Question 1: Every Output shown below is from the python **Assignment1_Question1.py**

PART A.

➔ Following are the results from the output of python code:

```
---------------------------------------------- PART A ----------------------------------------------
Mean =  75.0568006
Count =  4804
Standard Deviation =  27.4453554
Minimum =  11.55
Maximum =  195.6
10th Percentile =  42.823
25th Percentile =  55.46
Median =  71.825
75th Percentile =  91.1725
90th Percentile =  111.115
```

PART B.

As given in question we need to recommend a bin width having bins within range of 5 to 50. Considering this requirement, we can calculate the number of bins for every bin width given as follows:

➔ Number of bins = Round (Number of Maximum Value of Dataset / Bin Width)

Using this formula, we get following result:

D=Bin Width	Total Bins	Bins within 5-50 range
0.1	1966	N
0.2	978	N
0.25	782	N
0.5	391	N
1	195	N
2	98	N
2.5	78	N
5	38	Y
10	19	Y
20	10	Y
25	8	Y
50	4	N
100	2	N

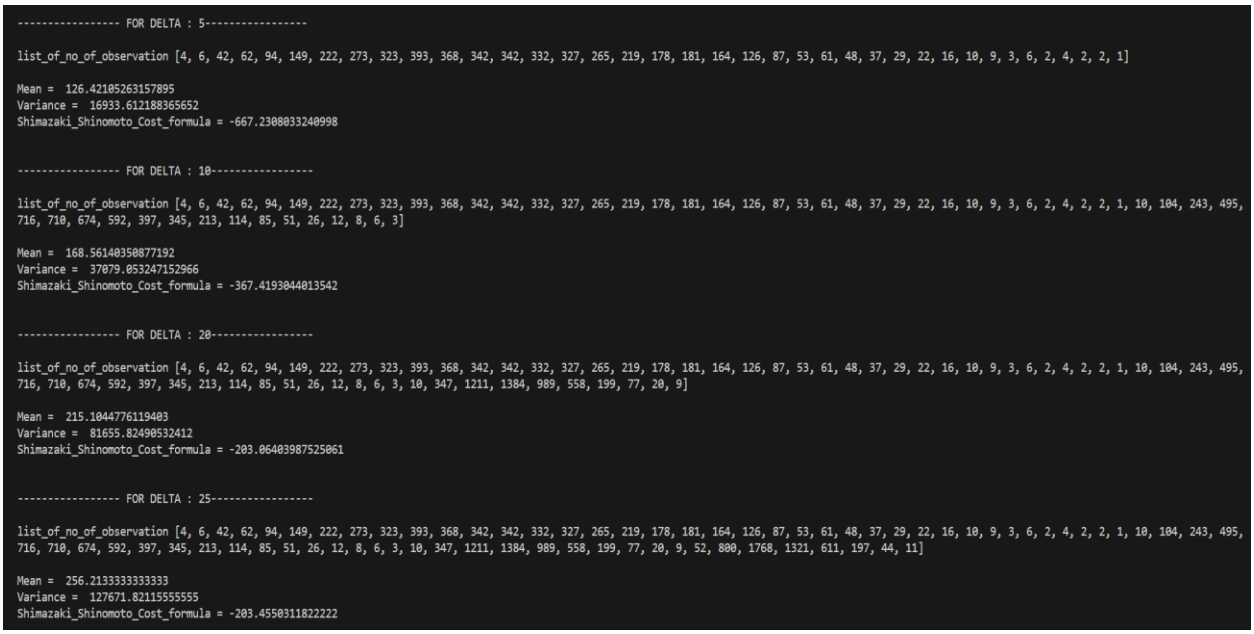
As given in the above table out of 13 bin widths only 4 widths fall under the required bin number condition.

- ➔ Now Using Shimazaki and Shinomoto method we can recommend the optimal bin width. In order to do so we need to calculate:
    1. Calculate the number of observations falling under each bin.
    2. Calculate the mean and variance.
    3. Compute the Criterion C(d) = (2*mean - variance)/d*d, where d is the bin width.
    4. The minimum value for C(d) from each d will be the optimal Bin Width
- ➔ Following is an example for the above method:

    Take Bin Width, d = 5.

    1. List of Observation for each bin: [4, 6, 42, 62, 94, 149, 222, 273, 323, 393, 368, 342, 342, 332, 327, 265, 219, 178, 181, 164, 126, 87, 53, 61, 48, 37, 29, 22, 16, 10, 9, 3, 6, 2, 4, 2, 2, 1]
    2. Mean = 126.42105263157895
       Variance = 16933.612188365652
    3. C(d) = -667.2308033240998

- ➔ Below is the Screenshot of output for the same for every possible Bin width:



➔ By this calculation we get:

D = Bin Width	Total Bins	C(d)
5	38	-667.2308033
10	19	-367.4193044
20	10	-203.0640399
25	8	-203.4550312

➔ From above calculation, the minimum value for criterion is for D = 5. Hence, we can say that 5 is the Optimal Bin Width.

The whole calculation is done in **Assignment1_Question1.py.**

PART C.

To draw the Density Estimator, we need to do following calculation:

1. Select a bin width h.
2. Calculate the midpoint of every bin.
3. For every midpoint **Mj** calculate -> **u = Xi – Mj/h**, where **Xi** is all the observation in the data
4. Based on u, we calculate the weight w(u) as:
   ➔ 1, if -0.5 < u <= 0.5
   ➔ 0, otherwise
5. Compute Density of midpoint $\hat{p}(u) = \Sigma\, w(u) / (Nh)$, where N is the number of total observation.

As given, we will take Bin Width h = 5 which was compute from Part B.

As per the algorithm:

1. Bin Width h = 5
2. Midpoint list = [2.5, 7.5, 12.5, 17.5, 22.5, 27.5, 32.5, 37.5, 42.5, 47.5, 52.5, 57.5, 62.5, 67.5, 72.5, 77.5, 82.5, 87.5, 92.5, 97.5, 102.5, 107.5, 112.5, 117.5, 122.5, 127.5, 132.5, 137.5, 142.5, 147.5, 152.5, 157.5, 162.5, 167.5, 172.5, 177.5, 182.5, 187.5, 192.5, 197.5]

3. After Calculation we the list of Density of every midpoint as shown in below Screenshot:

```
List of Midpoints [2.5, 7.5, 12.5, 17.5, 22.5, 27.5, 32.5, 37.5, 42.5, 47.5, 52.5, 57.5, 62.5, 67.5, 72.5, 77.5, 82.5, 87.5, 92.5, 97.5, 102.5, 107.5, 112.5, 117.5, 122.5, 127.5, 132.5, 137.5, 142.5, 147.
5, 152.5, 157.5, 162.5, 167.5, 172.5, 177.5, 182.5, 187.5, 192.5, 197.5]

Density List for Every Midpoint is :  [0.0, 0.0, 0.0001665278934221 4822, 0.0002497918401332223, 0.0017485428809325561, 0.0025811823480432973, 0.003913405495420483, 0.0062031640299750205, 0.009242298084929
226, 0.011365528726061615, 0.013447127393838468, 0.016361365528726062, 0.015320566194837635, 0.014238134887593672, 0.014238134887593672, 0.0138218151540383, 0.013613655287260617, 0.011032472939217318, 0.0
09117402164862615, 0.007410491257285 5956, 0.007535387177352207, 0.006827643630308077, 0.005245628642797669, 0.0036219816819317236, 0.002206494587843464, 0.002539955037468776, 0.001998334721065778 3, 0.00154
03830141548708, 0.001207327227310 5745, 0.0009159034138218152, 0.0006661115736885929, 0.0004163197335553705 4, 0.0003746877601998335, 0.0001248959206666 1114, 0.0002497918401332223, 8.326394671107411e-05, 0.
0001665278934221 4822, 8.326394671107411e-05, 8.326394671107411e-05, 4.163197335553705 5e-05]
```

After Calculating we can visualize this on graph as: