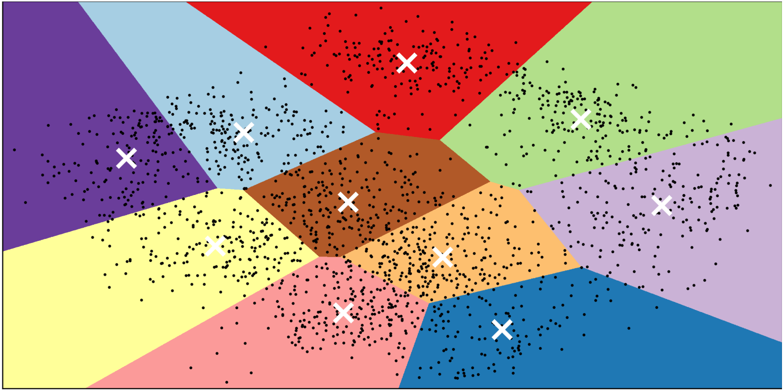
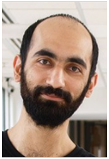


Replicable Clustering

Hossein Esfandiari, Amin Karbasi, Vahab Mirrokni, Grigoris Velegkas, Felix Zhou

Yale, Google Research

Personnel

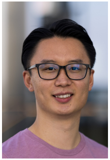


Hossein Esfandiari
(Google)

Amin Karbasi
(Yale, Google)

Vahab Mirrokni
(Google)

Grigoris Velegkas
(Yale)

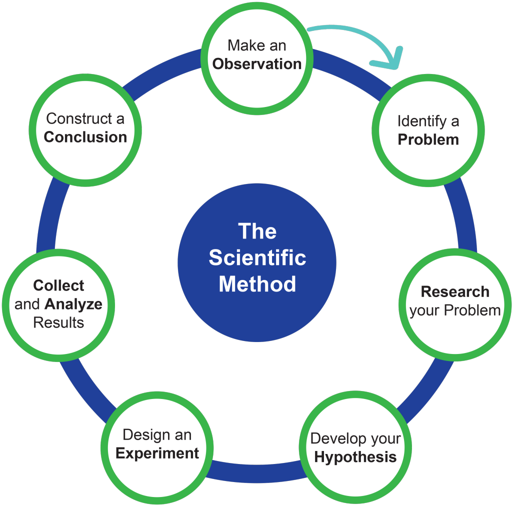


Felix Zhou
(Yale)

Table of Contents

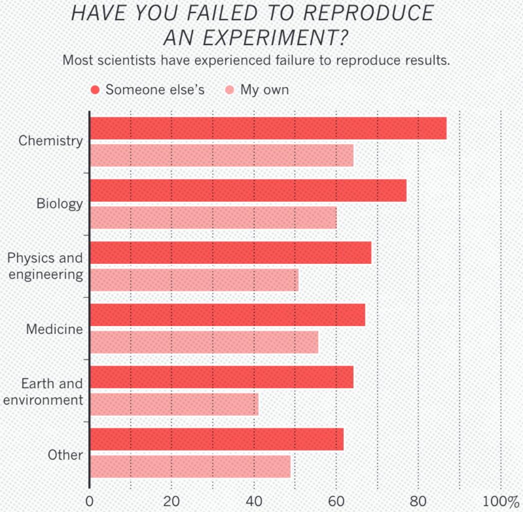
Table of Contents

Replicability



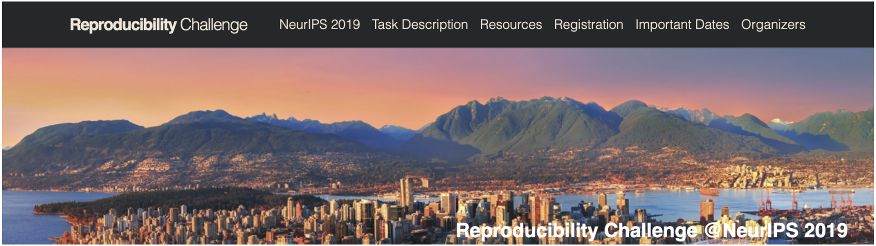
Reproducibility Crisis

"1500 Scientists Lift the Lid on Reproducibility."
Nature (2016)



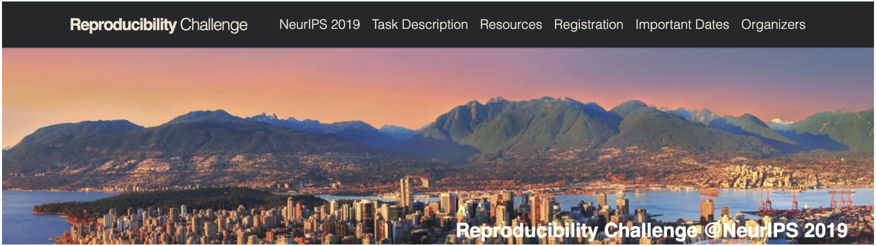
Yale

Reproducibility Crisis



- 2019 NeurIPS/ICLR Reproducibility Challenge (github.com/reproducibility-challenge)

Reproducibility Crisis



- 2019 NeurIPS/ICLR Reproducibility Challenge (github.com/reproducibility-challenge)
- Ongoing ML Reproducibility Challenge (paperswithcode.com/rc2022)

Trying to develop agreed-upon set of replicable practices is difficult.

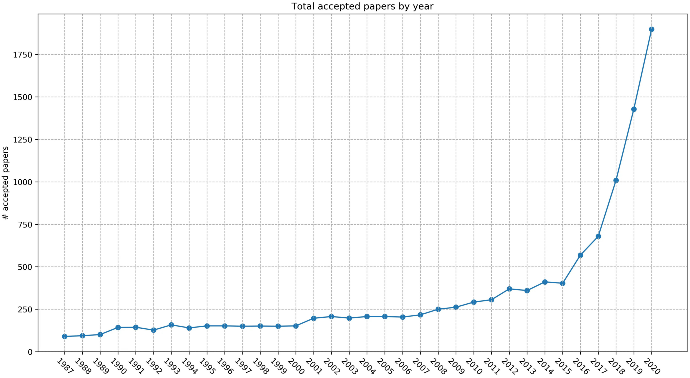


Figure: Number of accepted NeurIPS papers by year.

Reproducibility Crisis

Algorithmic Replicability

Goal: Design ML algorithms with replicability as theoretical guarantee.

Initiated by [Impagliazzo, Lei, Pitassi, and Sorell '22] (STOC'22).

Algorithmic Replicability

- X data domain

Algorithmic Replicability

- X data domain
- $\mathcal{D}$ distribution over X

Algorithmic Replicability

- X data domain
- $\mathcal{D}$ distribution over X
- $S_1, S_2 \sim_{i.i.d.} \mathcal{D}^n$ datasets of size n

Algorithmic Replicability

- X data domain
- $\mathcal{D}$ distribution over X
- $S_1, S_2 \sim_{i.i.d.} \mathcal{D}^n$ datasets of size n
- ξ uniformly random binary string

Algorithmic Replicability

- X data domain
- $\mathcal{D}$ distribution over X
- $S_1, S_2 \sim_{i.i.d.} \mathcal{D}^n$ datasets of size n
- ξ uniformly random binary string

DEFINITION (REPLICABLE ALGORITHM; [ILPS '22])

A randomized algorithm $\mathcal{A} : X^n \to Y$ is *ρ-replicable* if

$$\Pr_{S_1, S_2, \xi}[\mathcal{A}(S_1; \xi) = \mathcal{A}(S_2; \xi)] \geq 1 - \rho.$$

Algorithmic Replicability

- X data domain
- $\mathcal{D}$ distribution over X
- $S_1, S_2 \sim_{i.i.d.} \mathcal{D}^n$ datasets of size n
- ξ uniformly random binary string

DEFINITION (REPLICABLE ALGORITHM; [ILPS '22])

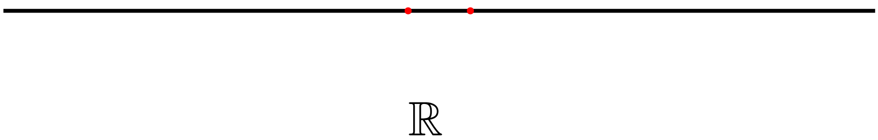
A randomized algorithm $\mathcal{A} : X^n \to Y$ is *ρ-replicable* if

$$\Pr_{S_1, S_2, \xi}[\mathcal{A}(S_1; \xi) = \mathcal{A}(S_2; \xi)] \geq 1 - \rho.$$

Remark: Replicability is trivial to obtain by itself!

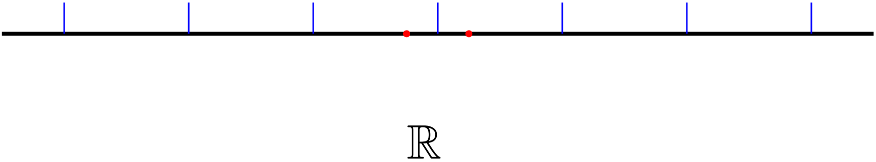
Example: Replicable Mean Estimation

1. Concentration of Measure



Example: Replicable Mean Estimation

1. Concentration of Measure
2. Discretization



Example: Replicable Mean Estimation

1. Concentration of Measure
2. Discretization
3. Shared Random Offset

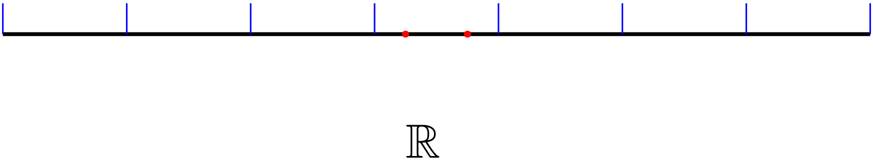


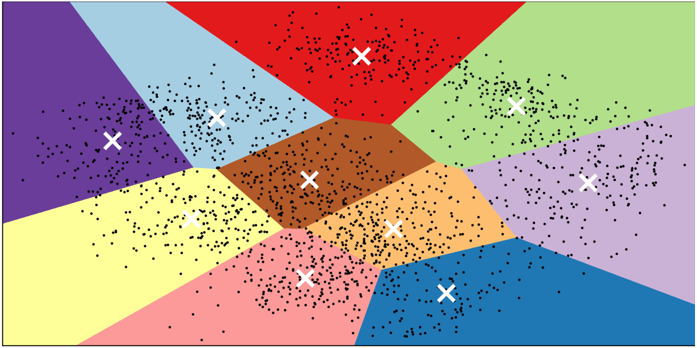
Table of Contents

Statistical k-Means

- **Given:** Sample access to distribution $\mathcal{D}$ over $[0,1]^d$

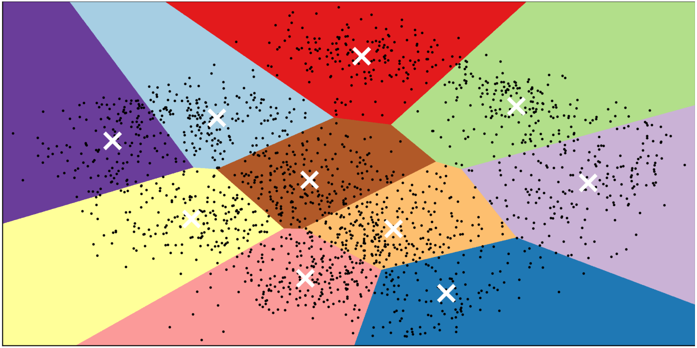
Statistical k-Means

- **Given:** Sample access to distribution $\mathcal{D}$ over $[0,1]^d$
- **Want:** Choose k centers $y_1, \ldots, y_k \in [0,1]^d$ minimizing expected "cost of travel" to nearest center



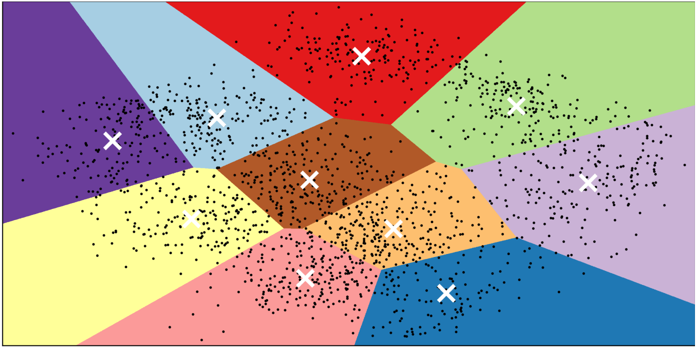
Statistical k-Means

- **Given:** Sample access to distribution $\mathcal{D}$ over $[0,1]^d$
- **Want:** Choose k centers $y_1, \ldots, y_k \in [0,1]^d$ minimizing expected "cost of travel" to nearest center
- Solve $\mathrm{argmin}_{y_1,\ldots,y_k \in [0,1]^d} \mathbb{E}_{X \sim \mathcal{D}} \left[\min_{j \in [k]} \|X - y_j\|_2^2 \right]$



Statistical k-Means

- **Given:** Sample access to distribution $\mathcal{D}$ over $[0,1]^d$
- **Want:** Choose k centers $y_1, \ldots, y_k \in [0,1]^d$ minimizing expected "cost of travel" to nearest center
- Solve $\operatorname{argmin}_{y_1,\ldots,y_k \in [0,1]^d} \mathbb{E}_{X \sim \mathcal{D}} \left[\min_{j \in [k]} \|X - y_j\|_2^2 \right]$
- Sample complexity? Time complexity?

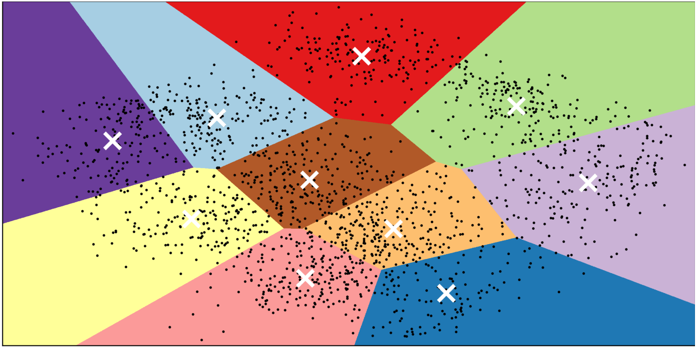


Sample k-Means

- **Given:** Points $x_1, \ldots, x_n \in [0,1]^d$

Sample k-Means

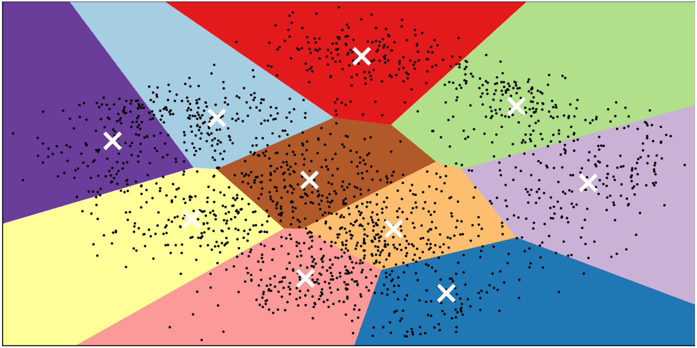
- **Given:** Points $x_1, \ldots, x_n \in [0,1]^d$
- **Want:** Choose k centers $y_1, \ldots, y_k \in [0,1]^d$ minimizing average "cost of travel" to nearest center



Yale

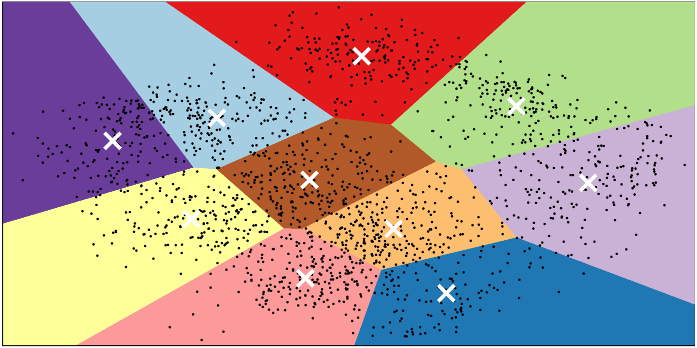
Sample k-Means

- **Given:** Points $x_1, \ldots, x_n \in [0,1]^d$
- **Want:** Choose k centers $y_1, \ldots, y_k \in [0,1]^d$ minimizing average "cost of travel" to nearest center
- Solve $\operatorname{argmin}_{y_1,\ldots,y_k \in [0,1]^d} \frac{1}{n} \sum_{i \in [n]} \left[\min_{j \in [k]} \|x_i - y_j\|_2^2 \right]$



Sample k-Means

- **Given:** Points $x_1, \ldots, x_n \in [0,1]^d$
- **Want:** Choose k centers $y_1, \ldots, y_k \in [0,1]^d$ minimizing average "cost of travel" to nearest center
- Solve $\mathrm{argmin}_{y_1,\ldots,y_k \in [0,1]^d} \frac{1}{n} \sum_{i \in [n]} \left[\min_{j \in [k]} \|x_i - y_j\|_2^2 \right]$
- Time complexity?



Yale

- Clustering algorithms reveal properties of the underlying population

Replicability in Clustering

- Clustering algorithms reveal properties of the underlying population
- Replicability is an important property for downstream applications

Replicable Statistical k-Means

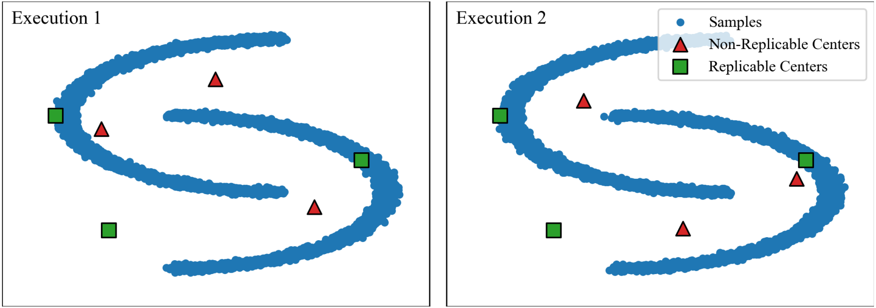
- **Given:** Sample access to distribution $\mathcal{D}$ over $[0, 1]^d$

Yale

Replicable Statistical k-Means

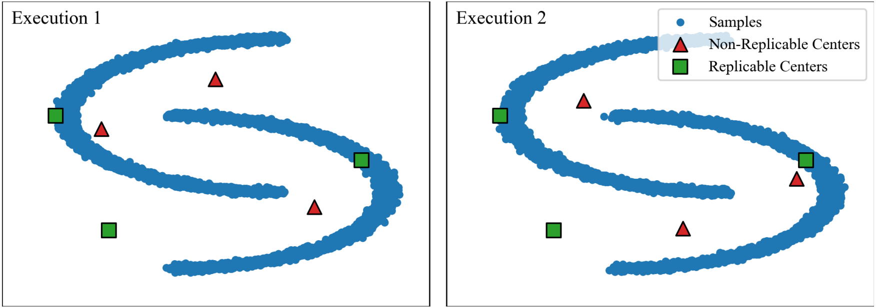
- **Given:** Sample access to distribution $\mathcal{D}$ over $[0,1]^d$
- **Want:** $\mathrm{argmin}_{y_1,\ldots,y_k \in [0,1]^d} \mathbb{E}_{X \sim \mathcal{D}} \left[\min_{j \in [k]} \|X - y_j\|_2^2\right]$

- **Given:** Sample access to distribution $\mathcal{D}$ over $[0,1]^d$
- **Want:** $\operatorname{argmin}_{y_1,\dots,y_k \in [0,1]^d} \mathbb{E}_{X\sim\mathcal{D}} \left[\min_{j\in[k]} \|X - y_j\|_2^2\right]$
- **And:** $\Pr_{X,X',\xi}\left[\{y_1,\dots,y_k\} = \{y_1',\dots,y_k'\}\right] \geq 1-\rho$



Replicable Statistical k-Means

- **Given:** Sample access to distribution $\mathcal{D}$ over $[0,1]^d$
- **Want:** $\text{argmin}_{y_1,\ldots,y_k \in [0,1]^d} \mathbb{E}_{X\sim\mathcal{D}} \left[\min_{j\in[k]} \|X - y_j\|_2^2\right]$
- **And:** $\Pr_{X,X',\xi}\left[\{y_1,\ldots,y_k\} = \{y'_1,\ldots,y'_k\}\right] \geq 1-\rho$
- Sample Complexity? Time Complexity?



Replicable Statistical k-Means

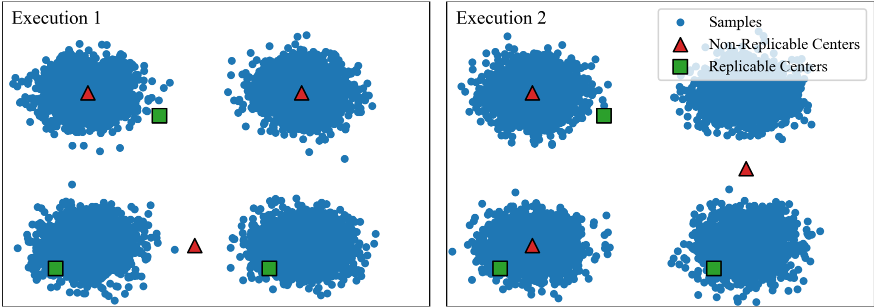


Table of Contents

Yale

Our Results

THEOREM (EKMVZ '23)

- Assume black-box polynomial time β-approximation oracle for sample k-means.

Our Results

THEOREM (EKMVZ '23)

- Assume black-box polynomial time β-approximation oracle for sample k-means.
- There is a ρ-replicable algorithm $\mathcal{A}$ for statistical k-means:

Yale

Our Results

THEOREM (EKMVZ '23)

- Assume black-box polynomial time β-approximation oracle for sample k-means.
- There is a ρ-replicable algorithm $\mathcal{A}$ for statistical k-means:
- with high probability, the cost of the solution is at most $(1+\varepsilon)\beta \cdot \mathrm{OPT}$.

Our Results

THEOREM (EKMVZ '23)

- Assume black-box polynomial time β-approximation oracle for sample k-means.
- There is a ρ-replicable algorithm $\mathcal{A}$ for statistical k-means:
- with high probability, the cost of the solution is at most $(1+\varepsilon)\beta \cdot \mathrm{OPT}$.
- $\mathcal{A}$ has time and sample complexity

$$\tilde{O}_{\rho,\varepsilon}\left(\mathrm{poly}(k,d)k^{\log\log k}\right).$$

More Precisely...

THEOREM (EKMVZ '23)

- Assume black-box polynomial time β-approximation oracle for sample k-means.
- There is a ρ-replicable algorithm $\mathcal{A}$ for statistical k-means:
- with probability at least $1 - \delta$, the cost of the solution is at most $(1 + \varepsilon)\beta \cdot \mathrm{OPT}$.
- $\mathcal{A}$ has time and sample complexity

$$\tilde{O}\left(\mathrm{poly}(k, d, 1/\rho)(2\sqrt{m}/\varepsilon)^{O(m)} \log \frac{1}{\delta}\right).$$

where $m = O(\varepsilon^{-2} \log k/\delta\varepsilon)$.

Overview of Techniques

1. Reduce problem on distribution to problem on samples

Overview of Techniques

1. Reduce problem on distribution to problem on samples
   - Uniform Law of Large Numbers

Overview of Techniques

1. Reduce problem on distribution to problem on samples
   - Uniform Law of Large Numbers
2. Consolidate multiple points into weighted point

Overview of Techniques

1. Reduce problem on distribution to problem on samples
   - Uniform Law of Large Numbers
2. Consolidate multiple points into weighted point
   - Replicable Coreset Construction

Overview of Techniques

0. Data-Oblivious Dimensionality Reduction

1. Reduce problem on distribution to problem on samples
   - Uniform Law of Large Numbers
2. Consolidate multiple points into weighted point
   - Replicable Coreset Construction

Overview of Techniques

0. Data-Oblivious Dimensionality Reduction
    - Johnson-Lindenstrauss transform
1. Reduce problem on distribution to problem on samples
    - Uniform Law of Large Numbers
2. Consolidate multiple points into weighted point
    - Replicable Coreset Construction

Table of Contents

Statistical k-Means

THEOREM (BEN-DAVID '07)

- Assume black-box polynomial time β-approximation oracle for sample k-means.
- There is an algorithm $\mathcal{A}$ for statistical k-means:
- with high probability, the cost of the solution is at most $(1+\varepsilon)\beta \cdot (4\,\mathrm{OPT})$.
- $\mathcal{A}$ has time and sample complexity

$$\tilde{O}\left(\mathrm{poly}(d, k, 1/\varepsilon)\right).$$

Stability in Statistical k-Means

- **Stable choice of k:** Produce clusterings that do not vary much from one sample to another. [BEG '01], [LRBB '04], [VB '05], [B '06], [RC '06], [V '10]

Stability in Statistical k-Means

- **Stable choice of k:** Produce clusterings that do not vary much from one sample to another. [BEG '01], [LRBB '04], [VB '05], [B '06], [RC '06], [V '10]
- [Ben-David, Pál, Simon; '07]: "for large sample sizes, stability is fully determined by the symmetry within the data."

Sample k-Means

- **Inspiration:** operations research and statistics [Lloyd '57], [Hakimi '64], [Steinhaus '57]

Sample k-Means

- **Inspiration:** operations research and statistics [Lloyd '57], [Hakimi '64], [Steinhaus '57]
- **Sample Metric k-Means:** polynomial time 9-approximation [ANS '19], $(1 + 8/e)$-approximation is NP-hard [CGKLL '19]

Sample k-Means

- **Inspiration:** operations research and statistics [Lloyd '57], [Hakimi '64], [Steinhaus '57]
- **Sample Metric k-Means:** polynomial time 9-approximation [ANS '19], $(1 + 8/e)$-approximation is NP-hard [CGKLL '19]
- **Sample Euclidean k-Means:** polynomial time 5.912-approximation [CEMN '22], 1.07-approximation is NP-hard [CK '19], [CLK '22]

Sample k-Means

- **Inspiration:** operations research and statistics [Lloyd '57], [Hakimi '64], [Steinhaus '57]
- **Sample Metric k-Means:** polynomial time 9-approximation [ANS '19], $(1 + 8/e)$-approximation is NP-hard [CGKLL '19]
- **Sample Euclidean k-Means:** polynomial time 5.912-approximation [CEMN '22], 1.07-approximation is NP-hard [CK '19], [CLK '22]
- **Dimensionality Reduction** Johnson-Lindenstrauss Transform [Johnson '84], [KKM '19]

Sample k-Means

- **Inspiration:** operations research and statistics [Lloyd '57], [Hakimi '64], [Steinhaus '57]
- **Sample Metric k-Means:** polynomial time 9-approximation [ANS '19], $(1 + 8/e)$-approximation is NP-hard [CGKLL '19]
- **Sample Euclidean k-Means:** polynomial time 5.912-approximation [CEMN '22], 1.07-approximation is NP-hard [CK '19], [CLK '22]
- **Dimensionality Reduction** Johnson-Lindenstrauss Transform [Johnson '84], [KKM '19]
- **Coresets** geometric coresets [FS '05], sampling-based coresets [HSYZ '18], survey [SW '18]

Replicability

- **Algorithm Design:** statistical queries, heavy-hitters, medians, learning halfspaces [ILPS '22], stochastic bandits [EKKKMV '23], reinforcement learning [KYGZ '23], [EHKS '23]

Replicability

- **Algorithm Design:** statistical queries, heavy-hitters, medians, learning halfspaces [ILPS '22], stochastic bandits [EKKKMV '23], reinforcement learning [KYGZ '23], [EHKS '23]
- **Learning Theory:** equivalence with DP [BGHILPSS '23], statistical indistinguishability [KKMV '23], list-replicability [CMY '23], [DPWV '23]

Table of Contents

Replicable Coreset Estimation

Intuition: Little hope for replicability with continuous distributions, need to discretize and round similar to mean estimation.

Idea: Replicably approximate the input distribution with a finite (discrete) distribution.

Replicable Coreset Estimation

For some $R : \mathcal{X} \to \mathcal{X}$ with small $|R(\mathcal{X})|$, uniformly approximate

$$\mathrm{cost}(y) := \mathbb{E}_X \left[\min_j \|X - y_j\|_2^2 \right]$$

Replicable Coreset Estimation

For some $R : \mathcal{X} \to \mathcal{X}$ with small $|R(\mathcal{X})|$, uniformly approximate

$$\begin{aligned}\mathrm{cost}(y) &:= \mathbb{E}_X \left[\min_j \|X - y_j\|_2^2\right] \\ &\approx \mathbb{E}_X \left[\min_j \|R(X) - y_j\|_2^2\right]\end{aligned}$$

Yale

Replicable Coreset Estimation

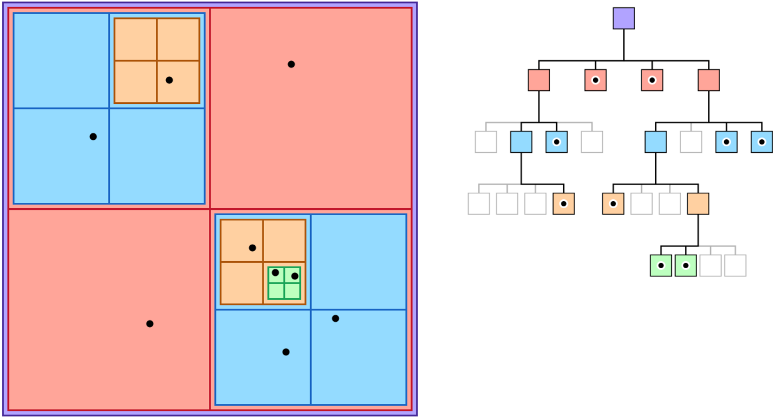
For some $R : \mathcal{X} \to \mathcal{X}$ with small $|R(\mathcal{X})|$, uniformly approximate

$$\begin{aligned}
\mathrm{cost}(y) &:= \mathbb{E}_X \left[\min_j \|X - y_j\|_2^2 \right] \\
&\approx \mathbb{E}_X \left[\min_j \|R(X) - y_j\|_2^2 \right] \\
&= \sum_{z \in R(\mathcal{X})} \min_j \|z - y_j\|_2^2 \cdot \mathbb{P}(R^{-1}(z))
\end{aligned}$$

Yale

Quad Tree

Idea: Recursively partition $[0, 1]^d$ into subcubes and consolidate mass from entire subcube into single point



Recall: Heavy-Hitters

- **Given:** Sample access to distribution, $\nu \in [0, 1]$.

Recall: Heavy-Hitters

- **Given:** Sample access to distribution, $\nu \in [0, 1]$.
- **Want:** All elements with mass at least ν.

Recall: Heavy-Hitters

- **Given:** Sample access to distribution, $\nu \in [0, 1]$.
- **Want:** All elements with mass at least ν.

THEOREM (ILPS '22)

There is a replicable heavy-hitters algorithm.

Replicable Quad Tree

Algorithm Replicable Quad Tree

```
rQuadTree(Node Z, error ε, depth i):
if diam(Z) ≥ C₁ε then
    return
end if
for Z′ ∈ subcubes(Z) do
    // Heavy Hitters Operation!
    if ℙ(Z′) ≥ C₂ · 2^i ε/k then
        Add Z′ as child of Z
        rQuadTree(Z′, ε, i + 1)
    end if
end for
```

Yale

Summary

0. Data-Oblivious Dimensionality Reduction

Summary

0. Data-Oblivious Dimensionality Reduction

1. Uniform Law of Large Numbers

Summary

0. Data-Oblivious Dimensionality Reduction
1. Uniform Law of Large Numbers
2. Replicable Coreset Estimation

Summary

0. Data-Oblivious Dimensionality Reduction
1. Uniform Law of Large Numbers
2. Replicable Coreset Estimation
   - A series of heavy hitter estimations that can be made replicable.

Future Work

- Polynomial sample/time complexity for replicable k-means

Future Work

- Polynomial sample/time complexity for replicable k-means
- Replicable (k, p)-clustering

Future Work

- Polynomial sample/time complexity for replicable k-means
- Replicable (k, p)-clustering
- Sample complexity lower bounds for (replicable) clustering

Thank You!