Sketching Algorithms for Big Data Fall 2017

Lecture 11 — October 5, 2017

Prof. Jelani Nelson *Scribe: Hong Hu*

1 Overview

In the last lecture, we discussed fast algorithms for computing JL transform using the idea of sampling. In this and the next few lectures, we are going to see how to use sampling and embedding technique to obtain faster algorithm for the following problems:

- Matrix multiplication
- Regression
- PCA/low-rank approximation

In this lecture, we mainly focus on fast matrix multiplication.

2 Matrix Multiplication

Suppose we have two matrices $A \in \mathbb{R}^{n\times d}$, $B \in \mathbb{R}^{n\times p}$, written as:

$$A^T = \begin{bmatrix} | & \cdots & | \\ a_1 & \cdots & a_n \\ | & \cdots & | \end{bmatrix}, \quad B = \begin{bmatrix} -b_1^T- \\ \vdots \\ -b_n^T- \end{bmatrix}$$

where $a_i \in \mathbb{R}^d, b_i \in \mathbb{R}^p$. We want to compute A^TB.

The naive approach for doing this requires $\mathcal{O}(ndp)$ for loops. There are some faster algorithms. For square matrix multiplication, the following algorithms are of less complexity $\mathcal{O}(\omega)$:

1. $\omega < \log_2 7$ (Strassen)
2. $\omega < 2.376$ (Coppersmith, Winograd)
3. $\omega < 2.374$ (Stothevs)
4. $\omega < 2.3728642$ (Vassilevke-Williams)
5. $\omega < 2.3728639$ (Le Gell)

and for multiplying arbitrary matrix, it suffices to break them into multiple square matrix multiplications. All these algorithms are exact computation. What we are going to show today are some randomized algorithms, which give us answers closed to the exact multiplication with high probability, to be more exact, we want to compute $C \in \mathbb{R}^{d\times p}$, s.t.

$$\|A^T B - C\|_X < \varepsilon, \text{with probability} > 1-\delta$$

where X is some matrix norm like Frobenius norm ($\|M\|_F = (\sum_{i,j} M_{i,j}^2)^{\frac{1}{2}}$), l_2 operator norm ($\|M\| = \sup_{\|x\|=1} |x^T M x|$), etc.

The randomized algorithm for matrix multiplication was first studied in [1]. The methods developed ever since then fall into two main categories:

- Sampling approach
- JL-based approach

We are going to analyze two algorithms, one in each category.

2.1 Sampling Approach

Here we analyzed the algorithm proposed in [1]. The starting point is to rewrite $A^T B$ as a sum of n rank-1 matrices:

$$A^T B = \sum_{i=1}^{n} a_i b_i^T$$

Then to reduce computational complexity, we can sample m rows from A and B using sampling matrix Π:

$$\Pi = \frac{1}{\sqrt{m}} \left.\overbrace{\begin{pmatrix} 0 & 0 & \cdots & \frac{1}{\sqrt{p_{i_1}}} & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 0 & \cdots & \frac{1}{\sqrt{p_{i_m}}} & 0 \end{pmatrix}}^{n}\right\} m$$

There is only one non-zero element in each row of Π, and in the ith row, the jth element $\Pi_{i,j} = \frac{1}{\sqrt{p_j}}$ with probability p_j. Then matrices $\Pi A \in \mathbb{R}^{m\times d}$ and $\Pi B \in \mathbb{R}^{m\times p}$ are the sampled versions of A and B. We use

$$C = (\Pi A)^T \Pi B = \frac{1}{m}\sum_{k=1}^{m} \frac{a_{i_k} b_{i_k}^T}{p_{i_k}}$$

to approximate $A^T B$, where i_k denotes the index of non-zero element in kth row of Π. Note here i_k is random.

To prove the correctness of this algorithm, we first show that $\mathbb{E}\, C = A^T B$. For each $\frac{a_{i_k} b_{i_k}^T}{p_{i_k}}$, we have:

$$\mathbb{E}\, \frac{a_{i_k} b_{i_k}^T}{p_{i_k}} = \sum_{j=1}^{m} p_j \frac{a_j b_j^T}{p_j} = A^T B$$

Therefore, we can easily get:

$$\mathbb{E}\, C = \frac{1}{m}\sum_{k=1}^{m} \mathbb{E}\, \frac{a_{i_k} b_{i_k}^T}{p_{i_k}} = A^T B$$

Next, the goal is to show $\mathbb{P}(\|C - A^T B\|_F > \varepsilon \|A\|_F \|B\|_F) < \eta$, which can be reached by using second moment method:

$$\mathbb{P}(\|C - A^T B\|_F^2 > \varepsilon^2 \|A\|_F^2 \|B\|_F^2) < \frac{\mathbb{E}\,\|C - A^T B\|_F^2}{\varepsilon^2 \|A\|_F^2 \|B\|_F^2}$$

The idea in [1] is to optimize over sampling probability p_i s.t. $\mathbb{E}\,\|C - A^T B\|_F^2$ is minimized. The optimal $p_i \propto \|a_i\|_2 \|b_i\|_2$. After minimization, it can be shown that $\frac{\mathbb{E}\,\|C - A^T B\|_F^2}{\varepsilon^2 \|A\|_F^2 \|B\|_F^2} < \frac{C}{\varepsilon^2 m}$, so it suffices to have $m \geq \frac{C}{\varepsilon^2 \eta}$. Note that after doing this, $m \propto \frac{1}{\eta}$ and in order to get $m \propto \log \frac{1}{\eta}$ as desired, we need to use the "median approach":

1. Run the above algorithm many times independently with $\eta = \frac{1}{3}$
2. Obtain $C_1, \ldots, C_t$, $t = \Theta(\lg \frac{1}{\delta})$
3. Pick C_i that is accurate enough.

For the 3rd step, it is impossible to check if $\|C - A^T B\|_F^2 > \varepsilon^2 \|A\|_F^2 \|B\|_F^2$, since we don't know the exact $A^T B$. In [3], the authors proposed a way to do this via checking the pairwise difference $\|C_i - C_j\|_F,\ i, j = 1, \ldots, t$. Let

$$S_i = |\{j : \|C_i - C_j\|_F \leq 2\varepsilon \|A\|_F \|B\|_F\}|$$

the algorithm returns any C_i s.t. $S_i \geq \frac{t}{2}$.

This algorithm can be understood as follows: if $\|C_i - A^T B\|_F \leq \varepsilon \|A\|_F \|B\|_F$, $\|C_j - A^T B\| \leq \varepsilon \|A\|_F \|B\|_F$, then by triangle inequality, we have $\|C_i - C_j\|_F \leq \|C_i - A^T B\|_F + \|C_j - A^T B\|_F \leq 2\varepsilon \|A\|_F \|B\|_F$. On the contrary, if $\|C_i - A^T B\|_F$ is large, then by triangle inequality again for any $\|C_j - A^T B\|_F$ small, $\|C_i - C_j\|_F \geq \|C_i - A^T B\|_F - \|C_j - A^T B\|_F$, which can still be large. Therefore, with $\eta = \frac{1}{3} < \frac{1}{2}$, for a good C_i, more than half $\|C_i - C_j\|_F$ will be less than $2\varepsilon \|A\|_F \|B\|_F$, while for a bad C_i, more than half $\|C_i - C_j\|_F$ can be large.

Since computing $\|A\|_F, \|B\|_F$ requires only $\mathcal{O}(nd + nr)$, the most time-consuming step is doing pair-wise comparison. The worst time complexity is of $\mathcal{O}(\lg^2 \frac{1}{\delta} rd)$, one open question is that if it is possible to reduce it to $\mathcal{O}(\lg \frac{1}{\delta} rd)$.

2.2 JL-based Approach

The JL-based approach was first proposed in [2]. First, we introduce (ε, δ, p) JL moment property (JLMP) to characterize l_p norm of the difference $\|\pi x\|^2 - \|x\|^2$ in JL mapping:

Definition 1. *$\Pi \in \mathbb{R}^{m \times n}$ and D is a distribution over Π. D satisfies the $(\varepsilon, \delta, p)-$JL moment property if for any x of unit norm, we have $\mathbb{E}_{\Pi \sim D} |\|\Pi x\|_2^2 - 1|^p < \varepsilon^p \delta$.*

In fact, there are several well-known matrices satisfying JLMP:

1. Dense sub-Gaussian matrix: $(\varepsilon, \delta, \lg \frac{1}{\delta})-$ JLMP, with $m \simeq \frac{1}{\varepsilon^2} \log \frac{1}{\delta}$
2. AMS sketch matrix: $(\epsilon, \delta, 2)-$ JLMP with $m \simeq 1/\epsilon^2\delta$.
3. Fast JL matrix: $(\epsilon, \delta, \lg(\frac{n}{\delta}))-$ JLMP with $m \simeq \frac{1}{\varepsilon} \lg \frac{1}{\delta}$

All these examples can be proved by using the fact:

$$\mathbb{E}\,|Z|^p = \int_0^\infty px^{p-1}P(|Z| > x)dx \tag{1}$$

and combining the probability tail bound on $\|\Pi x\| - 1$. A more detailed exploration on this can be found in [4]. Here, we assume that such a D exists and utilize some properties of JLMP for our construction.

One property of JLMP we are going to use below is that a random matrix Π satisfying $(\varepsilon, \delta, p)-$JLMP can preserve inner product w.r.t. l_p norm:

Claim 2. *If Π comes from $(\varepsilon, \delta, p)-$JLMP, $p \geq 1$, then $\forall x, y$ of unit norm,*

$$\| \langle \Pi x, \Pi y \rangle - \langle x, y \rangle \|_p \leq (3\varepsilon)\delta^{\frac{1}{p}}$$

Proof. The inner product of x, y can be expressed by their l_2 norm:

$$\langle x, y \rangle = \frac{1}{2}(\|x\|_2^2 + \|y\|_2^2 - \|x - y\|_2^2) \tag{2}$$

$$\langle \Pi x, \Pi y \rangle = \frac{1}{2}(\|\Pi x\|_2^2 + \|\Pi y\|_2^2 - \|\Pi(x - y)\|_2^2) \tag{3}$$

thus

$$\langle \Pi x, \Pi y \rangle - \langle x, y \rangle = \frac{1}{2}(\|\Pi x\|_2^2 - 1 + \|\Pi y\|_2^2 - 1 + \|\Pi(x - y)\|_2^2 - \|x - y\|_2^2)$$

By triangle inequality $\|x - y\|_2 \leq 2$ and also:

$$\begin{aligned}
\| \langle \Pi x, \Pi y \rangle - \langle x, y \rangle \|_p &\leq \frac{1}{2}\big\|\|\Pi x\|_2^2 - 1\big\|_p + \frac{1}{2}\big\|\|\Pi y\|_2^2 - 1\big\|_p + \frac{1}{2}\big\|\|\Pi(x - y)\|_2^2 - \|x - y\|_2^2\big\|_p \\
&\leq \frac{\varepsilon\delta^{\frac{1}{p}}}{2} + \frac{\varepsilon\delta^{\frac{1}{p}}}{2} + \frac{4\varepsilon\delta^{\frac{1}{p}}}{2} \\
&= (3\varepsilon)\delta^{\frac{1}{p}}
\end{aligned}$$

□

Now we are ready to state a theorem in [5], which shows how JLMP can help us bound the Frobenius distance between C and $A^T B$:

Theorem 3. *Suppose D has $(\varepsilon, \delta, p)-$JLMP for $p \geq 2$, then for A, B as before,*

$$P_{\Pi \sim D}(\|A^T B - (\Pi A)^T \Pi B\|_F > 3\varepsilon\|A\|_F\|B\|_F) < \delta$$

Proof. The idea is to first bound $\mathbb{P}_{\Pi\sim D}(\|A^TB-(\Pi A)^T\Pi B\|_F > 3\varepsilon\|A\|_F\|B\|_F)$ by Markov inequality:

$$\mathbb{P}_{\Pi\sim D}(\|A^TB-(\Pi A)^T\Pi B\|_F > 3\varepsilon\|A\|_F\|B\|_F) < \frac{\mathbb{E}\,\|A^TB-(\Pi A)^T\Pi B\|_F^p}{(3\varepsilon\|A\|_F\|B\|_F)^p} \tag{4}$$

and then bound $\mathbb{E}\,\|A^TB-(\Pi A)^T\Pi B\|_F^p$. Let $M \triangleq A^TB-(\Pi A)^T\Pi B$, we have

$$\begin{aligned}M_{ij}^2 &= (\langle \Pi a_i, \Pi b_j\rangle - \langle a_i, b_j\rangle)^2\\ &= \left(\left\langle \Pi\frac{a_i}{\|a_i\|}, \Pi\frac{b_j}{\|b_j\|}\right\rangle - \left\langle \frac{a_i}{\|a_i\|}, \frac{b_j}{\|b_j\|}\right\rangle\right)^2 \|a_i\|_2^2\|b_j\|_2^2\end{aligned}$$

and we define $X_{ij} \triangleq \left\langle \Pi\frac{a_i}{\|a_i\|}, \Pi\frac{b_j}{\|b_j\|}\right\rangle - \left\langle \frac{a_i}{\|a_i\|}, \frac{b_j}{\|b_j\|}\right\rangle$. The l_p norm of $\|M\|_F$ can be rewritten as:

$$\mathbb{E}\,\|M\|_F^p = \left\|\|M\|_F^2\right\|_{\frac{p}{2}}^{\frac{p}{2}} = \left\|\sum_{i,j} M_{ij}^2\right\|_{\frac{p}{2}}^{\frac{p}{2}} \tag{5}$$

Since $p \geq 2 \Rightarrow \frac{p}{2} \geq 1$, we can use triangle inequality over $\left\|\sum_{i,j} M_{ij}^2\right\|_{\frac{p}{2}}$:

$$\begin{aligned}\left\|\sum_{i,j} M_{ij}^2\right\|_{\frac{p}{2}} &= \left\|\sum_{i,j} X_{ij}^2\|a_i\|_2^2\|b_j\|_2^2\right\|_{\frac{p}{2}}\\ &\leq \sum_{i,j}\left\|X_{ij}^2\|a_i\|_2^2\|b_j\|_2^2\right\|_{\frac{p}{2}}\\ &= \sum_{i,j}\|a_i\|_2^2\|b_j\|_2^2\|X_{ij}\|_p^2\\ &\leq (3\varepsilon\delta^{\frac{1}{p}})^2\sum_{i,j}\|a_i\|_2^2\|b_j\|_2^2 \qquad \text{using Claim 2 on } \frac{a_i}{\|a_i\|}, \frac{b_i}{\|b_i\|}\\ &= (3\varepsilon\delta^{\frac{1}{p}})^2\|A\|_F^2\|B\|_F^2\end{aligned}$$

Combined with (5),

$$\begin{aligned}\mathbb{E}\,\|M\|_F^p &= \left\|\sum_{i,j} M_{ij}^2\right\|_{\frac{p}{2}}^{\frac{p}{2}}\\ &\leq (3\varepsilon\delta^{\frac{1}{p}})^p\|A\|_F^p\|B\|_F^p\end{aligned}$$

Back to (4), we get:

$$\mathbb{P}_{\Pi\sim D}(\|A^TB-(\Pi A)^T\Pi B\|_F > 3\varepsilon\|A\|_F\|B\|_F) < \delta$$

□

Comment 1: To achieve low storage and computation complexity, we need to ensure JL mapping matrix Π is sparse. Sub-Gaussian and AMS sketching matrix we used before are all dense matrices, which are not suitable here. In [6], it is shown that Countsketch matrix Π satisfies $(\varepsilon, \delta, 2)-$JLMP for $m \simeq \frac{1}{\varepsilon^2\delta}$ and we know that each column of Countsketch matrix has exact one non-zero element (±1), so it is sparse and can be applied here.

Comment 2: It can be seen that JLMP-based approach doesn't require any knowledge about matrix A and B, but for sampling-based approach discussed before, we need to know the norm of each row a_i, b_i to determine the sampling probability.

2.3 Subspace Embedding

Up to now, we use Frobenius norm to characterize the distance between C and A^TB. In some applications, however, other norms are more relevant. Next, we are going to analyze the cases where l_2 operator norm $\|\cdot\|$ is used.

As before, we expect to obtain results like:

$$\mathbb{P}(\|A^TB - (\Pi A)^T\Pi B\| > \varepsilon\|A\|\|B\|) < \delta \tag{6}$$

We consider the case $A = B$. In this case, (6) becomes:

$$\mathbb{P}(\|A^TA - (\Pi A)^T\Pi A\| > \varepsilon\|A\|^2) < \delta \tag{7}$$

Recall that for symmetric matrix $M \in \mathbb{R}^{d\times d}$,

$$\|M\| = \sup_{\|x\|=1} |x^TMx|$$

so (7) is equivalent to say that we want $\forall x, \|x\| = 1$,

$$\left|\|\Pi Ax\|_2^2 - \|Ax\|_2^2\right| < \varepsilon \sup_{\|z\|=1} \|Az\|_2^2$$

with probability greater than $1 - \delta$. In the following, we will base on something stronger:

$$\left|\|\Pi Ax\|_2^2 - \|Ax\|_2^2\right| < \varepsilon\|Ax\|_2^2$$

and such Π is called $\varepsilon-$*subspace embedding* of $\mathrm{Col}(A)$. A formal definition is as follows:

Definition 4. *For a linear subspace $E \subseteq \mathbb{R}^n$, we say Π is $\varepsilon-$subspace embedding (s.e.) for E if*

$$\left|\|\Pi x\|_2^2 - 1\right| \le \varepsilon, \quad \forall x \in E, \ \|x\|_2 = 1 \tag{8}$$

In fact, we have another equivalent definition by using the orthogonal basis U of E. Then E can be expressed as: $E = \{x : x = Uz, z \in \mathbb{R}^d\}$ with $U^TU = I, U \in \mathbb{R}^{n\times d}$. Therefore, (8) holds iff $\forall x = Uz \in E$,

$$\begin{aligned}
&(1-\varepsilon)\|x\|_2^2 \le \|\Pi x\|_2^2 \le (1+\varepsilon)\|x\|_2^2 \\
\Leftrightarrow\ &(1-\varepsilon)\|Uz\|_2^2 \le \|\Pi Uz\|_2^2 \le (1+\varepsilon)\|Uz\|_2^2, \quad \forall z \in \mathbb{R}^d \\
\Leftrightarrow\ &(1-\varepsilon)\|z\|_2^2 \le \|\Pi Uz\|_2^2 \le (1+\varepsilon)\|z\|_2^2, \quad \forall z \in \mathbb{R}^d \\
\Leftrightarrow\ &\|(\Pi U)^T\Pi U - I\| \le \varepsilon
\end{aligned} \tag{9}$$

We can see that (9) gives us an equivalent definition of $\varepsilon-$subspace embedding, in terms of l_2 operator norm.

Next, we are going to see an example of using operator norm in approximating matrix multiplication:

Example (ordinary least square regression): Given $X \in \mathbb{R}^{n\times d}, y \in \mathbb{R}^n$, least square (LS) regression computes the best linear approximation to data point y using X:

$$\begin{aligned}\beta^{LS} &= \underset{\beta\in\mathbb{R}^d}{\text{argmin}} \|X\beta - y\|_2^2 \\ &= (X^TX)^{-1}X^Ty\end{aligned}$$

Thus the best linear approximation is $X\beta^{LS} = X(X^TX)^{-1}X^Ty$ and $X(X^TX)^{-1}X^T$ is called *projection matrix*, which projects any $y \in \mathbb{R}^n$ onto the subspace Col(X) and the projection is $X(X^TX)^{-1}X^Ty$.

To calculate projection matrix, the term $(X^TX)^{-1}$ incurs highes complexity. The naive approach needs $\mathcal{O}(nd^2)$ for loops and we want to compute it faster. The main idea is to embed X and y into lower-dimensional space: $X \mapsto \Pi X, y \mapsto \Pi y$ and do regression on ΠX and Πy. First, we need to ensure such embedding will not introduce large errors, which can be guaranteed by using $\varepsilon-$subspace embedding:

Claim 5. *Define $E = span(\{Col(X), y\})$ and we assume $rank(X) = d$, so $\dim(E) \leq d+1$. If Π is an $\varepsilon-$ subspace embedding for E, then*

$$\|X\tilde{\beta}^{LS} - y\|_2^2 \leq \frac{1+\varepsilon}{1-\varepsilon}\|X\beta^{LS} - y\|_2^2$$

where $\tilde{\beta}^{LS} = \underset{\beta\in\mathbb{R}^d}{argmin}\|\Pi X\beta - \Pi y\|_2^2$.

Proof. First, we have

$$\begin{aligned}\|\Pi X\tilde{\beta}^{LS} - \Pi y\|_2^2 &\leq \|\Pi X\beta^{LS} - \Pi y\|_2^2 && \text{by definition of } \tilde{\beta}^{LS} \\ &\leq (1+\varepsilon)\|X\beta^{LS} - y\|_2^2 && \Pi \text{ is an } \varepsilon-\text{subspace embedding matrix}\end{aligned}$$

On the other hand,

$$\|\Pi X\tilde{\beta}^{LS} - \Pi y\|_2^2 \geq (1-\varepsilon)\|X\tilde{\beta}^{LS} - y\|_2^2$$

Combining the above two inequalities, we obtain the results. □

The remaining task is to find an $\varepsilon-$ subspace embedding matrix Π for Col$(\widetilde{X})$, where $\widetilde{X} = [X\ y]$. If $\widetilde{X} \in \mathbb{R}^{n\times(d+1)}$ is a tall matrix, i.e., $n \geq d+1$, a quick way to find Π is via *singular value decomposition* (SVD). The definition of SVD is given by the following theorem:

Theorem 6. *Every real matrix $A \in \mathbb{R}^{n\times d}$ with $rank(A) = r$ can be written as:*

$$A = U\Sigma V^T \tag{10}$$

where $U \in \mathbb{R}^{n\times r}, V \in \mathbb{R}^{d\times r}$, $U^TU = I$, $V^TV = I$ and $\Sigma = diag(\sigma_1, \sigma_2, \ldots, \sigma_r)$, $\sigma_i > 0$. Here, σ_i are called singular values.

If we write $U = [u_1 \cdots u_r]$ and $U = [v_1 \cdots v_r]$, where u_i, v_i are called left/right singular vectors, respectively, (10) can also be written as:

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T \tag{11}$$

We can see from Theorem 6 that $\{u_1, \ldots, u_r\}$ is an orthogonal basis of $\mathrm{Col}(A)$, so for tall matrix $\widetilde{X} = \widetilde{U}\widetilde{\Sigma}\widetilde{V}^T$, we can choose $\Pi = \widetilde{U}^T$ and $\Pi\widetilde{U} = I$, which apparently satisfies (9).

In practice, however, doing SVD requires the same complexity as doing original matrix multiplication, so we need other approaches to realize fast subspace embedding. This will be discussed in the next lecture.

References

[1] Petros Drineas, Ravi Kannan, Michael Mahoney. Fast Monte Carlo Algorithms for Matrices I: Approximating Matrix Multiplication. *SIAM J. Comput* 36(1):132157, 2006.

[2] Tamas Sarlos. Improved Approximation Algorithms for Large Matrices via Random Projections. *FOCS* 2006.

[3] Kenneth Clarkson, David Woodruff. Numerical Linear Algebra in the Streaming Model. *STOC*, 205–214, 2009.

[4] Mihir Bellare, John Rompel. Randomness-Efficient Oblivious Sampling. *FOCS*, 1994.

[5] Daniel M. Kane, Jelani Nelson. Sparser Johnson-Lindenstrauss Transforms. *J. ACM*, 61(1), 2014.

[6] Mikkel Thorup, Yin Zhang Tabulation based 4-universal hashing with applications to second moment estimation. *SODA*, 2004.