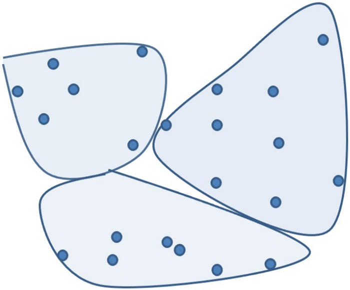
On the price of explainability for some clustering problems

Eduardo Laber and Lucas Murtinho

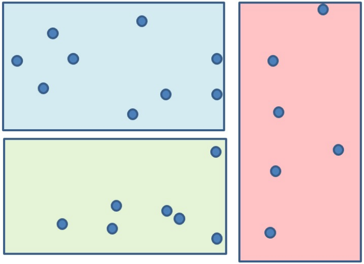
International Conference on Machine Learning
July 2021

Table of Contents

Decision-tree explainable clustering



Unrestricted partition



Explainable partition

Price of Explainability

For a **minimization** problem,

$$\text{PoE} = \max_{I \in \mathcal{I}} \left\{ \frac{OPT_e(I)}{OPT_u(I)} \right\}$$

- OPT_e: optimal cost for an explainable partition
- OPT_u: optimal cost for an unrestricted partition
- $\mathcal{I}$: the set of instances of the problem

Price of Explainability

For a **maximization** problem,

$$\text{PoE} = \max_{I \in \mathcal{I}} \left\{ \frac{OPT_u(I)}{OPT_e(I)} \right\}$$

- OPT_u: optimal value for an unrestricted partition
- OPT_e: optimal value for an explainable partition
- $\mathcal{I}$: the set of instances of the problem

Our results

Criterion	Lower bound	Upper bound
k-centers	$\Omega\left(\frac{\sqrt{d}k^{1-1/d}}{\log^{1.5} k}\right)$	$O\left(\sqrt{d}k^{1-1/d}\right)$
k-medians	$\Omega(\log k)$	$O(k)$, $O(d \log k)$
k-means	$\Omega(\log k)$	$O(k^2)$, $O(dk \log k)$
maximum spacing	$\Theta(n-k)$	

Lower and upper bounds for the PoE of different clustering problems. Bounds in red are from [Dasgupta et al., 2020, ICML].

Related work

- [Dasgupta et al., 2020, ICML]:
  - Price of Explainability
  - Bounds for k-medians and k-means
  - IMM algorithm

Related work

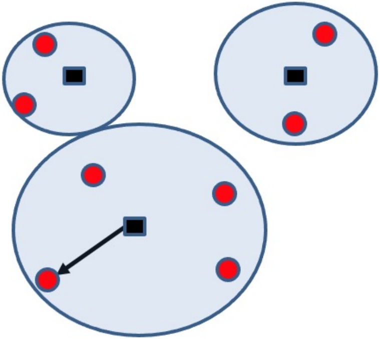
- [Dasgupta et al., 2020, ICML]:
  - Price of Explainability
  - Bounds for k-medians and k-means
  - IMM algorithm
- [Frost et al., 2020]:
  - ExKMC algorithm (not limited to k leaves)
  - Experimental results

Related work

- [Dasgupta et al., 2020, ICML]:
  - Price of Explainability
  - Bounds for k-medians and k-means
  - IMM algorithm
- [Frost et al., 2020]:
  - ExKMC algorithm (not limited to k leaves)
  - Experimental results
- [Charikar et al., STOC 00]:
  - Binary search tree with bound of $O(\log k)$ of finding one of k items

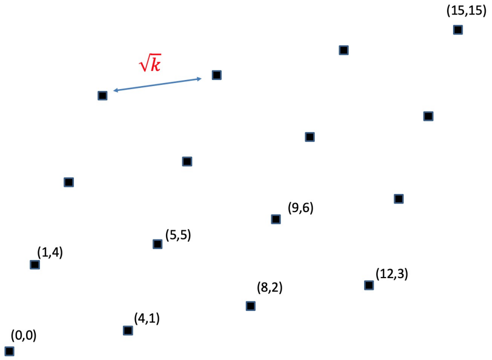
Table of Contents

k-centers: problem description



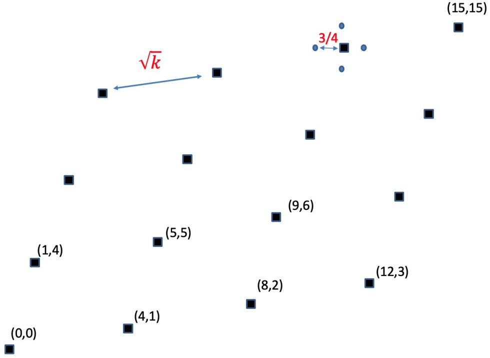
Minimize the maximum distance between a point and the closest reference center.

k-centers: PoE lower bound



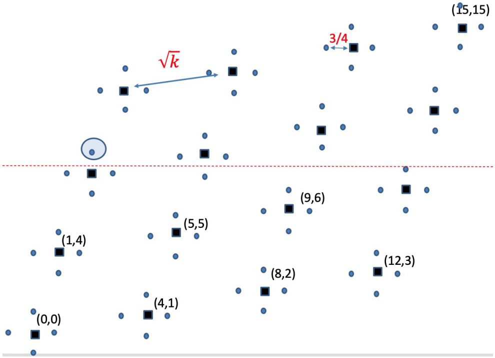
- $k = 16, d = 2$
- $c^i = (i, 4i \mod 15)$

k-centers: PoE lower bound



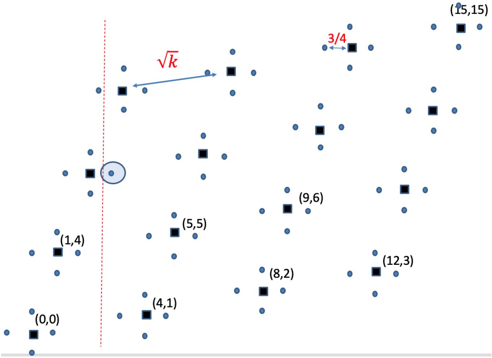
- $k = 16, d = 2$
- $c^i = (i, 4i \mod 15)$
- Unrestricted cost: $\frac{3}{4}$
- Distance between centers: $\approx \sqrt{k}$

k-centers: PoE lower bound



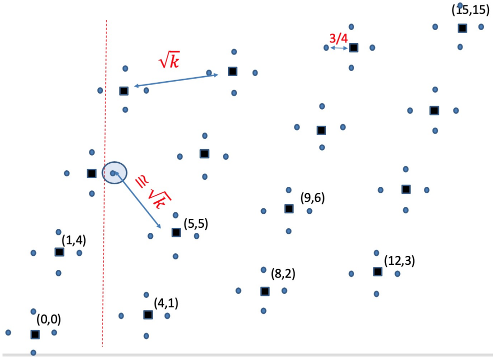
- $k = 16, d = 2$
- $c^i = (i, 4i \mod 15)$
- Unrestricted cost: $\frac{3}{4}$
- Distance between centers: $\approx \sqrt{k}$
- No **mistakeless cuts**

k-centers: PoE lower bound



- $k = 16, d = 2$
- $c^i = (i, 4i \mod 15)$
- Unrestricted cost: $\frac{3}{4}$
- Distance between centers: $\approx \sqrt{k}$
- No **mistakeless cuts**

k-centers: PoE lower bound



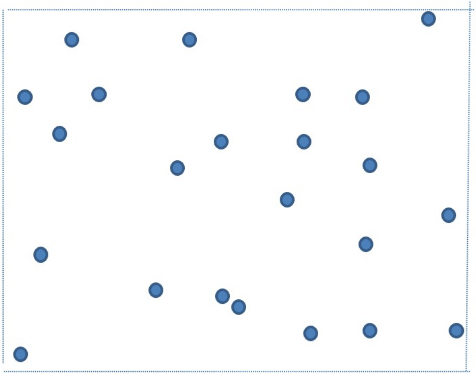
- $k = 16, d = 2$
- $c^i = (i, 4i \mod 15)$
- Unrestricted cost: $\frac{3}{4}$
- Distance between centers: $\approx \sqrt{k}$
- No **mistakeless cuts**
- Explainable cost: $\Omega(\sqrt{k})$ (for $d = 2$)

k-centers: PoE lower bound

For general d:

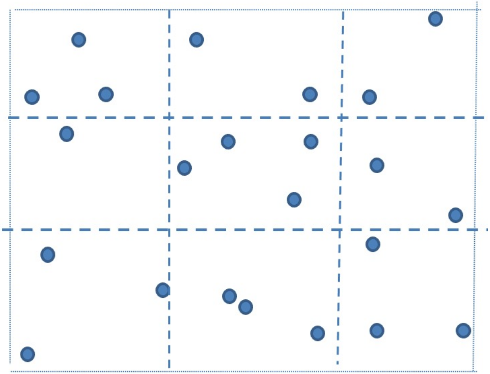
- Center c^i coordinates are shifts of i representation in base $b = k^{1/p}$, where $p = p(k, d)$
- $2d$ points associated to each center, identical to associated center in all but a single coordinate
- Price of Explainability:
  - $\Omega(k^{1-1/d})$ (if $d < \log k / \log \log k$)
  - $\Omega\left(\sqrt{d}\frac{k\sqrt{\log \log k}}{\log^{1.5} k}\right)$ (otherwise)

k-centers: PoE upper bound



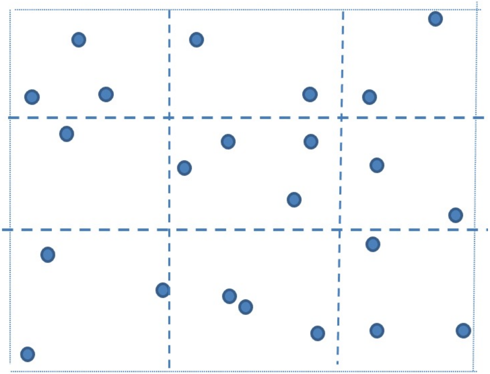
- $k = 9, d = 2$
- Bounding box of size $D_1 \times D_2$

k-centers: PoE upper bound



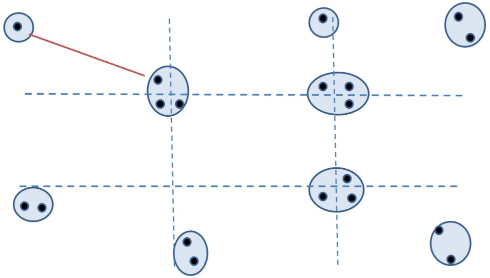
- $k = 9, d = 2$
- Bounding box of size $D_1 \times D_2$
- **Grid** strategy: equal-sized boxes $(D_1/\sqrt{k}) \times (D_2/\sqrt{k})$

k-centers: PoE upper bound



- $k = 9, d = 2$
- Bounding box of size $D_1 \times D_2$
- **Grid** strategy: equal-sized boxes $(D_1/\sqrt{k}) \times (D_2/\sqrt{k})$
- Cost $\leq \frac{\max\{D_1, D_2\}}{\sqrt{k}}$

k-centers: PoE upper bound



- $k = 9, d = 2$
- Bounding box of size $D_1 \times D_2$
- **Grid** strategy: equal-sized boxes $(D_1/\sqrt{k}) \times (D_2/\sqrt{k})$
- Cost $\leq \frac{\max\{D_1, D_2\}}{\sqrt{k}}$
- Can be arbitrarily bad

k-centers: PoE upper bound

- **Refined Grid**:
  1. Perform as many **mistakeless cuts** as possible
  2. Apply **Grid**

k-centers: PoE upper bound

- **Refined Grid**:
  1. Perform as many **mistakeless cuts** as possible
  2. Apply **Grid**
- If no more mistakeless cuts are possible, $OPT_{unrestricted} \geq \frac{\max\{D_1, D_2\}}{k}$

k-centers: PoE upper bound

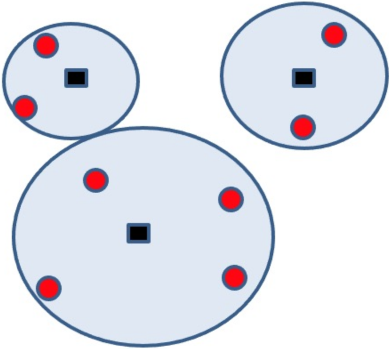
- **Refined Grid**:
  1. Perform as many **mistakeless cuts** as possible
  2. Apply **Grid**
- If no more mistakeless cuts are possible, $OPT_{unrestricted} \geq \frac{\max\{D_1, D_2\}}{k}$
- PoE is $O(\sqrt{k})$ for $d = 2$ (tight bound)

k-centers: PoE upper bound

- **Refined Grid**:
  1. Perform as many **mistakeless cuts** as possible
  2. Apply **Grid**
- If no more mistakeless cuts are possible, $OPT_{unrestricted} \geq \frac{\max\{D_1, D_2\}}{k}$
- PoE is $O(\sqrt{k})$ for $d = 2$ (tight bound)
- For general d, PoE is $O(\sqrt{d}k^{1-1/d})$

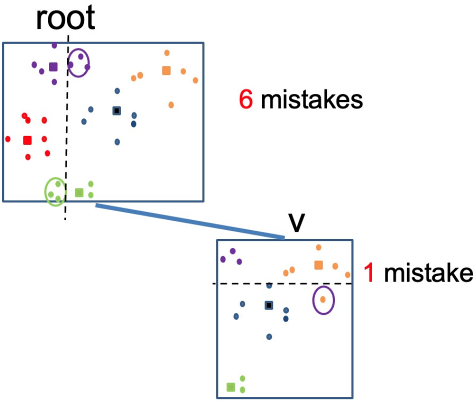
Table of Contents

k-medians: problem description



Minimize the sum of the ℓ_1 distances between each point and its reference center (the median of all points in the cluster).

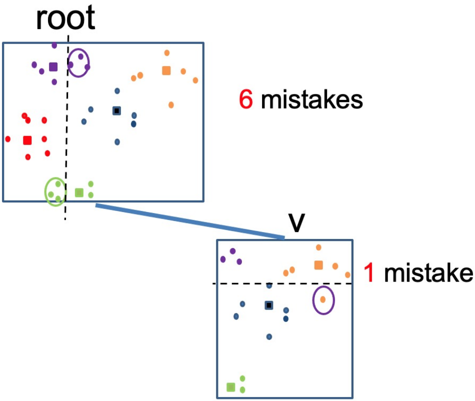
k-medians: original PoE upper bound



[Dasgupta et al., ICML 2020]:

- IMM algorithm: greedily apply cut that minimizes the number of mistakes
  - Cost$(D) = OPT + \sum_{v \in D}$ Excess(v)
  - Excess(v) $\leq$ #mistakes$(v) \cdot$ diam(v)

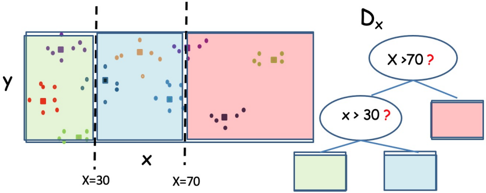
k-medians: original PoE upper bound



[Dasgupta et al., ICML 2020]:

- Theorem: IMM yields upper bound of $O(k)$ to PoE of k-medians
  - Independent of d
  - What happens when d is small?

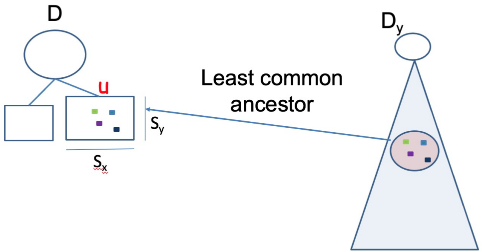
k-medians: improved PoE upper bound for low dimensions



Our approach:

- Build a tree D_i for each dimension $i = 1, \ldots, d$
  - Factor of $\log k$

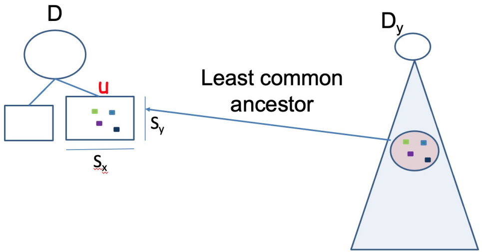
k-medians: improved PoE upper bound for low dimensions



Our approach:

- Build a tree D_i for each dimension $i = 1, \ldots, d$
  - Factor of $\log k$
- Build the final tree D selecting nodes from $D_1, \ldots, D_d$
  - Factor of d

k-medians: improved PoE upper bound for low dimensions



Our approach:

- Build a tree D_i for each dimension $i = 1, \ldots, d$
  - Factor of $\log k$
- Build the final tree D selecting nodes from $D_1, \ldots, D_d$
  - Factor of d
- PoE is $O(d \log k)$

k-medians: finding the tree for a single coordinate

- For a given i, minimizing

$$\text{Excess}(D, i) = \sum_{v \in D_i} \#\text{Mistakes}(v) \cdot \text{Diam}(v)_i$$

reduces to a binary search problem where items have distinct search probabilities and probing costs:

  - probing cost = # of mistakes
  - search probability = distance between item's adjacent centers at coordinate i

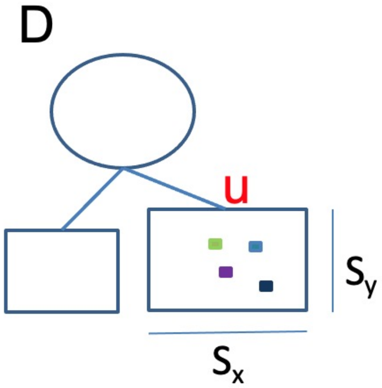
k-medians: finding the tree for a single coordinate

- For a given i, minimizing

$$\text{Excess}(D, i) = \sum_{v \in D_i} \#\text{Mistakes}(v) \cdot \text{Diam}(v)_i$$

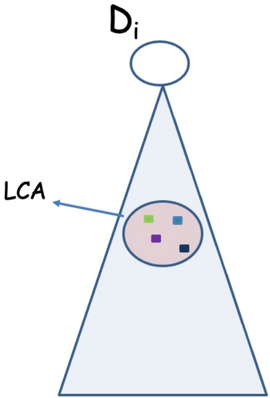
  reduces to a binary search problem where items have distinct search probabilities and probing costs:
  - probing cost $= \#$ of mistakes
  - search probability $=$ distance between item's adjacent centers at coordinate i
- [Charikar et al., STOC 00]: BST for k items where the cost of finding an item j is at most $O(\log k)$ larger than its probing cost

k-medians: selecting the best cut for the final tree



- Pick coordinate i associated to the largest side of the box that bounds the points in u

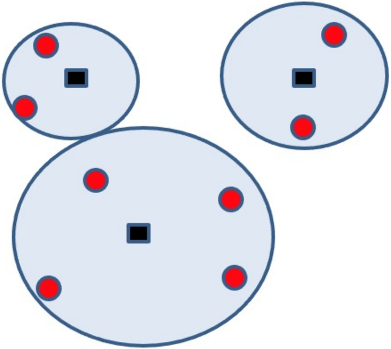
k-medians: selecting the best cut for the final tree



- Pick coordinate i associated to the largest side of the box that bounds the points in u
- Apply cut in D_i given by the least common ancestors of the centers that reached u

Table of Contents

k-means: problem description



Minimize the sum of the squared ℓ_2 distances between each point and its reference center (the mean of all points in the cluster).

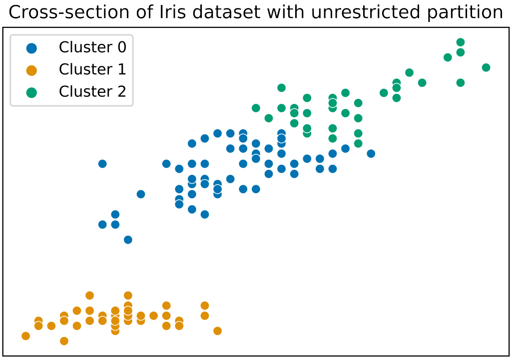
k-means: improved PoE upper bound for low dimensions

- Same algorithm as for k-medians
- The factor for each D_i is multiplied by k due to the cost function of k-means
- The PoE for k-means is $O(dk \log k)$

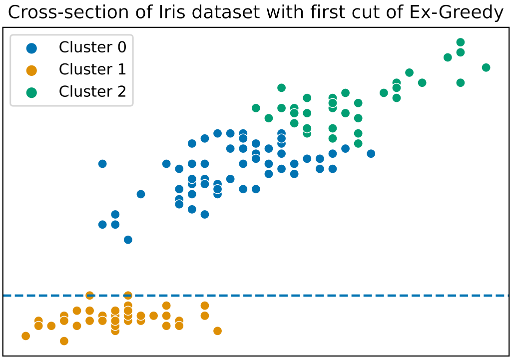
k-means: a practical algorithm

- **Ex-Greedy**: recursively find the best cut that separates at least two centers and that minimizes the k-means cost of a k-partition, considering that:
  - points cannot be assigned to centers from which they were separated
  - the k reference centers are always the same
- The algorithm maintains a k-partition as it runs, but only when it ends is it guaranteed that the partition is explainable
  - Contrast with **ExKMC** [Frost et al., 2020], in which explainable partitions with $2, 3, \ldots, k$ clusters are defined after each cut

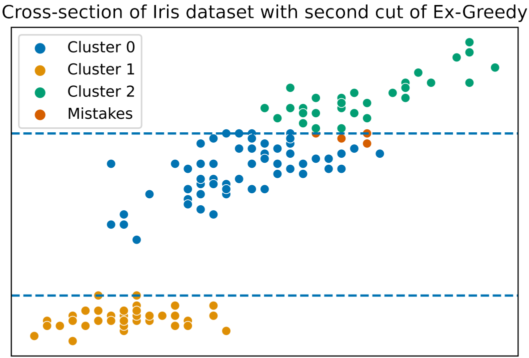
k-means: **Ex-Greedy** example



k-means: **Ex-Greedy** example



k-means: **Ex-Greedy** example



k-means: **Ex-Greedy** results

Table 1: Comparison of `Ex-Greedy` and `IMM` over 10 datasets

Dataset	n	d	k	IMM	Ex-Greedy
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

k-means: **Ex-Greedy** results

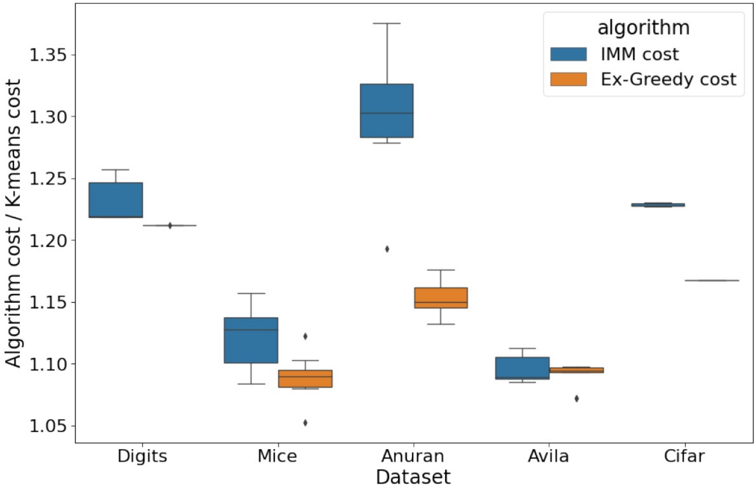
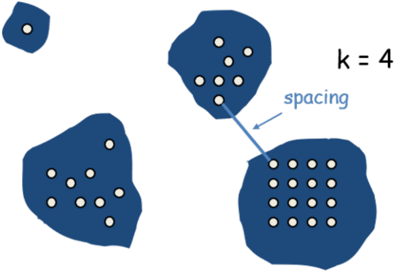


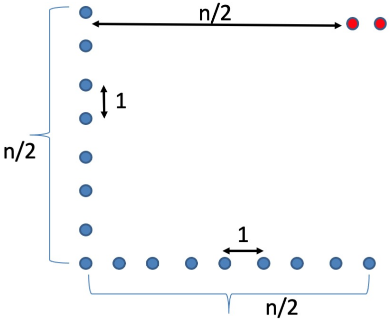
Table of Contents

Maximum spacing: problem description



Maximize the distance between the closest points that belong to different clusters.

Maximum spacing: PoE lower bound



- $k = d = 2$
- $OPT_{unrestricted} = n/2$
- $OPT_{explainable} = 1$
- For general d, dataset with unrestricted spacing $O(n-k)$ and explainable spacing 1
- PoE is $\Omega(n-k)$

Maximum spacing: PoE upper bound

$O(n-k)$ algorithm:

1. $C_{exp} \leftarrow$ all points
2. $C^* \leftarrow$ optimal unrestricted partition
3. Repeat $k-1$ times:
   - $S \leftarrow$ group in C_{exp} not contained in any group of C^*
   - Split S with the axis-aligned cut that yields two clusters with maximum spacing

Thank you!

Questions?