Assignment-1

2020102004

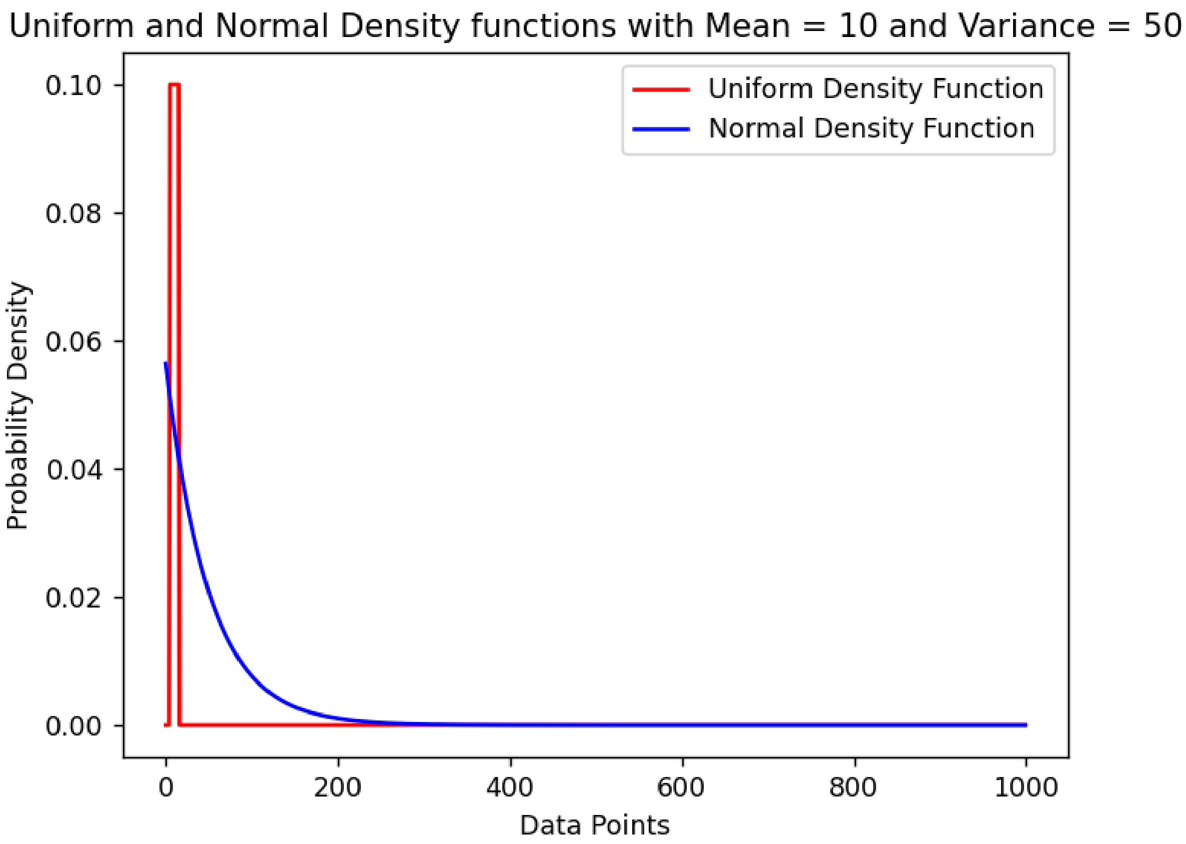
Programming based problems

Question-3

Let’s take an example of normal distribution and uniform distribution, with mean = 10 and variance = 50

```python
import numpy as np
from matplotlib import pyplot as plt

mean = 10
variance = 50
quantile = np.arange(0, 1000)

## Uniform distribution
a = 5
b = 3*a
udf = np.zeros(len(quantile))
a_idx = quantile.tolist().index(a)
b_idx = quantile.tolist().index(b)
udf[a_idx:b_idx+1] = (1/(b-a))

## Normal distribution
ndf = np.exp(-((quantile) *2)/(2*variance ))*(1/(2*3.1415*variance )**0.5)

## Plotting
plt.plot(quantile, udf, color = 'red', label ='Uniform Density Function')
plt.plot(quantile, ndf, color = 'blue', label ='Normal Density Function')
plt.xlabel("Data Points")
plt.ylabel("Probability Density")
plt.title("Uniform and Normal Density functions with Mean = {} and Variance = {}".format(mean, variance))
plt.legend()
plt.show()
```

PLOT



Question-6

Code snippets are as follows:

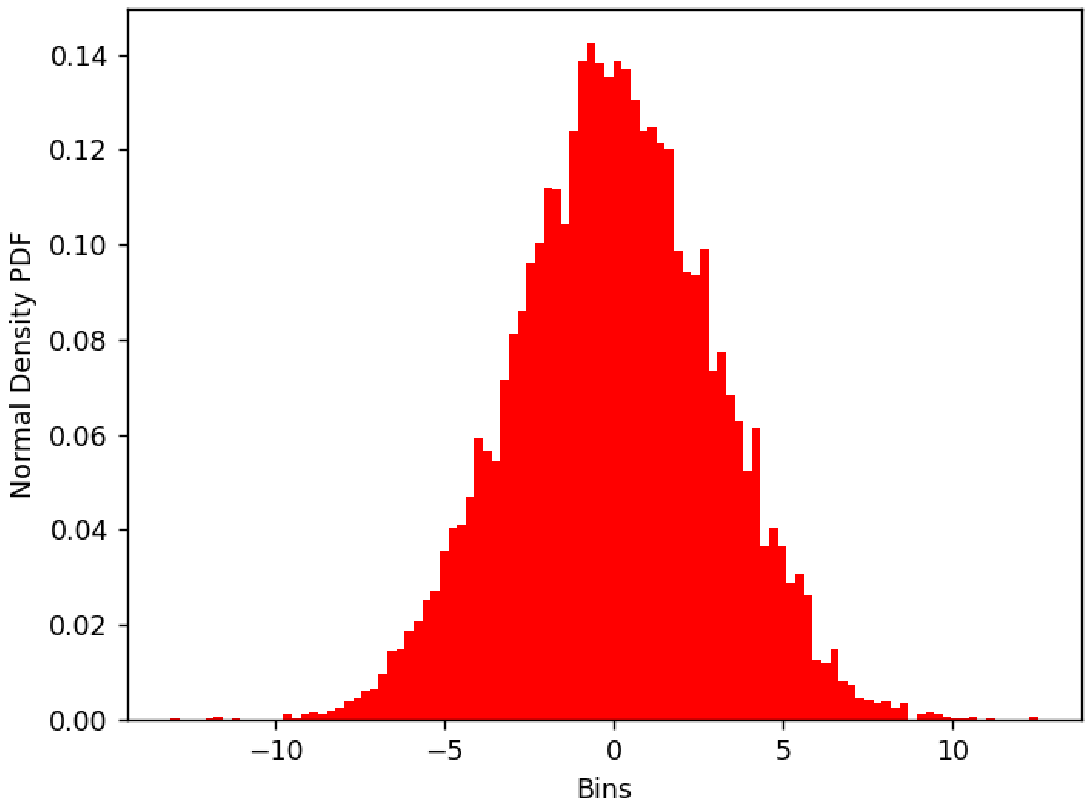
```
import math
import numpy as np
from matplotlib import pyplot as plt
from scipy.special import erfinv
from scipy.special import erf

N = 10000
data_points = np.random.uniform(0, 1, N)

```

```python
# a. Normal Density
sigma1 = 3
mu = 0
normal_icdf = sigma1*math.sqrt(2)*erfinv(2*data_points -1) + mu

# b. Rayleigh Density
sigma2 = 1.0
rayleigh_icdf = np.sqrt((2*( sigma2 )**2)* np.log(1/(1 - data_points)) )

# c. Exponential Density
lambd = 1.5
exponential_icdf = -np.log(1 - data_points)/lambd
```

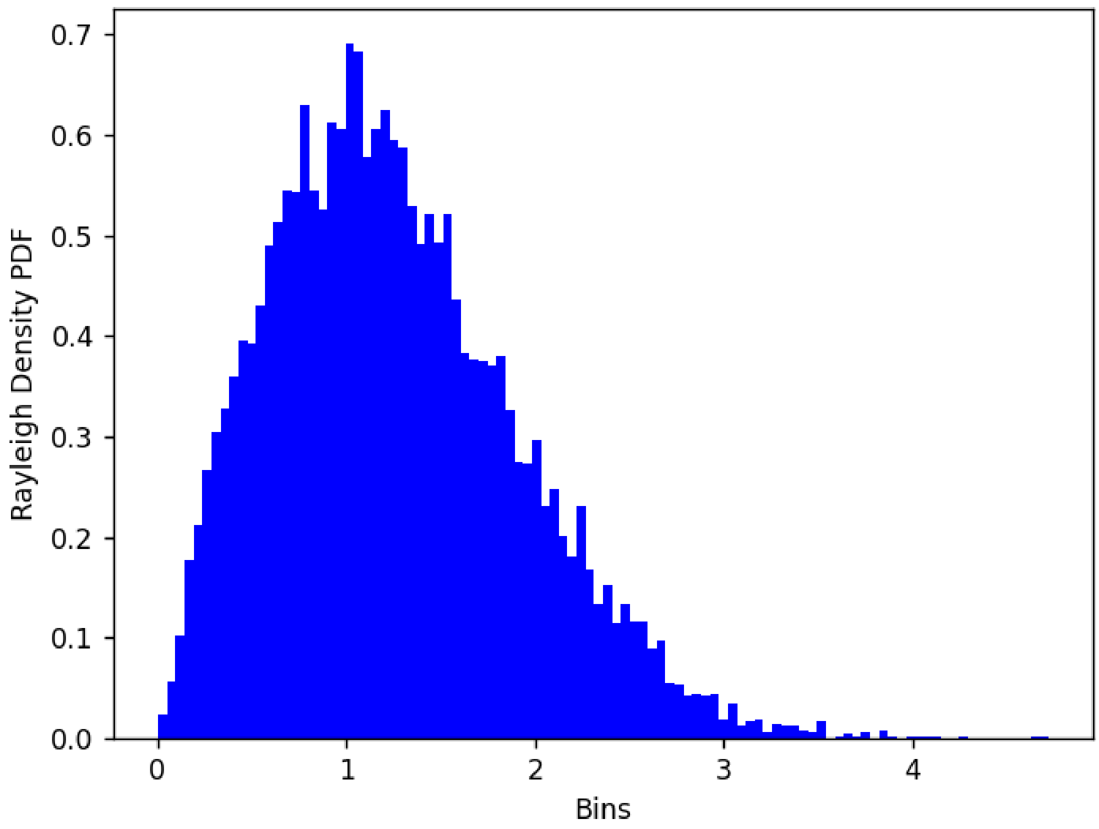
```python
## plot
plt.hist(normal_icdf, bins = 100, density = True, color = 'red', label = "Normal Density")
plt.xlabel("Bins")
plt.ylabel("Normal Density PDF")
plt.show()

plt.hist(rayleigh_icdf, bins =100, density = True, color = 'blue', label = "Rayleigh Density")
plt.xlabel("Bins")
plt.ylabel("Rayleigh Density PDF")
plt.show()

plt.hist(exponential_icdf, bins =100, density = True, color = 'green', label = "Exponential Density")
plt.xlabel("Bins")
plt.ylabel("Exponential Density PDF")
plt.show()
```

Plots are as follows:

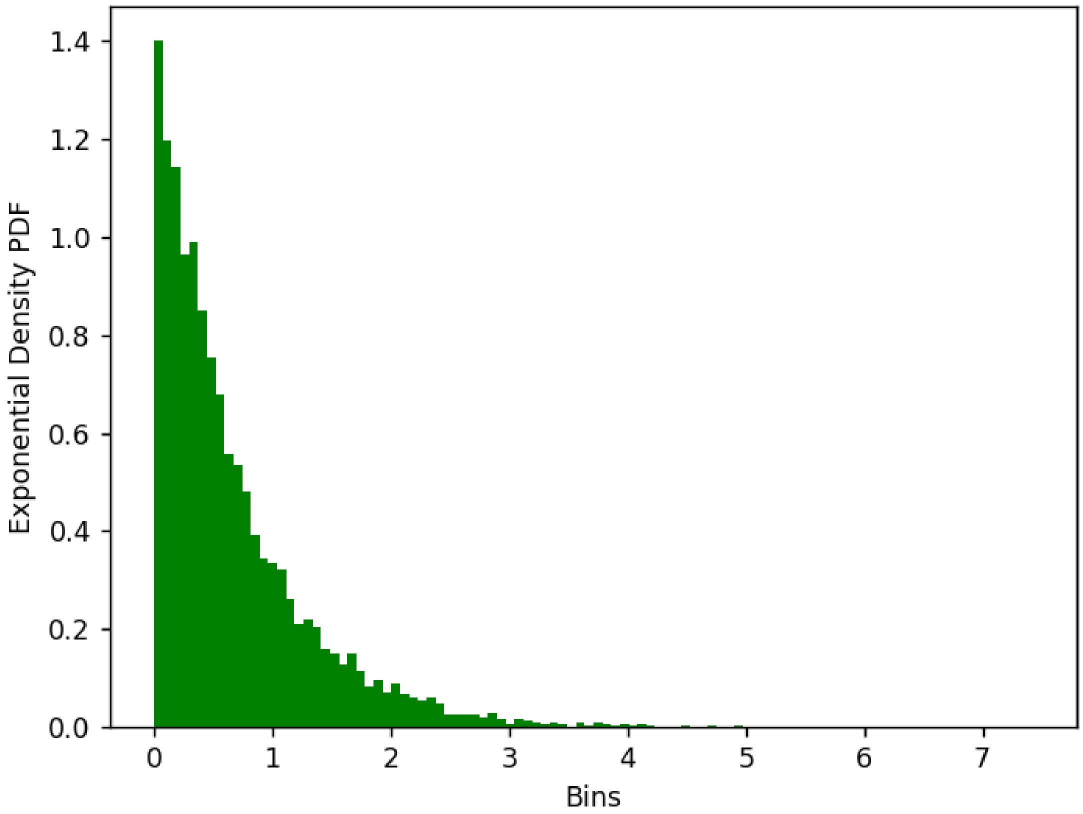
a. Normal Density, mean = 0, standard deviation = 3



b. Rayleigh Density, standard deviation = 1



c. Exponential Density, Lambda = 1.5



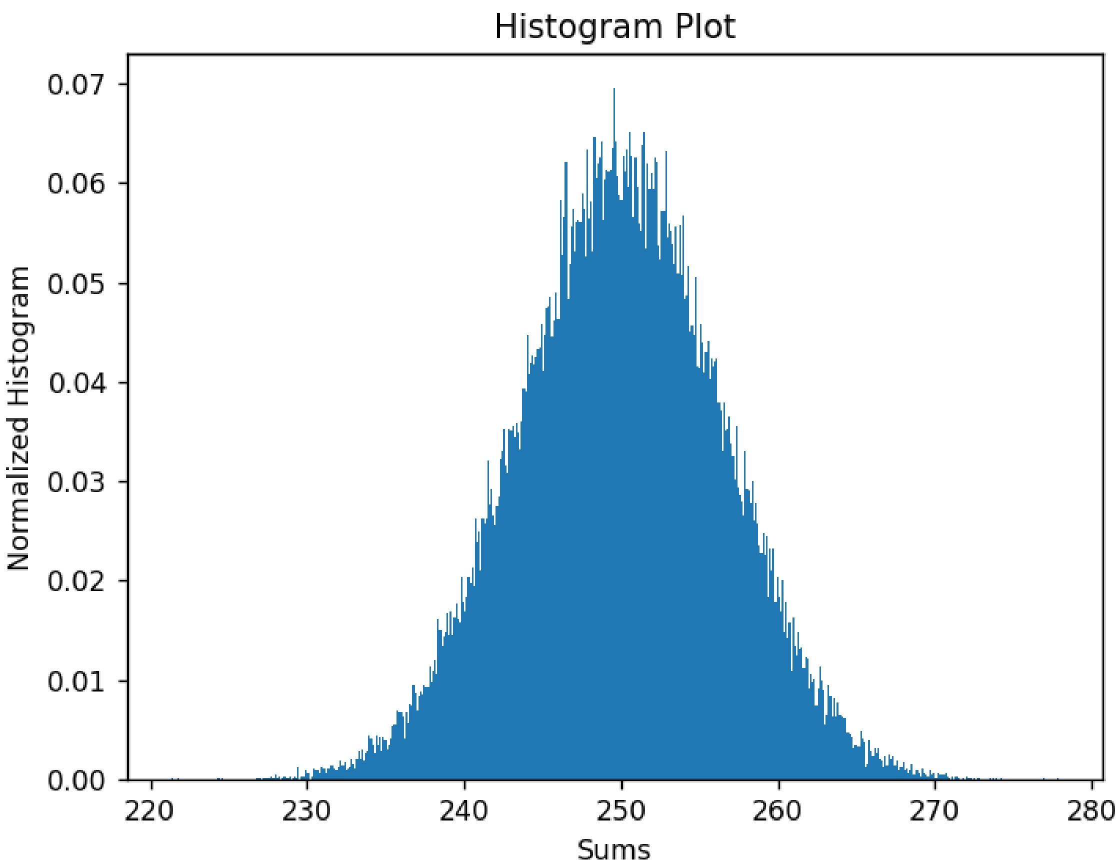
Inference: Inverse CDF can be used in mapping $U[0,1]$ to any set of random variables X with a PDF.

Question-7

Code snippet

```python
import numpy as np
import matplotlib.pyplot as plt
from numpy import random
from numpy.random import *

numGenerate = 500
N = 50000

sum_arr = [np.sum(np.random.uniform(0 ,1 ,500)) for i in range(N)]

plt.hist(sum_arr, bins = 500, density = True)
plt.title("Histogram Plot")
plt.xlabel("Sums")
plt.ylabel("Normalized Histogram")
plt.show()
```

PLOT



Inference: The above histogram plot looks similar to the one with a function of normal distribution, also suggesting the mean to lie somewhere in between the middle just like in the normal density function plot.