Esercises on Clustering

Exercise. Show that the L_1 (Euclidean) and Edit distances satisfy the four requirements for a metric space.

Solution. Recall that given a set M with distance function $d(\cdot)$, (M, d) is a *metric space* if the following conditions hold for any $x, y, x \in M$:

1. $d(x, y) \geq 0$;
2. $d(x, y) = 0$ if and only if $x = y$;
3. $d(x, y) = d(y, x)$; (symmetry)
4. $d(x, z) \leq d(x, y) + d(y, z)$; (triangle inequality)

L_1 **distance.** Recall that the L_1 distance between two points $x, y \in \Re^n$ is

$$d_{\mathrm{L1}}(x, y) = \sum_{i=1}^{n} |x_i - y_i|,$$

where the x_i's and y_i's denote the coordinates. We observe that for any three real numbers a, b and c, we have $|a - b| \leq |a - c| + |c - b|$, which can be easily proved considering all possible relative positions of a, b and c on the line. Now we prove that the four conditions above are satisfied.

1. $d_{\mathrm{L1}}(x, y) \geq 0$. It follows since all terms of the summation are ≥ 0.
2. $d_{\mathrm{L1}}(x, y) = 0$ if and only if $x = y$. It follows since, in order for $d_{\mathrm{L1}}(x, y)$ to be 0, all terms of the summation must be 0, hence $x_i = y_i$ for every i.
3. $d_{\mathrm{L1}}(x, y) = d_{\mathrm{L1}}(y, x)$. It follows since $|a - b| = |b - a|$ for any two reals a and b.
4. $d(x, z) \leq d(x, y) + d(y, z)$. By the initial observation, we have that

$$\begin{aligned} d_{\mathrm{L1}}(x, z) &= \sum_{i=1}^{n} |x_i - z_i| \\ &\leq \sum_{i=1}^{n} (|x_i - y_i| + |y_i - z_i|) \\ &= (\sum_{i=1}^{n} |x_i - y_i|) + (\sum_{i=1}^{n} |y_i - z_i|) \\ &= d_{\mathrm{L1}}(x, y) + d_{\mathrm{L1}}(y, z) \end{aligned}$$

Edit distance. Recall that the edit distance is defined for strings. For any two strings X e Y, over some alphabet, $d_{\text{edit}}(X,Y)$ is the minimum number of deletions or insertions that must be applied to transform X into Y. Equivalently, it holds that

$$d_{\text{edit}}(X,Y) = |X| + |Y| - 2|\text{LCS}(X,Y)|,$$

where $\text{LCS}(X,Y)$ is the Longest Common Subsequence in X and Y, that is, the longest string that is a subsequence of both X and Y (note that if Z is a subsequence of W, this means that the characters of Z occurr in W in the same order but, possibily, not consecutively). It is immediate to see that $|X|,|Y| \geq |\text{LCS}(X,Y)|$. Now we prove that the four conditions above are satisfied.

1. $d_{\text{edit}}(X,Y) \geq 0$. It follows since $|X|,|Y| \geq |\text{LCS}(X,Y)|$
2. $d_{\text{edit}}(X,Y) = 0$ if and only if $X = Y$. It follows since the relation $|X|,|Y| \geq |\text{LCS}(X,Y)|$ implies that $d_{\text{edit}}(X,Y) = 0$ if and only if $|\text{LCS}(X,Y)| = |X| = |Y|$, i.e., $\text{LCS}(X,Y) = X = Y$.
3. $d_{\text{edit}}(X,Y) = d_{\text{edit}}(Y,X)$. It follows since the role of X and Y in the second definition is perfectly symmetrical.
4. $d_{\text{edit}}(X,Z) \leq d_{\text{edit}}(X,Y) + d_{\text{edit}}(Y,Z)$. Considering the first definition, we can transform X into Z by first transforming X into Y, and then Y into Z. Therefore, the minimum number of deletions or insertions required to transform X into Z cannot be larger than the minimum number of such operations required to transform X into Y plus the minimum number of such operations required to transform Y into Z.

$\square$

Exercise. Consider a set P of N points in a metric space (M,d) and a subset $S \subseteq P$ of k centers from P. Design a 1-round MapReduce algorithm that implements $\text{Partition}(P,S)$ with $M_L = O(k)$ and $M_A = O(N)$. Suppose that each point $x \in P$ is represented as a key-value pair $(\text{ID}_x,(x,f))$, where ID_x is a distinct integer in $[0,N)$, and f is a binary flag which is 1 if $x \in S$ and 0 otherwise. You can assume that k and N are known and that k divides N. You must describe the map and reduce phases of the round, specifying the intermediate and output key-value pairs.

Solution. The algorithm is the following.

Round 1

- *Map phase*: map each pair $(\text{ID}_x,(x,f))$ with $f = 0$ into the intermediate pair $(\text{ID}_x \bmod N/k,(x,f))$, and each pair $(\text{ID}_x,(x,f))$ with $f = 1$ into the N/k pairs $(i,(x,f))$, with $0 \leq i < N/k$. Let P_i be the set of intermediate pairs $(i,(x,f))$ with $f = 0$ and S_i the set of intermediate pairs $(i,(x,f))$ with $f = 1$. (Note that S_i is a replica of the set of centers S.)

- *Reduce phase*: For each key $i \in [0, N/k)$ independently, gather the set P_i and S_i. For each $(i,(x,0)) \in P_i$ determine the pair $(i,(y,1)) \in S_i$ such that y is the closest center to x, breaking ties arbitrarily, and output the pair (x,y) (i.e., x is assigned to the cluster with center y. Also, if $i = 0$, for each $(i,(y,1)) \in S_i$, output the pair (y,y) (i.e., y is assigned to the cluster centered at itself).

The map phase requires constant local space, while in the reduce phase, each reducer gathers a subset P_i of size k and a replica S_i of the set of k centers. Therefore, $M_L = O(k)$. As for the aggregate space, each of the k centers is replicated N/k times, while other points are not replicated. Altogether, we have $M_A = O(N)$. □

Exercise. Let P be a set of points in a metric space (M,d), and let $T \subseteq P$. For any $k < |T|$, show that $\Phi^{\text{opt}}_{\text{kcenter}}(T,k) \leq 2\Phi^{\text{opt}}_{\text{kcenter}}(P,k)$, where $\Phi^{\text{opt}}_{\text{kcenter}}(X,k)$ is the minimum value of $\Phi_{\text{kcenter}}(\mathcal{C})$ over all possible k-clusterings $\mathcal{C}$ of a pointset X.

Solution. The solution is essentially embodied in the proof of the approximation bound for the MR-Farthest-First Traversal algorithm (Slides 47-49 on clustering, Part 1), where the application of the Farthest-First Traversal algorithm on subsets of a pointset P was analyzed. Let $S = \{c_1, c_2, \ldots, c_k\}$ be the k centers returned by the Farthest-First Traversal algorithm when applied on T, and let q be the point of T with maximum distance from S. Clearly, each point in T is at distance $\leq d(q,S)$ from S, therefore the clustering $\mathcal{C}$ of T induced by the centers in S is such that $\Phi_{\text{kcenter}}(\mathcal{C}) = d(q,S)$. This also implies that $\Phi^{\text{opt}}_{\text{kcenter}}(T,k) \leq \Phi_{\text{kcenter}}(\mathcal{C}) = d(q,S)$. Now, as claimed in the analysis of the MR-Farthest-First Traversal algorithm, we have that the set $\{c_1, c_2, \ldots, c_k, q\}$ comprises $k+1$ points of T, hence of P, whose pairwise distances are all $\geq d(q,S)$. Since at least two of these points must belong to the same cluster in the optimal k-clustering of P, we have that $d(q,S) \leq 2\Phi^{\text{opt}}_{\text{kcenter}}(P,k)$. By combining the various inequalities we obtain

$$\Phi^{\text{opt}}_{\text{kcenter}}(T,k) \leq d(q,S) \leq 2\Phi^{\text{opt}}_{\text{kcenter}}(P,k).$$

□

Observation. The previous exercise raises the natural question whether the relation proved there is tight, in the sense that there exists a pointset P, a subset $T \subseteq P$, and a value k such that $\Phi^{\text{opt}}_{\text{kcenter}}(T,k) = 2\Phi^{\text{opt}}_{\text{kcenter}}(P,k)$. Indeed, this is the case. For given k, consider the following set P of $k+2$ points in $\Re$ (i.e., on the line):

$$P = \{-1, 0, 1, x, 2x, \ldots, (k-1)x\},$$

where x is very large (e.g., x=1000). It is easy to see that an optimal k-clustering of P uses, as centers, the points ix, with $0 \leq i \leq k-1$, and that $\Phi^{\text{opt}}_{\text{kcenter}}(P,k) = 1$. Let

$$T = \{-1, 1, x, 2x, \ldots, (k-1)x\} \subseteq P.$$

It is easy to see that an optimal k-clustering of T uses, as centers, the points ix, with $1 \leq i \leq k-1$, plus either 1 or -1, and that $\Phi^{\text{opt}}_{\text{kcenter}}(T,k) = 2$.

Exercise. Let P be a set of N points in a metric space (M, d), and let $T \subseteq P$ be a coreset of $|T| > k$ points such that for each $x \in P$, we have $d(x, T) < \Phi^{\rm opt}_{\rm kcenter}(P, k)$, where $\Phi^{\rm opt}_{\rm kcenter}(P, k)$ is the minimum value of $\Phi_{\rm kcenter}(\mathcal{C})$ over all possible k-clusterings $\mathcal{C}$ of P.

1. Devise a MapReduce algorithm which receives in input P and T and computes a good solution for the k-center clustering problem on P, using local space $M_L = O(|T|)$ and aggregate space $M_A = O(N)$. You need not describe in detail map and reduce phases and key-value pairs. Also, you can assume that for a set of k centers $S \subseteq P$, the primitive Partition(P, S) can be implemented in MapReduce in 1 round, using local space proportional to k and linear aggregate space.

2. Determine the approximation ratio achieved by your algorithm.

Solution.

1. The algorithm is the following:

   - Round 1: gather T into one reducer and compute a set $S \subseteq T$ of k centers using the Farthest-First Traversal algorithm on T.
   - Round 2: Execute Partition (P, S) to compute the final clustering.

   The first round requires local space proportional to $|T|$ and linear aggregate space, while the primitive Partition, invoked in the second round, can be executed with local space proportional to k and linear aggregate space. Hence, since $|T| > k$, the algorithm needs $M_L = O(|T|)$ and $M_A = O(N)$.

2. Let q be the point of T with maximum distance from the centers of S (denote this distance by $d(q, S)$). By reasoning as in the proof of the approximation bound for the MR-Farthest-First Traversal algorithm (Slides 47-49 of the slides on Clustering Part 1), we can show that $d(q, S) \leq 2\Phi^{\rm opt}_{\rm kcenter}(P, k)$. Consider a point $x \in P$. From the hypothesis on T we know that there is a point $y \in T$ such that $d(x, y) \leq \Phi^{\rm opt}_{\rm kcenter}(P, k)$. Also, by the properties of S we known that there is a center $c \in S$ such that $d(y, c) \leq d(q, S)$. These considerations yield that

$$\begin{aligned} d(x,c) &\leq d(x,y) + d(y,c) \\ &\leq \Phi^{\rm opt}_{\rm kcenter}(P,k) + d(q,S) \\ &\leq \Phi^{\rm opt}_{\rm kcenter}(P,k) + 2\Phi^{\rm opt}_{\rm kcenter}(P,k) \\ &= 3\Phi^{\rm opt}_{\rm kcenter}(P,k). \end{aligned}$$

   Therefore, the clustering $\mathcal{C}$ returned by the algorithm is such that $\Phi_{\rm kcenter}(\mathcal{C}) \leq 3\Phi^{\rm opt}_{\rm kcenter}(P, k)$, that is, the approximation ratio is 3.

□

Exercise. Show how to implement one iteration of Lloyd's algorithm in MapReduce using a constant number of rounds, local space $M_L = O\left(\sqrt{N}\right)$ and aggregate space $M_A = O(N)$, when $k = O\left(\sqrt{N}\right)$. Assume that at the beginning of the iteration a set S of k centers and a current estimate Φ of the objective function are available. The iteration must compute the partition of P around S, the new set S' of k centers, and it must update Φ, unless it is the last iteration. You need not describe in detail map and reduce phases and key-value pairs.

Solution. The implementation is the following. Let $S = \{c_1, c_2, \ldots, c_k\}$.

- Round 1: Partition P arbitrarily into $\sqrt{N}$ subsets P_i, with $0 \le i < \sqrt{N}$, of size $\sqrt{N}$ each. For each $0 \le i < \sqrt{N}$ independently, gather P_i and a copy of S and compute

$$\begin{aligned} P_{ij} &= \{x \in P_i \ : \ d(x, S) = d(x, c_j)\} \\ X_{ij} &= \sum_{x \in P_{ij}} x \\ \text{num}_{ij} &= |X_{ij}|. \end{aligned}$$

- Round 2: for each $1 \le j \le k$ independently, gather the values X_{ij} and num_{ij}, with $0 \le i < \sqrt{N}$, and compute

$$c'_j = \frac{\sum_{i=0}^{\sqrt{N}-1} X_{ij}}{\sum_{i=0}^{\sqrt{N}-1} \text{num}_{ij}}.$$

  Let $S' = \{c'_1, c'_2, \ldots, c'_k\}$.

- Round 3: for each $0 \le i < \sqrt{N}$ independently, gather P_i and a copy of S' and compute

$$\phi_i = \sum_{x \in P_i} d^2(x, S').$$

- Round 4: gather all ϕ_i's into one reducer and compute

$$\Phi' = \sum_{i=0}^{\sqrt{N}-1} \phi_i.$$

  If $\Phi' < \Phi$ then update Φ with the new value Φ'.

Each round requires local space $O\left(k + \sqrt{N}\right) = O\left(\sqrt{N}\right)$. As for the aggregate space, the $\sqrt{N}$ copies of S and S' take, altogether, space $O(N)$, and all other data also require space $O(N)$. $\square$

Exercise. Consider a set P of N points in $\Re^D$. Let $\Phi^{\text{opt}}_{\text{kmeans}}(k)$ be the minimum value of $\Phi_{\text{kmeans}}(\mathcal{C})$ over all possible k-clusterings $\mathcal{C}$ of P, and let $\mathcal{C}_{\text{alg}} = \text{Partition}(S, P)$ be the k-clustering of P induced by the set S of centers returned by the k-means++.

1. Using Markov's inequality determine a value $c(k) > 0$ such that

$$\Pr\left(\Phi_{\text{kmeans}}(\mathcal{C}_{\text{alg}}) \le c(k) \cdot \Phi^{\text{opt}}_{\text{kmeans}}(k)\right) \ge 1/2.$$

   Recall that for a real-valued nonnegative random variable X with expectation $E[X]$ and a value $a > 0$, the Markov's inequality states that

$$\Pr(X \ge a) \le \frac{E[X]}{a}.$$

2. Show that by running $\log_2 N$ independent instances of k-means++ and by taking the best clustering $\mathcal{C}_{\text{best}}$ found among all repetitions, we have that

$$\Pr\left(\Phi_{\text{kmeans}}(\mathcal{C}_{\text{best}}) \le c(k) \cdot \Phi^{\text{opt}}_{\text{kmeans}}(k)\right) \ge 1 - 1/N.$$

Solution.

1. Because of the random choices made by k-means++, we have that $\Phi_{\text{kmeans}}(\mathcal{C}_{\text{alg}})$ is a real-valued nonnegative random variable, whose expectation is

$$E[\Phi_{\text{kmeans}}(\mathcal{C}_{\text{alg}})] \le 8(\ln k + 2) \cdot \Phi^{\text{opt}}_{\text{kmeans}}(k).$$

   Let us define $c(k) = 16(\ln k + 2)$. By Markov's inequality we have that

$$\Pr(\Phi_{\text{kmeans}}(\mathcal{C}_{\text{alg}}) \ge c(k) \cdot \Phi^{\text{opt}}_{\text{kmeans}}(k)) \le \frac{E[\Phi_{\text{kmeans}}(\mathcal{C}_{\text{alg}})]}{c(k) \cdot \Phi^{\text{opt}}_{\text{kmeans}}(k)} \le \frac{1}{2}.$$

   This implies that $\Pr(\Phi_{\text{kmeans}}(\mathcal{C}_{\text{alg}}) \le c(k) \cdot \Phi^{\text{opt}}_{\text{kmeans}}(k)) \ge 1/2$.

2. Let $\mathcal{C}_i$ be the k-clustering obtained in the i-th instance of k-means++, for $1 \le i \le \log_2 N$. Since the $\log_2 N$ instances are independent, we have that

$$\begin{aligned}\Pr(\Phi_{\text{kmeans}}(\mathcal{C}_{\text{best}}) \ge c(k) \cdot \Phi^{\text{opt}}_{\text{kmeans}}(k)) &= \Pi_{i=1}^{\log_2 N} \Pr(\Phi_{\text{kmeans}}(\mathcal{C}_i) \ge c(k) \cdot \Phi^{\text{opt}}_{\text{kmeans}}(k)) \\ &\le \frac{1}{2^{\log_2 N}} = \frac{1}{N}.\end{aligned}$$

   Therefore, $\Pr(\Phi_{\text{kmeans}}(\mathcal{C}_{\text{best}}) \le c(k) \cdot \Phi^{\text{opt}}_{\text{kmeans}}(k)) \ge 1 - 1/N$.

□

Exercise. Show that the PAM algorithm always terminates.

Solution. Refer to the pseudocode of the algorithm (see Slide 22 of the slides on Clustering Part 2). Let $\mathcal{C}_i$ be the clustering stored by variable $\mathcal{C}$ at the end of the i-th iteration of the while-loop, where $\mathcal{C}_0$ is the initial clustering. Note that each $\mathcal{C}_i$ is obtained by invoking the primitive Partition on some set of centers, hence it is uniquely determined by the centers. Also, by construction, the values $\Phi_{\text{kmedian}}(\mathcal{C}_i)$ are strictly decreasing with i, except for the last value. Therefore, the sets of centers that yield the $\mathcal{C}_i$'s *must be* all distinct except for the last one. This immediately implies that the number of iterations cannot be more than the number of subsets of k points, which is $\binom{N}{k}$, where N is the number of input points. □

Exercise. Let P be a dataset of N points in some metric space (M, d), let X be a random sample of t points from P, and let Y be a random sample of t points from M, for some $t = O\left(\sqrt{N}\right)$. Show that the Hopkins statistic $H(P)$ can be efficiently computed in MapReduce, assuming that P, X and Y are given as input. You need not describe in detail map and reduce phases and key-value pairs, but may assume that initially each point comes with a distinct key in $[0, N-1]$.

Solution. The algorithm is the following.

- Round 1. Do the following:
  - Partition P into $\sqrt{N}$ subsets P_j, with $0 \le j < \sqrt{N}$, of $\sqrt{N}$ points each
  - Create $\sqrt{N}$ copies of both X and Y.
  - For every $0 \le j < \sqrt{N}$ independently, gather P_j, a copy of X, a copy of Y, and compute $w_x(j) = \min_{z \in P_j, z \neq x} d(x, z)$, for each $x \in X$, and $u_y(j) = \min_{z \in P_j, z \neq y} d(y, z)$, for each $y \in Y$.
- Round 2. For each $x \in X$ (resp., $y \in Y$), independently, gather all $w_x(j)$'s (resp., $u_y(j)$'s) and compute $w_x = \min_{0 \le j < \sqrt{N}} w_x(j)$ (resp., $u_y = \min_{0 \le j < \sqrt{N}} u_y(j)$)
- Round 3. Gather all values w_x, with $x \in X$, and all values u_y, with $y \in Y$ and compute
$$H(P) = \frac{\sum_{y \in Y} u_y}{\sum_{y \in Y} u_y + \sum_{x \in X} w_x}.$$

Each round requires local space $O\left(\sqrt{N} + t\right) = O\left(\sqrt{N}\right)$. As for the aggregate space, the $\sqrt{N}$ copies of X and Y take, altogether, space $O\left(t\sqrt{N}\right) = O(N)$, and all other data also require space $O(N)$. □

Exercise. Let P be a set of N points in a metric space (M, d), and let $\mathcal{C} = (C_1, C_2, \ldots, C_k; c_1, c_2, \ldots, c_k)$ be a k-clustering of P. Design and analyze an efficient MapReduce algorithm that for each cluster cluster center $c \in \{c_1, \ldots, c_k\}$ determines the most distant point among those belonging to the cluster centered at c. (Assume that all distances between pairs of points are distinct.) Initially, each point $q \in P$ is represented by a pair $(\mathrm{ID}(q), (q, c(q)))$, where $\mathrm{ID}(q)$ is a distinct key in $[0, N-1]$ and $c(q) \in \{c_1, \ldots, c_k\}$ is the center of the cluster of q. Specify map and reduce phases, and intermediate and output pairs of each round. To get full score, the algorithm must use $o(N)$ local space and linear aggregate space.

Solution. The algorithm is the following:

Round 1

- *Map phase*: each pair $(\mathrm{ID}(q), (q, c(q)))$ is mapped into the intermediate pair $(j, (q, c(q)))$, with $j = \mathrm{ID}(q) \bmod \sqrt{N}$.
- *Reduce phase*: For each $0 \le j < \sqrt{N}$ independently, gather the set S_j of intermediate pairs with key j and select, for each center c, the most distant point q such that $(j, (q, c(q) = c)) \in S_j$ (if any such pair exists). Represent each selected point q as a new pair $(c(q), q)$, where $c(q)$ is the key.
  Observation: At the end of the round there will be at most $\sqrt{N}$ pairs with the same key c.

Round 2

- *Map phase*: identity.
- *Reduce phase*: for each center c independently, gather all pairs (c, q) (where $c = c(q)$) produced in the previous round, and return the pair with the largest values of $d(c, q)$.

Correctness immediately follows by the observation that for each center c, if the most distant point q from the center is represented by a pair $(j, (q, c(q) = c)) \in S_j$, such a pair will be surely be selected in the first round, hence q will be identified as the most distant point from c in the second round. Since each S_j has size $\Theta\left(\sqrt{N}\right)$ and, in the first round, at most $\sqrt{N}$ points are selected for each cluster, the local space is $M_L = \Theta\left(\sqrt{N}\right)$. Also, since for each point a constant number of pairs are ever used, the aggregate space linear in N. □