Statistical Data Analysis (Assignment)

sat sun mon tue wed thu fri Date: 28/10/2022

1. (i) $E[a+bX] \cancel{\neq} = a + bE[X]$, when $a, b \in R$

Now, $E[a+bX] = \sum_x (a+bX)\, P(x)$

[suppose, pmf of X is $P(X)$]

$$= \sum_x a \cdot P(x) + bX \cdot P(x)$$

$$= \sum_x a \cdot P(x) + \sum_x bX \cdot P(x)$$

$$= a\left(\sum_x P(x)\right) + b\left(\sum_x x \cdot P(x)\right)$$

As we know, $E[X] = \sum_x x \cdot p(x)$ and $\sum_x P(x)$ is 1.

$$\therefore E[a+bX] = a + bE[X]$$

(proofed)

(ii) $Var(X) = E[X^2] - (E[X])^2$

As we know, $Var(X) = E[(X-\mu)^2]$

$$= E[(X) - E[X])^2]$$

$$= E[X^2 - 2XE[X] + (E[X])^2]$$

$$= E[X^2] - 2E[X]E[X] + (E[X])^2$$

$$= E[X^2] - (E[X])^2$$

(proofed)

(iii) $Var(a+bX) = b^2 Var(X) \quad a, b \in \mathbb{R}$

Now, $Var(a+bX) = \mathbb{E}\{[a+bX - \mathbb{E}(a+bX)]^2\}$

$[\because Var(Y) = E\{(Y-E(Y))^2\}]$

$= \mathbb{E}\{[a+bX - a - b\,\mathbb{E}(X)]^2\}$

$= \mathbb{E}\{[bX - b\mathbb{E}(X)]^2\}$

$= \mathbb{E}\{b^2(X - \mathbb{E}(X))^2\}$

$= b^2 \mathbb{E}\{(X - E(X))^2\}$

$= b^2 Var(X)$ (proofed)

(iv) $Var(a) = 0$, $\quad a \in \mathbb{R}$

We know that, Expectation of a constant value, is itself. So, $\mathbb{E}(a) = a$.

As we know, $Var(a) = \mathbb{E}(a^2) - (\mathbb{E}(a))^2$

$= a^2 - a^2$

$= 0$ (proofed)

2.

(i) Given that,

$$S_n^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X}_n)^2$$

$$= \frac{1}{n-1}\left(\sum_{i=1}^{n}(x_i^2 - 2x_i\bar{x}_n + \bar{x}_n^2)\right)$$

$$= \frac{1}{n-1}\left(\sum_{i=1}^{n}x_i^2 - \sum_{i=1}^{n}2x_i\bar{x}_n + \sum_{i=1}^{n}\bar{x}_n^2\right)$$

$$= \frac{1}{n-1}\left(\sum_{i=1}^{n}x_i^2 - 2\bar{x}_n n\bar{x}_n + n\bar{x}_n^2\right)$$

$$= \frac{1}{n-1}\left(\sum_{i=1}^{n}x_i^2 - n\bar{x}_n^2\right)$$

(ii) We know,

$$E(X_i) = \mu \quad \text{and,} \quad Var(X) = \sigma^2$$

$$E(\sum X_i) = \sum(E(X_i))$$

$$E(cX) = cE(X)$$

$$Var(X) = E(X^2) - [E(X)]^2$$

$$\sigma^2 = E(X^2) - \mu^2$$

$$\therefore E(X^2) = \sigma^2 + \mu^2$$

And, $\quad Var(\bar{x}) = E(\bar{x}^2) - [E(\bar{x})]^2$

$$\frac{\sigma^2}{n} = E(\bar{x}^2) - \mu^2$$

$$\therefore E(\bar{x}^2) = \frac{\sigma^2}{n} + \mu^2 .$$

Now, $E\left[\sum (x_i - \bar{x})^2\right]$

$$= E\left[\sum (x_i^2 - 2x_i\bar{x} + \bar{x}^2)\right]$$

$$= E\left[\sum x_i^2 - 2\bar{x}\, n\bar{x} + n\bar{x}^2\right]$$

$$= E\left[\sum x_i^2 - n\bar{x}^2\right]$$

$$= \sum E(x_i^2) - E(n\bar{x}^2)$$

$$= \sum (\sigma^2 + \mu^2) - n\left(\frac{\sigma^2}{n} + \mu^2\right)$$

$$= n\sigma^2 + n\mu^2 - \sigma^2 - n\mu^2$$

$$= (n-1)\sigma^2$$

~~We know that~~, $E[s_n^2] = E\left[\dfrac{\sum (x_i - \bar{x})^2}{n-1}\right]$

$$= \frac{1}{n-1} E\left[\sum (x_i - \bar{x})^2\right]$$

$$= \frac{1}{n-1} (n-1)\sigma^2$$

$$= \sigma^2 \quad \text{(proofed)}$$

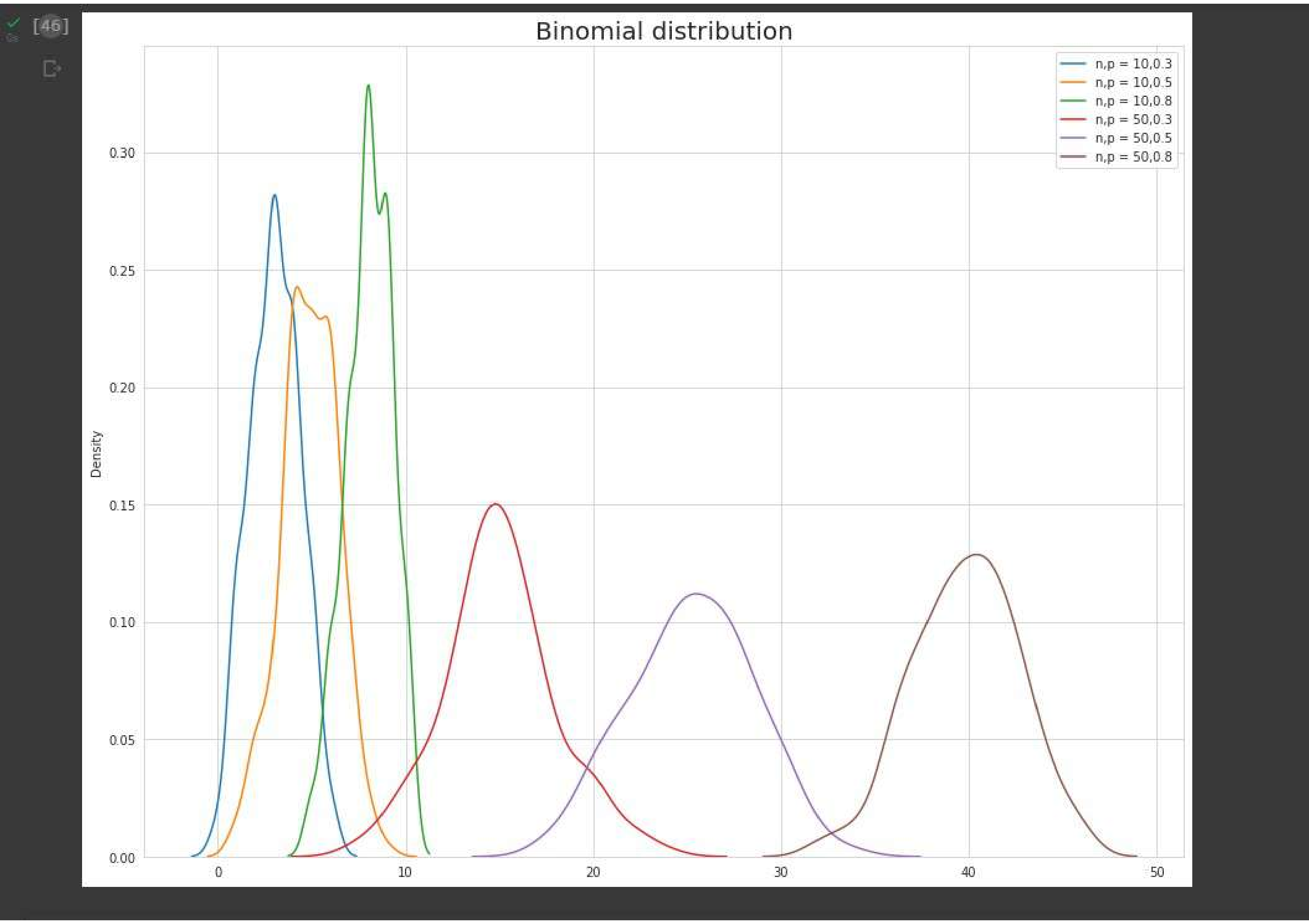
Statistical Data Analysis assignment -1

Excercise 3

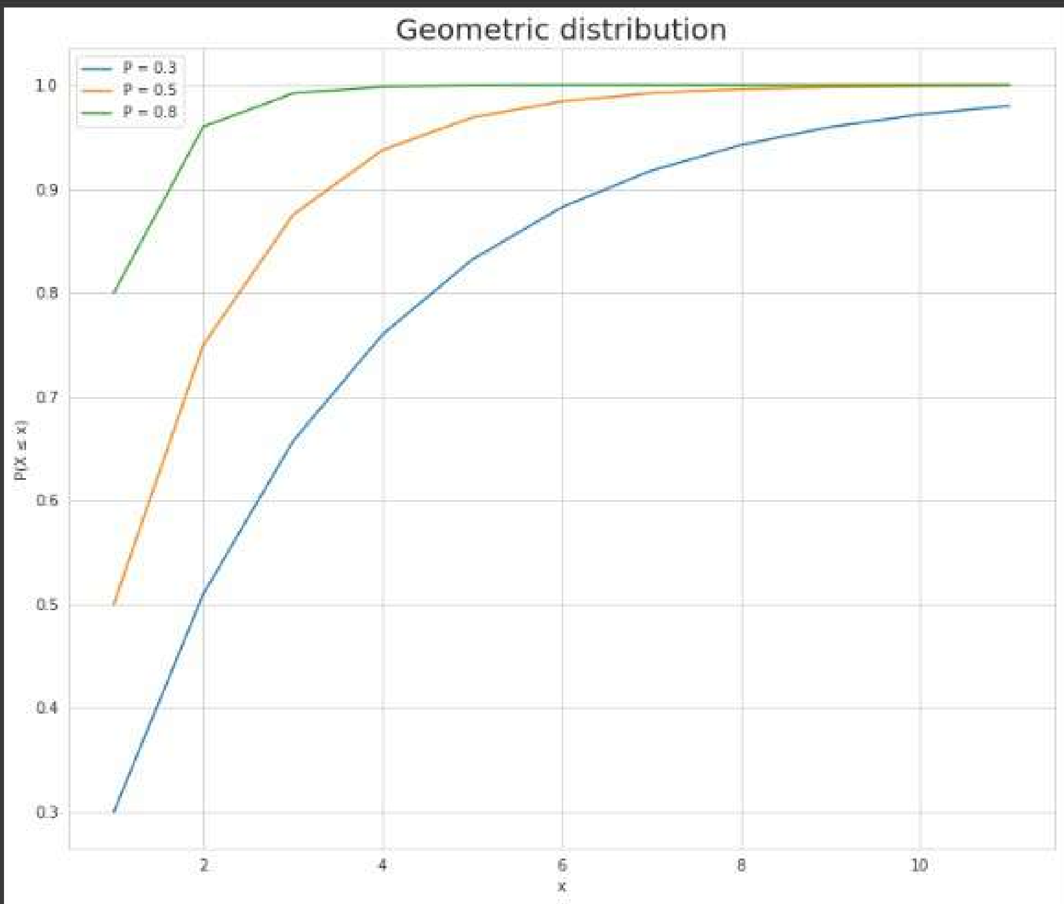
1. Plot the probability of a random variable that follows the Binomial distribution Bin(n,P) for different $P \in \{0.3, 0.5, 0.8\}$ and $n \in \{10,50\}$

[1]
```
#importing libraries
import matplotlib.pyplot as plt
%matplotlib inline
import numpy as np
import scipy
from scipy.stats import geom
import seaborn as sns
```

[46]
```
#setting random seed
np.random.seed(9)
size = 200
nb = [10,50]
pb = [0.3,0.5,0.8]
sns.color_palette("Set2")
sns.set_style("whitegrid")
#creating and plotting the distributions
for i in nb:
    for k in pb:
        sns.kdeplot(np.random.binomial(i, k, size), label = "n,p = {},{}".format(i,k))

plt.legend()
plt.title("Binomial distribution", fontsize = 20)
plt.rcParams['figure.figsize'] = [15, 12]
plt.show();
```


[46]
Binomial distribution
n,p = 10,0.3
n,p = 10,0.5
n,p = 10,0.8
n,p = 50,0.3
n,p = 50,0.5
n,p = 50,0.8
Density
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0
10
20
30
40
50

```
[48] #importing geometric distribution
     from scipy.stats import geom
     import matplotlib.pyplot as plt
     import numpy as np

     x = np.arange(1, 12,1)

     for p in [0.3, 0.5, 0.8]:
         plt.plot(x, geom.cdf(x, p), label = "P = {}".format(p))
     plt.ylabel("P(X ≤ x)")
     plt.xlabel("x")
     plt.title("Geometric distribution", fontsize = 20)
     plt.legend()
     plt.rcParams['figure.figsize'] = [12, 10]
     plt.show();
```



[50]
```
def poisson(x,lmbda):
    return lmbda**x* np.exp(-lmbda)/scipy.special.factorial(x)
x = np.arange(1,17)
for l in [0.3, 2, 6]:
    plt.plot(x,poisson(x,l),label=f'$\lambda$ = {l}')
plt.ylabel("P(X=x)")
plt.xlabel("x")
plt.legend()
plt.title("Poisson distribution", fontsize = 20)
plt.rcParams['figure.figsize'] = [12, 10]
plt.show()
```