CS 484: Introduction to Machine Learning

Spring 2021 Assignment 1 Answer Key

# Question 1 (25 points)

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

1. (5 points) Use the Pandas describe() function to find out the count, the mean, the standard deviation, the minimum, the 25th percentile, the median, the 75th percentile, and the maximum.

The describe() function returns the following statistics, shown up to 7 decimal places.

|  |  |
| --- | --- |
| Statistic | Value |
| count | 1001 |
| mean | 31.4145854 |
| std | 1.3976720 |
| min | 26.3 |
| 25% | 30.4 |
| 50% | 31.5 |
| 75% | 32.4 |
| max | 35.4 |

1. (5 points) What is the bin width recommended by the Izenman (1991) method? Please round your answer to the nearest tenths (i.e., one decimal place).

Since IQR = Q3 – Q1 = 32.4 – 30.4 = 2.0 and N = 1001, then 2\*IQR/N1/3 = 0.3998668. After rounded to the nearest tenths, the recommended bin width is 0.4.

1. (10 points) Use the Shimazaki and Shinomoto (2007) method and try = 0.1, 0.2, 0.5, 1.0, 2.0, and 5.0. What is the recommended bin width? You need to show your calculations to receive full credit.

The following table shows the results of trying these six values of delta.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Delta | C(Delta) | Low Boundary | Middle Value | High Boundary | Number of Bin |
| 0.1 | -6773.4163 | 26.2 | 31.4 | 35.4 | 92 |
| 0.2 | -7964.7592 | 26.2 | 31.4 | 35.4 | 46 |
| 0.5 | -8930.6272 | 26 | 31.5 | 35.5 | 19 |
| 1 | -9538.2900 | 26 | 31 | 36 | 10 |
| 2 | -7963.0400 | 26 | 32 | 36 | 5 |
| 5 | -4946.8622 | 25 | 30 | 40 | 3 |

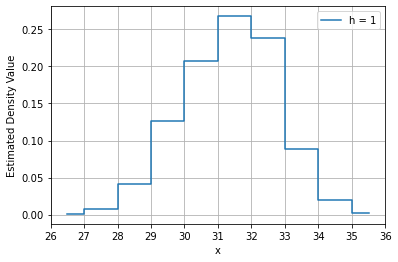
The C(Delta) function attains its minimum when delta = 1. Therefore, the recommended bin width is 1.

1. (5 points) Based on your recommended bin width answer in (c), list the mid-points and the estimated density function values. Draw the density estimator as a vertical bar chart using the matplotlib. You need to properly label the graph to receive full credit.

The mid-points and the estimated density function values are:

|  |  |
| --- | --- |
| Mid-point | Density |
| 26.5 | 0.000999 |
| 27.5 | 0.006993 |
| 28.5 | 0.041958 |
| 29.5 | 0.126873 |
| 30.5 | 0.206793 |
| 31.5 | 0.267732 |
| 32.5 | 0.237762 |
| 33.5 | 0.088911 |
| 34.5 | 0.019980 |
| 35.5 | 0.001998 |

Displaying it as a Step Graph, the histogram is:



# Question 2 (15 points)

The NormalSample.csv contains the *group* variable that has two values, namely, 0 and 1.

1. (5 points) What is the five-number summary of for each category of the *group*? What are the values of the 1.5 IQR whiskers for each category of the group?

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| group | Minimum | Q1 | Median | Q3 | Maximum |
| 0 | 26.3 | 29.4 | 30.0 | 30.6 | 32.2 |
| 1 | 29.1 | 31.4 | 32.1 | 32.7 | 35.4 |

For group = 0, the IQR = Q3 – Q1 = 30.6 – 29.4 = 1.2.

The lower whisker is Q1 – 1.5\*IQR = 29.4 – 1.5 \* 1.2 = 27.6.

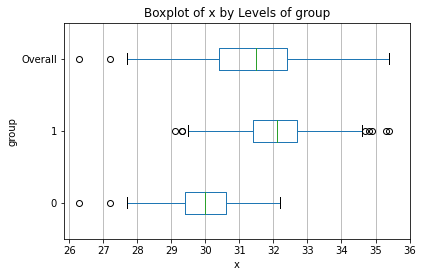
The upper whisker is Q3 + 1.5\*IQR = 30.6 + 1.5 \* 1.2 = 32.4.

For group = 1, the IQR = Q3 – Q1 = 32.7 – 31.4 = 1.3.

The lower whisker is Q1 – 1.5\*IQR = 31.4 – 1.5 \* 1.3 = 29.45.

The upper whisker is Q3 + 1.5\*IQR = 32.7 + 1.5 \* 1.3 = 34.65.

1. (10 points) Draw a graph where it contains the overall boxplot of , the boxplot of for each category of *group* (i.e., three horizontal boxplots within the same graph frame). Use the 1.5 IQR whiskers, identify any outliers of for the entire data and each category of the group. You must properly label your boxplots to receive full credits. *Hint: Consider using the CONCAT function in the PANDA module to append observations*.



There are two outliers in the entire data (i.e., group = Overall). They are:

|  |  |
| --- | --- |
| Index | x |
| 70 | 27.2 |
| 295 | 26.3 |

There are two outliers in the group = 0. They are the same as those of the entire data.

|  |  |
| --- | --- |
| Index | x |
| 70 | 27.2 |
| 295 | 26.3 |

There are eight outliers in the group = 1. They are:

|  |  |
| --- | --- |
| Index | x |
| 30 | 35.3 |
| 107 | 29.3 |
| 297 | 35.4 |
| 812 | 34.9 |
| 846 | 34.7 |
| 907 | 34.8 |
| 938 | 29.3 |
| 975 | 29.1 |

# Question 3 (35 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 = Fraud, 0 = Not Fraud. 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.

1. (5 points) What percent of investigations are found to be frauds? Please give your answer up to 4 decimal places.

We observed FRAUD = 0 in 4,771 observations and FRAUD = 1 in 1,189 observations. Thus, the percent of fraud is 1189 / (4771 + 1189) = 0.199497 = 19.9497%.

1. (10 points) Orthonormalize interval variables and use the orthonormalized columns for the nearest neighbor analysis. Use only the dimensions whose corresponding eigenvalues are greater than one.
   1. (5 points) How many dimensions are used?

The six eigenvalues are, in ascending order, 6.8472806e+03, 8.3879810e+03, 1.8063963e+04, 3.1583994e+05, 8.4453913e+07, and 2.8123332e+12. Since all six eigenvalues are greater than one, we will use all six dimensions.

* 1. (5 points) Please provide the transformation matrix? Show evidence that the orthonormalized columns are actually orthonormal.

The 6 x 6 transformation matrix is

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| -6.4986237e-08 | -2.4119469e-07 | 2.6994104e-07 | -2.4252587e-07 | -7.9049275e-07 | 5.9628673e-07 |
| 7.3165663e-05 | -2.9474198e-04 | 9.4885554e-05 | 1.7776154e-03 | 3.5160425e-06 | 2.2055991e-10 |
| -1.1869718e-02 | 1.7082833e-03 | -7.6868346e-04 | 2.0367335e-05 | 1.7640130e-07 | 9.0993897e-12 |
| 1.9252431e-06 | -5.3708551e-05 | 2.3203841e-05 | -5.7832774e-05 | 1.0875313e-04 | 4.3267244e-09 |
| 8.3498973e-04 | -2.2996451e-03 | -7.2550993e-03 | 1.1150824e-05 | 2.3923877e-07 | 2.8576871e-11 |
| 2.1096475e-03 | 1.0531944e-02 | -1.4566933e-03 | 4.8583763e-05 | 6.7660148e-07 | 4.6656523e-11 |

When we multiply the 5960 x 6 matrix of input fields with this 6 x 6 matrix, we got the 5960 x 6 matrix of transformed input fields. The product of the transpose of this matrix of transformed input fields and the matrix itself yields a 6 x 6 matrix which is shown below. Since this 6 x 6 matrix is practically an identity matrix, we can conclude that the columns of the matrix of transformed input fields are orthonormal.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **1.0000000e+00** | -6.2721096e-17 | -1.6975353e-15 | 8.7806823e-15 | 1.0025617e-15 | -2.1879200e-16 |
| -6.2721096e-17 | **1.0000000e+00** | -5.5858096e-16 | -1.9188535e-14 | -1.5126789e-15 | -2.9316827e-16 |
| -1.6975353e-15 | -5.5858096e-16 | **1.0000000e+00** | 1.7381929e-15 | 8.1532003e-17 | -2.9490299e-17 |
| 8.7806823e-15 | -1.9188535e-14 | 1.7381929e-15 | **1.0000000e+00** | 1.1328612e-14 | -3.8276673e-15 |
| 1.0025617e-15 | -1.5126789e-15 | 8.1532003e-17 | 1.1328612e-14 | **1.0000000e+00** | -6.2796990e-16 |
| -2.1879200e-16 | -2.9316827e-16 | -2.9490299e-17 | -3.8276673e-15 | -6.2796990e-16 | **1.0000000e+00** |

1. (10 points) Use the NearestNeighbors module to execute the Nearest Neighbors algorithm using exactly five neighbors and the orthonormalized columns you have chosen in (b). The KNeighborsClassifier module has a score function.
   1. (5 points) Find out from the documentation the purpose of the score function.

According to <https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html#sklearn.neighbors.KNeighborsClassifier.score> , this is the fraction of observations that are correctly classified. The commonly used Misclassification rate is one minus this number.

* 1. (5 points) Run the score function, show and explain the function return value.

The score function returns a scalar number of 0.8778523489932886. This indicates that our Nearest Neighbors Classifier has 87.79% accuracy. The misclassification rate is 100% - 87.79% = 12.21%.

1. (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 (b) before finding the neighbors*.

Since the data is sorted in ascending order to TOTAL\_SPEND, we can find this focal observation from the input data. This observation has an Index = 588. The five neighbors have indices 588, 2897, 1199, 1246, and 886. Therefore, the five neighbors are

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Index | TOTAL\_SPEND | DOCTOR\_VISITS | NUM\_CLAIMS | MEMBER\_DURATION | OPTOM\_PRESC | NUM\_MEMBERS |
| 588 | 7500 | 15 | 3 | 127 | 2 | 2 |
| 2897 | 16000 | 18 | 3 | 146 | 3 | 2 |
| 1199 | 10000 | 16 | 3 | 124 | 2 | 1 |
| 1246 | 10200 | 13 | 3 | 119 | 2 | 3 |
| 886 | 8900 | 22 | 3 | 166 | 1 | 2 |

1. (5 points) Follow-up with (d), what is the predicted probability of fraud (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 a fraud. Otherwise, not a fraud. Based on this criterion, will the observation in (d) be misclassified?

Since the FRAUD values of all five neighbors are 1, the predicted probability of fraud of the focal observation is also 1 and, therefore, the predicted FRAUD is 1. Since the focal observation has FRAUD = 1, this observation is NOT misclassified.

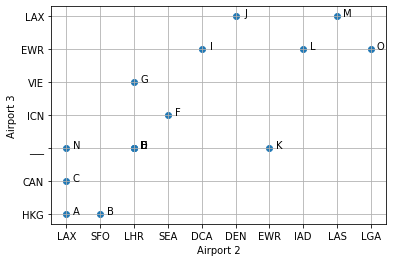
# Question 4 (25 points)

I found the following flights from Chicago O’Hare Airport (ORD) to Shanghai Pudong Airport (PVG).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Flight** | **Carrier 1** | **Carrier 2** | **Airport 1** | **Airport 2** | **Airport 3** | **Airport 4** |
| A | American | Cathay Pacific | ORD | LAX | HKG | PVG |
| B | American | Cathay Pacific | ORD | SFO | HKG | PVG |
| C | American | China Southern | ORD | LAX | CAN | PVG |
| D | American | Virgin Atlantic | ORD | LHR |  | PVG |
| E | British Airways | Virgin Atlantic | ORD | LHR |  | PVG |
| F | Delta |  | ORD | SEA | ICN | PVG |
| G | United | Austrian | ORD | LHR | VIE | PVG |
| H | United | Virgin Atlantic | ORD | LHR |  | PVG |
| I | United |  | ORD | DCA | EWR | PVG |
| J | United |  | ORD | DEN | LAX | PVG |
| K | United |  | ORD | EWR |  | PVG |
| L | United |  | ORD | IAD | EWR | PVG |
| M | United |  | ORD | LAS | LAX | PVG |
| N | United |  | ORD | LAX |  | PVG |
| O | United |  | ORD | LGA | EWR | PVG |

To answer the following questions, please replace empty string values in **Airport 3** with three underscore characters (i.e., ‘\_\_\_’).

1. (5 points) Generate a scatterplot of **Airport 3** (y-axis) versus **Airport 2** (x-axis). Please properly label the axes to receive full credits.



1. (5 points) Generate a frequency table of the airport codes in **Airport 2** and **Airport 3** combined.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Airport Code | CAN | DCA | DEN | EWR | HKG | IAD | ICN | LAS | LAX | LGA | LHR | SEA | SFO | VIE | \_\_\_ |
| Frequency | 1 | 1 | 1 | 4 | 2 | 1 | 1 | 1 | 5 | 1 | 4 | 1 | 1 | 1 | 5 |

1. (10 points) Suppose a new airline creates a new flight from ORD to PVG that first stops at LAX and then ICN. I want to know which flight(s) most resembles this new flight. Use the Cosine Distance to measure the differences between this flight and the existing flights.
   1. Create a vector of word counts for each flight. This vector has as many elements as the number of unique values found in (b).
   2. Initialize all elements in the vector to zeros.
   3. Count the number of times the airport codes appeared in **Airport 2** and **Airport 3**.
   4. Calculate the Cosine Distance between the new flight and the Flights A to O.

You will list the Cosine Distances in a table.

The following table shows the Cosine distances from the new flight to the existing flights.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Existing Flight | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O |
| Cosine Distance from New Flight | 0.5 | 1 | 0.5 | 1 | 1 | 0.5 | 1 | 1 | 1 | 0.5 | 1 | 1 | 0.5 | 0.5 | 1 |

1. (5 points) Which flight(s) have the shortest Cosine Distance from the new flight?

These existing six flights: A, C, F, J, M, and N have the shortest Cosine distances from the new flight. The common shortest Cosine distance is 0.5.