Lecture 14 Matrix Sketches

I. The frequent directions problem

$\boxed{\text{Input:}}$ n rows of a large matrix $A \in \mathbb{R}^{n\times d}$, one after the other, i.e., in a streaming fashion.

$\boxed{\text{Output:}}$ a sketch $B \in \mathbb{R}^{k\times d}$, where $k \ll n$, satisfying $A^TA \simeq B^TB$. More precisely

$\forall x, \|x\|=1, \quad 0 \leq \|Ax\|^2 - \|Bx\|^2 \leq 2\|A\|_F^2/k$

$$\boxed{\text{Or, } \quad B^TB \prec A^TA \text{ and } \|A^TA - B^TB\| \leq 2\|A\|_F^2/k}$$

Goal: small space

Motivation: save the running time for SVD/PCA; storage saving

Recall Frequent items: n items, each from $[N]$, find all items appearing $> \frac{n}{k}$ times.

Previous algorithm (e.g. Misra-Gries alg) gives us $\hat{f}_j$ s.t. $|\hat{f}_j - f_j| \leq \frac{n}{k}$

f_j is the frequency of j

An equivalent view: n rows, each from $\{\vec{e}_1, \vec{e}_2, \cdots, \vec{e}_N\}$

i.e. the standard orthonormal basis of N-dim.

each row $\vec{e}_j$ has frequency, $f_j = \|Ae_j\|^2$, $n = \|A\|_F^2$

A good sketch B has $\hat{f}_j = \|Be_j\|^2$ and

$$\|Ae_j\|^2 - \|Be_j\|^2 \leq \frac{\|A\|_F^2}{k}$$

Here Frequent directions: n rows, each row $\in \mathbb{R}^d$, a good sketch satisfying
$$0 \le \|Ax\|^2 - \|Bx\|^2 \le 2\|A\|_F^2/k$$

(Similar to frequent items alg, which finds all items that appear more than εn times, if we choose $\ell > \frac{1}{\varepsilon}$.

Here, the frequent directions alg can be used to uncover any unit vector (direction) with $\|Ax\|^2 \ge \varepsilon \|A\|^2$, if $\ell > \frac{2r}{\varepsilon}$. (where $r = \frac{\|A\|_F^2}{\|A\|^2}$ numeric rank of A)

$k \ll d \ll n$

Recall: SVD of $B \in \mathbb{R}^{k \times d}$, $B = UDV^T$

$U \in \mathbb{R}^{k \times k}$, $D = \mathbb{R}^{k \times k}$, $V = \mathbb{R}^{d \times k}$

$\sigma_1 \ge \sigma_2 \ge \cdots \ge \sigma_\ell \ge 0$

$U^T U = V^T V = I_k$

$$\underset{U}{\begin{pmatrix} & \end{pmatrix}_{k \times k}} \underset{D}{\begin{pmatrix} & \end{pmatrix}_{k \times k}} \underset{V^T}{\begin{pmatrix} & \end{pmatrix}_{k \times d}}$$

II The algorithm

① Initialize $B \in \mathbb{R}^{k \times d}$ to be all zeros matrix

② For each row A_i ($i \in [n]$) do:

Insert A_i into a zero valued row of B

If B has no zero valued rows, then

compute the SVD of B: $B = UDV^T$

let $C = DV^T$ (only for analysis)

Let $\delta = \sigma_{k/2}^2$ (σ_i: the i-th singular value of B)

let $\tilde{D} = \sqrt{\max(D^2 - I_k \cdot \delta, 0)}$

(I_k: $k \times k$ identity matrix)

let $B \leftarrow \tilde{D} V^T$ ($\geq \frac{k}{2}$ rows of B are all zero)

③ Return B

Note: all-zero valued rows always exist.

Inspired by Misra-Gries algorithm for frequent items

Claim: If B is the result of applying the algorithm to A, then $0 \preceq B^T B \preceq A^T A$

Proof: $B^T B \succeq 0$ easy!

For the second inequality, it suffices to prove $\forall x$, $x^T(A^T A - B^T B)x \geq 0$, i.e.

$$\|Ax\|^2 - \|Bx\|^2 \geq 0$$

Let B^i, C^i be the values of B and C after the main loop has been executed i times.

Note $B^0 = 0$, $B^n = B$ (i.e. the returned sketch)

Then $$\|Ax\|^2 - \|Bx\|^2 = \sum_{i=1}^{n} \left[\langle A_i, x \rangle^2 + \|B^{i-1}x\|^2 - \|B^i x\|^2 \right]$$

$$= \sum_{i=1}^{n} \left[\|C^i x\|^2 - \|B^i x\|^2 \right] \quad (*)$$

Why $\langle A_i, x \rangle^2 + \|B^{i-1}x\|^2 = \|C^i x\|^2$

① A_i is inserted in a zero valued row of B^{i-1}

② C^i is an isometric left rotation of a matrix containing the rows of B^{i-1} and A_i.

$\left(C^i = U^T B_1^i, \text{ where } B_1^i = B^{i-1} + \begin{pmatrix} 0 \\ A_i \\ 0 \\ \vdots \end{pmatrix} \right)$

Now by the shrinking step,

$$(C^i)^T(C^i) \succeq (B^i)^T(B^i)$$

$$\begin{pmatrix} \sigma_1 & & \\ & \sigma_2 & \\ & & \ddots \sigma_r \end{pmatrix} \begin{pmatrix} \ \end{pmatrix}$$

$D \quad V^T$

i.e.,

$$\|C^i x\|^2 - \|B^i x\|^2 \geq 0$$

$$\begin{pmatrix} \sqrt{\sigma_1^2 - \sigma^2} & & & \\ & \sqrt{\sigma_{\frac{k}{2}-1}^2 - \sigma^2} & & \\ & & 0 & \\ & & & \ddots 0 \end{pmatrix} V^T$$

B

Thus $(*) \geq 0$ $\square$

$$\boxed{\text{Claim 2: } \|A^TA - B^TB\| \leq 2\|A\|_F^2 / k}$$

Proof: let $\delta^{(i)}$ be the δ value a time i

If the algorithm does not enter "if" section at step i, then $\delta^{(i)} = 0$.

Let B^i, C^i, V^i be the values of B, C, V after the main loop is executed i times.

Let x be the unit eigenvector corresponding to the largest eigenvalue of $A^TA - B^TB$,

Then $\|A^TA - B^TB\| = \|Ax\|^2 - \|Bx\|^2$

(as $\|A^TA - B^TB\| = \|(A^TA - B^TB)x\|$, $= \|\lambda x\| = \lambda$ $\quad$ $x^T(A^TA - B^TB)x = x^T\lambda x = \lambda$)

[In general, for any X, $\|X\|_2 = \sqrt{\lambda_{max}(X^TX)} = \sigma_{max}(X)$,
also $\|X^TX\|_2 = \|XX^T\| = \|X\|_2^2$, as
$\|X^TX\|_2 = \sigma_{max}(X^TX) = \sigma_{max}(X)^2 = \|X\|_2^2$

$$
\begin{aligned}
\text{Thus } \|A^TA - B^TB\| &= \sum_{i=1}^{n} \left[\|C^i x\|^2 - \|B^i x\|^2 \right] \\
&\leq \sum_{i=1}^{n} \|(C^i)^T(C^i) - (B^i)^T(B^i)\| \\
&= \sum_{i=1}^{n} \|(D^i)^2 - (\tilde{D}^i)^2\| \\
&= \sum_{i=1}^{n} \delta^{(i)}
\end{aligned}
$$

On the other hand:

$$
\begin{aligned}
\|B^n\|_F^2 &= \sum_{i=1}^{n} \left[\|B^i\|_F^2 - \|B^{i-1}\|_F^2 \right] \\
&= \sum_{i=1}^{n} \left[(\|C^i\|_F^2 - \|B^{i-1}\|_F^2) - (\|C^i\|_F^2 - \|B^i\|_F^2) \right] \\
&= \sum_{i=1}^{n} \left(\|A_i\|^2 - \mathrm{Tr}\left[(C^i)^T(C^i) - (B^i)^T(B^i) \right] \right) \\
&= \|A\|_F^2 - \sum_{i=1}^{n} \mathrm{Tr}\left[(D^i)^2 - (\tilde{D}^i)^2 \right] \\
&\leq \|A\|_F^2 - (k/2) \sum_{i=1}^{n} \delta^{(i)}
\end{aligned}
$$

The matrix $(D^i)^2 - (\tilde{D}^i)^2$ contains k non-negative elements on its diagonal, at least half of which are $= \delta^{(2)}$

Thus, $\sum_{i=1}^{n} \delta^{(i)} \leq 2(\|A\|_F^2 - \|B\|_F^2)/k$

$$\Rightarrow \|A^TA - B^TB\| \leq 2(\|A\|_F^2 - \|B\|_F^2)/k$$
$$\leq 2\|A\|_F^2/k \qquad \square$$

Space: $O(d \cdot k)$ words of space

Total time: $T_{SVD}(k, d)$, SVD time for a $k \times d$ matrix
$= O(d \cdot k^2)$ if $k \leq d$.

times that SVD is computed $O(\frac{n}{k})$
since the shrinking step nullifies $\geq \frac{k}{2}$ rows in B

total time $O(\frac{n}{k} \cdot T_{SVD}(k, d)) = O(n \cdot dk)$

The above alg is suitable if n is extremely large (e.g. $n = 10^8$), and d is also uncomfortably large (e.g. $d = 10^5$). The above finds an approximation with $k \ll d \ll n$. (e.g. $k = 10$)

If d is already very small (e.g. $d=100$), then one can directly compute its covariance matrix B in $O(d^2)$ words of space.

① Initialize $B \in \mathbb{R}^{d \times d}$ to be a all zero matrix

② for rows $A_i \in A$ do
   update $B = B + A_i^T A_i$

③ Return B

At the end $B = A^T A = \sum_{i=1}^{n} A_i^T A_i$

It takes $O(d^2)$ space. $O(nd^2)$ time

III Other matrix sketches

Approximate A upto the accuracy of $A_k = \sum_{i=1}^{k} \sigma_i u_i v_i$

① Row sampling: (Running time is almost proportional to $nnz(A)$)

Ideas Ⓐ For each row $A_i \in A$, set $w_i = \|A_i\|_2^2$, then select $l = (k/\varepsilon)^2 \cdot \log(1/\delta)$ rows of A, each proportional to w_i. Then define

$$R = \begin{pmatrix} x_1 \\ \vdots \\ x_l \end{pmatrix}, \quad \text{each } x_i \text{ is a sampled row}$$

1 orthogonalization: Let $\Pi_R = R^T(RR^T)^{-1}R$ be the projection matrix of R.

Then $\|A - A\Pi_R\|_F \leq \|A - A_k\|_F + \varepsilon\|A\|_F$

w.p. $\geq 1-\delta$.

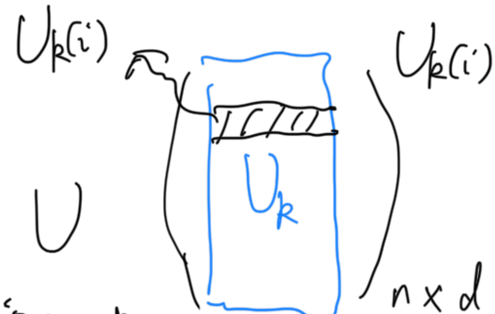
The sketch is $A\Pi_R$.

2 Sample each row proportional to leverage scores.

leverage score of each row A_i is

$s_i = \|U_k(i)\|^2$, where $A = UDV^T$

U_k the top k left singular vector



$U_k(i)$ the ith row of U_k

Then orthogonalize R as before, and

$\|A - A\Pi_R\|_F \leq (1+\varepsilon)\|A_k\|_F$

3 Random projection:

Ideas: Choose a random projection matrix

$S \in \mathbb{R}^{\ell \times n}$ $\rightarrow$ each entry i.i.d.

$\sim \sqrt{n/\ell} \cdot N(0,1)$

Choose $\ell \simeq k/\varepsilon$

Let $B = SA$, $V \in \mathbb{R}^{d \times d}$ the right singular vectors

$$\|A - [AV]_k V^T\|_F \leq (1+\varepsilon)\, \|A - A_k\|_F$$

Also choose $\ell \simeq d/\varepsilon^2$, oblivious subspace embedding

$$(1-\varepsilon) \leq \frac{\|Ax\|}{\|Bx\|} \leq (1+x) \quad \text{for any } x \in \mathbb{R}^d$$

④ Count Sketch hashing: ($nnz(A)$ running time)

Choose the previous $S \in \mathbb{R}^{\ell \times n}$,

such that each column has exactly one randomly chosen entry that is either 1 or -1, all the other entries are 0