15-859 Algorithms for Big Data — Fall 2019

Problem Set 1

Due: 08:00, Saturday, September 28

Please see the following link for collaboration and other homework policies:
`http://www.cs.cmu.edu/afs/cs/user/dwoodruf/www/teaching/15859-fall19/grading.pdf`

Problem 1: Uniform Sampling for Regression (13 points)
In class we saw an application of subspace embeddings based on random matrices S to over-constrained least squares regression. We gave several instantiations of S, from Gaussian, to Subsampled Randomized Hadamard Transform, to CountSketch matrices. Here we consider what happens if instead S is based on uniform sampling rows of a matrix A.

To obtain provable guarantees, we will need to make some assumptions on our $n \times d$ input matrix A, with $n \geq d$. Suppose each row A_i of A satisfies $\|A_i\|_2 = 1$. Define the condition number κ of A to be $\frac{\sigma_1(A)}{\sigma_d(A)}$, where $\sigma_i(A)$ is the i-th singular value of A.

Suppose we sample $s = O(\kappa^2 d (\log d)/\epsilon^2)$ rows of A uniformly at random with replacement. Let S be an $s \times n$ matrix so that $S \cdot A$ chooses these s rows of A, that is, $S_{j,i} = \frac{\sqrt{n}}{\sqrt{s}}$ if the i-th row is chosen in the j-th sample, and is 0 otherwise.

Argue that with probability at least 9/10, SA is a subspace embedding, that is, simultaneously for all $x \in \mathbb{R}^d$, $\|SAx\|_2^2 = (1 \pm \epsilon)\|Ax\|_2^2$.

HINT: you might try to adapt the analysis for the Subsampled Randomized Hadamard Transform we did in class, replacing the flattening lemma with an alternative analysis, after which you can try to follow the same analysis from class.

Problem 2: Approximate Matrix Product (12 points)
We saw for several random families of matrices S, given an $m \times n$ matrix A and an $n \times r$ matrix B, if S is an $n \times s$ matrix with $s = \Theta(1/\epsilon^2)$ columns, then

$$\Pr_S[\|ASS^T B - AB\|_F \leq \epsilon \|A\|_F \|B\|_F] \geq 2/3,$$

that is, S satisfies the *approximate matrix product* property.

(1) (7 points) Show that any randomized algorithm which for any inputs A and B, outputs an $m \times s$ matrix C and an $s \times r$ matrix D with

$$\Pr_S[\|CD - AB\|_F \leq (1/2)\|A\|_F \|B\|_F] \geq 2/3$$

has the following property: for every value possible values of $\text{nnz}(A)$ and $\text{nnz}(B)$, there is an input (A, B) where the algorithm reads $\Omega(\min(\text{nnz}(A), \text{nnz}(B)))$ entries of the input in expectation. You can assume you know $\text{nnz}(A)$ and $\text{nnz}(B)$ and are given a list of their non-zero locations. Here the probability and expectation are taken over the coin tosses of the algorithm. Note that it is important to be formal in proving a lower bound. In recitation we will go over Yao's minimax principle which is one way of formalizing this. Feel free to come to office hours to ask for help.

(2) (5 points) Since $\epsilon\|A\|_F\|B\|_F$ can be large, one could hope for alternative notions of approximate matrix product. The following guarantee is called operator norm approximate matrix product: given an $m \times n$ matrix A and an $n \times r$ matrix B, if S is an $n \times s$ matrix for an appropriately chosen s, then

$$\Pr_S[\|ASS^T B - AB\|_2 \leq \epsilon\|A\|_2\|B\|_2] \geq 2/3,$$

where for a matrix C, $\|C\|_2 = \sup_{\|x\|_2=1}\|Cx\|_2$ is the *operator norm* of C. Argue that if S is a matrix of i.i.d. $N(0, 1/s)$ random variables (that is, mean-0 and variance-$1/s$ normal random variables), then the above holds for $s = O((m + r)/\epsilon^2)$.

Problem 3: Properties of CountSketch (10 points)

(1) (5 points) Show that there is an $n \times d$ matrix A, $n \geq d$, for which if S is a randomly drawn CountSketch matrix with $o(d^2)$ rows, then with probability $1 - o(1)$, $S \cdot A$ does not satisfy the $(1 \pm 1/2)$-subspace embedding property. Namely, show that with this probability there is a vector $x \in \mathbb{R}^d$ for which either $\|SAx\|_2^2 > (3/2)\|Ax\|_2^2$ or $\|SAx\|_2^2 < (1/2)\|Ax\|_2^2$.

(2) (5 points) Suppose instead of using random signs in CountSketch, we use random $N(0, 1)$ random variables. More precisely, our sketching matrix $S \in \mathbb{R}^{s\times n}$ is specified as follows: for each of the n columns i of S, we independently sample a uniformly random position $h(i)$, and set $S_{h(i),i}$ to be an $N(0, 1)$ random variable, where $N(0, 1)$ denotes a mean-0 variance-1 normal random variable. The other entries in the i-th column of S are equal to 0. Show there is an $n \times d$ matrix A, $n \geq d$, for which if S is a random such matrix with any number s of rows, then with probability $1-\exp(-\Theta(d))$, $S \cdot A$ does not satisfy the $(1 \pm 1/2)$-subspace embedding property.

Problem 4: TensorSketch (15 points)
Given two vectors $x, y \in \mathbb{R}^n$, the tensor product $x \otimes y$ is a vector in $\mathbb{R}^{n^2}$ indexed by pairs $(i, j) \in \{0, 1, \ldots, n-1\} \times \{0, 1, \ldots, n-1\}$ for which $(x \otimes y)_{i,j} = x_i \cdot y_j$. Suppose one would like to apply a sketching matrix $T \in \mathbb{R}^{s\times n^2}$ to $x \otimes y$, given only x and y. One way is to first compute $x \otimes y$ and then apply T to $x \otimes y$. However, computing $x \otimes y$ can take up to n^2 time, so a natural question is if this can be improved.

Consider instead the following way of sketching a tensor product. Sample two CountSketch matrices $S^1, S^2 \in \mathbb{R}^{s\times n}$ and compute $S^1 \cdot x$ and $S^2 \cdot y$. Define the sketch $T \in \mathbb{R}^{s\times n^2}$ of $x \otimes y$ as follows, where we index coordinates starting at 0:

$$(T \cdot (x \otimes y))_i = \sum_{j,k \text{ with } j+k=i \bmod s} (S^1 \cdot x)_j \cdot (S^2 \cdot y)_k.$$

(1) (5 points) Suppose S^1 has associated hash function $h^1 : \{0, \ldots, n-1\} \to \{0, \ldots, s-1\}$ and sign function $\sigma^1 : \{0, \ldots, n-1\} \to \{-1, 1\}$, and similarly S^2 has associated hash functions h^2 and σ^2. Argue that T is a CountSketch matrix with associated hash function $H(i, j) = h^1(i) + h^2(j) \bmod s$ and sign function $\sigma(i, j) = \sigma^1(i) \cdot \sigma^2(j)$. Specifically, show that for every $i \in [s], j, k \in [n]$, T must satisfy $T_{i,(j,k)} = \sigma(j, k)$ if $H(j, k) = i$ and $T_{i,(j,k)} = 0$ otherwise, where the n^2 columns of T are indexed by pairs (j, k).

(2) (5 points) If $h^1, h^2, \sigma^1, \sigma^2$ are independent and each drawn from a 4-wise independent family, state the largest integers a, b that you can for which H is a-wise independent and σ is b-wise independent. Prove your answer.

(3) (5 points) Show how, given only x and y, one can compute $T \cdot (x \otimes y)$ in $\text{nnz}(x) + \text{nnz}(y) + O(s \log s)$ time. The Fast Fourier Transform may be useful, which shows that for degree at most d univariate polynomials $p(x)$ and $q(x)$ one can compute the coefficients of the polynomial $p(x) \cdot q(x)$ using $O(d \log d)$ arithmetic operations.