Learning Augmented Algorithms for $k$-Means Clustering

Samson Zhou
Learning Augmented Algorithms for $k$-Means Clustering

Jon Ergun
Zhili Feng
Sandeep Silwal
David P. Woodruff
Samson Zhou

Thy Nguyen
Anamay Chatuvedi
Huy Lê Nguyên

[NCN23] [EFSWZ22]
29 Data-Driven Algorithm Design

Maria-Florina Balcan
29.1 Motivation and Context
29.2 Data-Driven Algorithm Design via Statistical Learning
29.3 Data-Driven Algorithm Design via Online Learning
29.4 Summary and Discussion

30 Algorithms with Predictions

Michael Mitzenmacher and Sergei Vassilvitskii
30.1 Introduction
30.2 Counting Sketches
30.3 Learned Bloom Filters
30.4 Caching with Predictions
30.5 Scheduling with Predictions
30.6 Notes
Learning-Augmented Algorithms

• For a certain task and input, algorithm is given advice

• Advice could be “good”, advice could be “bad”

• Goal: “Good” performance if the advice is good, “normal” performance if the advice is bad
Learning-Augmented Algorithms

- **Better data structures**: Bloom filters with lower false positive rates [Mitzenmacher18], Binary search [LinLuoWoodruff22]
- **Better space-accuracy tradeoff for streaming algorithms**: Frequency estimation, e.g., CountMin, CountSketch [HsuIndykKatabiVakilian19], moment estimation, distinct elements [JiangLinRuanWoodruff20], triangle counting [ChenEdenIndykLinNarayananRubinfeldSilwalWagnerWoodruffZhang22]
- **Better size-accuracy tradeoff for sketching**: Low-rank approximation [IndykVakilianYuan19]
Learning-Augmented Algorithms

- **Warm-start to search algorithms:** Max-flow [ChenSilwalVakilianZhang22], [DaviesMoseleyVassilvitskiiWang23], matchings [DinitzImLavastidaMoseleyVassilvitskii21]
- **Better accuracy-sample complexity tradeoff:** Support size estimation [EdenIndykNarayananRubinfeldSilwalWagner21]
- **Better online algorithms:** Set cover [BamasMaggioriSvensson20], [GrigorescuLinSilwalSongZhou23], Scheduling [LattanziLavastidaMoseleyVassilvitskii20], [ScullyGrosofMitzenmacher22]
- **Better privacy-utility tradeoffs for DP:** Quantile estimation [KhodakAminDickVassilvitskii23]
- **Beating NP-hardness?**
Graph Searching with Predictions  Banerjee, Cohen-Addad, Gupta, Li  arXiv ’22
Scheduling with Predictions  Cho, Henderson, Shmoys  arXiv ’22
On the Power of Learning-Augmented BSTs  Chen, Chen  arXiv ’22
Algorithms with Prediction Portfolios  Dinitz, Im, Lavastida, Moseley, Vassilvitskii
Private Algorithms with Private Predictions  Amin, Dick, Khodak, Vassilvitskii
Paging with Succinct Predictions  Antoniadis, Boyar, Eliás, Favrholdt, Hoeksma, Larsen, Polak, Simon
Proportionally Fair Online Allocation of Public Goods with Predictions  Banerjee, Gkatzelis, Hossain, Jin, Micha, Shah
Canadian Traveller Problem with Predictions  Bampis, Escoffier, Xefteris  arXiv ’22 WAOA ’22
Learning-Augmented Algorithms for Online Linear and Semidefinite Programming  Grigorescu, Lin, Silwal, Song, Zhou
Clustering

- **Goal**: Given input dataset $X$, partition $X$ so that “similar” points are in the same cluster and “different” points are in different clusters.
$k$-Clustering

• **Goal**: Given input dataset $X$, partition $X$ so that “similar” points are in the same cluster and “different” points are in different clusters.

• There can be at most $k$ different clusters.

$k = 3$
$k$-Clustering

- **Question:** How do we measure the “quality” of each clustering?

$k = 3$
**k-Clustering**

- **Question**: How do we measure the “quality” of each clustering?

- Assign a “center” $c_i$ to each cluster

- Have a cost function induced by $c_i$ for all of the points $P_i$ assigned to cluster $i$
**k-Clustering**

- **Question**: How do we measure the “quality” of each clustering?

- Assign a “center” $c_i$ to each cluster

- Have a cost function induced by $c_i$ for all of the points $P_i$ assigned to cluster $i$
  - Assume points are in metric space with distance function $\text{dist}(\cdot, \cdot)$
  - Define $\text{Cost}(P_i, c_i)$ to be a function of $\{\text{dist}(x, c_i)\}_{x \in P_i}$
$k$-Clustering

• **Question**: How do we measure the “quality” of each clustering?

• Have a cost function induced by $c_i$ for all of the points $P_i$ assigned to cluster $i$
  • Define $\text{Cost}(P_i, c_i)$ to be a function of $\{\text{dist}(x, c_i)\}_{x \in P_i}$

• Suppose the set of centers is $C = \{c_1, ..., c_k\}$
  • Define clustering cost $\text{Cost}(X, C)$ to be a function of $\{\text{dist}(x, C)\}_{x \in C}$
$k$-Clustering

• Define clustering cost $\text{Cost}(X, C)$ to be a function of \( \{\text{dist}(x, C)\}_{x \in C} \)
$k$-Clustering

- Define clustering cost $\text{Cost}(X, C)$ to be a function of $\{\text{dist}(x, C)\}_{x \in C}$

- $k$-center: $\text{Cost}(X, C) = \max_{x \in X} \text{dist}(x, C)$
$k$-Clustering

• Define clustering cost $\text{Cost}(X, C)$ to be a function of $\{\text{dist}(x, C)\}_{x \in C}$

• $k$-center: $\text{Cost}(X, C) = \max_{x \in X} \text{dist}(x, C)$

• $k$-median: $\text{Cost}(X, C) = \sum_{x \in X} \text{dist}(x, C)$
Define clustering cost $\text{Cost}(X, C)$ to be a function of $\{\text{dist}(x, C)\}_{x \in C}$

- $k$-center: $\text{Cost}(X, C) = \max_{x \in X} \text{dist}(x, C)$
- $k$-median: $\text{Cost}(X, C) = \sum_{x \in X} \text{dist}(x, C)$
- $k$-means: $\text{Cost}(X, C) = \sum_{x \in X} (\text{dist}(x, C))^2$
**$k$-Clustering**

- Define clustering cost $\text{Cost}(X, C)$ to be a function of $\{\text{dist}(x, C)\}_{x \in C}$

- **$k$-center**: $\text{Cost}(X, C) = \max_{x \in X} \text{dist}(x, C)$

- **$k$-median**: $\text{Cost}(X, C) = \sum_{x \in X} \text{dist}(x, C)$

- **$k$-means**: $\text{Cost}(X, C) = \sum_{x \in X} (\text{dist}(x, C))^2$

- **($k, z$)-clustering**: $\text{Cost}(X, C) = \sum_{x \in X} (\text{dist}(x, C))^z$
Euclidean $k$-Clustering

• For Euclidean $k$-clustering, input points $X = x_1, \ldots, x_n$ are in $\mathbb{R}^d$ (for us, they will be in $[\Delta]^d := \{1,2, \ldots, \Delta\}^d$)

• $\text{dist}(x, y) = \|x - y\| = \sqrt{(x_1 - y_1)^2 + \cdots + (x_d - y_d)^2}$ is the Euclidean distance

• $(k, z)$-clustering problem:

$$\min_{C:|C| \leq k} \text{Cost}(X, C) = \min_{C:|C| \leq k} \sum_{x \in X} (\text{dist}(x, C))^z$$
Learning-Augmented Clustering

- **Goal:** Given dataset $X$ in $d$ dimensions, output a set $C$ of $k$ centers to minimize

$$\sum_{x \in X} \min_{c \in C} \|x - c\|_2^2$$

- **NP-hard** to even approximate within a factor of 1.07 [Cohen-AddadC.S.20, LeeSchmidtWright17]

- **Beyond worst-case:** Clustering on inputs from some “nice” distribution, similar inputs or inputs with auxiliary information

- **Hope:** ML can guide the clustering, so we can overcome worst-case with advice!
• Suppose $\Pi$ outputs noisy labels according to a $(1 + \alpha)$ approximate clustering $C$ and error rate $\lambda \leq \alpha$

What is the label of $x_1$?

$x_1$ belongs to cluster 3

What is the label of $x_2$?

$x_2$ belongs to cluster 7
Theoretical Guarantee

• Suppose \( \Pi \) outputs noisy labels according to a \((1 + \alpha)\) approximate clustering \( C \) and error rate \( \lambda \leq \alpha \)

• Main result [EFSWZ22]: Algorithm that outputs a \((1 + O(\alpha))\) approximate \( k \)-means clustering in nearly linear time

• “Predictions can overcome complexity hardness barriers!”
Naïve Approach Does Not Work

• Not enough to blindly follow predictions!

• Optimal cost $\approx 0$

• Predictor with arbitrary small error has large cost!
Naïve Approach Does Not Work

• Can a predictor even help?
Naïve Approach Does Not Work

• Can a predictor even help?

Cluster ?

single incorrect label

Cluster 2
Naïve Approach Does Not Work

• Can a predictor even help?

• **MUST** have assumptions about the accuracy on each cluster
Precision and Recall

• [EFSWZ22]: Assume cluster sizes are “balanced”

• [NCN23]: Let $P_i$ be the optimal cluster with label $i$ and $Q_i$ be the points that are labeled $i$. Then $|Q_i \setminus P_i| + |P_i \setminus Q_i| \leq \alpha \cdot |P_i|$. 

Precision Recall
Algorithmic Intuition

- **Our approach:** Closed-form solution for best center of a *fixed* set of points

\[
\text{argmin}_c \ [\text{cost}(c, P)] = \frac{1}{|P|} \sum_{p \in P} p
\]

\[
\text{argmin}_c \sum_{p \in P} \|p - c\|_2^2 = \frac{1}{|P|} \sum_{p \in P} p
\]
Algorithmic Intuition

- Consider each *dimension* separately

**Algorithm 1** Learning-augmented $k$-means clustering

**Input:** A point set $X$ with labels given by a predictor $\Pi$ with label error rate $\lambda$

**Output:** $(1+O(\alpha))$-approximate $k$-means clustering of $X$

1. **for** $i = 1$ to $i = k$ **do**
2. Let $Y_i$ be the set of points with label $i$.
3. Run CrdEST for each of the $d$ coordinates of $Y_i$.

4. Let $C_i'$ be the coordinate-wise outputs of CrdEST.

5. **end for**

6. **Return** $C_1', \ldots, C_k'$.
Algorithmic Intuition

• Consider each label separately

**Algorithm 1** Learning-augmented $k$-means clustering

**Input:** A point set $X$ with labels given by a predictor $\Pi$ with label error rate $\lambda$

**Output:** $(1+O(\alpha))$-approximate $k$-means clustering of $X$

1: for $i = 1$ to $i = k$ do
2: Let $Y_i$ be the set of points with label $i$.
3: Run CrdEst for each of the $d$ coordinates of $Y_i$.
4: Let $C_i'$ be the coordinate-wise outputs of CrdEst.
5: end for
6: Return $C_1', \ldots, C_k'$.
Algorithmic Intuition

• Example:

Cluster 1

Cluster 2

Cluster 3
Algorithmic Intuition

- **Example**: Consider the points with predicted label 1
Algorithmic Intuition

• Example: Consider the points with predicted label 1

• Consider each dimension separately
Algorithmic Intuition

- **Example**: Consider the histogram of points with predicted label 1.
Algorithmic Intuition

• **Example**: Consider the histogram of points with predicted label $1$

• Is it true that “pruning” away the outliers removes all incorrect points?
Algorithmic Intuition

• Is it true that “pruning” away the outliers removes all incorrect points? **NO!**
Algorithmic Intuition

• Example: Consider the points with label 1
Algorithmic Intuition

• Consider each label and each dimension separately

• Our approach: Use ideas from robust mean estimation

\[(1 - \alpha)P \quad \alpha Q\]
Algorithmic Intuition

- **Case 1**: \( Q \) is “far” from \( P \)
Algorithmic Intuition

• **Case 1**: $Q$ is “far” from $P$

• Can detect handle this case by “pruning” the distribution

\[(1 - \alpha)P\] \[\alpha Q\]
Algorithmic Intuition

- **Case 2:** $Q$ is “close” to $P$
Algorithmic Intuition

- **Case 2**: $Q$ is “close” to $P$

- $Q$ cannot heavily affect the empirical mean $P$

$$(1 - \alpha)P^\alpha Q$$
Algorithmic Intuition

- **Algorithm**: Find the mean of the shortest interval that contains \((1 - O(\alpha))\) fraction of the points
**Algorithm**

- **Algorithm**: Find the mean of the shortest interval that contains $(1 - O(\alpha))$ fraction of the points

```
Algorithm 2 Coordinate-wise estimation CRDEST

Input: Points $x_1, \ldots, x_{2m} \in \mathbb{R}$, corruption level $\lambda \leq \alpha$

1: Randomly partition the points into two groups $X_1, X_2$ of size $m$.
2: Let $I = [a, b]$ be the shortest interval containing $m(1 - 5\alpha)$ points of $X_1$.
3: $Z \leftarrow X_2 \cap I$
4: $z \leftarrow \frac{1}{|Z|} \sum_{x \in Z} x$
5: Return $z$
```
Analysis Overview

• Robust mean estimation gives additive $\alpha$ error to the location of the mean

• How does this affect the $k$-means clustering cost?
Analysis Overview

• **Analysis**: Robust mean gives $(1 + \alpha)$-approximation to the $1$-means clustering cost

• **Recall**: Consider each label and each dimension separately

![Diagram showing $(1 - \alpha)P$ and $\alpha Q$](image-url)
Analysis Overview

• **Analysis**: Robust mean gives \((1 + \alpha)\)-approximation to the \(k\)-means clustering cost

• **Lemma**: Let \(P, Q\) be sets of real numbers with \(|P| \geq (1 - \alpha)n\) and \(|Q| \leq \alpha n\). Let \(X = P \cup Q\), let \(C_X\) and \(C_P\) be the means of \(X\) and \(P\). Then

\[
\text{Cost}(X, C_P) \leq (1 + \alpha)\text{Cost}(X, C_X)
\]

• [InabaKatoHImai94]:

\[
\text{Cost}(X, C_P) \leq \text{Cost}(X, C_X) + |X| \cdot |C_P - C_X|^2
\]
Algorithm 1 Learning-augmented $k$-means clustering

**Input:** A point set $X$ with labels given by a predictor $\Pi$ with label error rate $\lambda$

**Output:** $(1+\mathcal{O}(\alpha))$-approximate $k$-means clustering of $X$

1: for $i = 1$ to $i = k$ do
2: Let $Y_i$ be the set of points with label $i$.
3: Run CrdEst for each of the $d$ coordinates of $Y_i$.
4: Let $C'_i$ be the coordinate-wise outputs of CrdEst.
5: end for
6: Return $C'_1, \ldots, C'_k$.

Algorithm 2 Coordinate-wise estimation CrdEst

**Input:** Points $x_1, \ldots, x_{2m} \in \mathbb{R}$, corruption level $\lambda \leq \alpha$

1: Randomly partition the points into two groups $X_1, X_2$ of size $m$.
2: Let $I = [a, b]$ be the shortest interval containing $m(1 - 5\alpha)$ points of $X_1$.
3: $Z \leftarrow X_2 \cap I$
4: $z \leftarrow \frac{1}{|Z|} \sum_{x \in Z} x$
5: Return $z$
**Algorithm [NCN23]**

**Algorithm 1 Deterministic Learning-augmented $k$-Means Clustering**

Require: Data set $P$ of $m$ points, Partition $P = P_1 \cup \ldots P_k$ from a predictor, accuracy parameter $\alpha$

for $i \in [k]$ do

for $j \in [d]$ do

Let $\omega_{i,j}$ be the collection of all subsets of $(1 - \alpha)m_i$ contiguous points in $P_{i,j}$.

$I_{i,j} \leftarrow \arg\min_{Z \in \omega_{i,j}} \text{cost}(Z, \overline{Z}) = \arg\min_{Z \in \omega_{i,j}} \sum_{z \in Z} z^2 - \frac{1}{|Z|} \left( \sum_{z' \in Z} z' \right)^2$

end for

Let $\hat{c}_i = (I_{i,j})_{j \in [d]}$

end for

Return $\{\hat{c}_1, \ldots, \hat{c}_k\}$

“Find the best interval that contains $(1 - \alpha)$ fraction of the points”
Additional Caveats

• Assignment of each point to clusters after finding centers in \( \tilde{O}_{\epsilon, \log k}(nd) \) time

• Dimensionality reduction to projects to lower-dimension space, use approximate nearest neighbors in lower-dimension space to assign points to clusters
Limitations

• Techniques specifically catered to $k$-means clustering (coordinate-wise decomposition, robust mean estimation)

• What about $k$-median clustering?
Algorithm [NCN23]

Algorithm 2 Learning-augmented $k$-Medians Clustering

Require: Data set $P$ of $m$ points, Partition $P = P_1 \cup \ldots \cup P_k$ from a predictor, accuracy parameter $\alpha < 1/2$

for $i \in [k]$ do
    Let $R = \frac{2}{1-2\alpha} \log \frac{2k}{\delta}$
    for $j \in [R]$ do
        Sample $x \sim P_i$ u.a.r.
        Let $P'_i$ be the $\left\lceil \alpha m_i \right\rceil$ points farthest from $x$
        $\hat{c}^j_i \leftarrow$ median of $P_i \setminus P'_i$
    end for
    Let $\hat{c}_i$ be the $\hat{c}^j_i$ with minimum cost
end for
Return $\{\hat{c}_1, \ldots, \hat{c}_k\}$

“Sample, prune, and find the geometric median, e.g., [CLMPS16]”
Experimental Results

• **Case Study:** Spectral clustering on graphs varying over time

• **Dataset:** Internet router graph varying over the course of a year

• **Methodology:** Compare to standard benchmarks while using various natural predictors, i.e., noisily perturb true labels and compare to baselines as function of error
Conclusion: Our algorithm (using predictor) outperforms benchmarks such as $k$-means ++ for low error while staying competitive with high corruptions.
Summary

- NP-hard to even approximate within a factor of 1.07 [Cohen-AddadC.S.20, LeeSchmidtWright17]
- Main result [EFSWZ22]: Algorithm that outputs a $\left(1 + O(\alpha)\right)$ approximate $k$-means clustering in nearly linear time
- Handles clustering with outliers
- Not enough to blindly follow predictions!
- Our approach: Use ideas from robust mean estimation
...and Beyond!

- Related work:
  - Semi-supervised active clustering (SSAC) framework: Same cluster queries, [AKB16], [KG17], [MS17], [GHS18], [ABJK18], ..., correlation clustering

- Future directions:
  - Spectral clustering? (Talk to me!!)
  - Other predictors (multiple labels per point), relationship with robust statistics, minimizing the number of queries
  - Algorithms for \((k, z)\)-clustering, i.e., \(\sum_{p \in P} \min_{c \in C} \|p - c\|_2^z\)
  - Algorithms for \(L_p\)-metrics, i.e., \(\sum_{p \in P} \min_{c \in C} \|p - c\|_p^p\)