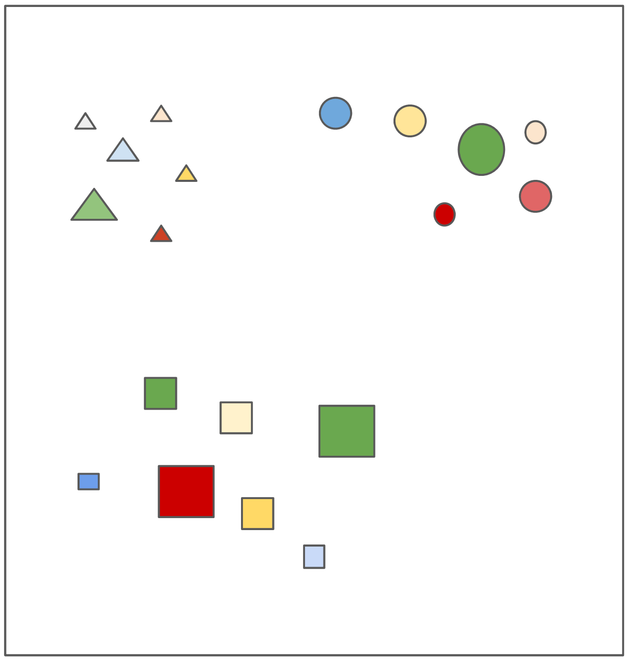
Strong Coresets for k-Median and Subspace Clustering: Goodbye Dimension

Christian Sohler and David Woodruff
(Google and TU Dortmund) and (CMU)

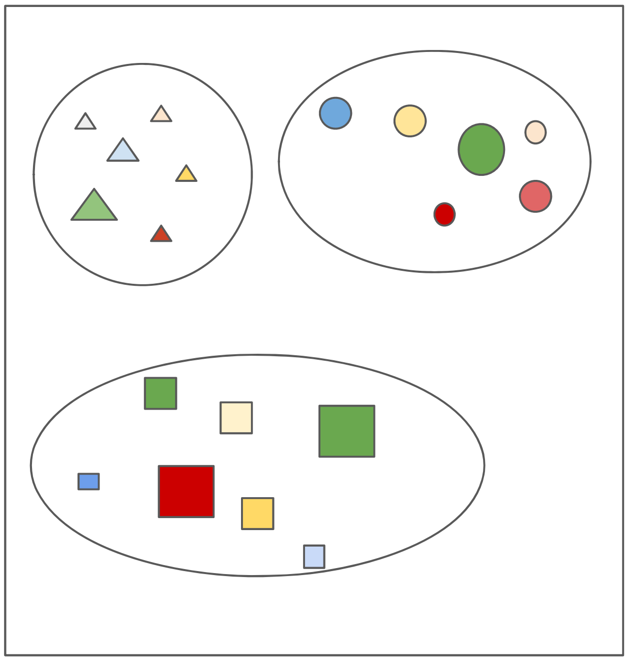
Clustering



General Goal

- Partition an input set into groups such that
- Items in the same group are similar
- Items in different groups are dissimilar

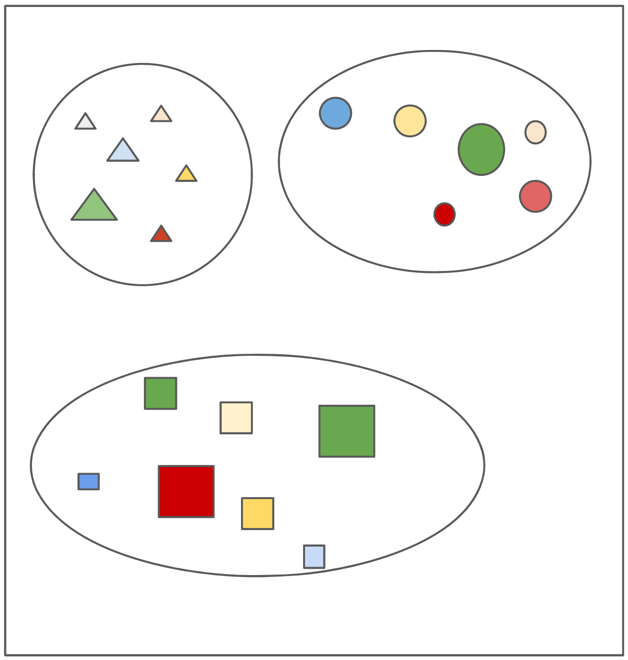
Clustering



General Goal

- Partition an input set into groups such that
- Items in the same group are similar
- Items in different groups are dissimilar

Clustering



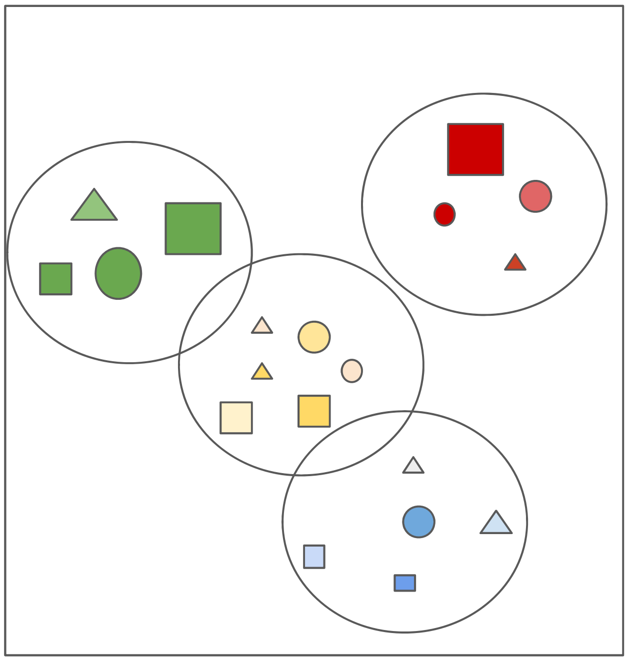
General Goal

- Partition an input set into groups such that
- Items in the same group are similar
- Items in different groups are dissimilar

But what

- If I care about colors?

Clustering



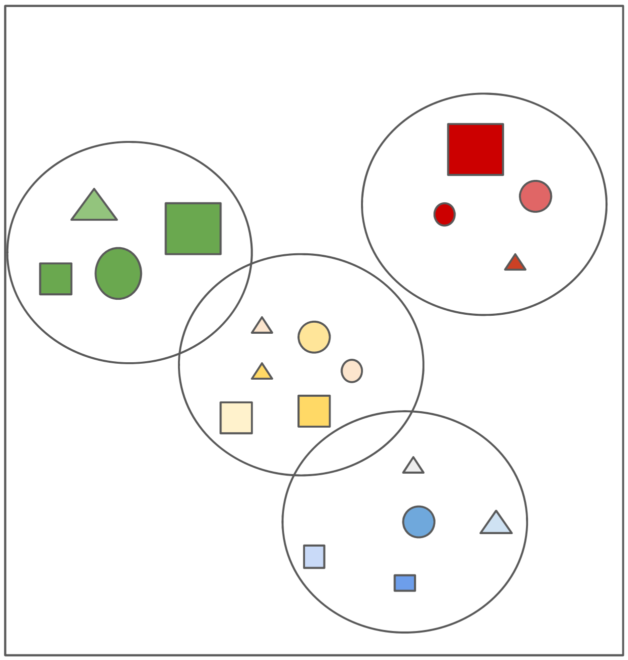
General Goal

- Partition an input set into groups such that
- Items in the same group are similar
- Items in different groups are dissimilar

But what

- If I care about colors?

Clustering



General Goal

- Partition an input set into groups such that
- Items in the same group are similar
- Items in different groups are dissimilar

But what

- If I care about colors?
- We need to define (dis)similarity!

Clustering

Examples of relevant distance and similarity measures

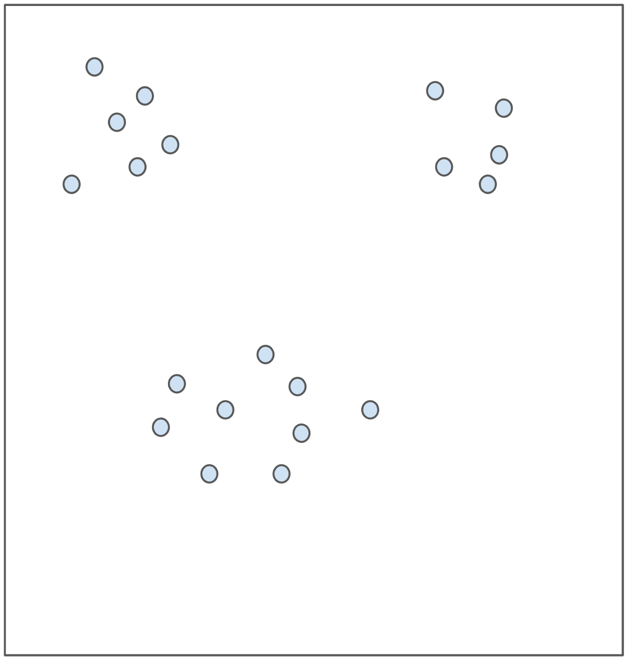
- Euclidean distance
- Squared Euclidean distance
- Metric
- Cosine similarity
- Jaccard coefficient
- Kullback-Leibler divergence
- And many more...

Clustering

Examples of relevant distance and similarity measures

- Euclidean distance
- Squared Euclidean distance
- Metric
- Cosine similarity
- Jaccard coefficient
- Kullback-Leibler divergence
- And many more...

k-Median Clustering

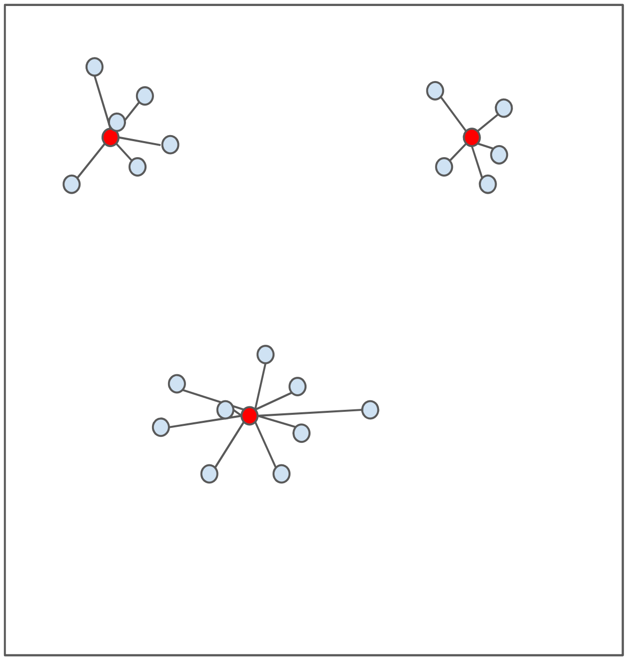


Problem Formulation

- Input: Set P of points in $\mathbb{R}^d$, number of clusters k
- Output: Set C of k centers in $\mathbb{R}^d$
- Objective:

$$\boxed{\text{minimize } \mathbf{cost(P,C)} := \sum \min_{c \in C} ||p-c||}$$

k-Median Clustering

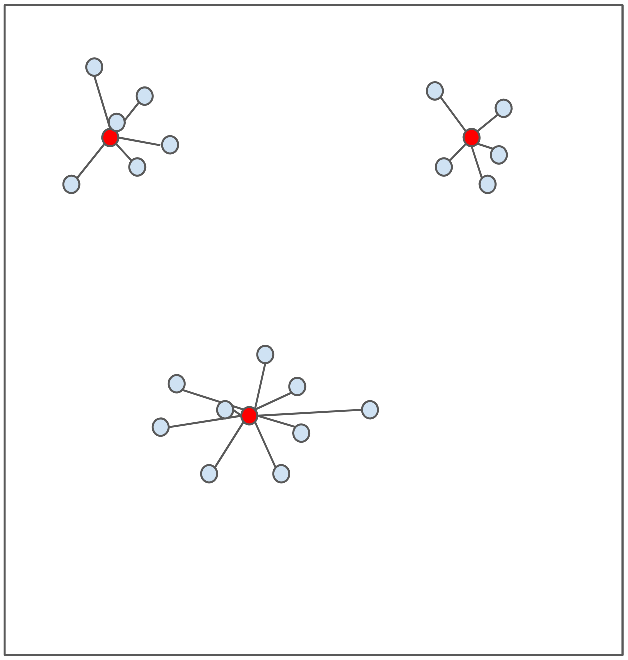


Problem Formulation

- Input: Set P of points in $\mathbb{R}^d$, number of clusters k
- Output: Set C of k centers in $\mathbb{R}^d$
- Objective:

$$\text{minimize } \mathbf{cost(P,C)} := \sum \min_{c \in C} ||p-c||$$

k-Median Clustering



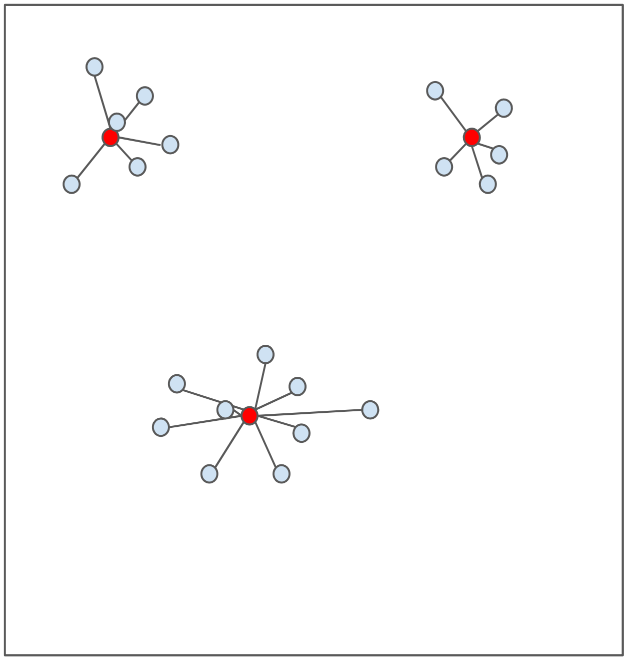
Problem Formulation

- Input: Set P of points in $\mathbb{R}^d$, number of clusters k
- Output: Set C of k centers in $\mathbb{R}^d$
- Objective:

$$\boxed{\text{minimize } \mathbf{cost(P,C)} := \sum \min_{c \in C} \|p-c\|}$$

- Could also use other distance measures

k-Means Clustering



Problem Formulation

- Input: Set P of points in $\mathbb{R}^d$, number of clusters k
- Output: Set C of k centers in $\mathbb{R}^d$
- Objective:

$$\boxed{\text{minimize } \mathbf{cost(P,C)} := \sum \min_{c \in C} ||p-c||^2}$$

- Could also use other distance measures

Clustering Very Large Data Sets

Todays Setting

- Very large input set
  - Does not fit into main memory
  - Requires distributed or streaming algorithms
- Moderate number of clusters k
  - we often think of k as being constant
- Possibly high dimensional data

Coresets

Basic Idea

- “Compress” input point set P to a small weighted set S such that S approximates P w.r.t. the problem of interest
- Many different notions of coresets around

Coresets

Definition [Har-Peled, Mazumdar, 2004]

- A weighted set S is an ($\boldsymbol{\varepsilon}$,k)-coreset for a set of points P with respect to the k-median (k-means) problem, if **for all** sets C of k centers we have

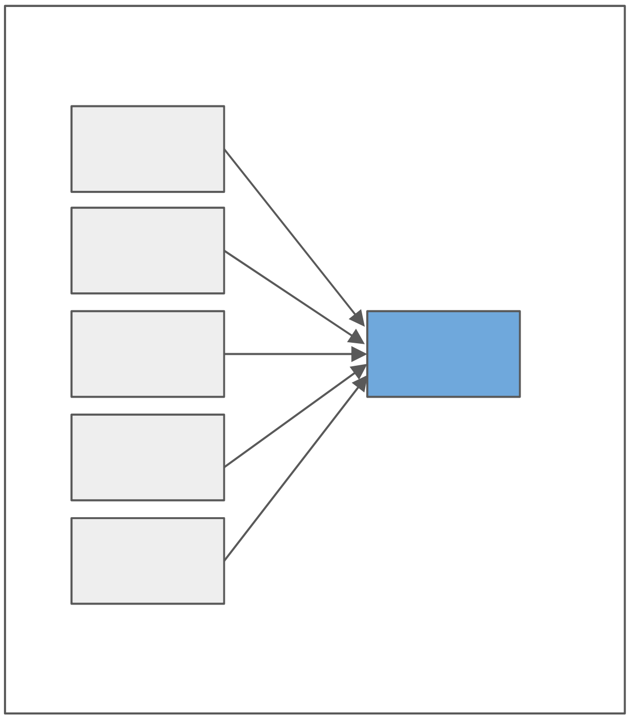
$$(1-\boldsymbol{\varepsilon})\ \mathrm{cost}(P,C) \leq \mathrm{cost}(S,C) \leq (1+\boldsymbol{\varepsilon})\ \mathrm{cost}(P,C)$$

Coresets

Composability

- Union of coresets for sets P and Q should be a coreset for $P \cup Q$

Coresets and Distributed Algorithms



Use in Distributed Algorithms

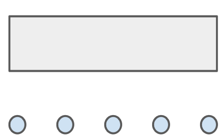
- Compute coreset locally
- Send coresets to central server
- Compute a solution on union of coresets

Coresets and Streaming Algorithms

[Agarwal, Har-Peled, Varadarajan, 2004] [Bentley, Saxe, 1980]

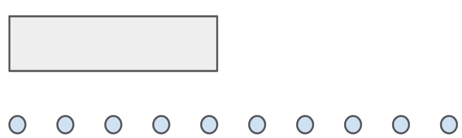
Coresets and Streaming Algorithms

[Agarwal, Har-Peled, Varadarajan, 2004] [Bentley, Saxe, 1980]



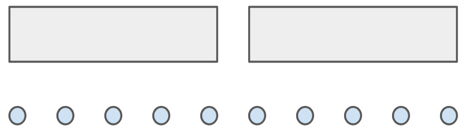
Coresets and Streaming Algorithms

[Agarwal, Har-Peled, Varadarajan, 2004] [Bentley, Saxe, 1980]



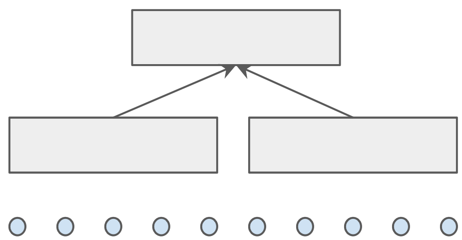
Coresets and Streaming Algorithms

[Agarwal, Har-Peled, Varadarajan, 2004] [Bentley, Saxe, 1980]



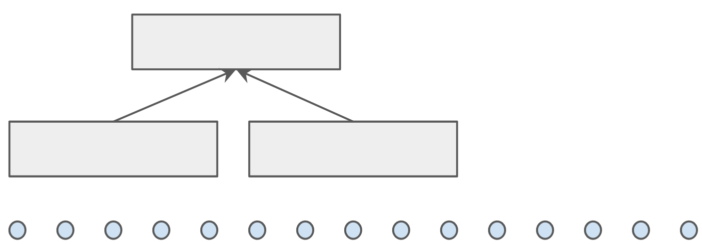
Coresets and Streaming Algorithms

[Agarwal, Har-Peled, Varadarajan, 2004] [Bentley, Saxe, 1980]



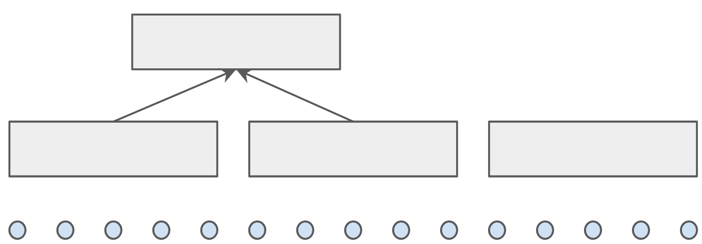
Coresets and Streaming Algorithms

[Agarwal, Har-Peled, Varadarajan, 2004] [Bentley, Saxe, 1980]



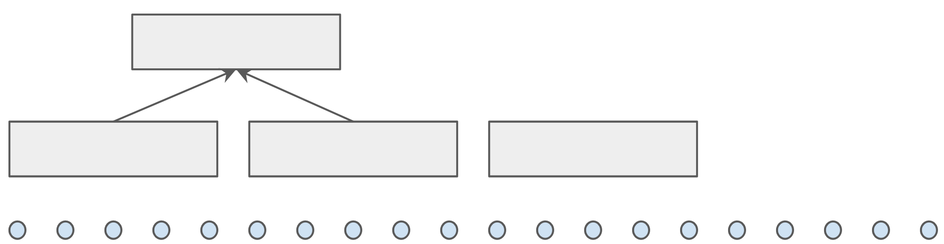
Coresets and Streaming Algorithms

[Agarwal, Har-Peled, Varadarajan, 2004] [Bentley, Saxe, 1980]



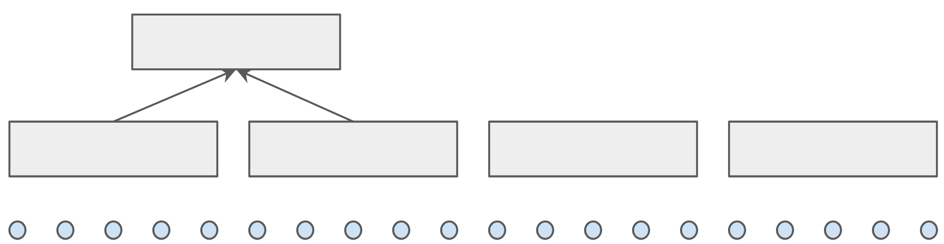
Coresets and Streaming Algorithms

[Agarwal, Har-Peled, Varadarajan, 2004] [Bentley, Saxe, 1980]



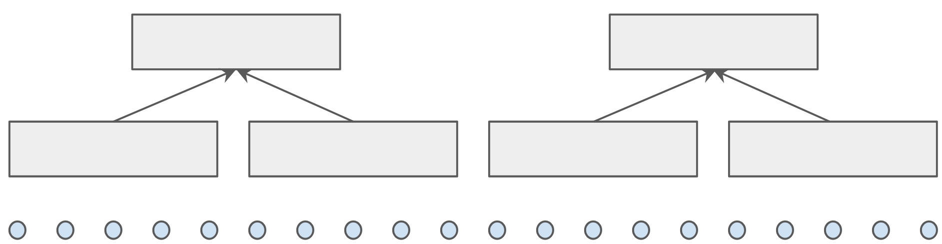
Coresets and Streaming Algorithms

[Agarwal, Har-Peled, Varadarajan, 2004] [Bentley, Saxe, 1980]

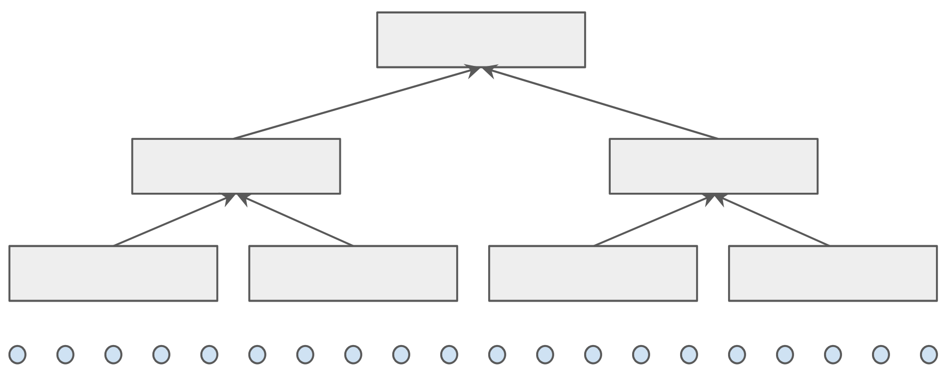


Coresets and Streaming Algorithms

[Agarwal, Har-Peled, Varadarajan, 2004] [Bentley, Saxe, 1980]



Coresets and Streaming Algorithms
[Agarwal, Har-Peled, Varadarajan, 2004] [Bentley, Saxe, 1980]



(Some) Related Work

Strong Coresets for k-Median

[Har-Peled, Mazumdar 2004]	$O_d(k \log n/\boldsymbol{\varepsilon}^d)$
[Har-Peled, Kushal 2005]	$O_d(k/\boldsymbol{\varepsilon}^d)$
[Chen 2009]	$O(k^2 d \log n/\boldsymbol{\varepsilon}^2)$
[Langberg, Schulman, 2010]	$\tilde{O}(k^3 d^2/\boldsymbol{\varepsilon}^2)$
[Feldman, Langberg, 2011], [Braverman, Feldman, Lang, 2016]	$\tilde{O}(kd/\boldsymbol{\varepsilon}^2)$

Main Result Presented in This Talk

Main Result [Sohler, W, FOCS 2018]

- There is a coreset with **for all** guarantee for the k-median problem with a number of points that is independent of n and d.

Two Steps

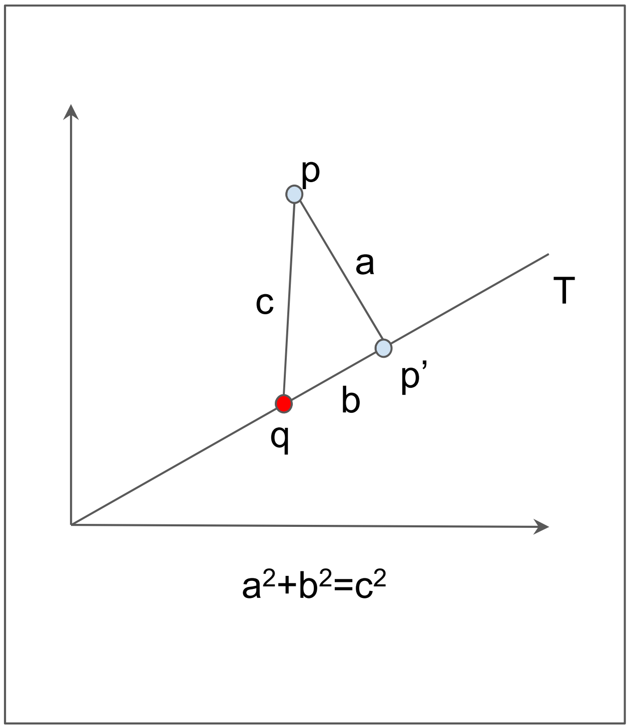
- Dimensionality reduction for k-median [new]
  - Reduces Dimensionality of input point set to $O(k/\varepsilon^2)$
- Apply existing coreset construction on the reduced input set

Main Result of This Talk

Outline

- Will first present a new proof of an earlier result of [Feldman, Schmidt, Sohler, SODA 2013]
- Will discuss why their approach does not work for k-median
- Will discuss our main new idea

Warmup: Pythagorean Theorem



Pythagorean Theorem

- Let T be a subspace containing a point q
- Let p’ be the projection of p onto T
- dist(p,p’)=a
- dist(p’,q)=b
- dist(p,q)=c
- $\text{dist}^2(p,q) = \text{dist}^2(p,p') + \text{dist}^2(p',q)$

Dimensionality Reduction

DimReduction()

1. Let Opt be the cost of the optimal k-means clustering
2. Compute optimal k-dimensional subspace S for minimizing sum of squares of distances
3. While we can add k dimensions to S to reduce the cost of the subspace approximation problem by ϵ^2 Opt
    a. Let S be the best such subspace
4. Return the projection of P on S and Δ its projection cost

Dimensionality Reduction

DimReduction()

1. Let Opt be the cost of the optimal k-means clustering
2. Compute optimal k-dimensional subspace S for minimizing sum of squares of distances
3. While we can add k dimensions to S to reduce the cost of the subspace approximation problem by ϵ^2 Opt

   This is the “k-means Opt”

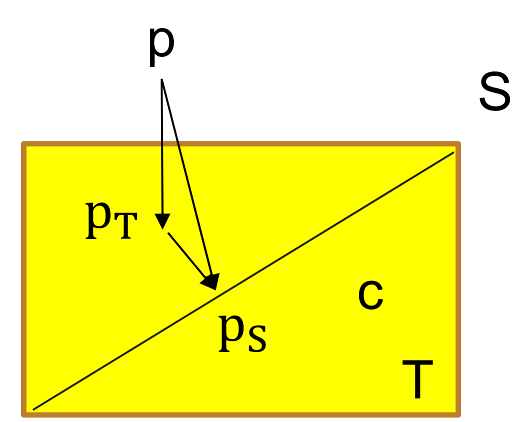
   a. Let S be the best such subspace
4. Return the projection of P on S and Δ its projection cost

Dimensionality Reduction

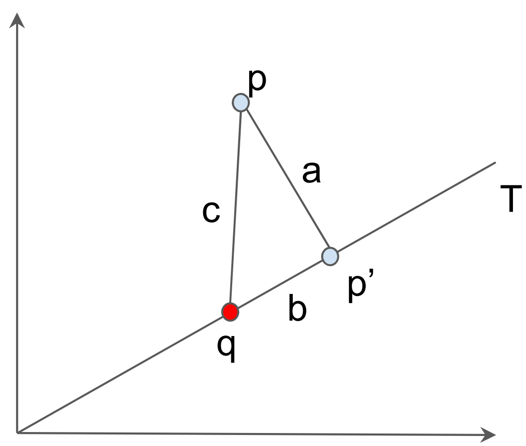
Analysis

$$\text{cost}(P,C) = \text{cost}(P,T) + \text{cost}(P_T,C) \approx \text{cost}(P,S) + \text{cost}(P_S,C)$$

- T is span of C and S
- P_T is projection of P on T
- P_S is projection of P on S



k-Means and Subspace Approximation



$a^2+b^2=c^2$

Idea

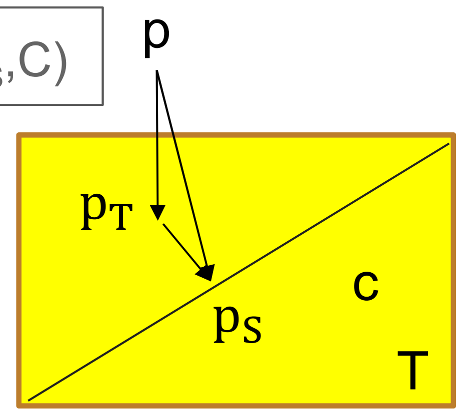
- Split the k-means cost into two parts
  - Cost of projecting on a subspace T
  - And cost within the subspace
  - T will contains set of centers C and is used only for analysis
- Find a subspace S that approximates all T
  - The projections on S should be close to the projections on T
  - T should contain S

Dimensionality Reduction

T improves S by at most ε^2 Opt

Analysis

$$\text{cost}(P,C) = \text{cost}(P,T) + \text{cost}(P_T,C) \approx \text{cost}(P,S) + \text{cost}(P_S,C)$$



- T is span of C and S
- P_T is projection of P on T
- P_S is projection of P on S

- Since S ⊆ T, the squared distance of a point p to S is the sum of squared distances of p to T and of p_T to p_S: cost(P, S) = cost(P, T) + cost(P_T, P_S)

$$\text{cost}(P_T, P_S) = \text{cost}(P,S) - \text{cost}(P,T) \le \epsilon^2 \text{ OPT}$$

If cost(P_T, P_S)$\le \epsilon^2$OPT, can show $|\text{cost}(P_S,C) - \text{cost}(P_T, C)| \le \epsilon \text{ OPT}$

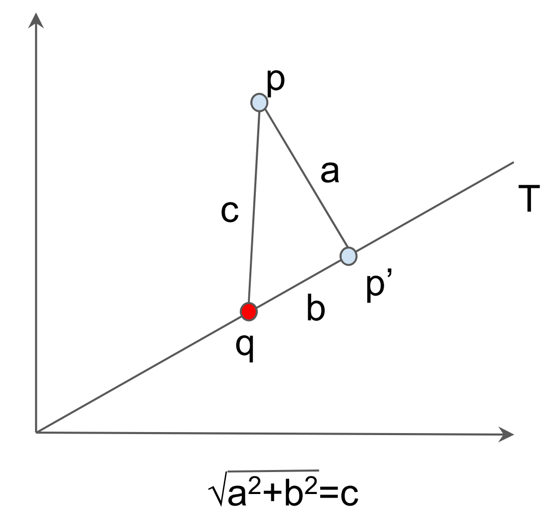
The k-Means Case

Theorem [Feldman, Schmidt, Sohler, 2013]

- Let A be a matrix storing n points from R^{d} as its rows. Let A_m be its m-rank approximation for some $m=O(k/\boldsymbol{\varepsilon}^2)$. Then there is a constant $\Delta=||A-A_m||_F^2$ such that for all sets of centers C

$$(1-\boldsymbol{\varepsilon})\ \mathrm{cost}(A,C) \ \leq\ \mathrm{cost}(A_m,C) + \Delta \ \leq\ (1+\boldsymbol{\varepsilon})\ \mathrm{cost}(A,C)$$

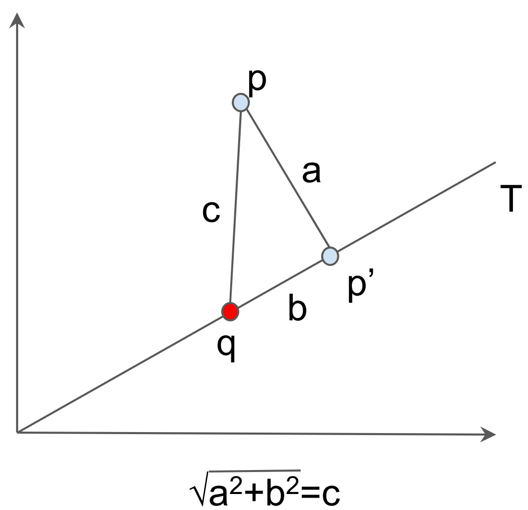
The k-Median Case



Still True

- If the distance from p to a point q in T is close to dist(p,T), then q is close to the projection of p onto T

The k-Median Case



Problem

- Cannot split cost into cost of projection and cost within subspace

The k-Median Case

Cannot hope for k-means type guarantee like

$$(1-\boldsymbol{\varepsilon})\ \mathrm{cost}(P,C) \leq \mathrm{cost}(P_S,C) + \Delta \leq (1+\boldsymbol{\varepsilon})\ \mathrm{cost}(P,C)$$

Counter Example (1-median)

- P is random from high dimensional unit ball centered at origin
- Project on m-dimensional subspace S
- If d>>m then projected points will all have tiny norm and Δ must be close to n in case our query center is (0, 0, …, 0)
- However, a center at (1,0,...,0) has cost roughly $\sqrt{2}$n for P, but $\mathrm{cost}(P_S,C) + \Delta$ is close to 2n

The k-Median Case

The Solution

- Add an extra special dimension to the projected points that is equal to distance to subspace S
- Compute coreset for this low dimensional point set
- Map points in C into new space by setting special dimension to 0

Dimensionality Reduction for k-Median

DimReduction()

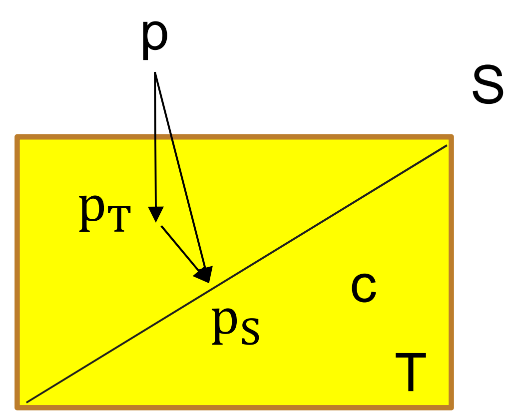
1. Let Opt be the cost of the optimal k-median clustering
2. Compute optimal k-dimensional subspace S for minimizing sum of distances
3. While we can add k dimensions to S to reduce the cost of the subspace approximation problem by $\boldsymbol{\varepsilon}^2$ Opt

   Let S be the best such subspace
4. For each point p in P,

   (a) compute its distance $\mathrm{d}(\mathrm{p}_S, \mathrm{p})$ to subspace S

   (b) return $(\mathrm{p}_S,\ \mathrm{d}(\mathrm{p}, \mathrm{p}_S))$

Analysis



- Let T be the space containing query centers C and S
- Lemma (Close Projections): cost(P_T, P_S) $\leq \epsilon$ ·OPT
- Proof: If Q = {p for which $d(p_T, p_S) \leq \epsilon\, d(p, p_S)$}, then $\sum_{p \in Q} d(p_T, p_S) \leq \epsilon\, \mathrm{OPT}$

  Also, $d(p_T, p_S) = (d(p, p_S)^2 - d(p, p_T)^2)^{1/2}$ and $d(p, p_S) \geq d(p, p_T)$ and so

  $$d(p_T, p_S) = (d(p, p_S)^2 - d(p, p_T)^2)^{1/2} = \left(\left(d(p, p_S) - d(p, p_T)\right)\left(d(p, p_S) + d(p, p_T)\right)\right)^{\frac{1}{2}}$$

  which if $d(p_T, p_S) \geq \epsilon \cdot d(p, p_S)$ is at most $\left(\frac{\left(d(p,p_S) - d(p,p_T)\right)^2}{\epsilon^2}\right)^{\frac{1}{2}} \leq \frac{d(p,p_S) - d(p,p_T)}{\epsilon}$

Analysis

- Lemma: (Distance To Subspace) cost(P, S) – cost(P, T) $\leq \epsilon^2$ OPT
- Proof:
    - Definition of algorithm

- Let $p \in P$, and c_p be p's closest center in C
- Lemma: (Distance Inside Subspace) $\sum_p |(d(p_T, c_p) - d(p_S, c_p))| \leq \epsilon$ OPT
- Proof:
    - $d(p_S, c_p) \leq d(p_S, p_T) + d(p_T, c_p)$
    - Hence, $\sum_p |(d(p_T, c_p) - d(p_S, c_p))| \leq \text{cost}(P_S, P_T) \leq \epsilon\, \text{OPT}$

Putting it All Together

- $d(p, c_p) = \left(d(p, p_T)^2 + d\left(p_T, c_p\right)^2\right)^{1/2}$
- $d((p_S, d(p, p_S)), (c_p, 0)) = \left(d(p, p_S)^2 + d\left(p_S, c_p\right)^2\right)^{1/2}$

- $\sum_p \; |d(p, c_p) - d((p_S, d(p, p_S)), (c_p, 0))|$ is small since
  - cost(P, S) ≈ cost(P, T) by Distance to Subspace Lemma
  - $\sum_p |(d(p_T, c_p) - d(p_S, c_p))|$ is small by Distance Inside Subspace Lemma

- $d(p,c_p) = \left(d(p,p_T)^2 + d\left(p_T, c_p\right)^2\right)^{1/2}$
- $d((p_S,\ d(p,p_S)),\ (c_p,0)) = \left(d(p,p_S)^2 + d\left(p_S, c_p\right)^2\right)^{1/2}$
- $|d(p,\ c_p) - d((p_S,\ d(p,p_S)),\ (c_p,0))|$

$$= |\left(d(p,p_T)^2 + d\left(p_T, c_p\right)^2\right)^{1/2} - \left(d(p,p_S)^2 + d\left(p_S, c_p\right)^2\right)^{1/2}|$$

$$= ||(d(p,p_T),\quad d\left(p_T, c_p\right))\ \ |_2 - \left|d(p,p_S), d\left(p_S, c_p\right)\right|_2|$$

$$\leq \left|d(p,p_T) - d(p,p_S), d\left(p_T, c_p\right) - d\left(p_S, c_p\right)\right|_2$$

$$\leq \left|d(p,p_T) - d(p,p_S), d\left(p_T, c_p\right) - d\left(p_S, c_p\right)\right|_1$$

$$= |d(p,p_T) - d(p,p_S)| + |d\left(p_T, c_p\right) - d\left(p_S, c_p\right)|$$

Distance to subspace + Distance inside subspace

Sum over $p \in P$ and get $\leq 2\epsilon \cdot \mathrm{OPT}$

The k-Median Case

The Solution

- Add an extra special dimension to the projected points that is equal to distance to subspace S
- Compute coreset for this low dimensional point set
- Map input space into new space by setting special dimension to 0

Result [Sohler, W, 2018]

- Coreset of size $O(k^2 \log k/\varepsilon^4)$ by combining dimensionality reduction with [Feldman, Langberg, 2011] or [Braverman, Feldman, Lang, 2016]

Summary

New Dimensionality Reduction Technique for

- k-median
- Subspace approximation
- Any other problem where centers fit into low dimensional subspace

...yields...

- New coresets for k-median and subspace approximation of size independent of n and d

Further Results

Subspace Approximation

- Same ideas yield coreset of size poly(k/ε) for subspace approximation with sum of distances error
- Can compute coreset in almost input sparsity time