CSCI 3320: Fundamental of Machine Learning **2020-2021 Term 2**

Programming Assignment 0

Instructor: Prof. John C.S. Lui Due: 23:59 on Sun. Feb. 14th, 2021

1 Introduction

In this programming assignment, you will use Python programing language and plotting functions in Matplotlib to derive and plot the distribution of the sample mean. The aim of this assignment is to help you get familiar with Python programming, the sampling theory and parameter estimation.

1.1 File Descriptions

To start, you need to download the `asgn0.zip` file from the Blackboard. In `asgn0.zip`, we provide the following files for you:

- `mean_distribution.py`: contains the basic python script, where you need to fill in your codes for calculating the sample mean distribution.
- `data.txt`: data file and has the following format:

```
6
0.0
1.0
0.5
0.6
0.1
0.3
0.2
0.4
```

The description of the file are as follows:

The first line contains the total number N of data points in the file, which is 6.

The second line contains the lower bound (no data point will be less than this bound) of the data points.

The third line contains the upper bound (no data point will be larger than this bound) of the data points.

The rest of the lines (N of them) are the data points, which are 0.5, 0.6, 0.1, 0.3, 0.2 and 0.4.

Note that the N numbers are generated from some probability distribution (may or may not be normally distributed)

Note: Please do **not** change the file names (`*.py`) of the files described above.

2 Distribution of Sample Mean(50%)

2.1 Generate Sampling Means and PMF

First, you need to write a Python function to generate samples and get their means:

- Read in N from the data file
- Take sampling size n as the first parameter of the function
- Read in a total of N/n groups of data, where each group has n numbers
- Generate a sampling mean for each group.

```
e.g. Use the data file data.txt Section 1.1
Parameter: sampling size n=3
Get N/n=6/3=2 groups of data.

         samples            mean
Group 1: [0.5, 0.6, 0.1]    0.4
Group 2: [0.3, 0.2, 0.4]    0.3
```

Then, for these N/n sampling means, you are asked to generate the corresponding probability mass function (PMF) or the distribution of the sampling mean. This can be accomplished in various ways. One possible way is to do the following:

- Define the width of a bin be δ, say $\delta = 0.1$ (or even less). Take δ as the second parameter of the function
- The i^{th} bin stores the number of sampling means which have values between x_i and $x_i + \delta$.
- After counting all N/n sampling means in their corresponding bin, normalize the numbers of all bins by dividing them by N/n.

For the bin mentioned above, it can be realized using the `dictionary` data type in Python. Dictionary implements the key-value pairs, with key being the x_i and value being the number of sampling mean between x_i and $x_i + \delta$.

```
e.g. Use the data file data.txt Section 1.1
Parameter: sampling size n=3; width of a bin delta=0.1
Get (max-min)/delta=(1.0-0.0)/0.1=10 bins.

bin    # of sampling means    PMF
[0.0,0.1):  0                  0
[0.1,0.2):  0                  0
[0.2,0.3):  0                  0
[0.3,0.4):  1                  0.5
[0.4,0.5):  1                  0.5
[0.5,0.6):  0                  0
[0.6,0.7):  0                  0
[0.7,0.8):  0                  0
[0.8,0.9):  0                  0
[0.9,1.0]:  0                  0
```

2.2 Estimator

The above procedure will give you an estimate of the sampling mean distribution. Note that it is an estimation due to the bins width δ.

Compute the mean of your PMF (or $E[\bar{X}]$), which is the estimator of the mean of population via the following: $E[\bar{X}] = \sum x_i Prob[\bar{X} = x_i]$. Note that $Prob[\bar{X} = x_i]$ is the normalized value of the bin for value x_i. You can compare this to the average of the N numbers.

2.3 Implementation Requirement

The function should takes 2 parameters: sampling size n and width of bin *delta*.
Then you should return 2 objects: (1) a `dictionary` with key being the x_i and value being the PMF value of x_i (normalized number of sampling mean between x_i and $x_i + \delta$); (2) the mean of your PMF (i.e. $E[\bar{X}]$).
Please code in the given python file `mean_distribution.py` which contains the function with format shown as below,

```
def estimate_mean(n, delta):
        #code here
        return bin_pmf_dict, mean_PMF
```

3 Plotting and analysis(50%)

In this section you need to plot your data and basically you are suggested to learn how to use **pyplot** to draw a graph, which can be a powerful tool in your future study and research. First install the matplotlib package using pip:

```
pip install matplotlib
```

After that, you will be able to import the pyplot as the following:

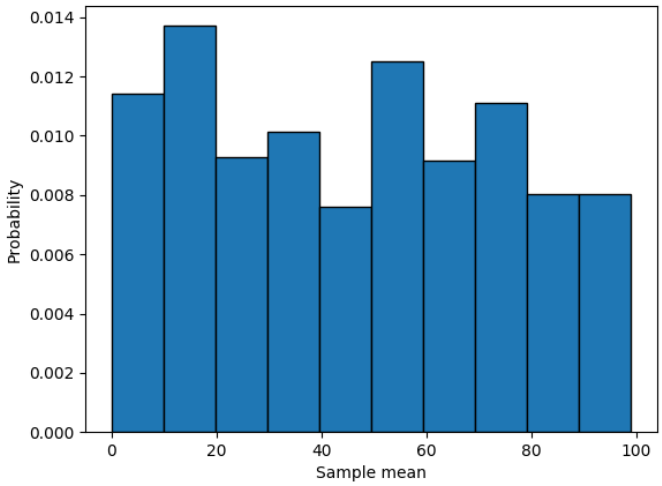
```
import matplotlib.pyplot as plt
```

If you can import it successfully, that means you have installed the package properly. Then follow the following instruction to finish this part of assignment.

1. **Plot the PMF for different δ and n. (25%)**
   As we know, different δ or n will make your probability mass functions different and in order to show the difference you can draw them using **pyplot**. In your figures, the x axis should be the key of your dictionary **value_count_dict** and y axis should be the corresponding probability (value of the dictionary). In this task, try different δ and n to see what is the difference, and also submit four figures you generate, where the parameters in four situations are:

   - $\delta = 0.1, n = 20$ (1)
   - $\delta = 0.1, n = 100$ (2)
   - $\delta = 0.001, n = 20$ (3)
   - $\delta = 0.001, n = 100$ (4)

   Please note that you need to name the figures according to the sequence we give you. For example, **1.png** is denoted as the first situation above. The result figures should look like the following (**Note**: this is just an example and your x axis and y axis should be designed according to your PMF):



2. **Analyze the effect of δ and n. (25%)**
   In this task, you need to write a brief report containing the following two parts of contents:

   - Describe how δ and n affect the PMF.
   - Describe how close the different PMF is to the true sampling distribution of the mean and your $E[\bar{X}]$ to the population mean with respect to different δ and n.
   - Give the clear explanation of why the result would be like that.

   Please submit your brief report in pdf format and name it as **report.pdf**.

4 Submission

Instructions for the submission are as follows. **Please follow them carefully.**

1. Test all your Python scripts before submission. Any script that has syntax error will not be marked.
2. Submit your code script, figures and the report and name them correctly.
3. Zip all files into a single zipped file named `<student-id>_asgn0.zip`, where `<student-id>` should be replaced with your own student ID. e.g., `1155012345_asgn0.zip`
4. Submit the zipped file `<student-id>_asgn0.zip` to CUHK Blackboard system (https://blackboard.cuhk.edu.hk) no later than 23:59 on Sun. Feb. 14th, 2021. If you do not know how to submit it through the CUHK Blackboard system, send us an email or ask us during office hours.