On the price of explainability for some clustering problems

Eduardo Laber [1] **Lucas Murtinho** [1]

Abstract

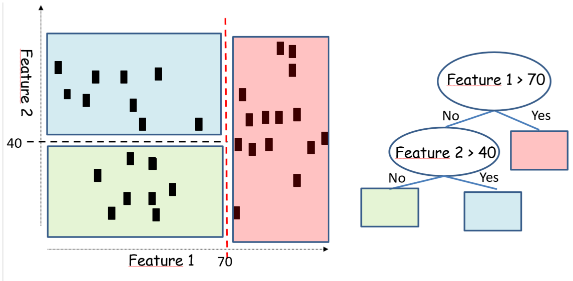
The price of explainability for a clustering task can be defined as the unavoidable loss, in terms of the objective function, if we force the final partition to be explainable. Here, we study this price for the following clustering problems: k-means, k-medians, k-centers and maximum-spacing. We provide upper and lower bounds for a natural model where explainability is achieved via decision trees. For the k-means and k-medians problems our upper bounds improve those obtained by [Dasgupta et. al, ICML 20] for low dimensions. Another contribution is a simple and efficient algorithm for building explainable clusterings for the k-means problem. We provide empirical evidence that its performance is better than the current state of the art for decision-tree based explainable clustering.

1. Introduction

Machine learning models and algorithms have been used in a number of systems that take decisions that affect our lives. Thus, explainable methods are desirable so that people are able to have a better understanding of their behavior, which allows for comfortable use of these systems or, eventually, the questioning of their applicability.

Although most of the work on the field of explainable machine learning has been focusing on supervised learning (Ribeiro et al., 2016; Lundberg & Lee, 2017; Vidal & Schiffer, 2020), there has recently been some effort to devise explainable methods for unsupervised learning tasks, in particular, for clustering (Dasgupta et al., 2020b; Bertsimas et al., 2020). We investigate the framework discussed by (Dasgupta et al., 2020b), where an explainable clustering is given by a partition, induced by the leaves of a decision tree, that optimizes some predefined objective function.

Figure 1 shows a clustering with three groups induced by a decision tree with 3 leaves. As an example, the blue cluster can be explained as the set of points that satisfy `Feature 1` ≤ 70 and `Feature 2` > 40. Simple explanations as this one are usually not available for the partitions produced by popular methods such as the Lloyd's algorithm for the k-means problem.



In order to achieve explainability, one may be forced to accept some loss in terms of the quality of the chosen objective function (e.g. sum of squared distances). In this sense, explainability has its price. (Dasgupta et al., 2020b) presents theoretical bounds on this price for the k-medians and the k-means objective functions.

Here, we expand on their work by presenting new bounds for these objectives and also providing nearly tight bounds for two other goals that arise in relevant clustering problems, namely, the k-centers and the maximum-spacing problems. We note that the objective for the latter is the one optimized by the widely known `Single-Linkage` method, employed for hierarchical clustering. We also give a more practice-oriented contribution by devising and evaluating a simple and efficient algorithm for building explainable clusterings for the k-means problem.

1.1. Problem definition

Let $\mathcal{X}$ be a set of n points in $\mathbb{R}^d$. We say that a decision tree is *standard* if each internal node v is associated with a test (cut), specified by a coordinate $i_v \in [d]$ and a real value θ_v, that partitions the points in $\mathcal{X}$ that reach v into two sets: those having the coordinate i_v smaller than or equal to θ_v and those having it larger than θ_v. The leaves of a standard decision tree induce a partition of $\mathbb{R}^d$ into axis-aligned boxes and, naturally, a partition of $\mathcal{X}$ into clusters.

[*] Equal contribution [1] Department of Computer Science, PUC-Rio, Brazil. Correspondence to: Eduardo Laber <eduardo.laber1@gmail.com>.

Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).

Let $k \geq 2$ be an integer. The clustering problems considered here consist of finding a partition of $\mathcal{X}$ into k groups, among those that can be induced by a standard decision tree with k leaves, that optimizes a given objective function. For k-means, k-medians and k-centers, in addition to the partition, a representative $\mu(C) \in \mathbb{R}^d$ for each group C must also be output.

For the k-means problem the objective (cost function) to be minimized is the Sum of the Squared Euclidean Distances (SSED) between each point $\mathbf{x} \in \mathcal{X}$ and the representative of the cluster where $\mathbf{x}$ lies. Mathematically, the cost (SSED) of a partition $\mathcal{C} = (C_1, \ldots, C_k)$ for $\mathcal{X}$ is given by

$$cost(\mathcal{C}) = \sum_{i=1}^{k} \sum_{\mathbf{x} \in C_i} ||\mathbf{x} - \mu(C_i)||_2^2.$$

The k-medians and the k-centers problems are also minimization problems. For the former, the cost of a partition $\mathcal{C} = (C_1, \ldots, C_k)$ is given by

$$cost(\mathcal{C}) = \sum_{i=1}^{k} \sum_{\mathbf{x} \in C_i} ||\mathbf{x} - \mu(C_i)||_1,$$

while for the latter it is given by

$$cost(\mathcal{C}) = \max_{i=1,\ldots,k} \max_{\mathbf{x} \in C_i} \{||\mathbf{x} - \mu(C_i)||_2\}.$$

The maximum-spacing problem is a maximization problem for which the objective to be maximized is the spacing $sp(\mathcal{C})$ of a partition $\mathcal{C}$, defined as

$$sp(\mathcal{C}) = \min\{||\mathbf{x}-\mathbf{y}||_2 : \mathbf{x} \text{ and } \mathbf{y} \text{ lie in distinct groups of } \mathcal{C}\}$$

We note that an optimal solution of the unrestricted version of any of these problems, in which the decision tree constraint is not enforced, might be a partition that is hard to explain in terms of the input features. Thus, the motivation for using decision trees.

Along the lines of (Dasgupta et al., 2020b), we define the price of explainability $\rho(\mathcal{P})$ for a clustering problem $\mathcal{P}$, with a minimization objective function, as

$$\rho(\mathcal{P}) = \max_I \left\{ \frac{OPT_{exp}(I)}{OPT_{unr}(I)} \right\},$$

where I runs over all instances of $\mathcal{P}$; $OPT_{exp}(I)$ is the cost of an optimal explainable clustering (via standard decision trees) for instance I and $OPT_{unr}(I)$ is the cost of an optimal unrestricted clustering for I. If $\mathcal{P}$ has a maximization objective function, then $\rho(\mathcal{P})$ is defined as

$$\rho(\mathcal{P}) = \max_I \left\{ \frac{OPT_{unr}(I)}{OPT_{exp}(I)} \right\}.$$

1.2. Our contributions

We provide bounds on the price of explainability as a function of the parameters k, d and n for the aforementioned objective functions. These objectives cover a spectrum that includes both intra- and inter-clustering criteria as well as worst-case and average-case measures.

First, we address the k-centers problem. We show that

$$\rho(k\text{-centers}) \in \begin{cases} \Omega(k^{1-1/d}), & \text{if } d \leq \frac{\ln k}{\ln \ln k} \\ \Omega\left(\sqrt{d} \cdot \frac{k \cdot \sqrt{\ln \ln k}}{\ln^{1.5} k}\right), & \text{otherwise} \end{cases}$$

and that $\rho(k\text{-centers})$ is $O(\sqrt{d}k^{1-1/d})$. Our bounds are tight, up to constant factors, when d is a constant. For an arbitrary d, there is only a polylogarithmic gap in k between the upper and the lower bounds. The magnitude of this gap is exponentially smaller than that of these bounds.

For the k-medians it is known that the price of explainability is $O(k)$ and $\Omega(\log k)$ (Dasgupta et al., 2020b). We contribute to the state of the art by showing that $O(d \log k)$ is also an upper bound – an exponential improvement for constant dimensions. The upper bound follows from an interesting connection with the literature of binary searching in the presence of non-uniform testing costs (Charikar et al., 2002; Laber et al., 2002).

For the k-means problem, we also improve, for low dimensions, the $O(k^2)$ bound from (Dasgupta et al., 2020b) since we prove that $\rho(k\text{-means})$ is $O(kd \log k)$. Still, for the k-means problem, we also give a more practice-oriented contribution by devising and evaluating a simple and efficient greedy algorithm. Our method outperformed the `IMM` method from (Dasgupta et al., 2020b) on an empirical study involving 10 real datasets. It should be noticed that `IMM` is a strong baseline since it got the best results against 5 other competitors on the same datasets according to (Dasgupta et al., 2020a; Frost et al., 2020).

Finally, for maximum-spacing we provide a tight bound by showing that the price of explainability is $\Theta(n - k)$. The lower bound is particularly interesting since it shows that this objective function is bad for guiding explainable clustering, losing much more than the other considered objectives in the worst-case.

To derive our upper bounds, we analyze polynomial-time algorithms that start with an optimal k-clustering and transform it into an explainable one. The unrestricted versions of all the problems considered here, except for the maximum-spacing problem, are NP-Hard (Megiddo & Supowit, 1984; Aloise et al., 2009). However, all of them admit polynomial-time algorithms with constant approximation (Williamson & Shmoys, 2011; Kanungo et al., 2004) and, hence, if we start with the partitions given by them, instead of the optimal ones, we obtain efficient algorithms with provable approximation guarantees. These guarantees are exactly the upper bounds that we prove on the price of explainability. Due to space constraints, most of the proofs can be found in the supplementary material.

We believe that our results are helpful for the construction of explainable clustering solutions as well as for guiding the choice of an objective function when explainability is required.

1.3. Related work

Our research is inspired by the recent work of (Dasgupta et al., 2020b), where they propose an algorithm, namely IMM, for building explainable clusterings, via standard decision trees, for both the k-means and the k-medians problems. At each node IMM selects the cut that minimizes the number of points separated from their representatives in a reference clustering. Our approach for these problems, while similar, uses a significantly different strategy to build the final decision tree, based on trees that look at a single dimension of the data. Moreover, as mentioned before, our algorithms provide better upper bounds for low dimensions.

Decision trees have long been associated to hierarchical agglomerative clustering (HAC), which produces a hierarchy of clusters that is usually represented by a dendrogram. Examples of models that explicitly use decision trees for HAC include (Fisher, 1987; Chavent et al., 1999; Blockeel et al., 2000; Basak & Krishnapuram, 2005). To our knowledge, the use of decision trees for non-hierarchical clustering was first suggested in (Liu et al., 2000), in which a standard classification tree is used to identify dense and sparse regions of data. In (Fraiman et al., 2013), unsupervised binary trees are also used to create interpretable clusters. More recently, an approach was presented in (Bertsimas et al., 2020) using optimal classification trees (Bertsimas & Dunn, 2017), which are built in a single step by solving a mixed-integer optimization problem. For numerical databases, (Loyola-González et al., 2020) presents a decision approach that decides on a split based on both the compactness of clusters and the separation between them.

The regions of space defined by decision-tree clustering will be hyper-rectangles (some of them may also be half-spaces if the overall region of interest is unbounded). Other approaches towards building hyper-rectangular clusters can be found in (Pelleg & Moore, 2001), with a generative model, and (Chen et al., 2016), with a discriminative one. Both models allow for probabilistic (soft) clustering, and (Chen et al., 2016) allows for incorporating previous knowledge to the model, but neither one guarantees that the resulting clusters can be represented by decision trees.

The main reason for using a (short) decision tree to build clusters is that the results of such algorithms are easily interpretable. Other avenues towards interpretable clustering have been explored in recent years. The technique presented in (Plant & Böhm, 2011) is based on the information-theoretic concept of minimum description length. In (Saisubramanian et al., 2020), a tunable parameter (the fraction of elements in a cluster that share the same feature value) leverages the tradeoff between clustering performance and interpretability. The same tradeoff is explored in (Frost et al., 2020) by relaxing the requirement from (Dasgupta et al., 2020b) that the explainable clustering should be induced by a tree with no more than k leafs. In (Horel et al., 2020), a feature selection model from (Horel & Giesecke, 2019) is used for clustering interpretation in the field of wealth management compliance. (Kauffmann et al., 2019) uses a two-step approach, rewriting k-means clustering models as neural networks and applying to these networks techniques for interpreting supervised learning models. More information regarding explainable clustering may be found in (Chen, 2018; Baralis et al.).

Of all the works mentioned in this section, only (Dasgupta et al., 2020b) presents approximation guarantees with respect to the optimal unrestricted (i.e., potentially uninterpretable) solution. Two algorithms from (Saisubramanian et al., 2020) also have an approximation guarantee, but with respect to the optimal restricted (interpretable) solution, and the definition of interpretability in that work is quite different than ours (interpretable clusters are therein defined as those in which a given proportion of points share the same value for a predefined feature of interest).

2. On the price of explainability for the k-centers problem

In this section we address the k-centers problem. We first present a lower bound by constructing an instance for which the price of explainability is high.

2.1. Lower bound

Let $p \leq \min\{d, \log_3 k\}$ be a positive integer whose exact value will be defined later in the analysis and let b be the largest integer for which $b^p \leq k$. Note that $b \geq 3$. Moreover, let $k' = b^p$.

Our instance I has $k + k' \cdot 2d$ points. We first discuss how to construct the k points, referred as centers, that will be set as representatives in an unrestricted k-clustering for I that has a low cost. The first k' centers will be obtained from the representation of the numbers $0, \ldots, k'-1$ in base b while the remaining $k - k'$ centers will be located sufficiently far from the others so that they will be isolated in the low-cost k-clustering for I. Let $\mathbf{c}^0, \ldots, \mathbf{c}^{k'-1}$ be the first k' centers.

For a number $i \in [k'-1]$ let $(i_{p-1}, \ldots, i_0)_b$ be its representation in base b. For $j \in [d]$, the value of the j-th component of center $\mathbf{c}^i$ is obtained by applying $(j-1)$ times a circular shift on $(i_{p-1},\dots,i_0)_b$. The values of the remaining $d-p$ components of $\mathbf{c}^i$ are obtained by copying the p first values d/p times so that $c^i_j = c^i_{j'}$ if $(j-j') \mod p = 0$.

As an example, if $b=3$, $p=3$ and $d=9$ then $\mathbf{c}^{14} = (14,22,16,14,22,16,14,22,16)$. In fact, since $14 = (1,1,2)_3$ we have that $c^{14}_1 = (1,1,2)_3 = 14$; $c^{14}_2 = (2,1,1)_3 = 22$ and $c^{14}_3 = (1,2,1)_3 = 16$. The values of $c^{14}_4,\dots,c^{14}_9$ are obtained by repeating the first 3 values.

The following observation is useful for our analysis.

Fact 1. *For every $\ell \in [p]$, the values of the ℓ-th coordinate of the k' first centers are a permutation of the integers $0,\dots,k'-1$.*

The remaining $k-k'$ centers, as mentioned above, should be far from each other and also far away from the k' first centers. We can achieve that by setting $\mathbf{c}^i = k^i\mathbf{1}$ for all $i > k'-1$, where $\mathbf{1}$ is the unit vector in $\mathbb{R}^d$.

The next lemma gives a lower bound on the distance between any two centers.

Lemma 1. *For any two centers $\mathbf{c}^i$ and $\mathbf{c}^j$,*

$$||\mathbf{c}^i - \mathbf{c}^j||_2 \geq \sqrt{\lfloor d/p \rfloor} \cdot (b^{p-1}/2).$$

Proof. If one of the two centers is not among the k' first centers the result clearly holds. Thus, we assume that $i,j \leq k'-1$.

It is enough to show that there is $\ell \in [p]$ for which $|c^i_\ell - c^j_\ell| \geq b^{p-1}/2$. In fact, if this inequality holds for some ℓ then $|c^i_{\ell'} - c^j_{\ell'}| \geq b^{p-1}/2$ for each ℓ' that is congruent to ℓ modulo p. Since there are $\lfloor d/p \rfloor$ of them, due to our construction, we get the desired bound.

Let $i = (i_{p-1},\dots,i_0)_b$ and $j = (j_{p-1},\dots,j_0)_b$ be the representations of i and j in base b, respectively. Let f be such that $|i_f - j_f|$ is maximum.

Thus, the difference between $\mathbf{c}^i$ and $\mathbf{c}^j$ in the coordinate $[(f+1) \mod p]+1$ is at least

$$|i_f - j_f| \cdot \left(b^{p-1} - \sum_{g=0}^{p-2} b^g \right) \geq b^{p-1}/2,$$

where the last inequality holds because $|i_f - j_f| \geq 1$ and $b \geq 3$. □

Now, we define the remaining points of instance I.

For each of the first k' centers we create $2d$ associated points: $\mathbf{x}^{i,1},\dots,\mathbf{x}^{i,2d}$. For $j=1,\dots,d$, the point $\mathbf{x}^{i,2j-1}$ is identical to $\mathbf{c}^i$ in all coordinates but on the j-th one, in which its value is $c^i_j - 3/4$. Similarly, the point $\mathbf{x}^{i,2j}$ is identical to $\mathbf{c}^i$ in all coordinates but in the j-th one, in which its value is $c^i_j + 3/4$. By considering the k-clustering for I where the k representatives are the k centers $\mathbf{c}^0,\dots,\mathbf{c}^{k-1}$ and each point $\mathbf{x}^{i,j}$ lies in the group of $\mathbf{c}^i$, we obtain the following proposition.

Proposition 1. *There exists an unrestricted k-clustering for instance I with cost $3/4$.*

Now we analyse the cost of an optimal explainable clustering for I. The following proposition is a simple consequence of Fact 1.

Proposition 2. *Let (j,θ) be a cut that separates at least two points from the set A that includes the k' first centers and its associated $k' \cdot 2d$ points. Then, (j,θ) separates one point from its associated center.*

Lemma 2. *Any explainable k-clustering for instance I has cost at least $\sqrt{\lfloor d/p \rfloor} \cdot (b^{p-1}/4) - 3/8$.*

Proof. Let $\mathcal{C}$ be an explainable k-clustering for instance I. It is enough to show that there is a cluster $C \in \mathcal{C}$ that contains two points, say $\mathbf{x}$ and $\mathbf{y}$, for which

$$||\mathbf{x} - \mathbf{y}||_2 \geq \sqrt{\lfloor d/p \rfloor} \cdot (b^{p-1}/2) - 3/4.$$

In fact, in this case, due to the triangle inequality, for any choice of the representative for C, either $\mathbf{x}$ or $\mathbf{y}$ will be at distance at least $\sqrt{\lfloor d/p \rfloor} \cdot (b^{p-1}/4) - 3/8$ from it.

If two centers lie in the same cluster of $\mathcal{C}$ then it follows from Lemma 1 that their distance is at least $\sqrt{\lfloor d/p \rfloor} \cdot (b^{p-1}/2)$.

On the other hand, if every center lies on a different cluster in $\mathcal{C}$ then let $\mathbf{x}$ be the point that was separated from its center, say $\mathbf{c}^i$, by a cut that satisfies the condition of Proposition 2. Then, $\mathbf{x}$ lies in the same cluster of $\mathbf{c}^j$, for some $j \neq i$. From the triangle inequality we have that

$$||\mathbf{c}^i - \mathbf{c}^j||_2 \leq ||\mathbf{c}^i - \mathbf{x}||_2 + ||\mathbf{c}^j - \mathbf{x}||_2.$$

Hence, $||\mathbf{c}^j - \mathbf{x}||_2 \geq \sqrt{\lfloor d/p \rfloor} \cdot (b^{p-1}/2) - 3/4$. □

By putting together Proposition 1 and Lemma 2 and, then, optimizing the value of p we obtain the following theorem.

Theorem 1. *The price of explainability for the k-centers problem satisfies*

$$\rho(k\text{-}center) \in \begin{cases} \Omega(k^{1-1/d}), & \text{if } d \leq \frac{\ln k}{\ln \ln k} \\ \Omega\left(\sqrt{d} \cdot \frac{k \cdot \sqrt{\ln \ln k}}{\ln^{1.5} k}\right), & \text{otherwise.} \end{cases}$$

2.2. Upper bound

In this section we show that the price of explainability for the k-center problem is $O\left(\sqrt{d} k^{\frac{d-1}{d}}\right)$. Note that, for constant d, the upper bound matches the lower bound given by Theorem 1.

To obtain the upper bound we analyze the cost of the explainable clustering induced by the decision tree built by the algorithm presented in Algorithm 1.

The algorithm has access to the set of representatives of an optimal k-clustering $\mathcal{C}^*$ for $\mathcal{X}$. These representatives are used as *reference centers* for the points in $\mathcal{X}$, that is, the reference center of a point $\mathbf{x}$ is the representative of $\mathbf{x}$'s group in $\mathcal{C}^*$.

Let $\mathcal{X}'$ and S be, respectively, the subset of points in $\mathcal{X}$ and the set of reference centers that reach a given node u. To split u, as long as it is possible, the algorithm applies an axis-aligned cut that does not separate any point $\mathbf{x} \in \mathcal{X}'$ from its reference center. This type of cut is referred as a *clean cut* with respect to $(\mathcal{X}', S)$. When there is no such cut available for u, the algorithm partitions the bounding box of the points in $\mathcal{X}' \cup S$ into $\lfloor |S|^{1/d} \rfloor^d$ axis-aligned boxes of the same dimensions by using a decision tree that emulates a grid. By the bounding box of $\mathcal{X}' \cup S$ we mean the smallest box (hyper-rectangle) with axis-aligned sides that includes the points in $\mathcal{X}' \cup S$.

Algorithm 1 Ex-kCenter($\mathcal{X}'$: set of points)

```
S ← reference centers of the points in $\mathcal{X}'$
if $|S| = 1$ then
  Return $\mathcal{X}'$ and the single reference center in $S$
else
  if there exists a clean cut w.r.t. $(\mathcal{X}', S)$ then
    $(\mathcal{X}'_L, \mathcal{X}'_R)$ ← partition induced by the clean cut
    Create a node $u$
    $u$.LeftChild ←Ex-kCenter($\mathcal{X}'_L$)
    $u$.RightChild ←Ex-kCenter($\mathcal{X}'_R$)
    Return the tree rooted at $u$
  else
    $H$ ← bounding box for $\mathcal{X}' \cup S$
    $D^u$ ← decision tree that partitions $H$ into $\lfloor |S|^{1/d} \rfloor^d$ identical axis-aligned boxes
    Return $D^u$ as well as an arbitrarily chosen representative for each of its leaves
  end if
end if
```

Theorem 2. *The price of explainability for k-centers is $O\left(\sqrt{d}k^{1-1/d}\right)$.*

Proof. We argue that for each leaf ℓ of the tree $\mathcal{D}$ built by Ex-kCenter($\mathcal{X}$), the maximum distance between a point in ℓ and its representative is $OPT\sqrt{d}k^{1-1/d}$, where OPT is the cost of the optimal unrestricted clustering.

We split the proof into two cases. The first case addresses the scenario in which only clean cuts are used in the path from the root of $\mathcal{D}$ to the leaf ℓ. The second case addresses the remaining scenarios.

Case 1. In this case all points that reach ℓ lie in the same cluster of the optimal unrestricted k-clustering $\mathcal{C}^*$. Thus, the maximum distance from a point in ℓ to the single reference center in S is upper bounded by OPT.

Case 2. Let u be the first node in the path from the root to ℓ for which a clean cut is not available. Moreover, let $\mathcal{X}^u$ be the set of points that reach u and let $s = |S|$, that is, the number of reference centers that reach u. In this case the algorithm splits the bounding box for $\mathcal{X}^u \cup S$ into boxes of dimensions

$$\frac{L_1}{\lfloor s^{1/d} \rfloor} \times \cdots \times \frac{L_d}{\lfloor s^{1/d} \rfloor},$$

where L_i is the difference between the maximum and minimum values of the i-th coordinate among points in $\mathcal{X}^u \cup S$.

The maximum distance between a point in ℓ and its representative can be upper bounded by the length of the diagonal of the axis-aligned box corresponding to ℓ. Let $m \in [d]$ be such that $L_m = \max\{L_1, \ldots, L_d\}$. Then, the length of the diagonal is upper bounded by $L_m\sqrt{d}/\lfloor s^{1/d} \rfloor \leq 2L_m\sqrt{d}/s^{1/d}$.

Thus, it suffices to show that $OPT \geq L_m/(2s)$. Let $\mathbf{c}^1, \ldots, \mathbf{c}^s$ be the s reference centers that reach node u. In addition, let $\mathbf{x}^j$ be a point in $\mathcal{X}^u$ with reference center $\mathbf{c}^j$ and such that $|x^j_m - c^j_m|$ is maximum, among the points in $\mathcal{X}^u$ with reference center $\mathbf{c}^j$. Then, we must have

$$\sum_{j=1}^{s} 2|x^j_m - c^j_m| \geq L_m,$$

for otherwise there would be a clean cut (m, θ), with $\theta \in [a, b]$, where $a = \min\{y_m | \mathbf{y} \in \mathcal{X}^u \cup S\}$ and $b = \max\{y_m | \mathbf{y} \in \mathcal{X}^u \cup S\}$. Hence, for some point $\mathbf{x}^j$, $|x^j_m - c^j_m| \geq L_m/(2s)$. Since $OPT \geq |x^j_m - c^j_m|$ we get that $OPT \geq L_m/(2s)$. □

3. Improved bounds on k-medians for low dimensions

We show that the price of explainability for k-medians is $O(d \log k)$, which improves the bound from (Dasgupta et al., 2020b) when $d = o(k/\log k)$.

As in the previous section we use an optimal unrestricted k-clustering $\mathcal{C}^*$ for $\mathcal{X}$ as a guide for building an explainable clustering. Again, by the reference center of a point $\mathbf{x} \in \mathcal{X}$ we mean its representative in $\mathcal{C}^*$.

We need some additional notation. For a decision tree $\mathcal{D}$ and a node $u \in \mathcal{D}$, let $diam(u)$ be the d-dimensional vector whose i-th coordinate $diam(u)_i$ is given by the difference between the maximum and the minimum values of coordinate i among the reference centers that reach u. Let t_u be the number of points that reach u and are separated from their reference centers by the cut employed in u. Note that a point $\mathbf{x} \in \mathcal{X}$ can only contribute to t_u if both $\mathbf{x}$ and its reference center reach u. Finally, we use OPT to denote the cost of the optimal unrestricted clustering $\mathcal{C}^*$.

The following lemma from (Dasgupta et al., 2020b), expressed in our notation, will be useful.

Lemma 3. *(Dasgupta et al., 2020b) Let $\mathcal{C}^*$ be an optimal unrestricted k-clustering for $\mathcal{X}$ and let $\mathcal{D}$ be a decision tree for $\mathcal{X}$ in which each representative of $\mathcal{C}^*$ lies in a distinct leaf. Then, the clustering $\mathcal{C}$ induced by $\mathcal{D}$ satisfies*

$$cost(\mathcal{C}) \leq OPT + \sum_{u \in \mathcal{D}} t_u ||diam(u)||_1. \qquad (1)$$

In order to obtain a low-cost explainable clustering we focus on finding a decision tree $\mathcal{D}$ for which the rightmost term of the above inequality is small. This is the approach taken by IMM (Dasgupta et al., 2020b), a greedy strategy that at each node u selects the cut that yields the minimum possible value for t_u.

Although we follow the same approach, our strategy for building the tree is significantly different. In order to explain it, we first rewrite the rightmost term of (1):

$$\sum_{u \in \mathcal{D}} t_u ||diam(u)||_1 = \sum_{i=1}^{d} \sum_{u \in \mathcal{D}} t_u diam(u)_i. \qquad (2)$$

Motivated by Lemma 3 and the above identity, our strategy constructs d decision trees $\mathcal{D}_1, \ldots, \mathcal{D}_d$, where $\mathcal{D}_i$ is built with the aim of minimizing

$$\sum_{u \in \mathcal{D}} t_u diam(u)_i, \qquad (3)$$

ignoring the impact on the coordinates $j \neq i$.

Next, it constructs a decision tree $\mathcal{D}$ for $\mathcal{X}$ by picking nodes from these d trees. More precisely, to split a node u of $\mathcal{D}$ the strategy first selects a coordinate $i \in [d]$ for which $diam(u)_i$ is maximum. Next, it applies the cut that is associated with the node in $\mathcal{D}_i$ which is the least common ancestor (LCA) of the set of reference centers that reach u.

In the pseudo-code presented in Algorithm 2, S' is a subset of the set S of representatives of $\mathcal{C}^*$. Moreover, $\mathcal{X}'$ is a subset of the points in $\mathcal{X}$. The procedure is called, initially, with $\mathcal{X}' = \mathcal{X}$ and $S' = S$.

Algorithm 2 BuildTree($\mathcal{X}' \cup S'$)

```
Create a node $u$ and associate it with $\mathcal{X}' \cup S'$
if $|S'| = 1$ then
  Return the leaf $u$
else
  Select $i \in [d]$ for which $diam(u)_i$ is maximum.
  $v \leftarrow$ node in $\mathcal{D}_i$ which is the LCA of the centers in $S'$
  Split $\mathcal{X}' \cup S'$ into $\mathcal{X}'_L \cup S'_L$ and $\mathcal{X}'_R \cup S'_R$ using the cut associated with $v$.
  $u$.LeftChild $\rightarrow$ BuildTree($\mathcal{X}'_L \cup S'_L$)
  $u$.RightChild $\rightarrow$ BuildTree($\mathcal{X}'_R \cup S'_R$)
  Return the decision tree rooted at $u$
end if
```

To fully specify the algorithm we need to explain how the decision trees $\mathcal{D}_i$ are built. Let $\mathbf{c}^1, \ldots, \mathbf{c}^k$ be the reference centers sorted by coordinate i, that is, $c_i^j < c_i^{j+1}$ for $j = 1, \ldots, k-1$. Moreover, let (i, θ^j) be the cut that separates the points in $\mathcal{X}$ with the i-th coordinate smaller than or equal to $\theta^j = (c_i^j + c_i^{j+1})/2$ from the remaining ones.

For $1 \leq a \leq b \leq k$, let $\mathcal{F}_{a,b}$ be the family of binary decision trees with $(b-a)$ internal nodes and $b-a+1$ leaves defined as follows:

(i) if $a = b$, then $\mathcal{F}_{a,b}$ has a single tree and this tree contains only one node.

(ii) if $a < b$, then $\mathcal{F}_{a,b}$ consists of all the decision trees $\mathcal{D}'$ with the following structure: the root of $\mathcal{D}'$ is identified by a number $j \in \{a, \ldots, b-1\}$ and associated with the cut (i, θ^j); one child of the root of $\mathcal{D}'$ is a tree in the family $\mathcal{F}_{a,j}$ while the other is a tree in $\mathcal{F}_{j+1,b}$.

For our analysis, in the next sections, it will be convenient to view $\mathcal{F}_{a,b}$ as the family of binary search trees for the numbers in the set $\{a, \ldots, b-1\}$.

Let T_j be the number of points in $\mathcal{X}$ that are separated from their centers by cut (i, θ^j). For every tree $\mathcal{D}' \in \mathcal{F}_{a,b}$ we define $UB_i(\mathcal{D}')$ as

$$UB_i(\mathcal{D}') = \sum_{j=a}^{b-1} T_j \cdot diam(j)_i,$$

where $diam(j)$ is the diameter of the node identified by j in $\mathcal{D}'$.

The tree $\mathcal{D}_i$ is, then, defined as

$$\mathcal{D}_i = \text{argmin}\{UB_i(\mathcal{D}') \mid \mathcal{D}' \in \mathcal{F}_{1,k}\}.$$

The motivation for minimizing $UB_i()$ is that for every tree $\mathcal{D}' \in \mathcal{F}_{1,k}$, $UB_i()$ is an upper bound on (3), that is,

$$\sum_{u \in \mathcal{D}'} t_u diam(u)_i \leq \sum_{j=1}^{k-1} T_j \cdot diam(j)_i = UB_i(\mathcal{D}').$$

To see that, let j be the integer identified with the node $u \in \mathcal{D}'$. By definition $diam(u)_i = diam(j)_i$. Moreover, we have $t_u \leq T_j$ because t_u only accounts the points that are separated from their reference centers among those that reach u, while T_j accounts all the points in $\mathcal{X}$ regardless of whether they reach u or not.

We discuss how to construct $\mathcal{D}_i$ efficiently. Let $OPT_{a,b} = \min\{UB_i(\mathcal{D}') \mid \mathcal{D}' \in \mathcal{F}_{a,b}\}$, if $a < b$, and let $OPT_{a,b} = 0$ if $a = b$. Hence, $UB_i(\mathcal{D}_i) = OPT_{1,k}$. The following relation holds for all $a < b$:

$$OPT_{a,b} = \min_{a \leq j \leq b-1} \left\{ T_j(c_i^b - c_i^a) + OPT_{a,j} + OPT_{j+1,b} \right\}. \tag{4}$$

Thus, given a set of k reference centers and the values T_j's, $\mathcal{D}_i$ can be computed in $O(k^3)$ time by solving equation (4), for $a = 1$ and $b = k$, via standard dynamic programming techniques.

3.1. Approximation analysis: overview

We prove that the cost of the clustering induced by $\mathcal{D}$ is $O(d \log k) \cdot OPT$. To reach this goal, we first show that

$$UB_i(\mathcal{D}_i) \leq 2 \log k \left(\sum_{j=1}^{k-1} (c_i^{j+1} - c_i^j) T_j \right). \tag{5}$$

The proof of this bound relies on the fact that $\mathcal{D}_i$ can be seen as a binary search tree with non-uniform probing costs. We use properties of this kind of tree, in particular the one proved in (Charikar et al., 2002) about its competitive ratio.

Let

$$OPT_i = \sum_{\mathbf{x} \in \mathcal{X}} |x_i - c(\mathbf{x})_i|$$

be the contribution of coordinate i to OPT, where $c(\mathbf{x})$ is the reference center of $\mathbf{x}$. Our second step consists of showing that

$$\left(\sum_{j=1}^{k-1} (c_i^{j+1} - c_i^j) T_j \right) / 2 \leq OPT_i. \tag{6}$$

Roughly speaking, the proof of this bound consists of projecting the points of $\mathcal{X}$ and the reference centers onto the axis i and then counting the number of times the interval $[c_i^j, c_i^{j+1}]$ appears in the segments that connect points in $\mathcal{X}$ to their reference centers. This is exactly the same line of reasoning employed to prove Lemma 6 from the supplementary version of (Dasgupta et al., 2020b).

At this point, from the two previous inequalities, we obtain

$$UB_i(\mathcal{D}_i) \leq 4 \log k \cdot OPT_i. \tag{7}$$

Finally, we prove that a factor of d is incurred when we build the tree $\mathcal{D}$ from the nodes of the trees $\mathcal{D}_1, \ldots, \mathcal{D}_d$:

$$\sum_{v \in \mathcal{D}} t_v ||diam(v)||_1 \leq d \sum_{i=1}^{d} UB_i(\mathcal{D}_i). \tag{8}$$

From (7), (8) and the identity $OPT = \sum_{i=1}^{d} OPT_i$, we obtain

$$\sum_{v \in \mathcal{D}} t_v ||diam(v)||_1 \leq 4d \log k \cdot OPT.$$

This together with Lemma 3 allows us to establish the main theorem of this section.

Theorem 3. *The price of explainability for k-medians is $O(d \log k)$.*

4. The k-means problem

4.1. Improved bounds for low dimensions

The result we obtained for the k-medians problem can be extended to the k-means problem:

Theorem 4. *The price of explainability for k-means is $O(dk \log k)$.*

From an algorithmic perspective, in order to establish the theorem, we only need to replace the definition of $UB_i(\mathcal{D}')$ for a tree $\mathcal{D}'$ in $\mathcal{F}_{a,b}$ with

$$UB'_i(\mathcal{D}') = \sum_{j=a}^{b-1} T_j \cdot (diam(j)_i)^2.$$

Note that the only difference is the replacement of $diam(j)_i$ with $(diam(j)_i)^2$. As a consequence, for the k-means problem, the tree $\mathcal{D}_i$ is defined as the tree $\mathcal{D}'$ in $\mathcal{F}_{1,k}$ for which $UB'_i(\mathcal{D}')$ is minimum. It can also be constructed via dynamic programming.

Theorem 4 can be proved by using arguments similar to those employed to bound the price of explainability for k-medians. In fact, the following inequalities are, respectively, counterparts of the inequalities (1), (5), (6) and (8):

$$cost(\mathcal{C}) \leq OPT + \sum_{v \in \mathcal{D}} t_v ||diam(v)||_2^2, \tag{9}$$

$$UB'_i(\mathcal{D}_i) \leq 2k \log k \left(\sum_{j=1}^{k-1} (c_i^{j+1} - c_i^j)^2 \cdot T_j \right), \tag{10}$$

$$\left(\sum_{j=1}^{k-1} (c_i^{j+1} - c_i^j)^2 \cdot T_j \right) / 2 \leq OPT_i, \tag{11}$$

$$\sum_{v \in \mathcal{D}} t_v ||diam(v)||_2^2 \le d \sum_{i=1}^{d} UB_i'(\mathcal{D}_i). \qquad (12)$$

The only significant difference occurs in inequality (10) since it incurs an extra factor of k with respect to its counterpart. From the three last inequalities and the identity $OPT = \sum_{i=1}^{d} OPT_i$, we obtain

$$\sum_{v \in \mathcal{D}} t_v ||diam(v)||_2^2 \le 4dk \log k \cdot OPT.$$

This together with the inequality (9) allows us to establish Theorem 4.

4.2. Ex-Greedy: a practical algorithm for explainable k-means

We propose a simple greedy algorithm, denoted by Ex-Greedy, for building explainable clustering for the k-means problem. We provide evidence that it performs very well in practice.

The algorithm starts with the set S of representatives of an unrestricted k-clustering $\mathcal{C}^{ini}$ for the dataset $\mathcal{X}$ and then builds a decision tree $\mathcal{D}$ with k leaves, where each of them includes exactly one representative from S.

Let u be a node of the decision tree and let $\mathcal{X}^u$ and $\mathcal{S}^u$ be, respectively, the set of points and the set of reference centers (representatives of $\mathcal{C}^{ini}$) that reach u. We define the cost of a partition (L, R) of the points in $\mathcal{X}^u \cup \mathcal{S}^u$ as

$$cost(L,R) = \sum_{\mathbf{x} \in L \cap \mathcal{X}^u} \min_{\mathbf{c} \in L \cap \mathcal{S}^u} ||\mathbf{x} - \mathbf{c}||_2^2 + \sum_{\mathbf{x} \in R \cap \mathcal{X}^u} \min_{\mathbf{c} \in R \cap \mathcal{S}^u} ||\mathbf{x} - \mathbf{c}||_2^2.$$

To split a node u, that is reached by more than one representative, Ex-Greedy selects the axis-aligned cut that induces a partition with minimum cost.

Ex-Greedy can be implemented in $O(ndkH + nd \log n)$ time, where H is the depth of the resulting decision tree. Note that $H \le k$ and in many relevant applications k is small. The time complexity corresponds to H iterations of Lloyd's k-means algorithm.

Experiments. (Dasgupta et al., 2020a; Frost et al., 2020) compared 6 methods that build explainable clusterings, over 10 datasets. These methods also allow the construction of decision trees with more than k leaves but this is not relevant for our experiments. For trees with k leaves, the IMM algorithm proposed in (Dasgupta et al., 2020b) obtained the best results, or was very close to it, for all datasets but one (CIFAR-10).

Table 1. Comparison of Ex-Greedy (Ex-G) and IMM over 10 datasets

Dataset	n	d	k	IMM	Ex-G
BreastCancer	569	30	2	1.00	1.00
Iris	150	4	3	1.04	1.04
Wine	178	13	3	1.00	1.00
Covtype	581,012	54	7	1.03	1.03
Mice	552	77	8	1.12	**1.09**
Digits	1,797	64	10	1.23	**1.21**
CIFAR-10	50,000	3,072	10	1.23	**1.17**
Anuran	7,195	22	10	1.30	**1.15**
Avila	20,867	12	12	1.1	**1.09**
Newsgroups	18,846	1,069	20	1.01	1.01

Given the success of IMM, we compared it with our method Ex-Greedy on the same datasets. The column IMM (resp. Ex-G) of Table 1 shows the average ratio between the cost of the clustering obtained by IMM (resp. Ex-Greedy) and that of the initial unrestricted clustering $\mathcal{C}^{ini}$ produced by scikit-learn's KMeans algorithm (Pedregosa et al., 2011). Following (Frost et al., 2020), the value of k is the number of classes for the classification task associated with the dataset.

Each dataset was run for 10 iterations, with random seeds from 1 to 10, to ensure the reproducibility of results. For each iteration, we initially achieve an unrestricted solution $\mathcal{C}^{ini}$ by running the KMeans algorithm provided in the scikit-klearn package with default parameters. We then pass $\mathcal{C}^{ini}$ to the implementation of IMM from (Frost et al., 2020), available at https://github.com/navefr/ExKMC, and to our implementation of Ex-Greedy, to find two explainable clustering solutions induced by decision trees.

For 5 datasets, the results were very similar while for the others (bold in Table 1) Ex-Greedy performed better than IMM. Figure 1 presents box plots for the 5 datasets where there was a difference of at least 0.01 on the average results. It is interesting to note that the dispersion of Ex-Greedy is considerably smaller.

In terms of running time both methods spent less than 1 second, for 6 datasets. For the remaining datasets IMM was the fastest as shown in Table 2. In spite of that, we understand that Ex-Greedy is fast enough to be used in practice.

Additional Details. All our experiments were executed in a MacBook Air, 8Gb of RAM, processor 1,6 GHz Dual-Core Intel Core i5, executing macOS Catalina, version 10.15.7. Our code is availble in https://github.com/

Figure 1. Box Plots for the datasets with difference at least 0.01

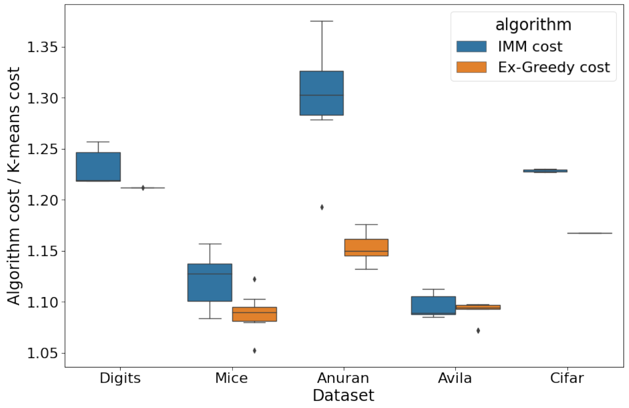


Table 2. Average running times in seconds for `Ex-Greedy` and `IMM`

Dataset	`IMM` (sec)	`Ex-Greedy` (sec)
`Avila`	1.7	2.4
`Covtype`	42	53
`Newsgroups`	41	102
`CIFAR-10`	312	378

`lmurtinho/ExKMC`.

The datasets `Iris`, `Wine`, `Breast Cancer`, `Digits`, `Covtype`, `Mice` and `Newsgroup` are available in Python's `scikit-learn`; `Cifar-10` is available in `TensorFlow`; `Anuran` and `Avila` were downloaded from UCI.

For `Mice`, the examples with missing values were removed. For `Avila`, the training set and the testing set are used together. Finally, for `Newsgroup`, we removed headers, footers, quotes, stopwords, and words that either appear in less than 1% or more than 10% of the documents, following (Frost et al., 2020).

5. Maximum-Spacing clustering

We show that the price of explainability for the maximum-spacing problem is $\Theta(n-k)$.

5.1. Lower bound

The following simple construction shows that the price of explainability is $\Omega(n-k)$.

Let $C_1 = \{(0,i)|0 \le i \le p\} \cup \{(i,0)|0 \le i \le p\}$. Moreover, for $i = 2, \dots, k$, let $C_i = \{(i-1)(p-1),(p-1)\}$. The dataset $\mathcal{X}$ for our instance is given by $C_1 \cup \dots \cup C_k$.

The unrestricted k-clustering $(C_1, \dots, C_k)$ has spacing $p-1 = (n-k)/2-1$. On the other hand, every explainable k-clustering has spacing 1. To see that, note that we cannot have all the points of $C_1 \cup C_2$ in the same cluster, for otherwise we would have at most $k-1$ clusters. Thus, we need to separate at least 2 points from $C_1 \cup C_2$ and the only way to accomplish that, via axis-aligned cuts, forces the separation of 2 points in C_1 that are at distance 1 from each other. Thus, the spacing will be 1.

Lemma 4. *The price of explainability for the maximum-spacing clustering problem is $\Omega(n-k)$.*

5.2. Upper bound

We present an algorithm that always obtains an explainable clustering with spacing $O((n-k)OPT)$, where OPT is the spacing of the optimal unrestricted clustering. That, together with the previous lemma, implies that the price of explainability for the maximum-spacing problem is $\Theta(n-k)$.

Algorithm 3 receives an optimal k-clustering $\mathcal{C}^*$ as input and uses it as a guide to transforming an initial single cluster containing all points of $\mathcal{X}$ into an explainable k-clustering. The existence of cluster C at line (*) follows from a simple pigeonhole argument. The motivation for this choice is that C has two points at distance at least OPT, which is used to show the existence of a cut with a large enough margin.

Algorithm 3 `Ex-SingleLink`($\mathcal{X}$)

```
$\mathcal{C}^* \leftarrow$ optimal unrestriced $k$-clustering for points in $\mathcal{X}$.
$\mathcal{C} \leftarrow$ single cluster containing all points of $\mathcal{X}$
for $i = 1, \dots, k-1$ do
    Select a cluster $C \in \mathcal{C}$ that contains two points that lie in
    different clusters in $\mathcal{C}^*$.   (*)
    Split $C$ using an axis-aligned cut that yields a 2-clustering
    $(C', C'')$ with maximum possible spacing.
    Remove $C$ from $\mathcal{C}$ and update $\mathcal{C}$ to $\mathcal{C} \cup \{C', C''\}$
end for
```

Lemma 5. *Given a set of points $\mathcal{X}$,* `Ex-SingleLink`*($\mathcal{X}$) obtains a k-clustering $\mathcal{C}$ with spacing at least $OPT/(n-k)$, where OPT is the spacing of an optimal unrestricted clustering.*

We can state the main result of this section.

Theorem 5. *The price of explainability for the maximum-spacing problem is $\Theta(n-k)$.*

Acknowledgements

We would like to thank the anonymous reviewers. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. The work of the first author is partially supported by CNPq (grant 307572/2017-0) and by FAPERJ, grant E- 26/202.823/2018. The second author is partially supported by CNPq, grant 141806/2020-6.

References

Aloise, D., Deshpande, A., Hansen, P., and Popat, P. Np-hardness of euclidean sum-of-squares clustering. *Mach. Learn.*, 75(2):245–248, 2009. doi: 10.1007/s10994-009-5103-0. URL https://doi.org/10.1007/s10994-009-5103-0.

Baralis, E., Pastor, D. E., and Cannone, M. Explainable ai for clustering algorithms.

Basak, J. and Krishnapuram, R. Interpretable hierarchical clustering by constructing an unsupervised decision tree. *IEEE Transactions on Knowledge and Data Engineering*, 17(1):121–132, 2005. doi: 10.1109/TKDE.2005.11.

Bertsimas, D. and Dunn, J. Optimal classification trees. *Machine Learning*, 106(7):1039–1082, 2017.

Bertsimas, D., Orfanoudaki, A., and Wiberg, H. Interpretable clustering: an optimization approach. *Machine Learning*, pp. 1–50, 2020.

Blockeel, H., De Raedt, L., and Ramon, J. Top-down induction of clustering trees. *arXiv preprint cs/0011032*, 2000.

Charikar, M., Fagin, R., Guruswami, V., Kleinberg, J. M., Raghavan, P., and Sahai, A. Query strategies for priced information. *J. Comput. Syst. Sci.*, 64(4):785–819, 2002. doi: 10.1006/jcss.2002.1828. URL https://doi.org/10.1006/jcss.2002.1828.

Chavent, M., Guinot, C., Lechevallier, Y., and Tenenhaus, M. Méthodes divisives de classification et segmentation non supervisée: Recherche d'une typologie de la peau humaine saine. *Revue de statistique appliquée*, 47(4):87–99, 1999.

Chen, J. *Interpretable Clustering Methods*. PhD thesis, Northeastern University, 2018.

Chen, J., Chang, Y., Hobbs, B., Castaldi, P., Cho, M., Silverman, E., and Dy, J. Interpretable clustering via discriminative rectangle mixture model. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, pp. 823–828. IEEE, 2016.

Dasgupta, S., Frost, N., Moshkovitz, M., and Rashtchian, C. Explainable k-means clustering: Theory and practice. In *XXAI: Extending Explainable AI Beyond Deep Models and Classifiers*, 2020a.

Dasgupta, S., Moshkovitz, M., Rashtchian, C., and Frost, N. Explainable k-means and k-medians clustering. In *ICML*, volume 119 of *Proceedings of Machine Learning Research*, pp. 7055–7065. PMLR, 2020b.

Fisher, D. H. Knowledge acquisition via incremental conceptual clustering. *Machine learning*, 2(2):139–172, 1987.

Fraiman, R., Ghattas, B., and Svarc, M. Interpretable clustering using unsupervised binary trees. *Advances in Data Analysis and Classification*, 7(2):125–145, 2013.

Frost, N., Moshkovitz, M., and Rashtchian, C. Exkmc: Expanding explainable k-means clustering. *arXiv preprint arXiv:2006.02399*, 2020.

Horel, E. and Giesecke, K. Computationally efficient feature significance and importance for machine learning models, 2019.

Horel, E., Giesecke, K., Storchan, V., and Chittar, N. Explainable clustering and application to wealth management compliance, 2020.

Kanungo, T., Mount, D. M., Netanyahu, N. S., Piatko, C. D., Silverman, R., and Wu, A. Y. A local search approximation algorithm for k-means clustering. *Comput. Geom.*, 28(2-3):89–112, 2004. doi: 10.1016/j.comgeo.2004.03.003. URL https://doi.org/10.1016/j.comgeo.2004.03.003.

Kauffmann, J., Esders, M., Montavon, G., Samek, W., and Müller, K.-R. From clustering to cluster explanations via neural networks. *arXiv preprint arXiv:1906.07633*, 2019.

Laber, E. S., Milidiú, R. L., and Pessoa, A. A. On binary searching with nonuniform costs. *SIAM J. Comput.*, 31(4):1022–1047, 2002. doi: 10.1137/S0097539700381991. URL https://doi.org/10.1137/S0097539700381991.

Liu, B., Xia, Y., and Yu, P. S. Clustering through decision tree construction. In *Proceedings of the ninth international conference on Information and knowledge management*, pp. 20–29, 2000.

Loyola-González, O., Gutierrez-Rodríguez, A. E., Medina-Pérez, M. A., Monroy, R., Martínez-Trinidad, J. F., Carrasco-Ochoa, J. A., and García-Borroto, M. An explainable artificial intelligence model for clustering numerical databases. *IEEE Access*, 8:52370–52384, 2020. doi: 10.1109/ACCESS.2020.2980581.

Lundberg, S. M. and Lee, S.-I. A unified approach to interpreting model predictions. In *Advances in neural information processing systems*, pp. 4765–4774, 2017.

Megiddo, N. and Supowit, K. J. On the complexity of some common geometric location problems. *SIAM J. Comput.*, 13(1):182–196, 1984. doi: 10.1137/0213014. URL https://doi.org/10.1137/0213014.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Pelleg, D. and Moore, A. Mixtures of rectangles: Interpretable soft clustering. In *ICML*, pp. 401–408, 2001.

Plant, C. and Böhm, C. Inconco: Interpretable clustering of numerical and categorical objects. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pp. 1127–1135, New York, NY, USA, 2011. Association for Computing Machinery. ISBN 9781450308137. doi: 10.1145/2020408.2020584. URL https://doi.org/10.1145/2020408.2020584.

Ribeiro, M. T., Singh, S., and Guestrin, C. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

Saisubramanian, S., Galhotra, S., and Zilberstein, S. Balancing the tradeoff between clustering value and interpretability. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pp. 351–357, 2020.

Vidal, T. and Schiffer, M. Born-again tree ensembles. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9743–9753. PMLR, 2020. URL http://proceedings.mlr.press/v119/vidal20a.html.

Williamson, D. P. and Shmoys, D. B. *The Design of Approximation Algorithms*. Cambridge University Press, 2011. ISBN 978-0-521-19527-0. URL http://www.cambridge.org/de/knowledge/isbn/item5759340/?site_locale=de_DE.