Assignment 3: RNG and Data Presentation

Nikesh Vijaykumar Hegde, 119220167

1 Distribution information

There are two distributions which we were asked to perform for experimentation i.e. Kumaraswamy and Logarithmic. Each expression will be explained below.

1.1 Kumaraswamy Distribution

Kumaraswamy distribution belongs to the family of continuous probability distributions defined on the interval (0,1). Simpler to use and simulation experiments can be expressed in a closed-form because of its probability density function, cumulative distribution function, and quantile functions. The extreme values for this distribution was [0,1].

Cumulative Distribution function (CDF) is defined as

$$f(x) = 1 - (1 - x^a)^b$$

where a and b are non-negative shape parameters and x ε (0,1). CDF was calculated for the Kumaraswamy distribution by passing random numbers passed for parameter x which was generated by taking user input for seed and range. Other parameters, such as a and b were also passed in the command line while executing the program. See Figure 1, where F(x,P(x)) is plotted along axes. X axis represent the random numbers and Y axis represent the CDF of the random number generated by the formula given above for CDF. Boxplot for the generated random variables can be seen in figure 2

Inverse Cumulative Distribution function (F^{-1}) is defined as

$$f^{-1}(x) = \sqrt[\alpha]{1 - \sqrt[\beta]{1 - x}}$$

where α and β are positive $0 \leq$ x ≤ 1. The generated CDF values were provided to inverse function to obtain back the randomly generated numbers provided initially.

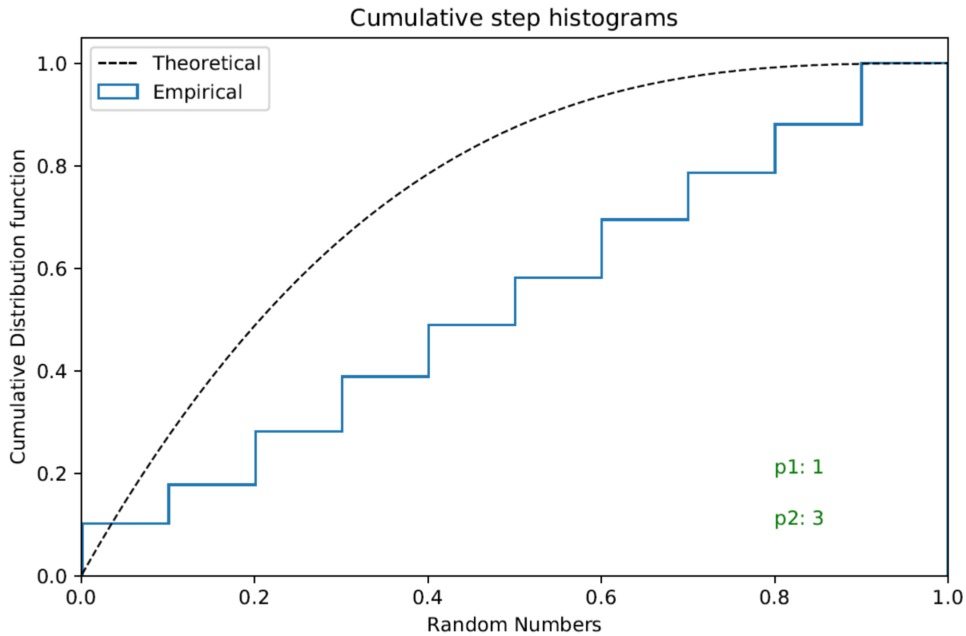

Figure 1: Generated Kumaraswamy Cumulative Distribution Function Plot

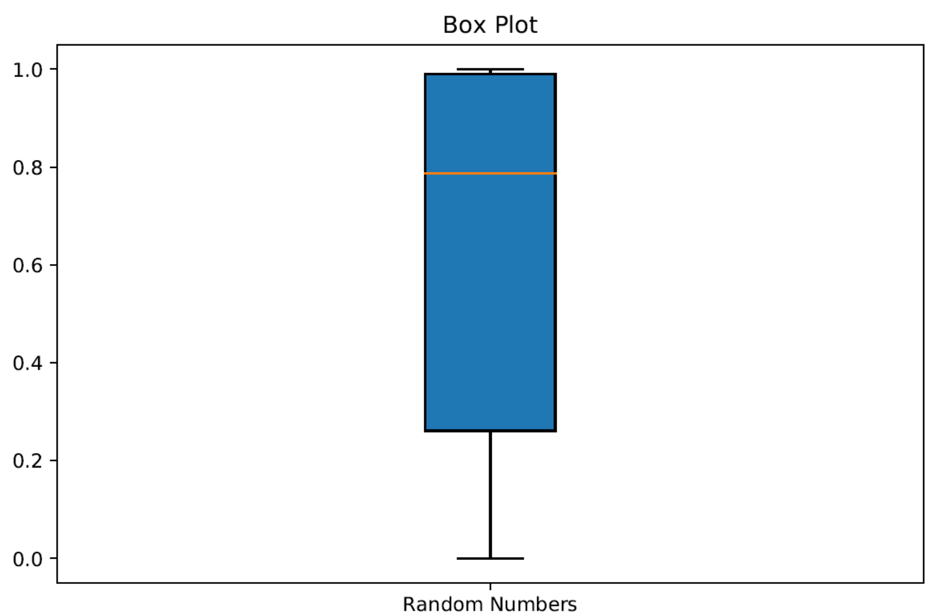

Figure 2: Boxplot for Kumaraswamy Distribution

1.2 Logarithmic Distribution

Logarithmic distribution belongs to the family of discrete probability distribution originated from Maclaurin series expansion

$$-\ln(1-p) = p + \frac{p^2}{2} + \frac{p^3}{3} + \cdots.$$

from this we obtain the identity

$$\sum_{k=1}^{\infty} \frac{-1}{\ln(1-p)} \frac{p^k}{k} = 1.$$

This leads to the Probability Mass function (PMF) of a log(p) distributed random varible

$$f(k) = \frac{-1}{\ln(1-p)} \frac{p^k}{k}$$

for k $\geq$ 1, and where 0 $\leq$ p $\leq$ 1. Distribution is properly normalized because of the identiy.

PMF was obatined by passing a parameter p from command line and k having discrete range of value. The cumulative distribution is obtained by taking a cumulative sum of the PMF passed to the function.The cumulative distribution function plot for function F(x,P(x)) can be seen in below figure 3,where x axis represents discrete value and y axis represents CDF. The boxplot for the same CDF can be seen in below figure 4.

2 Usage Guide

This is a short usage guide which helps you use the myRng.py code to generate and plot the sequences. Please follow the steps given below to get the desired output. (Note that the code has been implemented and tested in Windows OS)

- Before running the code please make sure Python 3.6.0 or higher version is installed in your computer/laptop
- To check if python is installed or not, please follow this link for instructions. https://edu.google.com/openonline/course-builder/docs/1.10/set-up-course-builder/check-for-python.html
- After installing python, please ensure you have packages like **matplotlib** and **numpy**.
- In the Command prompt, write python and then press enter. Please follow the below steps to check whether the package exists or not:

  >>> import sys

  Then test for installed modules:

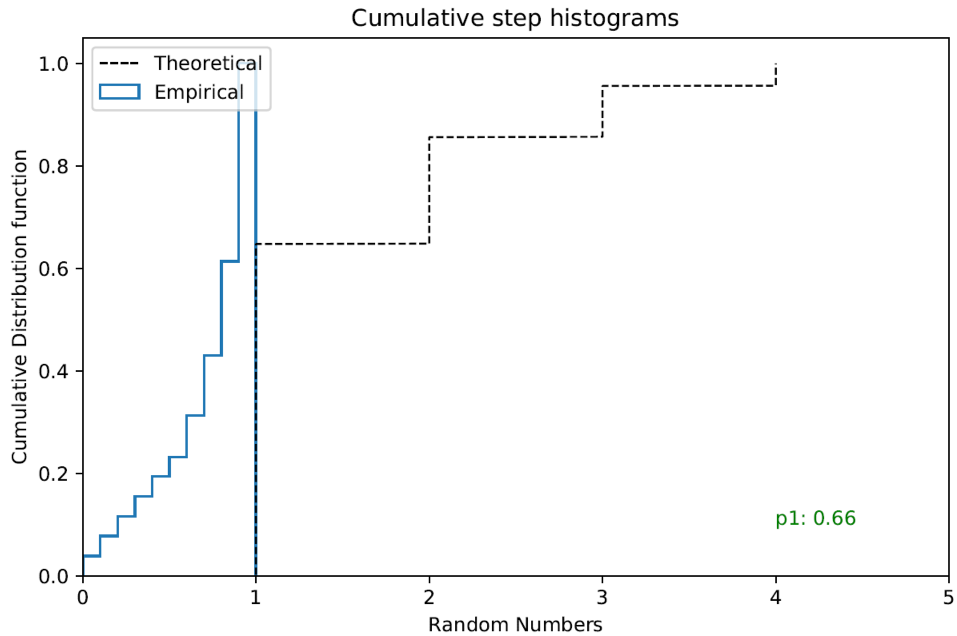

Figure 3: Generated Logarithmic Cumulative Distribution Plot

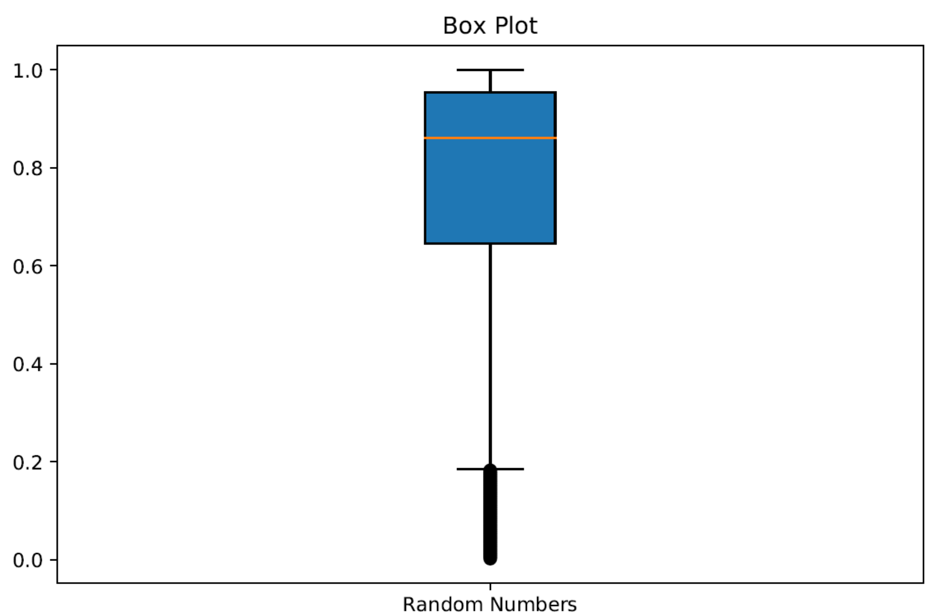

Figure 4: Box-plot for Logarithmic Distribution

>>> 'numpy' in sys.modules

True

>>> 'scipy' in sys.modules

False

It returns true if the package exists and false if it does not.

- You can install packages using pip in command line. Note, pip does not work if you are using proxy for connection.

- To install a package, write "python -m pip install <package_name>" or follow this guide here https://packaging.python.org/tutorials/installing-packages/

- Once you ensure that correct version of python is installed and that all the required packages are also installed, then open Command Prompt on windows and navigate to the folder where the myRng.py file is stored.

- Once you have navigated to the folder, type the below contents on command line "python mytool.py outputPrefix n seed distribution par1 par2"

- In the above Command prompt line, you will have to substitute values for 6 parameters outputPrefix, n, seed, distribution, par1 and par2. Please find below what each of them denotes.

1. **outputPrefix** - a prefix for the name of generated files
2. **n** - number of items per generated sequence
3. **seed** - seed value to be used for generating the sequence
4. **distribution** - the name of distribution, e.g., kumaraswamy, logarithmic
5. **par1** - the first distribution parameter
6. **par2** - the second distribution parameter

- An example of the command in the command prompt(Windows) will be as given below

1. Kumaraswamy - **"python myRng.py kum 1000 10 kumaraswamy 2 5"**
2. Logarithmic - "**python myRng.py log 1000 10 logarithmic 0.66**"

- The files that are generated using the tool are then stored in the same folder where the code exists.

- The file names will start with the OutputPrefix sent in the command line argument. For example, if the outputPrefix is "kum" then the file names will be kum_rv.txt, kum_cdf.txt and kum_cdf.pdf

1. %outputPrefix_rv.txt file contains the random variable sequence generated
2. %outputPrefix_cdf.txt file contains the CDF data that is generated
3. %outputPrefix_cdf.pdf file contains the CDF Plot and boxplot generated