

CS203B: Mathematics for Computer Science - III
Semester I, 2018-19, CSE, IIT Kanpur

Assignment 1

Deadline : 6:00PM, 11 August

Important Guidelines:

1. It is only through the assignments that one learns the most in any course. You are advised to refrain from searching for a solution on the net or from a book or from other fellow students. If you are cheating the instructor, you are cheating yourself first. The onus of learning from a course lies first on you and then on the quality of teaching of the instructor. So act wisely while working on this and other assignments.
2. This assignment has 2 parts: (1) programming, and (2) preparing report. The report has to be of single page only. Handwritten reports will not be accepted. You must type it using any word processor (For example : Latex, Microsoft Word, Google Doc). You will have to upload the code on moodle before the deadline.

1. Recruitment Problem

Google USA is going to visit IITK to hire the best qualified (good algorithmic and programming skills) student in his/her final year. There are n applicants and n is obviously really huge since Google USA is offering a huge salary. The following are the constraints:

Constraints

1. The interview panel will select only one applicant and the selection will be based totally on his/her qualifications which can be revealed only after the interview of the applicant.
2. The placement office of IITK has enforced that each applicant should be informed about his/her selection or rejection immediately after the interview.

Assumptions

1. The qualification score of each person, revealed through interview, is a distinct number. You may assume it to be a real number in the interval $(0,1)$. Alternatively, you may assume it to be a distinct integer in a sufficiently large range.
2. The applicants appear for interview in a uniformly random order (Each permutation is equally likely).

Considering the constraints and the assumptions stated above, the interview panel of Google USA has devised the following strategy.

The strategy

Fix a number $k < n$. Interview and reject first k applicants. After that continue conducting the interviews, and stop as soon as we find an applicant better than the first k applicants. If we don't find any applicant better than the first k applicants, then no one is hired.

The Objectives

Let p_{kn} be the probability of selecting the best qualified applicant for a given k and n .

- Objective 1 Think hard on calculating p_{kn} using all the probability tools you know. You must not discuss this with anyone else. You should spend at least one hour on this task. Note that you are not supposed to submit anything for this part.
- Objective 2 Estimate accurately the value of p_{kn} for various values of k for a given n , and repeat it for values of $n \in \{100, 200, 300, \dots, 1000\}$. You have to achieve this objective through a computer program. Calculate the value of k for each n that maximizes the probability p_{kn} .

2. The tasks of the assignment

The code

You have to design a simple, neat, and efficient program that estimates the probability mentioned above. You will have to upload its code before the deadline of the assignment.

The report

Prepare a report of single page. The report should have one plot that shows the behavior of p_{kn} as function of k for a particular n (ideally the largest). The report should also have a table where you have to enter the optimal value of k (that achieves maximum p_{kn}) for each $n \in \{100, 200, 300, \dots, 1000\}$. You may write any additional observation that you derived from the experiment. But be precise. It must not be more than a few sentences.

Hints and directions

1. Suppose you are given a coin that gives HEAD with probability p , and p is not equal to $1/2$. How will you estimate p ?
2. Learn about various random number generators in library of C and choose the one suitable for the assignment (there could be multiple such generators suitable for the assignment depending upon the code you write).
3. You might need to create an array of size n . Depending upon the code you write, you might either fill it up using random numbers in $(0,1)$ or first fill it up with integers from 1 to n , and then permute it randomly uniformly.
4. Choose the number of times you will repeat the experiment to estimate the corresponding probability (p_{kn}) accurately. Your program may take a lot of time for this task. So you should make attempts to make your program efficient.