Unsupervised Learning HW1

Due: Fri Oct 5, 2018 at 11:59pm

All homeworks (including this one) should be typesetted properly in pdf format. Late homeworks or handwritten solutions will not be accepted. You must include your name and UNI in your homework submission. To receive credit, a typesetted copy of the homework pdf must be uploaded to Gradescope by the due date. You must show your work to receive full credit. Discussing possible solutions for homework questions is encouraged on piazza and with your peers, but everyone must write their own individual solutions. You must cite all external references you used (including the names of individuals you discussed the solutions with) to complete the homework.

Note: Questions 3 and 6 are optional extra credit questions.

1 **[Readings]** Read any two papers of your choice from the following list, summarize their main results, discuss their significance and provide a short proof sketch of their technical results.

- "k-means++: The Advantages of Careful Seeding" by Arthur and Vassilvitskii
- "Clustering with Interactive Feedback" by Balcan and Blum
- "Incremental clustering: the case for extra clusters" by Ackerman and Dasgupta
- "Comparing Clusterings – An Axiomatic View" by Meila
- "Hartigan's Method: k-means Clustering without Voronoi" by Telgarsky and Vattani
- "Clustering large graphs via the singular value decomposition" by Drineas, Kannan, Frieze, Vempala, and Vinay

2 **[Hardness of k-center]** Show that the k-center problem is NP-hard.

(Hint: You can use the fact that the following variation of the Vertex-Cover problem is NP-hard.

Vertex-Cover*

- **Input:** An undirected unweighted graph $G = (V, E)$, with vertices V and edges E.
- **Output:** $V' \subseteq V$, such that $V' \cup \Big(\cup_{v' \in V'} \cup_{e(v',v) \in E} \; v \Big) = V$.
- **Goal:** minimize $|V'|$.

)

3 **[Optional – balanced k-means]** Consider the special case of k-means problem, where one is given n datapoints that is a multiple of the number of clusters k. The goal is to assign each cluster exactly n/k datapoints while minimizing the k-means cost function.

Provide an efficient exact algorithm for this problem, or show that it is NP-hard.

4 **[Coverings and Packings]** Given a metric space (X, d), and an $\epsilon > 0$, a set $C \subset X$ is called an ϵ-*cover* if $\forall x \in X, \exists c \in C$ such that, $d(x, c) \leq \epsilon$, a set $P \subset X$ is called an ϵ-*packing* if $\forall p, p' \in P$ distinct, $d(p, p') > \epsilon$.

(i) Show that for any compact metric space (X, ρ) and any $\epsilon > 0$, there exists $Y \subset X$, such that Y is both an ϵ-cover and an ϵ-packing.

ϵ-covering number of X, denoted by $N_\epsilon(X)$, is defined to be the size of *smallest* ϵ-cover of X, similarly ϵ-packing number of X, denoted by $P_\epsilon(X)$, is defined to be the size of *largest* ϵ-packing of X.

(ii) Show that for any metric space (X, ρ) and any $\epsilon > 0$,

$$P_\epsilon(X) \leq N_{\epsilon/2}(X) \leq P_{\epsilon/2}(X).$$

(iii) Let $B^d(x, r)$ denote the closed Euclidean ball of radius r centered at x in $\mathbb{R}^d$, that is $B^d(x, r) := \{p \in \mathbb{R}^d \mid \|x - p\|_2 \leq r\}$. Give an estimate of the ϵ-covering number of $B^d(0, 1)$ (ie, closed Euclidean unit ball in $\mathbb{R}^d$, centered at the origin).

(Hint: consider a maximal size packing of $B^d(0, 1)$ and compare the relative d-dimensional volumes of balls centered at elements of the packing with a ball containing all of them, using the fact that vol$(B^d(x, r)) = r^d$vol$(B^d(x, 1)$)

(iv) **[Application: approximating maximum singular value using a cover]** Given a $m \times n$ matrix A, recall that the maximum singular value of A is defined as

$$\sigma_{\max}(A) := \max_{x \in \mathbb{R}^n, \|x\|=1} \|Ax\|$$

An (impractical) solution to finding $\sigma_{\max}(A)$ is to test the condition $\|Ax\|$ on each $x \in S^{n-1} := \{x \in \mathbb{R}^n : \|x\| = 1\}$ and return the maximum value. The obvious problem with this approach is of course that the set S^{n-1} is of infinite size and thus one cannot return $\sigma_{\max}$ in finite time using such an approach.
One obvious remedy is to approximate $\sigma_{\max}$ by testing the value $\|Ax\|$ on a finite (albeit large) collection of x and approximates S^{n-1} well. More concretely, one can construct a *cover* C of S^{n-1}, and find the maximum $\|Ax\|$ on the each $x \in C$ (let's denote that as σ_C) and relate it to $\sigma_{\max}$.

Give a tight upperbound on the approximation $\sigma_{\max}$ as a function of σ_C, and the quality of the cover C.

5 **[Basic Properties of Laplacian]** Let $G = (V, E)$ be a graph. Let d be its derivative operator and L be its graph Laplacian.

(i) Let $S \subset E$ be a collection of edges. Denote by d_S the matrix minor of d where only the edges in S have rows in d_S. What graph does d_S correspond to?

(ii) Let $E_1, \ldots, E_k$ be a partition of the edge set E. That is, $E = E_1 \sqcup \cdots \sqcup E_k$. Let L_i be the Laplacian of the graph (V, E_i). Show that:

$$L = \sum_{i=1}^{k} L_i.$$

(iii) Let $G' = (V', E')$ be another graph, and L' be its graph Laplacian. Describe the corresponding Laplacian for the graph $\mathcal{G} = (V \sqcup V', E \sqcup E')$ created by the disjoint union of G and G'.

(iv) Let $|V| = n$. Give a tight upper bound for $\mathrm{rank}(L)$. (By tight, there exists a graph with n vertices whose rank achieves your bound). What vector(s) does $\ker(L)$ always contain?

6 **[Optional Problem – Trees and Laplacians]** Recall that a *tree* is a connected graph with no cycles.

(i) Let $G = (V, E)$ be a connected graph. Show that the following are equivalent:

- G is a tree.
- G is maximally acyclic. That is, adding any edge to G will create a cycle.
- $|E| = |V| - 1$.

(ii) Let d be the derivative on a graph G. Show that d is full rank if and only if G has no cycles.

(iii) Let T be a tree, and d the derivative operator on T. Denote by $\hat{d}$ the matrix minor of d produced by removing its first column. Prove:

$$\det(\hat{d}^T \hat{d}) = 1.$$

(iv) Let L be the Laplacian of a connected graph. Denote by $L_{1,1}$ the matrix minor of L produced by removing the first row and column. Use Cauchy-Binet to show that:

$$\det(L_{1,1}) = \text{\# of spanning trees of } G.$$

Hint: you may use part (iii) without proof.

7 **[Connectivity and the Laplacian]** Let G be a graph with n vertices and k connected components.

(i) Show that the number of nonzero eigenvalues of L is precisely $n - k$.

(ii) Here, we'll take $k = 1$, so that G is connected. Consider the following one-dimensional graph embedding, motivated from the mass-spring problem. Each vertex corresponds to a point mass, and each edge corresponds to a spring (assume the unit mass/spring constant). Let $f \in \mathbb{R}^V$ define an embedding of G into $\mathbb{R}^1$.

To define the problem, we need two concepts from physics:

- *energy*: a spring that is stretched a distance ℓ contributes ℓ^2 of units of energy to the total spring system. Thus, if each spring $e \in E$ is stretched to length ℓ_e, the amount of energy $\mathcal{E}$ required is:

$$\mathcal{E} = \sum_{e \in E} \ell_e^2.$$

– *moment of inertia*: a mass that is a distance of x from the center of mass contributes x^2 to the moment of inertia. Thus, if each vertex $i \in V$ is a distance of x_i from its *center of mass*, then its moment of inertia I is:

$$I = \sum_{i \in V} x_i^2.$$

These two concepts are quite useful from a clustering/embeddings perspective. As more *energy* is required to separate vertices that share many edges, we'd want to minimize the energy, to keep similar points together. On the other hand, a maximizing the *moment of inertia* allows us to discern different points better.

Thus, our optimization question: can we embed the graph in such a way that keeps similar points together (minimize energy) while separating natural clusters (maximize moment of inertia)? In particular, suppose that all the masses are initially at 0 (i.e. the position is described by $0 \in \mathbb{R}^V$; the initial moment of inertia is 0. **Prove a tight lower bound on the energy required to increase the moment of inertia of the spring system from 0 to 1. What embedding $f \in \mathbb{R}^V$ achieves this bound?** *Hint: can you write the energy in terms of the Laplacian? For which $f \in \mathbb{R}^V$ is the center of mass at 0?*