Partner 1
Partner 2 CSCI 1550 / 2540
Partner 3 March 21st, 2024

Homework 2

Due: March 21st, 2024
Remember to show your work for each problem to receive full credit.

Problem 1 (30 points)

We prove that the Randomized Quicksort algorithm sorts a set of n numbers in time $O(n \log n)$ with high probability. Consider the following view of Randomized Quicksort. Every point in the algorithm where it decides on a pivot element is called a **node**. Suppose the size of the set to be sorted at a particular node is s. The node is called **good** if the pivot element divides the set into two parts. each of size not exceeding $2s/3$. Otherwise the node is called **bad**. The nodes can be thought of as forming a tree in which the root node has the whole set to be sorted and its children have the two sets formed after the first pivot step and so on.

(a) Show that the number of good nodes in any path from the root to a leaf in this tree is not greater than $c \log_2 n$, where c is some positive constant.

(b) Let X_i be iid geometric random variables with probability $\frac{1}{3}$, and Y_i be iid Bernoulli random variables with probability $\frac{1}{3}$. For $n < m$ positive integers, show that

$$\mathbb{P}(\sum_{i=1}^{n} X_i > m) = \mathbb{P}(\sum_{j=1}^{m} Y_i < n).$$

(c) To simplify our analysis, we will assume in the remaining problem that on each root to leaf path, the algorithm will go through exactly $c \log_2 n$ good nodes. We will ignore rounding errors for simplicity, that is, the probability of getting a good node is $\frac{1}{3}$ before exceeding the total number of good nodes.

Show that, with high probability (greater than $1 - 1/n^2$), the number of nodes in a given root to leaf path of the tree is not greater than $c' \log_2 n$, where c' is another constant.

Argue informally that we can still obtain the same estimate as in the previous paragraph even without the assumption on going through exactly $c \log_2 n$ good nodes.

(d) Show that, with high probability (greater than $1 - 1/n$), the number of nodes in the longest root to leaf path is not greater than $c' \log_2 n$. Equivalently, show that with probability greater than $1 - 1/n$ that no pathes from root to leaf is greater than $c' \log_2 n$.

(**Hint:** How many nodes are there in the tree?)

(e) Use your answers to show that the running time of Quicksort is $O(n \log_2 n)$ with probability at least $1 - 1/n$.

Problem 2 (20 points)

In many wireless communication systems, each receiver listens on a specific frequency. The bit $b(t)$ sent at time t is represented by a 1 or -1. Unfortunately, noise from other nearby communications can affect the receiver's signal. A simplified model of this noise is as follows. There are n other senders, and the ith has strength $p_i \leq 1$. At any time t, the ith sender is also trying to send a bit $b_i(t)$ that is represented by 1 or -1. The receiver obtains the signal $s(t)$ given by

$$s(t) = b(t) + \sum_{i=1}^{n} p_i b_i(t)$$

If $s(t)$ is closer to 1 than -1, the receiver assumes that the bit sent at time t was a 1 otherwise, the receiver assumes that it was a -1.

Assume that all the bits $b_i(t)$ can be considered independent, uniform random variables. Give a Chernoff bound to prove the probability that the receiver makes an error in determining $b(t)$ is less than or equal to following quantity

$$\exp(\frac{-1}{2\sum_{i=1}^{n} p_i^2}).$$

Problem 3 (35 points)

Bob is facing a very challenging math question in **CSCI1550/2540**. Even if this math question is very hard, it has a simple binary answer $Y \in \{0, 1\}$ (both answers are equally likely). Bob asks for help from n fellow math-loving friends (numbered from 1 to n), and each of them provides an answer to this math question. However, as this math question is very hard, there is no guarantee that these answers are the same. In particular, friend i provides an answer $X_i \in \{0, 1\}$, for $i = 1, \ldots, n$. Bob knows the expertise of each friend; in particular, he knows that for each $i = 1, \ldots, n$, we have that:

$$X_i = \begin{cases} Y & \text{with probability } p_i > 1/2 \\ 1 - Y & \text{with probability } 1 - p_i \end{cases}$$

Formally, $X_1, \ldots, X_n$ are random variables function of Y. Bob also assumes that these friends won't collaborate with each other; that is, given Y, the random variables $X_1, \ldots, X_n$ are independent.

Bob wants to use a function $f(X_1, \ldots, X_n) : \{0, 1\}^n \to \{0, 1\}$ to obtain the final answer to the hard math problem. He would like to minimize the error that the function f makes a mistake, i.e., he wants to minimize:

$$\Pr(f(X_1, \ldots, X_n) \neq Y) \tag{1}$$

If a function f minimizes (1), we say that f is optimal. Let $\vec{X} = (X_1, \ldots, X_n)$.

(a) For $y \in \{0, 1\}$, let

$$g(\vec{x}, y) = \Pr(\vec{X} = \vec{x} | Y = y) = \prod_{i:x_i=y} p_i \prod_{i:x_i=1-y} (1 - p_i) = \exp\left(\sum_{i:x_i=y} \log p_i + \sum_{i:x_i=1-y} \log(1 - p_i)\right)$$

Show that a function f is optimal if and only if for any $\vec{x} \in \{0, 1\}^n$, it holds that

$$f(\vec{x}) = \arg\max_{y \in \{0,1\}} g(\vec{x}, y)$$

(b) Bob considers a family of functions that is called weighted majority vote. That is, he wants to assign a different weight to the answer of the different friends, based on their competence. Let $\vec{w} = (w_1, \ldots, w_n) \in \mathbb{R}^n$. Given $\vec{w}$, we define:

$$f(\vec{x}; \vec{w}) = \begin{cases} 1 & \text{if } \sum_{i=1}^n X_i w_i \geq \sum_{i=1}^n (1 - X_i) w_i \\ 0 & \text{otherwise} \end{cases}$$

Let $\vec{w}^* = (w_1^*, \ldots, w_n^*)$, where $w_i^* = \ln\left(\frac{p_i}{1-p_i}\right)$. Use the answer to question a. to show that the function $f(\bullet; \vec{w}^*)$ is optimal.

(c) Let $f^*(\bullet) = f(\bullet; \vec{w}^*)$. Use Hoeffding's bound to show an upper bound on the error probability

$$\Pr(f^*(X_1, \ldots, X_n) \neq Y)$$

Hint: Show that if $f^*(X_1, ..., X_n) \neq Y$, then the sum of weights w_i, whose corresponding answer X_i is correct, is less than or equal to $\frac{1}{2} \sum_{i=1}^{n} w_i$.

(d) Suppose that for each $i = 2, \ldots, n$, we have that $p_i = 0.9$, and let $p_1 \to 1$. What happens to the upper bound computed in question *c.*? Is this upper bound useful or not in this scenario?

Homework 2

Problem 4 (20 points)

The total distance to travel from Providence to Boston is 50 miles. We would like to estimate the expected driving time to reach Boston. We can divide the route into 5 segments (numbered from 1 to 5) of 10 miles each, and segment i requires expected travel time $E[T_i]$ (in minutes) to be completed, i.e. the expected total travel time is $E[T] = \sum_{i=1}^{5} E[T_i]$. Furthermore, we know that for each $i = 1, \ldots, 5$, we have that $5 \leq T_i \leq 15$.

We can collect data from many cars that travel along the roads that connect Providence to Boston. In particular, for each $i = 1, \ldots, 5$, we can collect multiple samples of the travel time required to complete segment i.

(a) For any $\delta \in (0, 1)$, suppose we collect m samples for each T_i, how much m do we need in order to estimate the total travel time within 5 minutes of its expected value with probability at least $1 - \delta$? Clearly state your assumptions.

Hint: You don't need to find the smallest m such that the probability is satifised, we are just asking you to find some specific m.

(b) Now, suppose that we can only collect samples of the total travel time to go from Providence to Boston. How does your answer to question a. change? Compare the two answers.

Problem 5 (25 points)

This problem demonstrate the difference between additive and multiplicative error deviation bounds.

Let $G = (V, E)$ be an undirected graph, $V = \{1, \dots, n\}$ and $E \subseteq \{\{i, j\} : i, j \in V \text{ and } i \neq j\}$. We know the number of vertices $|V| = n$. We want to estimate the fraction of pairs $\{i, j\}$ of connected by an edge, $\rho = m/\binom{n}{2}$, where $m = |E|$. We can query an oracle, that given a pair $\{i, j\}$, tells us if i and j are connected by an edge in the graph G, i.e. whether $\{i, j\} \in E$ or not.

(a) **Additive error bound:** Use the Hoeffding's bound to bound the number of queries of pairs, chosen uniformly at random, needed to estimate ρ within an ϵ additive error, i.e. output $\tilde{\rho}$ such that $|\tilde{\rho} - \rho| \leq \epsilon$ with probability at least $1 - \delta$.

(b) **Multiplicative error bound:**

1. Assume that you given a lower bound d on the fraction ρ. If this lower bound is true, how many random queries are needed to find an estimate $\tilde{\rho}$ that satisfies an ϵ multiplicative error, i.e. $|\tilde{\rho} - \rho| \leq \epsilon\rho$, with probability at least $1 - \delta$?
2. Assume now that you don't have a lower bound of ρ. Design and analyze an algorithm that estimates ρ with a number of sample adjusted to the unknown ρ. [**Hint:** Assume first that $\rho > 1/4$, if the condition doesn't hold assume $\rho > 1/8$, etc. Remember to bound the total error probability.]
3. For which values of ρ is it better to just check all pairs?