Heterogeneity for the Win:
Communication-Efficient Federated Clustering

Don Dennis, Virginia Smith

Carnegie Mellon University

NEURAL INFORMATION PROCESSING SYSTEMS

Abstract

- We explore the unique challenges and opportunities of clustering on federated networks.
- We develop k-FED, based on the classical Lloyd's method and show that,

  - Heterogeneity can be useful: Analyze k-FED under a center separation assumption where the number of clusters per device k' is smaller than the total clusters over the network, k, we can use heterogeneity to our advantage—significantly weakening the cluster separation requirements.

  - Practical benefits: Compute-lite, communication-efficient, asynchronous and can naturally handle node/network failures.

Background: Clustering and Center Separation

k-means Clustering.

- **Given:** data matrix $A \in R^{n \times d}$ and $k > 0$ an integer. (Each row A_i a d-dimensional data point).
- **Objective:** Partition data into $T_1, T_2, \dots, T_k$ to minimize:

$$\phi(T_1, \dots, T_k) = \sum_{r=1}^{k1} \sum_{i \in T_r} \left|\left|A_i - \mu(T_r)\right|\right|^2$$

Here $\mu(T_r) = \frac{1}{|T_r|} \sum_{i \in T_r} A_i$.

Separation Based Cluster (Mixture of Gaussians)

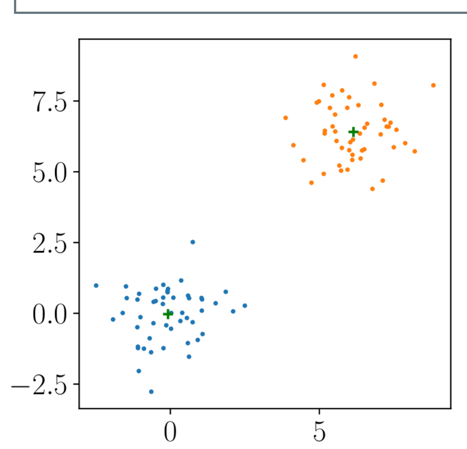
- **Given:** Data from mixture of two gaussians $N(\mu_1, \sigma_1), N(\mu_2, \sigma_2)$.
- **Objective:** Separate into T_1, T_2 based on which mixture data is from.

Rand. variables X, Y from same cluster, μ_1

$$E\left(||X-Y||^2\right) = E\left(||X - \mu_1 + \mu_1 - Y||^2\right) = 2d\sigma_1^2 \leq 2d(\sigma_1^2 + \sigma_2^2).$$

Random variables X, Y from different cluster,

$$E\left(||X-Y||^2\right) = E\left(||X - \mu_1 + \mu_1 - \mu_2 + \mu_2 - Y||^2\right) = 2d(\sigma_1^2 + \sigma_2^2) + ||\mu_1 - \mu_2||^2.$$



Separation assumption:

$$||\mu_1 - \mu_2||^2 \geq cd(\sigma_1^2 + \sigma_2^2).$$

Algorithm: Sort pairwise distance and match points.

Background: Spectral Clustering, Lloyds Algorithm

- [Awasthi-Sheffet] Analysis of Lloyd's algorithm in a deterministic setting with a center separation assumption.
- **Given:** Data matrix $A \in R^{n \times d}$ and $k > 0$ an integer.
  Objective: Recover target partitions $T_1, T_2, \dots, T_k$
  Algorithm: A slight variant of Lloyd's algorithm.
  Assumption:

$$||\mu_r - \mu_s|| \geq c\sqrt{k}\left(\frac{||A-C||}{\sqrt{n_r}} + \frac{||A-C||}{\sqrt{n_s}}\right)$$

  Here C is a matrix with row $C_i = \mu(T_r)$ for r such that $i \in T_r$.
- Under this assumption, all but $O\left(\frac{1}{c^2}\right)$ points correctly classified.
- The quantity $\frac{||A-C||}{\sqrt{n_r}}$ analogues to σ_r in Gaussian case. Infact the framework subsumes the mixture of gaussian case among others.
- Now require $O(\sqrt{k}\sigma)$ separation as opposed to $O(\sqrt{d}\sigma)$.

Algorithm 1: Local k^z-means

Input: The matrix of data points A^z and an integer k^z.

Project A^z onto the subspace spanned by the top k^z singular vectors to get $\hat{A}^z$. Run any standard 10-approximation algorithm on the projected data and estimate k^z centers $(\nu_1, \nu_2, \dots, \nu_{k^z})$.

Set $S_r \leftarrow \{i : \|\hat{A}_i^z - \nu_r\|_2 \leqslant \frac{1}{3}\|\hat{A}_i^z - \nu_s\|_2, \text{ for every } s\}$ and $\theta_r \leftarrow \mu(S_r)$.

Run Lloyd steps until convergence

$$U_r \leftarrow \{i : \|A_i^z - \theta_r\|_2 \leqslant \|A_i^z - \theta_s\|_2, \text{ for every } s\}$$
$$\theta_r \leftarrow \mu(U_r).$$

Result: Cluster assignments $(U_1, U_2, \dots, U_{k^z})$ and their means $\Theta' = (\theta_1, \dots, \theta_{k'})$.

Algorithm 1: The clustering algorithm presented by Awasthi and Sheffet. k-FED uses this as a subroutine.

Algorithm 2: k-FED

On each device $z \in [Z]$, run Algorithm-1 with local data A^z and k' and obtain local cluster centers $\Theta^z = (\theta_1^z, \dots, \theta_{k'}^z)$

For each θ_j^z, $(z, j) \in [k'] \times [Z]$, MAKE-SET($\theta_j^z$).

for *each pair of centers* (θ_j^z, θ_t^z) *ordered in increasing distance* $\|\theta_j^z - \theta_t^z\|_2$ **do**

    // UNION() and FIND() methods from the union-find data structure.

    if *FIND*(θ_j^z) $\neq$ *FIND*(θ_t^z) **then**

        UNION(FIND(θ_j^z), FIND(θ_t^z))

    end

    if *Number of sets* $\leqslant k - 1$ **then**

        Discard last union and return the k-sets.

    end

end

Result: Local cluster centers partitioned into k sets.

Algorithm 2: The central aggregation/cleanup part of k-FED.

k −FED: Clustering over Federated Network

- **Given**: Data generated on devices in a network.
  Objective: Partition data into k target clusters.
  Algorithm:
  Stage 1 – each device runs Local k-means and sends partial clustering to central server (Algo 1).
  Stage 2 – sever aggregates and generates final clustering. (Algo 2)
  Assumptions:
  - Each device has data from $k' \ll k$ clusters.
  - Active separation:

$$||\mu_r - \mu_s|| \geq ck'\left(\frac{||A-C||}{\sqrt{n_r}} + \frac{||A-C||}{\sqrt{n_s}}\right)$$

  - Inactive separation:

$$||\mu_r - \mu_s|| \geq c\sqrt{k'}\left(\frac{||A-C||}{\sqrt{n_r}} + \frac{||A-C||}{\sqrt{n_s}}\right)$$

- Better to use k-FED when $k' = O(\sqrt{k})$.
- **Communication Eff:** Only one round of communication required.
- **Compute-lite:** The client-side algorithm is a variant of Lloyd's and only executes once on each device.
- **Asynchronous:** No need to synchronous across devices. Each device sends its clustering estimate at its pace.
- **Device failures/stragglers:** Newly awake devices can participate with only server-side computation.

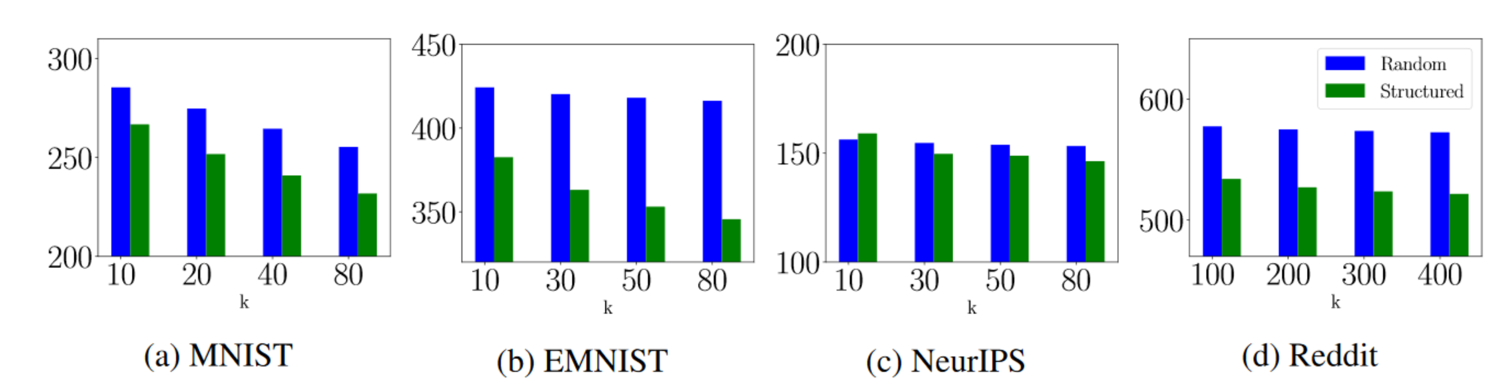


Figure 1: Heterogeneity helps: We compare clustering on IID vs. non-IID partitions and find that clustering with heterogeneous data leads to lower k-means cost ratio. Non-IID partition based on labelling information or other heuristics. Refer to manuscript for more information.

References

Kannan, R. and Vempala, S. Spectral algorithms (2010).

Awasthi, P. and Sheffet, O. Improved Spectral-Norm Bounds for Clustering

Kannan, R. and Kumar, A. Clustering with Spectral Norm and the k-means Algorithm.