DSCI 6607– Fall 2024

Assignment 2*

Question 1

Recall the bisection method, we leaned in assignment 1. The bisection method can be generalized to deal with the case $f(x_l)f(x_r) > 0$ (i.e., the two end points do not have opposite signs), by broadening the bracket. That is, we reduce x_l and/or increase x_r, and try again. A reasonable choice for broadening the bracket is to double the width of the interval $[x_l, x_r]$, that is

$$m \leftarrow (x_l + x_r)/2,$$
$$w \leftarrow x_r - x_l,$$
$$x_l \leftarrow m - w,$$
$$x_r \leftarrow m + w,$$

a. Incorporate bracket broadening into the bisection method. Note that broadening is not guaranteed to find x_l and x_r such that $f(x_l)f(x_r) \leq 0$, so you should include a limit on the number of times it can be tried.

b. Use your modified function to find a root of

$$f(x) = (x-1)^3 - 2x^2 + 10 - sin(x),$$

starting with $x_l = 1$ and $x_r = 2$. Use R software to solve this question. [**20 points**]

Question 2

We plan to test the equality for the means of two samples in `Python` in this question. [**20 points**]

1. Let x and y be two samples of sizes n_1 and n_2, respectively. To test $H_0 : \mu_x = \mu_y$ vs $H_1 : \mu_x \neq \mu_y$, we compute the test statistic

$$t = \frac{\bar{x} - \bar{y}}{\sqrt{s_p^2(\frac{1}{n_1} + \frac{1}{n_2})}}$$

where $\bar{x}, \bar{y}$ denote the means of x and y samples and

$$s_p^2 = \frac{(n_1 - 1)S_1^2 + (n_2 - 1)S_2^2}{n_1 + n_2 - 1}$$

where

$$s_1^2 = \frac{1}{n_1 - 1}\sum_{i=1}^{n_1}(x_i - \bar{x})^2$$

$$s_2^2 = \frac{1}{n_2 - 1} \sum_{i=1}^{n_2} (y_i - \bar{y})^2$$

2. Write a python function which takes x and y as two lists and returns the observed test statistic.
3. Generate 50 observations form normal distribution with mean =1 and standard deviation=2 and assign the data to list x. Generate 57 observations from uniform distribution between -2 and 2 and assign data to list y.
4. Apply your function and compute the test statistic for the two samples from part 3.

Question 3

1. Consider the `python` list

$$x = [3, 8, 13, 18, 108, 25, 23, 17, 203, 11, 23]$$

Write a `python` function where takes the list and uses only list comprehension and returns the odd values smaller than 23.
[**20 points**]

Question 4

Let $X_i \sim N(\mu, \sigma^2), i = 1, \ldots, 10$, where μ and σ^2 denote mean and variance of the population, respectively.
[**20 points**]

1. Find mathematically the distribution of statistic $\sum_{i=1}^{10} X_i$. Show all your mathematical work and explain them.
2. We would like to test the finding of part (1) numerically. To do that first generate a sample of size 10 from the Normal distribution with parameters $\mu = 23$ and $\sigma^2 = 3.6$ and the compute the sum of the generated observations.
3. Use python and simulate 10000 times part (2) and compute the sum of the generated samples of size 10.
4. Plot the histogram of the 10000 observed statistics from part (3). Then show the density curve of the theoretical distribution you found in part (1) on the histogram.
5. Explain your findings.

Question 5

The continuous random variable X has the following probability density function (pdf), for some positive constant c,

$$f(x) = \frac{3}{(1+x)^3}, \quad 0 \leq x \leq c.$$

a. Find c which makes f a legitimate pdf?

b. Use R and plot the pdf curve of the random variable.

c. What is $E(X)$?

d. Use R and simulate 1000 observations from this statistical population?

e. Use the generated data from part (d), estimate the mean and variance of the distribution? [**20 points**]

Question 6

1. Write a `Python` function which takes a list of numbers `x` and returns a dictionary including

- `Min` : minimum value of `x`
- `Q_1` : first quartile of `x`
- `M` : median of `x`
- `Q_3` : third quartile of `x`
- `Max` : maximum value of `x`
- `IQR` : $Q_3 - Q_1$,
- `Outliers` : a list of x values which are either smaller than $Q_1 - 1.5 \times IQR$ or greater than $Q3 + 1.5 \times IQR$.

2. Apply your function in part (1) to the following list [**20 points**]

$$x = [2, 36, 12, 14, 204, 21.6, 22.5, 1, 32.8, 32.1, 13, 10, 88, 3.3, 3.1, 88]$$

Question 7

In this question, we plan to learn how to implement leave-one-out cross validation in `Python`. [**20 points**]

1. In regression analysis, the coefficients of the regression model

$$y = \beta_1 x_1 + \ldots + \beta_p x_p, \tag{1}$$

are estimated by

$$\widehat{\beta} = \left(\mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{y} \tag{2}$$

where $\mathbf{X}$ is $n \times p$ design matrix (i.e., n observations with p columns) and $\mathbf{y}$ is the response vector of size n.

2. Write a python function which takes X and y where X is $(n \times p)$ and y is your response vector $n \times 1$. The function literately removes the i-th individual and estimate the coefficients your regression based on $(n-1)$ observations, that is $\widehat{\beta}^*$. The trained coefficients are used to predict the response of the i-th individual by
$$\widehat{y}_i = \mathbf{x}_i^\top \widehat{\beta}^*.$$

Your function then repeats the above process for all observations $i = 1, \ldots, n$ and computes $\widehat{y}_i$, $i = 1, \ldots, n$ in a similar fashion and finally reports the root MSE as

$$\sqrt{MSE} = \sqrt{\sum_{i=1}^{n} (y_i - \widehat{y}_i)^2}.$$

3. Load the diabetes data from sklearn package in python

```
from sklearn.datasets import load_diabetes
diab = load_diabetes()
```

Take a random sample of 56 observations from the data set and compute your X as the first three features of these 56 observations. The target of these observations will be your response vector.

4. Apply your function form part 2 to the data set of part 3 and report the root MSE.

Due on Friday, October 25, by 3 pm

Have fun!