Journal of Machine Learning Research 23 (2022) 1-48 Submitted 1/21; Revised 2/22; Published 4/22

Regularized K-means Through Hard-Thresholding

Jakob Raymaekers J.RAYMAEKERS@MAASTRICHTUNIVERSITY.NL
Department of Quantitative Economics
Maastricht University
Maastricht, The Netherlands

Department of Mathematics
KU Leuven
Leuven, Belgium

Ruben H. Zamar RUBEN@STAT.UBC.CA
Department of Statistics
University of British Columbia
Earth Sciences Building, 2207 Main Mall
Vancouver, Canada.

Editor: Samory Kpotufe

Abstract

We study a framework for performing regularized K-means, based on direct penalization of the size of the cluster centers. Different penalization strategies are considered and compared in a theoretical analysis and an extensive Monte Carlo simulation study. Based on the results, we propose a new method called hard-threshold K-means (HTK-means), which uses an ℓ_0 penalty to induce sparsity. HTK-means is a fast and competitive sparse clustering method which is easily interpretable, as is illustrated on several real data examples. In this context, new graphical displays are presented and used to gain further insight into the data sets.

Keywords: clustering, penalized, variable selection, ℓ_0

1. Introduction

Clustering is one of the most commonly used unsupervised learning techniques. The goal of clustering is to partition the data into homogeneous groups. We focus on K-means, a method introduced by Steinhaus (1956) and popularized by MacQueen et al. (1967). We assume that we observe an $n \times p$ data matrix $\underline{\boldsymbol{x}}$, of which each row $\underline{x}_i$ is a p-dimensional observation ($i = 1, \ldots, n$). The K-means clustering algorithm tries to find a $K \times p$ matrix $\underline{\boldsymbol{\mu}}$ containing K cluster centers $\underline{\mu}_1, \ldots, \underline{\mu}_K$ in its rows that minimize the within-cluster sum of squares (WCSS) defined as

$$\text{WCSS} = \frac{1}{n} \sum_{i=1}^{n} \min_{k \in \{1,\ldots,K\}} ||\underline{x}_i - \underline{\mu}_k||_2^2. \tag{1}$$

Based on these centers, the data can be partitioned into K clusters by assigning each observation to the cluster corresponding to the nearest (in Euclidean distance) center. Despite

being over 50 years old, the K-means algorithm is still very popular and widely used in a variety of scientific fields. See Jain (2010) for a recent overview.

Whereas in classical K-means all p features are used to partition the data, it might be desirable to identify a subset of features that partitions the data particularly well. This feature selection may lead to a more interpretable partitioning of the data and more accurate recovery of the "true" clusters. We note that feature selection is not only relevant for scenarios where $p >> n$, but also when $p < n$. The former scenario, with (many) more variables than observations, is likely to include many uninformative variables which do not contribute to clustering the data and are better left out of the analysis. The latter scenario is typically easier to work with, but may also produce data sets with variables which do not contribute to (and rather difficult) the partitioning of the data. To illustrate this, we consider the classical example of Fisher's Iris data (Fisher, 1936), collected by Anderson (1935). The data consists of 150 iris flowers which are described by 4 variables characterizing the dimensions of their sepal and petal. The flowers can be subdivided in 50 samples of each of three types of iris: Iris setosa, versicolor, and virginica. Figure 1 shows a plot of the data in which the different iris types appear in different colors. From this plot it is clear that not all the variables separate the flowers equally well. This becomes more evident after we cluster this data set using the K-means algorithm on all possible subsets of variables. Table 1 shows the adjusted rand index (ARI) for each of these clusterings. The ARI measures the agreement between two partitions, in our case an estimated partition and the "true" partition. An ARI of 1 corresponds to a perfect clustering. Interestingly, K-means performs best (ARI $= 0.89$) when variable 4 alone or variables 3 and 4 are used for the clustering. This ARI value is substantially higher than 0.73, the ARI obtained when we use all 4 variables. This example illustrates that even for data sets with very few variables, feature selection can be very useful.

The previous paragraph illustrates our general objective: we aim to find subspaces spanned by a number of original variables which yield "interesting" partitions of the data. Note that by "interesting" we do not necessarily mean that the partition resembles the solution to the classical K-means problem on the data. This is in stark contrast with a large body a research, driven mainly by the computer science community, which aims to approximate this classical K-means solutions while clustering on a reduced data set. Boutsidis et al. (2009, 2014); Cohen et al. (2015); Moshkovitz et al. (2020) are prime examples of methodological and theoretical developments which build on the result that a constant-factor approximation to the optimal K-means solution can be achieved by using only $\mathcal{O}(K)$ of the initial features. A general overview of several of these approaches can be found in Alelyani et al. (2018). If the initial features need not be preserved, it has been shown that $\mathcal{O}(\log(K))$ variables suffice to obtain a reliable partitioning of the data (see Makarychev et al., 2019; Becchetti et al., 2019). This body of research allows for greatly speeding up K-means clustering, thereby opening the door for the application of the method to data sets many times the size of what was previously possible. However, it evidently assumes that the classical K-means solution is the gold standard as otherwise there would be no point in trying to approximate it.

The approach we adopt is more common in the statistics community and leads to very different algorithms, theoretical results and applications. In particular, the approaches we consider become interesting precisely when the classical K-means solution does not provide

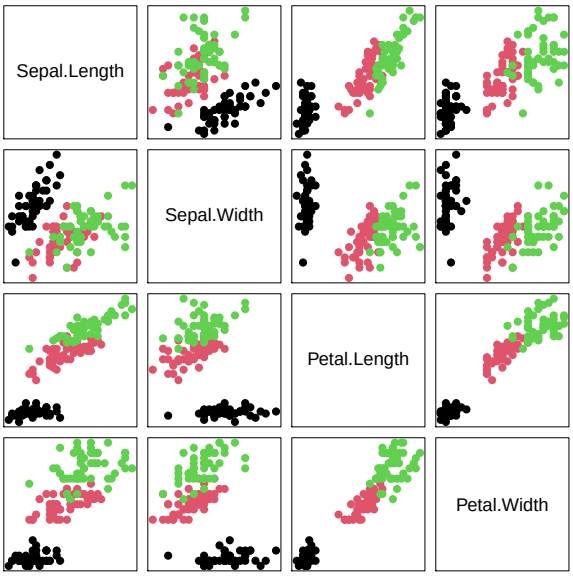

Figure 1: Pairs plot of the Iris data. Not all the variables separate the three different types of Iris equally well.

Used variables	ARI
(1)	0.38
(2)	0.15
(3)	0.85
(4)	**0.89**
(1, 2)	0.6
(1, 3)	0.7
(1, 4)	0.57
(2, 3)	0.8
(2, 4)	0.8
(3, 4)	**0.89**
(1, 2, 3)	0.7
(1, 2, 4)	0.61
(1, 3, 4)	0.73
(2, 3, 4)	0.87
(1, 2, 3, 4)	0.73

Table 1: ARI of the clustering of the Iris data using subsets of variables.

the most relevant partition of the data. As we will illustrate, the classical K-means solution can be pulled towards a random partition of the data by adding noise variables. One can imagine practical situations where many variables may not be relevant for the partitioning of the data, and in that case one should not fully trust the solution (or an approximation thereof) of the classical K-means.

When it comes to the K-means algorithm, an influential reference for the practice of combining feature selection with clustering is the paper by Witten and Tibshirani (2010) called *sparse K-means*. In this approach, the K-means objective function of Equation 1 is rewritten as a maximization problem, in which a vector of feature weights is introduced. An appropriate penalization strategy applied to the vector of feature weights induces sparsity in the variables and shrinkage in the estimated cluster centers. The new objective function can be optimized by iteratively maximizing it with respect to the cluster centers and the cluster memberships. Sun et al. (2012) proposed another regularized K-means approach based on direct penalization of the size of the cluster centers using an adaptive group-lasso penalty. This penalty also induces sparsity in the features and shrinkage in the estimated cluster centers.

Our main contributions are outlined in the following paragraph. We conduct a careful theoretical and empirical study of the regularized K-means approach and build a general regularization framework based on direct penalization of the size of the cluster centers, in which we consider lasso, ridge, group-lasso and ℓ_0-type penalties. We present a general iterative algorithm with a sparse starting value for the estimation of the cluster centers for each of these penalties, which gives insight into the effect of the different penalties on the

estimated cluster centers. Based on this study, we propose the use of the ℓ_0 penalty, which results in our proposal of the hard-thresholding K-means algorithm (HTK-means). Some theoretical advantages of the ℓ_0 penalty when compared with the other penalties under consideration are:

1. HTK-means can achieve consistency and variable selection consistency in a finite dimensional setting without shrinkage for a non-vanishing value of the regularization parameter (Corollary 8 of Section 3)

2. HTK-means can achieve $\sqrt{n/p}$-consistency and variable selection consistency in a high-dimensional regime where p is allowed to diverge at a rate slower than $n^{1/3}$ (Theorems 9 and 10 of Section 3)

3. the justified use of regularization parameter selection techniques based on the AIC and BIC criteria (Sections 4 and 5.5)

The complementary empirical studies indicate that:

1. the ℓ_0 penalty of HTK-means generally outperforms the lasso, ridge and group-lasso penalties in terms of cluster recovery and speed of computation (Section 5.2)

2. the absence of shrinkage in the estimated cluster centers allows for intuitive visualizations (Section 6)

The rest of the paper is organized as follows. Section 2 introduces the proposed framework for regularized K-means clustering and the penalties under consideration. It also presents the algorithm for the implementation of the clustering method for the different penalties. A theoretical analysis of the proposed method is presented in Section 3. Section 4 discusses different existing methods for selection of the regularization parameter. The results of four simulation studies are presented in Section 5. Finally, Section 6 introduces newly proposed graphical displays and illustrates the application of HTK-means on a number of real data examples. Finally, Section 7 concludes.

2. Methodology

In this section, we first introduce a general framework for regularized K-means clustering. We then describe the penalty functions considered throughout the paper. Next, we introduce an algorithm for computing the solution to the regularized K-means objective function and we end the section by briefly discussing the extension to model-based clustering.

2.1 Regularized K-means Clustering

We assume that we observe an $n \times p$ data matrix $\underline{\boldsymbol{x}}$, of which each row $\underline{x}_i$ is a p-dimensional observation ($i = 1, \ldots, n$). Suppose we want to cluster $\underline{x}_1, \ldots, \underline{x}_n$ in K clusters and that the variables are standardized so we have $\frac{1}{n}\sum_{i=1}^n x_{i,j} = 0$ and $\frac{1}{n}\sum_{i=1}^n x_{i,j}^2 = 1$ for every j. We consider the following general form of a regularized K-means objective function. Given the number of clusters K, let $\underline{\boldsymbol{\mu}} \in \mathbb{R}^{K\times p}$ be a $K \times p$ matrix of cluster centers and $C =$

$\{C_1, \ldots, C_K\}$ a collection of K disjoint sets of cluster indices satisfying $\bigcup_{k=1}^{K} C_k = \{1, \ldots, n\}$. We look for $\widehat{\underline{\boldsymbol{\mu}}}$ and $\widehat{C} = \{\widehat{C}_1, \ldots, \widehat{C}_K\}$ which minimize

$$\frac{1}{n} \sum_{k=1}^{K} \left\{ \sum_{i \in C_k} ||\underline{x}_i - \underline{\mu}_k||_2^2 \right\} + \lambda \mathcal{P}(\underline{\boldsymbol{\mu}}), \tag{2}$$

where $\lambda \geq 0$ is a tuning parameter and $\mathcal{P}(\underline{\boldsymbol{\mu}})$ is a penalty that depends on the cluster centers $\underline{\boldsymbol{\mu}}$. The first term in Equation 2 is the classical K-means objective (see Equation 1). Depending on how $\mathcal{P}$ is defined, different optimization problems arise. The important point in Equation 2 is that the penalization is based on $\underline{\boldsymbol{\mu}}$, which is not always the case in existing proposals (see, for example, Witten and Tibshirani, 2010).

The intuition for penalizing the size of the cluster centers stems from the fact that we expect that when a variable does not contribute to the partitioning of the data, its estimated cluster centers will be close to the overall mean of the variable (which is 0). The following proposition may add to the intuition. Consider the optimal population value of the classical K-means objective function of Equation 1: obj $= \int \min_{k \in \{1,\ldots,K\}} ||\underline{x} - \underline{\mu}_k||_2^2 Q(d\underline{x})$, where Q denotes the distribution of the m-variate random vector $\underline{X}$, where $m < p$. In this setting each cluster corresponds to a region in $\mathbb{R}^m$. Let $R_1, \ldots, R_K$ be these regions. Now suppose we add a variable to obtain the $(m+1)$-dimensional random vector $\underline{X}^* = (\underline{X}, Y)$ with distribution Q^*. We have the following proposition.

Proposition 1 *Denote with obj and obj* the optimal values of the K-means objective function on Q and Q^* respectively. Assume that the added variable is uninformative and independent from the original vector $\underline{X}$. More specifically, assume that:*

1. *$R_k^* = R_k \times \mathbb{R}$ for all $k = 1, \ldots, K$ (The value of Y doesn't affect the cluster assignment)*
2. *$\mathrm{d}Q^*(y|\underline{x}) = \mathrm{d}Q^*(y)$ (Y and $\underline{X}$ are independent)*

Then we have that:

1. *obj* $= 1 +$ obj*
2. *The optimal center of the added variable is 0 for every cluster $k = 1, \ldots, K$*

A proof can be found in Section A.1 of the Appendix. Of course, this is a simplified argument as it assumes that the cluster assignments do not change when adding the extra variable. The asymptotic assignments may in fact change if the added variable dominates the clustering structure but this is rather unlikely under the assumption that at least a few informative variables are already present and that the variables are standardized.

2.2 Penalty Functions

Throughout the paper we will consider several options for the penalty type which we name after their familiar counterparts from regularized regression:

$\boldsymbol{\ell_0}$:	$\mathcal{P}_0(\underline{\boldsymbol{\mu}}) = \sum_{j=1}^{p} \mathbf{1}_{\|\|\underline{\boldsymbol{\mu}}_{\cdot,j}\|\|_2 > 0}$
lasso:	$\mathcal{P}_1(\underline{\boldsymbol{\mu}}) = \sum_{j=1}^{p} \|\|\underline{\boldsymbol{\mu}}_{\cdot,j}\|\|_1$
ridge:	$\mathcal{P}_2(\underline{\boldsymbol{\mu}}) = \sum_{j=1}^{p} \|\|\underline{\boldsymbol{\mu}}_{\cdot,j}\|\|_2^2$
group-lasso:	$\mathcal{P}_3(\underline{\boldsymbol{\mu}}) = \sum_{j=1}^{p} \|\|\underline{\boldsymbol{\mu}}_{\cdot,j}\|\|_2$

where $\underline{\boldsymbol{\mu}}_{\cdot,j}$ denotes the j^{th} column of the matrix of cluster centers $\underline{\boldsymbol{\mu}}$. The penalty on $\underline{\boldsymbol{\mu}}$ balances the size of the cluster centers and their contribution to the objective function. Essentially, it implies that the cluster centers can be large only if they reduce the WCSS sufficiently. When a certain variable has only zero cluster centers, this variable becomes redundant in the clustering. An algorithm to optimize the objective of Equation 2 is derived in the next section. This algorithm also helps to better understand the effect of the different penalties on the clustering results.

2.3 Computation

In order to compute the cluster centers and indices resulting from the optimization in Equation 2, we use an adaptation of Lloyd's algorithm (Lloyd, 1982) for classical K-means:

Given an initial set of cluster centers:

1. Update the cluster indices $\hat{C}$ by minimizing Equation 1 with respect to the cluster indices while keeping the cluster centers fixed.
2. Update the cluster centers $\hat{\underline{\boldsymbol{\mu}}}$ by minimizing Equation 2 with respect to the cluster centers while keeping the cluster indices fixed.
3. Repeat 1. and 2. until convergence.

Like the classical K-means problem, the regularized version is NP-hard (Dasgupta, 2008; Aloise et al., 2009) and Lloyd's algorithm yields only locally optimal solutions. Therefore, the K-means algorithm is typically run using several starting values, after which the solution yielding the lowest objective function is retained. For the regularized K-means problem, one could take the starting centers as those resulting from the classical K-means algorithm. However, given that there is also a variable selection aspect to the clustering, these starting values may not perform well, especially when there are many uninformative variables. In order to incorporate the potential sparsity in the starting values, we use the following procedure:

(i) Cluster the data using classical K-means, obtaining K initial cluster centers $\underline{\mu}_1, \ldots, \underline{\mu}_K$.

(ii) Compute the Euclidean norm for each variable center: $d_j = ||\underline{\boldsymbol{\mu}}_{\cdot,j}||_2$ for $j = 1, \ldots, p$ and order them in descending order.

(iii) Execute K-means on the subset of variables corresponding to the $1, 2, 5, 10, 25$ and 50 % largest d_j.

(iv) Use the cluster indices of each of these K-means runs as an input for the regularized K-means version of Lloyd's algorithm, and choose the one yielding the lowest objective function.

The procedure outlined above allows the algorithm to start from several sparse solutions. The selection of the initial sparse solutions is based on the (Euclidean) norm of the variable centers, which is precisely what is penalized in regularized K-means clustering. The use of the proposed initialization slightly slows down our procedure, but does not increase its overall complexity. We have empirically evaluated this procedure. More precisely, we have compared it with a random initialization strategy, as well as starting from the classical K-means solution. These experiments showed that the proposed initialization strategy is distinctly superior to the alternatives. We refer to Section B of the Appendix for more details on these results.

Reconsidering the iterative algorithm, it is clear that in step 1, each point is assigned to the cluster corresponding with the nearest cluster center (in Euclidean distance), since keeping the cluster centers fixed also implies that the penalty term of Equation 2 is fixed. This is similar to the classical K-means objective function and the corresponding Lloyd's algorithm (Lloyd, 1982). Step 2 minimizes the objective function with respect to the cluster centers while keeping the cluster indices fixed. The penalty parameter is now dependent on the cluster centers $\underline{\boldsymbol{\mu}}$, and the resulting updated centers are therefore no longer necessarily equal to the cluster means. The following proposition presents the updating equations for the different penalties under consideration. The proof can be found in Section A.2 of the Appendix.

Proposition 2 *Suppose that we have an assignment of the elements into K clusters $C_1, \ldots, C_K$. Let $|C_k|$ be the number of elements in cluster k. Let $\underline{\boldsymbol{m}}$ be a $n \times K$ matrix with elements $m_{i,k} = 1$ if the i^{th} observation belongs to cluster k ($i = 1, \ldots, n$ and $k = 1, \ldots, K$), and $m_{i,k} = 0$ otherwise. Let $\underline{\boldsymbol{\mu}}^*$ be the corresponding $K \times p$ matrix of cluster means. Keeping this assignment fixed, minimizing the objective function in Equation 2 with respect to the $K \times p$ matrix of cluster centers $\underline{\boldsymbol{\mu}}$ gives the following expressions.*

$$\mathcal{P}(\underline{\boldsymbol{\mu}}) = \mathcal{P}_0(\underline{\boldsymbol{\mu}}) \text{ yields} \qquad \underline{\boldsymbol{\mu}}_{k,j} = \begin{cases} \underline{\boldsymbol{\mu}}^*_{k,j} \text{ if } ||\underline{\boldsymbol{x}}_{\cdot,j}||_2^2 > ||\underline{\boldsymbol{x}}_{\cdot,j} - \underline{\boldsymbol{m}}\, \underline{\boldsymbol{\mu}}^*_{\cdot,j}||_2^2 + n\lambda \\ 0 \text{ else} \end{cases}$$

$$\mathcal{P}(\underline{\boldsymbol{\mu}}) = \mathcal{P}_1(\underline{\boldsymbol{\mu}}) \text{ yields} \qquad \underline{\boldsymbol{\mu}}_{k,j} = \max\left(0, 1 - \frac{n\lambda}{2|C_k|\left|\underline{\boldsymbol{\mu}}^*_{k,j}\right|}\right) \underline{\boldsymbol{\mu}}^*_{k,j}$$

$$\mathcal{P}(\underline{\boldsymbol{\mu}}) = \mathcal{P}_2(\underline{\boldsymbol{\mu}}) \text{ yields} \qquad \underline{\boldsymbol{\mu}}_{k,j} = \frac{1}{1 + \frac{n\lambda}{|C_k|}} \underline{\boldsymbol{\mu}}^*_{k,j}$$

$$\mathcal{P}(\underline{\boldsymbol{\mu}}) = \mathcal{P}_3(\underline{\boldsymbol{\mu}}) \text{ yields} \qquad \underline{\boldsymbol{\mu}}_{k,j} = \frac{1}{1 + \frac{n\lambda}{(2|C_k|||\underline{\boldsymbol{\mu}}_{\cdot,j}||_2)}} \underline{\boldsymbol{\mu}}^*_{k,j} \text{ if } \underline{\boldsymbol{\mu}}_{\cdot,j} \neq \mathbf{0}$$

Remark 3 (Penalty effects) *These updating equations provide additional insight into the effect of the different penalty types. $\mathcal{P}_0$ leads to hard thresholding. It is the literal translation of "include a variable in the clustering if it sufficiently reduces the WCSS". If the variable is included (that is, the corresponding vector of cluster centers is non-zero), the cluster centers are given by the cluster means of each cluster as in classical K-means. $\mathcal{P}_1$ is a lasso-type penalty. It sets some of the coefficients to exactly zero, and others are shrunk towards 0. The updating equation uses a soft-thresholding operator, and bears strong resemblance to the solution of lasso regression with orthonormal covariates. $\mathcal{P}_2$ is a ridge-type penalty and shrinks all the cluster centers towards zero without setting them to zero exactly. Like in regression, it does not induce any sparsity and the shrinkage is proportional to $1/\lambda$. $\mathcal{P}_3$ is the only penalty which does not have an explicit updating equation, as the right hand side contains the euclidean norm of the vector of centers $||\underline{\boldsymbol{\mu}}_{\cdot,j}||_2$. The solution is thus implicit and can be found through an iterative algorithm. This penalty induces sparsity in the cluster centers, while shrinking in a ridge-type fashion each center that is not shrunk to zero.*

Remark 4 (Size-dependent penalties) *Note that the cluster sizes play a role in the update steps of penalties $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}_3$. These seem to be somewhat unnatural and can be removed by including penalties which depend linearly on the size of the clusters. For example, if we replace λ by $\lambda_i = \lambda\frac{|C_i|}{n}$, we would obtain more elegant expressions as both n and $|C_i|$ would disappear in the updating equations. For model-based clustering, this was done by Bhattacharya and McNicholas (2014). We did not pursue this path any further because it makes the optimization slightly slower and did not yield substantial improvements for $\mathcal{P}_1$ and $\mathcal{P}_2$ in the simulation study and in particular for $\mathcal{P}_3$ it is not immediately clear how this should be implemented without a substantial increase in computational cost. We suspect it may have some potential when the true cluster sizes are very unbalanced. Note that this normalization of the penalty parameter does not affect the $\mathcal{P}_0$ penalty.*

Remark 5 (Adaptive penalties) *In addition to making the penalties dependent on the cluster sizes, there is the option of making them adaptive. This idea was introduced by Zou (2006) in the context of lasso regression to obtain both $\sqrt{n}$-consistency as well as consistent variable selection. It was also used by Sun et al. (2012) in their version of regularized K-means clustering. It can be implemented by replacing λ in the updating equations of Proposition 2 by $\lambda_j = \frac{\lambda}{||\underline{\boldsymbol{\mu}}^*_{\cdot,j}||_2}$.*

2.4 A Note on Model-Based Clustering

Model based clustering is sometimes used as a flexible alternative to K-means. There exist a number of proposals for sparse and regularized model-based clustering including Friedman and Meulman (2004), Raftery and Dean (2006), Pan and Shen (2007), Wang and Zhu (2008) and Maugis et al. (2009). Casting model-based clustering in our regularization strategy would lead to a framework encompassing the lasso-based approaches of Pan and Shen (2007) and Sun et al. (2012). Suppose we would model the data distribution f as a mixture of K distribution functions, $f(\underline{x}) = \sum_{i=1}^{K} \pi_k f_k(\underline{x}; \underline{\theta}_k)$, where $\underline{\theta}_k$ contains all parameters needed to characterize f_k. For the ℓ_0 penalty, we then obtain the regularized

log-likelihood

$$\frac{1}{n}\sum_{i=1}^{n}\left\{\log\left(\sum_{k=1}^{K}\pi_k f_k(\underline{x}_i;\underline{\theta}_k)\right)\right\}-\lambda\sum_{j=1}^{p}\mathbf{1}_{||\underline{\boldsymbol{\mu}}_{\cdot,j}||_2>0}, \tag{3}$$

which can be optimized through the EM algorithm. The reason for not including this approach in more detail is twofold. First, several of the aforementioned methods were shown to be outperformed by the sparse K-means algorithm of Witten and Tibshirani (2010) (in the same paper), which we include in our simulation study. Secondly, for high-dimensional data, modeling the clusters using only a center already yields quite a lot of parameters ($\mathcal{O}(Kp)$) that need to be estimated. If more complex models are used, such as Gaussian clusters with arbitrary covariance matrices, we obtain $\mathcal{O}(Kp^2)$ parameters, which quickly becomes prohibitive in terms of computation time. Additionally, the sample covariance matrix may no longer be invertible (which is required for the calculation of the likelihood), and so one has to resort to other means. One possible assumption to simplify the objective (see, for example, Pan and Shen, 2007) is to assume that all clusters follow a multivariate normal distribution with the same diagonal covariance matrix given by $\text{diag}(\sigma_1^2,\ldots,\sigma_p^2)$. While this does make the calculations feasible in the high dimensional setting, it makes the extension from K-means only minor, as now the only difference is the ability to model elliptical clusters with axes parallel to the coordinate axes. The EM-algorithm for optimizing the objective function in Equation 3 under this assumption is given in Section C of the Appendix.

3. Consistency and Variable Selection

In order to investigate regularized K-means from a theoretical perspective, we first consider the asymptotic formulation of the objective function in Equation 2. Let Q be a probability measure on $\mathbb{R}^p$ and A a finite subset of possible cluster centers in $\mathbb{R}^p$. Let $\lambda \geq 0$ be fixed. We define the following objective function:

$$W(A,Q) \coloneqq \int \min_{a\in A}||x-a||_2^2 Q(dx) + \lambda\,\mathcal{P}(A)$$

where $\mathcal{P}(A)$ denotes the penalization on the cluster centers in A.

Let $p>0$ and $K>0$ be two fixed integers. Let X be a p-variate random variable with distribution F. Assume that

(a) $\int ||x||_2^2 F(dx) < \infty$

(b) For each $k=1,\ldots,K$, there is a unique set $\bar{A}(k)$ for which

$$W(\bar{A}(k),F) = \inf\{W(A,F)|A \text{ contains at most } k \text{ points}\}$$

(c) $\mathcal{P}$ is one of $\mathcal{P}_0$, $\mathcal{P}_1$, $\mathcal{P}_2$ or $\mathcal{P}_3$.

These assumptions are identical to the assumptions needed for the consistency of classical K-means, see Pollard (1981). The following theorem establishes the (strong) consistency of

regularized K-means in terms of the Hausdorff distance. For two finite sets A and B, the Hausdorff distance between them is given by

$$d_H(A,B) = \max\left\{\max_{a\in A}\min_{b\in B}||a-b||_2, \max_{b\in B}\min_{a\in A}||a-b||_2\right\}.$$

A proof of the theorem can be found in Section A.3 of the Appendix.

Theorem 6 *Let $\underline{x}_1 \ldots, \underline{x}_n$ be a random sample from F with empirical distribution function F_n and let A_n be the optimal set of at most K cluster centers for the sample. Under assumptions (a), (b) and (c) above, we have that:*

1. $W(A_n, F_n) \xrightarrow{a.s.} W(\bar{A}(k), F)$
2. $d_H(A_n, \bar{A}(k)) \xrightarrow{a.s.} 0$.

In addition to consistency, it is important that the penalization is guaranteed to work as intended, that is, that we perform variable selection. Suppose w.l.o.g. that the last $p-p_0+1$ variables are noise variables, in the sense that they are independent of all other variables and of any clustering structure. The optimal asymptotic solution to the classical as well as the regularized K-means problem is then a set of centers $\bar{A}$ for which $\bar{A}_{\cdot,j} = 0$ for all $j = p_0, \ldots, p$, that is, the last $p - p_0 + 1$ centers are zero. We would then like to have $P(\hat{A}_{\cdot,j} = 0) \to 1$ for each $j = p_0, \ldots, p$. The strong consistency implies that the true zero-centers converge in probability to zero. However, this doesn't guarantee that the probability that they are equal to zero converges to 1, which is what we need to guarantee variable selection. The following theorem shows that this does indeed happen for all but the $\mathcal{P}_2$ penalty, provided $\lambda > 0$. The proof can be found in Section A.4 of the Appendix.

Theorem 7 *Under conditions (a), (b) and (c) above, and assuming that $\lambda > 0$, we have that $P(\hat{A}_{\cdot,j} = 0) \to 1$ for all $j = p_0, \ldots, p$ for $\mathcal{P}_0$, $\mathcal{P}_1$ and $\mathcal{P}_3$.*

Ideally, one may wish that the non-zero centers are estimated as if the regular K-means algorithm would be executed on the "selected" variables, that is, those variables with non-zero cluster centers. The following corollary follows naturally from Theorems 6 and 7 as well as Proposition 2 and states that this can only happen for the $\mathcal{P}_0$ penalty, provided the value of λ is chosen correctly.

Corollary 8 *Under conditions (a), (b) and (c) above, and using penalty $\mathcal{P}_0$, there exists a $\lambda > 0$, such that we have that $P(\hat{A}_{\cdot,j} = 0) \to 1$ for all $j = p_0, \ldots, p$ and $\hat{A}_{\cdot,j} \xrightarrow{P} A^*_{\cdot,j}$ for all $j = 1, \ldots, p_0 - 1$. Here A^* is the set of optimal cluster centers obtained by dropping the penalty term from the objective function (that is, classical K-means on the first $p_0 - 1$ variables).*

We now consider the asymptotic properties when the dimension p is allowed to diverge together with n. More precisely, we assume that $n \to \infty$, $p_n \to \infty$ with $p_n = o(n)$ and $\lambda_n \to 0$. We will drop the subscript n in the following results. In this regime, the goal is to asymptotically approximate the solution to the "classical" K-means objective. Denote with $\bar{A}$ the minimizer of the objective $\int \min_{a\in A}||x-a||_2^2 F(dx)$. We then obtain the following result

Theorem 9 *Assume the assumptions (i) - (vi) in the Section A.5 of the Appendix hold. Then if* $\lambda p \to 0$ *as* $n \to \infty$*, then* $A_n \xrightarrow{a.s.} \bar{A}$ *and* $||A_n - \bar{A}|| = \mathcal{O}\left(p^{1/2}n^{-1/2}\right)$*, with the additional condition for* $\mathcal{P}_1, \mathcal{P}_2$ *and* $\mathcal{P}_3$ *that* $\lambda = \mathcal{O}(n^{-1/2})$*.*

Note that the theorem above implies that the conditions for $\sqrt{\frac{n}{p}}$-consistency are less stringent for the HT penalty $\mathcal{P}_0$ than for the other penalties under consideration, as the former does not require $\lambda = \mathcal{O}(n^{-1/2})$. This is in line with results from regression (see, for example, Fan and Peng, 2004), where it is known that the same requirement of $\lambda = \mathcal{O}(n^{-1/2})$ is needed in order to obtain $\sqrt{\frac{n}{p}}$-consistency for L_q-penalties with $q \geq 1$.

We now turn to variable selection, for which we obtain the following result.

Theorem 10 *Assume the assumptions (i) - (vii) in the Appendix hold. Then if* $n^{-1}\lambda^{-2}p \to 0$ *as* $n \to \infty$*, then we have that* $P(\hat{A}_{\cdot,j} = 0) \to 1$ *for all* $j = p_0, \ldots, p$ *for* $\mathcal{P} \in \{\mathcal{P}_0, \mathcal{P}_1, \mathcal{P}_3\}$*.*

We again have a parallel with regression here, as the condition $n^{-1}\lambda^{-2}p \to 0$ is exactly the same as the one used in, for example, Fan and Peng (2004); Kim et al. (2008); Cho and Qu (2013) to obtain model selection consistency. Note that, just like for regression, we cannot jointly achieve $\sqrt{\frac{n}{p}}$-consistency and model selection consistency for non-adaptive L_q-penalties with $q \geq 1$. In contrast, we can achieve this using $\mathcal{P}_0$ provided that $p = o(n^{1/3})$, which is the same rate as the one achieved in Huber (1973) and discussed in Fan and Peng (2004) for regression.

The theoretical results above indicate that there is merit in using the ℓ_0 penalty. Additionally, this penalty has the benefit of not shrinking the cluster centers of the variables used in the clustering. This allows for easier interpretation of the cluster centers and justifies the use of relatively simple techniques for selecting λ, as we will see in the next section. The ℓ_0 penalty leads to the hard-thresholding algorithm for K-means which we call *HTK-means*. This is the main proposal of our paper. The results of the synthetic data experiments in Section 5 further strengthen our case in favor of the use of the HTK-means algorithm.

4. Selection of λ

The selection of the regularization parameter λ is not an easy task. The main reason is that many techniques and heuristics for tuning hyperparameters in cluster analysis rely on some kind of distance between observations. In the setting of regularization, one could calculate distances on the selected variables, or on all of the variables. In the former case, the distances are not comparable over different values of the regularization parameter. In the latter case, the values of the distances can be dominated by uninformative variables which are not used for the clustering. Therefore, relying on distances between observations may be inappropriate in the setting of regularized clustering. This makes straight forward adoption of popular methods such as the gap statistic (Tibshirani et al., 2001) or the silhouette coefficient (Rousseeuw, 1987) impossible.

We focus on the selection of λ for the ℓ_0 penalty and the corresponding HTK-means algorithm, which has the advantage of not shrinking the cluster centers of the selected variables. We first discuss a number of approaches for selecting λ. These approaches are then compared empirically in Section 5.3.
We first consider the rather simple AIC and BIC criteria (Ramsey et al., 2008) given by

$$\text{AIC} = \text{WCSS} + 2kp$$
$$\text{BIC} = \text{WCSS} + k\ln(n)p,$$

where WCSS denotes the within-cluster sums of squares calculated on all variables. Possible improvements on these rather naive criteria could be in the form of more accurate estimation of the degrees of freedom in the BIC criteria (see, for example, Hofmeyr, 2020).

In addition to the AIC and BIC criteria, we consider methods based on clustering stability rather than coherence-type measures. The idea in this approach is that a good clustering method should yield "stable" clusters, in the sense that it should yield similar cluster assignments when estimated on different samples from the same population. Following Ben-David et al. (2006) and Wang (2010), we can define

Definition 11 (Clustering Distance) *The distance between any two clusterings ψ_1 and ψ_2 is defined as*

$$d(\psi_1, \psi_2) = \Pr[\mathbf{1}_{\{\psi_1(X)=\psi_1(Y)\}} + \mathbf{1}_{\{\psi_2(X)=\psi_2(Y)\}}],$$

where $\mathbf{1}_{\{.\}}$ denotes the indicator function and X and Y are independently sampled from F.

Based on the clustering distance above, we can define clustering instability as in Wang (2010):

Definition 12 (Clustering Instability) *The clustering instability of a clustering algorithm $\boldsymbol{\psi}$ is*

$$s(\boldsymbol{\psi}; \lambda) = \mathbb{E}[d\{\boldsymbol{\psi}(X_1; \lambda), \boldsymbol{\psi}(X_2; \lambda)\}],$$

where the expectation is taken with respect to X_1 and X_2 which are independent samples of size n from F.

Several ways to estimate $s(\boldsymbol{\psi}; \lambda)$ have been proposed. Wang (2010) propose to repeatedly split the data into 3 parts, 2 training sets and one validation set. The clustering method is trained on each of the training sets, and the stability is calculated as the expectation of their agreement in clustering the validation set. A potential problem with this approach is that the resulting data sets of sizes $n/3$ may be too small. We consider three alternatives:

1. **stab1:** Fang and Wang (2012) propose instead to use bootstrap samples by taking for each replication, 2 bootstrap data sets of size n, after which the original data is used as validation set.

2. **stab2:** Ben-Hur et al. (2001); Haslbeck and Wulff (2020) take the intersection of unique samples of the 2 bootstrapped training data sets as validation set.

3. **stab3:** Sun et al. (2012) use a variation where a third bootstrap data set is taken as validation set.

In Section 5.3 we empirically evaluate the discussed approaches for selecting λ in terms of clustering performance, variable selection performance and computational cost.

5. Synthetic Data Experiments

In this section we empirically evaluate the proposed methodology in four separate simulation studies. The first focuses on comparing the different penalty types within the regularized K-means framework of Equation 2. The second compares different techniques for selecting the regularization parameter λ. The third simulation study compares the proposed HTK-means method with other existing approaches to regularized clustering. The fourth and final simulation study investigates whether AIC and the proposed sparse starting values can improve the performance of other regularized clustering algorithms.

We start by describing the synthetic data generation process used throughout the simulation studies.

5.1 Synthetic Data Generation

The data generation process starts from the approach of Sun et al. (2012) and extends this in several directions. We generate data sets of $n \in \{80, 800\}$ observations $\underline{x}_1, \ldots, \underline{x}_n$ in $p \in \{50, 200, 500, 1000\}$ dimensions. First the true cluster assignment vector $\underline{y} = y_1, \ldots, y_n$ is sampled from $\{1, \ldots, K\}$ where $K \in \{2, 4, 8\}$. For each observation $\underline{x}_i$, only the first 50 variables are informative. They are sampled from $\mathcal{N}\left(\underline{\mu}\left(y_i\right), I_{50}\right)$, where $\underline{\mu}\left(y_i\right)$ is given by

$$
\begin{aligned}
\underline{\mu}_{K=2}(\boldsymbol{y}_i) &= \gamma \mathbf{1}_{50}\mathbf{1}_{\boldsymbol{y}_i=1} - \gamma \mathbf{1}_{50}\mathbf{1}_{\boldsymbol{y}_i=2} \\
\underline{\mu}_{K=4}(\boldsymbol{y}_i) &= (-\gamma \mathbf{1}_{25}, \gamma \mathbf{1}_{25})\mathbf{1}_{\boldsymbol{y}_i=1} + \gamma \mathbf{1}_{50}\mathbf{1}_{\boldsymbol{y}_i=2} + (\gamma \mathbf{1}_{25}, -\gamma \mathbf{1}_{25})\mathbf{1}_{\boldsymbol{y}_i=3} - \gamma \mathbf{1}_{50}\mathbf{1}_{\boldsymbol{y}_i=4} \\
\underline{\mu}_{K=8}(\boldsymbol{y}_i) &= (\gamma \mathbf{1}_{17}, \gamma \mathbf{1}_{17}, \gamma \mathbf{1}_{16})\mathbf{1}_{\boldsymbol{y}_i=1} + (\gamma \mathbf{1}_{17}, -\gamma \mathbf{1}_{17}, \gamma \mathbf{1}_{16})\mathbf{1}_{\boldsymbol{y}_i=2} \\
&\quad + (\gamma \mathbf{1}_{17}, \gamma \mathbf{1}_{17}, -\gamma \mathbf{1}_{16})\mathbf{1}_{\boldsymbol{y}_i=3} + (\gamma \mathbf{1}_{17}, -\gamma \mathbf{1}_{17}, -\gamma \mathbf{1}_{16})\mathbf{1}_{\boldsymbol{y}_i=4} \\
&\quad + (-\gamma \mathbf{1}_{17}, \gamma \mathbf{1}_{17}, \gamma \mathbf{1}_{16})\mathbf{1}_{\boldsymbol{y}_i=5} + (-\gamma \mathbf{1}_{17}, -\gamma \mathbf{1}_{17}, \gamma \mathbf{1}_{16})\mathbf{1}_{\boldsymbol{y}_i=6} \\
&\quad + (-\gamma \mathbf{1}_{17}, \gamma \mathbf{1}_{17}, -\gamma \mathbf{1}_{16})\mathbf{1}_{\boldsymbol{y}_i=7} + (-\gamma \mathbf{1}_{17}, -\gamma \mathbf{1}_{17}, -\gamma \mathbf{1}_{16})\mathbf{1}_{\boldsymbol{y}_i=8}
\end{aligned}
$$

Large values of γ produce well-separated clusters and small values of γ produce clusters with a lot of overlap. We vary the value of γ in $\{0.4, 0.5, 0.6, 0.7, 0.8\}$. The remaining $p - 50$ are noise variables sampled randomly from $\mathcal{N}(0, 1)$. For each of the 120 simulation settings, we generate 100 data sets and average the results over these replications. The range of simulations setups under consideration is fairly broad. The setups with $n = 800$ are roughly covered by the Theorems in Section 3, whereas the $n = 80$ settings complement these results in a higher-dimensional setting. For each combination of n and p, the range of γ values more or less covers the transition from poor performances (for low values of γ) to excellent performances (for larger γ). Finally the values of K cover many practical clustering tasks, where often a relatively small number of clusters are sought.

Before clustering, we standardize the data so that each variable has a mean of 0 and a variance of 1.

In order to evaluate clustering performance, we calculate the adjusted rand index (ARI) (Rand, 1971; Hubert and Arabie, 1985) between the estimated partition and the true clustering. The ARI has an expected value of 0 for random clusterings of the data. A perfect agreement between the true and estimated partitions corresponds to an ARI of 1.

We compare the regularized K-means algorithms with the different penalties. Each of the methods is calculated on a grid of 40 lambda values given by $10^{-2+4i/40}$, for $i =$

t]

	$\gamma = 0.4$	$\gamma = 0.5$	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$
classical	0.08 (0.05)	0.19 (0.08)	0.35 (0.11)	0.55 (0.12)	0.69 (0.12)
lasso	**0.15** (0.06)	0.32 (0.1)	0.69 (0.19)	0.97 (0.06)	**1** (0.02)
ridge	**0.15** (0.05)	0.28 (0.08)	0.44 (0.11)	0.61 (0.13)	0.75 (0.14)
glasso	**0.15** (0.06)	**0.36** (0.13)	0.80 (0.19)	0.99 (0.01)	0.99 (0.1)
ℓ_0	**0.15** (0.06)	0.34 (0.12)	**0.86** (0.17)	**1** (0.01)	**1** (0)

Table 2: mean ARI values (and standard deviation in brackets) of regularized K-means variants on data with $p = 1000$, $n = 80$ and $K = 4$.

$0, 1, \ldots, 39$, after which the best solution is retained. Throughout the paper, we will take the setting of $n = 80$, $K = 4$ and $p = 1000$ as the leading example to present the results. This is the setting used in Sun et al. (2012). Due to its difficulty, it allows us to clearly distinguish between different clustering approaches. The results for the other 115 settings are always qualitatively similar and a summary of these can be found in Section D of the Appendix.

Note that we assume K to be known throughout the simulation studies.

5.2 Comparison of Penalty Types

In this simulation we compare the different penalty types and make a case for $\mathcal{P}_0$. In order to evaluate the performance of the different penalties independently from the problem of selecting λ, we proceed as follows. For each generated data set and penalty type, we compute the solution to Equation 2 for the grid of values of the regularization parameter, after which we retain the partition that is closest to the theoretically correct partition (the one with the largest ARI). The performance for this data set and penalty type is this largest achieved ARI. The results of this experiment indicate which penalty has the most potential, provided that the tuning parameters are well chosen.

Table 2 shows the results for the setting with $n = 80$ and $p = 1000$. We note that the scenarios with $\gamma = 0.4$ and $\gamma = 0.5$ are very hard for all penalty types and none of them achieve a satisfactory performance. As the clusters get more separated, the penalized K-means start to substantially outperform classical K-means. Out of the different penalty types, the ridge penalty is the least effective, whereas the ℓ_0 is the most effective. The group-lasso penalty is a close second, and the lasso penalty falls somewhere in between.

Similar results are obtained for the other simulation settings. These results are summarized in Section D.1 of the Appendix. In addition to our theoretical results, we thus find empirical support for the use of the ℓ_0 penalty and the proposed HTK-means algorithm.

5.3 Comparison of Methods to Select λ

We now compare the methods for selecting λ described in Section 4 in a simulation study. We focus here on the ℓ_0 penalty, our main proposal. As before, we present the case of $K = 4$, $n = 80$ and $p = 1000$ here, and refer to Section D.2 of the Appendix for the results

t]

	$\gamma = 0.4$	$\gamma = 0.5$	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$
AIC	**0.09** (0.06)	**0.26** (0.12)	**0.80** (0.19)	0.98 (0.98)	**1** (0.01)
BIC	0.05 (0.05)	0.21 (0.12)	0.79 (0.24)	**0.99** (0.99)	**1** (0)
stab1	0.08 (0.05)	0.22 (0.08)	0.47 (0.21)	0.71 (0.71)	0.82 (0.17)
stab2	0.08 (0.05)	0.20 (0.09)	0.47 (0.23)	0.72 (0.72)	0.84 (0.17)
stab3	0.08 (0.06)	0.21 (0.09)	0.48 (0.21)	0.69 (0.69)	0.84 (0.2)

Table 3: mean ARI (and standard deviation in brackets) of lambda selection techniques on data with $p = 1000$, $n = 80$ and $K = 4$.

obtained under the other simulation setups. For the stability based methods, 20 replications of the resampling strategy were used.

Table 3 presents the resulting ARI values. First of all note that the simple AIC criterion is almost always among the best performing methods. While the BIC performs similar to the AIC in this setting, we found that the AIC is more consistent in situations with lower p, as can be seen from the additional simulation results in Section D.2 of the Appendix. The stability based selection techniques do not seem appropriate here, and only start performing reasonably well in the simplest case of $\gamma = 0.8$.

	$\gamma = 0.4$	$\gamma = 0.5$	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$
AIC	100 (18)	99 (26)	81 (24)	89 (11)	90 (7)
BIC	6 (3)	11 (6)	36 (10)	49 (2)	50 (1)
stab1	487 (317)	502 (303)	491 (299)	518 (302)	542 (285)
stab2	503 (318)	518 (314)	463 (314)	478 (302)	503 (301)
stab3	496 (333)	523 (313)	483 (310)	547 (326)	467 (301)

Table 4: mean number of selected variables (and standard deviation in brackets) of lambda selection techniques on data with $p = 1000$, $n = 80$ and $K = 4$.

In addition to the resulting ARI values, we consider the number of selected variables. Remember that the true number of informative variables is 50. Table 4 shows the number of selected variables for the scenarios under consideration. It is clear that it is quite difficult to select the correct number of variables. BIC seems to be closest to the true number of informative variables, but only when the cluster centers are very well separated. AIC appears to consistently overestimate the true number of informative variables, but as discussed before, this does not seem to lead to inferior performance in terms of recovering the class memberships. The stability-based methods select way too many variables, again suggesting that they are not preferable to use in combination with the ℓ_0 penalty.

Finally, we briefly consider the computation times of the different methods. It is clear that AIC and BIC are very quick to compute, since they require virtually no additional calculations once the regularized K-means algorithm has been executed on a grid of values

for λ. The stability-based criteria require substantially more computation time, and this is of course due to the repeated runs of regularized K-means on all the bootstrap samples.

	$\gamma = 0.4$	$\gamma = 0.5$	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$
AIC	0.23 (0.01)	0.23 (0.02)	0.23 (0.02)	0.25 (0.01)	0.23 (0)
BIC	0.17 (0.01)	0.17 (0.01)	0.17 (0)	0.18 (0.01)	0.18 (0)
stab1	878 (32)	886 (29)	918 (31)	944 (19)	944 (33)
stab2	878 (32)	886 (29)	918 (31)	944 (19)	944 (33)
stab3	878 (32)	886 (29)	918 (31)	944 (19)	944 (33)

Table 5: mean computation time in seconds (and standard deviation in brackets) of lambda selection techniques on data with $p = 1000$, $n = 80$ and $K = 4$.

We end the discussion on the selection of λ with a remark. While automatic selection of the regularization parameter is attractive, we have found that in practice, there seems not to be any one-size-fits all method. Therefore, we encourage applying several methods and comparing the conclusions and results. This process can be supported by graphical displays of the regularization path and changing partition for increasing values of λ, as we will illustrate in Section 6 with real data examples.

5.4 Comparison of HTK-means with Existing Proposals

In this section we compare HTK-means with the most popular competitors. The best-known competitor is the *sparse K-means* method of Witten and Tibshirani (2010). Sparse K-means was shown to outperform several alternative approaches such as the COSA method (Friedman and Meulman, 2004), the model-based clustering of Raftery and Dean (2006) and PCA followed by K-means. We use the implementation of sparse K-means provided in the `R`-package `sparcl` by Witten and Tibshirani (2018). The tuning parameter is chosen by the proposed permutation approach using 20 permutations searching over a grid of length 40. We additionally include the regularized K-means (henceforth Reg K-means) of Sun et al. (2012), where the tuning parameter is chosen using the proposed stability criterion over 20 bootstrap replications and a grid of 40 lambda values given by $10^{-2+4i/40}$, for $i = 0, 1, \ldots, 39$. Finally, we compare with classical K-means which serves as a reference.

Table 6 presents the ARI results on data of dimension $p = 1000$ with $K = 4$ and $n = 80$, whereas the results for the other settings are summarized in Section D.3 of the Appendix. Several things can be noted. First, we see again that the cases of little separation between the cluster centers ($\gamma = 0.4$ and $\gamma = 0.5$) are really difficult, and none of the methods has a satisfactory performance. As the clusters get more separated, the clustering task clearly becomes easier. The performance of HTK-means is never worse than that of the competitors, and substantially better in the case of $\gamma = 0.6$, $\gamma = 0.7$ and $\gamma = 0.8$. Sparse K-means is the second best performing method, with very competitive performance for $\gamma = 0.8$ and a reasonable performance for $\gamma = 0.6$ and $\gamma = 0.7$. Reg K-means does not seem to be doing much better than classical K-means in this simulation. We note that the outperformance of the other methods by HTK-means for $\gamma = 0.5, 0.6, 0.7, 0.8$ is statistically significant when tested based on the differences in means and standard deviations (for example using Welch's

t-test). Finally, note that classical K-means starts to perform reasonably well as the cluster centers get more and more separated.

	$\gamma = 0.4$	$\gamma = 0.5$	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$
HTK-means	**0.09** (0.06)	**0.26** (0.12)	**0.80** (0.19)	**0.98** (0.03)	**1** (0.01)
Reg K-means	**0.09** (0.05)	0.19 (0.08)	0.36 (0.12)	0.57 (0.15)	0.72 (0.14)
Sparse K-means	0.05 (0.05)	0.18 (0.1)	0.66 (0.27)	0.89 (0.15)	0.96 (0.08)
K-means	**0.09** (0.05)	0.19 (0.08)	0.36 (0.11)	0.56 (0.12)	0.69 (0.12)

Table 6: mean ARI (and standard deviation in brackets) on data with $p = 1000$, $n = 80$ and $K = 4$.

We now briefly consider the number of selected variables for each of the methods, shown in Table 7. The AIC criterion used for HTK-means consistently underestimates the sparsity of the signal and selects a few too many variables on average. However, out of all the methods, it is closest to the true number of signal variables (50) most of the time. Sparse K-means seems to select too many variables when the clusters are not very well separated. However, for well-separated clusters ($\gamma = 0.8$) it selects almost exactly 50 variables on average, albeit with high variability. Finally, Reg K-means heavily underestimates the sparsity of the signal.

	$\gamma = 0.4$	$\gamma = 0.5$	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$
HTK-means	100 (18)	99 (26)	81 (24)	89 (11)	90 (7)
Reg K-means	708 (242)	724 (239)	719 (252)	720 (246)	654 (274)
Sparse K-means	237 (331)	293 (306)	131 (208)	41 (12)	50 (93)
K-means	1000 (0)	1000 (0)	1000 (0)	1000 (0)	1000 (0)

Table 7: mean number of selected variables (and standard deviation in brackets) on data with $p = 1000$, $n = 80$ and $K = 4$.

Finally we take a brief look at the computation times of the different methods. Table 8 shows the computation times in seconds. It is immediately clear that classical K-means is by far the fastest method and the regularized alternatives have a computation time that is larger by several orders of magnitude. Of the alternatives to classical K-means, HTK-means is the fastest to compute followed by Sparse K-means which is about 4 times as slow on these data. Reg K-means is much slower than the competitors, and the bulk of this computation time is due to the stability-based tuning of the regularization parameter λ.

The computation times of the classical K-means method may seem rather low compared with the other variants. One of the main drivers of this discrepancy is the fact that the implementation of classical K-means uses highly optimized C code, whereas the other methods use pure `R` code. Additionally, the regularized and sparse variants are calculated on a grid of values for the regularization parameter, which increases the computation time proportionally to the size of this grid. Finally, the HTK-means in particular uses a sparse

	$\gamma = 0.4$	$\gamma = 0.5$	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$
HTK-means	28 (1)	28 (1)	29 (1)	29 (1)	28 (1)
Reg K-means	1991 (61)	1999 (55)	1983 (61)	2007 (54)	2000 (60)
Sparse K-means	100 (2)	100 (3)	101 (3)	103 (2)	101 (2)
K-means	1 (0)	1 (0)	1 (0)	1 (0)	1 (0)

Table 8: mean computation time in seconds (and standard deviation in brackets) on data with $p = 1000$, $n = 80$ and $K = 4$.

starting value, which increases the computation time by a constant factor when compared with classical K-means. That said, for a fixed value of λ, the computational complexity of HTK-means is the same as that of classical K-means. We believe that with enough engineering HTK-means could scale to the same settings that classical K-means can scale to.

5.5 Comparison with Modified Existing Proposals

In this section, we compare our proposal with a modified version of Sparse K-means and Reg K-means. More precisely, we investigate whether selecting λ using the AIC and using our sparse initialization strategy would also benefit both methods. This study can help us to gain insight into whether the improvement of HTK-means over existing methods is due to the use of the new initial values, the AIC for the selection of λ, or the penalty itself. Table 9 shows the ARIs on the simulated data with $p = 1000$ and $K = 4$ and $n = 80$. The "(m)" (for "modified") indicates the versions of Sparse K-means and Reg K-means using our sparse starting value as well as the AIC for the selection of λ. The results in this table indicate that Sparse K-means does not work well with our initialization procedure and the AIC. This is explained by the fact that our initialization is aimed at methods which regularize based on some norm of the cluster centers, which Sparse K-means doesn't do directly. Additionally, Sparse K-means induces (a lot of) shrinkage on the estimated cluster centers which can be detrimental to the use of the AIC as it relies on the WCSS. The difference between HTK-means and modified Reg K-means is very small. The table suggests that HTK-means may be slightly better when γ is small, but no definite conclusions can be drawn here.

	$\gamma = 0.4$	$\gamma = 0.5$	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$
HTK-means	**0.09** (0.06)	**0.26** (0.12)	**0.8** (0.19)	0.98 (0.03)	**1** (0.01)
Reg K-means (m)	0.03 (0.04)	0.23 (0.15)	0.76 (0.24)	**0.99** (0.02)	**1** (0.01)
Sparse K-means (m)	0.02 (0.03)	0.06 (0.05)	0.16 (0.05)	0.21 (0.06)	0.28 (0.09)

Table 9: mean ARI (and standard deviation in brackets) of the proposed method as well as the modified versions (denoted "(m)") of Reg K-means and Sparse K-means on data with $p = 1000$, $n = 80$ and $K = 4$.

Table 10 reports the computation times of this experiment. As is clear from these numbers, the proposed methods requires about half of the computation time of the modified Reg K-means approach and about one-fourth of the modified Sparse K-means method. The algorithm of the HTK-means method is the simplest in nature, yielding the best results from a computational perspective.

	$\gamma = 0.4$	$\gamma = 0.5$	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$
HTK-means	28 (1)	28 (1)	29 (1)	29 (1)	28 (1)
Reg K-means (m)	66 (4)	67 (3)	67 (7)	63 (4)	62 (2)
Sparse K-means (m)	108 (2)	109 (2)	109 (3)	109 (3)	109 (2)

Table 10: mean computation time (and standard deviation in brackets) of the proposed method as well as the modified versions (denoted "(m)") of Reg K-means and Sparse K-means on data with $p = 1000$, $n = 80$ and $K = 4$.

Finally, we take a brief look at the variable selection performance of HTK-means and the modified Sparse and Regularized K-means. Table 11 shows the number of variables selected for each of the methods under consideration. It is confirmed that the AIC method does not work well for Sparse K-means, as the number of selected variables is extremely low each time. For HTK-means, the procedure slightly overestimates the number of signal variables, but it does so in a stable way. For the modified Reg K-means however, the AIC provides a much more volatile selection of the number of relevant variables. For small values of γ it clearly underestimates the number of relevant variables. As suggested earlier, this is likely due to the fact that the WCSS is not comparable for different values of λ when there is (a lot of) shrinkage in the estimated cluster centers. This happens mostly in more difficult clustering scenarios (that is, for small values of γ), explaining the results in the table.

	$\gamma = 0.4$	$\gamma = 0.5$	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$
HTK-means	100 (18)	99 (26)	81 (24)	89 (11)	90 (7)
Reg K-means (m)	8 (4)	18 (13)	59 (16)	66 (7)	65 (6)
Sparse K-means (m)	3 (2)	3 (1)	3 (1)	3 (1)	3 (1)

Table 11: mean number of selected variables (and standard deviation in brackets) of the proposed method as well as the modified versions (denoted "(m)") of Reg K-means and Sparse K-means on data with $p = 1000$, $n = 80$ and $K = 4$.

6. Real Data Examples

In this section we analyze several real data examples using the HTK-means method. We start with a few simple examples and then turn to more complex data sets.

6.1 The Iris Data Set

We first reconsider the Iris data set discussed in the introduction, where it was clear that not all variables contribute equally to the partitioning of the data. More specifically, the third and fourth variable contain most information with respect to the true clustering structure. Adding the first and second variables does not help in recovering the underlying clustering, and in fact worsens the result. When applying HTK-means to these data, we obtain the regularization path on the left panel of Figure 2. As λ decreases, we see that the variables enter the active set one at a time. As discussed in the introduction, the best clustering performance is achieved when using only the petal dimensions, that is, the blue and yellow variables which first enter the regularization path. Including the sepal dimensions (the red and green variables) worsens the result.

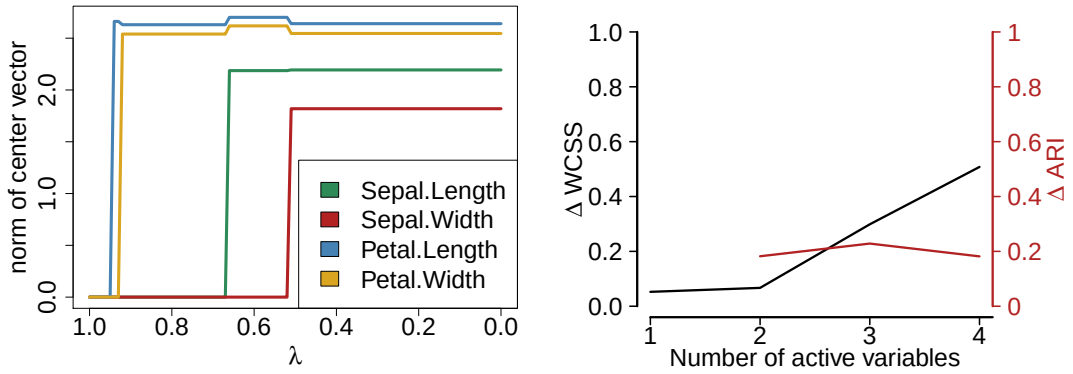

Figure 2: Regularization path of HTK-means on the Iris data (left). Diagnostic plot of HTK-means.

The AIC and BIC criteria select a λ parameter of 0, suggesting that all four variables should be used to cluster the Iris data. The stability based methods select λ between 0.67 and 0.92, meaning that they all select the 2 variables describing the dimensions of the petal and thus achieve the optimal ARI on this data set. In addition to these automatic methods for selecting λ, we may consider the diagnostic plot in the right panel of Figure 2. On the horizontal axis, it shows the number of active variables rather than the value of the regularization parameter. On the left vertical axis, it shows the increase in WCSS. If this is large, it suggests that the added variable is likely to be a noise variable. On the right vertical axis, it shows the difference in ARI between the subsequent partitions of the data. Large values means that the added variable has caused the current partition to change drastically. From this plot, we can infer that the inclusion of the third and fourth variable produces a large increase in WCSS compared with the inclusion of the first two variables. We also see that the partition keeps changing substantially when we add more variables. The graphical display consistently points to the optimal use of the petal dimensions for the clustering of the flowers, and a threshold of 0.2 on the increase in WCSS would yield the optimal partition.

6.2 The Banknote Data Set

As a second example we consider the banknote data set, which consists of six measurements for 100 genuine and 100 counterfeit old-Swiss 1000-franc bank notes. The data was analyzed by Flury and Riedwyl (1988) and is publicly available in the `R`-package `mclust` (Scrucca et al., 2016). For each bank note, we have the length, the width of the left and right edges, the bottom and top margin widths and the length of the diagonal. Figure 3 presents a pairs plot of the data, with the genuine and counterfeit bills colored in blue and red respectively. From this plot we may expect that not all variables contribute equally well to the separation between good and bad bank notes.

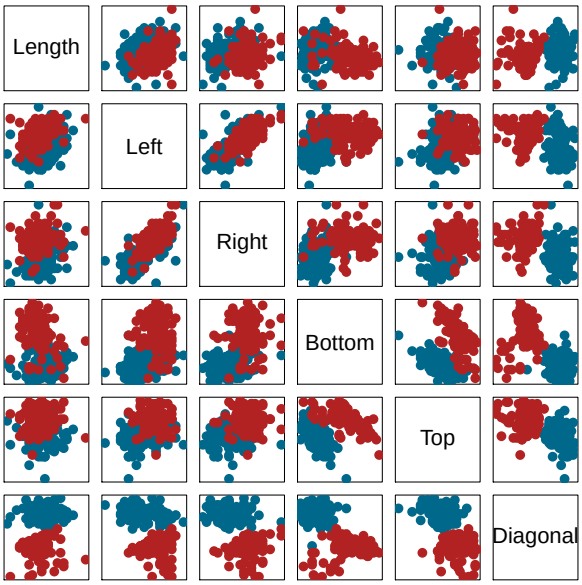

Figure 3: Pairs plot of the banknote data. The genuine bank notes are colored in blue, whereas the counterfeit notes correspond with the red dots.

If we cluster the bank note data using classical K-means, we obtain an ARI of about 0.85, which is already a very good performance. Figure 4 shows the regularization path resulting from applying HTK-means on the bank note data. From this plot, we immediately see that not all variables contribute equally to the clustering of the data. More specifically, it seems that the measurements of the diagonal of the bill is by far the most important variable, followed by the bottom margin variable. It turns out that if we cluster only based on the diagonal measurement, we obtain an ARI of 96 %. If we additionally include the second variable, the bottom margin, we obtain an ARI of 98 %, which is almost perfect recovery of the true clusters. Including additional variables slightly lowers the ARI, but it is the measurements of the length and left edge which make the ARI drop from around 0.95 to 0.85. Any λ value smaller than 0.33 includes these variables and thus we would like to

select a tuning parameter value of at least 0.33. The AIC and BIC criteria select a λ of 0.02, meaning that they leave out the length variable, but still include the left edge variable. The stability based criteria also select λ values between 0.02 and 0.33, essentially selecting 5 variables and yielding a suboptimal ARI. While the automatic tuning for λ does not seem to work well here, the diagnostic plot in the right panel of Figure 4 does points towards a better solution. It clearly shows that the last variable is a noise variable with an addition to the WCSS of the active variables which is almost one. However, the third, fourth and fifth variable also seem to increase the WCSS substantially, and if we would put a threshold at an increase of 0.2 as in the Iris data, we would obtain the partition based on one variable, which has almost perfect recovery.

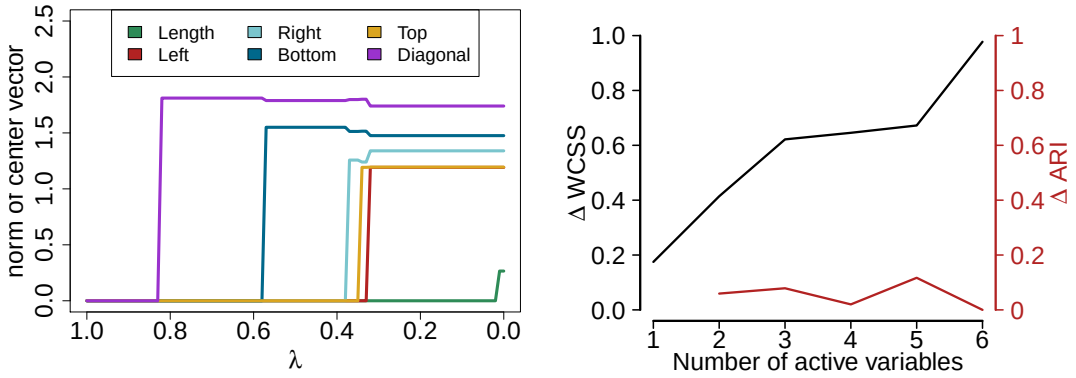

Figure 4: Regularization path of HTK-means on the bank note data (left). Diagnostic plot of HTK-means (right).

6.3 The Colon Cancer Data Set

We analyze the gene expression data publicly available in the `R`-package `antiProfilesData` (Bravo et al., 2020) and contains samples of normal colon tissue and colon cancer tissue collected from the Gene Expression Omnibus (Edgar et al., 2002; Barrett et al., 2012). The complete data set contains 68 gene expressions of length 5339, subdivided into 4 categories: adenoma, colorectal cancer, normal and tumor. There are 15 observations for each of the first three categories, and 23 of the tumor category. We are interested in clustering the data and thereby hopefully recovering the different tissue types in the obtained partition. Furthermore, if we are able to do so using a limited number of variables, this would add valuable insight regarding potentially important genes.

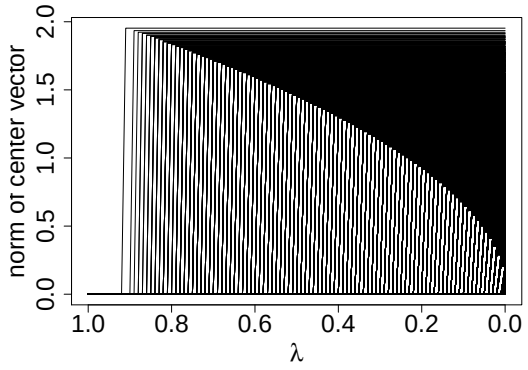

Figure 5: Regularization path of HTK-means on the colon cancer data.

We first consider the simplified problem of separating the normal tissue from the tumor, which together form a data set of size 38×5339. Interestingly, when clustering this data using classical K-means, we obtain a perfect recovery of the true clusters: normal tissue vs. tumor tissue. However, classical K means evidently uses all variables to obtain this partition, and offers no insight into whether all of these variables are needed or whether some of them may be redundant. Figure 5 shows the regularization path resulting from applying HTK-means to the colon cancer data. Clearly, not all variables contribute equally to the clustering, as even for very small values of the regularization parameter λ, many variables are dropped from the clustering. As even classical K-means clusters this data perfectly, we cannot hope to perform better in that respect, but we can try to identify potentially interesting features as well as try to obtain the same perfect partition using fewer variables.

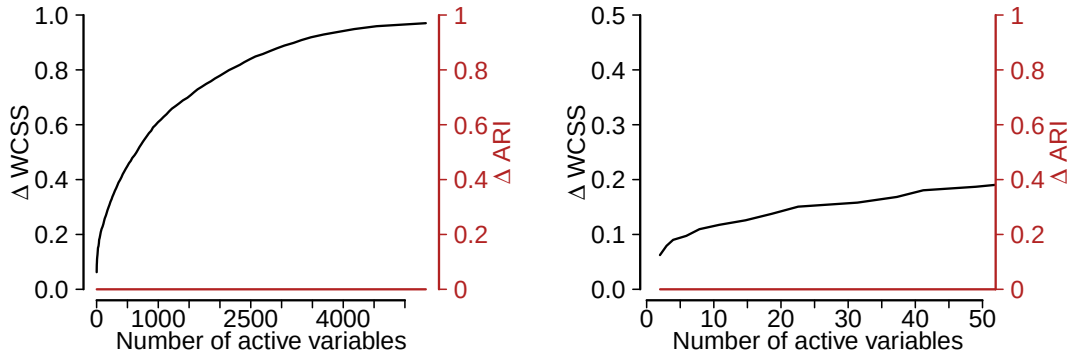

Figure 6: Diagnostic plot of HTK-means on the colon cancer data (left) with zoomed-in plot on the right.

AIC and BIC suggest tuning parameters of 0.11 and 0.19 respectively, which yield clustering models of size 2697 and 1999. The stability based methods stab1, stab2 and stab3 suggest using λ equal to 0.14, 0.08 and 0.26 respectively which correspond with clustering

models of size 2396, 3046 and 1530 respectively. All of these options achieve perfect clustering, but none of them achieve a very sparse clustering. In fact, all sub models along the regularization path achieve perfect clustering, as can be seen from the flat red line on the diagnostic plot in Figure 6. Using a threshold of 0.2 on the increase of WCSS as in the previous two examples, we would find a very sparse model of roughly fifty genes.

Given that the partition is essentially determined by a single variable, it is interesting to consider the first few variables which enter the active set of clustering features. Figure 7 shows the expression levels of the first 4 variables which enter the clustering model. All 4 of these variables perfectly separate the normal tissue samples from the tumor samples, explaining why the data is rather easy to cluster regardless of the tuning parameter. However, HTK-means allows us to identify these variables as they appear first in the regularization path.

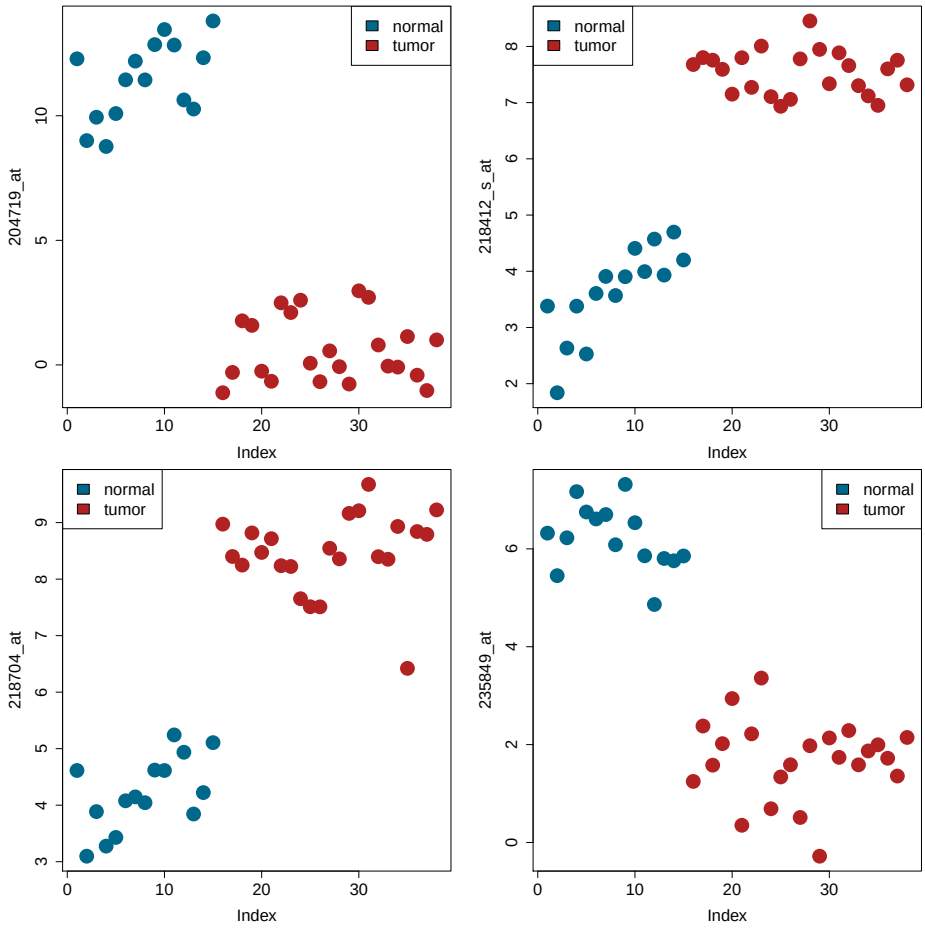

Figure 7: Expression levels of the first 4 features entering the clustering model. All four variables perfectly separate the normal from the tumor samples.

We now consider the full data set, with all 4 classes. This clustering task is significantly more difficult, as evidenced by the ARI of 0.61 achieved by classical K-means. Figure 8 shows the regularization path of HTK-means which is a little bit more noisy than before, but clearly shows unequal importance of the variables in the clustering.

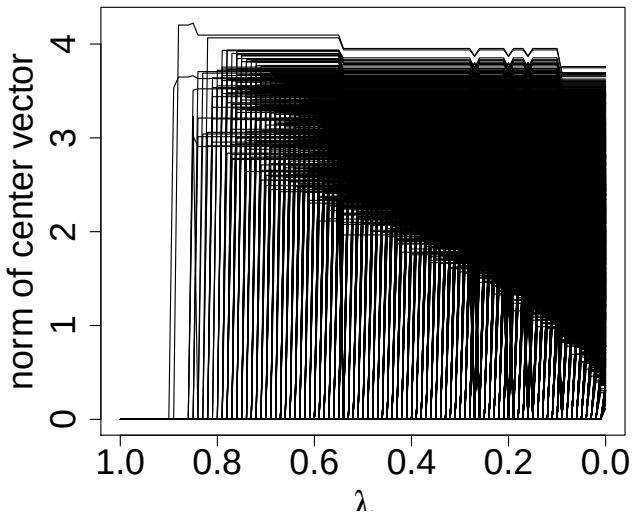

Figure 8: Regularization path of the full colon cancer data set.

For the selection of λ, AIC and BIC suggest values of 0.12 and 0.25, corresponding with partitions based on 3910 and 2568 variables respectively and both achieving an ARI of 0.654. The stability based methods result in more sparse models of between 97 and 329 variables which both have an ARI of about 0.68, the highest achievable using K-means on this data set. Considering the diagnostic plots of Figure 9 however, it seems that these models are still not sparse enough. More in particular, the partition obtained using only the 7 best variables remains unaltered when the next 700 (!) variables are added to the model. In other words, the optimal partition which can be achieved using K-means on a subset of the variables in this data set can be reached using only 7 variables. While the diagnostic plot with a threshold on the WCSS of 0.2 would yield a model of roughly 50 variables, it also shows that it leads to the same partition as the model based on 7 variables, as the red line is flat at 0.

Given that we can essentially reach the 0.68 ARI with only 7 variables, it is interesting to look at the variables which first enter the model. The first 4 variables are shown in Figure 10. Interestingly, the variable which first enters the model, named `204719_at`, was also among the first variables entering when we only considered the normal and tumor categories. This gene seems very important as it clearly distinguishes between healthy tissue and non-healthy tissue of different types. We further see that the other variables which enter the model early mainly distinguish between the adenoma and the other tissue. They also suggest the existence of a sub-cluster within the normal tissue, as all three of the genes `1552863_a_at`, `44673_at` and `213451_x_at` indicate a difference between the first 8 and the last 7 blue dots. It turns out that these observations correspond with tissue

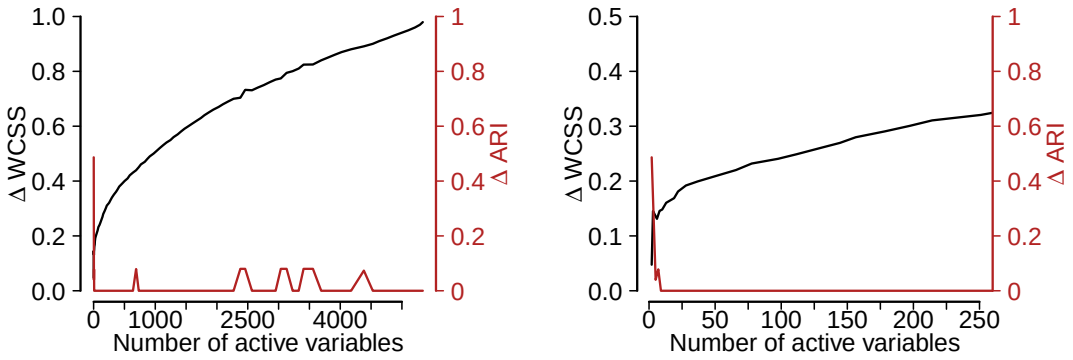

Figure 9: Diagnostic plot of HTK-means on the full colon cancer data (left) with zoomed-in plot on the right.

collected from the rectum mucosa instead of the colon, and so it can in fact be considered a sub-cluster. Finally, we notice that the colorectal cancer and tumor tissue are hard to separate making the clustering task more difficult.

7. Conclusion

Classical K-means is a very popular clustering technique. Despite its popularity, the performance of K-means can be impaired by noise variables which do not contribute to a reliable partitioning of the data. Regularized K-means aims to mitigate this issue by distinguishing important clustering features from noise variables. While variable identification is interesting in its own right, this information can be leveraged to improve the partitioning of the data. We have introduced and analyzed a framework for performing regularized K-means, based on direct penalization of the size of the cluster centers. Based on an extensive simulation study and theoretical analysis, we have proposed a new method called hard-threshold K-means, which uses an ℓ_0 penalty to induce sparsity.

Theoretically, HTK-means is consistent and variable-selection consistent for fixed dimensions without requiring λ to vanish. In the high-dimensional regime where $p = o(n^{1/3})$, we additionally showed that HTK-means can achieve $\sqrt{n/p}$-consistency while also preserving variable selection consistency (unlike the other penalties under consideration). We conjecture that this is the optimal rate for this regime (without any further assumptions on the sparsity structure), as that is the rate achieved for penalized regression of which K-means clustering with $K = 1$ is a special case. Empirically, we have shown that a substantial part of the superior performance of HTK-means is explained by its sparse starting value. The penalty itself shows similar empirical performance to the group-lasso penalty, while outperforming the regular lasso and ridge penalties. The ℓ_0 penalty has the additional benefit of yielding the fastest algorithm and avoiding shrinkage in the non-zero cluster centers. This last property enhances interpretability and visualization, and allows for the justified use of simpler tuning parameter selection methods like AIC and BIC.

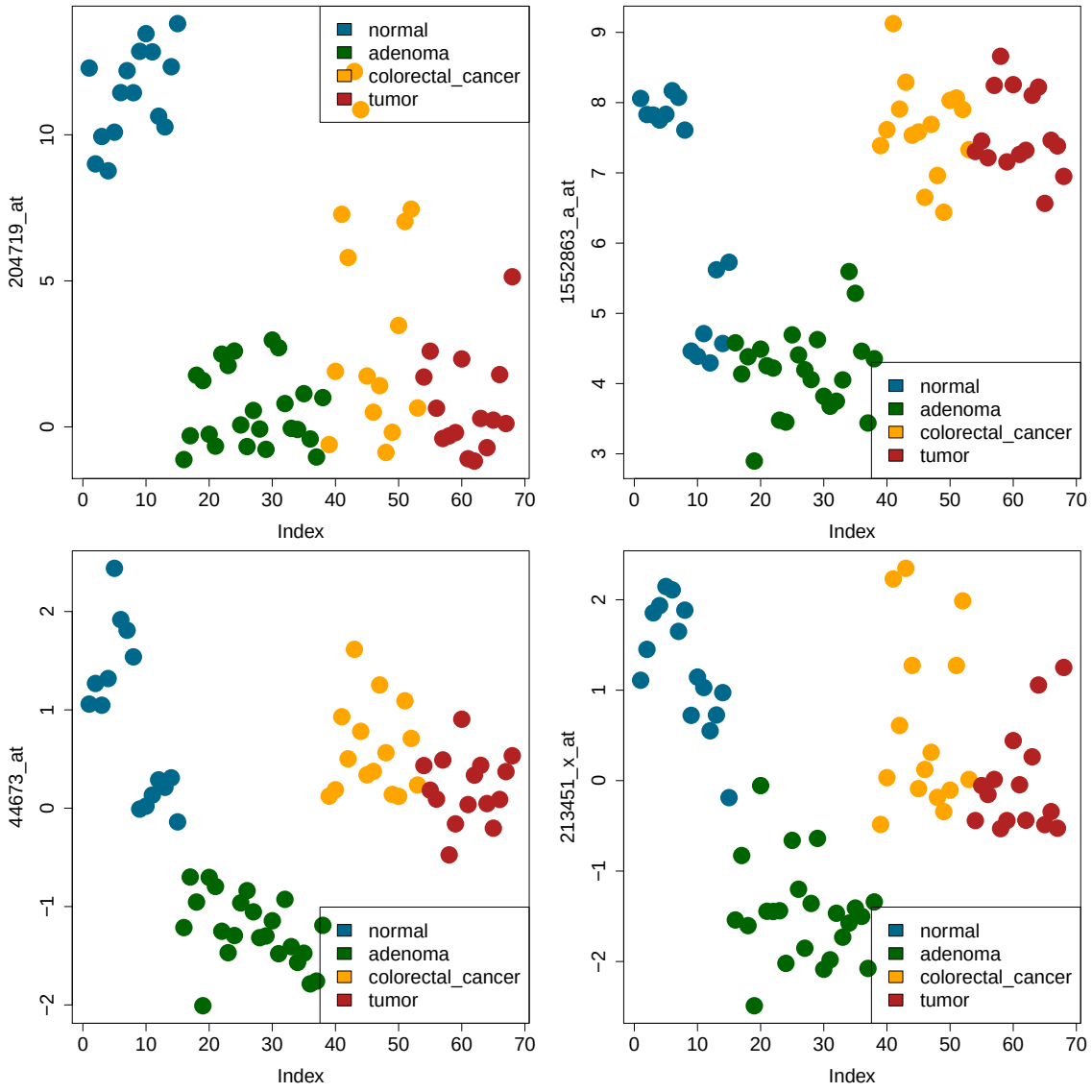

Figure 10: Expression levels of the first 4 features entering the clustering model.

The application of HTK-means to real data examples illustrates its potential in identifying the important clustering variables, which is useful even when a good partitioning of the data can be found without regularization. Additionally, new graphical displays were presented and used to gain further insight into the data sets.

Software Availability. The code for HTK-means and the proposed diagnostic plots is included in the `R` package `clusterHD` (Raymaekers and Zamar, 2022) available on CRAN.

Acknowledgments

This work was supported by internal funds of the KU Leuven [to J.R.] and by the Discovery grant from NSERC [to R.Z.].

Appendix A. Proofs

A.1 Proof of Proposition 1

Proof We want to show that $\text{obj}^* = \text{obj} + 1$. We have:

$$\begin{aligned}
\text{obj}^* &= \min_{\{(\underline{a}_k, c_k)|k=1\ldots,K\}} \sum_{k=1}^{K} \int \cdots \int_{m+1} \left[\min_k ||\underline{x} - \underline{a}_k||^2 + \min_k (y - c_k)^2 \right] \mathbf{1}_{R_k^*}(\underline{x}, y) \mathrm{d}Q^*(\underline{x}, y) \\
&= \min_{\{(\underline{a}_k, c_k)|k=1\ldots,K\}} \sum_{k=1}^{K} \int \cdots \int_{m+1} \left[\min_k ||\underline{x} - \underline{a}_k||^2 + \min_k (y - c_k)^2 \right] \\
&\quad \mathbf{1}_{R_k}(\underline{x}) \mathbf{1}_{\mathbb{R}}(y) \mathrm{d}Q(\underline{x}) \mathrm{d}Q^*(y|\underline{x}) \\
&= \min_{\{(\underline{a}_k, c_k)|k=1\ldots,K\}} \sum_{k=1}^{K} \int \cdots \int_{m+1} \left[\min_k ||\underline{x} - \underline{a}_k||^2 + \min_k (y - c_k)^2 \right] \mathbf{1}_{R_k}(\underline{x}) \mathrm{d}Q(\underline{x}) \mathrm{d}Q^*(y) \\
&\geq \text{obj} + \min_{c_1,\ldots,c_k} \sum_{k=1}^{K} \int \cdots \int_{m} \left[\int \min_k (y - c_k)^2 \mathrm{d}Q^*(y) \right] \mathbf{1}_{R_k}(\underline{x}) \mathrm{d}Q(\underline{x}) \\
&= \text{obj} + \left[\min_{c_1,\ldots,c_k} \int \min_k (y - E[Y]))^2 \mathrm{d}Q^*(y) \right] \sum_{k=1}^{K} \int \cdots \int_{m} \mathbf{1}_{R_k}(\underline{x}) \mathrm{d}Q(\underline{x}) \\
&= \text{obj} + \text{Var}[Y] \sum_{k=1}^{K} P(R_k) \\
&= \text{obj} + 1
\end{aligned}$$

Note that the reverse inequality, $\text{obj}^* \leq \text{obj} + 1$ is obvious, since we can reach this value of the objective function by putting all the cluster centers of the new variable to zero. Assuming the optimum is unique, we thus conclude that the cluster centers for the added variable must be all equal to zero. ■

A.2 Proof of Proposition 2

Proof Suppose that we have an assignment of the elements into K clusters $C_1, \ldots, C_K$. Keeping this assignment fixed, we try to minimize the objective function. We denote the cluster centers obtained by taking the within cluster averages by $\underline{\boldsymbol{\mu}}^*$ throughout. For example, $\underline{\boldsymbol{\mu}}^*_{k,j} := \frac{1}{|C_k|} \sum_{i \in C_k} x_{i,j}$.

$\mathcal{P}_0$ For the ℓ_0 penalty, we have that $\underline{\boldsymbol{\mu}}_{\cdot,j}$ must be the solution to

$$\arg\min_m \frac{1}{n} ||\underline{\boldsymbol{x}}_{\cdot,j} - \underline{\boldsymbol{m}} m||_2^2 + \lambda \mathbf{1}_{||m||_2 > 0}$$

Now suppose we have $||m||_2 \neq 0$. In that case we obtain the exact same optimization step as in classical K-means, since the penalty term is fixed at λ. Therefore, we must have

$$m = \underline{\boldsymbol{\mu}}^*_{\cdot,j} = \left(\underline{\boldsymbol{\mu}}^*_{1,j}, \ldots, \underline{\boldsymbol{\mu}}^*_{K,j} \right).$$

The other option is that we have $||m||_2 = 0$, that is, $m = \mathbf{0}$, which is chosen if it results in a lower objective function. In other words, we have that:

$$\mu_{\cdot,j} = \begin{cases} 0 & \text{if } ||\underline{\boldsymbol{x}}_{\cdot,j}||_2^2 \leq ||\underline{\boldsymbol{x}}_{\cdot,j} - \underline{\boldsymbol{m}}\underline{\boldsymbol{\mu}}^*_{\cdot,j}||_2^2 + n\lambda \\ \mu^*_{\cdot,j} & \text{if } ||\underline{\boldsymbol{x}}_{\cdot,j}||_2^2 > ||\underline{\boldsymbol{x}}_{\cdot,j} - \underline{\boldsymbol{m}}\underline{\boldsymbol{\mu}}^*_{\cdot,j}||_2^2 + n\lambda \end{cases}$$

It goes without saying that this is solved very easily. It is the literal translation of "include a variable iff it reduces the first term of the objective function sufficiently and exclude it otherwise". If a variable is included, then the estimated centers are the within cluster means, that is, there is no shrinkage on the included variables.

$\mathcal{P}_1$ Note that for $\mathcal{P}_1$, the objective function can be split up and optimized for each $\mu_{k,j}$ separately. The optimization problem becomes:

$$\mu_{k,j} = \arg\min_m \frac{1}{n} \sum_{i \in C_k} (\underline{\boldsymbol{x}}_{i,j} - m)_2^2 + \lambda |m|,$$

where m is now a scalar. Taking the derivative with respect to m of the objective function yields:

$$\frac{\partial}{\partial m} \left\{ \frac{1}{n} \sum_{i \in C_k} (\underline{\boldsymbol{x}}_{i,j} - m)_2^2 + \lambda |m| \right\} = -\frac{2}{n} \sum_{i \in C_k} \underline{\boldsymbol{x}}_{i,j} + \frac{2}{n} |C_k| m + \lambda \partial(|m|) = 0$$

where ∂ denotes the subderivative of $|m|$ with respect to m, that is, $\partial(|m|) = \text{sign}(m)$ if $m \neq 0$ and $\partial(|m|) = [-1, 1]$ otherwise.
Now if $m > 0$, we must have $m = \frac{1}{|C_k|} \sum_{i \in C_k} \underline{\boldsymbol{x}}_{i,j} - \frac{n\lambda}{2|C_k|} = \underline{\boldsymbol{\mu}}^*_{k,j} - \frac{n\lambda}{2|C_k|}$, whereas if $m < 0$, we have $m = \frac{1}{|C_k|} \sum_{i \in C_k} \underline{\boldsymbol{x}}_{i,j} + \frac{n\lambda}{2|C_k|} = \underline{\boldsymbol{\mu}}^*_{k,j} + \frac{n\lambda}{2|C_k|}$. The solution is thus given by

$$\underline{\boldsymbol{\mu}}_{k,j} = \max \left(0, 1 - \frac{n\lambda}{2|C_k| \left| \underline{\boldsymbol{\mu}}^*_{k,j} \right|} \right) \underline{\boldsymbol{\mu}}^*_{k,j}.$$

$\mathcal{P}_2$ Analogously to the previous case, we obtain that $\underline{\boldsymbol{\mu}}_{k,j}$ must be the solution to the optimization problem $\arg\min_m \frac{1}{n} \sum_{i \in C_k} (\underline{\boldsymbol{x}}_{i,j} - m)_2^2 + \lambda m^2$, where m is again a scalar. Taking the derivative with respect to m of the objective function yields:

$$\frac{\partial}{\partial m} \left\{ \frac{1}{n} \sum_{i \in C_k} (\underline{\boldsymbol{x}}_{i,j} - m)_2^2 + \lambda m^2 \right\} = -\frac{2}{n} \sum_{i \in C_k} \underline{\boldsymbol{x}}_{i,j} + \frac{2}{n} |C_k| m + 2\lambda m = 0$$

It immediately follows that $\underline{\boldsymbol{\mu}}_{k,j} = \frac{1}{1 + \frac{n\lambda}{|C_k|}} \frac{1}{|C_k|} \sum_{i \in C_k} \underline{\boldsymbol{x}}_{i,j} = \frac{1}{1 + \frac{n\lambda}{|C_k|}} \underline{\boldsymbol{\mu}}^*_{k,j}$ is the solution.

$\mathcal{P}_3$ In this scenario, $\underline{\boldsymbol{\mu}}_{\cdot,j}$ is the solution to

$$\arg\min_m \frac{1}{n} ||\underline{\boldsymbol{x}}_{\cdot,j} - \underline{\boldsymbol{m}} m||_2^2 + \lambda ||m||_2$$

Karush–Kuhn–Tucker conditions yield:

$$-\frac{2}{n}\underline{\boldsymbol{m}}^T(X - \underline{\boldsymbol{m}}m) + \lambda s = 0,$$

where $s = \partial||m||_2$ is the subdifferential given by

$$s_k = \begin{cases} \frac{m_k}{||m||_2} & \text{if } m \neq 0 \\ \{a| \ ||a||_2 \leq 1\} & \text{if } m = 0. \end{cases}$$

Note that $\underline{\boldsymbol{m}}^T X = \left(\sum_{i\in C_1} \underline{\boldsymbol{x}}_{i,j}, \ldots, \sum_{i\in C_k} \underline{\boldsymbol{x}}_{i,j}\right)$ and $\underline{\boldsymbol{m}}^T\underline{\boldsymbol{m}} = \text{diag}(|C_1|, \ldots, |C_k|)$. As a result, we have that the k-th component of $\underline{\boldsymbol{\mu}}_{\cdot,j}$ has to be a solution to:

$$\mu_{k,j} = \frac{1}{|C_k| + n\lambda/(2||\underline{\boldsymbol{\mu}}_{\cdot,j}||_2)} \sum_{i\in C_k} \underline{\boldsymbol{x}}_{i,j} = \frac{1}{1 + n\lambda/(2|C_k|||\underline{\boldsymbol{\mu}}_{\cdot,j}||_2)} \underline{\boldsymbol{\mu}}^*_{k,j}.$$

■

A.3 Proof of Theorem 6

We will make use of the following lemma's to prove Theorem 6.

Lemma 13 *Let $\underline{x}_1 \ldots, \underline{x}_n$ be a random sample from F. Let A_n be the optimal set of (at most k) cluster centers for the sample. Then there exists a $M > 0$ such that for n large enough, we must have*

1. $A_n \subset \mathrm{B}(M)$
2. $\bar{A}(k) \subset \mathrm{B}(M)$

where $\mathrm{B}(M)$ denotes closed the ball with radius M centered at the origin.

Proof We first show that we can find a $M_0 > 0$ so that for n large enough, at least one point of A_n is contained in the closed ball around the origin $\mathrm{B}(M_0)$.

First we take a radius $r > 0$ such that the closed ball $\mathrm{B}(r)$ has positive measure, that is, $\int_{\mathrm{B}(r)} F(d\underline{x}) > 0$. Now we can choose M_0 large enough such that $||M_0 - r||^2 \int_{\mathrm{B}(r)} F(d\underline{x}) > \int ||\underline{x}||_2^2 F(d\underline{x})$. Denote the set which only contains the origin with $A_0 = \{\mathbf{0}\}$. Given that A_n is the set of optimal sample centers, it must hold that $W(A_n, P_n) \leq W(A_0, P_n)$. Additionally, as $\mathcal{P}(A_0) = 0$, we have that

$$\begin{aligned} W(A_0, F_n) &= \int ||\underline{x}||_2^2 F_n(d\underline{x}) + \lambda\, \mathcal{P}(A_0) \\ &= \int ||\underline{x}||_2^2 F_n(d\underline{x}) \\ &\xrightarrow{a.s.} \int ||\underline{x}||_2^2 F(d\underline{x}) \end{aligned}$$

Now suppose that not a single element of A_n would be in $\mathrm{B}(M_0)$ for infinitely many values of n. Then we would have that:

$$\begin{aligned}
\limsup_n W(A_n, F_n) &= \limsup_n \int \min_{a \in A_n} ||\underline{x} - a||_2^2 F_n(d\underline{x}) + \lambda \mathcal{P}(A_n) \\
&\geq \lim_n ||M_0 - r||^2 \int_{\mathrm{B}(r)} F_n(d\underline{x}) \\
&\xrightarrow{a.s.} ||M_0 - r||^2 \int_{\mathrm{B}(r)} F(d\underline{x}) \\
&> \int ||\underline{x}||_2^2 P(d\underline{x}).
\end{aligned}$$

This leads to a contradiction, since if this were true, we would have that $W(A_n, P_n) > W(A_0, P_n)$ infinitely often such that A_n cannot be the optimal set of sample centers. We thus know that there is at least one point, a_0, which lies in A_n and in $\mathrm{B}(M_0)$.

Now we show by induction that as n grows, $\mathrm{B}(5M_0)$ eventually contains all the points of A_n. Suppose that for $1, \ldots, k-1$ cluster centers, $\mathrm{B}(5M_0)$ contains all the optimal sample centers for n large enough. Suppose that A_n would never be contained in $B(5M_0)$ as n grows large. We will show that this would allow for the construction of an allocation of the points to $k-1$ centers which has a lower objective function than the optimal allocation, which is a contradiction.

We start by choosing ε such that $W(\bar{A}(k), F) + \varepsilon < W(\bar{A}(k-1), F)$ and M_0 large enough so that $2\int_{||\underline{x}||>2M_0} ||\underline{x}||_2^2 F(d\underline{x}) < \varepsilon$. Now suppose that there is a point $a^* \in A_n$ which is not in $\mathrm{B}(5M_0)$. If we were to remove this point from A_n, then at worst, all the points currently assigned to the center a^* will now be assigned to the center a_0, known to lie within $B(M_0)$. Additionally, for each of the points $\underline{x}_i$ currently assigned to a^* it must hold that $||\underline{x}_i - a^*||_2^2 < ||\underline{x}_i - a_0||_2^2$. Note that this implies that points must be at a distance of at least $2M_0$ from the origin. Deleting a^*, would therefore cause an increase of the objective function by at most

$$\begin{aligned}
&\int_{||\underline{x}||>2M_0} ||\underline{x} - a_0||_2^2 F_n(d\underline{x}) + \lambda \left(\mathcal{P}(A_n \backslash a^*) - \mathcal{P}(A_n)\right) \\
\leq &\int_{||\underline{x}||>2M_0} ||\underline{x}||_2^2 + ||a_0||_2^2 F_n(d\underline{x}) \\
\leq &2 \int_{||\underline{x}||>2M_0} ||\underline{x}||_2^2 F_n(d\underline{x}).
\end{aligned}$$

Let A_n^* be the set of centers obtained by deleting all the centers from A_n which are not contained in $\mathrm{B}(5M_0)$. Denote with B_n the optimal set of $k-1$ or fewer centers. It is clear that A_n^* should not have a lower objective function than B_n, that is, $W(B_n, F_n) \leq W(A_n^*, F_n)$.

Now, under the assumption that A_n would never be contained in $\mathrm{B}(5M_0)$ as n grows large, we can find a subsequence n_i such that $A_{n_i} \nsubseteq B(5M_0)$ for all $i = 1, 2, \ldots$. We would

then obtain that:

$$\begin{aligned}
W(\bar{A}(k-1),F) &\leq \liminf_{n_i} W(A^*_{n_i},F_n) \text{ a.s.} \\
&= \liminf_{n_i} \int \min_{a\in A^*_{n_i}} ||\underline{x}-a||_2^2 F_n(d\underline{x}) + \lambda\,\mathcal{P}(A^*_{n_i}) \\
&\leq \limsup_{n_i} W(A_n,F_n) + 2\int_{||\underline{x}||>2M_0} ||\underline{x}||_2^2 F_n(d\underline{x}) \\
&\leq \limsup_{n_i} W(\bar{A},F_n) + 2\int_{||\underline{x}||>2M_0} ||\underline{x}||_2^2 F(d\underline{x}) \text{ a.s.} \\
&\leq W(\bar{A}(k),F)+\varepsilon \\
&< W(\bar{A}(k-1),F).
\end{aligned}$$

Which is again a contradiction. Therefore, if we put $M = 5M_0$, it must hold that A_n and $\bar{A}(k)$ are eventually in $\mathrm{B}(M)$ for n large enough. ■

The following lemma proves a uniform strong law of large numbers on the compact set of the previous lemma.

Lemma 14 (Uniform strong law of large numbers) *Given a value $M > 0$, denote with $\mathcal{K} = \{A \subset \mathrm{B}(M) | A$ contains at most k points$\}$*

$$\sup_{A\in\mathcal{K}} |W(A,F_n) - W(A,F)| \xrightarrow{a.s.} 0$$

Proof Note that due to the fact that the penalty is independent of F_n, we have for every A that:

$$|W(A,F_n) - W(A,F)| = \left| \int \min_{a\in A} ||\underline{x}-a||_2^2 F_n(d\underline{x}) - \int \min_{a\in A} ||\underline{x}-a||_2^2 F(d\underline{x}) \right|.$$

Therefore, the uniform SLLN for classical K-means (Pollard, 1981) directly yields the desired result. ■

The following lemma shows the continuity of W with respect to the Hausdorff distance for all penalties except for the ℓ_0.

Lemma 15 (Continuity of W) *For ridge, lasso and group-lasso penalties, the function $C \to W(C,F)$ is continuous on $\mathcal{K}$ under the topology induced by the Hausdorff distance.*

Proof Take $A, B \in \mathcal{K}$ so that $H(A,B) < \delta$. Then $\forall b \in B$, $\exists a(b) \in A$ so that $||a(b)-b||_2 < \delta$.

We first show that that the penalties under consideration are continuous with respect to the Hausdorff metric. If δ is taken small enough, and in particular smaller than the minimal distance between two points of the set A, then we can construct a bijection between the points of A and the points of B such that each point a of A corresponds with exactly one point b of B for which $||a-b||_2 < \delta$. Therefore, if we denote the matrices

$$A = \begin{bmatrix} a_{1,1} & \dots & a_{1,p} \\ \vdots & \ddots & \vdots \\ a_{K,1} & \dots & a_{K,p} \end{bmatrix} \qquad B = \begin{bmatrix} b_{1,1} & \dots & b_{1,p} \\ \vdots & \ddots & \vdots \\ b_{K,1} & \dots & b_{K,p} \end{bmatrix}$$

then we have that, after possible rearrangement of the rows of A and B, $H(A,B) < \delta$ implies that $|a_{i,\cdot} - b_{i,\cdot}| < \delta$ for $i \in \{1,\dots,K\}$ and thus also that $|a_{i,j} - b_{i,j}| < \delta$ for all $i \in \{1,\dots,K\}$ and $j \in \{1,\dots,p\}$. This guarantees that $\mathcal{P}(A) - \mathcal{P}(B)$ can be made arbitrarily small as long as δ is chosen small enough. Because of the continuity of the penalty functions, we can bound the difference in penalties of A and B as $\mathcal{P}(A) - \mathcal{P}(B) \leq f(\delta)$ where f is non-decreasing and such that $\lim_{\delta \to 0} f(\delta) = 0$.

If $R > 5M + \delta$, we now have:

$$\begin{aligned}
W(A,Q) - W(B,Q) &= \int \min_{a \in A} ||\underline{x} - a||_2^2 Q(d\underline{x}) - \int \min_{b \in B} ||\underline{x} - b||_2^2 Q(d\underline{x}) + \lambda \left(\mathcal{P}(A) - \mathcal{P}(B)\right) \\
&\leq \int \max_{b \in B} \left\{ (||\underline{x} - a(b)||_2^2 - ||\underline{x} - b||_2^2) \right\} Q(d\underline{x}) + \lambda f(\delta) \\
&\leq \int \sum_{b \in B} \left\{ (||\underline{x} - a(b)||_2^2 - ||\underline{x} - b||_2^2) \right\} Q(d\underline{x}) + \lambda f(\delta) \\
&\leq \int \sum_{b \in B} \left\{ ((||\underline{x} - b||_2 + \delta)^2 - ||\underline{x} - b||_2^2) \right\} Q(d\underline{x}) + \lambda f(\delta) \\
&\leq k \sup_{||\underline{x}|| \leq R} \sup_{b \in B(5M)} \left| \left((||\underline{x} - b||_2 + \delta)^2 - ||\underline{x} - b||_2^2 \right) \right| \\
&+ k \int_{||\underline{x}|| > R} \left(||\underline{x}||_2 + ||b||_2 + \delta \right)^2 P(d\underline{x}) + \lambda f(\delta) \\
&\leq k \sup_{||\underline{x}|| \leq R} \sup_{b \in B(5M)} \left| \left((||\underline{x} - b||_2 + \delta)^2 - ||\underline{x} - b||_2^2 \right) \right| \\
&+ 2k \int_{||\underline{x}|| > R} ||\underline{x}||_2^2 P(d\underline{x}) + \lambda f(\delta)
\end{aligned}$$

where the last inequality holds because $R > 5M + \delta$ and $b \leq 5M$. The second term will be small when R is chosen large enough, while the first and third will be small when δ is chosen small enough.

■

Lemma 16 *We have that*

$$W(A_n, F_n) \xrightarrow{a.s.} W(\bar{A}, F)$$

Proof We have that

$$W(A_n, F_n) - W(A_n, F) \leq W(A_n, F_n) - W(\bar{A}, F) \leq W(\bar{A}, F_n) - W(\bar{A}, F)$$

The left hand side converges to zero almost surely because of Lemma 14. The right hand side converges to zero almost surely due to the strong law of large numbers. We thus have that

$$W(A_n, F_n) - W(\bar{A}, F) \xrightarrow{a.s.} 0$$

■

Lemma 17 *Consider the function* $\rho : \mathcal{K} \mapsto \mathbb{R} : C \to W(C, F)$ *with* $\mathcal{K} = \{A \subset \mathrm{B}(M) | A \text{ contains at most } k \text{ points}\}$ *with unique minimum* $\rho(\bar{A}) = W(\bar{A}, F)$. *Suppose we have a sequence of sets of centers* $(A_n)_{n\in\mathbb{N}} \subseteq \mathcal{K}$ *for which*

$$\forall \eta > 0, \exists n_0 : \forall n \geq n_0, \rho(A_n) < \eta + \rho(\bar{A})$$

then A_n *converges to* $\bar{A}$ *in Hausdorff distance.*

Proof Note that $\mathcal{K}$ is a compact metric space under the Hausdorff metric.
Choose $\varepsilon > 0$ and consider the subset of $\mathcal{K}_1$ of $\mathcal{K}$ which contains sets of cluster centers located at a Hausdorff distance of at least ε from the optimal set $\bar{A}$. In other words, we have:

$$\mathcal{K}_1 \coloneqq \mathcal{K} \setminus \mathring{\mathrm{B}}_{\bar{A}}(\epsilon),$$

where $\mathring{\mathrm{B}}_{\bar{A}}(\varepsilon)$ denotes the open ball centered around $\bar{A}$ with radius ε. Note that $\mathcal{K}_1$ is also compact.

We now claim that there is a $B^* \in \mathcal{K}_1$ such that

$$\min_{B\in\mathcal{K}_1} \rho(B) = \rho(B^*)$$

In case ρ is continuous as in Lemma 15, its minimum is well-defined over $\mathcal{K}_1$ and so the claim above is naturally true. Now, suppose we are dealing with the ℓ_0 penalty, that is, $W(A, F) = \int \min_{a\in A} ||\underline{x} - a||_2^2 F(d\underline{x}) + \lambda \sum_{j=1}^p \mathbf{1}_{||a_{\cdot,j}||_2 > 0}$.

Consider $\mathcal{K}_{(s)} \coloneqq \mathcal{K}_1 \cap \mathbb{R}^p_s$, where $s \in \{1, \ldots, p\}$ and $\mathbb{R}^p_s = \{\underline{x} \in \mathbb{R}^p | \underline{x} \text{ has at most } s \text{ non-zeroes}\}$. That is, we subdivide the space $\mathcal{K}_1$ into sets which are restricted to have at most s variables equal to zero. So we have that $\mathcal{K}_{(0)} \subset \mathcal{K}_{(1)} \subset \ldots \subset \mathcal{K}_{(p)} = \mathcal{K}_1$.

Consider the first part of the objective function W in: $\rho^* : \mathcal{K} \mapsto \mathbb{R} : C \to W^*(C, F) = \int \min_{c\in C} ||\underline{x} - c||_2^2 F(d\underline{x})$. This is the classical K-means objective function, which is continuous with respect to the Hausdorff metric.

Now let $m_0 = W(\mathbf{0}, F) = W^*(\mathbf{0}, F) = \int ||\underline{x}||_2^2 F(d\underline{x})$ and consider $\mathcal{K}_{(1)} \supset \mathcal{K}_{(0)}$. Note that ρ^* is continuous on $\mathcal{K}_{(1)}$ (ρ is not, since it is not continuous in $\mathbf{0}$) and it therefore attains a minimum, m_1^*. Now take $m_1 = \min(m_0, m_1^* + \lambda)$, then we must have that $\min_{B\in\mathcal{K}_{(1)}} \rho(B) = m_1$. We can continue in similar fashion for $\mathcal{K}_{(2)}$. ρ^* is continuous on $\mathcal{K}_{(2)}$ and attains a minimum m_2^*. We can then take $m_2 = \min(m_1, m_2^* + 2\lambda)$ so that $\min_{B\in\mathcal{K}_{(2)}} \rho(B) = m_2$. Continuing like this we find a value m_p so that $\min_{B\in\mathcal{K}_{(p)}} \rho(B) = m_p$. Denote the set of centers in which this minimum is attained by B^*, that is, $\rho(B^*) = m_p$.

We now have shown that B^* exists for all of the penalties under consideration. Due to the uniqueness of $\bar{A}$, we must have

$$\rho(B^*) > \rho(\bar{A})$$

Now, if we take $\eta = \rho(B^*) - \rho(\bar{A})$, the Lemma hypothesis guarantees that there exists an n_0 such that, for all $n \geq n_0$,

$$\rho(A_n) < \eta + \rho(\bar{A}) = \rho(B^*).$$

This ensures A_n cannot belong to $\mathcal{K}_1$ and thus $A_n \in \mathring{\mathrm{B}}_{\bar{A}}(\epsilon)$ for all $n \geq n_0$. Therefore, A_n has to converge to $\bar{A}$ in Hausdorff distance. ■

The proof of Theorem 6 now follows:

Proof Lemma 13 ensures that the optimal cluster centers for the sample, A_n as well as for the population, $\bar{A}(k)$, will eventually be contained in a closed ball around the origin. We now also have that the sequence A_n of optimal centers of the sample is a minimizing sequence for $W(\cdot, F)$:

$$W(A_n, F) \xrightarrow{a.s.} W(\bar{A}, F).$$

This follows from

$$0 \leq W(A_n, F) - W(\bar{A}, F) \leq (W(A_n, F) + W(A_n, F_n)) + (W(A_n, F_n) - W(\bar{A}, F))$$

where both terms on the right hand side converge to zero almost surely, as a consequence of Lemmas 14 and 16. As a result, the condition of Lemma 6 is satisfied, after which the convergence in Hausdorff metric of A_n to $\bar{A}$ follows. ■

A.4 Proof of Theorem 7

For Theorem 7 we have:

Proof Suppose $\lambda > 0$.

$\mathcal{P}_0$ For the ℓ_0 penalty, we have that $\hat{A}_{\cdot,j} \neq 0$ only if

$$\begin{aligned} &\frac{1}{n}||\underline{\boldsymbol{x}}_{\cdot,j}||_2^2 > \frac{1}{n}||\underline{\boldsymbol{x}}_{\cdot,j} - \underline{\boldsymbol{m}}\hat{A}_{\cdot,j}||_2^2 + \lambda \\ \Leftrightarrow &\frac{1}{n}||\underline{\boldsymbol{x}}_{\cdot,j}||_2^2 - \frac{1}{n}||\underline{\boldsymbol{x}}_{\cdot,j} - \underline{\boldsymbol{m}}\hat{A}_{\cdot,j}||_2^2 > \lambda \end{aligned}$$

As the left hand side converges in probability to zero, we must have that $P(\hat{A}_{\cdot,j} = 0) \to 1$.

$\mathcal{P}_1$ The KKT conditions yield that if $\hat{A}_{\cdot,j} \neq 0$, we must have

$$1 - \frac{\lambda}{2|C_k| \left| \frac{1}{|C_k|} \sum_{i \in C_k} \underline{\boldsymbol{x}}_{i,j} \right|} > 0$$

Therefore, we must have $\frac{1}{|C_k|} \sum_{i \in C_k} \underline{\boldsymbol{x}}_{i,j} > \frac{\lambda}{2|C_k|}$. The left hand side converges in probability to 0 as a result of the consistency. Therefore, we must have that $P(\hat{A}_{\cdot,j} = 0) \to 1$.

$\mathcal{P}_3$ Based on KKT conditions, we have that if $\hat{A}_{\cdot,j} \neq 0$, we must have

$$-\frac{2}{n} \underline{\boldsymbol{m}}^T (\underline{\boldsymbol{x}}_{\cdot,j} - \underline{\boldsymbol{m}} \hat{A}_{\cdot,j}) + \lambda \frac{\hat{A}_{\cdot,j}}{||\hat{A}_{\cdot,j}||_2} = 0$$

Now, note that $\left|\left| \lambda \frac{\hat{A}_{\cdot,j}}{||\hat{A}_{\cdot,j}||_2} \right|\right|_2 = \lambda > 0$, whereas $\left|\left| \frac{2}{n} \underline{\boldsymbol{m}}^T (\underline{\boldsymbol{x}}_{\cdot,j} - \underline{\boldsymbol{m}} \hat{A}_{\cdot,j}) \right|\right|_2 \xrightarrow{P} 0$ ■

Corollary 8 is now a direct consequence of Theorem 6, Theorem 7 and Proposition 2.

A.5 Proof of Theorem 9

We make the following assumptions in this section, in line with the assumptions used in Sun et al. (2012):

(i) $X_j = \bar{\underline{\boldsymbol{m}}} \bar{A}_{\cdot,j} + \varepsilon_j$ for $j = 1, \ldots, p$, where $\varepsilon_j = (\varepsilon_{1j}, \ldots, \varepsilon_{nj})'$ with ε_{ij} independent, $E[\varepsilon_{ij}] = 0$ and $E[\varepsilon_{ij}^2] < \infty$.

(ii) The true cluster centers $\bar{A}$ is unique up to relabeling of its coordinates

(iii) $\int ||x||_2^2 F(dx) < \infty$

(iv) F has a continuous density f on $\mathbb{R}^p$.

(v) There exists a dominating function g such that $f(x) \leq g(||x||)$ for which $r^p g(r)$ is integrable.

(vi) The matrix Γ of Pollard (1982) is positive semi definite at $\bar{A}$.

(vii) $\arg\min_{1 \leq k \leq K} ||X - \bar{A}_{k,\cdot}||^2$ is unique with probability one.

(viii) $\min_{1 \leq j < p_0} \frac{||A_{n;\cdot,j}||}{\lambda_n} \to 0$ as $n \to \infty$.

where $\bar{\underline{\boldsymbol{m}}}$ denotes matrix of cluster assignments based on the true centers $\bar{A}$.

Proof Our proof follows that of Sun et al. (2012), to which we refer for the details. Some adjustments need to be considered for the changing penalty terms, and those are highlighted here. First we show that for large n, the estimated cluster centers lie in a compact region of $\mathbb{R}^p$. This follows from the fact that the statement holds for classical K-means, in combination with the observation that all of the regularized K-means variants can

be written as a constrained optimization problem where the constraint vanishes as $n \to \infty$. As an example, for the ℓ_0 penalty, we can rewrite the objective as

$$\int \min_{a \in A} ||x - a||_2^2 F_n(dx) \text{ so that } \sum_{j=1}^{p} \mathbf{1}_{||A_{\cdot,j}||>0} \leq s_n$$

where $s_n \to \infty$ as $n \to \infty$. We know that the cluster centers of the classical K-means are contained in a closed ball $\mathrm{B}(M)$ around the origin for large n. We can then pick N large enough so that the constraint s_N does not interfere with this ball. The other penalties can be treated in similar fashion.
Secondly, a uniform SLLN for $W(C, F)$ over the subsets of $\mathrm{B}(M)$ can be obtained by considering that

$$\begin{aligned}
&\sup_{A \in B(M)} |W(A, F_n) - W(A, F)| \\
\leq &\sup_{A \in B(M)} \left| \int \min_{a \in A} ||\underline{x} - a||_2^2 F_n(d\underline{x}) - \int \min_{a \in A} ||\underline{x} - a||_2^2 F(d\underline{x}) \right| \\
+ &\sup_{A \in B(M)} \lambda \mathcal{P}(A).
\end{aligned}$$

For the first term, there is almost sure convergence to zero as $p = o(n)$ and the uniform SLLN of classical K-means. The second term goes to zero because for all the penalties under consideration, we have $\sup_{A \in B(M)} \lambda \mathcal{P}(A) = \mathcal{O}(Mp\lambda) = \mathcal{O}(p\lambda)$ and $p\lambda \to 0$ as $n \to \infty$. Finally, we obtaining a rate of convergence for $||A_n - \bar{A}||$. Under assumptions (iii)-(vi) and Lemma D of Pollard (1982) we then obtain the expansion

$$\begin{aligned}
W(A_n, F_n) = &W\left(\bar{A}, F_n\right) - n^{-1/2} Z_n' \left(v\left(A_n\right) - v\left(\bar{A}\right)\right) \\
&+ \frac{1}{2} \left(v\left(A_n\right) - v\left(\bar{A}\right)\right)' \Gamma \left(v\left(A_n\right) - v\left(\bar{A}\right)\right) \\
&+ \left(\mathcal{P}\left(A_n\right) - \mathcal{P}\left(\bar{A}\right)\right) + o_p(n^{-1/2} r_n) + o_p(r_n^2),
\end{aligned}$$

where $r_n = ||A_n - \bar{A}||$, $v(\cdot)$ is the operator that vectorizes a matrix, and $Z_n \in \mathbb{R}^{K \times p}$ is asymptotically normal with zero mean and the covariance matrix given in Lemma D of Pollard (1982). The positive definiteness of Γ together with the fact that $W(A_n, F_n) \leq W(A, F_n)$ now yields the inequality

$$\mathcal{O}_p(r_n^2) + \mathcal{O}_p(\lambda(\mathcal{P}(A_n) - \mathcal{P}(\bar{A}))) \leq \mathcal{O}_p(n^{-1/2} p^{1/2} r_n) + o_p(n^{-1/2} r_n) + o_p(r_n^2).$$

In case of the lasso, ridge or group-lasso penalty, we have $\lambda(\mathcal{P}(A_n) - \mathcal{P}(\bar{A})) = \mathcal{O}(p^{1/2}\lambda r_n)$, from which it follows that we need $\lambda = \mathcal{O}(n^{-1/2})$ in order to obtain $\sqrt{\frac{n}{p}}$-consistency. For the ℓ_0 penalty, we have $\lambda(\mathcal{P}(A_n) - \mathcal{P}(\bar{A})) = \mathcal{O}(\lambda)$ under assumption (viii). Therefore, the ℓ_0 penalty does not impose any additional restrictions on the convergence rate of λ. ■

A.6 Proof of Theorem 10

We work under the same assumptions as those of the previous section.

Proof First consider $\mathcal{P} \in \{\mathcal{P}_1, \mathcal{P}_3\}$. Take $j \in \{p_0, \dots, p\}$ and suppose that $P(\hat{A}_{\cdot,j} = 0) \not\to 1$. Based on KKT conditions, we have that if $\hat{A}_{\cdot,j} \neq 0$, we must have

$$-\frac{2}{\sqrt{n}}\underline{\boldsymbol{m}}^T(\underline{\boldsymbol{x}}_{\cdot,j} - \underline{\boldsymbol{m}}\hat{A}_{\cdot,j}) + \lambda\sqrt{n}\,\frac{\partial}{\partial m}\mathcal{P}(m)\Big|_{m=\hat{A}_{\cdot,j}} = 0$$

For the lasso penalty, we have $\left|\left|\lambda\sqrt{n}\,\frac{\partial}{\partial m}\mathcal{P}(m)\big|_{m=\hat{A}_{\cdot,j}}\right|\right| = \left|\left|\lambda\sqrt{n}\,\text{sign}\left(\hat{A}_{\cdot,j}\right)\right|\right| = \mathcal{O}(\sqrt{n}\lambda)$.
For the group-lasso penalty, we have $\left|\left|\lambda\sqrt{n}\,\frac{\partial}{\partial m}\mathcal{P}(m)\big|_{m=\hat{A}_{\cdot,j}}\right|\right| = \left|\left|\lambda\sqrt{n}\frac{\hat{A}_{\cdot,j}}{||\hat{A}_{\cdot,j}||}\right|\right| = \mathcal{O}(\lambda\sqrt{n})$.
Now as we have $\sqrt{n}\lambda \to \infty$, the last term diverges to infinity whereas the first term can be shown to be $\mathcal{O}_p(1)$ (see Sun et al., 2012), which leads to a contradiction. We thus conclude that $P(\hat{A}_{\cdot,j} = 0) \to 1$. By the assumption $n^{-1}\lambda^{-2}p \to 0$, it follows that this holds jointly for all $j = p_0, \dots, p$.
Now consider $\mathcal{P} = \mathcal{P}_0$ and take $j \in \{p_0, \dots, p\}$. have that $\hat{A}_{\cdot,j} \neq 0$ only if

$$\begin{aligned} &\frac{1}{n}||\underline{\boldsymbol{x}}_{\cdot,j}||_2^2 > \frac{1}{n}||\underline{\boldsymbol{x}}_{\cdot,j} - \underline{\boldsymbol{m}}\hat{A}_{\cdot,j}||_2^2 + \lambda \\ \Leftrightarrow &\frac{1}{n}||\underline{\boldsymbol{x}}_{\cdot,j}||_2^2 - \frac{1}{n}||\underline{\boldsymbol{x}}_{\cdot,j} - \underline{\boldsymbol{m}}\hat{A}_{\cdot,j}||_2^2 > \lambda \end{aligned}$$

Now we have that the left hand side of this equality is smaller than $\frac{1}{n}||\underline{\boldsymbol{m}}\hat{A}_{\cdot,j}||_2^2 = \frac{1}{n}\mathcal{O}(nr_n^2)$ in absolute value. This yields the inequality $\mathcal{O}(p/n) = \mathcal{O}(r_n^2) > \mathcal{O}(\lambda)$, which contradicts the assumption that $n^{-1}\lambda^{-2}p \to 0$ leading us to conclude that $P(\hat{A}_{\cdot,j} = 0) \to 1$. As $\lambda p \to 0$, this holds jointly for all $j = p_0, \dots, p$. ■

Appendix B. A Note on the Starting Value

Table 12 shows the ARI values obtained when comparing the proposed starting value with 7 random initializations and with classical K-means as a starting value on the simulated data of dimensions $n = 80$ and $p = 1000$ with $K = 4$. It is clear that the lasso, group lasso, and ℓ_0 penalty all benefit from the proposed sparse start. The ridge penalty is least affected, which is somewhat expected as it does not induce sparsity in the variables, in contrast with the other penalties. Note that this simulation was run in the same way as the simulation of Section 5.2, that is, the optimal lambda was selected based on the knowledge of the true clusters. In this way, we eliminate the effect of choosing the regularization parameter on the results of our simulation which allows us to isolate the effect of the starting value.

penalty & start	$\gamma = 0.4$	$\gamma = 0.5$	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$
lasso.random	0.07 (0.03)	0.11 (0.06)	0.2 (0.13)	0.27 (0.13)	0.33 (0.12)
ridge.random	0.08 (0.03)	0.11 (0.03)	0.14 (0.04)	0.18 (0.04)	0.22 (0.05)
glasso.random	0.07 (0.04)	0.14 (0.11)	0.26 (0.17)	0.36 (0.15)	0.47 (0.23)
HT.random	0.07 (0.03)	0.1 (0.06)	0.25 (0.16)	0.47 (0.2)	0.62 (0.22)
lasso.kmeans	0.14 (0.06)	0.3 (0.1)	0.48 (0.11)	0.66 (0.12)	0.81 (0.13)
ridge.kmeans	0.15 (0.06)	0.28 (0.08)	0.43 (0.1)	0.61 (0.12)	0.74 (0.11)
glasso.kmeans	0.15 (0.06)	0.35 (0.12)	0.62 (0.2)	0.93 (0.13)	0.98 (0.08)
HT.kmeans	0.14 (0.06)	0.32 (0.12)	0.74 (0.23)	0.97 (0.08)	0.99 (0.05)
lasso.sparse	0.15 (0.06)	0.34 (0.12)	0.71 (0.2)	0.98 (0.04)	1 (0.03)
ridge.sparse	0.15 (0.06)	0.28 (0.08)	0.44 (0.11)	0.63 (0.15)	0.77 (0.14)
glasso.sparse	0.15 (0.06)	0.35 (0.12)	0.81 (0.18)	0.99 (0.01)	1 (0)
HT.sparse	0.15 (0.06)	0.36 (0.13)	0.85 (0.19)	0.99 (0.01)	1 (0)

Table 12: ARIs of Regularized K-means variants based on 7 random starts, the classical K-means start, and the proposed 7 sparse starts on data of dimensions $n = 80$ and $p = 1000$ where $K = 4$.

Appendix C. EM-algorithm for Model-Based Clustering

The expected value of the log-likelihood function given the parameters $\underline{\theta}^{(m)} = (\underline{\theta}_1^{(m)}, \ldots, \underline{\theta}_K^{(m)})$ at iteration m of the algorithm is given by

$$Q(\underline{\theta}, \underline{\theta}^{(m)}) = \sum_{k=1}^{K} \sum_{i=1}^{n} \underline{\boldsymbol{m}}_{ik}^{(m)} \left(\log(\pi_k) + \log(f_k(x_i; \underline{\theta}_k))\right) - \lambda \sum_{j=1}^{p} \mathbf{1}_{||\underline{\boldsymbol{\mu}}_{\cdot,j}||_2 > 0}, \tag{4}$$

where $\underline{\boldsymbol{m}}_{ik}^{(m)} = \frac{\pi_k^{(m)} f_k(x_i; \underline{\theta}_k^{(m)})}{\sum_{k=1}^{K} \pi_k^{(m)} f_k(x_i; \underline{\theta}_k^{(m)})}$ is the posterior probability of x_i belonging to cluster k. The M-step now requires maximizing the quantity of Equation 4 with respect to $\underline{\theta}$. This yields the update equations

$$\begin{aligned}
\pi_k^{(m+1)} =& \frac{1}{n} \sum_{i=1}^{n} \underline{\boldsymbol{m}}_{ik}^{(m)} \\
\sigma_j^{2,(m+1)} =& \frac{1}{n} \sum_{k=1}^{K} \sum_{i=1}^{n} \underline{\boldsymbol{m}}_{ik}^{(m)} (\underline{\boldsymbol{x}}_{ij} - \underline{\boldsymbol{\mu}}_{kj}^{(m)})^2 \\
\underline{\boldsymbol{\mu}}_{\cdot,j}^{(m+1)} =& \begin{cases} \underline{\boldsymbol{\mu}}_{\cdot,j}^{*,(m+1)} & \text{if } ||\underline{\boldsymbol{x}}_{\cdot,j}||_2^2 > ||\underline{\boldsymbol{x}}_{\cdot,j} - \underline{\boldsymbol{m}}^{(m)}\ \underline{\boldsymbol{\mu}}_{\cdot,j}^{*,(m+1)}||_2^2 + 2n\lambda\sigma_j^{2,(m+1)} \\ 0 & \text{else} \end{cases}
\end{aligned}$$

where $\underline{\boldsymbol{\mu}}_{\cdot,j}^{*,(m+1)} = \frac{\sum_{i=1}^{n} \underline{\boldsymbol{m}}_{ik}^{(m+1)} \underline{x}_i}{\sum_{i=1}^{n} \underline{\boldsymbol{m}}_{ik}^{(m+1)}}$. Given a certain starting value, we can thus alternate between updating $\underline{\boldsymbol{m}}^{(m)}$ and the other parameters until convergence. Note that this is very similar in spirit to what we are doing in our adaptation of Lloyd's algorithm, with the difference that we do not have updates of the quantities σ_j^2 and π_k.

Appendix D. Additional Simulation Results

In this section, the results of the complete simulation study are shown. Rather than showing all of the 23 additional tables with simulation results, we have chosen to summarize them by taking ranks of the different methods. For a single simulation setup, a rank of 1 corresponds with the best performing method as measured by ARI. In case of ties in performance, the average rank is given to all methods with the same performance. For each value of γ, these ranks are then averaged over all 24 simulation settings and this average rank is shown in the tables. The conclusions drawn in the main text remain valid for the other simulation setups. In particular, the relative performance of the methods does not change much depending on the parameters of the simulation. The section is subdivided in three subsections, each corresponding with one simulation study discussed in the main text.

D.1 Comparison of Penalty Types

Table 13 below shows the average ranks for the simulation comparing the different penalty types. We see that the performance of the ℓ_0 penalty is close to that of the group-lasso, but more often than not it does perform slightly better. The classical K-means is the worst performing method here, with the ridge as a fairly close second. The performance of the lasso penalty somewhere in between that of the ridge and the group-lasso penalties.

	$\gamma = 0.4$	$\gamma = 0.5$	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$
classical	4.42	4.21	4.04	3.85	3.65
ridge	3.02	3.25	3.31	3.46	3.35
lasso	2.79	2.96	2.88	2.81	2.73
glasso	2.50	**2.21**	2.48	2.50	2.77
HT	**2.27**	2.38	**2.29**	**2.38**	**2.50**

Table 13: Average ranks of the performance of the different penalties in the Regularized K-means framework. A rank of 1 indicates the best performance, whereas a rank of 5 is the worst possible performance.

D.2 Comparison of Techniques for Selecting λ

Table 14 below shows the average ranks for the performance of the different techniques for selecting λ over the 24 simulation setups. We can see that the AIC performs best overall, and consistently attains the highest relative rank over different setups. Given that it is also the fastest to compute (together with BIC), we propose to use it as our technique for selecting λ.

	$\gamma = 0.4$	$\gamma = 0.5$	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$
AIC	**1.90**	**1.98**	**1.85**	**1.83**	**2.15**
BIC	2.58	2.23	2.31	1.96	2.35
stab1	2.52	2.54	2.60	2.79	2.62
stab2	2.83	3.04	2.88	3.02	2.67
stab3	2.67	2.71	2.85	2.90	2.71

Table 14: Average ranks of the performance of the different λ-selection techniques in the Regularized K-means framework. A rank of 1 indicates the best performance, whereas a rank of 5 is the worst possible performance.

D.3 Comparison with Other Regularized Clustering Methods

Table 15 below shows the average ranks for the performance of the different techniques for regularized and sparse K-means clustering over the 24 simulation setups. Note that there are some relatively easy simulation setups in which all methods perform (almost) perfectly. Those setups have quite a lot of ties in the performance measures, which means the average rank is given to these tied performances. This somewhat "shrinks" all the performances towards 2.5. We clearly see that the proposed HTK-means procedure performs the best overall. The other methods are somewhat comparable, with the regularized K-means slightly outperforming Sparse K-means which in turn slightly outperforms classical K-means when γ is not too small.

	$\gamma = 0.4$	$\gamma = 0.5$	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$
HTK-means	**1.77**	**1.90**	**1.83**	**1.85**	**1.96**
Reg K-means	2.65	2.73	2.71	2.65	2.60
Sparse K-means	3.25	2.75	2.69	2.75	2.67
K-means	2.33	2.62	2.77	2.75	2.77

Table 15: Average ranks of the performance of the different proposals for regularized and sparse K-means. A rank of 1 indicates the best performance, whereas a rank of 4 is the worst possible performance.

References

Salem Alelyani, Jiliang Tang, and Huan Liu. Feature selection for clustering: A review. *Data Clustering*, pages 29–60, 2018.

Daniel Aloise, Amit Deshpande, Pierre Hansen, and Preyas Popat. Np-hardness of euclidean sum-of-squares clustering. *Machine learning*, 75(2):245–248, 2009.

Edgar Anderson. The irises of the gaspe peninsula. *Bulletin of the American Iris Society*, 59:2–5, 1935.

Tanya Barrett, Stephen E. Wilhite, Pierre Ledoux, Carlos Evangelista, Irene F. Kim, Maxim Tomashevsky, Kimberly A. Marshall, Katherine H. Phillippy, Patti M. Sherman, Michelle Holko, Andrey Yefanov, Hyeseung Lee, Naigong Zhang, Cynthia L. Robertson, Nadezhda Serova, Sean Davis, and Alexandra Soboleva. NCBI GEO: archive for functional genomics data sets—update. *Nucleic Acids Research*, 41(D1):D991–D995, 11 2012. ISSN 0305-1048. doi: 10.1093/nar/gks1193. URL `https://doi.org/10.1093/nar/gks1193`.

Luca Becchetti, Marc Bury, Vincent Cohen-Addad, Fabrizio Grandoni, and Chris Schwiegelshohn. Oblivious dimension reduction for k-means: beyond subspaces and the Johnson-Lindenstrauss lemma. In *ACM SIGACT Symposium on Theory of Computing*, pages 1039–1050, 2019.

Shai Ben-David, Ulrike Von Luxburg, and Dávid Pál. A sober look at clustering stability. In *International Conference on Computational Learning Theory*, pages 5–19. Springer, 2006.

Asa Ben-Hur, Andre Elisseeff, and Isabelle Guyon. A stability based method for discovering structure in clustered data. In *Biocomputing*, pages 6–17. World Scientific, 2001.

Sakyajit Bhattacharya and Paul D McNicholas. A lasso-penalized bic for mixture model selection. *Advances in Data Analysis and Classification*, 8(1):45–61, 2014.

Christos Boutsidis, Petros Drineas, and Michael W Mahoney. Unsupervised feature selection for the k-means clustering problem. In *Advances in Neural Information Processing Systems*, pages 153–161, 2009.

Christos Boutsidis, Anastasios Zouzias, Michael W Mahoney, and Petros Drineas. Randomized dimensionality reduction for k-means clustering. *IEEE Transactions on Information Theory*, 61(2):1045–1062, 2014.

Hector Corrada Bravo, Matthew McCall, and Rafael A. Irizarry. *antiProfilesData: Normal colon and cancer preprocessed affy data for antiProfile building.*, 2020. R package version 1.24.0.

Hyunkeun Cho and Annie Qu. Model selection for correlated data with diverging number of parameters. *Statistica Sinica*, pages 901–927, 2013.

Michael B Cohen, Sam Elder, Cameron Musco, Christopher Musco, and Madalina Persu. Dimensionality reduction for k-means clustering and low rank approximation. In *ACM Symposium on Theory of Computing*, pages 163–172, 2015.

Sanjoy Dasgupta. The hardness of k-means clustering. Technical report, CS2008-0916, University of California, 2008.

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

Jianqing Fan and Heng Peng. Nonconcave penalized likelihood with a diverging number of parameters. *The annals of statistics*, 32(3):928–961, 2004.

Yixin Fang and Junhui Wang. Selection of the number of clusters via the bootstrap method. *Computational Statistics & Data Analysis*, 56(3):468 – 477, 2012. ISSN 0167-9473. doi: https://doi.org/10.1016/j.csda.2011.09.003. URL http://www.sciencedirect.com/science/article/pii/S0167947311003215.

Ronald A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2):179–188, 1936. doi: 10.1111/j.1469-1809.1936.tb02137.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1469-1809.1936.tb02137.x.

Bernhard Flury and Hans Riedwyl. *Multivariate Statistics: A Practical Approach*. Statistics texts. Springer Netherlands, 1988. ISBN 9780412300301. URL https://books.google.be/books?id=HiuIIepPOeEC.

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

Jonas Haslbeck and Dirk U Wulff. Estimating the number of clusters via a corrected clustering instability. *Computational Statistics*, 35(4):1879–1894, 2020.

David P. Hofmeyr. Degrees of freedom and model selection for k-means clustering. *Computational Statistics & Data Analysis*, 149:106974, 2020. ISSN 0167-9473. doi: https://doi.org/10.1016/j.csda.2020.106974. URL http://www.sciencedirect.com/science/article/pii/S0167947320300657.

Peter J Huber. Robust regression: asymptotics, conjectures and monte carlo. *The annals of statistics*, pages 799–821, 1973.

Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of classification*, 2(1): 193–218, 1985.

Anil K Jain. Data clustering: 50 years beyond k-means. *Pattern recognition letters*, 31(8): 651–666, 2010.

Yongdai Kim, Hosik Choi, and Hee-Seok Oh. Smoothly clipped absolute deviation on high dimensions. *Journal of the American Statistical Association*, 103(484):1665–1673, 2008.

Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982.

James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Berkeley symposium on mathematical statistics and probability*, volume 1(14), pages 281–297. Oakland, CA, USA, 1967.

Konstantin Makarychev, Yury Makarychev, and Ilya Razenshteyn. Performance of Johnson-Lindenstrauss transform for k-means and k-medians clustering. In *ACM SIGACT Symposium on Theory of Computing*, pages 1027–1038, 2019.

Cathy Maugis, Gilles Celeux, and Marie-Laure Martin-Magniette. Variable selection for clustering with gaussian mixture models. *Biometrics*, 65(3):701–709, 2009.

Michal Moshkovitz, Sanjoy Dasgupta, Cyrus Rashtchian, and Nave Frost. Explainable k-means and k-medians clustering. In *International Conference on Machine Learning*, pages 7055–7065. PMLR, 2020.

Wei Pan and Xiaotong Shen. Penalized model-based clustering with application to variable selection. *Journal of machine learning research*, 8(5), 2007.

David Pollard. Strong consistency of k-means clustering. *The Annals of Statistics*, 9(1): 135–140, 01 1981. doi: 10.1214/aos/1176345339. URL `https://doi.org/10.1214/aos/1176345339`.

David Pollard. A central limit theorem for k-means clustering. *The Annals of Probability*, 10(4):919–926, 1982. ISSN 00911798. URL `http://www.jstor.org/stable/2243547`.

Adrian E Raftery and Nema Dean. Variable selection for model-based clustering. *Journal of the American Statistical Association*, 101(473):168–178, 2006.

Stephen A Ramsey, Sandy L Klemm, Daniel E Zak, Kathleen A Kennedy, Vesteinn Thorsson, Bin Li, Mark Gilchrist, Elizabeth S Gold, Carrie D Johnson, Vladimir Litvak, et al. Uncovering a macrophage transcriptional program by integrating evidence from motif scanning and expression dynamics. *PLoS Comput Biol*, 4(3):e1000021, 2008.

William M Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850, 1971.

Jakob Raymaekers and Ruben Zamar. *clusterHD: Tools for Clustering High-Dimensional Data*, 2022. R package version 1.0.0.

Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

Luca Scrucca, Michael Fop, T. Brendan Murphy, and Adrian E. Raftery. mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. *The R Journal*, 8(1):289–317, 2016. URL `https://doi.org/10.32614/RJ-2016-021`.

Hugo Steinhaus. Sur la division des corps matériels en parties. *Bulletin de l'Académie Polonaise des Sciences, Classe III*, 4(12):801–804, 1956.

Wei Sun, Junhui Wang, and Yixin Fang. Regularized k-means clustering of high-dimensional data and its asymptotic consistency. *Electron. J. Statist.*, 6:148–167, 2012. doi: 10.1214/12-EJS668. URL https://doi.org/10.1214/12-EJS668.

Robert Tibshirani, Guenther Walther, and Trevor Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423, 2001.

Junhui Wang. Consistent selection of the number of clusters via crossvalidation. *Biometrika*, 97(4):893–904, 2010. ISSN 00063444. URL http://www.jstor.org/stable/29777144.

Sijian Wang and Ji Zhu. Variable selection for model-based high-dimensional clustering and its application to microarray data. *Biometrics*, 64(2):440–448, 2008.

Daniela M. Witten and Robert Tibshirani. A framework for feature selection in clustering. *Journal of the American Statistical Association*, 105(490):713–726, 2010. doi: 10.1198/jasa.2010.tm09415. PMID: 20811510.

Daniela M. Witten and Robert Tibshirani. *sparcl: Perform Sparse Hierarchical Clustering and Sparse K-Means Clustering*, 2018. URL https://CRAN.R-project.org/package=sparcl. R package version 1.0.4.

Hui Zou. The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006. doi: 10.1198/016214506000000735.