Differentially Private Clustering

Ivan Gutierrez, Connor Buchheit, Mohamed Al-Duraji

Background — Clustering Algorithms

Clustering is a fundamental unsupervised learning task that partitions data into coherent groups. A common algorithm is **Lloyd's K-Means**, which alternates between assigning points to the nearest center and updating centers.
In sensitive applications (e.g., health, finance), clustering must be done without leaking individual-level data. This motivates the use of **Differential Privacy (DP)**—a rigorous framework for protecting individual data.

K-Means Algorithm

Lloyd's K-Means Algorithm:

Require: Dataset $\mathbf{X} = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$, number of clusters K
Ensure: Cluster centers $\{\mu_1, \ldots, \mu_K\}$ and assignments $\{z_1, \ldots, z_n\}$

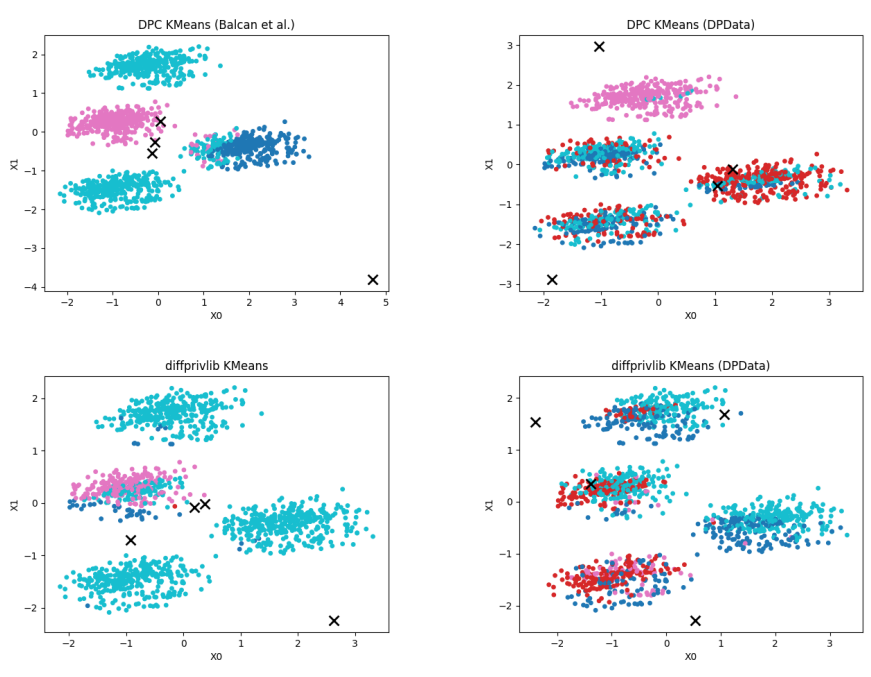
1: Initialize cluster centers $\mu_1, \ldots, \mu_K$ (e.g., randomly choose K points from $\mathbf{X}$)
2: **repeat**
3: **for** each point $x_i \in \mathbf{X}$ **do**
4: Assign x_i to the nearest cluster:
 $z_i \leftarrow \arg\min_{k \in \{1,\ldots,K\}} \|x_i - \mu_k\|^2$
5: **end for**
6: **for** each cluster $k = 1$ to K **do**
7: Update center μ_k as mean of assigned points:
 $\mu_k \leftarrow \frac{1}{|C_k|} \sum_{i:z_i=k} x_i$
8: **end for**
9: **until** convergence (e.g., no change in assignments or centers)

Main Contributions

We evaluate four differentially private k-means methods:

***DPC KMeans (DPData):** Adds noise to the input data, then clusters using our own implementation of Lloyd's algorithm.
***DPC KMeans (Balcan et al.):** Projects data to lower dimensions, constructs a private grid, selects candidate centers using the exponential mechanism, and refines them with noisy averaging.
diffprivlib KMeans: Uses IBM's diffprivlib module to perform DP Lloyd's k-means with built-in privacy guarantees.
diffprivlib KMeans (DPData): Splits the privacy budget between input noise and diffprivlib's internal DP routine.
* indicates methods we implemented ourselves.

Random Sample of Clustering Results Across Methods at $\epsilon = 1$



References

[1] Maria-Florina Balcan et al. "Differentially Private Clustering in High-Dimensional Euclidean Spaces". In: *Proceedings of the 34th International Conference on Machine Learning*. Ed. by Doina Precup and Yee Whye Teh. Vol. 70. Proceedings of Machine Learning Research. PMLR, 2017, pp. 322–331. URL: https://proceedings.mlr.press/v70/balcan17a.html.

[2] Dong Su et al. "Differentially Private K-Means Clustering and a Hybrid Approach to Private Optimization". In: *ACM Trans. Priv. Secur.* 20.4 (Oct. 2017). ISSN: 2471-2566. DOI: 10.1145/3133201. URL: https://doi.org/10.1145/3133201.

Evaluating Algorithms — ARI and NMI

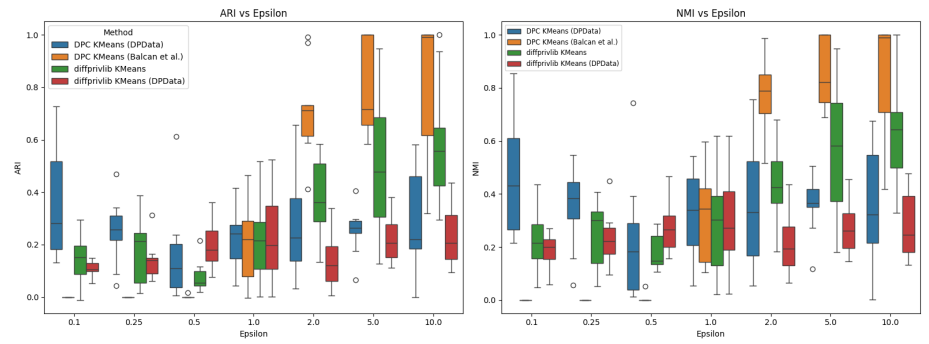
We use three different metrics to evaluate the different clustering algorithms we use, comparing private and non-private algorithms.

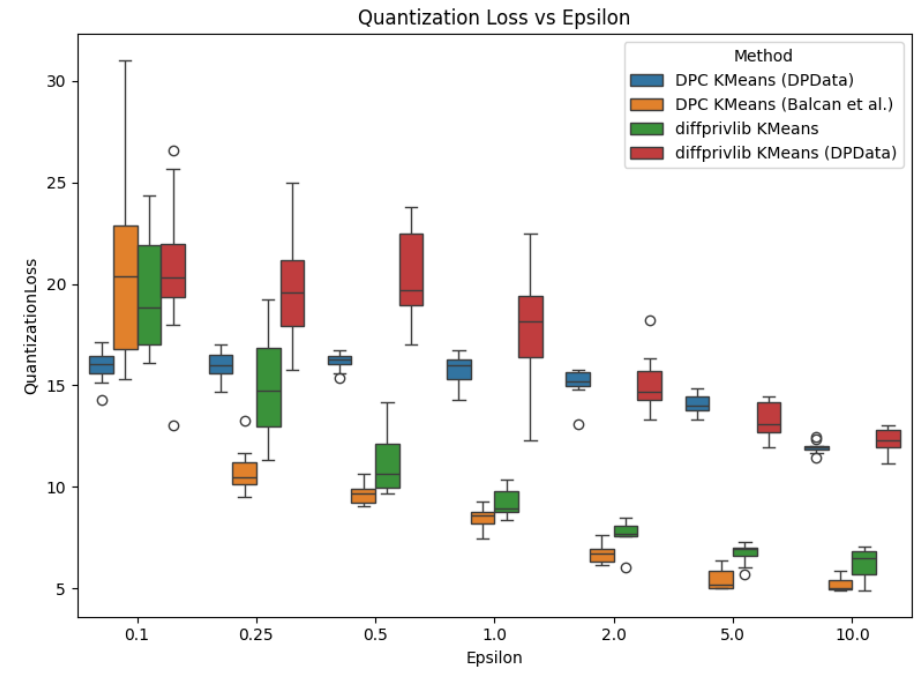
Metric 1: Adjusted Rand Index (ARI) ARI checks how often pairs of points that belong together in the true clusters also end up together in the predicted clusters. A score of 1 means perfect agreement; 0 means random guessing.

Metric 2: Normalized Mutual Information (NMI) NMI measures how much information is shared between the true and predicted clusters.

Metric 3: Quantization Loss This measures how well the chosen cluster centers summarize the data. A lower loss means the clusters are tightly packed and accurately placed.

We then ran said algorithms on a dataset of the attributes of customers and how much they pay for insurance with $N \approx 1300$ entries. We assumed a normal distribution for the data, so we normalized it such that it had a mean of 0 and a standard deviation of 1, and then clipped the data to be between $[-3, 3]^d$ where d represents the number of features any given row has. The results from over 10 trials per run are displayed below.





Results and Conclusions

Results Summary:

- **DPC KMeans (Balcan et al.)** and **diffprivlib** generally perform better than naive DP implementations in terms of ARI and NMI.
- Noise injection methods (e.g., DPData) show significantly higher quantization loss and slower decay as $\epsilon \to \infty$.
- Performance can vary based on the allocation of budget. For any algorithm that implemented differential privacy into KMeans and also used differentially private data, we split the budget in half for both.

Conclusions:

- Dimensionality reduction and careful use of the exponential mechanism yield more utility-preserving clustering under DP constraints.
- However, libraries like `diffprivlib` are more readily available for use and are easier to understand than state-of-the-art algorithms like Balcan et al.