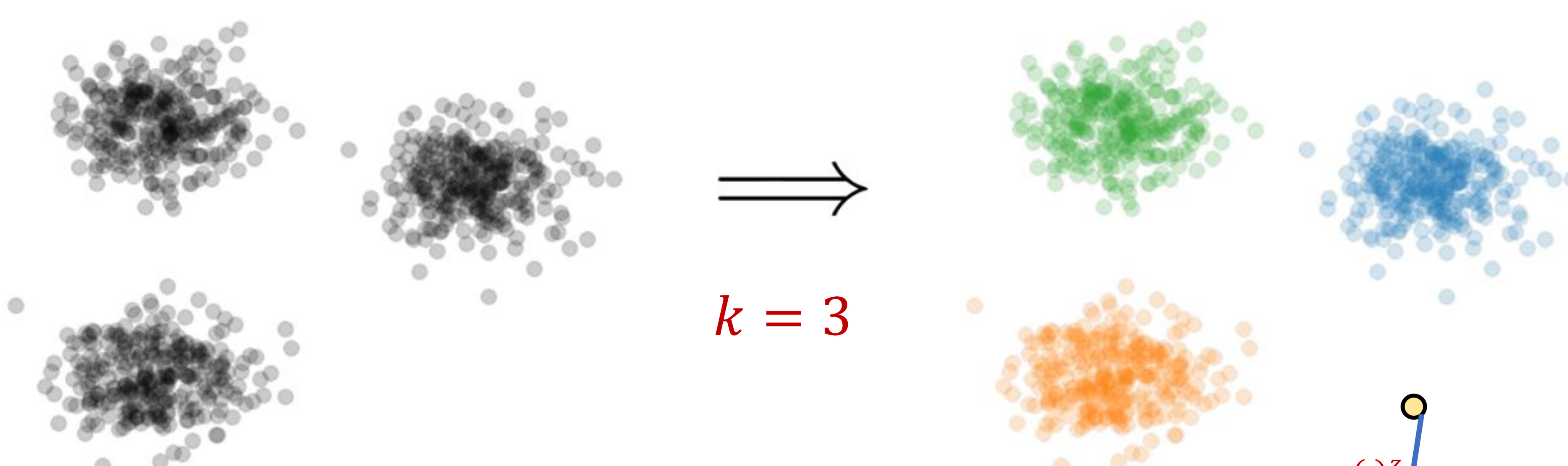
Near-Optimal k-Clustering in the Sliding Window Model

David P. Woodruff (Carnegie Mellon University)
Peilin Zhong (Google Research)
Samson Zhou (Texas A&M University)

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

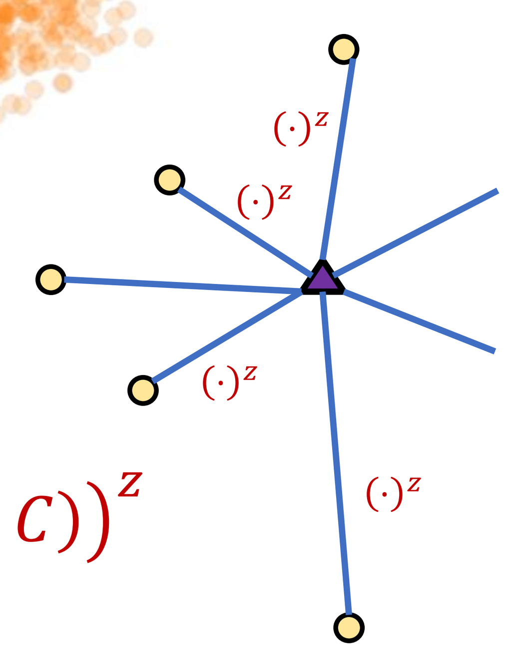
There can be at most k different clusters



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

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

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



Goal: Find a set C of k centers that achieves a $(1+\varepsilon)$-approximation to

$$\min_{C:|C| \le k} \text{Cost}(X, C) = \min_{C:|C| \le k} \sum_{x \in X} \left(\text{dist}(x, C)\right)^z$$

Sliding Window Model

Input: Elements of an underlying data set S, which arrives sequentially

Output: Evaluation (or approximation) of a given function

Goal: Use space *sublinear* in the size m of the input S

Sliding Window: “Only the m most recent updates form the underlying data set S”

1 0 1 1 1 0 0 1 1 0

Related Literature

Reference	Accuracy	Space	Setting
[BDMO03]	$2^{O(1/\varepsilon)}$	$O\left(\frac{k}{\varepsilon^4} W^{2\varepsilon} \log^2 W\right)$	k-median, $\varepsilon \in \left(0, \frac{1}{2}\right)$
[BLLM16]	$C > 2$	$O\left(k^3 \log^6 W\right)$	k-median and k-means
[ELVZ17]	$C > 2^{14}$	$k \text{ polylog}(W, \Delta)$	(k, z)-clustering
[EMMZ22]	$(1+\varepsilon)$	$\frac{(kd + d^{Cz})}{\varepsilon^3} \text{ polylog}\left(W, \Delta, \frac{1}{\varepsilon}\right)$, $C \ge 7$	(k, z)-clustering
Our work	$(1+\varepsilon)$	$\frac{k}{\min(\varepsilon^4, \varepsilon^{2+z})} \text{ polylog} \frac{n\Delta}{\varepsilon}$	(k, z)-clustering

Table 1: Summary of (k, z)-clustering results in the sliding window model for input points in $[\Delta]^d$ on a window of size W

Our Results

Theorem: There exists an algorithm that samples $\frac{k}{\min(\varepsilon^4, \varepsilon^{2+z})} \text{polylog} \frac{n\Delta}{\varepsilon}$ points and with high probability, outputs a $(1+\varepsilon)$-approximation to (k, z)-clustering for the Euclidean distance on $[\Delta]^d$ in the sliding window model

Theorem: There exists an algorithm that samples $\frac{k}{\min(\varepsilon^4, \varepsilon^{2+z})} \text{polylog} \frac{n\Delta}{\varepsilon}$ points and with high probability, outputs a $(1+\varepsilon)$-coreset to (k, z)-clustering on $[\Delta]^d$ in the sliding window model

Theorem: There exists an algorithm that samples $\frac{k}{\min(\varepsilon^4, \varepsilon^{2+z})} \text{polylog} \frac{n\Delta}{\varepsilon}$ points and with high probability, outputs a $(1+\varepsilon)$-online coreset to (k, z)-clustering on $[\Delta]^d$

Theorem: Let $\varepsilon \in (0,1)$. For sufficiently large n, d, and Δ, there exists a $X \subset [\Delta]^d$ of n points such that any $(1+\varepsilon)$-online coreset for k-means clustering on X requires $\Omega\left(\frac{k}{\varepsilon^2} \log n\right)$ points

Note: Last theorem provides a separation from the offline setting, i.e., [CLSS22]

References

[Mey01] Adam Meyerson. Online facility location. FOCS 2001

[CLSS22] Vincent Cohen-Addad, Kasper Green Larsen, David Saulpic, and Chris Schwiegelshohn. Towards optimal lower bounds for k-median and k-means coresets. STOC 2022

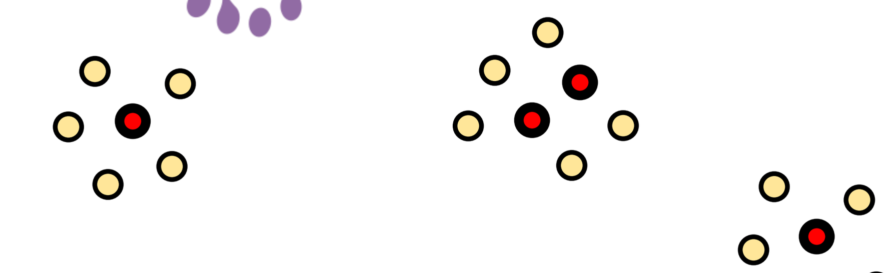
[BDMO03] Brian Babcock, Mayur Datar, Rajeev Motwani, and Liadan O’Callaghan. Maintaining variance and k-medians over data stream windows. PODS 2003

[BLLM16] Vladimir Braverman, Harry Lang, Keith Levin, and Morteza Monemizadeh. Clustering problems on sliding windows. SODA 2016

[ELVZ17] Alessandro Epasto, Silvio Lattanzi, Sergei Vassilvitskii, and Morteza Zadimoghaddam. Submodular optimization over sliding windows. WWW 2017

[EMMZ22] Alessandro Epasto, Mohammad Mahdian, Vahab S. Mirrokni, and Peilin Zhong. Improved sliding window algorithms for clustering and coverage via bucketing-based sketches. SODA 2022

Coresets



Subset X' of representative points of X for a specific clustering objective

$\text{Cost}(X, C) \approx \text{Cost}(X', C)$ for all sets C with $|C| = k$

Given a set X and an accuracy parameter $\varepsilon > 0$, we say a set X' with weight function w is an $(1+\varepsilon)$-*multiplicative coreset* for a cost function Cost, if for all queries C with $|C| \le k$, we have

$$(1-\varepsilon)\text{Cost}(X, C) \le \text{Cost}(X', C, w) \le (1+\varepsilon)\text{Cost}(X, C)$$

Intuition

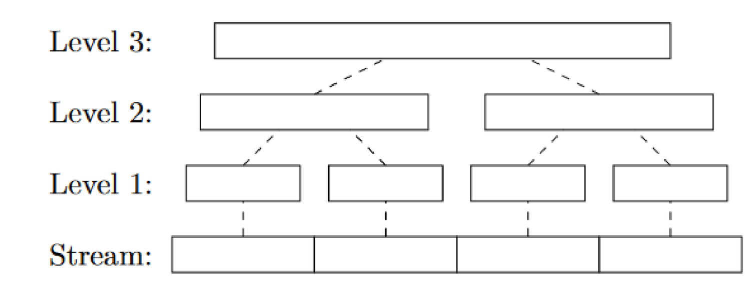
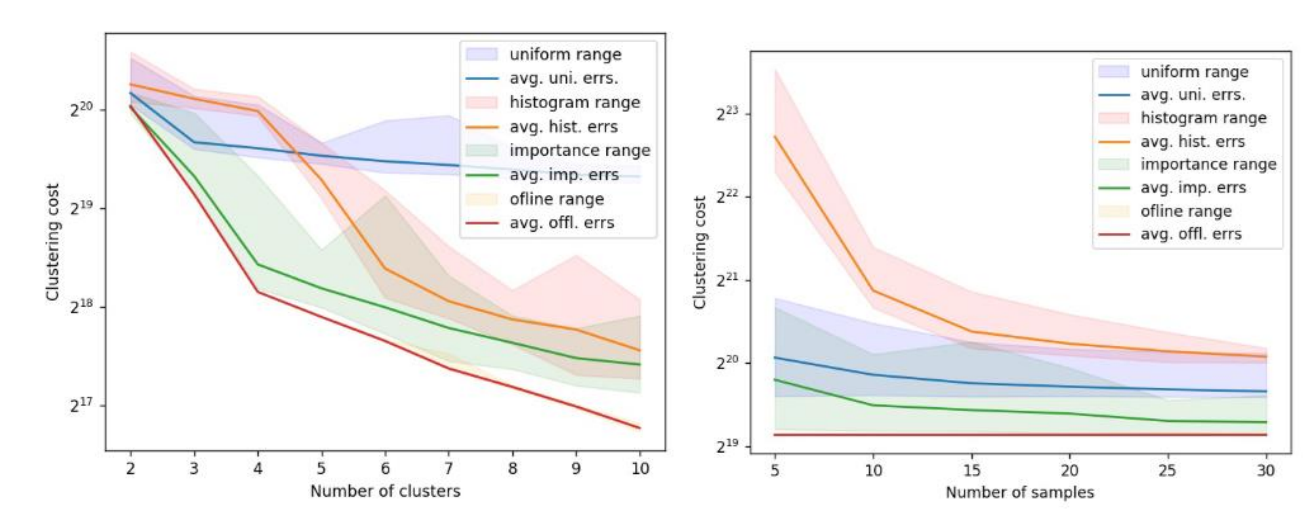


Fig. 1: Merge and reduce framework on a stream of length n. The coresets at level 1 are the entire blocks. The coresets at level i for $i > 1$ are each $\left(1 + O\left(\frac{\varepsilon}{2\log n}\right)\right)$-coresets of the coresets at their children nodes in level $i-1$.

- Online Coreset: Data structure that not only approximately preserves the cost of the data stream, but also the costs of all prefixes of the data stream
- We show there exists an online coreset using the Meyerson sketch [Mey01] and an independent sampling version of known coresets, e.g., [CLSS22]
- Run the online coreset *in reverse* at each time

Empirical Evaluations



(a) Comparisons for varying k. (b) Comparisons for varying m.

Fig. 2: Comparison of average clustering costs made by uniform sampling, histogram-based algorithm, and our coreset-based algorithm across various settings of space allocated to the algorithm, given a synthetic dataset. For comparison, we also include the offline k-means++ algorithm as a baseline, though it is inefficient because it stores the entire dataset.