{\theta},\tilde{\boldsymbol{\xi}},\boldsymbol{\lambda},\mathbf{x}) = \min_{k\in\{1,\ldots,K^*\}} \sum_{j=1}^{p} \lambda_j\,\mathcal{Q}(x_j,\theta_j,\xi_{kj}) - \sum_{j=1}^{p} \log\left[\lambda_j(\theta_j(1-\theta_j))\right]
$$

is continuous as minimum of continuous functions.

$U(\boldsymbol{\theta},\boldsymbol{\xi},\boldsymbol{\lambda},\mathbf{x})$ can be bounded by a P-integrable function: Let ξ^+ an upper bound for the components $|\xi_{kj}|$ (assumed to be in a compact set here). Then,

$$
U(\boldsymbol{\theta},\boldsymbol{\xi},\boldsymbol{\lambda},\mathbf{x}) \quad \leq \quad U^+(\mathbf{x}) = \sum_{j=1}^{p} \lambda^+(|x_j| + \xi^+) - \sum_{j=1}^{p} \log\left[\frac{\lambda^-\theta^-}{2}\right],
$$

$$\int U^+(\mathbf{x})dP(\mathbf{x}) \quad < \quad \infty \text{ because of A1.}$$

Therefore,

$$\sup_{(\boldsymbol{\theta},\tilde{\boldsymbol{\xi}},\boldsymbol{\lambda})\in\mathcal{C}} |V_{n,K^*}(\boldsymbol{\theta},\tilde{\boldsymbol{\xi}},\boldsymbol{\lambda},\tilde{\mathbf{x}}_n) - V_{K^*}(\boldsymbol{\theta},\tilde{\boldsymbol{\xi}},\boldsymbol{\lambda},P)| \to 0 \text{ a.s.} \tag{15}$$

In particular,

$$\sup_{(\boldsymbol{\theta},\tilde{\boldsymbol{\xi}},\boldsymbol{\lambda})\in\mathcal{C}} |V_{n,K-1}(\boldsymbol{\theta},\tilde{\boldsymbol{\xi}},\boldsymbol{\lambda},\tilde{\mathbf{x}}_n) - V_{K-1}(\boldsymbol{\theta},\tilde{\boldsymbol{\xi}},\boldsymbol{\lambda},P)| \to 0.$$

Going back to (14), choose m large enough that $S_K < S_{K-1} - \delta$. By definition of the optimizers,

$$V_{n,K}(\boldsymbol{\theta}_n,\tilde{\boldsymbol{\xi}}_n,\boldsymbol{\lambda}_n,\tilde{\mathbf{x}}_n) \leq V_{n,K}(\boldsymbol{\theta}_K,\tilde{\boldsymbol{\xi}}_K,\boldsymbol{\lambda}_K,\tilde{\mathbf{x}}_n) \to S_K \text{ a.s.},$$

and, for large enough n, a.s.,

$$S_{K-1} \leq V_{n,K-1}(\boldsymbol{\theta}_n,\tilde{\boldsymbol{\xi}}^*_n,\boldsymbol{\lambda}_n,\tilde{\mathbf{x}}_n),$$

but also, eventually,

$$S_K \geq V_{n,K}(\boldsymbol{\theta}_n,\tilde{\boldsymbol{\xi}}_n,\boldsymbol{\lambda}_n,\tilde{\mathbf{x}}_n) \geq V_{n,K-1}(\boldsymbol{\theta}_n,\tilde{\boldsymbol{\xi}}^*_n,\boldsymbol{\lambda}_n,\tilde{\mathbf{x}}_n) - \delta \geq S_{K-1} - \delta,$$

contradicting $S_K < S_{K-1} - \delta$. This implies that $\boldsymbol{\xi}_{nk}$ is eventually also captured in a convex set $\mathcal{C}$.

(15) now ensures uniform convergence of $V_{n,K}$ to V_K over $\mathcal{C}$. The existence of an integrable envelope of U together with continuity of U imply the continuity of V_K as function of $(\boldsymbol{\theta},\tilde{\boldsymbol{\xi}},\boldsymbol{\lambda}) \in \mathcal{C}$. This and A2 imply $T_{n,K}(\tilde{\mathbf{x}}_n) \to T_K(P)$ a.s., because otherwise with probability > 0 a subsequence of $T_{n,K}(\tilde{\mathbf{x}}_n)$ can converge against $(\boldsymbol{\theta}^*,\tilde{\boldsymbol{\xi}}^*,\boldsymbol{\lambda}^*) \neq T_K(P)$ but $\in \mathcal{C}$ and with $V_K(\boldsymbol{\theta}^*,\tilde{\boldsymbol{\xi}}^*,\boldsymbol{\lambda}^*,P) = V_K(T_K(P))$, with contradiction to A2.

A.6. Proof of Theorem 2

The idea here is to show that if for arbitrarily large m a cluster in $T_K(P_m)$ can be found that has a nonempty intersection with at least two of the central sets $\{\|\mathbf{x} - \boldsymbol{\rho}_{mk}\| < \epsilon\}$, S_K would be larger than what could be achieved by putting all the cluster barycenters at the cluster centers, contradicting the optimality of $T_K(P_m)$.

Write $\gamma_m = \gamma_{T_K(P_m)}$. Define $(\boldsymbol{\theta}^*_m,\tilde{\boldsymbol{\xi}}^*_m,\boldsymbol{\lambda}^*_m)$ as follows. For $j = 1,\ldots,p,\ k = 1,\ldots,K,\ \theta^*_{mj} = \frac{1}{2},\ \lambda^*_{mj} = 1,\ \xi^*_{mkj} = \rho_{mkj}$. Then, because of A4,

$$V_K(\boldsymbol{\theta}^*_m,\tilde{\boldsymbol{\xi}}^*_m,\boldsymbol{\lambda}^*_m,P_m) \leq \int \sum_{j=1}^{p} \frac{1}{2}|x_j - \rho_{mkj}|dP_m(\mathbf{x}) - p\log\frac{1}{4} \leq \frac{B}{2} - p\log\frac{1}{4}.$$

Similar to the proof of Theorem 1, $\exists \theta^- > 0, \lambda^- > 0$ so that for large enough m: $\min(\theta_{m1},\dots,\theta_{mp}, 1-\theta_{m1},\dots,1-\theta_{mp}) \geq \theta^-$, $\min(\lambda_{m1},\dots,\lambda_{mp}) \geq \lambda^-$, because otherwise the penalty term $-\sum_{j=1}^p \log \lambda_{mj}\theta_{mj}(1-\theta_{mj})$ can be driven to infinity and $(\boldsymbol{\theta}_m^*, \tilde{\boldsymbol{\xi}}_m^*, \boldsymbol{\lambda}_m^*)$ would achieve a smaller and therefore better V_K.

For $k_1, k_2 \in \{1,\dots,K\}$ let $I_{mk_1k_2} = \{\|\mathbf{x} - \boldsymbol{\rho}_{mk_1}\| < \epsilon\} \cap \{\gamma_m(\mathbf{x}) = k_2\}$. Now assume that for at least a subsequence of $m \to \infty$, eventually,

$$I_{m11} \neq \emptyset \text{ and } I_{m21} \neq \emptyset,$$

where the cluster numbering has been chosen so that, w.l.o.g.,

$$\min[P_m(I_{m11}), P_m(I_{m21})] = \max_{(k_1,k_2,k_3)\in\mathcal{K}} \{\min[P_m(I_{mk_1k_3}), P_m(I_{mk_2k_3})]\}. \quad (16)$$

Suppose first that

$$\limsup_{m\to\infty} \min[P_m(I_{m11}), P_m(I_{m21})] = \tau > 0.$$

Let $b_m = \max(\|\boldsymbol{\xi}_{m1} - \boldsymbol{\rho}_{m1}\| - \epsilon, \|\boldsymbol{\xi}_{m1} - \boldsymbol{\rho}_{m2}\| - \epsilon)$. Because of A3, $\lim_{m\to\infty} b_m = \infty$. Obviously, for at least one $k \in \{1,2\}$ and all $\mathbf{x} \in \{\|\mathbf{x} - \boldsymbol{\rho}_{mk}\| < \epsilon\}$:

$$\sum_{j=1}^p |x_j - \xi_{m1j}| \geq \|\mathbf{x} - \boldsymbol{\xi}_{m1}\| \geq b_m.$$

Then

$$\begin{aligned} V_K(\boldsymbol{\theta}_m, \tilde{\boldsymbol{\xi}}_m, \boldsymbol{\lambda}_m, P_m) &\geq \int \mathbb{1}_{[\gamma_m(\mathbf{x})=1]} \sum_{j=1}^p \lambda_{mj} \mathcal{Q}(x_j, \theta_j, \xi_{m1j}) dP_m(\mathbf{x}) \\ &\quad - \sum_{j=1}^p \log \lambda_{mj}\theta_{mj}(1-\theta_{mj}) \\ &\geq \tau\lambda^-\theta^- b_m - p\log\frac{\lambda^+}{4} \to \infty, \end{aligned} \quad (17)$$

so this cannot happen for the minimizer of V_K.

Therefore assume w.l.o.g. $\limsup_m P_m(I_{m11}) = 0$. If also $\limsup_m P_m(I_{m21}) = 0$, for this subsequence, $\{\gamma_m(\mathbf{x}) = 1\}$ has no nonzero probability overlap with any mixture component's central set (all of which have probability $\geq \delta$ because of (13)), and there are $K-1$ clusters left to cover K central sets, in contradiction to (16). Therefore $\limsup_m P_m(I_{m21}) = \tau > 0$. This means that $\|\boldsymbol{\xi}_{m1} - \boldsymbol{\rho}_{m2}\|$ must be bounded, otherwise the argument leading to (17) applies again. Therefore $\|\boldsymbol{\xi}_{m1} - \boldsymbol{\rho}_{m1}\|$ is unbounded. There must be another cluster, w.l.o.g., $\{\gamma_m(\mathbf{x}) = 2\}$, so that $P_m(I_{m12}) = \tau^* > 0$. For $\mathbf{x} \in I_{m11} \neq \emptyset$:

$$\sum_{j=1}^p \lambda_{mj}(\theta_{mj} + (1-2\theta_{mj})\mathbb{1}_{[x_j<\xi_{m1j}]})|x_j - \xi_{m1j}|$$

$$\leq \sum_{j=1}^{p} \lambda_{mj}(\theta_{mj} + (1-2\theta_{mj})\mathbb{1}_{[x_j<\xi_{m2j}]})|x_j - \xi_{m2j}|.$$

This, together with $\|\boldsymbol{\xi}_{m1} - \boldsymbol{\rho}_{m1}\| \to \infty$ at least for a subsequence, enforces $\|\boldsymbol{\xi}_{m2} - \boldsymbol{\rho}_{m1}\| \to \infty$ as well. But then, as above, with $b_m^* = \|\boldsymbol{\xi}_{m2} - \boldsymbol{\rho}_{m1}\| - \epsilon$,

$$\begin{aligned} V_K(\boldsymbol{\theta}_m, \tilde{\boldsymbol{\xi}}_m, \boldsymbol{\lambda}_m, P_m) &\geq \int \mathbb{1}_{[\gamma_m(\mathbf{x})=2]} \sum_{j=1}^{p} \lambda_{mj} \mathcal{Q}(x_j, \theta_j, \xi_{m2j}) dP_m(\mathbf{x}) \\ &\quad - \sum_{j=1}^{p} \log \lambda_{mj}\theta_{mj}(1-\theta_{mj}) \\ &\geq \tau^* \lambda^- \theta^- b_m^* - p \log \frac{\lambda^+}{4} \to \infty, \end{aligned}$$

and again this is impossible for the minimizer of V_K.

Taken together, with any numbering of clusters,

$$I_{m11} \neq \emptyset \text{ and } I_{m21} \neq \emptyset$$

cannot happen together, so all the central sets $\{\|\mathbf{x} - \boldsymbol{\rho}_{mk}\| < \epsilon\},\ k \in \{1, \ldots, K\}$ must eventually be subsets of different clusters.

Appendix B: Detailed description of the simulation study

Simulation study on the performance of the quantile-based clustering algorithm. Five different scenarios are considered:

1. Symmetric Multivariate Student t-distributed variables $W \sim t_3$; data come from $K = 2, 3$ and 5 populations:
   - $K = 2$, two populations $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$: $X_j^{(1)} \sim W_j$ and $X_j^{(2)} \sim (W_j + 1)$, $j = 1, \ldots, p$.
   - $K = 3$, three populations $\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$ and $\mathbf{X}^{(3)}$: $X_j^{(1)} \sim W_j$, $X_j^{(2)} \sim (W_j + 1)$ and $X_j^{(3)} \sim (W_j - 1)$, $j = 1, \ldots, p$.
   - $K = 5$, five populations $\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$, $\mathbf{X}^{(3)}$, $\mathbf{X}^{(4)}$ and $\mathbf{X}^{(5)}$: $X_j^{(1)} \sim W_j$, $X_j^{(2)} \sim (W_j + 1)$, $X_j^{(3)} \sim (W_j + 2)$, $X_j^{(4)} \sim (W_j - 1)$ and $X_j^{(5)} \sim (W_j - 2)$, $j = 1, \ldots, p$.
2. Highly skewed data i.i.d. vectors $W \sim MVN(\mathbf{0}_p, \boldsymbol{\Sigma})$ transformed by using the exponential function; data come from $K = 2, 3$ and 5 populations:
   - $K = 2$, two populations $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$: $X_j^{(1)} \sim \exp(W_j)$ and $X_j^{(2)} \sim (\exp(W_j) + 0.6)$, $j = 1, \ldots, p$.
   - $K = 3$, three populations $\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$ and $\mathbf{X}^{(3)}$: $X_j^{(1)} \sim \exp(W_j)$, $X_j^{(2)} \sim (\exp(W_j) + 0.6)$ and $X_j^{(3)} \sim (\exp(W_j) - 0.6)$, $j = 1, \ldots, p$.

- $K = 5$, five populations $\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$, $\mathbf{X}^{(3)}$, $\mathbf{X}^{(4)}$ and $\mathbf{X}^{(5)}$: $X_j^{(1)} \sim \exp(W_j)$, $X_j^{(2)} \sim (\exp(W_j) + 0.6)$, $X_j^{(3)} \sim (\exp(W_j) + 1.2)$, $X_j^{(4)} \sim (\exp(W_j) - 0.6)$ and $X_j^{(5)} \sim (\exp(W_j) - 1.2)$, $j = 1, \ldots, p$.

3. Different distributions for the p variables. Firstly, $W \sim MVN(\mathbf{0}_p, \mathbf{\Sigma})$ and then split p into five balanced blocks to which different transformation were applied; data come from $K = 2, 3$ and 5 populations (subscripts in square brackets indicate variable block):
   - $K = 2$, two populations $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$:
     - $X_{j[1]}^{(1)} \sim W_j$ and $X_{j[1]}^{(2)} \sim W_j + 0.7$, $j = 1, \ldots, p$;
     - $X_{j[2]}^{(1)} \sim exp(W_j)$ and $X_{j[2]}^{(2)} \sim exp(W_j + 0.7)$, $j = 1, \ldots, p.$;
     - $X_{j[3]}^{(1)} \sim log(|W_j|)$ and $X_{j[3]}^{(2)} \sim log(|W_j + 0.7|)$, $j = 1, \ldots, p$;
     - $X_{j[4]}^{(1)} \sim W_j^2$ and $X_{j[4]}^{2} \sim (W_j + 0.7)^2$, $j = 1, \ldots, p$;
     - $X_{j[5]}^{(1)} \sim \sqrt{|W_j|}$ and $X_{j[5]}^{(2)} \sim \sqrt{|W_j + 0.7|}$, $j = 1, \ldots, p$.
   - $K = 3$, three populations $\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$ and $\mathbf{X}^{(3)}$:
     - $X_{j[1]}^{(1)} \sim W_j$, $X_{j[1]}^{(2)} \sim W_j{+}0.7$ and $X_{j[1]}^{(3)} \sim (W_j{+}1.4)$, $j = 1, \ldots, p$;
     - $X_{j[2]}^{(1)} \sim exp(W_j)$, $X_{j[2]}^{(2)} \sim exp(W_j + 0.7)$ and $X_{j[2]}^{(3)} \sim exp(W_j + 1.4)$, $j = 1, \ldots, p$;
     - $X_{j[3]}^{(1)} \sim log(|W_j|)$, $X_{j[3]}^{(2)} \sim log(|W_j + 0.7|)$ and $X_{j[3]}^{(3)} \sim log(|W_j + 1.4|)$, $j = 1, \ldots, p$;
     - $X_{j[4]}^{(1)} \sim W_j^2$, $X_{j[4]}^{2} \sim (W_j + 0.7)^2$ and $X_{j[4]}^{3} \sim (W_j + 1.4)^2$, $j = 1, \ldots, p$;
     - $X_{j[5]}^{(1)} \sim \sqrt{|W_j|}$, $X_{j[5]}^{(2)} \sim \sqrt{|W_j + 0.7|}$ and $X_{j[5]}^{(3)} \sim \sqrt{|W_j + 1.4|}$, $j = 1, \ldots, p$.
   - $K = 5$, five populations $\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$, $\mathbf{X}^{(3)}$, $\mathbf{X}^{(4)}$ and $\mathbf{X}^{(5)}$:
     - $X_{j[1]}^{(1)} \sim W_j$, $X_{j[1]}^{(2)} \sim W_j + 0.7$, $X_{j[1]}^{(3)} \sim (W_j + 1.4)$, $X_{j[1]}^{(4)} \sim (W_j + 2.1)$ and $X_{j[1]}^{(5)} \sim (W_j + 2.8)$, $j = 1, \ldots, p$;
     - $X_{j[2]}^{(1)} \sim exp(W_j)$, $X_{j[2]}^{(2)} \sim exp(W_j + 0.7)$, $X_{j[2]}^{(3)} \sim exp(W_j + 1.4)$, $X_{j[2]}^{(4)} \sim exp(W_j + 2.1)$ and $X_{j[2]}^{(5)} \sim exp(W_j + 2.8)$, $j = 1, \ldots, p$;
     - $X_{j[3]}^{(1)} \sim log(|W_j|)$, $X_{j[3]}^{(2)} \sim log(|W_j{+}0.7|)$, $X_{j[3]}^{(3)} \sim log(|W_j{+}1.4|)$, $X_{j[3]}^{(4)} \sim log(|W_j + 2.1|)$ and $X_{j[3]}^{(5)} \sim log(|W_j + 2.8|)$, $j = 1, \ldots, p$;
     - $X_{j[4]}^{(1)} \sim W_j^2$, $X_{j[4]}^{(2)} \sim (W_j + 0.7)^2$, $X_{j[4]}^{(3)} \sim (W_j + 1.4)^2$, $X_{j[4]}^{(4)} \sim (W_j + 2.1)^2$ and $X_{j[4]}^{(5)} \sim (W_j + 2.8)^2$, $j = 1, \ldots, p$;
     - $X_{j[5]}^{(1)} \sim \sqrt{|W_j|}$, $X_{j[5]}^{(2)} \sim \sqrt{|W_j + 0.7|}$, $X_{j[5]}^{(3)} \sim \sqrt{|W_j + 1.4|}$, $X_{j[5]}^{(4)} \sim \sqrt{|W_j + 2.1|}$ and $X_{j[5]}^{(5)} \sim \sqrt{|W_j + 2.8|}$, $j = 1, \ldots, p$.

4. Different distributional shapes and levels of skewness even for different classes within the same variable. Within each class, data were generated according to beta distributions with parameters a and b in the interval $(1, 10)$ randomly generated for each class within each variable. The absolute difference between the class expected values is bounded from above by 0.2:
   - $K = 2$, two populations $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$: $X_j^{(1)} \sim \text{Beta}(\alpha, \beta)$, where $\alpha, \beta \sim U(1, 10)$, and $X_j^{(2)} \sim \text{Beta}(\alpha, \beta)$, where $\alpha, \beta \sim U(1, 10)$, $j = 1, \dots, p$.
   - $K = 3$, three populations $\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$ and $\mathbf{X}^{(3)}$: $X_j^{(1)} \sim \text{Beta}(\alpha, \beta)$, where $\alpha, \beta \sim U(1, 10)$, $X_j^{(2)} \sim \text{Beta}(\alpha, \beta)$, where $\alpha, \beta \sim U(1, 10)$, and $X_j^{(3)} \sim \text{Beta}(\alpha, \beta)$, where $\alpha, \beta \sim U(1, 10)$, $j = 1, \dots, p$.
   - $K = 5$, five populations $\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$, $\mathbf{X}^{(3)}$, $\mathbf{X}^{(4)}$ and $\mathbf{X}^{(5)}$: $X_j^{(1)} \sim \text{Beta}(\alpha, \beta)$, where $\alpha, \beta \sim U(1, 10)$, $X_j^{(2)} \sim \text{Beta}(\alpha, \beta)$, where $\alpha, \beta \sim U(1, 10)$, $X_j^{(3)} \sim \text{Beta}(\alpha, \beta)$, where $\alpha, \beta \sim U(1, 10)$, $X_j^{(4)} \sim \text{Beta}(\alpha, \beta)$, where $\alpha, \beta \sim U(1, 10)$, and $X_j^{(5)} \sim \text{Beta}(\alpha, \beta)$, where $\alpha, \beta \sim U(1, 10)$, $j = 1, \dots, p$.
5. Different distributional shapes and levels of skewness even for different classes within each variable. Within each class, data are generated according to beta distributions with parameters a and b randomly chosen to be in the intervals: $(0, 1)$ and $(1, 5)$, or $(0, 1)$ and $(1, 5)$, $(1, 3)$ and $(5, 10)$, $(1, 3)$ and $(1, 3)$, for each class within each variable. The absolute difference between the class expected values is bounded from above by 0.1, and the so chosen interval guarantees a higher level of skewness for some variables.
   - $K = 2$, two populations $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$: $X_j^{(1)} \sim \text{Beta}(\alpha, \beta)$, where either:
     - $\alpha \sim U(0.1, 1)$ and $\beta \sim U(1, 10)$, or
     - $\alpha \sim U(1, 10)$ and $\beta \sim U(0.1, 1)$;

     and the same for $X_j^{(2)}$, $j = 1, \dots, p$.
   - $K = 3$, three populations $\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$ and $\mathbf{X}^{(3)}$: $X_j^{(1)} \sim \text{Beta}(\alpha, \beta)$, where either:
     - $\alpha \sim U(0.1, 1)$ and $\beta \sim U(1, 10)$, or
     - $\alpha \sim U(1, 10)$ and $\beta \sim U(0.1, 1)$;

     and the same for $X_j^{(2)}$; $X_j^{(3)} \sim Beta(\alpha, \beta)$, where $\alpha \sim U(1, 3)$ and $\beta \sim U(5, 10)$, $j = 1, \dots, p$.
   - $K = 5$, five populations $\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$, $\mathbf{X}^{(3)}$, $\mathbf{X}^{(4)}$ and $\mathbf{X}^{(5)}$: $X_j^{(1)} \sim \text{Beta}(\alpha, \beta)$, where either
     - $\alpha \sim U(0.1, 1)$ and $\beta \sim U(1, 5)$, or
     - $\alpha \sim U(1, 5)$ and $\beta \sim U(0.1, 1)$, or

- $\alpha \sim U(1,3)$ and $\beta \sim U(5,10)$, or
- $\alpha \sim U(5,10)$ and $\beta \sim U(1,3)$, or
- $\alpha \sim U(1,3)$ and $\beta \sim U(1,3)$;

and the same for $X_j^{(2)}$, $X_j^{(3)}$, $X_j^{(4)}$ an $X_j^{(5)}$ $j = 1, \dots, p$.

For each of the five scenarios and for each set of K populations, $K = 2, 3, 5$, we evaluated combinations of $p = \{50, 100, 500\}$, $n = \{50, 100, 500\}$, different percentages of relevant variables for grouping structure, i.e., 100%, 50% and 10%, independent or dependent variables (limited to scenarios 1-3), for a total of 648 different settings. The dependence structure among variables was modeled via the function `rcorrmatrix` from the R package `clusterGeneration`, so that the correlation matrices are uniformly distributed over the space of positive definite correlation matrices, with each correlation marginally distributed as $Beta(p/2, p/2)$ on $(-1, 1)$. The irrelevant noise variables were generated independently of each other from the base distribution of each scenario (for scenario 3 this means that all five base distributions were used in equal frequency, and for scenario 4 and 5 random parameters were used as within clusters for the informative variables).

For the case of imbalanced classes, data were generated from the five scenarios considering $K = 3$ and $n = 500$; the cluster sizes are set to $n_1 = 50$, $n_2 = 150$ and $n_3 = 300$.

The number of clusters is taken as known (and equal to the number of populations data are generated from) for every method. The clustering procedures that have been considered are the following:

- Common θ and Unscaled variables (CU) K-quantiles clustering algorithm;
- Variable-wise θ_j and Unscaled variables (VU) K-quantiles clustering algorithm;
- Common θ and Scaled variables (CS) K-quantiles clustering algorithm;
- Variable-wise θ_j and Scaled variables (VS) k-quantile clustering algorithm;
- Mixture of Gaussian distributions, estimated by the default options of function `Mclust` from the R package `mclust`;
- Mixture of skew-t distributions, estimated by the `EmSkew` function from the R package `EMMIXskew`, argument `distr` equal to `mst`, and initialised by the k-means clustering algorithm;
- Mixture of Factor Analyzers, estimated by the `fma` function from the `FactMixtAnalysis` R package, by fitting models with number of latent factors from 1 to 20;
- k-means clustering algorithm, run by the `kmeans` function from the `stats` R package, with five random starts;
- Partition Around Medoids, run by the default options of the `pam` function from the `cluster` R package;
- Agglomerative hierarchical clustering with average link, run by the `hclust` function, option `method='average'`, of the `stats` R package;
- Spectral clustering, estimated by the default options of function `specc` from the R package `kernlab`;

- Affinity Propagation clustering, estimated by the function `apclusterK`, with similarities computed as squared negative distances, from the R package `apcluster`.

For each setting 100 simulations were run. The average Adjusted Rand Index values and the corresponding standard errors are reported in the following tables, multiplied by 100; for each method and scenario the number of valid cases out of 100 runs is included as well.

B.1. Simulation results in the case of imbalanced classes

The performance of different clustering algorithms relative to the Common Unscaled K-quantiles clustering algorithm for imbalanced clusters (results on $K = 3$ and $n = 500$ only) is shown in Table 2. The labels along the horizontal axis refer to the different methods: CS, Common Scaled k-quantile; VU, Variable-wise Unscaled k-quantile; VS, Variable-wise Scaled K-quantile; *Mixt-N*, mixture of Normal distributions; *MFA*, mixture of factor analyzers; *KM*, k-means clustering; *PAM*, Partition Around Medoids algorithm; *h-avg*, hierarchical clustering with average linkage (UPGMA); *SpeCl*, spectral clustering; *AffPr*, affinity propagation. The five panels show the distribution of the Adjusted

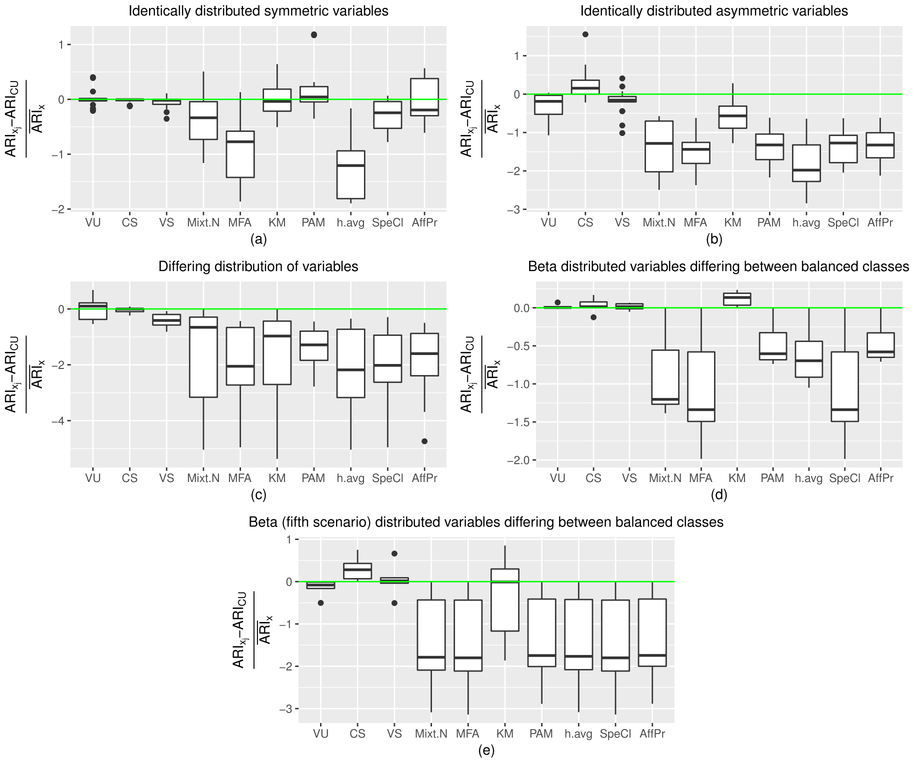

Table 2. *Average computing times in seconds.*

Method	$n = 50$, $p = 50$	$n = 50$, $p = 500$		$n = 500$, $p = 50$		$n = p = 500$		$n = 5000$, $p = 100$		$n = 50000$, $p = 100$	
CU	0.41 (0.09)	2.99	(0.08)	0.53	(0.03)	4.58	(0.16)	7.46	(1.35)	49.95	(0.00)
VU	0.95 (0.22)	6.95	(1.83)	1.96	(0.65)	28.43	(10.50)	34.96	(14.32)	60.45	(0.00)
CS	1.22 (0.22)	15.98	(2.58)	1.57	(0.15)	12.59	(0.87)	15.87	(4.73)	68.82	(0.00)
VS	3.14 (1.44)	5.78	(0.12)	4.40	(1.98)	13.82	(3.48)	56.32	(18.41)	120.38	(0.00)
Mixt-N	0.24 (0.23)	0.08	(0.01)	45.92	(24.07)	0.82	(0.02)	114.55	(64.09)	394.68	(0.00)
MFA	7.73 (2.13)	459.02	(101.37)	24.04	(3.31)	305.58	(36.11)	317.48	(34.72)	2832.56	(367.84)
k-means	0.11 (0.10)	0.05	(0.01)	0.05	(0.00)	0.55	(0.01)	1.19	(0.02)	12.05	(0.00)
PAM	0.00 (0.00)	0.00	(0.00)	0.02	(0.00)	0.10	(0.00)	2.99	(0.07)	486.63	(0.00)
h-avg	0.00 (0.00)	0.00	(0.00)	0.03	(0.00)	0.33	(0.00)	6.42	(0.15)	NaN	(NA)
SpeCl	0.05 (0.02)	0.24	(0.20)	3.04	(0.17)	2.90	(0.26)	2150.28	(153.00)	2248.24	(0.00)
AffPr	0.05 (0.02)	0.05	(0.01)	3.14	(0.36)	3.06	(0.37)	879.92	(98.54)	489.64	(0.00)

Rand Index for (a) identically distributed symmetric variables, (b) asymmetric variables, (c) different distributions of variables, (d) different distributions of classes within variables in balanced and unbalanced populations and (e) different (skew) distributions of classes within variables in balanced and unbalanced populations.

Appendix C: Computing time

Table 2 contains the average times (in seconds) – and the corresponding standard errors in brackets – required by each algorithm (excluding mixtures of ts, mixtures of skew-Normals and mixtures of skew-ts, as they could not always reach the convergence) to cluster the a single data set from each of the five scenarios, considering the cases with 50% of relevant variables, $K = 2$, both dependent and independent variables; all the procedures run on a Lenovo PC, Intel Core i5-6500 CPU, 3.20 GHz, 20 Gb of RAM. NaN/NA values mean that the method did not deliver a solution for at least one dataset.

References

[1] Banerjee, A., I. S. Dhillon, J. Ghosh, and S. Sra (2005). Clustering on the unit hypersphere using von Mises-Fisher distributions. *Journal of Machine Learning Research 6*(Sep), 1345–1382. MR2249858

[2] Ben-Israel, A. and C. Iyigun (2008). Probabilistic d-clustering. *Journal of Classification 25*(1), 5. MR2429670

[3] Bryant, P. and J. A. Williamson (1978). Asymptotic behaviour of classification maximum likelihood estimates. *Biometrika 65*(2), 273–281.

[4] Bubeck, S. and U. von Luxburg (2009). Nearest neighbor clustering: A baseline method for consistent clustering with arbitrary objective functions. *Journal of Machine Learning Research 10*, 657–698. MR2491753

[5] Clemencon, S. (2014). A statistical view of clustering performance through the theory of u-processes. *Journal of Multivariate Analysis 124*, 42–56. MR3147310

[6] Diaconis, P. (1988). *Grouped Representations in probability and statistics*, Volume 11. Hayward, CA: Institute of Mathematical Statistics Lecture Notes – Monograph Series. MR0964069

[7] Fligner, M. A. and J. S. Verducci (1986). Distance based ranking models. *Journal of the Royal Statistical Society. Series B (Methodological) 48*(3), 359–369. MR0876847

[8] Frey, B. J. and D. Dueck (2007). Clustering by passing messages between data points. *Science 315*(5814), 972–976. MR2292174

[9] Friedman, J. H. and J. J. Meulman (2004). Clustering objects on subsets of attributes (with discussion). *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 66*(4), 815–849. MR2102467

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

[11] Golub, T., D. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. Mesirov, H. Coller, M. Loh, J. Downing, M. Caligiuri, C. Bloomfield, and E. Lander (1999). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science 286*, 531–537.

[12] Halkidi, M., M. Vazirgiannis, and C. Hennig (2016). Method-independent indices for cluster validation and estimating the number of clusters. In C. Hennig, M. Meila, F. Murtagh, and R. Rocci (Eds.), *Handbook of Cluster Analysis*, Chapter 26, pp. 595–618. Chapman & Hall/CRC, Boca Raton, FL. MR3644729

[13] Hennig, C. (2015). What are the true clusters? *Pattern Recognition Letters 64*, 53–62. Philosophical Aspects of Pattern Recognition.

[14] Hennig, C. (2016). Clustering strategy and method selection. In C. Hennig, M. Meila, F. Murtagh, and R. Rocci (Eds.), *Handbook of Cluster Analysis*, Chapter 31, pp. 703–730. Chapman & Hall/CRC, Boca Raton FL. MR3644705

[15] Hennig, C. and C. Viroli (2016). Quantile-based classifiers. *Biometrika 103*(2), 435. MR3509897

[16] Hennig, C., C. Viroli, and L. Anderlucci (2019). Quantile-based clustering, arXiv:1806.10403 (supplement included from v2).

[17] Hubert, L. and P. Arabie (1985). Comparing partitions. *Journal of the Classification 2*, 193–218.

[18] Iyigun, C. (2010). Probabilistic distance clustering. In J. J. Cochran, L. A. Cox, P. Keskinocak, J. P. Kharoufeh, and J. C. Smith (Eds.), *Wiley Encyclopedia of Operations Research and Management Science*. John Wiley & Sons.

[19] Jain, A. K. (2010). Data clustering: 50 years beyond k-means. *Pattern Recognition Letters 31*(8), 651–666.

[20] Joe, H. (2006). Generating random correlation matrices based on partial correlations. *Journal of Multivariate Analysis 97*, 2177–2189. MR2301633

[21] Kaufman, L. and P. Rousseeuw (2005). *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley. MR1044997

[22] Kozubowski, T. J. and K. Podgorski (2000). A multivariate and asymmetric generalization of Laplace distribution. *Computational Statistics 15*(4), 531–540. MR1818032

[23] Leisch, F. (2016). Resampling methods for exploring clustering stability. In C. Hennig, M. Meila, F. Murtagh, and R. Rocci (Eds.), *Handbook of Cluster Analysis*, Chapter 28, pp. 637–652. Chapman & Hall/CRC, Boca Raton, FL. MR3644731

[24] Linder, T., G. Lugosi, and K. Zeger (1994). Rates of convergence in the source coding theorem, in empirical quantizer design, and in universal lossy source coding. *IEEE Transactions on Information Theory 40*(6), 1728–1740. MR1322387

[25] Lloyd, S. (1982). Least squares quantization in pcm. *IEEE Trans. Inf. Theor. 28*(2), 129–137. MR0651807

[26] Mallows, C. L. (1957). Non-null ranking models. I. *Biometrika 44*(1/2), 359–369. MR0087267

[27] McLachlan, G. J. and D. Peel (2000). *Finite Mixture Models*. Wiley. MR1789474

[28] Murphy, T. B. and D. Martin (2003). Mixtures of distance-based models for ranking data. *Computational Statistics & Data Analysis 41*, 645–655. MR1973732

[29] Ng, A. Y., M. I. Jordan, and Y. Weiss (2002). On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, pp. 849–856.

[30] Ostrovsky, R., Y. Rabani, L. J. Schulman, and C. Swamy (2006). The effectiveness of lloyd-type methods for the k-means problem. In *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science, Berkeley*. IEEE.

[31] Pollard, D. (1981). Strong consistency of k-means clustering. *Ann. Statist. 9*(1), 135–140. MR0600539

[32] van der Vaart, A. W. (1998). *Asymptotic Statistics*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press. MR1652247

[33] von Luxburg, U., M. Belkin, and O. Bousquet (2008, 04). Consistency of spectral clustering. *The Annals of Statistics 36*(2), 555–586. MR2396807