ICML
International Conference
On Machine Learning

Fast k-means Seeding Under The Manifold Hypothesis

Poojan Shah

CSE @ IIT Delhi

Shashwat Agrawal

CSE @ IIT Delhi

Ragesh Jaiswal

CSE @ IIT Delhi

k-means Clustering

- Given $\mathcal{X} = \{x_1, \ldots, x_n\} \subset \mathbf{R}^D$ and integer k, find k centers C minimizing

$$\texttt{cost}(\mathcal{X}, C) = \sum_{x \in \mathcal{X}} \min_{c \in C} \|x - c\|^2$$

k-means Clustering

- Given $\mathcal{X} = \{x_1, \ldots, x_n\} \subset \mathbf{R}^D$ and integer k, find k centers C minimizing

$$\texttt{cost}(\mathcal{X}, C) = \sum_{x \in \mathcal{X}} \min_{c \in C} \|x - c\|^2$$

- NP-hard in general; practical heuristic is **Lloyd's iterations** (alternating assignment and centroid update)

k-means Clustering

- Given $\mathcal{X} = \{x_1, \ldots, x_n\} \subset \mathbf{R}^D$ and integer k, find k centers C minimizing

$$\texttt{cost}(\mathcal{X}, C) = \sum_{x \in \mathcal{X}} \min_{c \in C} \|x - c\|^2$$

- NP-hard in general; practical heuristic is **Lloyd's iterations** (alternating assignment and centroid update)
- Lloyd's needs a good **initialization (seeding)** to avoid poor local minima

k-means++ and D^2-Sampling

- Choose $c_1 \in \mathcal{X}$ uniformly. Given centers C_t, sample next center via the D^2 *distribution*:

$$\Pr[c_{t+1} = x \mid C_t] = \frac{\texttt{cost}(x, C_t)}{\texttt{cost}(\mathcal{X}, C_t)}$$

k-means++ and D^2-Sampling

- Choose $c_1 \in \mathcal{X}$ uniformly. Given centers C_t, sample next center via the D^2 *distribution*:

$$\Pr[c_{t+1} = x \mid C_t] = \frac{\texttt{cost}(x, C_t)}{\texttt{cost}(\mathcal{X}, C_t)}$$

- **Guarantee:** $O(\log k)$ expected approximation (Arthur & Vassilvitskii, 2007)

k-means++ and D^2-Sampling

- Choose $c_1 \in \mathcal{X}$ uniformly. Given centers C_t, sample next center via the D^2 *distribution*:

$$\Pr[c_{t+1} = x \mid C_t] = \frac{\texttt{cost}(x, C_t)}{\texttt{cost}(\mathcal{X}, C_t)}$$

- **Guarantee:** $O(\log k)$ expected approximation (Arthur & Vassilvitskii, 2007)
- **Runtime:** $O(nkD)$ — each round scans all n points in $\mathbf{R}^D$
- Too slow for large n, k, D — and speeding it up while retaining guarantees is an open challenge

Beyond Worst-Case Analysis

Worst-case algorithms are designed to handle *any* input - including pathological ones.

Beyond Worst-Case Analysis

Worst-case algorithms are designed to handle *any* input - including pathological ones.

- Can we model the *typical* inputs one actually encounters?

Beyond Worst-Case Analysis

Worst-case algorithms are designed to handle *any* input - including pathological ones.

- Can we model the *typical* inputs one actually encounters?
- **Prior approaches** for k-means assume properties of the *optimal solution* - α-separation, approximation stability, perturbation resilience - which are difficult to verify on real datasets

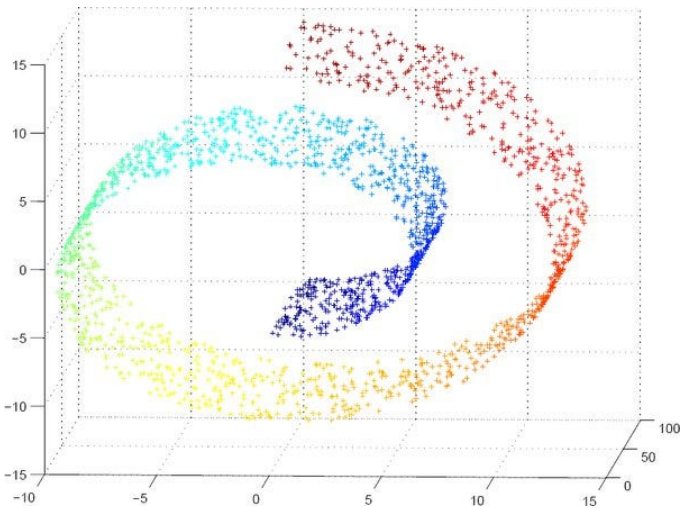
Beyond Worst-Case Analysis

Worst-case algorithms are designed to handle *any* input - including pathological ones.

- Can we model the *typical* inputs one actually encounters?
- **Prior approaches** for k-means assume properties of the *optimal solution* - α-separation, approximation stability, perturbation resilience - which are difficult to verify on real datasets
- **This work:** assume structure on the *input distribution* instead, via the manifold hypothesis

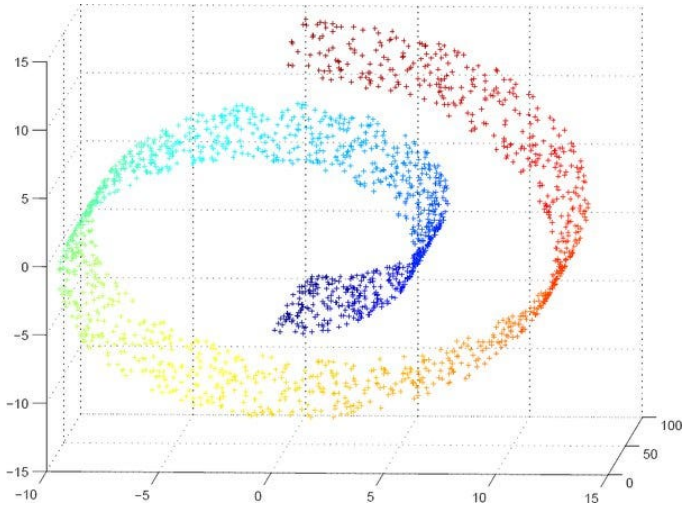
Manifold Hypothesis

Assumption. Data in $\mathbf{R}^D$ concentrates near a smooth submanifold $\mathcal{M}$ of *intrinsic dimension* $d \ll D$.



Manifold Hypothesis

Assumption. Data in $\mathbf{R}^D$ concentrates near a smooth submanifold $\mathcal{M}$ of *intrinsic dimension* $d \ll D$.



- Well-established assumption underlying modern deep learning (Bengio et al., 2013; Fefferman et al., 2016)
- Assumption on the *data*, not on the optimal solution $\Longrightarrow$ easier to validate empirically

Scaling Laws

Define two data-dependent parameters for the dataset $\mathcal{X}$:

$$\beta_k(\mathcal{X}) = \frac{\mathtt{opt}_1(\mathcal{X})}{\mathtt{opt}_k(\mathcal{X})}, \qquad \eta(\mathcal{X}) = \frac{\max_{i \neq j} \|x_i - x_j\|}{\min_{i \neq j} \|x_i - x_j\|}$$

Scaling Laws

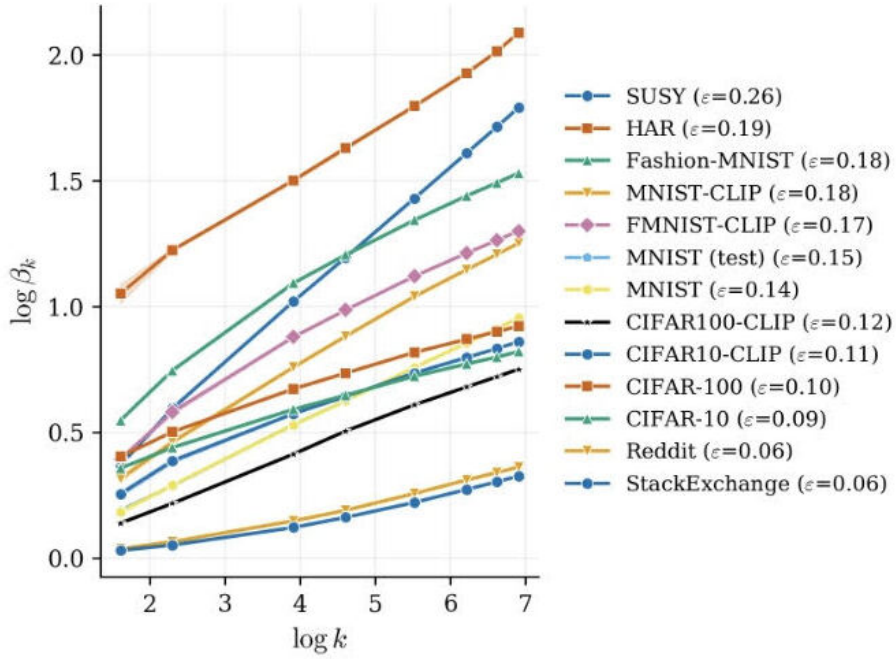
Define two data-dependent parameters for the dataset $\mathcal{X}$:

$$\beta_k(\mathcal{X}) = \frac{\mathtt{opt}_1(\mathcal{X})}{\mathtt{opt}_k(\mathcal{X})}, \qquad \eta(\mathcal{X}) = \frac{\max_{i \neq j} \|x_i - x_j\|}{\min_{i \neq j} \|x_i - x_j\|}$$

Theorem (Scaling Laws, informal)

If $\mathcal{X}$ is sampled from a density supported on a d-dimensional manifold, then with high probability:

$$\beta_k(\mathcal{X}) \approx k^{2/d}, \ \eta(\mathcal{X}) \approx n^{3/d}$$



Rejection Sampling for D^2-Sampling

Goal: simulate the D^2 distribution $\pi(x \mid C) \propto \texttt{cost}(x, C)$ cheaply.

Rejection Sampling for D^2-Sampling

Goal: simulate the D^2 distribution $\pi(x \mid C) \propto \texttt{cost}(x, C)$ cheaply.

- Use the **proposal distribution** $\kappa(x \mid C) \propto \|x\|^2 + \|c_1\|^2$, sampleable in $O(\log n)$ via a binary tree.

Rejection Sampling for D^2-Sampling

Goal: simulate the D^2 distribution $\pi(x \mid C) \propto \texttt{cost}(x, C)$ cheaply.

- Use the **proposal distribution** $\kappa(x \mid C) \propto \|x\|^2 + \|c_1\|^2$, sampleable in $O(\log n)$ via a binary tree.
- Accept proposal x with probability $\propto \texttt{cost}(x, C) \,/\, \kappa(x)$; the max Rényi divergence $D_\infty(\pi \,\|\, \kappa) \leq \log(4\beta_k)$ controls the expected number of proposals needed.

Rejection Sampling for D^2-Sampling

Goal: simulate the D^2 distribution $\pi(x \mid C) \propto \texttt{cost}(x, C)$ cheaply.

- Use the **proposal distribution** $\kappa(x \mid C) \propto \|x\|^2 + \|c_1\|^2$, sampleable in $O(\log n)$ via a binary tree.
- Accept proposal x with probability $\propto \texttt{cost}(x, C) \,/\, \kappa(x)$; the max Rényi divergence $D_\infty(\pi \,\|\, \kappa) \leq \log(4\beta_k)$ controls the expected number of proposals needed.
- Use an **LSH-based ANNS** data structure to evaluate $\texttt{cost}(x, C)$ approximately, avoiding the $O(kD)$ per-point scan. Querying and Insertion times depend on $\log \eta(\mathcal{X})$.

Rejection Sampling for D^2-Sampling

Goal: simulate the D^2 distribution $\pi(x \mid C) \propto \texttt{cost}(x, C)$ cheaply.

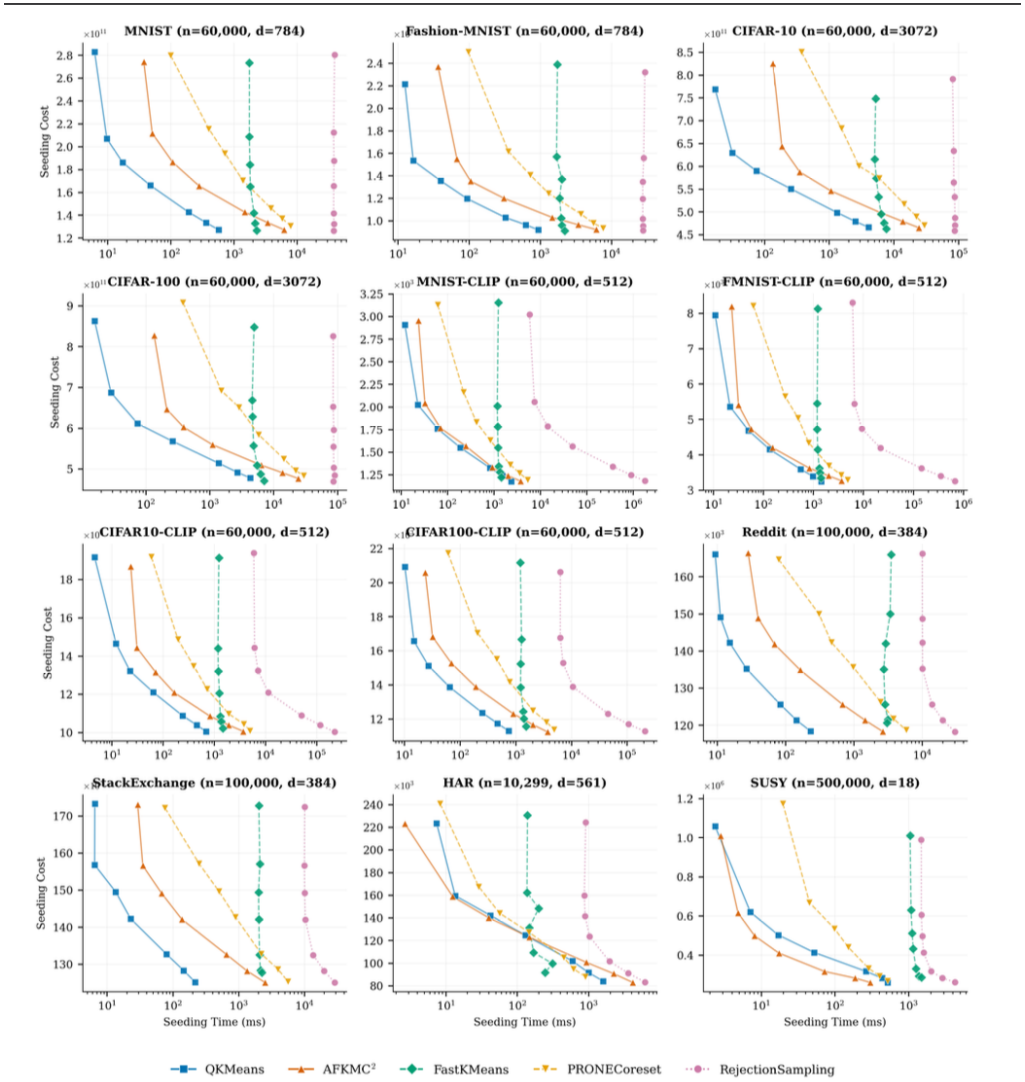
- Use the **proposal distribution** $\kappa(x \mid C) \propto \|x\|^2 + \|c_1\|^2$, sampleable in $O(\log n)$ via a binary tree.
- Accept proposal x with probability $\propto \texttt{cost}(x, C) \,/\, \kappa(x)$; the max Rényi divergence $D_\infty(\pi \,\|\, \kappa) \leq \log(4\beta_k)$ controls the expected number of proposals needed.
- Use an **LSH-based ANNS** data structure to evaluate $\texttt{cost}(x, C)$ approximately, avoiding the $O(kD)$ per-point scan. Querying and Insertion times depend on $\log \eta(\mathcal{X})$.

Main guarantee: Under the manifold hypothesis,

$$\mathbb{E}[\texttt{cost}(\mathcal{X}, C)] \;\in\; O\big(\varrho^{-2} \log k\big) \cdot \texttt{opt}_k(\mathcal{X})$$

in total time $O(nD \log k) + \tilde{O}\big(\varrho^{-1} k^{1+\varrho+2/d}\big)$.

Empirical Performance



Our algorithm is consistently among the fastest seeders while maintaining comparable solution quality

Takeaways

- The **manifold hypothesis** is an empirically verifiable alternative to hard-to-check stability conditions
- It implies **predictable scaling laws** for β_k and η.
- These laws enable a sub-quadratic k-means seeding with an $O(\varrho^{-2} \log k)$ approximation guarantee
- A promising direction for **beyond-worst-case** algorithm design in clustering

ICML
International Conference
On Machine Learning

Thank You

Fast k-means Seeding Under The Manifold Hypothesis

Code: `https://github.com/PoojanCShah/Fast-k-means-seeding-under-the-manifold-hypothesis-ICML2026`

Questions welcome!