15-859 Algorithms for Big Data — Fall 2019

Problem Set 2

Due: before or in class, Thursday, October 17

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: The ABCs of Low Rank Approximation (17 points)

(1) (5 points) In class we saw that if we choose a random CountSketch matrix S with r rows and n columns, then for any fixed matrix $A \in \mathbb{R}^{n \times d}$ and $B \in \mathbb{R}^{n \times m}$, with probability at least $2/3$, simultaneously for all $X \in \mathbb{R}^{d \times m}$,

$$\|SAX - SB\|_F^2 = (1 \pm \epsilon)\|AX - B\|_F^2.$$

We did not specify the exact probability or number r of rows necessary to achieve this property, which we called an affine embedding. Assume for any fixed matrix C, we have $\|SC\|_F^2 = (1 \pm \epsilon)\|C\|_F^2$ with probability at least $9/10$ provided S has $\Omega(1/\epsilon^2)$ rows.

What is the best bound you can show on r so that S satisfies the above affine embedding property?

(2) (12 points) Suppose you are given three matrices $A, B, C \in \mathbb{R}^{n \times n}$. Show how to output a rank-k matrix D, in factored form, so that with probability at least $9/10$,

$$\|A \cdot B \cdot C - D\|_F^2 \leq (1 + \epsilon)\|[A \cdot B \cdot C]_k - A \cdot B \cdot C\|_F^2,$$

where $[A \cdot B \cdot C]_k$ is the best rank-k approximation to $A \cdot B \cdot C$. Your algorithm should run in $(\text{nnz}(A) + \text{nnz}(B) + \text{nnz}(C) + n) \cdot \text{poly}(k/\epsilon)$ time.

Problem 2: Sublinear Time High Accuracy Regression (17 points) We are given an $n \times d$ matrix A with $n \geq d$. Further, A is structured, which for this problem means that for any $r \times s$ submatrix B of A, for $1 \leq r \leq n$ and $1 \leq s \leq d$, one can compute $x^T B$ and By in time at most $T \cdot (rs)/(nd)$, where $x \in \mathbb{R}^r$ and $y \in \mathbb{R}^s$ are arbitrary vectors. Here T is a parameter that depends on A. Note that we always have $T = O(\text{nnz}(A))$, and you can assume $T \geq n + d$.

Show how to obtain an $x' \in \mathbb{R}^d$ in time $(T \cdot \text{poly}(\log(n/\epsilon)) + d^2 \log(1/\varepsilon) + \text{poly}(d))$, such that with probability at least $9/10$, $\|Ax' - b\|_2 \leq (1 + \epsilon) \min_x \|Ax - b\|_2$. Note your running time should have a dependence on ϵ which is logarithmic. Notice also that the running time of this algorithm is much less than $\text{nnz}(A)$ if $T \ll \text{nnz}(A)$.

Note: your algorithm may have a running time better than stated, and the $\text{poly}(\log(n/\epsilon))$ can be as small as $\log(1/\epsilon)$. You will however get full credit if you give any algorithm within the above stated time bounds.

Problem 3: Fun with Leverage Scores (16 points) In this problem $A \in \mathbb{R}^{n \times d}$ is a given input matrix.

(1) (2 points) Let $\ell_i(A)$ denote the i-th column leverage score of A, for $i = 1, 2, \ldots, d$. Note in class we instead looked at the row leverage scores, but we can equivalently look at the column leverage scores, which are defined to be the squared column norms of a basis for the row span of A with orthonormal rows. Suppose the columns of A are linearly independent. What is $\ell_i(A)$ equal to, for $i = 1, 2, \ldots, d$?

(2) (5 points) Now suppose the columns of A are not necessarily linearly independent. Argue that if you append an additional column to A, resulting in an $n \times (d+1)$ matrix A', that for any $1 \leq i \leq d$, it holds that $\ell_i(A') \leq \ell_i(A)$.

For $\lambda = \frac{\|A-A_k\|_F^2}{k}$, we now define the i-th *column rank-k ridge leverage score* $\tau_i = a_i^T(AA^T + \lambda I)^{-1} a_i$, where a_i is the i-th column of A, treated as a column vector. Here we assume that $\|A - A_k\|_F^2 > 0$, and in this case the inverse $(AA^T + \lambda I)^{-1}$ turns out to be well-defined since all of its eigenvalues are positive. We first observe that $\sum_{i=1} \tau_i(A) \leq 2k$. To see this, we can write $A = U\Sigma V^T$:

$$\begin{aligned}\tau_i &= a_i^T \left(U\Sigma^2 U^T + \frac{\|A - A_k\|_F^2}{k} UU^T \right)^{-1} a_i \\ &= a_i^T (U \Omega U^T) a_i\end{aligned} \tag{1}$$

Where $\Omega \in \mathbb{R}^{n \times n}$ is a diagonal matrix with $\Omega_{i,i} = (\Sigma_{i,i}^2 + \|A - A_k\|_F^2/k)^{-1}$. So $\sum_i \tau_i = \sum_i \left(A^T(U\Omega U^T)A\right)_{i,i}$, which is:

$$\begin{aligned}\mathrm{Tr}\left(A^T(U\Omega U^T)A\right) &= \mathrm{Tr}\left(V\Sigma\Omega\Sigma V^T\right) \\ &= \mathrm{Tr}\left(\Sigma\Omega\Sigma\right) \\ &= \sum_{i=1} \frac{\Sigma_{i,i}^2}{\Sigma_{i,i}^2 + \|A - A_k\|_F^2/k}\end{aligned} \tag{2}$$

Now because $\|A - A_k\|_F^2 = \sum_{i>k} \Sigma_{i,i}^2$, we can bound $\sum_{i>k} \frac{\Sigma_{i,i}^2}{\Sigma_{i,i}^2 + \|A-A_k\|_F^2/k} \leq k$, where we used the definition of $\|A - A_k\|_F^2/k$ and dropped $\Sigma_{i,i}^2$ from the denominator. We can also bound $\sum_{i \leq k} \frac{\Sigma_{i,i}^2}{\Sigma_{i,i}^2 + \|A-A_k\|_F^2/k} \leq k$, where we just upper bounded each summand by 1. In the next part, you can use the fact that $\sum_{i=1} \tau_i(A) \leq 2k$.

(3) (9 points) The Matrix Chernoff argument in class implies that if $\bar{\ell}_i \geq \ell_i(A)$, $n \leq d$, and $p_i = \frac{\bar{\ell}_i}{\sum_i \bar{\ell}_i}$, then if we sample $t = O((n \log n)/\epsilon^2)$ columns of A with replacement according to the p_i, forming a sampling and rescaling matrix S, then with probability at

least 9/10, $\|x^T AS\|_2 = (1 \pm \epsilon)\|x^T A\|_2$ for all $x \in \mathbb{R}^n$ simultaneously. Letting $C = AS$, this statement can be equivalently written as

$$(1-\epsilon)CC^T \preceq AA^T \preceq (1+\epsilon)CC^T,$$

where for symmetric matrices P, Q with non-negative eigenvalues, we write $P \preceq Q$ if $x^T Px \leq x^T Qx$ simultaneously for all vectors x.

Let $p_i = \frac{\tau_i}{\sum_j \tau_j}$. Suppose we sample $t = O((k \log n)/\epsilon^2)$ columns of A with replacement according to the distribution of the p_i, and create a sampling and rescaling matrix $C = AS$, where if row i is sampled in the j-th trial, then $S_{i,j} = \frac{1}{\sqrt{tp_i}}$. Argue that with probability at least 9/10,

$$(1-\epsilon)CC^T - \epsilon\lambda I \preceq AA^T \preceq (1+\epsilon)CC^T + \epsilon\lambda I.$$

Note that this can be seen as an analogue of a subspace embedding with additive error.

HINT: Consider the matrix $D = [A, \sqrt{\lambda}I]$, set $\bar{\ell}_i = \tau_i$, and apply the analysis from class. It might also help to note that the standard definition of a leverage score can be rewritten as $\ell_i = a_i^T (AA^T)^{-1} a_i$, where a_i is the i-th column of A.