Question 1

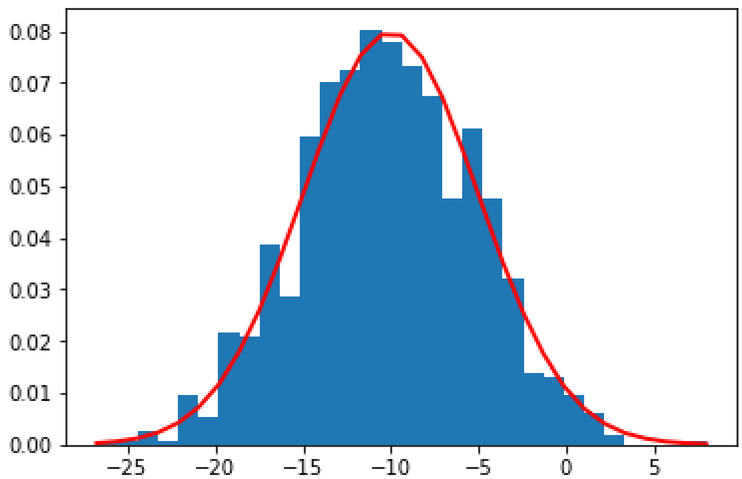
Create 1000 samples from a Gaussian distribution with mean -10 and standard deviation 5. Create another 1000 samples from another independent Gaussian with mean 10 and standard deviation 5.

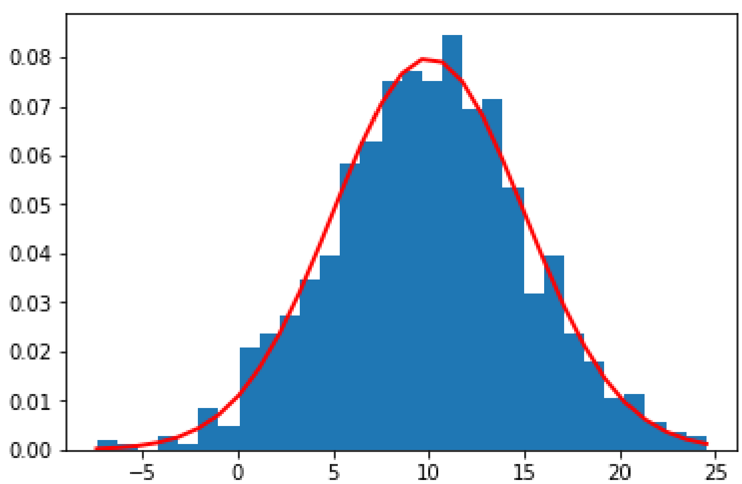
(a) Take the sum of 2 these Gaussians by adding the two sets of 1000 points, point by point, and plot the histogram of the resulting 1000 points. What do you observe?

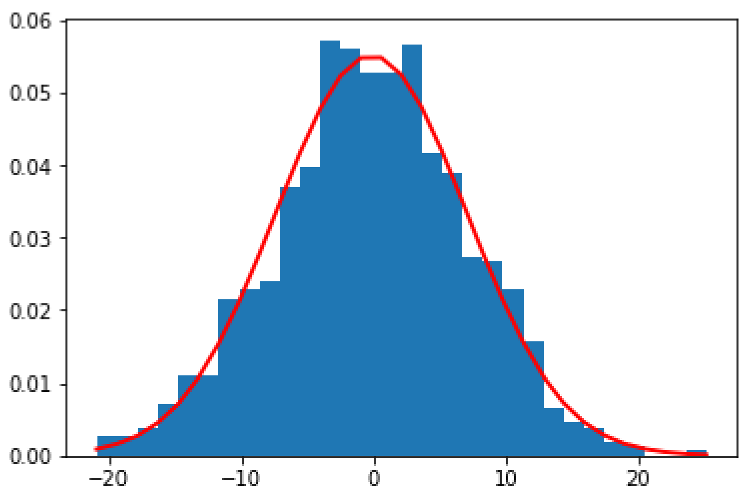
(b) Estimate the mean and the variance of the sum.

In [17]:
```
import matplotlib.pyplot as plt
import numpy as np
import matplotlib.mlab as mlab

mu, sigma = -10, 5 # mean and standard deviation
s = np.random.normal(mu, sigma, 1000)

mu2, sigma2 = 10, 5 # mean and standard deviation
s2 = np.random.normal(mu2, sigma2, 1000)

count, bins, ignored = plt.hist(s, 30, normed=True)
plt.plot(bins, 1/(sigma * np.sqrt(2 * np.pi)) * np.exp( - (bins - mu)**2
 / (2 * sigma**2) ),linewidth=2, color='r')
plt.show()

count, bins, ignored = plt.hist(s2, 30, normed=True)
plt.plot(bins, 1/(sigma2 * np.sqrt(2 * np.pi)) * np.exp( - (bins - mu2)*
*2 / (2 * sigma2**2) ),linewidth=2, color='r')
plt.show()



s3 = s + s2
mu3 = s3.sum()/s3.size
sigma3 = s3.std()
mu3_var = s3.var()


count, bins, ignored = plt.hist(s3, 30, normed=True)
plt.plot(bins, 1/(sigma3 * np.sqrt(2 * np.pi)) * np.exp( - (bins - mu3)*
*2 / (2 * sigma3**2) ),linewidth=2, color='r')
plt.show()

print("The estimated mean is :",mu3)
print("The estimated variance is :",mu3_var)
```

```
/anaconda3/lib/python3.6/site-packages/matplotlib/axes/_axes.py:6462: U
serWarning: The 'normed' kwarg is deprecated, and has been replaced by
the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```





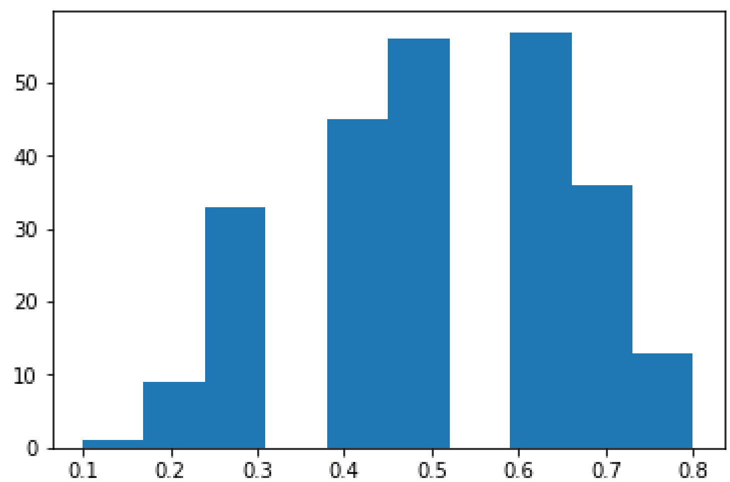



```
The estimated mean is : -0.18282693942272818
The estimated variance is : 52.334462063516604
```

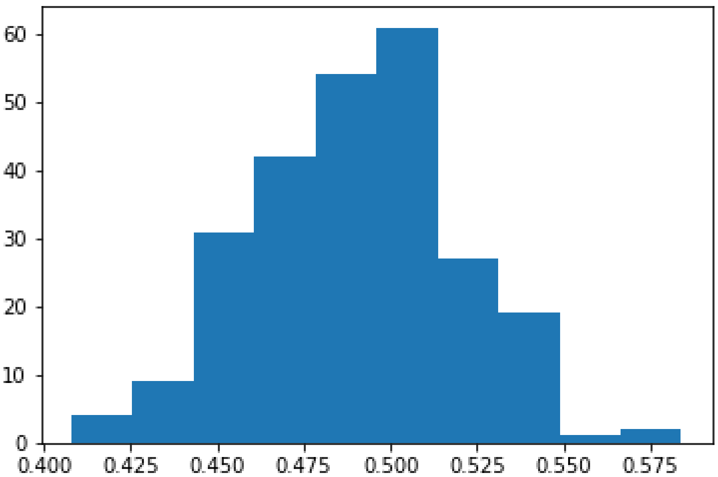
Question 2

Central Limit Theorem. Let Xi be an iid Bernoulli random variable with value {-1,1}. Look at the random variable Zn = 1 □ Xi. By taking 1000 draws from Zn, plot its histogram. n Check that for small n (say, 5-10) Zn does not look that much like a Gaussian, but when n is bigger (already by the time n = 30 or 50) it looks much more like a Gaussian. Check also for much bigger n: n = 250, to see that at this point, one can really see the bell curve.

In [4]:
```
import random
import numpy as np
import matplotlib.pyplot as plt

binomial_samples = np.random.binomial(size = 1000, n = 1, p = 0.5)
samples = [np.mean(random.choices(binomial_samples, k = 10)) for _ in ra
nge(250)]
plt.hist(samples)
plt.show()
```



In [5]:
```
binomial_samples = np.random.binomial(size = 1000, n = 1, p = 0.5)
samples = [np.mean(random.choices(binomial_samples, k = 250)) for _ in r
ange(250)]
plt.hist(samples)
plt.show()
```



Question 3

Estimate the mean and standard deviation from 1 dimensional data: generate 25,000 samples from a Gaussian distribution with mean 0 and standard deviation 5. Then estimate the mean and standard deviation of this gaussian using elementary numpy commands, i.e., addition, multiplication, division (do not use a command that takes data and returns the mean or standard deviation).

In [7]:

```
import numpy as np

mu, sigma = 0, 5 # mean and standard deviation
sample = np.random.normal(mu, sigma, 25000)

estimated_mean = sample.sum()/sample.size
print("The estimated mean is :",estimated_mean)

estimated_variance = sum(pow(x-estimated_mean,2) for x in sample) / samp
le.size
print("The estimated variance is :",estimated_variance)

estimated_std = np.sqrt(estimated_variance)
print("The estimated standard deviation is :",estimated_std)
```

```
The estimated mean is : -0.024751599473765592
The estimated variance is : 24.663817819155504
The estimated standard deviation is : 4.966267997113678
```

Question 4

Estimate the mean and covariance matrix for multi-dimensional data: generate 10,000 samples of 2 dimensional data from the Gaussian distribution Xi Yi ~ N –5 5 , 20 .8 .8 30 .

(1) Then, estimate the mean and covariance matrix for this multi-dimensional data using elementary numpy commands, i.e., addition, multiplication, division (do not use a command that takes data and returns the mean or standard deviation).

In [99]:
```
import numpy as np

samples = np.array(([20,.8],[.8,30]))
print('List')

number = np.random.multivariate_normal([5,-5],samples,10000)
print(number)

estimated_mean = sum(number) / len(number)
print('The estimated Mean is',estimated_mean)

estimated_variance = np.sum((number - estimated_mean)**2,axis=0) / 10000
print("The estimated variance is :",estimated_variance)


estimated_covariance = np.multiply(*(number - estimated_mean).T).sum() /
 9999
print("The estimated covariance is :",estimated_covariance)

print("Covariance matrix is {}".format([[estimated_variance[0], estimate
d_covariance], [estimated_covariance, estimated_variance[1]]]))

n = [x - estimated_mean for x in number]
covariance_matrix = np.matmul( np.transpose(n), n) / 9999
print("The estimated covariance using matmul is :",covariance_matrix)
```

```
List
[[  3.11009474   0.40876771]
 [ 12.99919485 -12.26232351]
 [  9.58684442 -11.56425375]
 ...
 [  6.12729268  -1.81861621]
 [  6.03220957  -8.24918797]
 [  1.76441483   3.2996818 ]]
The estimated Mean is [ 4.97993269 -5.01403295]
The estimated variance is : [20.05730781 29.68609556]
The estimated covariance is : 0.7934479043078843
Covariance matrix is [[20.05730781187715, 0.7934479043078843], [0.79344
79043078843, 29.68609556155429]]
The estimated covariance using matmul is : [[20.05931374  0.7934479 ]
 [ 0.7934479  29.68906447]]
```

Method 2 :

In [84]:
```
mean = [-5,5]
cov = [[20, 0.8], [0.8, 30]];

sample = np.random.multivariate_normal(mean, cov, 10000)
estimated_mean = np.sum(sample, axis=0) / np.size(sample, axis=0)
print("The estimated mean is : ", estimated_mean)

numbers = [x - mean for x in sample]
covariance_matrix = np.matmul( np.transpose(numbers), numbers) / 9999

print("The covariance matrix is :", covariance_matrix)
```

```
The estimated mean is :  [-4.94798231  5.0118986 ]
The covariance matrix is : [[20.48170762  0.62789166]
 [ 0.62789166 30.2038061 ]]
```

Question 5

Each row is a patient and the last column is the condition that the patient has. Do data exploration using Pandas and other visualization tools to understand what you can about the dataset.

(a) How many patients and how many features are there?

In [4]:
```
import pandas as pd

df = pd.read_csv('/Users/aparnaaidith/Desktop/My Python Projects/BIG DAT
A/PatientData.csv',header=None,na_values=['?'])
```

In [5]:
```
no_of_patients = df.shape[0]
print("The total number of patients : ",no_of_patients)
no_of_features = df.shape[1]
print("The total number of features : ",no_of_features)
```

```
The total number of patients :  452
The total number of features :  280
```

(b) What is the meaning of the first 4 features? See if you can understand what they mean.

1) Feature - Age of the patient

2) Feature - Gender of the patient 0 : Male 1 : Female

3) Feature - Height of the patient

4) Feature - Weight of the patient

(c) Are there missing values? Replace them with the average of the corresponding feature column

In [6]:
```
df[13]
```

```
Out[6]: 0        NaN
        1        NaN
        2       23.0
        3        NaN
        4        NaN
        5        NaN
        6        NaN
        7        NaN
        8       84.0
        9        NaN
        10       NaN
        11       NaN
        12       NaN
        13       NaN
        14       NaN
        15       NaN
        16       NaN
        17       NaN
        18       NaN
        19       NaN
        20       NaN
        21       NaN
        22       NaN
        23       NaN
        24       NaN
        25       NaN
        26       NaN
        27       NaN
        28       NaN
        29     160.0
              ...
        422      NaN
        423      NaN
        424    103.0
        425      NaN
        426    -84.0
        427      NaN
        428      NaN
        429      NaN
        430    -44.0
        431      NaN
        432      NaN
        433      NaN
        434      NaN
        435      NaN
        436      NaN
        437      NaN
        438      NaN
        439      NaN
        440    -90.0
        441      NaN
        442      NaN
        443      NaN
        444      NaN
        445      NaN
        446      NaN
        447      NaN
```

```
448      NaN
449     84.0
450    103.0
451      NaN
Name: 13, Length: 452, dtype: float64
```

In [7]:
```
df.fillna(df.mean(),inplace=True)
```

In [8]:
```
df[13]
```

```
Out[8]: 0      -13.592105
        1      -13.592105
        2       23.000000
        3      -13.592105
        4      -13.592105
        5      -13.592105
        6      -13.592105
        7      -13.592105
        8       84.000000
        9      -13.592105
        10     -13.592105
        11     -13.592105
        12     -13.592105
        13     -13.592105
        14     -13.592105
        15     -13.592105
        16     -13.592105
        17     -13.592105
        18     -13.592105
        19     -13.592105
        20     -13.592105
        21     -13.592105
        22     -13.592105
        23     -13.592105
        24     -13.592105
        25     -13.592105
        26     -13.592105
        27     -13.592105
        28     -13.592105
        29     160.000000
                  ...
        422    -13.592105
        423    -13.592105
        424    103.000000
        425    -13.592105
        426    -84.000000
        427    -13.592105
        428    -13.592105
        429    -13.592105
        430    -44.000000
        431    -13.592105
        432    -13.592105
        433    -13.592105
        434    -13.592105
        435    -13.592105
        436    -13.592105
        437    -13.592105
        438    -13.592105
        439    -13.592105
        440    -90.000000
        441    -13.592105
        442    -13.592105
        443    -13.592105
        444    -13.592105
        445    -13.592105
        446    -13.592105
        447    -13.592105
```

```
448    -13.592105
449     84.000000
450    103.000000
451    -13.592105
Name: 13, Length: 452, dtype: float64
```

(d) How could you test which features strongly influence the patient condition and which do not?

In [9]:
```
df.corr()[279].sort_values(ascending = False)
```

```
Out[9]: 279    1.000000
        90     0.368876
        4      0.323879
        92     0.313982
        102    0.282523
        223    0.235488
        233    0.218811
        17     0.195198
        29     0.183083
        94     0.174346
        52     0.173243
        125    0.170670
        191    0.165693
        68     0.152534
        239    0.151782
        77     0.143284
        152    0.141506
        221    0.141274
        56     0.141103
        113    0.140502
        119    0.132195
        181    0.130360
        149    0.130056
        228    0.128490
        65     0.127210
        198    0.125873
        249    0.124823
        193    0.122928
        107    0.121711
        88     0.120726
                 ...
        8     -0.122003
        208   -0.134657
        161   -0.135180
        202   -0.142731
        252   -0.150610
        172   -0.158536
        247   -0.159612
        260   -0.162153
        270   -0.164321
        168   -0.171763
        1     -0.178080
        242   -0.189458
        162   -0.197555
        19          NaN
        67          NaN
        69          NaN
        83          NaN
        131         NaN
        132         NaN
        139         NaN
        141         NaN
        143         NaN
        145         NaN
        151         NaN
        156         NaN
        157         NaN
```

```
164          NaN
204          NaN
264          NaN
274          NaN
Name: 279, Length: 280, dtype: float64
```

Written Questions

Question 1

Consider two random variables X,Y that are not independent. Their probabilities of are given by the following table:

X=1 Y=0 1/4 Y=1 1/3

(a) What is the probability that X = 1?
(b) What is the probability that X = 1 conditioned on Y = 1?
(c) What is the variance of the random variable X?
(d) What is the variance of the random variable X conditioned that Y = 1?
(e)WhatisE[$X_3+X_2+3Y_7$|Y =1|]?

SOLUTION

Question 1 :-

	X = 0	X = 1	
Y = 0	1/4	1/4	1/2
Y = 1	1/6	1/3	1/2
	5/12	7/12	1

(a) What is the probability that $X = 1$?

$$\frac{7}{12}$$

(b) What is the probability that $X = 1$ conditioned on $Y = 1$?

$$P(X=1 \mid Y=1) = \frac{P(X=1 \cap Y=1)}{P(Y=1)}$$

$$\frac{1}{3} \times \frac{2}{3} = \frac{2}{3}$$

(c) What is the variance of random variable X?

$$\mu_x = \left(0 \times \frac{5}{12}\right) + \left(1 \times \frac{7}{12}\right) = \frac{7}{12}$$

$$\left(0 - \frac{7}{12}\right)^2 \times \frac{1}{4} + \left(0 - \frac{7}{12}\right)^2 \times \frac{1}{6} + \left(1 - \frac{7}{12}\right)^2 \times \frac{1}{4}$$

$$+ \left(1 - \frac{7}{12}\right)^2 \times \frac{1}{3} = \frac{35}{144}$$

$$\sigma_X^2 = E\left[(x - \mu)^2\right]$$

(d) What is the variance of the random variable X conditioned that $Y=1$?

$$\mu = 0 \times \frac{1}{6} + 1 \times \frac{1}{3} = 0 + \frac{1}{3} = \frac{1}{3}$$

σ_x^2 conditioned $Y=1$:-

$$\left(0-\frac{1}{3}\right)^2 \times \frac{1}{6} + \left(1-\frac{1}{3}\right)^2 \times \frac{1}{3}$$

$$= \left(\frac{1}{9} \times \frac{1}{6}\right) + \left(\frac{4}{9} \times \frac{1}{3}\right)$$

$$= \frac{1}{9}\left(\frac{1}{6} + \frac{4}{3}\right) = \frac{1}{9}\left(\frac{1+8}{6}\right)$$

$$= \frac{1}{9} \times \frac{9}{6} = \frac{1}{6}$$

(e) What is $E[X^3 + X^2 + 3Y^7 \mid Y = 1]$?

$E[c] = c$ is constant

$E[c \cdot u[x]] = c \cdot E[U[x]]$.

$$P(X=0 \mid Y=1) = \frac{P(X=0 \text{ and } Y=1)}{P(Y=1)}$$

$$= \frac{1}{6} \times \frac{2}{1} = \frac{1}{3}$$

$$P(X=1 \mid Y=1) = \frac{P(X=1 \text{ and } Y=1)}{P(Y=1)}$$

$$= \frac{1}{3} \times \frac{2}{1} = \frac{2}{3}$$

$E[X^3 + X^2 + 3Y^7 \mid Y = 1]$

$$\Rightarrow E[X^3 \mid Y=1] = 0^3 \times \frac{1}{3} + 1^3 \times \frac{2}{3} = \frac{2}{3}$$

$$E[X^2 \mid Y=1] = 0^2 \times \frac{1}{3} \times 1^2 \times \frac{2}{3} = \frac{2}{3}$$

$$E[3Y^7 \mid Y=1] = 3 \cdot E[Y^7 \mid Y=1]$$

$$= 3 \cdot E[Y=1 \mid Y=1]$$

$$= 3.$$

$$\frac{2}{3} + \frac{2}{3} + 3 = \frac{2+2+9}{3} = \frac{13}{3}$$

NOTE: I have a calculation mistake on question (b) of Question 1.So I am submitting that particular question again.

WRITTEN QUESTIONS

Question 1 :-

(b) What is the probability that $X = 1$ conditioned on $Y = 1$?

$$P(X = 1 \mid Y = 1) = \frac{P(X = 1 \cap Y = 1)}{P(Y = 1)}$$

$$\frac{1}{3} \times \frac{2}{1} = \frac{2}{3}$$

Question 2

Consider the vectors v1 = [1, 1, 1] and v2 = [1, 0, 0]. These two vectors define a 2-dimensional subspace of R3. Project the points P1 = [3,3,3],P2 = [1,2,3],P3 = [0,0,1] on this subspace. Write down the coordinates of the three projected points. (You can use numpy or a calculator to do arithmetic if you want).

SOLUTION

I have used the calculator to perform the matrix multiplic

Question 2 :-

$V_1 = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$ $\quad V_2 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$

$P_1 = \begin{bmatrix} 3 & 3 & 3 \end{bmatrix}$ $\quad P_2 = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}$ $\quad P_3 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$

$V_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$ $\quad V_2 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$

$$U = \begin{bmatrix} V_1 & V_2 \end{bmatrix}$$

$$\pi_U(X) = B\lambda$$

$$B = \begin{bmatrix} V_1 \,|\, V_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 1 & 0 \end{bmatrix}$$

$$\boxed{\lambda = (B^T B)^{-1} B^T X}$$

$$X = \begin{bmatrix} 3 & 1 & 0 \\ 3 & 2 & 0 \\ 3 & 3 & 1 \end{bmatrix}$$

$$B^T X = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 3 & 1 & 0 \\ 3 & 2 & 0 \\ 3 & 3 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 9 & 6 & 1 \\ 3 & 1 & 0 \end{bmatrix}$$

$$B^T B = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 3 & 1 \\ 1 & 1 \end{bmatrix}$$

$$(B^T B)^{-1} = \begin{bmatrix} 1/2 & -1/2 \\ -1/2 & 3/2 \end{bmatrix}$$

$$(B^T B)^{-1} B^T X = \begin{bmatrix} 1/2 & -1/2 \\ -1/2 & 3/2 \end{bmatrix} \begin{bmatrix} 9 & 6 & 1 \\ 3 & 1 & 0 \end{bmatrix}$$

$$\lambda = \begin{bmatrix} 3 & 5/2 & 1/2 \\ 0 & -3/2 & -1/2 \end{bmatrix}$$

$$\pi_u[x] = B\lambda$$

$$= \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 3 & 5/2 & 1/2 \\ 0 & -3/2 & -1/2 \end{bmatrix} = \underline{\begin{bmatrix} 3 & 1 & 0 \\ 3 & 5/2 & 1/2 \\ 3 & 5/2 & 1/2 \end{bmatrix}}$$

$$c_1 = \begin{bmatrix} 3 & 3 & 3 \end{bmatrix}$$

$$c_2 = \begin{bmatrix} 1 & 5/2 & 5/2 \end{bmatrix}$$

$$c_3 = \begin{bmatrix} 0 & 1/2 & 1/2 \end{bmatrix}$$

Question 3

Consider a coin such that probability of heads is 2/3. Suppose you toss the coin 100 times. Estimate the probability of getting 50 or fewer heads. You can do this in a variety of ways. One way is to use the Central Limit Theorem. Be explicit in your calculations and tell us what tools you are using in these.

SOLUTION

Question 3:-

$P(\text{Heads}) = \frac{2}{3}$ Toss the coin 100 times.

$P(\text{getting 50 or fewer heads}) = ?$

Let

$$X_i = \begin{cases} 1, & \text{if } i^{th} \text{ toss is heads} \\ 0, & \text{otherwise.} \end{cases}$$

Since X_i are i.i.d and binomially distributed we know,

$$\mu = EX_i$$
$$= np$$
$$= 1 \times \frac{2}{3} = \underline{\underline{\frac{2}{3}}}$$

$$\sigma^2 = var(X_i)$$
$$= np(1-p)$$
$$= 1 \times \frac{2}{3}\left(1 - \frac{2}{3}\right) = \underline{\underline{\frac{2}{9}}}$$

By central limit theorem,

$$P(S_{100} \le 50) = P\left(\frac{S_{100} - 100\mu}{\sqrt{100\sigma^2}} \le \frac{50 - \frac{200}{3}}{\sqrt{\frac{200}{3}}}\right)$$

$$= P\left(\frac{S_{100} - 100\mu}{\sqrt{100\sigma^2}} \le \frac{-50}{\sqrt{200}}\right)$$

$$= P\left(\frac{S_{100} - 100\mu}{\sqrt{100\sigma^2}} \le -3.5\right)$$

$$\approx 1 - \phi(3.5)$$

From z-score table value of $z = 3.5$ is equal to 0.9998

$$= 1 - 0.9998$$

$$= 0.0002$$

I used calculator and z score table for the calculation purpose.