Differentially Private k-Means via Exponential Mechanism and Max Cover

Huy L. Nguyễn*, Anamay Chaturvedi*, Eric Z Xu*
Khoury College of Computer Sciences, Northeastern University
440 Huntington Ave
Boston, Massachusetts 02115
{hu.nguyen, chaturvedi.a, xu.er}@northeastern.edu

Abstract

We introduce a new (ϵ_p, δ_p)-differentially private algorithm for the k-means clustering problem. Given a dataset in Euclidean space, the k-means clustering problem requires one to find k points in that space such that the sum of squares of Euclidean distances between each data point and its closest respective point among the k returned is minimised. Although there exist privacy-preserving methods with good theoretical guarantees to solve this problem, in practice it is seen that it is the additive error which dictates the practical performance of these methods. By reducing the problem to a sequence of instances of maximum coverage on a grid, we are able to derive a new method that achieves lower additive error than previous works. For input datasets with cardinality n and diameter Δ, our algorithm has an $O(\Delta^2(k\log^2 n\log(1/\delta_p)/\epsilon_p + k\sqrt{d\log(1/\delta_p)}/\epsilon_p))$ additive error whilst maintaining constant multiplicative error. We conclude with some experiments and find an improvement over previously implemented work for this problem.

Introduction

Clustering is a well-studied problem in theoretical computer science. A relatively general variant of this problem is when given a dataset D of size n to find k centers that minimize the sum of distances of each point to its closest center. When the ambient space is Euclidean and the distance is the square of the Euclidean metric this is known as the k-means problem.

When algorithms handle sensitive information, an important requirement that they might be expected to fulfill is that of being *differentially private* (Dwork et al. 2016). Differential privacy provides a framework for capturing the loss in privacy that occurs when sensitive data is processed. In this work we are interested in the centralized model of differential privacy, where the algorithm whose privacy loss we want to bound is executed by a trusted curator with access to many agents' private information and who must reveal their answer publicly.

In the theoretical study of the k-means problem, reducing the worst-case multiplicative approximation factor has been the focus of a major line of work (Kanungo et al. 2004; Ahmadian et al. 2020). However, even Lloyd's algorithm (Lloyd 1982), which has a tight sub-optimal multiplicative guarantee of $O(\log k)$, works well in practice. This behaviour can be understood by showing (Aggarwal, Deshpande, and Kannan 2009) that Lloyd's finds a solution with $O(1)$ multiplicative error with constant probability, or that for a general class of datasets satisfying a certain separability condition (Ostrovsky et al. 2012) the multiplicative error again has a strong $O(1)$ bound.

In contrast, when the algorithm is required to be (ϵ_p, δ_p)-differentially private, no pure multiplicative approximation is attainable and additive error is necessary. This principle is formalised for the closely related discrete k-medians[1] problem in theorem 4.4 of Gupta et al. (2010) which shows that there is a family of instances whose optimal clustering cost is 0 but any differentially private algorithm must incur an $\Omega(\Delta^2 k(\log n/k)/\epsilon_p)$ expected cost. In practice, for many datasets it is seen that although the non-private clustering cost naturally decreases as the number of centers k increases, the costs incurred by differentially private algorithms quickly plateau (as in the experiments of Balcan et al. (2017)), suggesting that they have reached their limit in the additive error. Given this fundamental barrier, a major question is:

Question: Is it possible to obtain a finite approximation with additive error nearly linear in k?

Contributions

We introduce a differentially private k-means clustering algorithm for the global model of differential privacy. The additive error is nearly linear in k in contrast to a polynomial overhead in previous works, and the multiplicative error is constant, which is competitive with all previous works. The algorithm also exhibits an improvement experimentally over earlier work on synthetic and real-world datasets (Balcan et al. 2017). For a specific setting of parameters with constants applicable for experiments, we have the following bound. More general bounds can be found in subsequent sections.

Theorem 1. *There is an (ϵ_p, δ_p) differentially private algorithm for the k-means problem that achieves a utility bound of $O(1)f_D(OPT_D) + O\left(\frac{k\Delta^2\log^2 n\log 1/\delta_p}{\epsilon_p}\right) +$*

*Equal contribution, names in no particular order.

[1]The discrete k-medians problem is formulated similarly except the distance function is a metric, and the centers come from a public finite set and not the whole ambient space.

Reference	
Mult. Approx.	**Add. Approx.**
Balcan et al. (2017)	
$O\left(\log^3 n\right)$	$\tilde{O}\left(k^2+d\right)$
Stemmer and Kaplan (2018)	
$O(1/\gamma)$	$\tilde{O}\left(k^{1.5}+d^{0.5+\gamma}k^{1+\gamma}\right)$
Jones, Nguyen, and Nguyen (2020)	
$O(1/\gamma)$	$\tilde{O}\left(k+d^{0.5+\gamma}k^{1+\gamma}\right)$
This work	
$O(1)$	$\tilde{O}(k\sqrt{d})$

Table 1: Comparison with prior works where we omit all $\log$ terms and the common Δ^2 factor in the additive error, the dependence on privacy parameters and set $\delta_p = 1/n^{1.5}$.

$O\left(\frac{\Delta^2 k\sqrt{d\log 1/\delta_p}}{\epsilon_p}\right)$, *where* D *is the input dataset,* $f_D(OPT_D)$ *is the optimal* k*-means cost for the input dataset* D*,* d *is the ambient dimension of* D*,* n *is the cardinality of* D*,* Δ *is the diameter of* D*, and the failure probability of the algorithm is polynomially small in* n*.*

We extend the construction of Gupta et al. (2010) for the discrete k-medians problem to our setting and show that a linear dependence on k in the additive error is necessary for any finite multiplicative approximation.

Theorem 2 (Informal)**.** *Any* (ϵ_p, δ_p)*-differentially private algorithm must incur an expected additive error of* $\Omega\left(\frac{\Delta^2 k\ln(\epsilon_p/\delta_p)}{\epsilon_p}\right)$.

The same construction also implies a lower bound for $(\epsilon_p, 0)$-differential privacy.

Theorem 3 (Informal)**.** *Any* $(\epsilon_p, 0)$*-differentially private algorithm must incur an expected additive cost of* $\Omega\left(\frac{\Delta^2 kd}{\epsilon_p}\right)$.

All full proofs may be found in the supplementary material. We finish with an experimental evaluation of our algorithm, where we find better performance than an implementation of previous work (Balcan et al. 2017).

Related Work and Techniques

In Gupta et al. (2010), the authors gave an algorithm for solving the discrete k-medians problem and subsequent works focused on identifying good discretizations of the continuous domain to invoke their algorithm. A recent approach by Stemmer and Kaplan (2018) uses locality sensitive hashing (LSH) to identify a small set of points that serve as potential centers. Inherent in this approach is a trade-off between the multiplicative approximation and the size of this discrete set, which comes from the trade-off in LSH between the approximation and the number of hash functions. The number of candidate centers increases additive error and thereby causes a trade-off between the multiplicative $\tilde{O}(1/\gamma)$ and additive $\tilde{O}(k + d^{0.5+\gamma}k^{1+\gamma})$ errors. The work Jones, Nguyen, and Nguyen (2020) achieved $\tilde{O}(k^{1+\gamma}d^{0.5+\gamma})$ additive error but the multiplicative error remained $O(1/\gamma)$.

In this work, we reduce the minimum additive error in this trade-off to nearly linear in k and also eliminate the resulting blow-up in the multiplicative factor using the most natural approach: discretizing the space using a grid and using all grid points as candidate centers. We reduce the data dimensions to $O((\log n)/\epsilon^2)$ and preserve all distances. However, there can be $(n)^{\log n}$ many points in the grid that we construct since the grid size must start from $1/n$ for negligible additive error. It is not clear how to implement a selection algorithm (such as the exponential mechanism (McSherry and Talwar 2007)) on such a large number of choices. This hurdle, identified in Balcan et al. (2017), prompted subsequent works to find alternative approaches. Resolving it directly is our key contribution.

We observe that it is not inherently difficult to sample uniformly among a large number of choices. Our task is non-trivial since the k-means cost objective is a complex function. To simplify the sampling weights, we exploit the connection between clustering and coverage and reduce the problem to finding maximum coverage: count the number of data points within a given radius of each candidate center. The crucial observation is that there are at most $n^{\tilde{O}(1/\epsilon^2)}$ grid points within the threshold radius of any data point, meaning that there are only a polynomial number of grid points with non-zero coverage. Thus, all but a polynomial number of choices have the same coverage of 0 making it possible to implement the exponential mechanism in polynomial time.

Given the implementation of the exponential mechanism for coverage, we follow the approach of Jones, Nguyen, and Nguyen (2020) to cover the points using clusters of increasing radii. Note that the approach goes back to the non-private coreset construction of Chen (2009). However, the use of coverage for dealing with each radius has another crucial advantage: as in Jones, Nguyen, and Nguyen (2020), by using the technique of Gupta et al. (2010), the privacy loss only increases by a $\log 1/\delta_p$ factor even though the algorithm has $\Omega(k)$ adaptive rounds of exponential mechanism.

Preliminaries

We are given a dataset D of n points that lies in a ball $B_{\Delta/2}(0)$ (the ball of radius $\Delta/2$ centered at 0) in some high dimensional space $\mathbb{R}^d$. The goal is to find a set of k points $S = \{\mu_1, \ldots, \mu_k\}$ such that $\sum_{p\in D} d(p, S)$ is minimal. Here $d(\cdot,\cdot) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is the square of the Euclidean distance, that is $d(p, q) := \sum_{i=1}^{d}(p_i - q_i)^2$. Abusing notation, $d(p, S) := \min_{\mu\in S} d(p, \mu)$. Define

$$f_D(S) = \sum_{p\in D} d(p, S),$$

so when S is a set of size k, $f_D(S)$ is the k-means cost of the solution S for the dataset D.

Differential privacy is formalised as follows:

Definition 4. *Two datasets $D, D' \in X^n$ are called* neighbouring *if there is exactly one element in their symmetric difference. We say that an algorithm A is (ϵ, δ)-differentially private if for any two neighbouring datasets D, D' and any measurable output set U lying in the co-domain of A,*

$$P(A(D) \in U) \leq e^{\epsilon} P(A(D') \in U) + \delta.$$

Lower Bounds

Following the construction in theorem 4.4 of Gupta et al. (2010), we derive lower bounds for k-means clustering in the (ϵ, δ) and $(\epsilon, 0)$-differential privacy regimes.

Theorem 5. *For any $0 < \epsilon_p, \delta_p \leq 1$ and integer k, there is a family of k-means instances over the cube $[0, \Delta/\sqrt{d}]^d$ with $d = O(\ln(k/(\epsilon_p \delta_p)))$ dimensions such that the optimal clustering cost is 0 but any (ϵ_p, δ_p)-differentially private algorithm would incur an expected cost of $\Omega\left(\frac{\Delta^2 k \ln(\epsilon_p/\delta_p)}{\epsilon_p}\right)$.*

Proof. Let $d = \Theta(\ln(k/((e^{\epsilon_p} - 1)\delta_p))$ and W be the set of codewords of an error correcting code with constant rate and constant relative distance in $\{0,1\}^d$. The dimension d and codewords W are chosen so that $|W| \geq k/((e^{\epsilon_p} - 1)\delta_p)$. Let $L = \ln((e^{\epsilon_p} - 1)/(4\delta_p))/(2\epsilon_p)$. Our input domain is $[0,1]^d$ which has diameter $\Delta = \sqrt{d}$. Note that for other values of Δ, we can simply re-scale the construction.

Suppose M is an arbitrary (ϵ_p, δ_p)-differentially private algorithm that on input $D \subset [0,1]^d$ outputs a set of k locations. Let M' be the algorithm that first runs M on the input and then snaps each output point to the nearest point in W; by post-processing, it has the same privacy guarantee. Furthermore, observe that if the input points are located at a subset of W then the cost of M' is within a factor 4 of the cost of M. Let A be a size k subset of W chosen uniformly at random and the dataset D_A be a multiset containing each point in A with multiplicity L. Note that the optimal cost for D_A is 0.

We would like to analyze $\phi = \mathbb{E}_{A,M'}[|A \cap M'(D_A)|]/k$. We have:

$$\begin{aligned} k\phi &= \mathop{\mathbb{E}}_{A,M'}\left[\sum_{i \in A} 1_{i \in M'(D_A)}\right] \\ &= k \mathop{\mathbb{E}}_{A,M'} \mathop{\mathbb{E}}_{i \in A}[1_{i \in M'(D_A)}] \\ &= k \mathop{\mathbb{E}}_{i \in W} \mathop{\mathbb{E}}_{A,M'}[1_{i \in M'(D_A)} | i \in A] \end{aligned}$$

Let i' be an random point in W not in A. Changing A to $A' = A \setminus \{i\} \cup \{i'\}$ requires changing $2L$ elements of D_A. Notice that for random $A \setminus \{i\}$ in $W \setminus \{i\}$ and random i' in $W \setminus A$, we have that A' is still a uniformly random subset of $W \setminus \{i\}$. Thus,

$$\begin{aligned} &\mathop{\mathbb{E}}_{i \in W} \mathop{\mathbb{E}}_{A',M'}[1_{i \in M'(D_{A'})} | i \notin A'] \\ &\geq \left(\mathop{\mathbb{E}}_{i \in W} \mathop{\mathbb{E}}_{A,M'}[1_{i \in M'(D_A)} | i \in A]\right) e^{-\epsilon_p \cdot 2L} - \frac{\delta_p}{e^{\epsilon_p} - 1} \end{aligned}$$

Here we use the fact that M' is (ϵ_p, δ_p)-differentially private, and that the δ_p losses in expectation decrease geometrically with factor $\exp(-\epsilon_p)$ so the net leakage from the δ term can be lower bounded by the sum of an infinite geometric progression. Continuing,

$$\begin{aligned} \mathop{\mathbb{E}}_{i \in W} \mathop{\mathbb{E}}_{A',M'}[1_{i \in M'(D_{A'})}] &\geq \phi \exp(-\epsilon_p \cdot 2L) - \frac{\delta_p}{e^{\epsilon_p} - 1} \\ &\geq 4\phi\delta_p/(e^{\epsilon_p} - 1) - \frac{\delta_p}{e^{\epsilon_p} - 1} \end{aligned}$$

Since $M'(D_{A'})$ has at most k points, the LHS is at most $k/|W|$. Thus, $\phi \leq (k/|W| + \delta_p/(e^{\epsilon_p} - 1))/(4\delta_p/(e^{\epsilon_p} - 1)) \leq 1/2$.

For each point in $A \setminus M'(D_A)$, the algorithm incurs a cost of $\Theta(L\Delta^2)$ due to the multiplicity of L of points in D_A and the fact that all points in W are at distance $\Theta(\Delta)$ apart. The expected cost of M', and consequently the cost of M, is hence $\Omega(kL\Delta^2) = \Omega\left(\frac{\Delta^2 k \ln(\epsilon_p/\delta_p)}{\epsilon_p}\right)$. □

Using that $|W| = 2^{\Omega(d)}$, and by setting $L = \ln(|W|/(2k))/(2\epsilon_p)$, tracing the same proof one obtains a lower bound for $(\epsilon_p, 0)$-differential privacy.

Theorem 6. *For any $0 < \epsilon_p \leq 1$ and integers k and $d = \Omega(\ln(k))$, there is a family of k-means instances over the cube $[0, \Delta/\sqrt{d}]^d$ such that the optimal clustering cost is 0 but any $(\epsilon_p, 0)$-differentially private algorithm would incur an expected cost of $\Omega\left(\frac{\Delta^2 kd}{\epsilon_p}\right)$.*

Algorithm

Our algorithm can be described in four steps.

Step 1: The dataset $D \subset B(0, \Delta/2) \subset \mathbb{R}^d$ is preprocessed via dimension reduction, scaling and projection to produce a dataset $D' \subset B_1(0) \subset \mathbb{R}^{d'}$ where $d' = O((\log n)/\epsilon^2)$. We let G_i be multi-dimensional grids of side lengths t_i and observe that if μ is a center of a cluster with radius $\leq r_i$ in the optimal solution, then by the triangle inequality a ball of radius $r_i + t_i\sqrt{d'}$ centered at $\lfloor \mu \rfloor^{(i)}$ (the "floor" of μ the in grid) contains all the points of the same cluster.

Step 2: The threshold radii r_i increase geometrically by a factor of $(1 + \epsilon)$ from $1/n$ to 2; the unit length of grid G_i is $t_i = \epsilon r_i/\sqrt{d}$. From G_i we choose candidate centers of clusters with radii in the interval $[r_{i-1}, r_i)$. This is done by counting the number of datapoints within $r_i + t_i\sqrt{d'}$ of every grid point G_i. We calculate a set of valid offsets V_i and, iterating over $p \in D$, increment counts for all grid points within an offset of $\lfloor p \rfloor^{(i)}$. We use the exponential mechanism to greedily identify the set of $k \log\lceil 1/\epsilon \rceil$ best grid points C_i that attains close to optimal coverage. The candidate set C is the union of $C_1, C_2, \ldots, C_{\log_{1+\epsilon} 2/(1/n)}$.

Step 3: We want to construct a proxy dataset D'' by moving each datapoint in D' to its closest center in C. To maintain privacy, we compute the count n_c of datapoints whose closest center is c and add Laplace noise to get $\tilde{n}_c$. D'' then contains $\tilde{n}_c$ copies of c for all $c \in C$.

Algorithm 1: Private k-means

Data: $D \subset \mathbb{R}^d$ dataset, $|D| = n$.
Result: $S = \{\tilde{\mu}_1, \ldots, \tilde{\mu}_k\} \subset \mathbb{R}^d$
$T \sim \text{JohnsonLindenstrauss}(n, \epsilon)$; // Step 1
$D' \leftarrow T(D)$
$d' \leftarrow \dim(T) = O((\log n)/\epsilon^2)$
Scale D' down by a factor of $\frac{\Delta(1+\epsilon)}{2}$ and project to $B_1(0)$
Let T' be the composition of T with the scaling and projection so that $T'(D) = D'$
$r_1 \leftarrow 1/n$; // Step 2
$t_1 \leftarrow \epsilon/(n\sqrt{d'})$
for $i = 1, \ldots, m = \lceil \log_{1+\epsilon} 2n \rceil$ **do**
  $C_i \leftarrow algorithm\ 2(D', t_i, r_i)$
  $r_{i+1} \leftarrow (1+\epsilon) r_i$.
  $t_{i+1} \leftarrow (1+\epsilon) t_i$.
end
$D' \leftarrow T'(D)$; // Step 3
$C = \bigcup_{i=1}^{m} C_i$
Assign all $p \in D'$ to the closest point C, denoted $\mathsf{grid}[p]$
Let n_c be the number of points in D' assigned to c
For each $c \in C$ set $n'_c = n_c + \text{Lap}\left(\frac{1}{\epsilon_L}\right)$
Let D'' be the dataset where every $c \in C$ is repeated n'_c times
$S'' = \{\mu''_1, \ldots, \mu''_k\} \leftarrow \text{Lloyd}(D'')$; // Step 4
$D'_i \leftarrow \{p \in D' : \arg\min_{\mu'' \in S''} d(p, \mu'') = \mu''_i\}$ for $i = 1, \ldots, k$
for $i = 1, \ldots, k$ **do**
  $\tilde{\mu}_i = algorithm\ 3(D, 1_{D'_i}, \epsilon_G, \delta_G)$; // $1_{D'_i}(p)$ indicates whether $T'(p) \in D'_i$ for $p \in D$
end
return $\tilde{S} = \{\tilde{\mu}_1, \ldots, \tilde{\mu}_k\}$

Step 4: In the final step we apply any non-private k-means clustering algorithm to D'' to get some cluster centers S''. We cluster D' using these cluster centers to get clusters C', and define final clusters for D by identifying points with their images under the projection and re-scaling. To stay private we use NoisyAVG (Nissim, Stemmer, and Vadhan 2016) to derive centers by averaging over cluster sets.

The formal pseudocode requires some additional justification; the construction of the offset set V_i, and the polynomial time implementation of the exponential mechanism.

Lemma 7. *A data point p is within distance $r_i + t_i\sqrt{d'}$ of a grid point $t_i b$ for $b \in \mathbb{Z}^{d'}$ only if $\sum_{j=1}^{d'} \min((\lfloor p \rfloor_j^{(i)} - t_i b_j)^2, (\lfloor p \rfloor_j^{(i)} - t_i(b_j+1))^2) \leq (r_i + t_i\sqrt{d'})^2$. Let $V_i = \{v : v \in \mathbb{N}^{d'}, \sum_{j=1}^{d'} t_i^2 v_j^2 < (r_i + t_i\sqrt{d'})^2\}$. If $d(p, t_i b) < (r_i + t_i\sqrt{d'})^2$ then for some $s \in \{0,1\}^d$ and $v \in V_i$, $t_i b = \lfloor p \rfloor^{(i)} + t_i s + (2s - \bar{1}) t_i v$, where $\bar{1} = (1, 1, \ldots, 1)$.*

Sketch of proof. For any real number, either its floor or its ceil is closer to a given integer than it is. Applying this principle coordinate-wise in the grid, we see that a point can lie within a distance $r_i + t_i\sqrt{d'}$ of a given grid point only if the vertex of the grid unit cube closest to that grid point were also to lie within a distance of $r_i + t_i\sqrt{d'}$. The second half of the claim follows by noting that the expression $t_i s + (2s - \bar{1}) t_i v$ is exactly the closest vertex of the grid unit cube containing p to $t_i b$. □

Lemma 8. *After computing the cover of each grid point, algorithm 2 executes the exponential mechanism correctly and in polynomial time.*

Proof. We know that for any data point the only grid points whose cover must be updated lie in V_i. It suffices to show $|V_i| < n^{\tilde{O}(1/\epsilon^2)}$. The number of ordered tuples $v \in \mathbb{N}^{d'}$ for which $\sum_j t_i^2 v_j^2 < (r_i + t_i\sqrt{d'})^2 \Leftrightarrow \sum_j v_j^2 < d'(\frac{1}{\epsilon} + 1)^2$, equals the number of ways of partitioning $d'(\frac{1}{\epsilon} + 1)^2 + d' + 1$ balls into $d' + 1$ distinguishable bins. It follows that $|V| = 2^{d'} \binom{d'(\frac{1}{\epsilon}+1)^2 + d' + 1}{d'+1} < 2^{d'} \left(\frac{e d'(\frac{1}{\epsilon}+1)^2 + d' + 1}{d'+1}\right)^{d'+1} = 2^{d'} O(1/\epsilon^2)^{d'+1} = n^{O((1/\epsilon^2)\log 1/\epsilon)}$, using that $d' = O\left((\log n)/\epsilon^2\right)$.

We want that the grid point $g \in G_i$ be sampled with probability $P(g) = \frac{\exp\left(\frac{\epsilon_E |\mathsf{cover}[g]|}{2}\right)}{\sum_{h \in G_i} \exp\left(\frac{\epsilon_E |\mathsf{cover}[h]|}{2}\right)}$. Since all but polynomially many grid points $\{g : \mathsf{cover}[g] = 0\}$ are being sampled with the smallest probability any point is sampled with, we can use the law of total probability to write this sampling distribution as a uniform distribution on the entire grid with some probability $1 - P_{samp}$, and a second distribution with P' supported only on the polynomially many grid points with non-zero cover with probability P_{samp}, i.e. $P(g) = P_{samp} P'(g) + (1 - P_{samp})\frac{1}{|G_i|}$. Setting g to be any point with 0 cover for which $P'(g) = 0$, one derives the necessary expression for P_{samp}. □

Utility

To derive a bound for the utility attained by algorithm 1, we have three steps; first we show that the set C constructed by choosing points from the grid contains a discretized version of any optimal k-means solution with high probability. Second, we show that a k-means solution for the proxy dataset constructed using C works well for the dimension reduced dataset D'. Third, we derive cluster centers for the original dataset D by taking the average of all datapoints in each cluster.

The analysis of the first step proceeds as in Jones, Nguyen, and Nguyen (2020). We let $o_i = \{p \in D' : d(p, \mathtt{OPT}_{D'}) \in [r_{i-1}, r_i)\}$ and $a_i = D' \cap B_{r_i + t_i\sqrt{d'}}(C_i)$, where C_i is the set of grid points selected from G_i in the ith call to algorithm 2. The ith call to algorithm 2 would be successful if the grid points $C_i \subset G_i$ returned cover close to the maximum possible.

Algorithm 2: Private grid set cover

Data: D' dataset (passed by reference), t_i grid unit length, r_i threshold radius
Result: set $C_i \subset G_i$
$C_i \leftarrow \emptyset$
repeat k' **times**
  cover $\leftarrow$ empty linked list
  $V_i \leftarrow \{v : v \in \mathbb{N}^{d'}, \sum_{j=1}^{d'}(t_i v_j)^2 < (r_i + t_i\sqrt{d'})^2\}$
  for *all* $p \in D'$ **do**
    for *all* $v \in V_i$ **do**
      for *all* $s \in \{0,1\}^{d'}$ **do**
        $t_i b = \lfloor p \rfloor^{(i)} + t_i s + (2s - \bar{1})t_i v$;
        `// where` $\bar{1}$ `is the all-ones vector`
        if $d(t_i b, p) < (r_i + t_i\sqrt{d'})^2$ **then**
          cover$[t_i b]$ += $\{p\}$
        end
      end
    end
  end
  totalCover $\leftarrow 0$
  for $g \in$ cover **do**
    totalCover += $\exp\left(\frac{\epsilon_E |\text{cover}[g]|}{2}\right)$
  end
  totalCover += $|G_i| - len[\text{cover}]$
  Let $P_{samp} = 1 - \frac{|G_i|}{\text{totalCover}}$.
  if *Ber* $(P_{samp}) = 1$ **then**
    $g \leftarrow i \in [len[\text{cover}]]$ w.p. $\sim P(g) \propto \exp\left(\frac{\epsilon_E |\text{cover}[g]|}{2}\right) - 1$
  else
    $g \leftarrow$ uniformly at random from G_i
  end
  $C_i \leftarrow C_i \cup \{g\}$
  $D' \leftarrow D' \backslash \text{cover}[g]$
end
return C_i

Algorithm 3: NoisyAVG(Nissim, Stemmer, and Vadhan 2016, Algorithm 5)

Data: Multiset V of vectors in $\mathbb{R}^d$, predicate g, parameters ϵ, δ
Set
$\hat{m} = |\{v \in V : g(v) = 1\}| + \text{Lap}(5/\epsilon) - \frac{5}{\epsilon}\ln(2/\delta)$.
If $\hat{m} < 0$, output a uniformly random point in the domain $B_{\Delta/2}(0)$.
Denote $\sigma = \frac{5\Delta}{4\epsilon\hat{m}}\sqrt{2\ln(3.5/\delta)}$, and let $\eta \in \mathbb{R}^d$ be a random noise vector with each coordinate sampled independently from $N(0, \sigma^2)$.
return $g(V) + \eta$

Lemma 9. *If M_l is the maximum number of points that can be covered within distance $r_l + t_l\sqrt{d'}$ of k grid points in G_l, then with probability $1 - \gamma$*

$$|a_l| \geq (1-\epsilon)M_l - O\left(\frac{k\log n}{\epsilon_E \cdot \text{poly}(\epsilon)}\log\frac{n}{\gamma}\right).$$

where ϵ_E is the privacy parameter used in the exponential mechanism.

Sketch of proof. The set cover function that counts the number of datapoints that lie within $r_i + t_i\sqrt{d'}$ within any set of grid points C_i is submodular. It follows that greedily picking points by maximising the marginal increase in cover leads to covering $(1-\epsilon)$ as many points as the maximum, provided we pick $O(\log\lceil 1/\epsilon\rceil)$ as many grid points than there are in the optimal solution. Calls to the exponential mechanism lead to covers within logarithmic loss of the maximum and after accounting for these losses we get the stated bound. □

To use the fact that the number of datapoints covered in the ith call to algorithm 2 $|a_i|$ is close to $|o_i|$, we bound the optimal total movement of points when mapping each datapoint $p \in D'$ to its closest candidate grid point $g \in C$ in terms of the optimal clustering cost.

Lemma 10. *The thresholded cost obeys the bound*

$$\sum_{i=1}^{m}|o_i|r_i \leq (1+\epsilon)f_{D'}(\texttt{OPT}_{D'}) + 1. \tag{1}$$

Similarly, we can bound the actual increase in cost incurred from total distance moved by datapoints when constructing the proxy dataset D' in terms of a_i.

Lemma 11. *The total movement of points $p \in D'$ to the closest point* grid$[p] \in C$ *is can be bounded in terms of the a_i as follows:*

$$\sum_{p\in D'} d(p, \text{grid}[p]) \leq (1+\epsilon)\sum_{i=1}^{m}|a_i|r_i.$$

From the previous two lemmata and by deriving a relation between the sums $\sum_{i=1}^{m}|a_i|r_i$ and $\sum_{i=1}^{m}|o_i|r_i$, we can complete the first step of the proof.

Lemma 12. *The total movement of points $p \in D'$ to the closest point* grid$[p] \in C$ *can be bounded in terms of the optimal cost as follows:*

$$\sum_{p\in D'} d(p, \text{grid}[p]) \leq \left(1 + \frac{3\epsilon}{1-\epsilon-\epsilon^2}\right) f_{D'}(\texttt{OPT}_{D'}) + O\left(\frac{k\log n}{\epsilon_E \cdot \text{poly}(\epsilon)}\log\frac{n}{\gamma}\right).$$

Proof. Let $O_i = \sum_{j=i}^{m}|o_j|$ and $A_i = \sum_{j=i}^{m}|a_j|$. Then $\sum_{i=1}^{m}|a_i|r_i = \sum_{i=1}^{m}A_i(r_i - r_{i-1})$. Centers in $\texttt{OPT}_{D'}$ cover $n - O_{i+1}$ points at a maximum distance of r_i. We also know that algorithm 1 has already covered $n - A_i$ points at a distance of $r_{i-1} + t_{i-1}\sqrt{d'}$. It then follows that there are some k grid points in G_i (snapping the centers in $\texttt{OPT}_{D'}$ to grid)

that cover at least $A_i - O_{i+1}$ points in o_i. From the lemma 9 guarantee, we know

$$|a_i| \geq (1-\epsilon)(A_i - O_{i+1}) - E,$$

where $E = O\left(\frac{k \log n}{\epsilon_E \cdot \text{poly}(\epsilon)} \log \frac{n}{\gamma}\right)$. Since $A_i = |a_i| + A_{i+1}$, we have that

$$A_{i+1} \leq \left(\frac{\epsilon}{1-\epsilon}\right)|a_i| + O_{i+1} + \frac{E}{1-\epsilon}.$$

Substituting this in the telescoping $\sum_{i=1}^{m} A_i(r_i - r_{i-1})$,

$$\begin{aligned}\sum_{i=1}^{m}|a_i|r_i &\leq \sum_{i=1}^{m}\left(\frac{\epsilon|a_{i-1}|}{1-\epsilon} + O_i + \frac{E}{1-\epsilon}\right)(r_i - r_{i-1}) \\ &\leq \sum_{i=1}^{m}\left(\frac{\epsilon^2|a_{i-1}|}{1-\epsilon}\right)r_{i-1} + \sum_{i=1}^{m}|o_i|r_i + \frac{2E}{1-\epsilon}\end{aligned}$$

where to get from the second to the third line we use that $r_i - r_{i-1} = \epsilon r_{i-1}$, and that $r_m - r_0 = 2$. Rearranging, we get

$$\begin{aligned}&\sum_{i=1}^{m}|a_i|r_i\left(1 - \frac{\epsilon^2}{1-\epsilon}\right) \leq \sum_{i=1}^{m}|o_i|r_i + \frac{2E}{1-\epsilon} \\ &\Rightarrow \sum_{i=1}^{m}|a_i|r_i \leq \frac{1-\epsilon}{1-\epsilon-\epsilon^2}\sum_{i=1}^{m}|o_i|r_i + \frac{2E}{1-\epsilon-\epsilon^2}\end{aligned}$$

Substituting the order term E, using that ϵ is bounded away from 1 and applying the previous two lemmata we get the desired inequality. □

In lemma 13 we bound the k-means cost of the proxy dataset D'' in terms of the k-means cost of D'. Doing so requires us to account for the noisy counts used to construct the proxy dataset; this leads to the additional $O(\frac{k \log n}{\epsilon_L})$ error term. The proof of this step essentially follows from two applications of the triangle inequality, where since d is not a true metric we must gain a factor of 2 in the multiplicative loss for every application.

Lemma 13. *With probability* $1-\gamma$, $f_{D''}(OPT_{D'})$ *is at most* $\left(4 + \frac{6\epsilon}{1-\epsilon-\epsilon^2}\right) f_{D'}(OPT_{D'}) + O\left(\frac{k \log n}{\epsilon_E \cdot \text{poly}(\epsilon)} \log \frac{n}{\gamma}\right) + O\left(\frac{k \log n}{\epsilon_L \cdot \text{poly}(\epsilon)}\right)$.

In lemma 14, by essentially applying the triangle inequality, we bound the cost incurred by using the non-private k-means clustering solution for the proxy dataset D'' for the dataset D' completing the second step of the analysis.

Lemma 14. *Let* $\mathcal{A}$ *be the clustering algorithm used in algorithm 1 of algorithm 1. If* $\mathcal{A}$ *has a multiplicative loss of* E_M*, then* $f_{D'}(\mathcal{A}(D'')) \leq (8E_M + 2 + (8E_M + 4)\epsilon) f_{D'}(OPT_{D'}) + O\left(\frac{k \log n}{\epsilon_E \cdot \text{poly}(\epsilon)} \log \frac{kn}{\gamma}\right) + O\left(\frac{k \log n}{\epsilon_L \cdot \text{poly}(\epsilon)}\right)$.

To complete the utility analysis, we need to account for the projection and scaling as well as the Gaussian noise added to maintain privacy. In theorem 15 we account for the scaling and projection and then add the cost incurred due to the privacy preserving NoisyAVG of (Nissim, Stemmer, and Vadhan 2016). This gives us an upper bound for the net cost incurred by algorithm 1.

Theorem 15. *Algorithm 1 returns a set of points* $\tilde{S}$ *such that* $\mathbb{E}\left[f_D(\tilde{S})\right] \leq (1+\epsilon)\left(8 + \frac{12\epsilon}{1-\epsilon-\epsilon^2}\right)(E_M + 1) f_D(OPT_D) + O\left(\frac{k\Delta^2 \log n}{\epsilon_E \cdot \text{poly}(\epsilon)} \log \frac{n}{\gamma}\right) + O\left(\frac{k\Delta^2 \log n}{\epsilon_L \cdot \text{poly}(\epsilon)}\right) + O\left(\frac{k\Delta^2 \sqrt{d \log 1/\delta_G}}{\epsilon_G}\right) + O\left(\frac{k\Delta^2 \log n/\delta_G}{\epsilon_G}\right)$.

Sketch of proof. The scaling factor was picked according to the Johnson-Lindenstrauss (JL) transform to ensure that with high probability nothing needs to be projected, so we only need account for the re-scaling. We recall that the k-means clustering cost can be expressed only using the cluster sets $D_1, \ldots, D_k$ via the expression $\sum_{p \in D'} d(p, S) = \sum_{i=1}^{k} \frac{\sum_{p \neq q \in D_i} d(p,q)}{|D_i|}$. As the JL transform preserves the square of the Euclidean distance within a factor of $(1+\epsilon)$, the k-means cost can increase by a factor of at most $1+\epsilon$ when using the same clusters for D as were found in D'.

The final centers are computed using the NoisyAVG subroutine of Nissim, Stemmer, and Vadhan (2016). NoisyAVG modifies the well-known Gaussian mechanism by using a noisy version of the cluster size since in this application the cluster size is also private. For large clusters the noisy count does not increase the variance too much and for small clusters the worst-case cost is dominated by other error terms. This noisy averaging adds the $O\left(\frac{k\Delta^2 \sqrt{d \log 1/\delta_G}}{\epsilon_G}\right) + O\left(\frac{k\Delta^2 \log n/\delta_G}{\epsilon_G}\right)$ term to the clustering cost. □

Privacy

The main result of this section is the following:

Theorem 16. *Algorithm 1 is* $\left(\frac{e\epsilon_E \ln \delta_E^{-1}}{2} + \epsilon_L + \epsilon_G, \delta_E + \delta_G\right)$*-differentially private.*

From the basic and parallel composition laws of differential privacy and the privacy guarantees of the Laplace mechanism and algorithm 3 most of the expression for the bound on privacy loss claimed in this result follows relatively straightforwardly. To bound the privacy loss incurred in the calls to algorithm 2, we adapt a technique from Gupta et al. (2010). We use this technique in the following lemma to show that the privacy loss when using the exponential mechanism many times successively can be bounded as an expression of the sum of expected gains in the cover. For the set cover function this sum of expected gains can be shown to decay exponentially, which leads to a strong bound on the privacy loss.

Lemma 17. *The* m *calls to algorithm 2 from algorithm 1 that construct the set of centers* C *are collectively* $\left(\frac{e\epsilon_E \ln \delta_E^{-1}}{2}, \delta_E\right)$*-differentially private.*

Proof of theorem 16. First, we bound the loss in privacy that occurs when constructing the proxy dataset D''. From lemma 17 we know that in the m calls to algorithm 2 the net loss in privacy is $(\frac{e\epsilon_E \log \delta_E^{-1}}{2}, \delta_E)$. In the calculation of noisy counts we see that two neighbouring datasets can only differ in their true counts by 1 unit at one center of C. It follows that the ℓ_1 sensitivity of the tuple of all counts is 1 unit; this justifies the choice of parameter in the Laplace mechanism. Using basic composition along with the privacy loss bound for the Laplace mechanism we see that the net loss in privacy on releasing the proxy dataset D'' is $\left(\frac{e\epsilon_E \log \delta_E^{-1}}{2} + \epsilon_L, \delta_E\right)$.

Now D'' is publicly known and the low-dimensional domain can be partitioned by identifying each point in the domain with the closest point in the set returned by the non-private clustering algorithm used (a Voronoi diagram). To bound any further loss in privacy we use the parallel composition theorem of McSherry (2010) along with the privacy guarantee of algorithm 3. Since each application of algorithm 3 on the separate clusters is (ϵ_G, δ_G)-differentially private, by parallel composition the net privacy loss over all k applications is still (ϵ_G, δ_G). By basic composition the stated result follows. □

Experiments

We present an experimental comparison between algorithm 1[2], the private k-means clustering algorithm from Balcan et al. (2017), and the non-private Lloyd's algorithm. We are not aware of any other private clustering methods which have been implemented. Two datasets are used; a synthetic dataset reproducing the construction in Balcan et al. (2017) and the MNIST training dataset (Lecun et al. 1998).

The empirical results shown here for Balcan et al.'s algorithm (Balcan et al. 2017) come largely from their MATLAB implementation available on Github. Some corrections were made to the implementation of Balcan et al. (2017); although the pseudocode uses a noisy count of the cluster sizes when computing the noisy average of the clusters found their implementation used the non-private exact count. We replaced this subroutine with algorithm 3 to use the best method we know for privately computing the average.

Implementation details: We set $\epsilon = 1$ and $\delta = n^{-1.5}$ for both algorithms. Similar to Balcan et al. (2017), we project to a smaller subspace of dimension $\log(n)/2$ rather than $O(\log(n)/\epsilon^2)$ - this does not affect privacy. At the conclusion of both algorithms, we run one round of differentially private Lloyd's algorithm; adding this call to the differentially private Lloyd's yielded better empirical results for both algorithms. The addition of these rounds of Lloyd's requires adjusting privacy parameters by a constant factor but otherwise does not affect the privacy guarantees of the original algorithms. Being $(\epsilon, 0)$-private, Balcan et al. (2017) use the Laplace mechanism for their noisy average which we replaced with the noisyAVG routine of Nissim, Stemmer, and Vadhan (2016) for a comparison in the (ϵ, δ)-regime. Lloyd's algorithm was executed with 10 iterations.

[2]The code used for our experiments is available at https://github.com/Anamay-Chaturvedi/Differentially-private-k-means

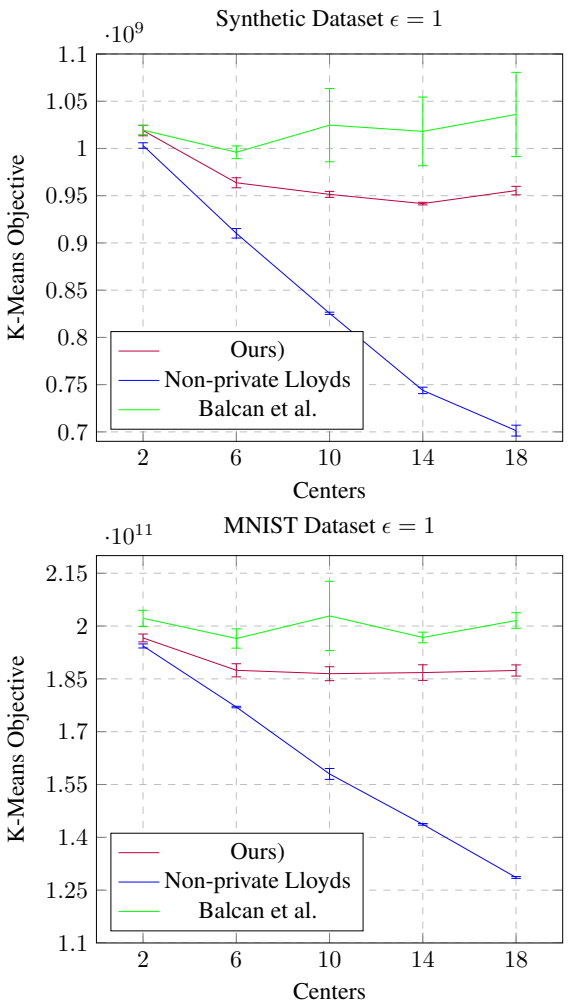

Figure 1: Empirical comparison of algorithm 1 and the private k-means clustering algorithm from (Balcan et al. 2017). Averages and standard deviations computed over 5 runs.

Datasets: The synthetic dataset is comprised of 50,000 points randomly sampled from a mixture of 64 Gaussians in $\mathbb{R}^{100}$. The MNIST training dataset uses the raw pixels; it is comprised of 60,000 points with 784 features each.

Results: As can be seen in fig. 1, our algorithm achieves a lower k-means objective score for both datasets. Similar to the experimental results in Balcan et al. (2017), increasing the number of centers results in a decrease in the cost in the non-private algorithm but did not result in a concomitant decrease in the cost of the private algorithms. This behavior suggests that these algorithms are limited by their additive error and that perhaps further decreasing even the constants in the additive error would improve the gap between them and their non-private counterparts.

Acknowledgements

This material is based upon work supported by the National Science Foundation under NSF grants NSF AF 1909314 and NSF CAREER 1750716. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

Aggarwal, A.; Deshpande, A.; and Kannan, R. 2009. Adaptive Sampling for k-Means Clustering. In Dinur, I.; Jansen, K.; Naor, J.; and Rolim, J. D. P., eds., *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, 12th International Workshop, APPROX 2009, and 13th International Workshop, RANDOM 2009, Berkeley, CA, USA, August 21-23, 2009. Proceedings*, volume 5687 of *Lecture Notes in Computer Science*, 15–28. Springer. doi:10.1007/978-3-642-03685-9_2. URL https://doi.org/10.1007/978-3-642-03685-9_2.

Ahmadian, S.; Norouzi-Fard, A.; Svensson, O.; and Ward, J. 2020. Better Guarantees for k-Means and Euclidean k-Median by Primal-Dual Algorithms. *SIAM J. Comput.* 49(4). doi:10.1137/18M1171321. URL https://doi.org/10.1137/18M1171321.

Balcan, M.; Dick, T.; Liang, Y.; Mou, W.; and Zhang, H. 2017. Differentially Private Clustering in High-Dimensional Euclidean Spaces. In Precup, D.; and Teh, Y. W., eds., *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, 322–331. PMLR. URL http://proceedings.mlr.press/v70/balcan17a.html.

Chen, K. 2009. On Coresets for k-Median and k-Means Clustering in Metric and Euclidean Spaces and Their Applications. *SIAM J. Comput.* 39(3): 923–947. doi:10.1137/070699007. URL https://doi.org/10.1137/070699007.

Dwork, C.; McSherry, F.; Nissim, K.; and Smith, A. D. 2016. Calibrating Noise to Sensitivity in Private Data Analysis. *J. Priv. Confidentiality* 7(3): 17–51. doi:10.29012/jpc.v7i3.405. URL https://doi.org/10.29012/jpc.v7i3.405.

Gupta, A.; Ligett, K.; McSherry, F.; Roth, A.; and Talwar, K. 2010. Differentially Private Combinatorial Optimization. In Charikar, M., ed., *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2010, Austin, Texas, USA, January 17-19, 2010*, 1106–1125. SIAM. doi:10.1137/1.9781611973075.90. URL https://doi.org/10.1137/1.9781611973075.90.

Jones, M.; Nguyen, H. L.; and Nguyen, T. 2020. Differentially Private Clustering via Maximum Coverage. *CoRR* abs/2008.12388. URL https://arxiv.org/abs/2008.12388.

Kanungo, T.; Mount, D. M.; Netanyahu, N. S.; Piatko, C. D.; Silverman, R.; and Wu, A. Y. 2004. A local search approximation algorithm for k-means clustering. *Comput. Geom.* 28(2-3): 89–112. doi:10.1016/j.comgeo.2004.03.003. URL https://doi.org/10.1016/j.comgeo.2004.03.003.

Lecun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 86(11): 2278–2324.

Lloyd, S. P. 1982. Least squares quantization in PCM. *IEEE Trans. Inf. Theory* 28(2): 129–136. doi:10.1109/TIT.1982.1056489. URL https://doi.org/10.1109/TIT.1982.1056489.

McSherry, F. 2010. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. *Commun. ACM* 53(9): 89–97. doi:10.1145/1810891.1810916. URL https://doi.org/10.1145/1810891.1810916.

McSherry, F.; and Talwar, K. 2007. Mechanism Design via Differential Privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2007), October 20-23, 2007, Providence, RI, USA, Proceedings*, 94–103. IEEE Computer Society. doi:10.1109/FOCS.2007.41. URL https://doi.org/10.1109/FOCS.2007.41.

Nissim, K.; Stemmer, U.; and Vadhan, S. P. 2016. Locating a Small Cluster Privately. In Milo, T.; and Tan, W., eds., *Proceedings of the 35th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2016, San Francisco, CA, USA, June 26 - July 01, 2016*, 413–427. ACM. doi:10.1145/2902251.2902296. URL https://doi.org/10.1145/2902251.2902296.

Ostrovsky, R.; Rabani, Y.; Schulman, L. J.; and Swamy, C. 2012. The effectiveness of lloyd-type methods for the k-means problem. *J. ACM* 59(6): 28:1–28:22. doi:10.1145/2395116.2395117. URL https://doi.org/10.1145/2395116.2395117.

Stemmer, U.; and Kaplan, H. 2018. Differentially Private k-Means with Constant Multiplicative Error. In Bengio, S.; Wallach, H. M.; Larochelle, H.; Grauman, K.; Cesa-Bianchi, N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, 5436–5446. URL http://papers.nips.cc/paper/7788-differentially-private-k-means-with-constant-multiplicative-error.