Unsupervised Learning (2018 Fall)
Homework #1

Youki Cao -

Oct 5 2018

Problem 1

- "k-means++: The Advantages of Careful Seeding" by Arthur and Vassilvitskii

The paper proposes a great improvement for the initializing part in the k-means algorithm. The classical k-means method offers no accuracy guarantees, but k-means++ is O($\log\ k$)-competitive with optimal clustering. Instead of choosing the centers at random, k-means++ weighs the data points according to their squared distance from the closest center that has already been chosen. The significance of it lies in the enhancement of both speed and accuracy.

Then I'll show some technical analysis of the paper. The main idea is to prove the expectation of the potential function ϕ is $O(\log k)$-competitive. Firstly, we need to prove theorem 3.1: If C is constructed with k-means++, then eht corresponding potential function ϕ satisfies, $E[\phi] \leq 8(\ln k + 2)\phi_{OPT}$. To prove it, we need to clarify two lemmas: the first is Lemma 3.2, let A be a cluster from C_{OPT}, let C be just one cluster uniformly at random from A, then $E[\phi(A)] \leq 2\phi_{OPT}(A)$. To prove it, we need to use a small conclusion $E[||x-y||^2] = 2E[||x-E(x)||^2]$. Lemma 3.2 ensures us the first step of k-means++. For remaining centers, we need to use lemma 3.3: let A be an arbitrary cluster in C_{OPT}, and let C be an arbitrary clustering. If we add a random center to C from A, chosen with D^2 weighting, then $E[\phi(A)] \leq 8\phi_{OPT}(A)$. So far we know that as long as it chooses centers from each cluster of C_{OPT}, then the algorithm is competitive. Finally we use lemma 3.4 to obtain the desired bound $E[\phi]$. And then we can finally prove the theorem.

The next part is to show that k-means++ is no better than $\Omega(\log k)$ competitive. Fix k, and then choose n, Δ, δ, construct (X) with n points. First choose $c_1, \cdots, c_k$ so that $||c_i - c_j||^2 = \Delta^2 - (\frac{n-k}{n})\delta^2$. And now add data points centered at c_i, and each distance $\sqrt{\frac{n-k}{2n}}\delta$ from c_i. And then we prove our algorithm is $\Omega(\log k)$ worse than the optimal clustering in this case. Now we need to prove lemma 4.1 to give a lower bound of $E[\phi']$ by induction. And therefore we get this kind of seeding is no better than $2(\ln k)$-competitive, which shows the analysis is tight.

Apart from the main results shown above, there is also some corollaries in this paper. And

there is a general theorem shows if C is constructed with D^l seeding, then the corresponding potential function $\phi^{[l]}$ satisfies, $E[\phi^{[l]}] \leq 2^{l+2}(\ln k+2)\phi_{OPT}$. In a nutshell, this kind of seeding method performs better in theory and experiments.

- "Hartigan's Method: k-means Clustering without Voronoi" by Telgarsky and Vattani

From this paper, three formulations of the Hartigan's Method's heuristics were given (including the original version). From these formulations, we can get some great properties of Hartigan's method, and we also get the data partition is always quite separated from the induced Voronoi partition. Based on the theoretical analysis and empirical tests, it concludes that Hartigan's method has good optimization performance and good running time, compared to Lloyd's method.
Then I'll introduce some technical results with more details. The main idea of Hartigan's Method is repeatedly pick a point, and determine its optimal cluster assignment by comparing the value of a particular function before and after moving that point to another cluster. Here we have the bias-variance decomposition of k-means cost: $\phi(C, z) = \phi(C) + C||\mu(C) - z||^2$. And denote the cost of merging two clusters A, B by $\Delta(A, B) = \phi(A \cup B) - \phi(A) - \phi(B) = \frac{AB}{A+B}||\mu(A) - \mu(B)||^2$. These two formulas will be used in the following context. To ensure we get to the optima, we need to set a terminal condition.

1. Holistic Formulation
   This is the most natural form. After we move x from its previous cluster S to another cluster T, we have the delta value

   $$\phi(x; S, T) = \phi(S) + \phi(T) - \phi(S\backslash\{x\}) - \phi(T \cup \{x\})$$

   Using the formula in the last paragraph, we have the first termination whether$\Phi(x; S, T) > 0$, where

   $$\Phi(x; S, T) = \frac{T}{T+1}||\mu(T) - x||^2 - \frac{S}{S-1}||\mu(S) - x||^2$$

   Here the most crucial part is how to select a point that could improve most.

2. Point-relative Formulation
   From the termination condition above , we yields

   $$||\mu(C_j) - x_i|| < ||\mu(C_{y_i}) - x_i||\sqrt{\frac{C_{y_i}(C_j + 1)}{C_j(C_{y_j} - 1)}}$$

   So here we can give the second formulation. We define $\alpha(S, T) = \sqrt{S(T + 1)/(T(S - 1))}$, and we will implement the iteration if $\exists i, j$, s.t. $||\mu(C_j)-x_i|| < \alpha(C_y, C_j)||\mu(C_{y_i})-x_i||$. From here it is not hard to prove this method has some good properties: (1) The cost sequence is strictly decreasing (Lloyd has this property too); (2) The resulting partition has no empty clusters; (3) The resulting partition has distinct means. The property(2) and (3) are not satisfied by Lloyd.

3. Cluster-relative Formulation
   The termination condition is there exists no i, j so that $||x_i - v(C_y, C_j)|| > \rho(C_y, C_j)$, where $\rho(S;T) = \alpha||v(S;T) - \mu(S)||$. This is simply the Voronoi partition. And we can deduce the following theorem. Let S, T be two clusters as provided by the termination of H_3. Then every point of $S \cup T$ is at a distance of at least $||\mu(S) - \mu(T)||/(2S + 2T)$ from the Voronoi boundary (hyperplane) between S and T. We have the k-means global optimum must be a (local) optimum of Hartigan's method.

   Therefore we have the Hartigan's method has opportunities to escape the local optima of Lloyd's method. And we can show that Hartigan's method can outperform Lloyd's method by quantifying the gap between the circlonoi partition and the Voronoi partition.

Problem 2

For any instance of Vertex-Cover* $G = (V, E)$, Let $W = \{w_e, e \in E\}$, which is a new set of vertices, each of them is an edge in Vertex-Cover*. Define the distance over $G \cup W$ as below:

- $\rho(u, v) = 1$ if $u, v \in V$ with $(u, v) \in E$;
- $\rho(u, v) = 1$ if $u \in V$ and $v = (u, t) \in E$ for some t;
- $\rho(u, v) = 2$ otherwise.

Lemma 1. *The solution of the Vertex-Cover* problem G is V', with size $|V'| \leq k$ iff $(V \cup W, \rho)$ has a k-center solution of 1.*

Proof. [$\Rightarrow$] There exists a $V' \subset V$ such that $V' \cup (\cup_{v' \in V'} \cup_{e(v',v) \in E} v) = V$, and $|V'| = k$. Consider any $v \in V \cup W$:

- If $v \in V'$, then $\rho(v, V') = 0$;
- If $v \in V - V'$, then there exists some edge $e(u, v) \in E$, and $u \in V'$. By definition we have $\rho(v, V') = \rho(v, u) = 1$;
- If $v \in W$, denote $v = (p, q) \in E$. Then either p or q is in V'. So $\rho(v, V') \leq \min\{\rho(v, p), \rho(v, q)\} = 1$.

So we have $\rho(v, V') \leq 1$ for all $v \in V \cup W$. On the other hand, all the distance is at least 1. Therefore $(V \cup W, \rho)$ has a k-center solution of 1.

[$\Leftarrow$] Let T be a k-center solution of cost 1. If $\exists w_e \in W \cap T$, where $w_e = (u, v)$. Then replace T with $(T - \{w_e\}) \cup \{u\}$. Here we get a new set $T' \subset V$. $|T'| \leq k$ and $\rho(v, T') \leq 1$ for all $v \in V \cup W$. Then I'll prove T' is a vertex-cover of V. For every $w_e = (u, v) \in E$, since $\rho(w_e, T') = 1$, then either $u \in T'$ or $v \in T'$. So T' is a vertex-cover of V, with size $\leq k$. □

By Lemma we know each instance in Vertex-Cover* can be modified in polynomial time into an instance of k-center problem, and vice versa. Using the fact that Vertex-Cover* is NP-hard, we know that k-center problem is NP-hard. #

Problem 3

Balanced k-means problem is an NP-hard problem.
We just need to prove balanced 2-means problem is an NP-hard problem, and then we can naturally know that balanced k-means problem is an NP-hard problem. Here we also use NAE-3SAT* to prove the hardness. Introduce the same distance matrix $D(\phi)$ as def25. From lemma26 we know ϕ is NAE-3SAT* satisfied iff $D(\phi)$ admits to a generalized 2-means cost of $\text{cost}(\phi) = n - 1 + 2\delta m/n$. Also from lemma 27, we know that if any partition contains an element and its negation, then the cost will be greater than $n - 1 + 2\delta m/n$. That means if the partition is not balanced then, the cost will be greater than $n - 1 + 2\delta m/n$. So here we actually loosen lemma 27, and add an condition that partition is balanced. It's easy to get then. After that, we can use the same embedding method to prove balanced 2-means is NP-hard. #

Problem 4

1. I'll construct a $Y \subset X$ that is both an ϵ-cover and an ϵ-packing.

   (a) Select x_1 arbitrarily from X. Consider the closed ball $B_1 := B[x_1, \epsilon] = \{x : \rho(x_1, x) \leq \epsilon\}$.

   (b) Select x_i $(i \geq 2)$ arbitrarily from $X - \cup_{j=1}^{i-1} B_j$. Denote B_i as the closed ball $B_j := B[x_j, \epsilon] = \{x : \rho(x_j, x) \leq \epsilon\}$.

   (c) Repeat step(b) until $\cup_j B_j = X$. Let $Y = \cup_j \{x_j\}$, which is both an ϵ-cover and an ϵ-packing.

   Then I'll prove the correctness.

   Proof. Firstly, I'll show that this algorithm will stop within a finite number of times. Because X is a compact metric space, it is totally bounded. So there exists a finite collection of open balls in X of radius $\epsilon/4$ whose union contains X. Therefore there are finite disjoint open balls $M = \{O_1, \cdots, O_m\}$ of radius $\epsilon/4$ so that their union will cover $Y \subset X$. They are disjoint because the distance of each two points in Y ie greater than ϵ. So $|Y| \leq |M| < \infty$. Therefore this algorithm will stop within finite times.

   Secondly, I'll prove that Y is an ϵ-cover. Because $\cup_j B_j = X$, $\forall x \in X$, $\exists i$ s.t. $x \in B_i$. So we have $x_i \in Y$ and $\rho(x, x_i) \leq \epsilon$.

   Lastly, I'll show that Y is an ϵ-packing. $\forall i < j$, we have $x_j \notin \cup_{j=1}^{i-1} B_j$, that shows $x_j \notin B_i$. That is $\rho(x_i, x_j) > \epsilon$ for $\forall i \neq j$. □

2. For the first part of the inequality, I'll prove by contradiction. Assume that there exists an ϵ-packing $P = \{p_1, \cdots, p_M\}$, and an $\epsilon/2$-cover $C = \{c_1, \cdots, c_N\}$, and $M \geq N + 1$. Then from Pigeonhole principle, there exist p_i and p_j $(i \neq j)$ that belong to the same closed ball $B[c_k, \epsilon]$. So $\rho(c_k, p_i) \leq \epsilon/2$ and $\rho(c_k, p_j) \leq \epsilon/2$. From the triangle inequality of metric space, we have

   $$\rho(p_i, p_j) \leq \rho(p_i, c_k) + \rho(p_j, c_k) \leq \epsilon$$

   This leads to a contradiction because $\rho(p_i, p_j) > \epsilon$ for an ϵ-packing. Hence the size of any ϵ-packing is no larger than the size of any $\epsilon/2$-cover. So $P_\epsilon(X) \leq N_{\epsilon/2}(X)$.

   For the second part of the inequality, denote $P = \{p_1, \cdots, p_M\}$ is a max $\epsilon/2$-packing. Then for any $x \in X - P$, $\exists 1 \leq i \leq M$, so that $\rho(x_i, x) \leq \epsilon/2$. (Otherwise, $P \cup \{x\}$ is a $\epsilon/2$-packing, whose size is larger than the size of P. This will lead to a contradiction of our maximal assumption.) So we have $x \in B(x_i, \epsilon/2)$. Therefore P is a $\epsilon/2$-cover. So the size of the smallest $\epsilon/2$-cover is no larger than the size of p_M. So $N_{\epsilon/2}(X) \leq P_{\epsilon/2}(X)$.

   Therefore we conclude that $P_\epsilon(X) \leq N_{\epsilon/2}(X) \leq P_{\epsilon/2}(x)\#$

3. First we estimate the lower bound of $N_\epsilon(B^d(0,1))$.
Consider an ϵ-cover of $B^d(0,1)$ of size $N = N_\epsilon(B^d(0,1))$: $C = \{c_1, \cdots, c_N\}$. Then we have $B^d(0,1) \subset \bigcup_{i=1}^N c_i$, so

$$\text{vol}(B^d(0,1)) \leq N\text{vol}(B^d(0,\epsilon)) = N\epsilon^d\text{vol}(B^d(0,1))$$

So $N \geq 1/\epsilon^d$.

Next let's estimate the upper bound.
For any ϵ-packing of $B^d(0,1)$ of size $M = P_\epsilon(B^d(0,1))$: $P = \{p_1, \cdots, p_M\}$. Then we consider the balls $B^d(p_i, \epsilon/2)$. These balls are disjoint because P is an ϵ-packing. Besides, all balls lie in $B^d(0, 1+\epsilon/2)$. Therefore, $\bigcup_{j=1}^M B^d(p_i, \epsilon/2) \subset B^d(0, 1+\epsilon/2)$.

$$M\text{vol}(B^d(p_i, \epsilon/2)) \leq \text{vol}(B^d(0, 1+\epsilon/2))$$

$$M(\frac{\epsilon}{2})^d\text{vol}(B^d(0,1)) \leq (1+\epsilon/2)^d\text{vol}(B^d(0,1))$$

Hence we have $M \leq (\frac{1+\epsilon/2}{\epsilon/2})^d = (\frac{\epsilon+2}{\epsilon})^d$. So $N_\epsilon(B^d(0,1)) \leq M \leq (\frac{\epsilon+2}{\epsilon})^d$.

In conclusion, we have $(\frac{1}{\epsilon})^d \leq N_\epsilon(B^d(0,1)) \leq (\frac{\epsilon+2}{\epsilon})^d$. #

4. Consider the singular value decomposition of $A = U\Sigma V^*$, Σ is a $m \times n$ matrix. $\Sigma = (\text{diag}(\lambda_1, \cdots, \lambda_m))$. Here $\{\lambda_i\}$ are the singular values of A, in a decreasing order of their absolute value.

$$\sigma_{\max}(A) = \max_{x \in \mathcal{R}^n, ||x||=1} ||Ax|| = \max_{x \in \mathcal{R}^n, ||x||=1} ||U\Sigma V^* x|| = \max_{x \in \mathcal{R}^n, ||x||=1} ||\Sigma x|| = |\lambda_1|$$

Then for any ϵ-cover C_0 of S^{n-1}, we have V^*C_0 is also an ϵ-cover of S^{n-1}. So we need to find the difference between $|\lambda_1|$ and $\max_{x \in C} ||U\Sigma x|| = \max_{x \in C} ||\Sigma x||$. Let $x_0 = \arg\min_{x \in S^{n-1}} ||\sigma x|| = (1, 0, \cdots, 0)$. Because C is an ϵ-cover, there exists some $x = (x_1, \cdots, x_n) \in C$ so that $||x - x_0||_2^2 \leq \epsilon^2$. So we have

$$(1-x_1)^2 + \sum_{i=2}^n x_i^2 \leq \epsilon^2$$

$$\sum_{i=1}^n {x_i}^2 = 1$$

So we get $x_1 \geq 1 - \epsilon^2/2$. Therefore $||\Sigma x||_2^2 = \sum_{i=1}^m \lambda_i^2 x_i^2 \geq |\lambda_1|^2(1-\epsilon^2/2)^2$.
So $\sigma_{\max} \leq \frac{\sigma_C}{1-\epsilon^2/2}$. Therefore the tight upperbound of $\sigma_{\max}$ is $\frac{\sigma_C}{1-\epsilon^2/2}$. If $m < n$, it can be achieved when $x_1 = 1 - \epsilon^2/2$, $x_{m+1} = \sqrt{1-(1-{x_1}^2)}$, $x_i = 0$ otherwise. If $m \geq n$, then it cannot be achieved, so it is just a loose tight upperbound.
Plus it is natural that $\sigma_C \leq \sigma_{\max}$, which is a tighter lower-bound of $\sigma_{\max}$. #

Problem 5

1. Let $|V| = n$, $|E| = m$. d_S is the sub-matrix only has rows of edges in S, so its dimension is $|S| \times n$. Let $e = (i, j) \in S$ is an edge. Then the e^{th} row of d_S is all zero expects that i_{th} and j_{th} columns. There one of the entries is 1, and the other is -1. So d_S has differential operator equal to the graph $G_S = (V, S)$, that is the graph eliminating the edges in $E \backslash S$ from the original graph G.

2. I'll show each entries of L is correct for $L = \sum_{i=1}^{k} L_i$ by two cases: entries in diagonal, and others. Denote $L = (l_{pq})_{n\times n}$, $L_i = (l_{i_{pq}})_{n\times n}$. By definition we have $L = D - A$, where D is the degree matrix and A is the adjacency matrix. So for $1 \le p, q \le n$, we have

$$l_{pq} = \begin{cases} \deg(v_p) & p = q \\ -1 & p \neq q \text{ and} (v_p, v_q) \in E \\ 0 & o.w. \end{cases} \tag{1}$$

   (a) For entries l_{pp} in diagonal, $l_{pp} = \deg(v_p)$, that is the number of edges that connect v_p. Because $E_1, \cdots, E_k$ is a partition of E, $E = \bigcup_{i=1}^{k} E_k$ is a disjoint union of all the edges. So each edge connecting v_p will exist and only exist in some E_i. The number of edges connect v_p is equal to the sum of the number of edges connect v_p in each E_i, that is $l_{i_{pp}}$. So $\sum_{i=1}^{k} l_{i_{pp}} = \sum_{i=1}^{k} \deg_i(v_p) = \#\text{edges connect } v_p = \deg(v_p) = l_{pp}$.

   (b) For entries l_{pq} with $p \neq q$. If there is no edge of (v_p, v_q), then $l_{pq} = 0$. On the other hand, there is naturally no edge of (v_p, v_q) in any E_i. (Because E_i is a subset of E.) Therefore $\forall 1 \le i \le n$, $l_{i_{pq}} = 0$. So we have $l_{pq} = 0 = \sum_{i=1}^{n} l_{i_{pq}}$ in this situation.

   Besides, when there exists an edge $e = (v_p, v_q) \in E$, then we have $l_{pq} = -1$ by definition. Because $\{E_i\}_{i=1}^{k}$ is a disjoint union of all edges, there will exist and only exist one t so that $e = (v_p, v_q) \in E_t$. So we have $l_{t_{pq}} = -1$, and $l_{i_{pq}} = 0$ for $i \neq t$. Hence $\sum_{i=1}^{k} l_{t_{pq}} = -1 + 0 + \cdots + 0 = -1 = l_{pq}$.

   Therefore we conclude that $L = \sum_{i=1}^{k} L_i$. #

3. The graph $\mathcal{G} = (V \cup V', E \cup E')$ is the disjoint union of G and G'. Consider $G \cup v(G') := (V_G \cup V_{G'}, E_G)$ and $G' \cup v(G) := (V_G \cup V_{G'}, E_{G'})$. Then we have,

$$L_{G\cup v(G')} = \begin{bmatrix} L_G & 0 \\ 0 & 0 \end{bmatrix}$$

$$L_{G'\cup v(G)} = \begin{bmatrix} 0 & 0 \\ 0 & L_{G'} \end{bmatrix}$$

   This is natural because we only add isolated vertex to the graph, without doing anything on edges. Then we have $\mathcal{G} = (V \cup V', E \cup E') = (L_{G\cup v(G')}) \cup (L_{G'\cup v(G)})$

Here E and E' are the disjoint, from question 5.2 we have:

$$L_{\mathcal{G}} = L_{L_{G\cup v(G')}} + L_{L_{G'\cup v(G)}} = \begin{bmatrix} L_G & 0 \\ 0 & L_{G'} \end{bmatrix}$$

Here the first $|V|$ column/row represent vertices in G, and the rest $|V'|$ column/row represent vertices in G'.

4. For the p_{th} row of L, consider each entries l_{pq}. If $p = q$, $l_{pq} = \deg(v_p)$. If $p \neq q$, $l_{pq} = -1$ when $(v_p, v_q) \in E$, else 0. So $\sum_{1\leq q\leq n, q\neq p} l_{pq} = (-1) * \#$ edges connect $v_p = -\deg(v_p)$. Hence $\sum_{q=1}^{n} l_{pq} = \deg(v_p) - \deg(v_p) = 0$. Denote $a = (1, 1, \cdots, 1)^T$. For each row L_p, we have $L_p a = \sum_{q=1}^{n} l_{pq} = 0$. So $La = 0$.

   Here we got the vector $a \in \ker(L)$. So $\dim(\ker(L)) \geq 1$. By Rank nullity theorem, we have $\text{rank}(L) \leq n - 1$. Next I'll show it's a tight upper bound.

   Construct a graph $G = (V, E)$, $V = \{v_1, \cdots, v_n\}$, $E = \{(x_1, x_j) | 2 \leq j \leq n\}$. There are n edges and n vertices in total. It's Laplacian is:

$$L = \begin{bmatrix} n-1 & -1 & -1 & \cdots & -1 \\ -1 & 1 & 0 & \cdots & 0 \\ -1 & 0 & 1 & \cdots & 0 \\ \cdots & & & & \\ -1 & 0 & 0 & \cdots & 1 \end{bmatrix}$$

   Assume $x = (x_1, \cdots, x_n)^T \in \ker(L)$. $Lx = 0$. Then we get the following equations:

$$(n-1)x_1 - x_2 - \cdots - x_n = 0$$

$$-x_1 + x_2 = 0$$

$$-x_1 + x_3 = 0$$

$$\cdots$$

$$-x_1 + x_n = 0$$

   The solution is $x_1 = x_2 = \cdots = x_n = c$. So the solution space is spanned by the vector $\mathbf{1}$, which means $\dim(\ker(L)) = 1$. Then by rank nullity theorem, we have $\text{rank}(L) = n - 1$, which achieves my bound.

   In conclusion, the tight upper bound for $\text{rand}(L)$ is $n - 1$. $\ker(L)$ always contains $(c, c, \cdots, c)^T$, where c is a constant in $\mathbb{R}$. #

Problem 6

1. $1 \Rightarrow 2$ Let $G = (V, E)$. Now we add the edge $e = (i, j)$ to G, $e \notin E$. Because G is connected , there exists a simple path from i to j in G. Suppose $e_1, \cdots, e_n$ are the edge sequence defining this path. Then after the new edge e is added, $e_1, \cdots, e_n, e$ define a path from i to itself, that means it will form a circuit, so we have G is maximally acyclic.

   $2 \Rightarrow 1$ Because G is connected and maximally acyclic, G is naturally a tree.

   $1 \Rightarrow 3$ We'll use induction to prove this part.
   For $|V| = 2$, it is natual that $|E| = 1, |E| = |V| - 1$.
   Suppose we have proved that for a tree $G = (V, E)$ with k nodes, $|E| = |V| - 1 = k - 1$. Then given a tree G with $k + 1$ nodes, we can first remove a leaf node and the edge connected to it, so that we have a new graph $G' = (V', E')$. G' is also connected and acyclic, so G' is a tree with k nodes. By the assumption of the induction, $|E'| = |V'| - 1 = k - 1$. V' is the union of V and the removed edge, so we have $|V| = |V'| + 1 = k = |G| - 1$.
   So by induction, we proved the conclusion.

   $3 \Rightarrow 1$ For $G = (V, E)$, $|E| = |V| - 1$. Because G is connected graph, we only need to show it is acyclic. Assume that G has a cycle. Then we can remove an edge of that cycle without affecting the connectivity of it. After removing all that edges from all cycles, we obtain a new graph $G' = (V', E')$, which is a spanning tree of G. So we have $|E| > |E'| = |V'| - 1 = |V| - 1$ This contradicts the original claim that $|E| = |V| - 1$, so there is no cycle in G. That is, G is a tree. #

2. [$\Rightarrow$] For $G = (V, E)$, its derivative d is full rank. If there exists a circle in the graph, which has t edges and t vertices. Here $t \leq \min(m, n)$. Denote the edge set as $T = \{e_1, e_2, \cdots, e_t\}$. Consider the submatrix of d with all the nodes and only the edges in T, denote it as d_T, whose dimension is $t \times n$. Because for each nodes in the cycle, it has exactly one "in" and one "out" in this cycle, that is for the column representing these nodes, there is only one "1" and one "-1", whose sum is 0. For other columns, because there is no edges connecting them, the corresponding column in d_T is all zero. Then $\sum_{i=1}^{T} = 0$ because for each column, the sum of the elements is zero. Therefore the rank of $d_T \leq t - 1$. So one row of d can be the linear combination of the some other rows. That means, d is not row full rank.

   Also for d's each column vectors $\{u_1, \cdots, u_n\}$. Because for each row $(x_1, \cdots, x_n)$, there is exactly one "1" and one "-1". So $\sum_{i=1}^{n} u_i = \mathbf{0}$. That means d is not column full rank too. Therefore, d is not full rank, which causes a contradiction. So we have G has no cycles.

[⇐] Suppose a graph has no cycles, then it is a tree or some connected components (forest). If it contains several connected components, we can order the nodes and edges according to it, and finally get a block diagonal matrix. In order to show it's full rank we need to prove each block of it is full rank. So it's suffice to show the derivative of a tree $G = (V, E)$ is full rank.

Here we can use the conclusion that we'll prove in question 6.3. That is for a tree, after removing the derivative matrix's first column, the remain matrix $\hat{d}$ holds $det(\hat{d}) = \pm 1$. That means $\hat{d}$ is full rank, and so the row vectors of $\hat{d}$ are linearly independent. But if d is not full rank. Because it's a tree, we have $|E| = |V| - 1$. That means the row vectors $\{v_i\}_{i=1}^m$ are linearly dependent. So there exists $\{\lambda_i\}_{i=1}^m$ not all equal to zero, satisfying $\sum_{i=1}^m \lambda_i v_i = \mathbf{0}$. But here since $\hat{d}$ is removing a column from d, we will have for row columns $\{\hat{v_i}\}$ of $\hat{d}$, there is $\sum_{i=1}^m \lambda_i \hat{v_i} = \mathbf{0}$. This causes a contradiction to linear independence. So we draw the conclusion that d is full rank, which further prove that if G has no cycles, d is full rank.

3. Firstly I will use induction prove that

$$det(\hat{d}) = \pm 1 \tag{2}$$

   (a) Base: Consider a tree with 2 nodes. By question 6(1), there is only one edge. After removing the first column, we have $\hat{d} = (\pm 1)$, which shows the conclusion is right for $|V| = 2$.

   (b) Inductive Assumption: Assume for any tree G with n nodes, the derivative is d_{n-1}. We have $det(\hat{d_{n-1}}) = \pm 1$

   (c) Inductive Step: Consider a tree G with n nodes, the derivative is d. Suppose that the edge $e_j = (1, k)$ connects the first node and the i_{th} node. So after we remove the first column of d, the only non-zero entry of $\hat{d}$ is the k_{th} element, which is 1 or -1. So let's compute the determinate of $\hat{d}$ by expanding along the j_{th} row. Denote $\hat{d'}$ as the remain matrix after removing the j_{th} row and $(k-1)_{th}$ column from $\hat{d}$. Then we have

$$det(\hat{d}) = \hat{d}_{j,k-1} det(\hat{d'}) = \pm det(\hat{d'})$$

   Next, I'll show $\hat{d'}$ is the matrix after removing one column from the derivative of a tree with $n-1$ nodes. Suppose the graph constructed as below: Remove the first node and the edges connecting it. Then connect all the neighbours of the first node (except the k_{th} node) with the k_{th} node. Here we get a graph with $n-1$ nodes. This graph $G' = (V', E')$ is a tree because $|E'| = |E| - n + (n-1) = |E| - 1 = |V| - 2 = |V'| - 1$.

   Then let the k_{th} node to be the first column of the new derivative matrix, and keep the order of other nodes and edges the same as before. Then after removing the first column of the derivative, we will get a matrix exactly same as $\hat{d'}$.

   By induction assumption, we have for graph G', $det(\hat{d'}) = \pm 1$.

   So we have $det(\hat{d}) = \pm det(\hat{d'}) = \pm 1$.

   (d) Therefore we can conclude that $det(\hat{d'}) = 1$.

Finally we have

$$det(\hat{d}^T\hat{d}) = det(\hat{d}^T)det(\hat{d}) = (det(\hat{d}))^2 = 1$$

4. For a connected graph $G = (V, E)$, $|V| = n$, $|E| = m$.Denote by $\hat{d}$ the matrix monior of d produced by removing its first column. Let $S \subset E$ be a collection of edges, denote d_S as the submatrix of d where only contains the edges in S. Denote the Laplacian matrix of G as L, the derivative as d. $d = (a_1, \hat{d})$. From the conclusion of HW0, we have

$$L = d^T d = (a_1\hat{d})^T(a_1\hat{d}) = \begin{bmatrix} a_1^T a_1 & a_1^T\hat{d} \\ \hat{d}^T a_1 & \hat{d}^T\hat{d} \end{bmatrix}$$

So we have $L_{1,1} = \hat{d}^T\hat{d}$. And we need to calculate the determinant of $\hat{d}^T\hat{d}$. Let $\mathcal{S} = \{S \subset E||S| = n-1\}$. Then for each $S \in \mathcal{S}$, $\hat{d_S}$ is a $(n-1)\times(n-1)$ matrix. From Cauchy-Binet Formula, we have:

$$L_{1,1} = det(\hat{d}^T\hat{d}) = \sum_{S\in\mathcal{S}} det(\hat{d_S}^T)det(\hat{d_S})$$

Let $G_S = (V, S)$. Then we will have the following scenarios.

(a) If G_S is a tree. From question 6.3, we have $det(\hat{d_S}) = \pm 1$. Therefore we have $det(\hat{d_S}^T)det(\hat{d_S}) = 1$.

(b) If G_S is connected, but there exists a cycle in G_S, we can remove edges from S until it becomes a tree $G'_S = (V, S')$. Assume we have totally removed $k \geq 1$ edges. Then we have $|S| = n - 1 = |V| - 1 = |S'| = |S| - k$, which causes a contradiction. So this scenario can never happen.

(c) If G_S is not connected. Then we can see it as k connected component $G_i = (V_i, S_i)$, where $1 \leq i \leq k$. If all components are trees. Denote $|V_i| = n_i$, $|S_i| = m_i$. So we have $n - 1 = \sum_{i=1}^{n} n_i - 1 = \sum_{i=1}^{k}(m_i + 1) - 1 = \sum_{i=1}^{k} m_i + p - 1 = m + p - 1 = n+p-2$. So we have $p = 1$, which leads to the contraction to G_S is not connected. Hence, there exists a cycle, assume G_1 is a cycle. Using question 6.2, d_1 is not full rank. So $det(\hat{d_1}) = 0$. Therefore $det(\hat{d_S}) = 0$. That means $det(\hat{d_S}^T)det(\hat{d_S}) = 0$.

From all above, we can conclude that $L_{1,1}$ is the number of the spanning trees of G. #

Problem 7

1.

Lemma 2. *Let $x \in \mathbb{R}^n$, $x = (x_1, \cdots, x_n)^T$. L is the Laplacian matrix of graph $G = (V, E)$. Then $x^T L x = \sum_{(v_i,v_j)\in E}(x_i - x_j)^2$.*

Proof. By definition of Laplacian $L = D - A$, we have

$$\begin{aligned} x^T L x &= x^T D x - x^T A x \\ &= (x_1, \cdots, x_n)\text{diag}(\deg(v_1), \cdots, \deg(v_n))(x_1, \cdots, x_n)^T \\ &- (x_1, \cdots, x_n)A(x_1, \cdots, x_n)^T \\ &= \sum_{i=1}^{n} \deg(v_i)x_i^2 - (x_1, \cdots, x_n)A(x_1, \cdots, x_n)^T \end{aligned}$$

For each entries a_{ij} in A, $a_{ij} = 1$ if there exists an edge $(v_i, v_j) \in E$.
So $(x_1, \cdots, x_n)A(x_1, \cdots, x_n)^T = \sum_{(v_i,v_j)\in E} 2x_i x_j$.

$$x^T L x = \sum_{i=1}^{n} \deg(v_i)x_i^2 - 2\sum_{(v_i,v_j)\in E} x_i x_j = \sum_{(v_i,v_j)\in E}(x_i - x_j)^2$$

□

Lemma 3. *If G is a connected graph, the algebraic multiplicity of eigenvalue 0 of its Laplacian is 1.*

Proof. Suppose the algebraic multiplicity of eigenvalue 0 is greater than 1. Then there exists two linearly independence eigenvectors x_1, x_2 with eigenvalue 0. We have $Lx_1 = 0$, $Lx_2 = 0$. From Lemma 2, we have $\sum_{(i,j)\in E}(x_{1_i} - x_{1_j})^2 = x_1^T L x_1 = 0$, and $\sum_{(i,j)\in E}(x_{2_i} - x_{2_j})^2 = x_2^T L x_2 = 0$. So for $\forall (i,j) \in E$, $x_{1_i} = x_{1_j}$ and $x_{2_i} = x_{2_j}$. Because G is connected, there is path between any pair of vertices. Therefore $x_{1_i} = c_1, x_{2_i} = c_2$, for $1 \le i \le n$. That means x_1, x_2 are both in span$\{\mathbf{1}\}$, which causes a contradiction to the independence of x_1 and x_2. □

Here we have a graph G of k connected components. Then the adjacency matrix can be block diagonal with k blocks. Since the degree matrix is naturally a diagonal matrix, the Laplacian $L = D - A$ can also be block diagonal with k blocks $L_1, \cdots, L_k$. By Lemma 3, each block matrix has eigenvalue 0 with multiplicity one. Then $det(\lambda I - L) = det(\lambda I - L_1)\cdots(\lambda I - L_k)$. Hence the algebraic multiplicity of eigenvalue 0 in L is the sum of algebraic multiplicity of eigenvalue 0 in each L_i, that is equal to k. Therefore, the number of nonzero eigenvalues of L is precisely $n - k$. #

2. For embedding $f \in \mathbb{R}^V$, so that $f(i) = x_i$, for $i \in V$. We want to minimize the energy

$$\mathcal{E} = \sum_{e\in E} l_e^2 = \sum_{(i,j)\in E}(x_i - x_j)^2$$

In the meantime we have the constraint that the moment of inertia is 1, that is,

$$I = \sum_{i \in V} x_i^2 = 1$$

Let $x = (x_1, \cdots, x_n)^T$. Then our constraint becomes $x^T x = 1$. Then for the energy function,

$$\begin{aligned}
\mathcal{E} &= \sum_{(i,j)\in E} (x_i - x_j)^2 \\
&= \sum_{(i,j)\in E} (x_i^2 + x_j^2 - 2x_i x_j) \\
&= \sum_{(i,j)\in E} x_i^2 + \sum_{(i,j)\in E} x_j^2 - 2 \sum_{(i,j)\in E} x_i x_j \\
&= \sum_{i \in V} deg(x_i) x_i^2 + \sum_{(i,j)\in E, i<j} (-1) * x_i x_j + \sum_{(i,j)\in E, i>j} (-1) * x_i x_j \\
&= x^T L x
\end{aligned}$$

Here L is the Laplacian, and it holds because

$$l_{pq} = \begin{cases} \deg(v_p) & p = q \\ -1 & p < q \text{ and} (v_p, v_q) \in E \\ -1 & p > q \text{ and} (v_p, v_q) \in E \\ 0 & o.w. \end{cases}$$

Then to ensure the center of masses is equal to 0. Introduce the centering matrix $H = I - \frac{1}{n}\mathbf{1}\mathbf{1}^T$. Because $\sum_{(i,j)\in E}(x_i - x_j)^2 = \sum_{(i,j)\in E}[(x_i - \bar{x}) - (x_j - \bar{x})]^2$, so we can transfer the original question to:

$$\min_x \quad x^T HLHx$$

$$s.t. \quad x^T Hx = 1$$

Let $y = Hx$. Because $H\mathbf{1} = 0$, $HH = H(I - \frac{1}{n}\mathbf{1}\mathbf{1}^T) = H$. Then $y^T y = x^T HHy = x^T Hx = 1$. And $x^T HLHx = y^T Ly$.

So we transfer our optimization problem to:

$$\min_y \quad y^T Ly$$

$$s.t. \quad y^T y = 1$$

Because $L = d^T d$, where d is the derivative of the graph, $y^T Ly = y^T d^T dy = (dy)^T (dy) = ||dy||_2^2$. Also $y^T y = ||y||_2^2$. Hence we transfer the optimization to:

$$\min_y ||dy||_2^2$$

$$s.t. \quad ||y||_2^2 = 1$$

This is a problem to solve d's minimal singular value and its corresponding singular vectors. But now we notice that $\mathbf{1}y = \mathbf{1}Hx = 0$, so we need y to be vertical to $\mathbf{1}$. Because L has a eigenvalue 0, so d has a singular value 0, which is the minimal singular value for d. However $\ker(L) = span(\mathbf{1})$, so the singular vector corresponding to 0 is $\mathbf{1}$, which is not vertical to $\mathbf{1}$. Therefore, we cannot use the minimal singular value 0. We consider the singular values of d, by the definition of singular values, they are the square root of eigenvalues of L. By question 7.1 we know the algebraic multiplicity of eigenvalue 0 of L is 1. So the algebraic multiplicity of the singular value 0 of d is 1. Therefore the solution is the minimal non-zero singular value of d, and y is equal to its corresponding singular vectors. This is correct because the singular vectors are orthogonal.

In a nutshell, we find the tight lower bound is the minimal non-zero singular value of d, and the embedding f achieves this bound is to map $\{i\}$ to the singular vector corresponding to the minimal non-zero singular value of d.