School of Computer Science and Technology
University of Science and Technology of China

Marked Exercises for Algorithms for Big Data 2022 Spring

Due 27 March 2022 at 23:59

Exercise 1 *10 points*
Let $\sum_{i=1}^{r} \sigma_i u_i v_i^T$ be the SVD of A, where $A \in \mathbb{R}^{n\times d}$. Show that $|u_1^T A| = \sigma_1$ and $|u_1^T A| = \max_{\|u\|=1} \|u^T A\|$, where $\|x\| = \sqrt{\sum_{i=1}^{d} x_i^2}$ for a vector $x \in \mathbb{R}^d$.

Proof. Since the left singular vectors are pairwise orthogonal, we have

$$u_1^T A = u_1^T (\sum_{i=1}^{r} \sigma_i u_i v_i^T) = \sigma_1 v_1^T \tag{1}$$

provided that v_1 is a unit vector, therefore

$$|u_1^T A| = \|u_1^T A\| = \|\sigma_1 v_1^T\| = \sigma_1 \|v_1^T\| = \sigma_1 \|v_1\| = \sigma_1 \tag{2}$$

Moreover, for any $u \in \mathbb{R}^d$, write it as $u = \sum_{j=1}^{r} \alpha_j u_j$, then

$$u^T A = (\sum_{j=1}^{r} \alpha_j u_j^T)(\sum_{i=1}^{r} \sigma_i u_i v_i^T) \tag{3}$$

$$= \sum_{i=1}^{r} (\alpha_i \sigma_i v i^T) \tag{4}$$

then

$$|u^T A| = \|u^T A\| = \|\sum_{i=1}^{r} (\alpha_i \sigma_i v i^T)\| \tag{5}$$

$$= \sqrt{\sum_{i=1}^{r} \alpha_i^2 \sigma_i^2} \tag{6}$$

Let u^* be the unit vector maximizing the above. Since u^* is unit, i.e.

$$\|u^*\|^2 = \sum_{i=1}^{r} \alpha_i^2 = 1 \tag{7}$$

such an optimal u^* satisfies that

$$\alpha_i = \begin{cases} 1 & i = 1, \\ 0 & \text{otherwise} \end{cases}$$

i.e.

$$u^* = u_1 \tag{8}$$

We concluded that

$$|u_1^T A| = |u_*^T A| = \max_{\|u\|=1} \|u^T A\| \tag{9}$$

Combining (2) and (9), Q.E.D. □

Exercise 2 *20 points*
Let $\sum_{i=1}^{r} \sigma_i u_i v_i^T$ be the SVD of a rank r matrix A. Let $A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T$ be a rank k-approximation to A for some $k < r$. Express the following quantities in terms of the singular values $\{\sigma_i, 1 \leq i \leq r\}$.

(a) $\|A_k\|_F^2$

(b) $\|A_k\|_2^2$

(c) $\|A - A_k\|_F^2$

(d) $\|A - A_k\|_2^2$

Solution. Since the rows of A_k are projections of the rows of A onto the subspace $V_k = \text{span}\{v_1, ..., v_k\}$, we have

$$\|A_k\|_F^2 = \sum_{i=1}^{n} \text{proj}^2(a_i, V_k) = \sum_{i=1}^{n}\sum_{j=1}^{k}(a_i \cdot v_j)^2 = \sum_{j=1}^{k}\sum_{i=1}^{n}(a_i \cdot v_j)^2 \tag{10}$$

$$= \sum_{j=1}^{k} \sigma_i^2 \tag{11}$$

Similarly, we have

$$\|A - A_k\|_F^2 = \sum_{i=1}^{n} \text{dist}^2(a_i, V_k) = \sum_{i=1}^{n}(\|a_i\|^2 - \text{proj}^2(a_i, V_k)) = \|A\|_F^2 - \|A_k\|_F^2 \tag{12}$$

$$= \sum_{j=k+1}^{r} \sigma_i^2 \tag{13}$$

With respect to 2-norm, for any $v \in \mathbb{R}^d$, write it as $v = \sum_{j=1}^{r} \alpha_j v_j$, then

$$A_k v = (\sum_{i=1}^{k} \sigma_i u_i v_i^T)(\sum_{j=1}^{r} \alpha_j v_j) \tag{14}$$

$$= \sum_{i=1}^{k} \alpha_i \sigma_i u_i \tag{15}$$

so

$$\|A_k v\| = norm \sum_{i=1}^{k} \alpha_i \sigma_i u_i = \sqrt{\sum_{i=1}^{k} \alpha_i^2 \sigma_i^2} \tag{16}$$

Let v^* be the unit vector maximizing the above. Since v^* is unit, i.e.

$$\|v^*\|^2 = \sum_{i=1}^{r} \alpha_i^2 = 1 \tag{17}$$

such an optimal v^* satisfies that

$$\alpha_i = \begin{cases} 1 & i = 1, \\ 0 & \text{otherwise} \end{cases}$$

then

$$\|A_k\|_2 = \max_{\|v\|=1} \|A_k v\| = \|A_k v_*\| = \sigma_1 \tag{18}$$

i.e.

$$\|A_k\|_2^2 = \sigma_1^2 \tag{19}$$

Similarly, we have

$$(A - A_k)v = (\sum_{i=k+1}^{r} \sigma_i u_i v_i^T)(\sum_{j=1}^{r} \alpha_j v_j) \tag{20}$$

$$= \sum_{i=k+1}^{r} \alpha_i \sigma_i u_i \tag{21}$$

so

$$\|(A - A_k)v\| = \| \sum_{i=k+1}^{r} \alpha_i \sigma_i u_i \| = \sqrt{\sum_{i=k+1}^{r} \alpha_i^2 \sigma_i^2} \tag{22}$$

an optimal unit vector v^* satisfies that

$$\alpha_i = \begin{cases} 1 & i = k+1, \\ 0 & \text{otherwise} \end{cases}$$

then

$$\|A - A_k\|_2 = \max_{\|v\|=1} \|(A - A_k)v\| = \|(A - A_k)v_*\| = \sigma_{k+1} \tag{23}$$

i.e.

$$\|A - A_k\|_2^2 = \sigma_{k+1}^2 \tag{24}$$

Exercise 3 *15 points*
Let $k < d$. Let $U \in \mathbb{R}^{d \times k}$ be a random matrix such that its (i, j)-th entry is denoted as u_{ij}, where $\{u_{ij}\}$ are independent random variables such that

$$u_{ij} = \begin{cases} 1 & \text{with probability } \frac{1}{2}, \\ -1 & \text{with probability } \frac{1}{2} \end{cases}$$

Now we use matrix U as a random projection matrix. That is, for a (row) vector $a \in \mathbb{R}^d$, we map it to

$$f(a) = \frac{1}{\sqrt{k}} aU$$

For each j such that $1 \le j \le k$, define $b_j = [f(a)]_j$, i.e., b_j is the j-th entry of $f(a)$.

- What is the expectation $\mathrm{E}[b_j]$?
- What is $\mathrm{E}[b_j^2]$?
- What is $\mathrm{E}[\|f(a)\|^2]$?

Solution. Let $U = \{u_1, u_2, ..., u_k\}$, each $u_i, 1 \le i \le k$ is a column vector of matrix U. Then

$$b_j = \frac{1}{\sqrt{k}} a u_j = \frac{1}{\sqrt{k}} \sum_{i=1}^{d} a_i u_{ij} \tag{25}$$

So

$$\mathrm{E}[b_j] = \mathrm{E}[\frac{1}{\sqrt{k}} \sum_{i=1}^{d} a_i u_{ij}] = \frac{1}{\sqrt{k}} \sum_{i=1}^{d} a_i \mathrm{E}[u_{ij}] = 0 \tag{26}$$

Similarly, we can get the variance of b_j, i.e.

$$\mathrm{Var}[b_j] = \mathrm{Var}[\frac{1}{\sqrt{k}}\sum_{i=1}^{d} a_i u_{ij}] = \frac{1}{k}\sum_{i=1}^{d} a_i^2 \mathrm{Var}[u_{ij}] \tag{27}$$

$$= \frac{1}{k}\sum_{i=1}^{d} a_i^2 \tag{28}$$

As $\mathrm{Var}[x] = \mathrm{E}[x^2] - \mathrm{E}^2[x]$, we have

$$\mathrm{E}[b_j^2] = Var[b_j] + \mathrm{E}^2[b_j] = \frac{1}{k}\sum_{i=1}^{d} a_i^2 = \frac{1}{k}\|a\|^2 \tag{29}$$

Moreover, since

$$\|f(a)\|^2 = \sum_{j=1}^{k} b_j^2 \tag{30}$$

we can get that

$$\mathrm{E}[\|f(a)\|^2] = \mathrm{E}[\sum_{j=1}^{k} b_j^2] = \sum_{j=1}^{k} \mathrm{E}[b_j^2] = \frac{1}{k}\sum_{j=1}^{k}\|a\|^2 = \|a\|^2 \tag{31}$$

Exercise 4 *15 points*
In the class, we have seen an algorithm, denoted by $\mathcal{A}$, for the (c, r)-ANN problem with success probability at least 0.6. That is, upon a queried vertex x such that there exists a point a^* in the set $\mathcal{P}$ with $d(x, a^*) \leq r$, the algorithm $\mathcal{A}$ outputs some $a \in \mathcal{P}$ with $d(x, a) \leq c \cdot r$ with probability at least 0.6.
Let $\delta \in (0, 1)$. Using the above $\mathcal{A}$ as a subroutine, give a new algorithm $\mathcal{B}$ with success probability at least $1 - \delta$. That is, for the above query vertex x, the algorithm $\mathcal{B}$ outputs some $a \in \mathcal{P}$ with $d(x, a) \leq c \cdot r$ with probability at least $1 - \delta$. Your algorithm should use as little query time as possible. Explain the correctness of your algorithm and state its query time, assuming the query time of $\mathcal{A}$ is $T_{\mathcal{A}}$.

Solution. Algorithm $\mathcal{B}$:

(a) Independently run $t = \lceil \frac{25}{18} \ln \frac{1}{\delta} \rceil$ passes of algorithm $\mathcal{A}$, and return the first a_i that is not FAIL

(b) If no such a_i is found, output FAIL

Proof. For each $i \leq t$, let

$$Y_i = \begin{cases} 1 & \text{if the i-th pass of algorithm } \mathcal{A} \text{ succeeds,} \\ 0 & \text{otherwise} \end{cases}$$

Let $Y = \sum_{i=1}^{t} Y_i$ be the number of passes of algorithm $\mathcal{A}$ that succeeds. Note that $\Pr[Y_i = 1] \geq 0.6$, then

$$\mathrm{E}[Y] \geq 0.6t \tag{32}$$

Therefore, by Chernoff-Hoeffding bound, we have

$$\Pr[Y = 0] < \Pr[Y - \mathrm{E}[Y] < -0.6t] \leq e^{-2 \cdot 0.6^2 \cdot t} \leq \delta \tag{33}$$

if $t \geq \frac{25}{18} \ln \frac{1}{\delta}$. □

The query time of algorithm $\mathcal{B}$ is $O(\log \frac{1}{\delta} T_{\mathcal{A}})$.

Exercise 5 *20 points*
Let $\alpha \in (0, 1]$. Suppose we change the (basic) Morris algorithm to the following:

(a) Initialize $X \leftarrow 0$

(b) For each update, increment X by 1 with probability $\frac{1}{(1+\alpha)^X}$

(c) For a query, output $\tilde{n} = \frac{(1+\alpha)^X - 1}{\alpha}$.

Let X_n denote X in the above algorithm after n updates. Let $\tilde{n} = \frac{(1+\alpha)^{X_n} - 1}{\alpha}$.

- Calculate $\mathrm{E}[\tilde{n}]$ and upper bound $\mathrm{Var}[\tilde{n}]$.
- Let $\epsilon, \delta \in (0, 1)$. Based upon the above algorithm, give a new algorithm such that with probability at least $1 - \delta$, it outputs an estimator $\tilde{n}$ such that $|\tilde{n} - n| \le \epsilon n$. Explain the correctness and the space complexity (i.e., the number of used bits) of your algorithm. It suffices to give an algorithm with space complexity that is a polynomial function of $1/\delta$.

Solution. We can prove that $\mathrm{E}[\tilde{n}] = n$ by induction.

Proof. If $n = 0$, then $X_n = 0$, hence

$$\tilde{n} = \frac{(1+\alpha)^{X_n} - 1}{\alpha} = 0 \tag{34}$$

Assume the proposition holds when $n \le k$, then when $n = k + 1$, we have

$$\mathrm{E}[\tilde{n}] = \mathrm{E}[\frac{(1+\alpha)^{X_{k+1}} - 1}{\alpha}] = \frac{1}{\alpha}\mathrm{E}[(1+\alpha)^{X_{k+1}}] - \frac{1}{\alpha} \tag{35}$$

Because

$$\begin{aligned}
\mathrm{E}[(1+\alpha)^{X_{k+1}}] &= \sum_{j=0}^{\infty} \Pr[X_k = j]\mathrm{E}[(1+\alpha)^{X_{k+1}} | X_k = j] && (36)\\
&= \sum_{j=0}^{\infty} \Pr[X_k = j]\{(1 - \frac{1}{(1+\alpha)^j}) \cdot (1+\alpha)^j + \frac{1}{(1+\alpha)^j} \cdot (1+\alpha)^{j+1}\} && (37)\\
&= \sum_{j=0}^{\infty} \Pr[X_k = j](1+\alpha)^j + \sum_{j=0}^{\infty} \Pr[X_k = j]\alpha && (38)\\
&= \mathrm{E}[(1+\alpha)^{X_k}] + \alpha && (39)
\end{aligned}$$

therefore

$$\begin{aligned}
E[\tilde{n}] &= \mathrm{E}[\frac{(1+\alpha)^{X_{k+1}} - 1}{\alpha}] = \frac{1}{\alpha}(\mathrm{E}[(1+\alpha)^{X_k}] + \alpha) - \frac{1}{\alpha} && (40)\\
&= \mathrm{E}[\frac{(1+\alpha)^{X_k} - 1}{\alpha}] + 1 && (41)\\
&= k + 1 && (42)
\end{aligned}$$

Q.E.D. □

By similar calculations, we have

$$\mathrm{E}[(1+a)^{2X_n}] = 1 + (\alpha^2 + 2\alpha)(\frac{\alpha}{2}n^2 + (1 - \frac{\alpha}{2})n) \tag{43}$$

then

$$\begin{aligned}
\mathrm{Var}[\tilde{n}] &= \frac{1}{\alpha^2}\mathrm{Var}[(1+\alpha)^{X_n}] && (44)\\
&= \frac{1}{\alpha^2}\{\mathrm{E}(1+\alpha)^{2X_n} - (n\alpha + 1)^2\} && (45)\\
&= \frac{1}{\alpha^2}\{1 + (\alpha^2 + 2\alpha)(\frac{\alpha}{2}n^2 + (1 - \frac{\alpha}{2})n) - (n\alpha + 1)^2\} && (46)\\
&= \frac{\alpha}{2}n^2 - \frac{\alpha}{2}n < \frac{\alpha}{2}n^2 && (47)
\end{aligned}$$

Thus, by Chebyshev's inequality, we have

$$\Pr[|\tilde{n} - n| > \epsilon n] \le \frac{\mathrm{Var}[\tilde{n}]}{\epsilon^2 n^2} < \frac{\alpha}{2\epsilon^2} \tag{48}$$

Therefore, we can get the following algorithm based on the above discussion such that with probability at least $1 - \delta$, it outputs an estimator $\tilde{n}$ such that $|\tilde{n} - n| \le \epsilon n$:

(a) Initialize $X \leftarrow 0$ and $\alpha \leftarrow 2\epsilon^2\delta$

(b) For each update, increment X by 1 with probability $\frac{1}{(1+\alpha)^X}$

(c) For a query, output $\tilde{n} = \frac{(1+\alpha)^X - 1}{\alpha}$.

If $\alpha = 2\epsilon^2\delta$,then

$$\Pr[|\tilde{n} - n| > \epsilon n] < \frac{\alpha}{2\epsilon^2} = \delta \tag{49}$$

i.e.

$$\Pr[|\tilde{n} - n| \le \epsilon n] \ge 1 - \delta \tag{50}$$

Hence the algorithm satisfies the criterion. The space complexity of the algorithm is $O(\log\log\frac{n}{\delta})$.
Note: The algorithm can also be improved by choosing the median of the means of basic estimations (which is from the Morris algorithm with $\alpha = 2\epsilon^2\delta$).

Exercise 6 *20 points*
Consider a stream of m integers $a_1, a_2, \ldots, a_m$ such that each $a_i \in [n] = \{1, 2, \ldots, n\}$. We would like to estimate the *median* of these numbers using small space. Formally, let $S = \{a_1, a_2, \ldots, a_m\}$, and define $\mathrm{rank}(b) = |\{a \in S : a \le b\}|$. For simplicity, suppose elements in S are distinct, and m is known to the algorithm. Given $\varepsilon, \delta \in (0, 1)$, our goal is to find a number b such that

$$\Pr[|\mathrm{rank}(b) - \frac{m}{2}| > \varepsilon m] < \delta. \tag{51}$$

Consider the following algorithm:

- Maintain t uniform samples from S (e.g., by using Reservoir sampling)
- Output the median of these t samples

Choose the smallest possible t so that inequality (51) holds. Give an explanation of the correctness of the resulting algorithm and state its space complexity.

Hint: You can partition S into 3 groups: $S_L = \{a \in S : \mathrm{rank}(a) \le m/2 - \varepsilon m\}$, $S_M = \{a \in S : m/2 - \varepsilon m \le \mathrm{rank}(a) \le m/2 + \varepsilon m\}$, and $S_H = \{a \in S : \mathrm{rank}(a) \ge m/2 + \varepsilon m\}$. Note that if less than $t/2$ elements from both S_L and S_H are present in the sample, then the median of the samples is a "good" estimator.

Solution. For each $i \le t$, let

$$Y_i = \begin{cases} 1 & \text{if the i-th copy of uniform sampling is in } S_M, \\ 0 & \text{otherwise} \end{cases}$$

Since the algorithm outputs a uniform sample independently, we have

$$\Pr[Y_i = 1] = \frac{2\epsilon}{m} \tag{52}$$

Hence

$$\mathrm{E}[Y] = \frac{2\epsilon}{m} t \tag{53}$$

Let the output of the algorithm be $\tilde{n}$, by Chernoff-Hoeffding bound, we concluded that

$$\begin{aligned}
\Pr[\tilde{n} \text{ is bad}] &\leq \Pr[Y < \frac{t}{2}] = \Pr[Y - \mathrm{E}[Y] < -\frac{4\epsilon - m}{2m} t] &(54)\\
&\leq e^{-2(\frac{4\epsilon - m}{2m})^2 t} &(55)\\
&\leq \delta &(56)
\end{aligned}$$

if

$$t \geq \frac{2m^2}{(4\epsilon - m)^2} \ln \frac{1}{\delta} \tag{57}$$

Therefore, we can choose a smallest possible t as

$$t = \lceil \frac{2m^2}{(4\epsilon - m)^2} \ln \frac{1}{\delta} \rceil \tag{58}$$

In this case, inequality (51) holds. The algorithm's space complexity is $O(\log \frac{1}{\delta} (\log n + \log m))$.