中国科学技术大学
计算机科学与技术学院

《大数据算法》作业
2022 年春

截止日期：2022 年 3 月 27 日 23:59

Exercise 1 *10 分*

令 $\sum_{i=1}^{r}\sigma_i u_i v_i^T$ 为 A 的 SVD 分解，其中 $A\in\mathbb{R}^{n\times d}$。证明 $|u_1^T A|=\sigma_1$ 和 $|u_1^T A|=\max_{\|u\|=1}\|u^T A\|$。（对于向量 $x\in\mathbb{R}^d$，$\|x\|=\sqrt{\sum_{i=1}^{d}x_i^2}$。）

Exercise 2 *20 分*

令 $\sum_{i=1}^{r}\sigma_i u_i v_i^T$ 为一个秩为 r 的矩阵 A 的 SVD 分解。对于某个 $k<r$，$A_k=\sum_{i=1}^{k}\sigma_i u_i v_i^T$ 是矩阵 A 的一个秩为 k 的近似。用奇异值 $\{\sigma_i, 1\le i\le r\}$ 表达以下几个量。

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

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

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

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

Exercise 3 *15 分*

假设 $k<d$。假设 $U\in\mathbb{R}^{d\times k}$ 是一个随机矩阵，其第 (i,j) 个元素记作 u_{ij}。这里 $\{u_{ij}\}$ 是独立的随机变量，满足：

$$u_{ij}=\begin{cases}1 & \text{以 } \frac{1}{2} \text{ 的概率,}\\ -1 & \text{以 } \frac{1}{2} \text{ 的概率}\end{cases}$$

我们使用矩阵 U 作为一个随机投影矩阵。也就是说，对于一个行向量 $a\in\mathbb{R}^d$，我们把它映射到

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

对于 $1\le j\le k$ 中的每个 j，定义 $b_j=[f(a)]_j$，即 b_j 是 $f(a)$ 的第 j 个元素。

- 计算 $\mathrm{E}[b_j]$。
- 计算 $\mathrm{E}[b_j^2]$。
- 计算 $\mathrm{E}[\|f(a)\|^2]$。

Exercise 4 *15 分*

在本课程中，我们学习了一个成功概率至少为 0.6 的解决 (c,r)-ANN 问题的算法，记作 $\mathcal{A}$。也就是说，针对

一个查询点 x，如果数据集合 $\mathcal{P}$ 中存在一个点 a^* 满足 $d(x,a^*) \le r$，那么算法 $\mathcal{A}$ 将会以至少 0.6 的概率输出某个点 $a \in \mathcal{P}$，其满足 $d(x,a) \le c \cdot r$。
假设 $\delta \in (0,1)$。使用上述算法 $\mathcal{A}$ 作为一个子程序，给出一个成功概率至少为 $1-\delta$ 的新算法 $\mathcal{B}$。也就是说，对于上述查询点 x，算法 $\mathcal{B}$ 将会以至少 $1-\delta$ 的概率输出某个点 $a \in \mathcal{P}$，其满足 $d(x,a) \le c \cdot r$。你的算法应该用尽可能少的查询时间。假设 $\mathcal{A}$ 的查询时间是 $T_{\mathcal{A}}$。解释你算法的正确性，并且表达其查询时间。

Exercise 5 *20 分*
假设 $\alpha \in (0,1]$。假如我们将（基本的）Morris 算法修改如下：

(a) 初始化 $X \leftarrow 0$

(b) 对于每次更新, 以 $\frac{1}{(1+\alpha)^X}$ 的概率使 X 加 1

(c) 对于查询, 输出 $\tilde{n} = \frac{(1+\alpha)^X - 1}{\alpha}$.

记 X_n 为上述算法中 n 次更新以后的 X。令 $\tilde{n} = \frac{(1+\alpha)^{X_n}-1}{\alpha}$。

- 计算 $\mathrm{E}[\tilde{n}]$ 并且给出 $\mathrm{Var}[\tilde{n}]$ 的一个上界。
- 假设 $\epsilon, \delta \in (0,1)$。基于以上算法，给出一个新算法，使得新算法以至少 $1-\delta$ 的概率输出一个估计 $\tilde{n}$，满足 $|\tilde{n} - n| \le \epsilon n$。解释你的算法的正确性与空间复杂度（即算法使用的比特数）。（你的算法只需要具有关于 $1/\delta$ 的多项式量级的空间复杂度即可。）

Exercise 6 *20 分*
考虑一个数据流，其中包含 m 个整数 $a_1, a_2, \ldots, a_m$。这里 $a_i \in [n] = \{1, 2, \ldots, n\}$。我们想要用较小的空间估计这些数据的中位数。严格来说，设 $S = \{a_1, a_2, \ldots, a_m\}$，定义 $\mathrm{rank}(b) = |\{a \in S : a \le b\}|$。简单起见，假设 S 中的元素各不相同，并且 m 对于算法是已知的。给定 $\varepsilon, \delta \in (0,1)$, 我们的目标是找到一个数 b，使得

$$\Pr[|\mathrm{rank}(b) - \frac{m}{2}| > \varepsilon m] < \delta. \tag{1}$$

考虑如下算法：

(a) 从 S 中，保存 t 个均匀采样 (例如，使用 Reservoir 采样)

(b) 输出这 t 个采样的中位数

选择尽可能最小的 t，使得不等式 (1) 成立。解释最终算法的正确性，并给出其空间复杂度。

提示: 你可以将 S 划分成 3 组: $S_L = \{a \in S : \mathrm{rank}(a) \le m/2 - \varepsilon m\}$, $S_M = \{a \in S : m/2 - \varepsilon m \le \mathrm{rank}(a) \le m/2 + \varepsilon m\}$, 和 $S_H = \{a \in S : \mathrm{rank}(a) \ge m/2 + \varepsilon m\}$。注意到，如果样本中少于 $t/2$ 个数来自 S_L 及 S_H，那么样本的中位数是一个“好的”估计。
