In the name of God, the Merciful, the Compassionate

Fall 2018 CE 40181 **Probability and Statistics Project**

Early report deadline: Wednesday, Nov 22 2017.

Phase one deadline (the Pilot section and Part 1): Wednesday, Dec 13 2017

Phase two deadline (Parts 2 and 3): Wednesday, Jan 03 2017

1 Overview

The purpose of this project is to provide a comprehensive overview of familiar probability distributions for students given their computer science background. The project has been defined in two general phases. The first phase includes a pilot alongside a part, while the second phase is categorized into two parts. In this document, the pilot section is introduced first, while it is about defining discrete probability distributions in personal computers. Then, the first part is introduced, while it is about defining some continuous probability distributions employing the functions defined in the pilot. After that, the second part which is about defining a package and designing a graphical user interface is defined. Finally, the last part, i.e. third part, is described to be used as a data analyzer tool.

The project will be done in 4-person teams. Each team selects a leader who is responsible for any communications between the team members and the TAs or the instructor. All project and report files are uploaded by the leader on the course page. Hence, the leader is a communicating person of the group. Each team has to provide an **EARLY REPORT**. This report contains the names of the members, the leader and a detailed description of the job scheduling in the team. Note that this report should be more than a simple email notification which is sent to the instructor. The grading policy is according to the tasks finally done according to the first report. Thus, be careful in defining the responsibilities of group members. It is worthy to note that, there will be a face to face delivery after the deadline of each phase. There is also another report, which is sent at the deadline of the 2nd phase of the project. The last report contains a concise description of each member's job and project steps (Two pages suffices).

2 Phase One: Discrete and Continuous Probability Distributions

2.1 Pilot: Efficient Pseudo-Random Generator

You have learned about pseudo-random numbers and some random-generating procedures previously. Do a bit research to find out about the most accurate procedures (more accuracy = more randomness in numbers). You have to choose a procedure, describe your decision, and assert it. Note that you must commit to it in the next steps as well; so choose the procedure carefully!
step 1: Write a function "rgenerator", that takes the parameters of the generator as input and generates random numbers.

2.2 Uniform Numbers

In this section you should generate uniform random numbers using the previous step.
step 2: Using "rgenerator", write a function "dugen", taking the two integer numbers as input and generates random numbers uniformly distributed between two integer inputs.
step 3: Using "dugen", write a function "cugen", which generates a random uniform number between 0 and 1.

2.3 Bernoulli Numbers

Bernoulli distribution is one the most simple as well as famous distributions. In each Bernoulli trial there are two possible outcomes, namely success or failure, for the probability of success equal to p, the probability of failure is $1 - p$.
step 4: Using "cugen", write a function "brgen", taking a float number as the parameter of Bernoulli distribution and generates a random number from$\{0, 1\}$.

2.4 Binomial Distribution

A binomial random variable can be seen as the result of repeated Bernoulli Trials.
step 5: Using "brgen", write a function "bigen", taking two parameters, one float as the probability parameter and one integer "n" as the number of trials, and generates a random number from 0 to n.

2.5 Geometric Distribution

The number of failures in Bernoulli trials, between two wins, follows the geometric distributions.
step 6: Using "brgen", write a function "gegen", taking a parameters and generates a geometric random variable.

2.6 Exponential Distribution

Exponential distribution is a popular distribution which is used to model waiting times and memoryless processes. An exponential distribution with parameter λ can be calculated as $-\frac{1}{\lambda}log(X)$. Where X is a uniformly distributed random variable in $[0, 1]$.
step 7: Write a function "expgen", using "cugen" which takes the parameter λ and generates an exponentially distributed random variable.

2.7 Gamma distribution

Summation of k i.i.d exponential random variables leads to a gamma distributed random variable.
step 8: Using "expgen", write a function "gagen", taking a float number as the parameter of the

underlying exponential distribution and an integer as k and generates a random number from gamma distribution.

2.8 Poisson Distribution

If an exponentially distributed variable is modeled as the waiting time before an arrival, the Poisson distributed variable can be modeled as the number of arrivals during a period of time of length t .
step 9: Using "expgen", write a function "pogen", taking a float number as the parameter of the underlying exponential distribution and a float as the length of time interval and generates a random number from Poisson distribution.(the generated Poisson variable will be of parameter λt.)

2.9 Normal Distribution

The Poisson (λ) distribution can be considered as an approximation of N(λ,λ).
step 10: Using "pogen", write a function "nogen", taking two parameters, one fload "u" as the mean and one float "s" as the variance of the distribution, and generates a random normal number with mean "u" and variance "s". (First, generate a Poisson number and then use scale and transition to achieve the desired mean and variance.)

2.10 Visualization

step 11: For each distribution defined in the previous parts, add a function powered by **ggplot2** to plot each distribution function.

3 Phase Two: A package and a graphical user interface in R

So far, you have implemented 8 familiar probability distributions. In this part, after adding functions for visualization, you are going to share your defined functions with other users through an R package as a GitHub repository and a software designed using Shiny in R.

3.1 Package

Do the following step by step (each step composes 10% of the score):

1. Create your package directory.

2. Put all of the functions defined in the previous parts in your package.

3. Add help to your package. Introduce each of your functions clearly. Including:

   - A brief description.
   - The method you use for generating pseudo-random.
   - Inputs, outputs and examples of the function usage.
   - Related functions in your package.

4. Make your package a **github** repository.

3.2 GUI Software

Use **Shiny** to design an interactive, web-based software representing your package graphically.
In this step, you are going to offer a service to a user which can be unfamiliar with computer environment, hence take care of simplicity of your software.
Another point of which the group should be aware is providing a scientifically reliable service. So pay much attention to the theoretical basis of your application.

- Create a tab for each of the eight distributions.
- In each tab define proper text boxes in order to get parameter values of the distribution. The number of text boxes will be equal to the number of distribution parameters.
- In a tab, define two types of bottom. One for generating a random number from the distribution; one for plotting the distribution. Use your previous defined functions to generate and plot the random variables.

3.3 Add estimation feature to GUI software

This is the final part of the project. In this part you are going to simply add an estimation function to each of the tabs in your software.

1. For each tab define a box to get a .txt formated data from the user containing a vector including numbers generated under a distribution (the distribution of the tab).
2. Afterwards, you have to estimate the parameters of the underlying distribution using maximum likelihood. Show the estimated parameters in a pre-defined text box.
3. Finally, predict another number from the estimated distribution and plot it in the tab.

Good Luck