University of Wrocław: Algorithms for Big Data (Fall'19) 18/11/2019

Lecture 6: Approximate Matrix Multiplication

Lecturer: *Przemysław Uznański* Scribe: -

Many problems use linear algebra primitive operations. Exact operations are costly to compute. Our goal is to show how approximate versions of those can be computed faster.

1 Matrix multiplication

$A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{n \times p}$. Compute $C = A \times B \in \mathbb{R}^{m \times p}$ such that $c_{ik} = \sum_j a_{ij} b_{jk}$.

Brute-force algorithm $\mathcal{O}(mnp)$. (In case of square matrices its $\mathcal{O}(n^3)$.) A line of algorithmic research improves the complexity of the algorithm: Strassen's algoritm $\mathcal{O}(n^{\log_2 7})$, ..., Le Gall $\mathcal{O}(n^{2.3728639})$. It is not of our concern, since the algorithms (beside Strassen's) are hugely impractical.

Our goal: randomized algorithm that outputs C such that $\|C - A \times B\|$ is *small* with high probability. (Need to define what does $\| \odot \|$ mean.)

1.1 Matrix norms

E.g. Frobenius norm

$$\|A\|_F = \sqrt{\sum_{i,j} a_{i,j}^2}$$

or ℓ_2 norm

$$\|A\|_2 = \sup_{\|x\|=1} |x^T A x|$$

1.2 Sampling approach

We represent A as a column matrix and B as row matrix

$$A = \begin{bmatrix} | & | & & | \\ a_1 & a_2 & \cdots & a_n \\ | & | & & | \end{bmatrix}$$

$$B = \begin{bmatrix} - & b_1 & - \\ - & b_2 & - \\ & \vdots & \\ - & b_n & - \end{bmatrix}$$

We then rewrite

$$A \times B = \sum_i a_i \times b_i$$

where $a_i \times b_i$ is a rank-1 multiplication of vector with vector.

Idea:

- pick some distribution over $i \in [n]$ with probabilities $\{p_i\}$
- let $x = \frac{a_i}{\sqrt{p_i}}$ with $i = [1..n]$ picked according to distribution
- let $y = \frac{b_i}{\sqrt{p_i}}$ with the same i

Then $\mathbb{E}[x \times y] = \sum_i p_i \cdot (\frac{a_i}{\sqrt{p_i}} \times \frac{b_i}{\sqrt{p_i}}) = \sum_i a_i \times b_i = A \times B$. So the approach is to repeat this process t times and average. This fits into the matrix notation:

$$\Pi = \frac{1}{\sqrt{t}} \overbrace{\begin{bmatrix} 0 & \cdots & 0 & \frac{1}{\sqrt{p_{i_1}}} & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \vdots & & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \frac{1}{\sqrt{p_{i_t}}} & 0 & \cdots & 0 \end{bmatrix}}^{n}$$

Algorithm then outputs

$$C = (A \times \Pi^T) \times (\Pi \times B)$$

with runtime $\mathcal{O}(mnt + npt)$. We already shown that $\mathbb{E}[C] = A \times B$. Next we show that

$$\Pr\left[\|C - A \times B\|_F > \varepsilon \|A\|_F \|B\|_F\right] \leq \delta$$

for sufficiently large k and sufficiently chosen $p_1, \ldots, p_t$.

$$\begin{aligned} \Pr\left[\|C - A \times B\|_F > \varepsilon \|A\|_F \|B\|_F\right] &= \Pr\left[\|C - A \times B\|_F^2 > \varepsilon^2 \|A\|_F^2 \|B\|_F^2\right] \\ &\leq \frac{\mathbb{E}\|C - A \times B\|_F^2}{\varepsilon^2 \|A\|_F^2 \|B\|_F^2} \end{aligned}$$

$$\begin{aligned} \mathbb{E}\|C - A \times B\|_F^2 &= \sum_{ik} \mathbb{E}[c_{ik} - (A \times B)_{ik}]^2 \\ &= \sum_{ik} \operatorname{Var}[c_{ik}] \end{aligned}$$

Recall that C is a sum of t rank-1 matrices obtained independently by identical random process, thus $c_{ik} = \sum_{j=1}^{t} c_{ik}^{(j)}$ and

$$\begin{aligned} \operatorname{Var}[c_{ik}] &= \sum_j \operatorname{Var}[c_{ik}^{(j)}] \\ &= t \operatorname{Var}[c_{ik}^{(1)}] \\ &\leq t \mathbb{E}[(c_{ik}^{(1)})^2] \\ &= t \sum_{j=1}^{n} p_j \frac{a_{ij}^2 b_{jk}^2}{t^2 p_j^2} \end{aligned}$$

So

$$\begin{aligned}\mathbb{E}\|C - A \times B\|_F^2 &\leq \frac{1}{t}\sum_j \frac{1}{p_j}\sum_{ik} a_{ij}^2 b_{jk}^2 \\ &= \frac{1}{t}\sum_j \frac{1}{p_j}\left(\sum_i a_{ij}^2\right)\left(\sum_k b_{jk}^2\right) \\ &= \frac{1}{t}\sum_j \frac{\|a_j\|_2^2\|b_j\|_2^2}{p_j}\end{aligned}$$

This is minimized when $p_j \sim \|a_j\|_2\|b_j\|_2$, [EXERCISE] so then (by Cauchy-Schwartz)

$$\begin{aligned}\mathbb{E}\|C - A \times B\|_F^2 &= \frac{1}{t}\left(\sum_j \|a_j\|_2\|b_j\|_2\right)^2 \\ &\leq \frac{1}{t}\left(\sum_j \|a_j\|_2^2\right)\left(\sum_j \|b_j\|_2^2\right) \\ &= \frac{1}{t}\|A\|_F^2\|B\|_F^2\end{aligned}$$

So our bound on probability is:

$$\Pr\left[\|C - A \times B\|_F > \varepsilon\|A\|_F\|B\|_F\right] \leq \frac{\mathbb{E}\|C - A \times B\|_F^2}{\varepsilon^2\|A\|_F^2\|B\|_F^2} \leq \frac{1}{t\varepsilon^2}$$

and it is enough to set $t = \Theta(\varepsilon^{-2})$ for constant success probability. Now we want to amplify probability:

- Run above algorithm $d = \mathcal{O}(\log \delta^{-1})$ times (with 2/3 guarantee of success)
- Obtain $C_1, \ldots, C_d$.
- Pick C_i that is accurate enough.

Standard median trick doesn't work, since we don't have proper ordering for matrices. One idea is to: compute all pairwise differences $\|C_i - C_j\|_F$, and output C_i such that for at least $d/2$ different i' there is $\|C_i - C_{i'}\|_F \leq 2\varepsilon\|A\|_F\|B\|_F$. This follows since:

- if C_i and $C_{i'}$ are good then by triangle inequality their distance is small
- by Chernoff at least 1/2 of all C_i's are good
- if C_i is close to some $C_{i'}$ that is good then C_i is almost good (with distance $2\varepsilon\|A\|_F\|B\|_F$).

1.3 JL approach

Let S be a dimensionality reduction projection from JL lemma. That is, e.g. $S \in \mathbb{R}^{d\times n}$ for $d = \mathcal{O}(\varepsilon^{-2}\log n)$ with entries independent $\mathcal{N}(0,1)$ + normalization. For set X of size n of vectors in $\mathbb{R}^n$ we have $\forall_{v\in X}(1-\varepsilon)\|v\|^2 \leq \|Sv\|^2 \leq (1+\varepsilon)\|v\|^2$ with high probability.

Lemma 1. *For $u, v \in X$ there is $|\langle u, v\rangle - \langle Su, Sv\rangle| = \mathcal{O}(\varepsilon\|u\|\|v\|)$.*

Proof. First assume that $\|u\| = 1$ and $\|v\| = 1$.

$$\langle u, v\rangle = \frac{1}{2}(\|u\|^2 + \|v\|^2 - \|u - v\|^2)$$

and

$$\langle Su, Sv\rangle = \frac{1}{2}(\|Su\|^2 + \|Sv\|^2 - \|S(u - v)\|^2)$$

so

$$\begin{aligned}|\langle u, v\rangle - \langle Su, Sv\rangle| &\leq \frac{1}{2}(\varepsilon\|u\|^2 + \varepsilon\|v\|^2 + \varepsilon\|u - v\|^2)\\ &= \varepsilon(2 + \|u - v\|^2)/2\\ &\leq 2\varepsilon\end{aligned}$$

Now we drop length assumption. Let $u' = u/\|u\|$ and $v' = v/\|v\|$. We have $\langle u, v\rangle = \|u\| \cdot \|v\| \cdot \langle u', v'\rangle$ and $\langle Su, Sv\rangle = \|u\| \cdot \|v\| \cdot \langle Su', Sv'\rangle$.

□

Now our goal is to multiply $A \in \mathbb{R}^{m\times n}$ with $B \in \mathbb{R}^{n\times p}$.

We represent A as a row matrix and B as column matrix

$$A = \begin{bmatrix} - & a_1 & - \\ - & a_2 & - \\ & \vdots & \\ - & a_m & - \end{bmatrix}$$

$$B = \begin{bmatrix} | & | & & | \\ b_1 & b_2 & \cdots & b_p \\ | & | & & | \end{bmatrix}$$

and so

$$(A \times B)_{i,j} = \langle a_i^T, b_j\rangle$$

We claim that $C = (A \times S^T) \times (S \times B)$ is a good approximation to $A \times B$. Indeed,

$$\begin{aligned}\|C - (A \times B)\|_F^2 &= \sum_{i,j} \left(\langle a_i^T, b_j\rangle - \langle Sa_i^T, Sb_j\rangle\right)^2\\ &\leq \sum_{i,j} 4 \cdot \|a_i\|_2^2\|b_j\|_2^2\varepsilon^2\\ &= 4\varepsilon^2 \left(\sum_i \|a_i\|_2^2\right)\left(\|b_j\|_2^2\right)\\ &= 4\varepsilon^2\|A\|_F^2\|B\|_F^2\end{aligned}$$

We also observe that $\mathbb{E}(S^T \times S) = I$, thus $\mathbb{E}(C) = A \times I \times B = A \times B$. The same proof and algorithm as in previous section follows.

In fact, the stronger guarantees can be derived to use the full power of JL dimensionality reduction, so that the "matrix median" is not necessary – but that requires analysis using higher moments and is outside of our scope.

2 Subspace embeddings

JL-type approach is limited by the fact that we can guarantee dimensionality reduction for finite size sets. Consider $A \in \mathbb{R}^{m \times n}$. For $x \in \mathbb{R}^n$, Ax spans a linear space.

Definition 2. *Matrix $S \in \mathbb{R}^{d \times m}$ defines* subspace embedding *if whp*

$$\forall_{x \in \mathbb{R}^n} \|Ax\|^2(1-\varepsilon) \leq \|SAx\|^2 \leq \|Ax\|^2(1+\varepsilon)$$

Our goal is to show that JL-matrix is good for appropriate dimension d. Note, we cannot apply JL-lemma directly, since union-bound fails for sets of infinite size.

Idea: show that it holds for finite size subset of Ax, and then lift the property to whole of Ax. Just picking a basis for linspace is not good enough for our purposes.

First, we note that it is enough to show the property for:

- $\|x\| = 1$ since we can always scale linearly
- A being orthonormal matrix from $\mathbb{R}^{m \times m}$ and x goes over $\mathbb{R}^m$, since we can always pick SVD of $A = U\Sigma V^T$ where U is orthonormal, prove that the property holds for Sx', $x' \in \mathbb{R}^m$, and then actually use it for $x' \in \Sigma V^T x$ for $x \in \mathbb{R}^n$.

Thus we limit ourselves to $S_{m-1} = \{x : \|x\| = 1\}$.

Definition 3. *For $X \subseteq \mathbb{R}^m$ we call $N \subseteq X$ an ε-net if following condition holds:*

$$\forall_{x \in X} \exists_{v \in N} \|x - v\|_2 \leq \varepsilon$$

[EXERCISE] there is ε-net of S_{m-1} of size $(\mathcal{O}(1/\varepsilon))^m$. We take 1/2-net of S_{m-1} and denote it $\mathcal{M}$. It has size $\mathcal{O}(1)^n$.

Lemma 4. *For any $x \in S_{m-1}$ there is a sequence $\alpha_0, \alpha_1, \ldots$ and $y_0, y_1, \ldots$ such that $0 \leq \alpha_i \leq 1/2^i$ and $y_i \in \mathcal{M}$ and*

$$x = \sum_{i=0}^{\infty} \alpha_i y_i.$$

Proof.

- set $x_0 = x$
- find $y_0 \in \mathcal{M}$ such that $\|x_0 - y_0\| \leq 1/2$ (since $\mathcal{M}$ is 1/2-net)
- denote $z_1 = x_0 - y_0$
- set $a_1 = \|z_1\|$ and $x_1 = z_1/\|z_1\| \in S_{n-1}$
- We have $x = y_0 + a_1 \cdot x_1$
- In the limit we have $x = y_0 + a_1 \cdot y_1 + (a_1 a_2) \cdot y_2 + (a_1 a_2 a_3) \cdot y_3 + \ldots$ for $0 \leq a_i \leq 1/2$ and $y_i \in \mathcal{M}$.

□

We set the dimension of S. For S to work, we use JL lemma on the set on the $A\mathcal{M}$. Since we take union-bound over set of size $\exp(\mathcal{O}(m))$, the failure probability needs to be $\delta \sim \exp(-\mathcal{O}(m))$. Thus the dimension needs to be $d = \mathcal{O}(\varepsilon^{-2} \log \delta^{-1}) = \mathcal{O}(m/\varepsilon^2)$.

We have, from orthonormality of A and from JL-lemma guarantee

$$\forall_{y_i, y_j \in B} \langle SAy_i, SAy_j \rangle = \langle Ay_i, Ay_j \rangle \pm \varepsilon = \langle y_i, y_j \rangle \pm \varepsilon$$

Take $x \in \mathbb{R}^m$ such that $\|x\| = 1$.

$$\begin{aligned}
\|SAx\| &= \|\sum_{i=0}^{\infty} SA\alpha_i y_i\| \\
&= \langle \sum_{i=0}^{\infty} SA\alpha_i y_i, \sum_{i=0}^{\infty} SA\alpha_i y_i \rangle \\
&= \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} \alpha_i \alpha_j \langle SAy_i, SAy_j \rangle \\
&= \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} \alpha_i \alpha_j (\langle y_i, y_j \rangle \pm \varepsilon) \\
&= \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} \alpha_i \alpha_j \langle y_i, y_j \rangle \pm \varepsilon \cdot \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} \alpha_i \alpha_j \\
&= \langle \sum_{i=0}^{\infty} \alpha_i y_i, \sum_{j=0}^{\infty} \alpha_j y_j \rangle \pm \varepsilon \left(\sum_i \alpha_i\right)\left(\sum_j \alpha_j\right) \\
&= \langle x, x \rangle \pm \mathcal{O}(\varepsilon) \\
&= \|x\| \pm \mathcal{O}(\varepsilon) \\
&= 1 \pm \mathcal{O}(\varepsilon)
\end{aligned}$$

and this finishes the proof.