COL202: Discrete Math	**Tutorial 8**
Instructor: Venkata Koppula	Release date: 19th October

The solutions for the (⋆) marked problems must be submitted on Gradescope by 11:59 am on 25th October.
The ♦ marked problems will be discussed in the tutorial.
The 🍫 marked problems are challenge problems, and solutions for these problems will not be provided. You are encouraged to try these problems, and reach out to the instructor in case you'd like to discuss them.

1 General Discrete Probability

1.1. (♦) **k-wise independence:** A set of n events $A_1, A_2, ..., A_n$ is *k-wise independent* if for every set $I \subset [n]$ such that $|I| = k$, the set of events $\{A_i\}_{i \in I}$ are mutually independent.

Consider a source of randomness, where you press a button, and get a uniformly random element in $\mathbb{Z}_q$ (where q is a prime). Each button-pressing costs *Rs.* 1000. You want a set of n numbers in $\mathbb{Z}_q$ s.t. they're *k-wise independent*. One direct way is to press the button n times to obtain n uniformly random numbers, but do we really need this many calls to the source for *k-wise independence*? Describe a solution that uses only $O(k)$ samples.

More formally, let $t = \Theta(k)$, and for any subset $I \subset [n]$, let $(y_1, y_2, \ldots, y_n)_I$ denote the sequence $(y_i)_{i \in I}$. Describe a deterministic function $F : \mathbb{Z}_q^t \to \mathbb{Z}_q^n$ such that for any subset $I \subset [n]$, $|I| = k$, and any $\theta_1, \theta_2, \ldots, \theta_k \in \mathbb{Z}_q$,

$$\Pr\left[(F(x_1, x_2, \ldots, x_t))_I = (\theta_1, \theta_2, \ldots, \theta_k)\right] = \frac{1}{q^k}$$

where the probability is over the choice of $x_1, x_2, \ldots, x_n$, sampled independently and uniformly at random from $\mathbb{Z}_q$.

2 Concentration Inequalities

2.1. There are n balls and n bins. Each ball is tossed into one of the n bins, uniformly at random. Let X denote the number of bins with at least two balls. What is $\mathbb{E}[X]$? Give an upper bound on $\Pr[X > 3n/8]$.

(🍫) Optional : run a simulation, and plot the probability distribution for random variable X.

2.2. (⋆) In class, we saw the exit polls problem. There are n balls in a bag, where each ball is either red or blue colored. You are given the guarantee that either there are at least $2n/3$ blue balls, or at least $2n/3$ red balls. We want to determine whether there are more red balls or more blue balls, by sampling as few balls as possible.

We discussed the following algorithm: let $t = c \log n$ where c is a sufficiently large constant. For $i = 1$ to t:

1. Sample a ball from the bag, uniformly at random. Let z_i denote the color of the ball. After noting the color, put the ball back in the bag.

If the majority of $\{z_1, z_2, \ldots, z_n\}$ are red, then output "More red balls", else output "More blue balls".

We showed that if $t \geq c \log n$ where c is a sufficiently large constant, then the above algorithm gives the correct output with probability at least $1 - 1/n$.

1. Suppose we change the first step of the algorithm as follows: Sample a ball from the bag, uniformly at random. Let z_i denote the color of the ball. After noting the color, **do not** put the ball back in the bag.

Show that there exists a constant c such that the modified algorithm gives the correct output with probability at least $1 - 1/n$.

2.3. We have two coins: one is a fair coin, but the other produces heads with probability $3/4$. One of the two coins is picked, and this coin is tossed n times. Use the Chernoff Bound to determine the smallest n which allows determination of which coin was picked with 95% confidence.

2.4. We want to store 2 billion records into a hash table that has 1 billion slots. Assuming the records are randomly and independently chosen with uniform probability of being assigned to each slot, two records are expected to be stored in each slot. Of course under a random assignment, some slots may be assigned more than two records.

1. Show that the probability that a given slot gets assigned more than 23 records is less than e^{-36}.
2. Show that the probability that there is a slot that gets assigned more than 23 records is less than e^{-15}.

2.5. Suppose we throw m balls in n bins where each ball is thrown in a bin uniformly at random and independent of other balls. Suppose $m = 2\sqrt{n}$. Use Chernoff bounds plus the union bound to bound the probability that no bin has more than 1 ball.

2.6. Suppose you are given an array A with n distinct integers, and you want to find an approximate median of these n numbers. Any deterministic algorithm will require accessing $\Omega(n)$ elements of A. However, using randomization, we can do this using only $O(\log n)$ accesses. Consider the following algorithm:

1: Choose $k = c \log n$ elements from A, uniformly at random, with replacement.
2: Sort these k elements
3: Return the median x of the k elements

Prove that with probability at least $1 - 1/n$, at least $n/3$ numbers in A are smaller than x, and at least $n/3$ numbers in A are larger than x. You must choose c to be an appropriately large constant.

2.7. (♦) Show that on tossing a fair coin n times, the length of longest contiguous sequence of heads will be $O(\log n)$ with probability at least $1 - 1/n^2$.

2.8. In this problem we consider the task of *supervised learning*. We are given a set S of n train samples of the form $(x_0, y_0), \ldots, (x_{n-1}, y_{n-1})$ drawn i.i.d. from some unknown distribution D over pairs (x, y). For simplicity $x_i \in \{0,1\}^m$ and $y_i \in \{0,1\}$. The goal is to find a *classifier* $h : \{0,1\}^m \to \{0,1\}$ that will minimize the test error. One way to find such a classifier is to consider a collection $\mathcal{C}$ of potential classifiers and look at the classifier that does best on the training set S. The test error is defined as $L(h) = Pr_{(x,y)\sim D}[h(x) \neq y]$ and the train error is defined as $\hat{L}_S(h) = \sum_{i\in[n]} |h(x_i) - y_i|/n$, and using the Chernoff bound we can show that as long as the number n of samples is sufficiently larger than logarithm of $|\mathcal{C}|$, the test error will be close to the train error $\forall h \in \mathcal{C}$. So, prove that for every $\epsilon, \delta > 0$, if $n > \log|\mathcal{C}| \log(1/\delta)/\epsilon^2$, then, $\Pr_S[\forall_{h\in\mathcal{C}} |L(h) - \hat{L}_S(h)| \leq \epsilon] > 1 - \delta$, where the probability is taken over the choice of the set of samples S.

In particular, this tells you that if you have a set of (input, output) pairs in your training data, and you need to choose a hypothesis for future *test* inputs, then the *best* hypothesis is the one that minimizes the error on training data.

2.9. (🍫🍫🍫) **Rumor-spreading:** There are n friends. On day 1, person 1 thinks of a rumor, and the rumor spreads as follows: every day, if a person knows the rumor, then he/she calls up a uniformly random friend (one of the remaining $n-1$ people, uniformly at random) and shares the rumor. Let X denote the number of days for everyone to know the rumor. Clearly, $X \geq \log n$ since the number of people knowing the rumor can only double everyday.

Prove that there exist constants c_1 and c_2 such that

$$\mathbb{E}[X] \leq c_1 \log n \text{ and } \Pr[X > c_2 \log n] \leq \frac{1}{n}.$$

2.10. (🍫🍫🍫) **Balls and Bins, revisited:** There are n balls and n bins. We throw the balls in the bins, one by one, as follows. For $i = 1$ to n, do the following:

1. sample two bins uniformly at random.
2. pick the bin with lower load. If both bins have the same load, then pick the left one. Toss the i^{th} ball in this bin.

Let X denote the max load. What is $\mathbb{E}[X]$? To build intuition for this process, we recommend simulating it for large values of n.

3 Probabilistic Method

3.1. Let $\mathcal{F}$ be a family of subsets of $[2n]$. We say that $\mathcal{F}$ is an *antichain* if for any $A, B \in \mathcal{F}$, A is not a subset of B, and B is not a subset of A. For example, consider the family all subsets of $[2n]$ of size exactly n. Check that these form an antichain. This is the largest possible antichain.

More formally, prove that for any antichain $\mathcal{F}$,

$$|\mathcal{F}| \leq \frac{(2n)!}{n! \cdot n!}.$$

3.2. (♦) Prove that for every $n \times n$ matrix $\mathbf{A}$ with binary entries, there exists a vector $\mathbf{b} \in \{-1, +1\}^n$ such that the absolute value of the largest entry of $\mathbf{A} \cdot \mathbf{b}$ is at most $4\sqrt{n \ln n}$.