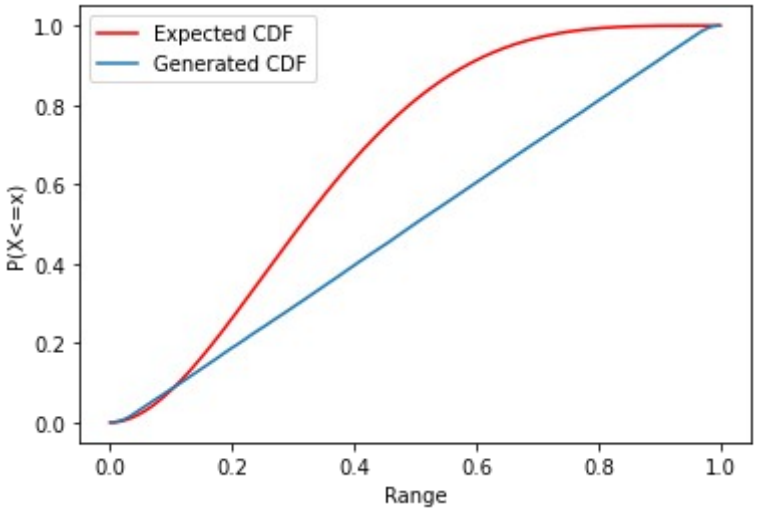
TRINAYAN DAS
180123051

In [1]:

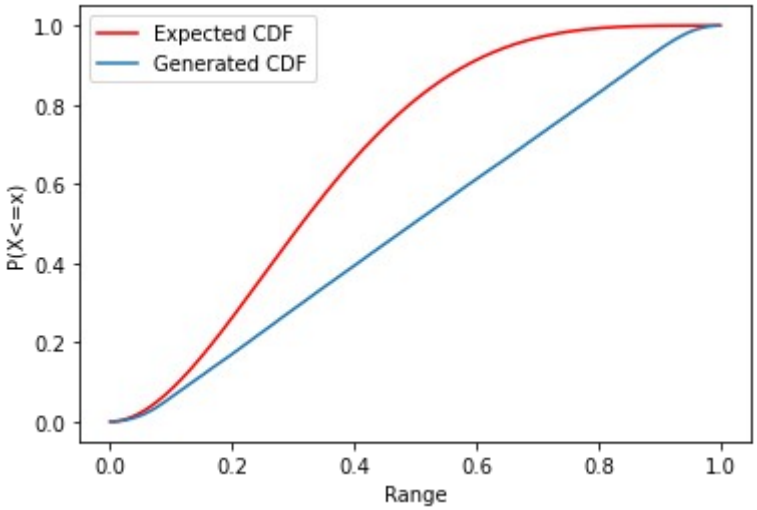
```
import numpy as np
import matplotlib.pyplot as plt
for num in range(0,4):
    c=[2.109375,5,10,100]
    j=100000
    u=np.random.rand(1,j)
    u=u.T
    g=np.random.rand(1,j)
    g=g.T
    y=[]
    it=[]
    for i in range(0,j):
        if u[i]<((20*g[i]*(1-g[i])*3)/c[num]):
            y.append(g[i])
            it.append(i)
    accepted_var=len(y)
    print(accepted_var)
    fit=[]
    fit.append(it[0])
    for i in range(1,len(it)):
        fit.append(it[i]-it[i-1])
    avg_of_iterations=(sum(fit) / len(fit))
    print(avg_of_iterations)

    x=np.linspace(0.0025,0.9975,399)
    z=np.linspace(0,1,401)

    Y=[]
    for i in range(0,401):
        Y.append( -4*(z[i]**5)+15*(z[i]**4)-20*(z[i]**3)+10*(z[i]**2))
    freq,bins=np.histogram(y,z)
    cfreq=np.zeros((401,1))
    cfreq[0][0]=freq[0]
    for i in range(1,400):
        cfreq[i][0]=cfreq[i-1][0]+freq[i]
    cfreq=cfreq/len(y)
    p1=plt.plot(z,Y,'-r')
    p2=plt.plot(x[0:399],cfreq[0:399])
    plt.xlabel('Range')
    plt.ylabel('P(X<=x)')
```

```
    plt.legend((p1[0], p2[0]), ('Expected CDF','Generated CDF'))
    plt.show()
```

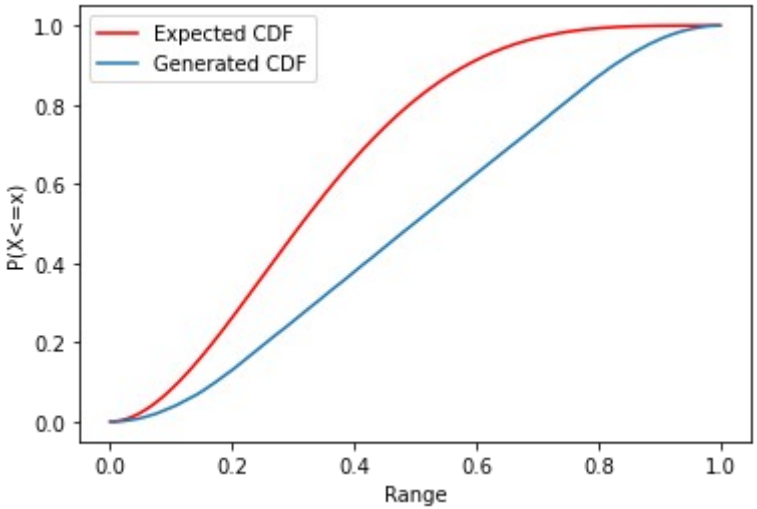
96463
1.0366565418865263



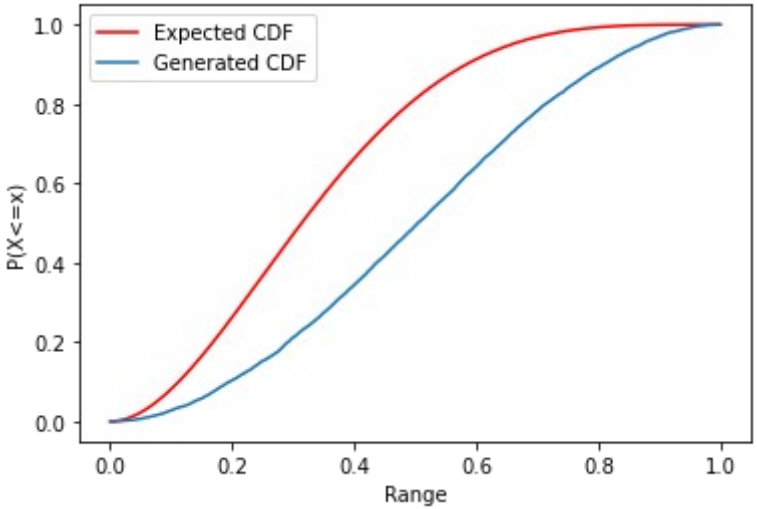
91189
1.0966125300200682



80594
1.2407747474998139




```
10031
9.968796730136576
```



In [ ]: