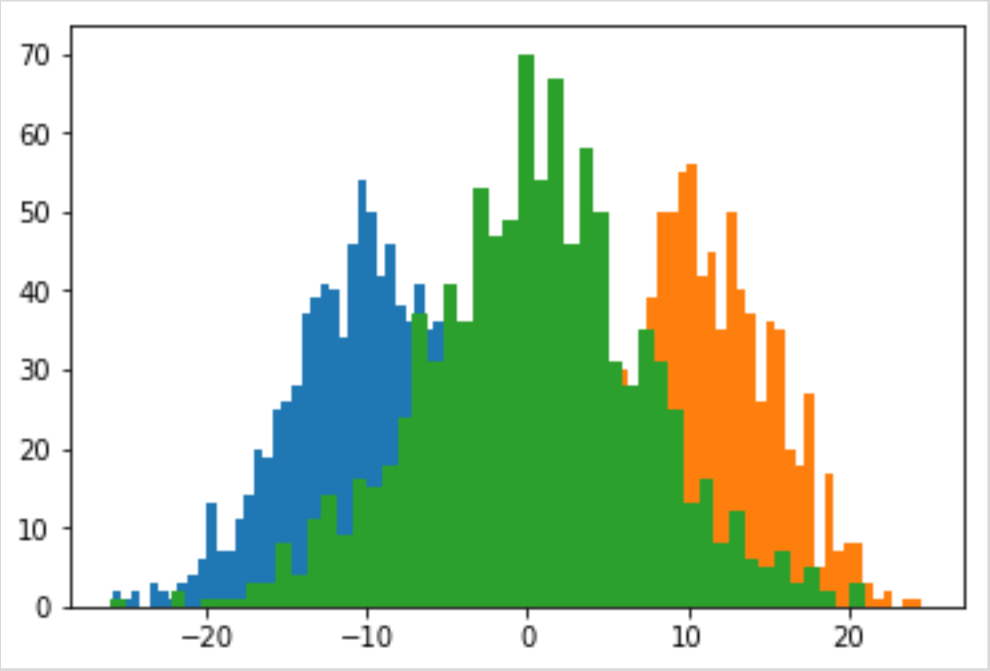
EE 379K Lab Report

by: Pratyush Singh (pks629) and George Doykan (gd7448)

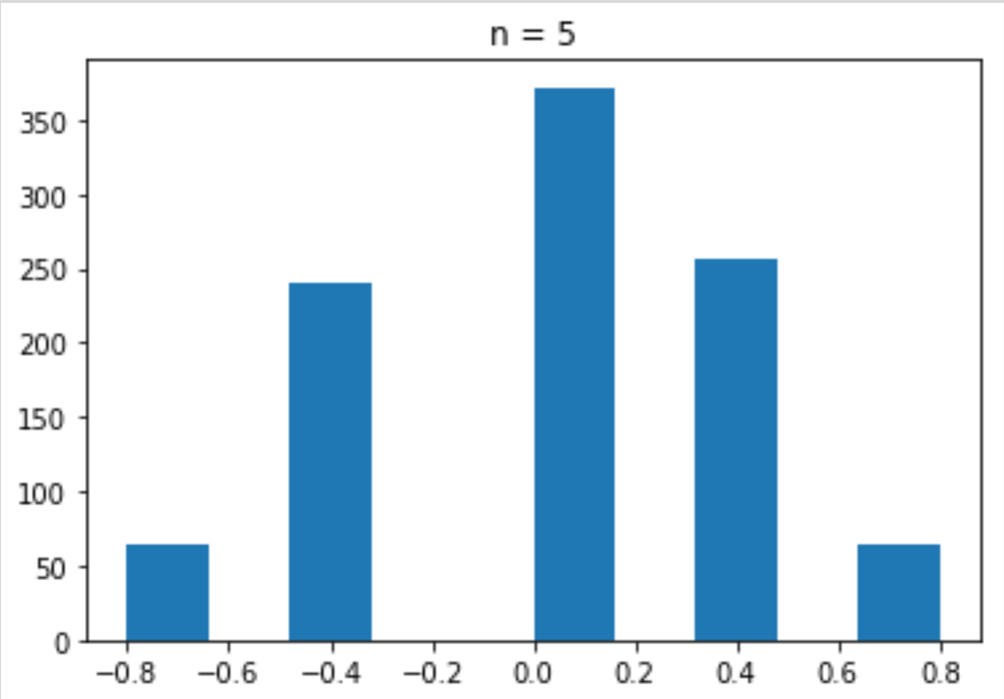
```python
#Problem 1
import numpy as np
import matplotlib.pyplot as plt

#creating the distributions
mean = -10
std = 5
distribution_one = np.random.normal(mean, std, 1000)
plt.hist(distribution_one, 50)

mean = 10
distribution_two = np.random.normal(mean, std, 1000)
plt.hist(distribution_two, 50)

#sum the 2 normal distributions
distributionSum = distribution_one + distribution_two
plt.hist(distributionSum, 50)

plt.show()

print("The mean of the sum is: " + str(np.mean(distributionSum)))
print("The variance of the sum is: " + str(np.var(distributionSum)))
```

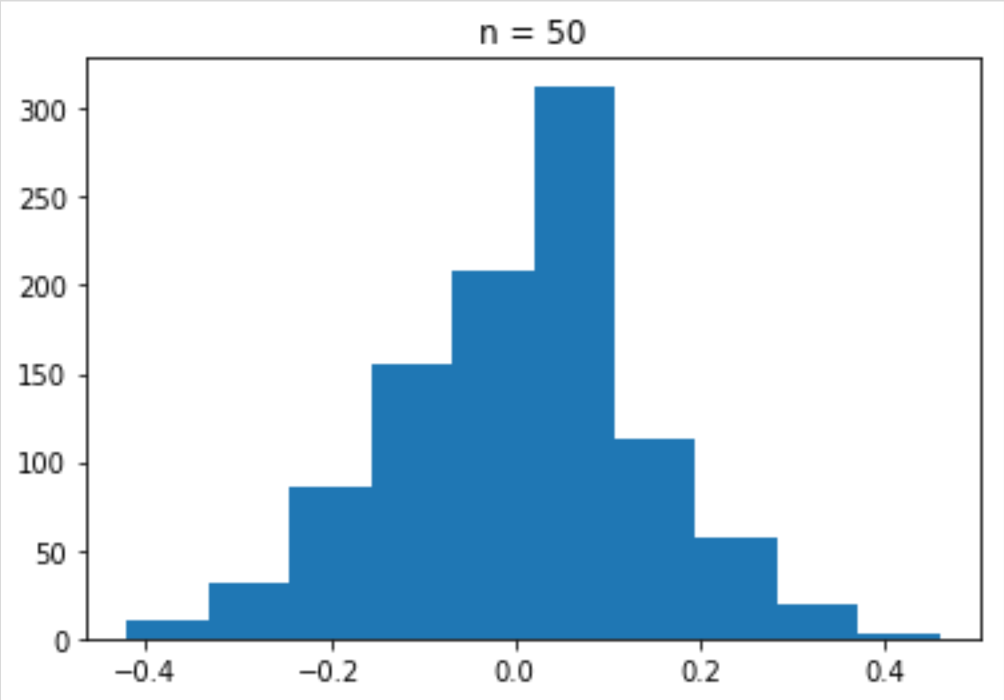


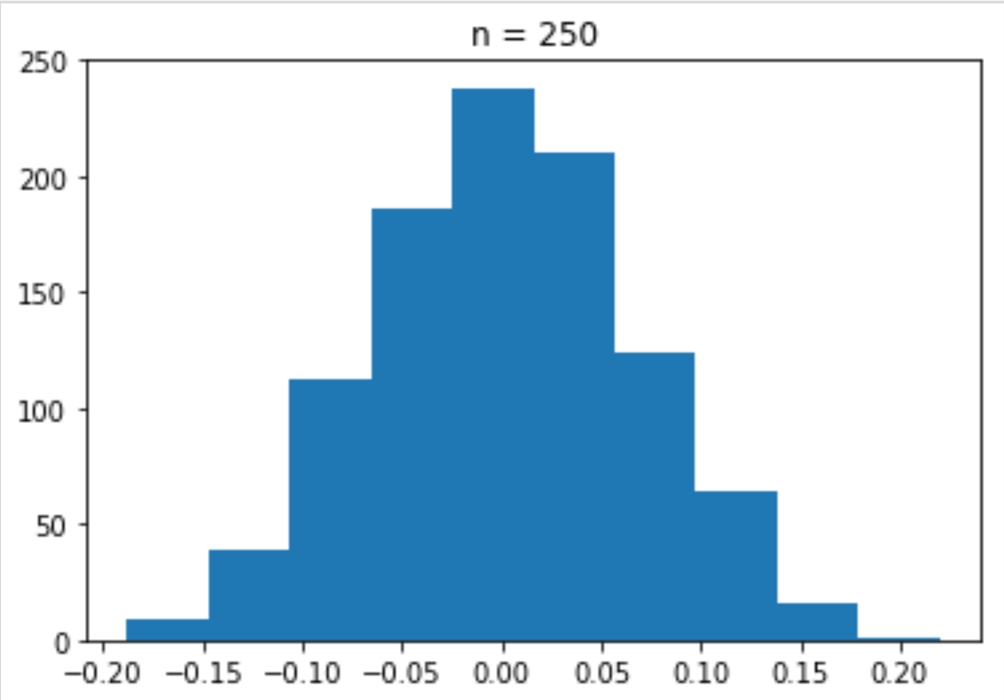

```
The mean of the sum is: 0.34913336396
The variance of the sum is: 48.577546012
```

```
#Problem 2
X = [-1,1]
Z = []
Z1 = []
Z2 = []

for i in range(1, 1000):
    sum = 0
    sum1 = 0
    sum2 = 0
    for j in range(1, 250):
        bernoulli = np.random.choice(a=X, p=[.50,.50])
        sum = sum + bernoulli
    Z.append(sum/250)

    for k in range(1, 5):
        bern = np.random.choice(a=X, p=[.50,.50])
```

```
        sum1 = sum1 + bern
    Z1.append(sum1/5)

    for l in range(1, 50):
        b = np.random.choice(a=X, p=[.50,.50])
        sum2 = sum2 + b
    Z2.append(sum2/50)



#print(Z)
plt.hist(Z1)
plt.title("n = 5")
plt.show()
plt.hist(Z2)
plt.title("n = 50")
plt.show()
plt.hist(Z)
plt.title("n = 250")
plt.show()
```








```
#Problem 3
x = np.random.normal(0, 5, 25000)


#mean
sum = np.sum(x)
average = sum/25000
print("The average is: " + str(average))
```

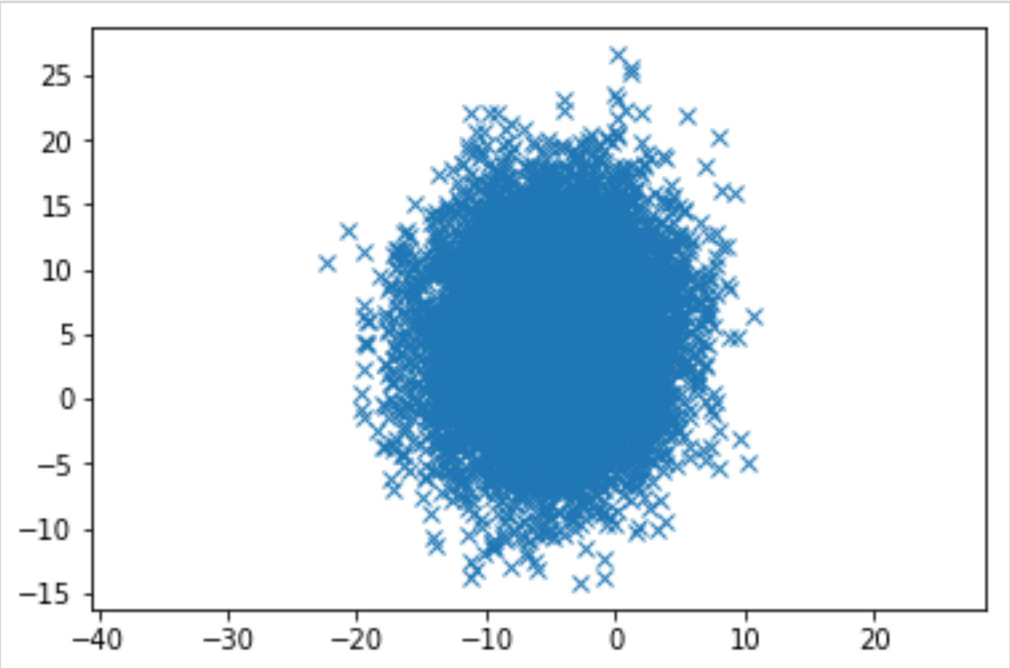
```
#std
diff = 0
for i in x:
    temp = i - average
    temp = temp ** 2
    diff = diff + temp

std = (diff/(25000 - 1)) ** 0.5 #take the square root
print("The standard deviation: " + str(std))
```

```
The average is: 0.0142500652467
The standard deviation: 4.98379361444
```

```
#Problem 4
mean = [-5, 5]
cov = [[20, 0.8], [0.8, 30]]
x, y = np.random.multivariate_normal(mean, cov, 10000).T

#estimating mean for x
sum_x = np.sum(x)
average_x = sum_x/10000
print("The estimated mean for x vals is: " + str(average_x))

#estimating mean for y
sum_y = np.sum(y)
average_y = sum_y/10000
print("The estimated mean for the y vals is: " + str(average_y))

#calculate covariance
sum = 0
for i in range(0, 10000):
```

```
    sum = sum + ((x[i] - average_x) * (y[i] - average_y))

cov = 1/(10000 - 1) * sum
print("The estimated covariance is: " + str(cov))

plt.plot(x, y, 'x')
plt.axis('equal')
plt.show()
```

```
The estimated mean for x vals is: -4.95178856717
The estimated mean for the y vals is: 4.93243286274
The estimated covariance is: 0.912986137926
```




```
#Problem 5
import pandas as pd

patientDF = pd.read_csv('/Users/pratyushsingh/Data Science Lab EE379K/
PatientData.csv', header=None, na_values=["?"])
```

```
#data exploration to figure the features for the
print(patientDF.info())

print()
print()

print("The max for the first column: " + str(np.max(patientDF[0])))
print("The min for the first column: " + str(np.min(patientDF[0])))
print("The average for the first column: " + str(np.mean(patientDF[0])))

print()
print()

print("The max for the third column: " + str(np.max(patientDF[2])))
print("The min for the third column: " + str(np.min(patientDF[2])))
print("The average for the third column: " + str(np.mean(patientDF[2])))
print("The median for the third column: " + str(np.median(patientDF[2])))
index  = patientDF[2].idxmax()
minIndex = patientDF[2].idxmin()
print("The age for the patient with the max value in the third column: "
      + str(patientDF.iloc[index][0]))
print("The age for the patient with the min value in the third column: "
      + str(patientDF.iloc[minIndex][0]))

print()
print()

print("The max for the fourth column: " + str(np.max([3])))
print("The min for the fourth column: " + str(np.min(patientDF[3])))
print("The average for the fourth column: " + str(np.mean(patientDF[3])))
print("The median for the fourth column: " + str(np.median(patientDF[3])))
index  = patientDF[3].idxmin()
maxIndex = patientDF[3].idxmax()
print("The age for the patient with the min value in the fourth column: "
      + str(patientDF.iloc[index][0]))
print("The age for the patient with the max value in the fourth column: "
      + str(patientDF.iloc[maxIndex][0]))
print("The gender for the patient with the max value in the fourth column: "
      + str(patientDF.iloc[maxIndex][1]))
```

```
#5c missing values
null_data = patientDF[patientDF.isnull().any(axis=1)] #rows with missing data
patientDF = patientDF.fillna(patientDF.mean()) #fill missing values with
average

#finding the values with the three highest correlations
firstCorr = secondCorr = thirdCorr = float('-Inf')
first = second = third = float('-Inf')

for i in range(0, 279):
    corr = patientDF[i].corr(patientDF[279])
    if(corr > firstCorr):
        firstCorr = corr
        first = i
    elif(corr > secondCorr):
        secondCorr = corr
        second = i
    elif(corr > thirdCorr):
        thirdCorr = corr
        third = i
print()
print()

#the columns with the three highest correlated features

print("The columns with the three highest correlated features are: " +
str(first) + ', ' + str(second) + ', ' +
      str(third))
```

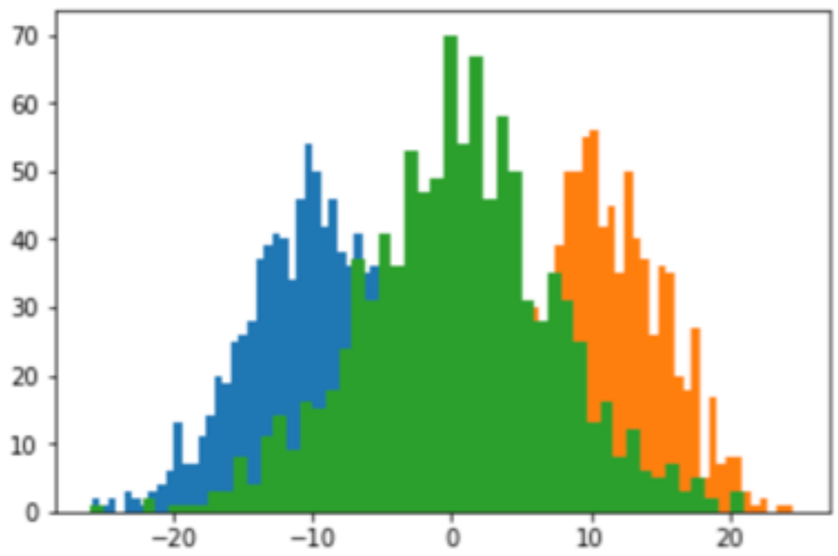
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 452 entries, 0 to 451
Columns: 280 entries, 0 to 279
```

```
dtypes: float64(125), int64(155)
memory usage: 988.8 KB
None


The max for the first column: 83
The min for the first column: 0
The average for the first column: 46.4712389380531


The max for the third column: 780
The min for the third column: 105
The average for the third column: 166.18805309734512
The median for the third column: 164.0
The age for the patient with the max value in the third column: 1.0
The age for the patient with the min value in the third column: 3.0


The max for the fourth column: 3
The min for the fourth column: 6
The average for the fourth column: 68.17035398230088
The median for the fourth column: 68.0
The age for the patient with the min value in the fourth column: 1.0
The age for the patient with the max value in the fourth column: 53.0
The gender for the patient with the max value in the fourth column: 0.0


The columns with the three highest correlated features are: 90, 92, 102
```

EE 379K Lab Report

Programming Questions

1a) When the two distributions are added together, the summed distribution is in the middle of the two distributions. As it can be seen here:




```
The mean of the sum is: 0.34913336396
The variance of the sum is: 48.577546012
```

The blue represents the first distribution with mean -10, the orange represents the second distribution with mean 10, and the green represents the summed distribution. The mean of the summed distribution is close to 0, which is expected behavior of normal distributions. Furthermore, the variance is close to 50, which is expected since the variance of the individual distributions is 25.

5a) There are 452 patients (the number of rows) and 279 features (the number of columns).

5b) The first feature is age this is because the minimum for that column is 0, the maximum value for that column is 83, and the average is approximately 46. Thus age seems the most appropriate since there are no values below 0 and there are no overly large numbers. Furthermore, during a hospital visit usually the weight of the patient is one of the first things measured.

```
The max for the first column: 83
The min for the first column: 0
The average for the first column: 46.4712389380531
```

The second column is gender evidenced by the binary values present in the column.

The third column is height in centimeters because the average height is 168 centimeters which is approximately 5.5 feet. There are a few outliers, the maximum value in this column is 780 and this value belongs to a patient that is one years old. This suggests that the measurement for this patient was not taken in centimeters, but possibly in millimeters (780 mm is approximately 2.5 feet).

```
The max for the third column: 780
The min for the third column: 105
The average for the third column: 166.18805309734512
The median for the third column: 164.0
The age for the patient with the max value in the third column: 1.0
The age for the patient with the min value in the third column: 3.0
```

The fourth column is weight because the average for that column is approximately 68 pounds, the max weight is 176 pounds and that belongs to a patient that is 53 years old. Furthermore the minimum value in that column is 6, and this belongs to a patient that is 1 year old.

```
The max for the fourth column: 176
The min for the fourth column: 6
The average for the fourth column: 68.17035398230088
The median for the fourth column: 68.0
The age for the patient with the min value in the fourth column: 1.0
The age for the patient with the max value in the fourth column: 53.0
The gender for the patient with the max value in the fourth column: 0.0
```

5c) Yes there are missing values. The code to replace the missing values is shown below.

```
#5c missing values
null_data = patientDF[patientDF.isnull().any(axis=1)] #rows with missing data
patientDF = patientDF.fillna(patientDF.mean()) #fill missing values with average
```

5d) The best way to figure out which features strongly influence the patient condition is to determine if a correlation exists between a feature and the patient condition. A strong correlation would suggest that there is some relationship between the feature and the result.

The top 3 most important features was calculated by using correlation function built into pandas. The result came out to be that the features 90, 92, and 102 had the highest correlation to the patient condition. A snapshot of the code is provided below.

```
#finding the values with the three highest correlations
firstCorr = secondCorr = thirdCorr = float('-Inf')
first = second = third = float('-Inf')

for i in range(0, 279):
    corr = patientDF[i].corr(patientDF[279])
    if(corr > firstCorr):
        firstCorr = corr
        first = i
    elif(corr > secondCorr):
        secondCorr = corr
        second = i
    elif(corr > thirdCorr):
        thirdCorr = corr
        third = i
print()
print()

#the columns with the three highest correlated features

print("The columns with the three highest correlated features are: " + str(first) + ', ' + str(second) + ', ' +
      str(third))
```

Written Questions

1a

	$X=0$	$X=1$
$Y=0$	$1/4$	$1/4$
$Y=1$	$1/6$	$1/3$

$P(X=1) = P(X=1 \text{ and } Y=0) + P(X=1 \text{ and } Y=1)$
$= 1/4 + 1/3 = \boxed{7/12}$

1b $P(X=1 \mid Y=1) = \dfrac{P(X=1 \text{ and } Y=1)}{P(Y=1)} = \dfrac{1/3}{1/2} = \dfrac{1}{3} \cdot 2 = \boxed{\dfrac{2}{3}}$

1c Marginal Distribution of X

$$P_X(X=x) = \begin{cases} 5/12 & x=0 \\ 7/12 & x=1 \end{cases}$$

$E[X]$ = Expected Value $= 0 \cdot \frac{5}{12} + 1 \cdot \frac{7}{12} = \frac{7}{12}$

$E[X^2] = 0 \cdot \frac{5}{12} + 1 \cdot \frac{7}{12} = \frac{7}{12}$

$\mathrm{Var}(x) = E[X^2] - (E[X])^2 = \frac{7}{12} - \left(\frac{7}{12}\right)^2 = \boxed{\frac{35}{144}}$

1d $\mathrm{Var}(X \mid Y=1) = E[(X \mid Y=1)^2] - (E[X \mid Y=1])^2$

Conditional Distribution $Y=1$

$$P_X(X=x \mid Y=1) = \begin{cases} 1/3 & x=0 \\ 2/3 & x=1 \end{cases}$$

$E[X] = 0 \cdot \frac{1}{3} + 1 \cdot \frac{2}{3} = \frac{2}{3}$ $\rightarrow \mathrm{Var}(X \mid Y=1) = \frac{2}{3} - \left(\frac{2}{3}\right)^2$

$E[X^2] = 0 \cdot 1/3 + 1 \cdot \frac{2}{3} = \frac{2}{3}$ $= \boxed{2/9}$

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

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

$E[g(x)] = \sum g(x) \cdot f(x)$ ← Lotus

	x=0	x=1	total
Y=0	1/4	1/4	.5
Y=1	1/6	1/3	.5
total	5/12	7/12	1

under the universe for $E[X \mid Y=1]$

$\frac{\frac{1}{6}}{.5} = \frac{1}{3} \rightarrow x = 0$

$\frac{\frac{1}{3}}{.5} = \frac{2}{3} \rightarrow x = 1$

$E[X^3 \mid Y=1] = \sum x^3 \cdot f(x) = 0 \cdot \frac{1}{3} + \frac{2}{3} \cdot 1 = \frac{2}{3}$

$E[X^2 \mid Y=1] = \sum x^2 \cdot f(x) = 0 \cdot \frac{1}{3} + \frac{2}{3} \cdot 1 = \frac{2}{3}$

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

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

Problem #2

$$h(v_1 \cdot v_1) + k(v_1 \cdot v_2) = P_1 \cdot v_1$$

$$h(v_2 \cdot v_1) + k(v_2 \cdot v_2) = P_1 \cdot v_2$$

Projecting P_1:

$$h(3) + k(1) = 9$$

$$h(1) + k(1) = 3$$

$$h = 3, \quad k = 0$$

$$3(1,1,1) - 0(1,0,0) = (3,3,3)$$

$$\boxed{\text{point} = (3,0)}$$

Projecting P_2:

$$h(3) + k(1) = 6$$

$$h(1) + k(1) = 1$$

$$h = 2.5$$

$$k = -1.5$$

$$2.5[1,1,1] - 1.5[1,0,0] = [1, 2.5, 2.5]$$

$$\boxed{\text{point} = (2.5, -1.5)}$$

Projecting P_3:

$$3h + k = 1$$

$$h + k = 0$$

$$h = 1/2 \quad k = -1/2$$

$$\boxed{\text{Point} = (1/2, -1/2)}$$

$$1/2[1,1,1] - \frac{1}{2}[1,0,0] = [0, 1/2, 1/2]$$

3. CLT : $P(X < 50) = \Phi\left(\frac{\bar{X} - n\mu}{\sqrt{n\sigma^2}}\right)$

$n = 100 \quad E[X] = \mu = 2/3 \quad \bar{X} = 50$

$var(x) = \sigma^2 = p(1-p) = 2/3(1/3) = 2/9$

$$P(X<50) = \Phi\left(\frac{50 - 100(2/3)}{\sqrt{100(2/9)}}\right)$$

$$= \Phi\left(\frac{-16.667}{4.53}\right) = -3.53$$ use z-table

$P(X<50) = 1 - \Phi(3.53) = 1 - 0.9998 = \boxed{0.0002}$