TRINAYAN DAS
180123051

In [3]:

```
pi=3.14


import math
import matplotlib.pyplot as plt
import random
import time
sample_1=math.sqrt(5)

t_c=[1000,100000]


def main(run):

 comp=0
 for t in t_c:
   if comp==1:
     print(" ********  ")
     print("      ")
   main1=[]
   print("The sample size is: ", 2*t)
   print("      ")
   if comp==1:
     comp=2
   elif comp==0:
     comp=1
   if run==1:
     print("BOX MULLER")
   if run==2:
     print("MARSAGLIA & BRAY")
   target=0
   total=0
   print("     ")
   start=time.time()
   while target<=t:
     if run==1:
       a=random.random()
       b=random.random()
       R=-2*math.log(a)
       V=2*pi*b
       Z1=math.sqrt(R)*math.cos(V)
       Z2=math.sqrt(R)*math.sin(V)
       main1.append(Z1)
       main1.append(Z2)
     if run==2:
       a=random.random()
       b=random.random()
       X=(2*a-1)*(2*a-1)+(2*b-1)*(2*b-1)
       total=total+1
       while X>1:
        a=random.random()
        b=random.random()
        X=(2*a-1)*(2*a-1)+(2*b-1)*(2*b-1)
        total=total+1
       k=-2*math.log(X)/X
       k=math.sqrt(k)
       Z1,Z2=(2*a-1)*k,(2*b-1)*k
       main1.append(Z1)
       main1.append(Z2)
```

```
    target=target+1
  sum=0
  if run==1:
    print("Time taken in seconds: ", time.time()-start)
  if run==2:
    print("Fraction rejected are: ", 1-t/total)
    print("Time taken in s", time.time()-start)
  for i in range(len(main1)):
    sum=sum+main1[i]
  print("Sample mean of the initial N(0,1): ", sum/len(main1))
  k=sum/len(main1)
  sum=0
  for i in range(len(main1)):
    sum=sum+(main1[i]-k)*(main1[i]-k)
  print("Sample variance of the initial N(0,1): ", sum/len(main1))
  print("     ")
  print("Frequency graph for the distribution")
  freq_l5=[]
  numpy_l6=[]
  for i in range(80):
    freq_l5.append(0)
    numpy_l6.append(-20+i*0.5)
  for i in range(len(main1)):
    if main1[i]>=20 or main1[i]<=20:
      z=math.floor(main1[i]+20)
      z2=main1[i]+20-z
      z2=z2*2
      z2=math.floor(z2)
      freq_l5[2*z+z2]=freq_l5[2*z+z2]+1
  plt.plot(numpy_l6,freq_l5)
  plt.show()
  test1=[]
  test2=[]
  print("     ")
  print("N(0,5), The actual density & sample density")
  for i in range(10000):
    run_file=-20+i/250
    test1.append(run_file)
    run_file=1/(sample_1*math.sqrt(2*pi))*math.exp((-0.5*(run_file)*(run_file))
/(sample_1*sample_1))
    test2.append(run_file)
  plt.plot(test1, test2)



  mainl1=[]
  mainl2=[]
  control_l3=[]
  control_l4=[]
  freq_l5=[]
  numpy_l6=[]
  numpy_l7=[]
  freq_l8=[]

  for i in range(200):
    freq_l5.append(0)
    freq_l8.append(0)
    numpy_l6.append(-20+i*0.2)
    numpy_l7.append(-15+i*0.2)
  for i in range(t*2):
```

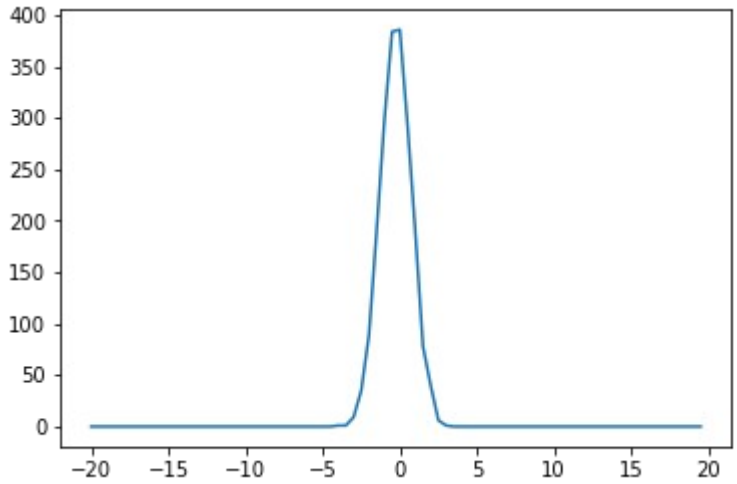
```
      control_l3.append(main1[i])
      control_l4.append(main1[i])
  for i in range(len(control_l3)):
    control_l3[i]=sample_1*control_l3[i]
    control_l4[i]=sample_1*control_l4[i]+5
  for i in range(len(control_l3)):
    if control_l3[i]>=-20 and control_l3[i]<=20:
      mainl1.append(control_l3[i])
    if control_l4[i]>=-15 and control_l4[i]<=25:
      mainl2.append(control_l4[i])
  for i in range(len(mainl1)):
    z=math.floor(mainl1[i]+20)

    z2=mainl1[i]+20-z
    z2=z2*5
    z2=math.floor(z2)

    freq_l5[5*z+z2]=freq_l5[5*z+z2]+1
  for i in range(len(mainl2)):
    z=math.floor(mainl2[i]+15)
    z2=mainl2[i]+15-z
    z2=z2*5
    z2=math.floor(z2)
    freq_l8[5*z+z2]=freq_l8[5*z+z2]+1
  a1=freq_l5[99]

  a2=0.178/a1
  a3=freq_l8[99]
  a4=0.178/a3
  for i in range(200):
    freq_l5[i]=freq_l5[i]*a2
    freq_l8[i]=freq_l8[i]*a4
  plt.plot(numpy_l6,freq_l5)
  plt.show()
  print("     ")
  print("N(5,5), The actual density & sample density")
  test1=[]
  test2=[]
  for i in range(10000):
    run_file=-20+i/250
    test1.append(run_file)
    run_file=1/(sample_1*math.sqrt(2*pi))*math.exp((-0.5*(run_file-5)*(run_file
-5))/(sample_1*sample_1))
    test2.append(run_file)
  plt.plot(test1, test2)
  plt.plot(numpy_l7,freq_l8)
  plt.show()
  print(" ")
  if comp==2:
    print(" ********  ")
    print("     ")
main(1)
main(2)
```

The sample size is: 2000

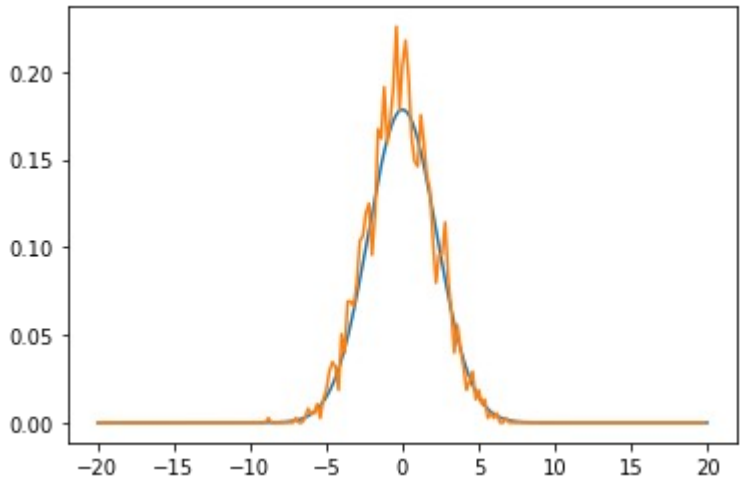
BOX MULLER

Time taken in seconds: 0.001798391342163086
Sample mean of the initial N(0,1): -0.010368895983279277
Sample variance of the initial N(0,1): 0.9744937264812379

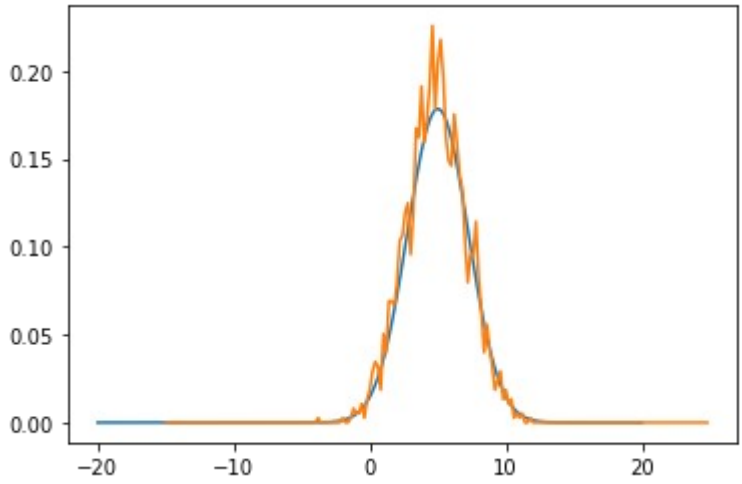
Frequency graph for the distribution



N(0,5), The actual density & sample density



N(5,5), The actual density & sample density



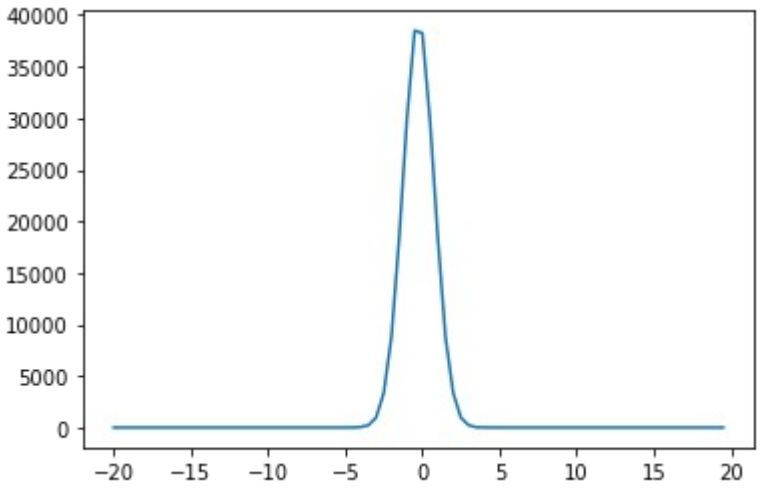
****

The sample size is: 200000

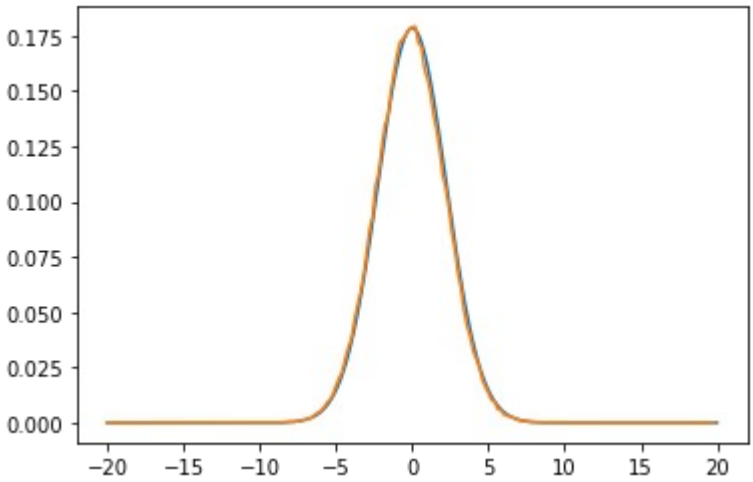
BOX MULLER

Time taken in seconds: 0.14984726905822754
Sample mean of the initial N(0,1): -0.0019196392381215 71
Sample variance of the initial N(0,1): 1.0008625818502632

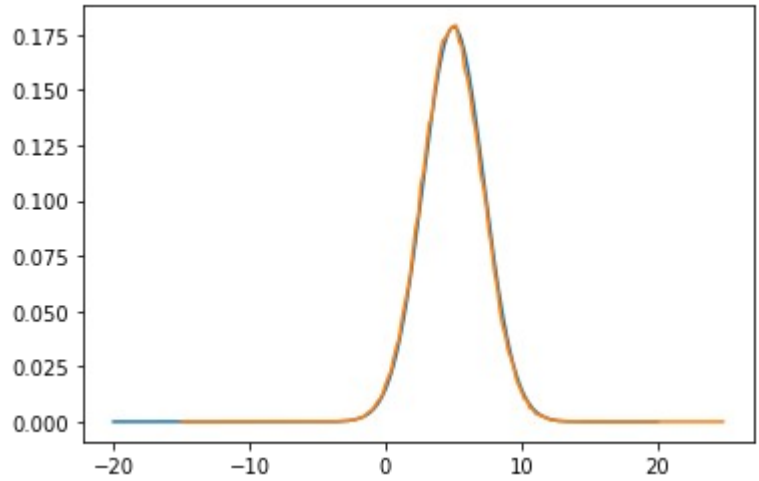
Frequency graph for the distribution



N(0,5), The actual density & sample density



N(5,5), The actual density & sample density



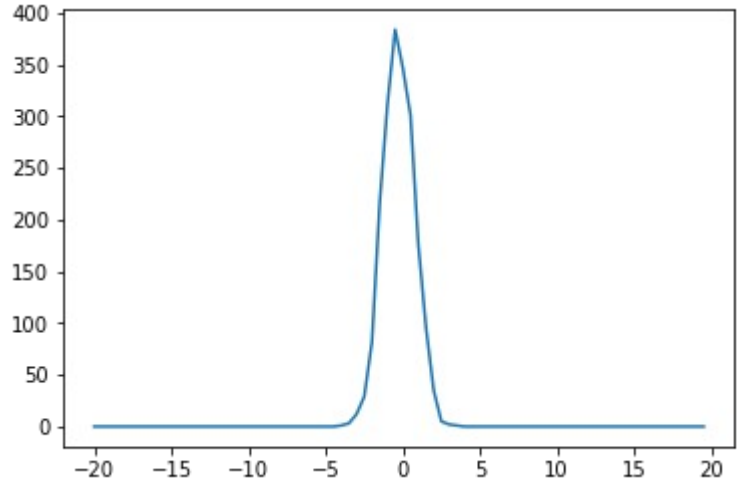
****

The sample size is: 2000

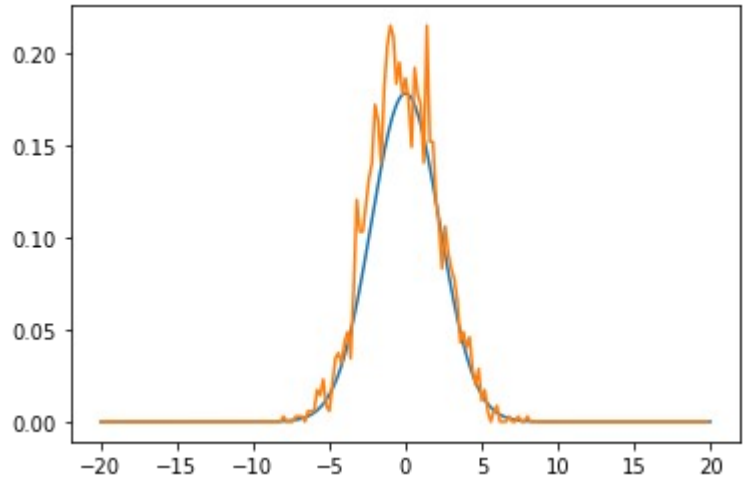
MARSAGLIA & BRAY

Fraction rejected are: 0.21073401736385167
Time taken in s 0.003022909164428711
Sample mean of the initial N(0,1): -0.02540435884582018
Sample variance of the initial N(0,1): 1.0308932491686353

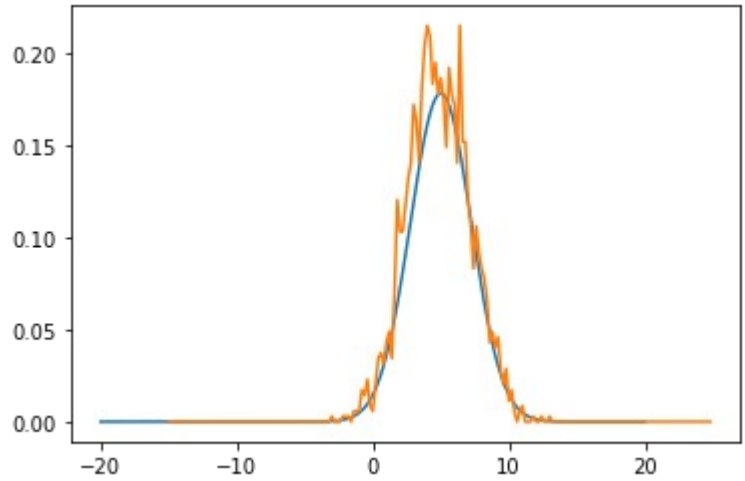
Frequency graph for the distribution



N(0,5), The actual density & sample density



N(5,5), The actual density & sample density



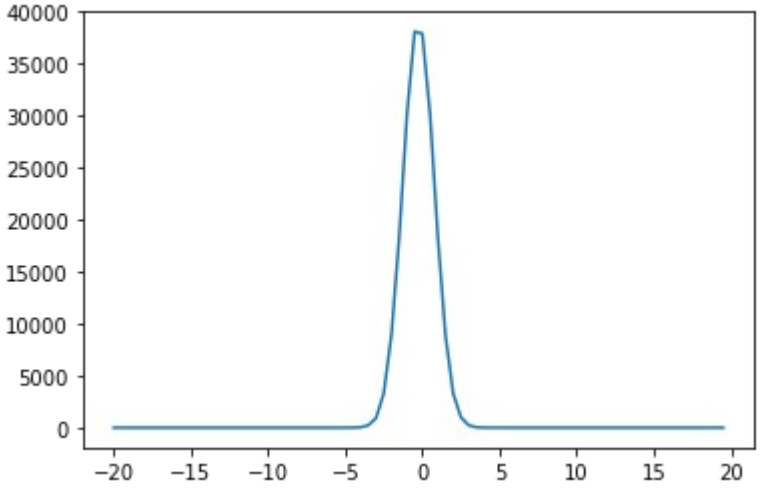
****

The sample size is: 200000

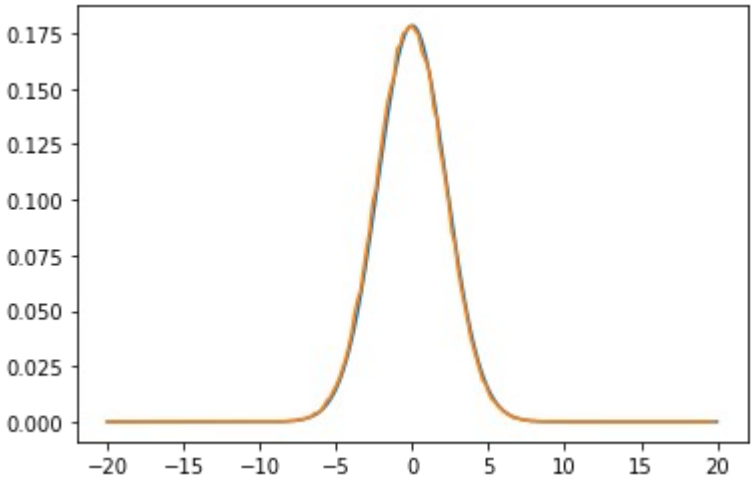
MARSAGLIA & BRAY

Fraction rejected are: 0.21430591784782682
Time taken in s 0.1890873908996582
Sample mean of the initial N(0,1): 0.0005288524657846653
Sample variance of the initial N(0,1): 1.0034881166658445

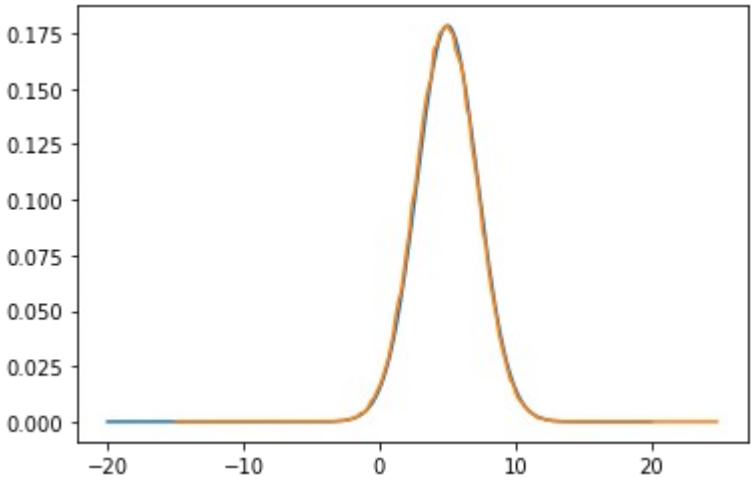
Frequency graph for the distribution



N(0,5), The actual density & sample density



N(5,5), The actual density & sample density



****

In [ ]: