NEGATIVE BINOMIAL DISTRIBUTION

M.Aamani(S20180020219)
V.Haneesha(S20180010184)
K.Keerthana(S20180010084)

CODE FOR VARYING r

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

def factorial(k):
    if k==0:
        return 1
    return k*factorial(k-1)

k = []
for i in range(10,80):
    k.append(i)

r = []
for i1 in range(3,10):
    r.append(i1)

y=[]*10
p=0.2

for i in range(len(r)):
    y=[]
    for j in range(len(k)):
        A=factorial(k[j]-1)/(factorial(k[j]-r[i])*factorial(r[i]-1))
        B=(p**r[i])*(1-p)**(k[j]-r[i])
        num= A*B
        y.append(num)

    plt.bar(k,y,label='r={}'.format(r[i]))
plt.legend()
```

CODE FOR VARYING p

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

def factorial(k):
    if k==0:
        return 1
    return k*factorial(k-1)

k = []
for i in range(10,80):
    k.append(i)

r=10
y=[]*10
p=0.2
for i in range(4):
    y=[]
    for j in range(len(k)):
        A=factorial(k[j]-1)/(factorial(k[j]-r)*factorial(r-1))
        B=(p**r)*(1-p)**(k[j]-r)
        num= A*B
        y.append(num)

    plt.bar(k,y,label='p={}'.format(p))
    p=p+0.2


plt.legend()
```

CODE FOR MEAN AND VARIANCE for r=10

```
import numpy as np
import matplotlib.pyplot as plt
p=np.arange(0.2,1,0.1)
r=10
mean=r*(1-p)/(p)
var=r*(1-p)/(p*p)
plt.plot(p,mean,label='mean p=0.2')
plt.plot(p,var,label='varinace p=0.2')
plt.legend()
```

CODE FOR MEAN AND VARIANCE for p=0.2

```
import numpy as np
import matplotlib.pyplot as plt
r=np.arange(10,20,1)
p=0.2
mean=r*(1-p)/(p)
var=r*(1-p)/(p*p)
plt.plot(r,mean,label='mean p=0.2')
plt.plot(r,var,label='varinace p=0.2')
plt.legend()
```

CENTRAL LIMIT THEOREM VERIFICATION

```
p=0.2

ns=10
for i in range(4):

    samplemean=[]
    for j in range(ns):
        sum=0

        x = np.random.negative_binomial(10,0.2,n)
        for k in x:
            sum=sum+k
        samplemean.append(sum/n)
    fig, ax = plt.subplots(figsize =(10, 7))
    ax.hist(samplemean, bins ='auto')

    plt.title("NUMBER OF SAMPLES ={}".format(ns))
    plt.show()
```