CS369M: Algorithms for Modern Massive Data Set Analysis **Lecture 3 - 09/30/2009**

Approximating matrix multiplication and low-rank approximation

Lecturer: Michael Mahoney *Scribes: Richa Bhayani and Daniel Chen*

**Unedited notes*

1 Matrix Multiplication using sampling

Given two matrices A of size $m * n$ and B of size $n * p$,our goal is to produce an approximation to the matrix multiplication product $A * B$. We do this by performing c independent trials , where in each trial we randomly sample an element of $1, 2, ..., n$ with an appropriate probability distribution P.We form a matrix C of size $m * c$ consisting of sampled columns of A and a matrix R of size $c * n$ consisting of sampled rows of B, both scaled appropriately.If P and the scaling factors are chosen judiciously, we show that CR is a good approximation of AB. More precisely we show that,

$$||(AB) - (CR)||_F = O(||A||_F||B||_F/\sqrt{c}) \tag{1}$$

1.1 Pass efficiency

The pass-efficient model is based on the notion that you have the ability to store large amounts of data but do not have random access to it.Data is assumed to be on an external disk, to be presented to the algorithm on a read-only tape.The only access the algorithm to the data is via a pass, where a pass is a sequential read of the entire data. Therefore, the resources of interest are the number of passes over the data,the additional space and time required.

1.2 Lemmas

Lemma 1.1. *Suppose that $a_1, a_2, ..., a_n$, $a_i >= 0$, are read in one pass i.e.,one sequential read over the data, by the SELECT algorithm.Then the SELECT algorithm requires $O(1)$ i.e.,constant with respect to n, additional space and returns a random i^* sampled from the probability distribution $Pr[i^* = i] = a_i/\sum_{i'=1}^{n} a_{i'}$.*

Note that this choice of probability assignment naturally favors heavier elements.

Algorithm 1 SELECT algorithm

Require: $a_1, ..., a_n$, $a_i >= 0$, read in one pass.

```
D=0.
for i = 1 to n do
    D = D + a.
    With probability a_i/D, let i_* = i and a_{i*} = a_i.
end for
return i_*, a_{i*} .
```

Lemma 1.2. *Suppose that $A \epsilon R^{m*n}$ is presented in the sparse-unordered representation and is read in one pass by the SELECT algorithm. Then, the algorithm requires $O(1)$ and returns $Pr[i^* = 1 and j^* = j] = |A_i|^2/||A||_F{}^2$*

$Pr[i^* = 1 and j^* = j] = |A_i|^2/||A||_F{}^2$ is obtained simply by marginal probability definition, whereas the first claim follows from the previous lemma.

Algorithms such as SELECT which select elements without looking at all the data, are known as reservoir algorithms.

1.3 The BasicMatrixMultiplication algorithm

This is a simple algorithm to approximate the product AB.Randomly sample with replacement, the terms from A and B , in the summation c times according to a probability distribution p_i where i runs from 1 to n; scale each term appropriately and output the sum of the scaled terms.

Algorithm 2 BASICMATRIXMULTIPLICATION algorithm

Require: $A \epsilon R^{m*n}$ and $B \epsilon R^{n*p}$, $C \epsilon Z^+$ such that 1 <= c <=n and $(p_i)_{i=1}^n$, are such that $p_i > = 0 and \sum {p_i}_{i=1}{}^n = 1$

1: **for** $t = 1$ to c **do**
2: Pick $i_t \epsilon 1, ..., n$ with $Pr[i_t = k] = p_k, k = 1, ..., n$, independently.
3: Set $C^{(t)} = A^{(i_t)}/\sqrt{cp_{i_t}}$ and $R_{(t)} = B_{(i_t)}/\sqrt{cp_{i_t}}$
4: **end for**
5: **return** C, R .

1.4 Analysis of the sampling and running time

In the case of uniform sampling, you simply perform 'c' trials and construct C and R , thus needing $O(c(m + p))$ additional time and space. In the case of non-uniform sampling however, one must first scan through column-row pairs to decide which to sample, requiring $O(n)$ additional time and space.

1.5 Analysis of the algorithm for arbitrary probabilities

We make use of the Jensen's inequality to prove upper bounds for $||AB - CR||_F^2$.

Lemma 1.3. *Suppose* $A \epsilon R^{m*n}$ *and* $B \epsilon R^{n*p}$, $C \epsilon Z^+$ *such that* $1 <= c <= n$ *and* $(p_i)_{i=1}^n$, *are such that* $p_i > = 0 and \sum {p_i}_{i=1}{}^n = 1$. *Then,* $E[(CR)_{ij}] = (AB)_{ij}$ *and* $Var[(CR)_{ij}] = (1/c) * (AB)_{ij}$

This lemmma proves that the expectation of the (i, j)th element of the approximation is equal to the (i, j)th element of the exact product. The proof for this lemma is discussed in class. We fix i, j, and define for some t , $X_t = (A^{(i_t)} B_{(i_t)})/cp_{i_t})_{ij}$. Thus, using the definition of expectation of a random variable, we sum over different values of t , ranging from 1 to n and get the desired expressions for expectation and variance.

Lemma 1.4. *Suppose* $A \epsilon R^{m*n}$ *and* $B \epsilon R^{n*p}$, $C \epsilon Z^+$ *such that* $1 <= c <= n$ *and* $(p_i)_{i=1}^n$, *are such that* $p_i > = 0 and \sum {p_i}_{i=1}{}^n = 1$. *Then,* $E[||AB - CR||_F{}^2] = \sum_{k=1}^n |A^k|^2 |B_k|^2)/cp_k - 1/c * ||AB||_F{}^2$. *Furthermore, if* $p_k = |A^{(k)}||B_{(k)}|/\sum_{k'=1}^n |A^{(k')}||B_{(k')}|$ *then ,* $E[||AB - CR||_F{}^2] = 1/c * \sum_{k=1}^n (|A^k||B_k|)^2 - 1/c * ||AB||_F{}^2$

Note that by definition $E[||AB - CR||_F{}^2] = \sum_{i=1}^m \sum_{j=1}^p Var[(CR)_{ij}]$ and so from lemma above, by mere substitution we get the result. To prove that this result is optimal however, we use a Lagrange multiplier. Let $f(p_1, ...p_n) = \sum_{k=1}^n 1/p_k * (|A^k|^2)|B_k|^2$ Then, let λ be the Lagrange multiplier and we define a $g(p_1, ..., g_n) = f.. + \lambda(\sum_{k=1}^n p_k - 1)$. We get the minimum by taking the derivatives.And hence, the results.

1.6 Analysis of the algorithm for near-optimal probabilities

We proved the above results by using Jensen's. Now to prove the tightness we analyse for near optimal probabilities. Optimal probabilities are those that minimize the $E[||AB - CR||_F{}^2$. For proving that the

results change in only small β dependent loss in accuracy if the probabilities were defined instead to be $p_k >= \beta * |A^{(k)}|B_{(k)}|/\sum_{k'=1}^{n}|A^{(k')}|B_{(k')}|$.

Theorem 1.5. *Suppose* $A \epsilon R^{m*n}$ *and* $B \epsilon R^{n*p}$, $C \epsilon Z^{+}$ *such that* $1 <= c <= n$ *and* $(p_i)_{i=1}^{n}$ *and for some small constant* $\beta <= 1$, $p_k >= \beta * |A^{(k)}|B_{(k)}|/\sum_{k'=1}^{n}|A^{(k')}|B_{(k')}|$ *then,* $E[||AB - CR||_F{}^2 <= (1/\beta * c) * ||A||_F{}^2||B||_F{}^2$ *. Furthermore, let* $\delta \epsilon (0,1)$ *and* $\eta = 1 + \sqrt{(8/\beta)log(1/\beta)}$*. Then, with probability atleast* $1 - \delta$, $((||AB - CR||)_F)^2 <= (\eta^2/\beta * c) * ||A||_F{}^2||B||_F{}^2$

Using the same steps as previous lemma , only using the updates values of probabilities we get the updated expectation, which by Cauchy Schwartz gives you $E[||AB - CR||_F{}^2 <= (1/\beta * c) * ||A||_F{}^2||B||_F{}^2$. To derive the rest one can make use of triangle inequality and Cauchy Schwartz. Interested readers can look at the paper.

An important consequence of this theorem is that by choosing enough number of column-row pairs, the error can be made arbitrarily small.

2 Low-Rank Matrix Approximation by Sampling

Given a matrix A, we seek to compute what is in some sense an approximation to the SVD of A. When we compute a SVD, we find a rank-k matrix U_k that best approximates the column space of A. Here, we seek a rank-k matrix H_k which does not do much worse than U_k. To do so, we begin with two facts from matrix perturbation theory that will be central to our analysis: If $A, E \in \mathbb{R}^{m \times n}$, then

$$\max_{1 \le t \le n} |\sigma_t(A + E) - \sigma_t(A)| \le \|E\|_2 \tag{2}$$

and

$$\sum_{k=1}^{n} (\sigma_t(A + E) - \sigma_t(A))^2 \le \|E\|_F^2 \tag{3}$$

The algorithm is described as follows:

Algorithm 3 LinearTimeSVD

Require: $A \in \mathbb{R}^{m \times n}$, $c, k \in \mathbb{Z}^+$ such that $1 \le k \le c \le n$, $\{p_i\}_{i=1}^{n}$ such that $p_i \le 0$ and $\sum_{i=1}^{n} p_i = 1$.
Ensure: $H_k \in \mathbb{R}^{m \times k}$ and $\sigma_t(C), t = 1, \ldots, k$.

1: **for** $t = 1$ to c **do**
2: Pick $i_t \in 1, \ldots, n$ with $\Pr(i_t = \alpha) = p_\alpha, \alpha = 1, \ldots, n$.
3: Set $C^{(t)} = A^{(i_t)}/\sqrt{cp_{i_t}}$.
4: **end for**
5: Compute $C^T C$ and its SVD; say $C^T C = \sum_{t=1}^{c} \sigma_t^2(C) y^t y^{tT}$.
6: Compute $h^t = Cy^t/\sigma_t(C)$ for $t = 1, \ldots, k$.
7: **return** H_k, where $H_k^{(t)} = h^t$, and $\sigma_t(C), t = 1, \ldots, k$.

This algorithm is pass efficient because the probabilities p_α can be calculated in one pass and the matrix C can be determined in another pass. It uses $O(mc^2)$ time for computing $C^T C$ and its SVD, and uses $O(mck)$ total time to compute the columns h^t. Therefore, the total additional time required by this algorithm is $O(mc^2)$, which is linear in the dimension m. The approximation H_k achieves the following bounds:

Theorem 2.1. *Suppose* $A \in \mathbb{R}^{m \times n}$ *and let* H_k *be constructed from the LinearTimeSVD algorithm. Then*

$$\|A - H_k H_k^T A\|_2^2 \le \|A - A_k\|_2^2 + 2\|AA^T - CC^T\|_2 \tag{4}$$

and

$$\|A - H_k H_k^T A\|_F^2 \le \|A - A_k\|_F^2 + 2\sqrt{k}\|AA^T - CC^T\|_F \tag{5}$$

We will prove Equation (4) in class. We will sketch the proof for Equation (5) but Interested readers can find the the full proof in the references.

Proof for Equation (4). Let $\mathcal{H}_k = \text{range}(H_k) = \text{span}(h^{(1)}, h^{(2)}, \ldots, h^{(k)})$ and let $\mathcal{H}_{m-k}$ be the orthogonal complement of $\mathcal{H}_k$. Furthermore, consider any vector $x \in \mathbb{R}^m$. We can write x as $\alpha y + \beta z$ where $y \in \mathcal{H}_k$ and $z \in \mathcal{H}_{m-k}$. Then, we have:

$$\begin{aligned}
\|A - H_k H_k^T A\|_2 &= \max_{x \in \mathbb{R}^m, |x|=1} |x^T (A - H_k H_k^T A)| && (6)\\
&= \max_{y \in \mathcal{H}_k, |y|=1, z \in \mathcal{H}_{m-k}, |z|=1, \alpha^2+\beta^2=1} |(\alpha y^T + \beta z^T)(A - H_k H_k^T A)| && (7)\\
&\leq \max_{y \in \mathcal{H}_k, |y|=1} |y^T (A - H_k H_k^T A)| + \max_{z \in \mathcal{H}_{m-k}, |z|=1} |z^T (A - H_k H_k^T A)| && (8)\\
&= \max_{z \in \mathcal{H}_{m-k}, |z|=1} |z^T A| && (9)
\end{aligned}$$

Equation (6) follows because for any matrix B, $\|B\|_2 = \sigma_1(B) = \sigma_1(B^T) = \|B^T\|_2$ and equation (9) follows because y projected onto $\mathcal{H}_k$ is y and z projected onto $\mathcal{H}_k$ is 0. Then, we consider $z^T A$ with $z \in \mathcal{H}_{m-k}$:

$$\begin{aligned}
|z^T A|^2 &= z^T A A^T z && (10)\\
&= z^T C C^T z + z^T (A A^T - C C^T) z && (11)\\
&\leq \sigma_{k+1}^2(C) + \|A A^T - C C^T\|_2 && (12)\\
&= \sigma_{k+1}(C C^T) + \|A A^T - C C^T\|_2 && (13)\\
&\leq \sigma_{k+1}(A A^T) + 2\|A A^T - C C^T\|_2 && (14)\\
&= \sigma_{k+1}^2(A) + 2\|A A^T - C C^T\|_2 && (15)\\
&= \|A - A_k\|_2^2 + 2\|A A^T - C C^T\|_2 && (16)
\end{aligned}$$

The first half of inequality (12) follows because z is in $\mathcal{H}_{m-k}$ and hence the maximum value of $|z^T C|$ is $\sigma_{k+1}(C)$. The second half of (12) follows from submultiplicativity of the spectral norm. Inequality (14) follows from (2). □

Sketch of Proof for Equation (5). We first note that $||X||_F^2 = \text{Tr}(X^T X)$. Then, we can expand $\|A - H_k H_k^T A\|_F^2$ as a trace of a matrix product and obtain $\|A\|_F^2 - \|A^T H_k\|_F^2$. Then, we prove the following claim:

$$\left| \|A^T H_k\|_F^2 - \sum_{t=1}^{k} \sigma_t^2(C) \right| \leq \sqrt{k} \|A A^T - C C^T\|_F \tag{17}$$

With (3) we can also show:

$$\left| \sum_{t=1}^{k} \sigma_t^2(C) - \sum_{t=1}^{k} \sigma_t^2(A) \right| \leq \sqrt{k} \|A A^T - C C^T\|_F \tag{18}$$

From these two inequalities, we obtain (5). □

From (4) and (5), it can then be show that with high probability,

$$\|A - H_k H_k^T A\|_2 \leq \|A - A_k\|_2^2 + \epsilon \|A\|_F^2 \tag{19}$$

and

$$\|A - H_k H_k^T A\|_F \leq \|A - A_k\|_F^2 + \epsilon \|A\|_F^2 \tag{20}$$

if p_i are nearly optimal and ϵ is chosen judiciously.