Arinjay Jain (A20447307)

CS 584: Machine Learning

Spring 2020 Assignment 1

Question 1 (40 points)

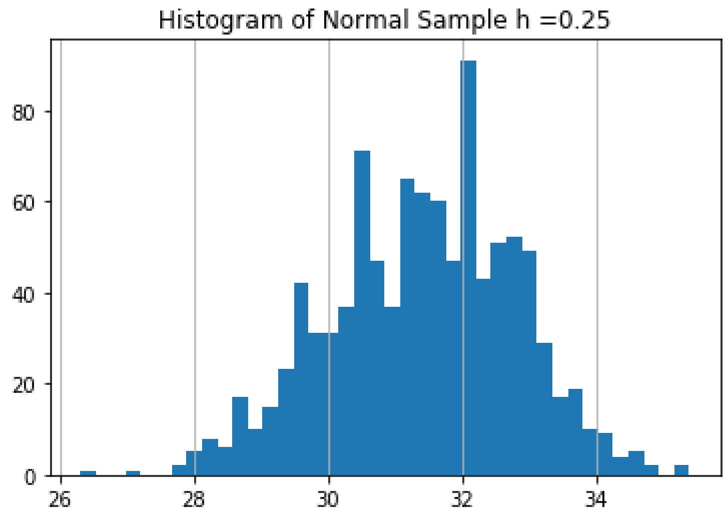
Write a Python program to calculate the density estimator of a histogram. Use the field *x* in the NormalSample.csv file.

a) (5 points) According to Izenman (1991) method, what is the recommended bin-width for the histogram of x?
Ans: The recommended bin-width (lzenman) : **0.3998667554864774**

b) (5 points) What are the minimum and the maximum values of the field x?
Ans: **Max value of X: 35.4 and Min value of X: 26.3**

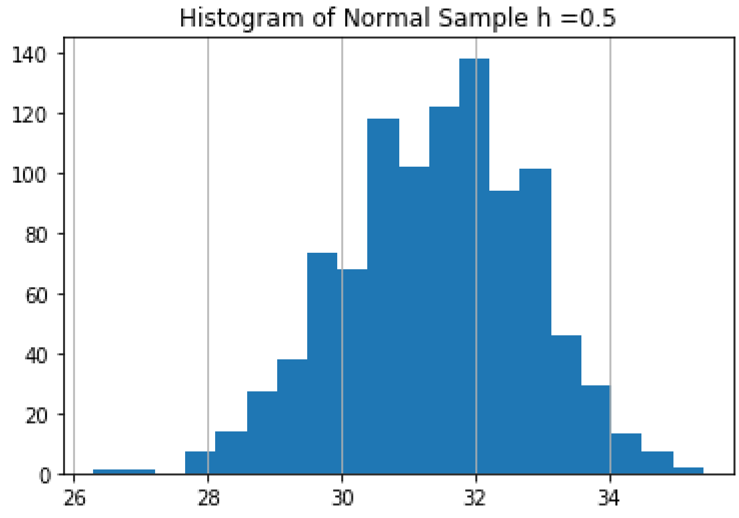
c) (5 points) Let a be the largest integer less than the minimum value of the field x, and b be the smallest integer greater than the maximum value of the field x. What are the values of a and b?
Ans: **Largest integer less than the minimum value of the field x: a= 26**
Smallest integer greater than the maximum value of the field x: b= 36

d) (5 points) Use h = 0.25, minimum = a and maximum = b. List the coordinates of the density estimator. Paste the histogram drawn using Python or your favorite graphing tools.
Ans: List the coordinates of the density estimator for **h = 0.25**
[(26.125, 0.0), (26.375, 0.003996003996003996), (26.625, 0.0), (26.875, 0.0), (27.125, 0.003996003996003996), (27.375, 0.0), (27.625, 0.007992007992007992), (27.875, 0.015984015984015984), (28.125, 0.023976023976023976), (28.375, 0.03596403596403597), (28.625, 0.03596403596403597), (28.875, 0.07192807192807193), (29.125, 0.059940059940059943), (29.375, 0.14785214785214784), (29.625, 0.11188811188811189), (29.875, 0.1878121878121878), (30.125, 0.14785214785214784), (30.375, 0.2677322677322677), (30.625, 0.1838161838161838), (30.875, 0.22777222777222778), (31.125, 0.17582417582417584), (31.375, 0.33166833166833165), (31.625, 0.23976023976023977), (31.875, 0.32367632367632365), (32.125, 0.22777222777222778), (32.375, 0.2837162837162837), (32.625, 0.21178821178821178), (32.875, 0.22777222777222778), (33.125, 0.10789210789210789), (33.375, 0.13186813186813187), (33.625, 0.05194805194805195), (33.875, 0.06393606393606394), (34.125, 0.03596403596403597), (34.375, 0.023976023976023976), (34.625, 0.011988011988011988), (34.875, 0.007992007992007992), (35.125, 0.0), (35.375, 0.007992007992007992), (35.625, 0.0), (35.875, 0.0)]



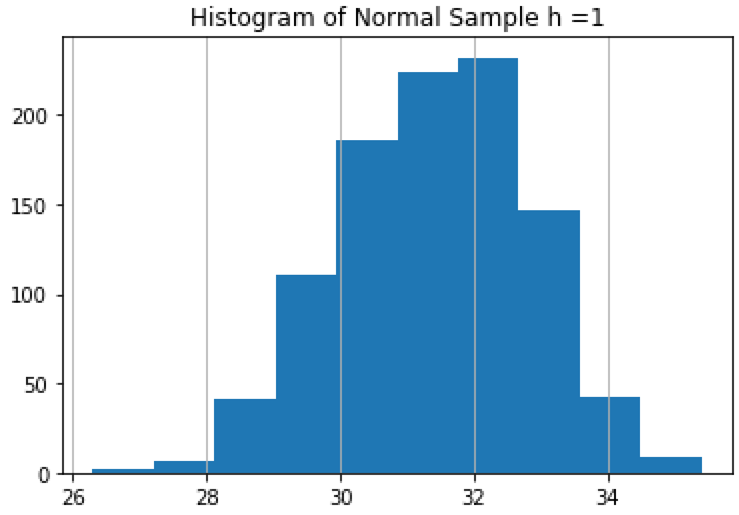
Bin Width for h = 0.25 Bins = 40

e) (5 points) Use h = 0.5, minimum = a and maximum = b. List the coordinates of the density estimator. Paste the histogram drawn using Python or your favorite graphing tools.
Ans: List the coordinates of the density estimator for h = **0.5**
[(26.25, 0.001998001998001998), (26.75, 0.0), (27.25, 0.001998001998001998), (27.75, 0.011988011988011988), (28.25, 0.029970029970029972), (28.75, 0.053946053946053944), (29.25, 0.1038961038961039), (29.75, 0.14985014985014986), (30.25, 0.2077922077922078), (30.75, 0.2057942057942058), (31.25, 0.25374625374625376), (31.75, 0.2817182817182817), (32.25, 0.25574425574425574), (32.75, 0.21978021978021978), (33.25, 0.11988011988011989), (33.75, 0.057942057942057944), (34.25, 0.029970029970029972), (34.75, 0.00999000999000999), (35.25, 0.003996003996003996), (35.75, 0.0)]



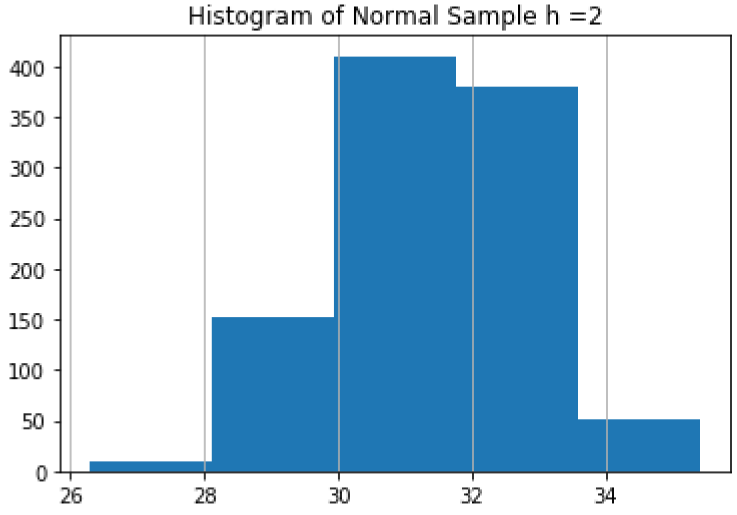
Bin Width for h = 0.5 Bins = 20

f) (5 points) Use h = 1, minimum = a and maximum = b. List the coordinates of the density estimator. Paste the histogram drawn using Python or your favorite graphing tools.
Ans: List the coordinates of the density estimator for h = 1
[(26.5, 0.000999000999000999), (27.5, 0.006993006993006993), (28.5, 0.04195804195804196), (29.5, 0.12687312687312688), (30.5, 0.20679320679320679), (31.5, 0.2677322677322677), (32.5, 0.23776223776223776), (33.5, 0.08891108891108891), (34.5, 0.01998001998001998), (35.5, 0.001998001998001998)]



Bin Width for h = 1 Bins = 10

g) (5 points) Use h = 2, minimum = a and maximum = b. List the coordinates of the density estimator. Paste the histogram drawn using Python or your favorite graphing tools.
Ans: List the coordinates of the density estimator for h = 2
[(27.0, 0.003996003996003996), (29.0, 0.08441558441558442), (31.0, 0.23726273726273725), (33.0, 0.16333666333666333), (35.0, 0.01098901098901099)]



Bin Width for h = 2 Bins = 5

h) (5 points) Among the four histograms, which one, in your honest opinions, can best provide your insights into the shape and the spread of the distribution of the field x? Please state your arguments.
Ans : **Among the four histograms, I can say h=0.5 is more close to Izenman bin-width (0.399) and it more seem likes Normal Distribution, Symmetric about x = 32**

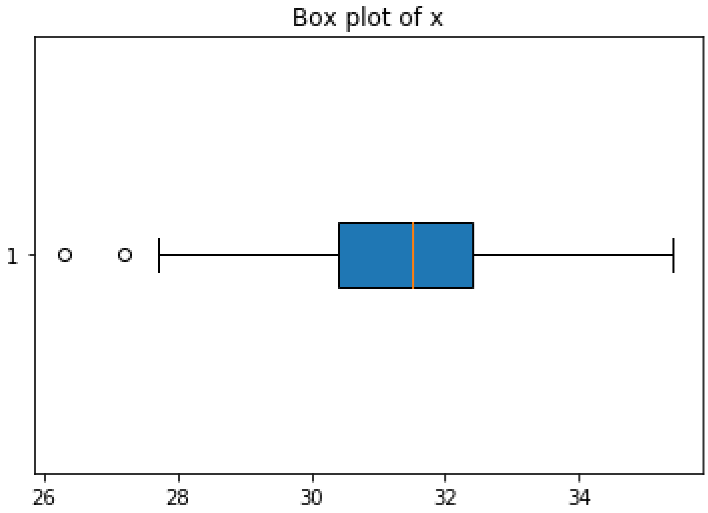
Question 2 (20 points)

Use in the NormalSample.csv to generate box-plots for answering the following questions.

a) (5 points) What is the five-number summary of x? What are the values of the 1.5 IQR whiskers?
Ans: **Minimum of X = 26.3**
First Quartile of X = 30.4
Median of X = 31.5
Third Quartile = 32.4
Maximum of X = 35.4
IQR of X: 2.0
1.5 * IQR whiskers is : 3.0
The lower whisker extends to the larger of Q1 – 1.5 * IQR = 27.4
The upper whisker extends to the smaller of Q3 + 1.5 * IQR = 35.4

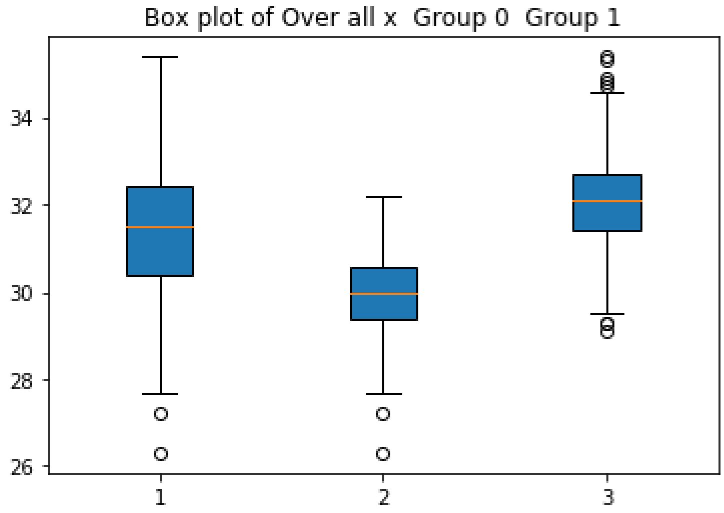
b) (5 points) What is the five-number summary of x for each category of the group? What are the values of the 1.5 IQR whiskers for each category of the group?
Ans: **Five Number Summary for X where group value is:'0'**
Minimum of X = 26.3
First Quartile of X = 29.4
Median of X = 30.0
Third Quartile = 30.6
Maximum of X = 32.2
IQR of X: 1.2000000000000028
1.5 * IQR whiskers is : 1.8000000000000043
The lower whisker extends to the larger of Q1 – 1.5 * IQR = 27.599999999999994
The upper whisker extends to the smaller of Q3 + 1.5 * IQR = 32.400000000000006
Five Number Summary for X where group value is:'1'
Minimum of X = 29.1
First Quartile of X = 31.4
Median of X = 32.1
Third Quartile = 32.7
Maximum of X = 35.4
IQR of X: 1.3000000000000043
1.5 * IQR whiskers is : 1.9500000000000064
The lower whisker extends to the larger of Q1 – 1.5 * IQR = 29.449999999999992
The upper whisker extends to the smaller of Q3 + 1.5 * IQR = 34.650000000000006

c) (5 points) Draw a boxplot of x (without the group) using the Python boxplot function. Can you tell if the Python's boxplot has displayed the 1.5 IQR whiskers correctly?



From this box plot we can say it showing the same values as we calculated in 2.a), boxplot has correctly displayed the 1.5 IQR whiskers.

d) (5 points) Draw a graph where it contains the boxplot of x, the boxplot of x for each category of Group (i.e., three boxplots within the same graph frame). Use the 1.5 IQR whiskers, identify the outliers of x, if any, for the entire data and for each category of the group.
Hint: Consider using the CONCAT function in the PANDA module to append observations.



Outliers of x: [27.2, 26.3]
Outliers of where Group 0: [27.2, 26.3]
Outliers of where Group 1: [35.3, 29.3, 35.4, 34.9, 34.7, 34.8, 29.3, 29.1]

Question 3 (40 points)

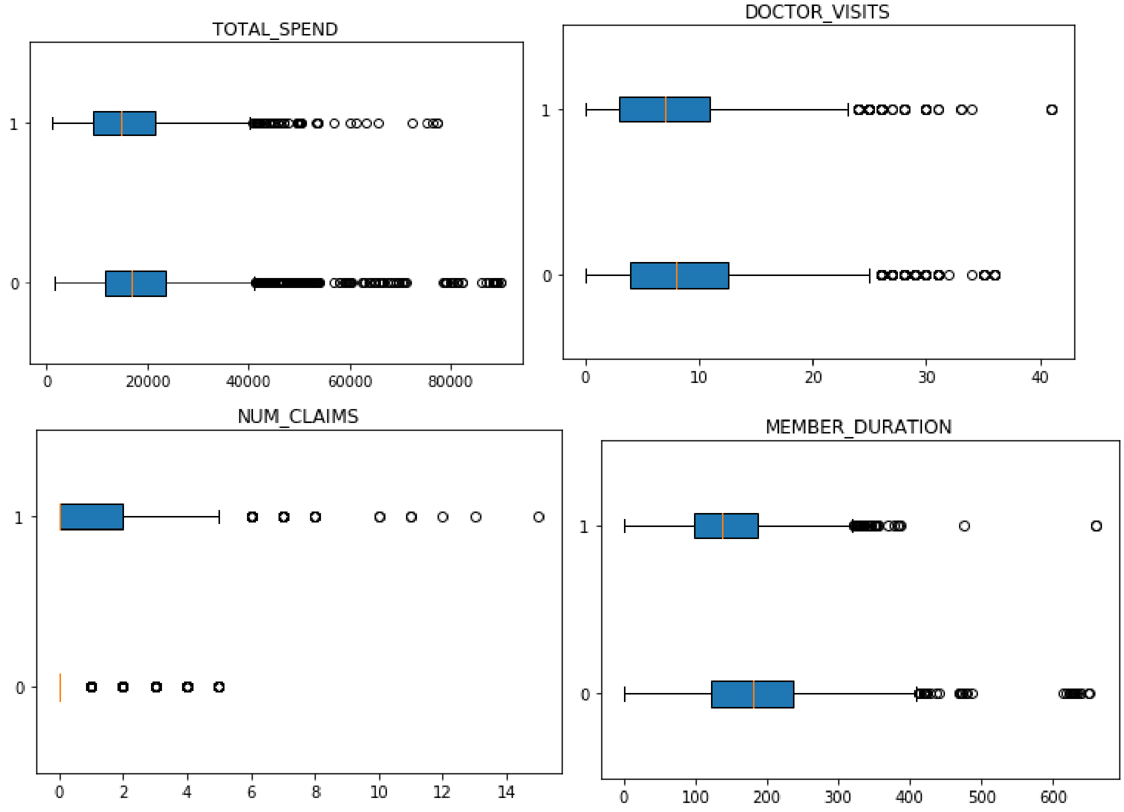
The data, FRAUD.csv, contains results of fraud investigations of 5,960 cases. The binary variable FRAUD indicates the result of a fraud investigation: 1 = Fraudulent, 0 = Otherwise. The other interval variables contain information about the cases.

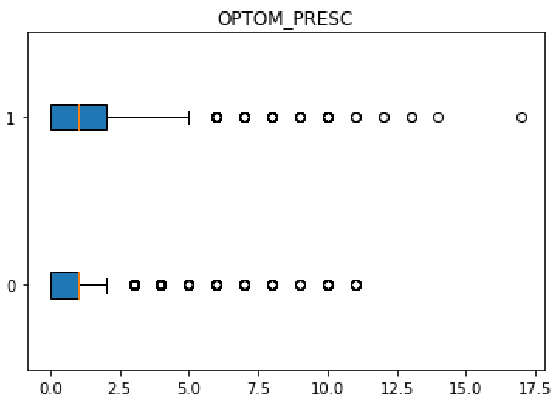
1. TOTAL_SPEND: Total amount of claims in dollars
2. DOCTOR_VISITS: Number of visits to a doctor
3. NUM_CLAIMS: Number of claims made recently
4. MEMBER_DURATION: Membership duration in number of months
5. OPTOM_PRESC: Number of optical examinations
6. NUM_MEMBERS: Number of members covered

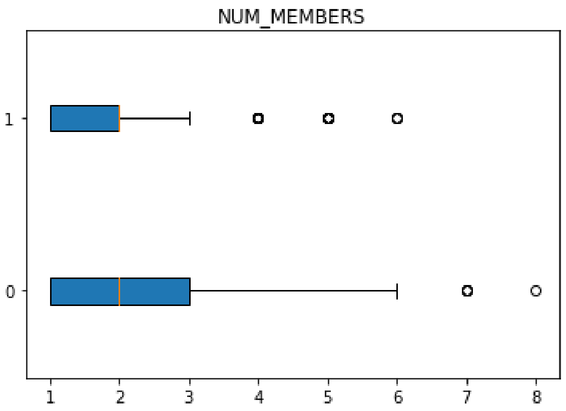
You are asked to use the Nearest Neighbors algorithm to predict the likelihood of fraud.

a) (5 points) What percent of investigations are found to be fraudulent? Please give your answer up to 4 decimal places.
Ans: **Percent of investigations are found to be fraudulent**: **19.9497 %**

b) (5 points) Use the BOXPLOT function to produce horizontal box-plots. For each interval variable, one box-plot for the fraudulent observations, and another box-plot for the non-fraudulent observations. These two box-plots must appear in the same graph for each interval variable.







c) (10 points) Orthonormalize interval variables and use the resulting variables for the nearest neighbor analysis. Use only the dimensions whose corresponding eigenvalues are greater than one.

i. (5 points) How many dimensions are used?

Ans: **6**, *Eigenvalues: [6.84728061e+03 8.38798104e+03 1.80639631e+04 3.15839942e+05 8.44539131e+07 2.81233324e+12] all are greater than one so we have to take all 6 dimension.*

ii. (5 points) Please provide the transformation matrix? You must provide proof that the resulting variables are actually orthonormal.

Ans: The transformation matrix is a 6 x 6 matrix:

Transformation Matrix =

```
[[-6.49862374e-08 -2.41194689e-07  2.69941036e-07 -2.42525871e-07
  -7.90492750e-07  5.96286732e-07]
 [ 7.31656633e-05 -2.94741983e-04  9.48855536e-05  1.77761538e-03
   3.51604254e-06  2.20559915e-10]
 [-1.18697179e-02  1.70828329e-03 -7.68683456e-04  2.03673350e-05
   1.76401304e-07  9.09938972e-12]
 [ 1.92524315e-06 -5.37085514e-05  2.32038406e-05 -5.78327741e-05
   1.08753133e-04  4.32672436e-09]
 [ 8.34989734e-04 -2.29964514e-03 -7.25509934e-03  1.11508242e-05
   2.39238772e-07  2.85768709e-11]
 [ 2.10964750e-03  1.05319439e-02 -1.45669326e-03  4.85837631e-05
   6.76601477e-07  4.66565230e-11]]
```

The Transformed x =

```
[[ 5.96859502e-03  1.02081629e-02 -6.64664861e-03  1.39590283e-02
   9.39352141e-03  6.56324665e-04]
 [-2.09672310e-02  5.01932025e-03  8.51930607e-04  5.16174400e-03
   1.22658834e-02  7.75702220e-04]
 [ 7.64597676e-03  1.97528525e-02 -7.38335310e-03 -1.71350853e-03
   1.50348109e-02  8.95075830e-04]
 ...
 [-7.18408819e-05 -1.62580211e-02  2.75078514e-02 -7.13245766e-03
```

```
  -4.74021952e-02  5.31896971e-02]
 [-1.80147801e-04 -1.62154130e-02  2.76213381e-02 -9.17125411e-03
  -4.76625006e-02  5.35474776e-02]
 [-2.21157680e-03 -2.73884697e-02  2.93391341e-02 -7.81347172e-03
  -4.70861917e-02  5.36071324e-02]]
Identity Matrix =
 [[ 1.00000000e+00 -3.00432422e-16 -4.61219604e-16  5.45323877e-15
   1.20996962e-15 -1.28911638e-16]
 [-3.00432422e-16  1.00000000e+00 -6.44449771e-16 -2.76820667e-14
  -1.23512311e-15  7.78890841e-16]
 [-4.61219604e-16 -6.44449771e-16  1.00000000e+00  3.50891191e-15
   1.00613962e-16 -2.25514052e-16]
 [ 5.45323877e-15 -2.76820667e-14  3.50891191e-15  1.00000000e+00
   1.14860378e-14 -3.47812057e-15]
 [ 1.20996962e-15 -1.23512311e-15  1.00613962e-16  1.14860378e-14
   1.00000000e+00 -6.31439345e-16]
 [-1.28911638e-16  7.78890841e-16 -2.25514052e-16 -3.47812057e-15
  -6.31439345e-16  1.00000000e+00]]
The orthonormalize x =
 [[-6.56324665e-04  9.39352141e-03  1.39590283e-02 -6.64664861e-03
   1.02081629e-02 -5.96859502e-03]
 [-7.75702220e-04  1.22658834e-02  5.16174400e-03  8.51930607e-04
   5.01932025e-03  2.09672310e-02]
 [-8.95075830e-04  1.50348109e-02 -1.71350853e-03 -7.38335310e-03
   1.97528525e-02 -7.64597676e-03]
 ...
 [-5.31896971e-02 -4.74021952e-02 -7.13245766e-03  2.75078514e-02
  -1.62580211e-02  7.18408819e-05]
 [-5.35474776e-02 -4.76625006e-02 -9.17125411e-03  2.76213381e-02
  -1.62154130e-02  1.80147801e-04]
 [-5.36071324e-02 -4.70861917e-02 -7.81347172e-03  2.93391341e-02
  -2.73884697e-02  2.21157680e-03]]
Identity Matrix =
 [[ 1.00000000e+00 -1.11022302e-16  9.67108338e-17 -7.63278329e-17
   1.99493200e-17 -7.91467586e-18]
 [-1.11022302e-16  1.00000000e+00  1.83447008e-16  2.25514052e-17
  -1.38777878e-17 -3.03576608e-18]
 [ 9.67108338e-17  1.83447008e-16  1.00000000e+00 -6.67868538e-17
  -7.91467586e-18  2.55465137e-17]
 [-7.63278329e-17  2.25514052e-17 -6.67868538e-17  1.00000000e+00
  -9.10729825e-17  1.63660318e-16]
 [ 1.99493200e-17 -1.38777878e-17 -7.91467586e-18 -9.10729825e-17
   1.00000000e+00  3.25748543e-16]
 [-7.91467586e-18 -3.03576608e-18  2.55465137e-17  1.63660318e-16
```

```
3.25748543e-16  1.00000000e+00]]
```

After checking the multiplication of orthonormalize's transpose to orthonormalize matrix. we get an Identity Matrix that justify that the columns of the matrix of transformed input fields are orthonormal.

d) (10 points) Use the NearestNeighbors module to execute the Nearest Neighbors algorithm using exactly five neighbors and the resulting variables you have chosen in c). The KNeighborsClassifier module has a score function.

i. (5 points) Run the score function, provide the function return value
Ans: **Score is 0.8414429530201343**

ii. (5 points) Explain the meaning of the score function return value.
Ans: This score value(0.8414429530201343) represent the percentage of accuracy (**84.144 %)** and Misclassified percentage is (1- 0.8414429530201343) **15.85%**

e) (5 points) For the observation which has these input variable values: TOTAL_SPEND = 7500, DOCTOR_VISITS = 15, NUM_CLAIMS = 3, MEMBER_DURATION = 127, OPTOM_PRESC = 2, and NUM_MEMBERS = 2, find its **five** neighbors. Please list their input variable values and the target values. *Reminder: transform the input observation using the results in c) before finding the neighbors*.
Ans: The Input values = [7500, 15, 3, 127, 2, 2]
Five Neighbors: [588 2897 1199 1246 886] these are index value of our data sets.

Index	TOTAL_SPEND	DOCTOR_VISITS	NUM_CLAIMS	MEMBER_DURATION	OPTOM_PRESC	NUM_MEMBERS
588	7500	6	4	345	1	1
2897	16000	3	0	190	0	3
1199	10000	15	2	109	3	1
1246	10200	1	0	105	1	1
886	8900	18	0	280	0	1

f) (5 points) Follow-up with e), what is the predicted probability of fraudulent (i.e., FRAUD = 1)? If your predicted probability is greater than or equal to your answer in a), then the observation will be classified as fraudulent. Otherwise, non-fraudulent. Based on this criterion, will this observation be misclassified?
Ans: **Predicted probability of fraudulent = 1**
Fraud values of all five neighbors are 1. And the predicted probability is also one so we can say there is no misclassified factor in our observation.