Homework 4

Algorithms for Big Data
CS498ABD Spring 2019
Due: 10am, Friday, April 19th

Instructions:

- Unlike previous homeworks, you need only do **3 out of the 4** problems. (Of course you're encouraged to try and welcome to submit all 4!)
- Each home work can be done in a group of size at most two. Only one home work needs to be submitted per group. However, we recommend that each of you think about the problems on your own first.
- Homework needs to be submitted in pdf format on Gradescope. See `https://courses.engr.illinois.edu/cs374/fa2018/hw-policies.html` for more detailed instructions on Gradescope submissions.
- Follow academic integrity policies as laid out in student code. You can consult sources but cite all of them including discussions with other class mates. Write in your own words. See the site mentioned in the preceding item for more detailed policies.

Problem 1. Fast JL Recall the JL Lemma where we pick a random $m \times n$ matrix Π and show that for $m = O(1/\epsilon^2)$, with at least $2/3$ probability,

$$(1-\epsilon)\|x\|_2^2 \leq \|\Pi x\|_2^2 \leq (1+\epsilon)\|x\|_2^2. \tag{1}$$

- Imagine picking Π as follows: for each $i \in \{1, \ldots, n\}$ we pick a uniformly random number $h_i \in \{1, \ldots, m\}$. We then set $\Pi_{h_i,i} = \pm 1$ for each $i \in \{1, \ldots, n\}$ (the sign is chosen uniformly at random from $\{-1, 1\}$), and all other entries of Π are set to 0. This Π has the advantage that in turnstile streams, we can process updates in constant time. Show that using this Π still satisfies the conditions of Equation 1 with $2/3$ probability for $m = O(1/\epsilon^2)$.
- Show that the matrix Π from the first part can be specified using $O(\log n)$ bits such that Equation 1 still holds with at least $2/3$ probability, and so that given any $i \in \{1, \ldots, n\}$, $\Pi_{h_i,i}$ and h_i can both be calculated in constant time. *Hint:* Use limited independence hash functions to generate the h_i.

Solution (sketch). Let $\sigma_i \in \{-1, 1\}$ denote the sign of the ith coordinate. For $i, j \in [n]$, let

$$H_{ij} = \begin{cases} 1 & \text{if } h(i) = h(j) \\ 0 & \text{otherwise.} \end{cases}$$

We have

$$\|\Pi x\|^2 = \sum_{i=1}^{n} \sum_{j=1}^{n} \sigma_i \sigma_j H_{ij} x_i x_j = \sum_{i=1}^{n} x_i^2 + 2 \sum_{i<j} \sigma_i \sigma_j H_{ij} x_i x_j.$$

We have

$$\mathrm{E}\big[\|\Pi x\|^2\big] = \sum_{i=1}^{n} x_i^2 = \|x\|^2 \text{ since } \mathrm{E}[\sigma_i \sigma_j] = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j. \end{cases}$$

Now,

$$\begin{aligned}
\mathrm{E}\big[\|\Pi x\|^4\big] &= \|x\|^4 + 2\|x\|^2 \,\mathrm{E}\left[\sum_{i<j} \sigma_i \sigma_j H_{ij} x_i x_j\right] + \mathrm{E}\left[\left(2 \sum_{i<j} \sigma_i \sigma_j H_{ij} x_i^2 x_j^2\right)^2\right] \\
&= \|x\|^4 + 4 \sum_{i_1<j_1} \sum_{i_2<j_2} \mathrm{E}\big[\sigma_{i_1} \sigma_{i_2} \sigma_{j_1} \sigma_{j_2} H_{i_1 j_1} H_{i_2 j_2} x_{i_1}^2 x_{i_2}^2 x_{j_1}^2 x_{j_2}^2\big] \\
&\overset{(a)}{=} \|x\|^4 + \mathrm{E}\left[4 \sum_{i<j} H_{ij} x_i^2 x_j^2\right] \\
&= \|x\|^4 + \frac{3}{m} \sum_{i<j} x_i^2 x_j^2 \\
&\leq \|x\|^4 + \frac{2}{m} \left(\sum_i x_i^2\right)^2 \\
&= \left(1 + \frac{2}{m}\right) \|x\|^4,
\end{aligned}$$

where (a) uses the fact that the hash function is (at least) 4-wise independent. Thus $\mathrm{Var}\big[\|\Pi x\|^2\big] = \mathrm{E}\big[\|\Pi x\|^4\big] - \mathrm{E}\big[\|\Pi x\|^2\big]^2 = \frac{2}{m}\|x\|^4$. One now applies Chebyshev to get the desired bound.

For the second part, we need 4-wise independence for the variance calculation: namely, for expectation of terms of the form $\sigma_i \sigma_j \sigma_k \sigma_\ell H_{ij} H_{ik}$. Only $O(\log n)$ bits are needed per 4-wise independent hash function, and $O(m) = O\big(1/\epsilon^2\big)$ such hash functions are needed.

Exercise 2: Improved net argument for subspace embeddings Recall that in oblivious subspace embeddings we want to preserve lengths of all vectors in a subspace of dimension d (assuming vectors are in dimenstion R^n where $n > d$). For this we showed that a JL matrix

with $m = O(d/\epsilon^2)$ rows suffices via a net argument. More formally the claim is that there is a fixed set Q of $\exp(O(d))$ vectors such that preserving their lengths to a $(1 \pm \epsilon)$ factor suffices to preserve lengths of all vectors in that subspace (we then use a union bound). In lecture we describe a construction that yielded a net of size $\exp(d \log d)$ which is weaker. In this problem you will see the stronger bound via the following two parts.

- Define $Q_\gamma = \{w : w \in \frac{\gamma}{\sqrt{d}}\mathbb{Z}^d, \|w\|_2 \leq 1\}$ for $\gamma \in (0, 1)$. Prove $|Q_\gamma| \leq e^{d \cdot f(\gamma)}$ for some function $f(\gamma)$ (which needn't be optimized).

  Hint: Given $z \in Q_\gamma$ define a cube C_z centered at z with side length $\gamma/\sqrt{d}$. Note these cubes are all disjoint, then use a volume argument (you may use that an ℓ_2 ball of radius r in $\mathbb{R}^d$ has volume $(C_d \cdot r/\sqrt{d})^d$ for some constant C_d which is $\Theta(1)$ as d grows).

- Show that if for some $A \in \mathbb{R}^{d \times d}$ we have $|u^T A v| \leq \epsilon$ for all $u, v \in Q_\gamma$, then $|x^T A x| \leq \epsilon/(1-\gamma)^2$ for all $x \in \mathbb{R}^d$ of unit ℓ_2 norm.

  Hint: Write $y = (1-\gamma)x$ and round down the coordinates of y to obtain $z \in Q_\gamma$. Argue that $y \in C_z$ and use that any point in a convex polytope can be written as a convex combination of the vertices of that polytope.

- Finish up the details to argue that JL matrix with $m = O(d/\epsilon^2)$ rows is yields an oblivious subspace embedding with constant probability.

Solution (sketch). For each $z \in Q_\gamma$, let C_z be centered at z with side length $\gamma/\sqrt{d}$. That is, $C_z = \left\{z + \theta : \|z\|_\infty \leq \gamma/2\sqrt{d}\right\}$. These cubes are disjoint, and every point $z + \theta \in C_z$ has length at most

$$\|z \pm \theta\| \leq \|z\| + \|\theta\| \leq 1 + \sqrt{d}\|\theta\|_\infty \leq 1 + \gamma/2.$$

That is, all the cubes pack into a ball with radius $r = 1 + \gamma/2$. Each cube has volume $\left(\frac{\gamma}{\sqrt{d}}\right)^d$, and the ball has radius $1 + \gamma/2$ has volume at most $\left(\frac{c(1+\gamma/2)}{\sqrt{d}}\right)^d \leq \left(\frac{2c}{\sqrt{d}}\right)^d$ for some constant c. Thus the total number of cubes is

$$|Q_\gamma| \leq \left(\frac{2c}{\sqrt{d}}\right)^d \left(\frac{\sqrt{d}}{\gamma}\right)^d = \left(\frac{2c}{\gamma}\right)^d = e^{d \log(2c/\gamma)}.$$

Let $y = (1 - \gamma x)$. We first want to show that y is in the convex hull of Q_γ. Let z round each coordinate of y towards zero to a multiple of $\gamma/\sqrt{d}$. Consider the cube C centered at z with side length $2\gamma/\sqrt{d}$; i.e.,

$$C + \left\{z + \theta : \|\theta\|_\infty \leq \frac{\gamma}{\sqrt{d}}\right\}.$$

Every point $z + \theta \in C$ has length

$$\|z + \theta\| \leq \|z\| + \|\theta\| \leq 1 - \gamma + \sqrt{d}\|\theta\|_\infty = 1;$$

in particular, all the corners of Z are in our net Q_γ. Let $C' = C \cap Q_\gamma$ be the corners of our cube, and write y as a convex combination

$$\sum_{u \in C'} \alpha_u u \text{ where } \alpha_u \geq 0 \text{ for all } u \text{ and } \sum_{u \in C'} \alpha_u = 1.$$

Now

$$|\langle y, Ay \rangle| = \left| \left\langle \sum_u \alpha_u u, A\left(\sum_u \alpha_u u \right) \right\rangle \right| \leq \epsilon \sum_{u,v \in C'} \alpha_u \langle u, Av \rangle = \epsilon \sum_{u,v \in C'} \alpha_u \alpha_v = \epsilon \left(\sum_u \alpha_u \right)^2 = \epsilon.$$

The last part now follows from taking $A = \pi^T \pi - I$. We know from the previous homework that π reserves a particular dot product with up to $\pm\epsilon$ additive error, and we can take the union bound over Q_γ. Thus A satisfies the second property.

In general, we are not necessarily embedding the first d coordinates, but some hidden subspace. But we can rotate the entire space as a thought experiment so that our subspace becomes the first d coordinates. Our JL matrix, which has a Gaussian for every entry, is rotationally invariant.

Exercise 3: LSH for Hamming Distance In class we saw an LSH scheme for nearest neighbor search for n binary strings of length d in the hamming distance metric. The scheme was based on a decision version where for a given r the data structure would be able to answer with good probability whether there is a point in the data base with distance at most r from q or whether every point is at least $(1+\epsilon)r$. The final data structure is composed of $O(\log d/\epsilon)$ data structures for different values of r. Do we need this reduction to the decision version? Read Charikar's paper on similarity search for a variant of the basic scheme that avoids this in a simple way. Describe and analyze the scheme in your own words.

Problem 4. Matchings with additional constraint We saw an algorithm in the semi-streaming model for finding a constant factor approximation to the maximum cardinality and maximum weight matching problem. Now consider the following variant. We are given a graph $G = (V, E)$. Moreover each edge has a color from $\{1, 2, ..., k\}$ and each color i has an integer upper bound b_i. The goal is to find a maximum cardinality matching M which satisfies the additional constraint that the number of edges in M from a color class i is at most b_i. Assume that you are given the b_i values ahead of time and the each edge when it arrives in the stream specifies its end points and its color. Describe a constant factor approximation for this problem in the semi-streaming setting. **Extra credit:** Develop a constant factor for the weighted case.

Solution (sketch). Consider the greedy algorithm, which simply adds an edge while it can.

Let $M^\star$ be an optimal matching, and let M be a greedy matching. Our goal is to show that $|M^\star \setminus M| \leq c|M|$ for some constant $c > 0$. Write $M^\star \setminus M = A \cup B_1 \cup \cdots B_k$, where

- A is the set of edges $e \in M^\star \setminus M$ incident to at least one edge in M.
- B is the set of edges $e \in M^\star \setminus M$ not incident to any edge in M.

For each edge $e \in A$, let $\pi(e) \in M$ be some edge incident to e. Then each edge $f \in M$ is equal to $\pi(e)$ for at most two edges $e \in A$. Thus

$$|A| \leq \sum_{f \in M} \left|\pi^{-1}(e)\right| \leq 2|M|.$$

Any edge $e \in B$ must have been excluded from M because we had already taken the maximum number of edges in that color. Let $I \subseteq [k]$ be the set of colors i such that the number of edges in M of color i is the maximum capacity b_i. For each $i \in I$, let $B_i \subseteq B$ be the subset of edges of color I. Observe that $B \subseteq \bigcup_{i \in I} B_i$ because every edge $e \in B$ is excluded due to color. We have

$$|B| = \sum_{i \in I} B_i \leq \sum_{i \in I} b_i \overset{\text{(b)}}{\leq} |M|,$$

where (b) is because M has b_i edges of color i for each $i \in I$.

Altogether, we have

$$|M^\star| - |M| \leq |M^\star \setminus M| \leq |A| + |B| \leq 3|M|.$$

Thus the greedy matching gives a 4 approximation.

For extra credit, a similar algorithm to weighted matching gives a constant factor approximation. Given an edge in the stream, if we can find a set of edges in our current matching to exchange for the current edge, such that the new edge has weight a factor of (say) 4 greater than the sum weight of the kicked out edges, do so. Otherwise we do not take the new edge.