Lab Assignment 1

CIS492

Michael Hylton

2788154

[Lab 1 3](#_Toc190721092)

[Part 1 3](#_Toc190721093)

[Part 2 4](#_Toc190721094)

[Handling Null values to Replace: 4](#_Toc190721095)

[Sampling Random Data: 4](#_Toc190721096)

[Full Data Set Min, Max, Mean/Variance/Standard Deviation 5](#_Toc190721097)

[Number of distinct values 5](#_Toc190721098)

[Box plot, Histogram, or Z-score Visualization 6](#_Toc190721099)

[Calculating Outliers 6](#_Toc190721100)

[Discretionization 7](#_Toc190721101)

[Hot2Encoder 7](#_Toc190721102)

[Column Standardization 8](#_Toc190721103)

[Min-Max Normalization 8](#_Toc190721104)

[Part 3 8](#_Toc190721105)

[Part 4 9](#_Toc190721106)

# Lab 1

While the answers from Part 1 are below, the answer to the last part of part 2 is the second number under each variable.

## Part 1

Customer Key:

1. Nominal, Discrete
2. Binarization

Geographic Key:

1. Nominal, Discrete
2. Binarization

Customer Alternate Key:

1. Nominal, Discrete
2. Binarization

Marital Status:

1. Nominal, Discrete
2. Binarization

Gender:

1. Nominal, Discrete
2. Binarization

Yearly Income:

1. Ratio, Discrete
2. Discretization -> Normalization

Total Children:

1. Ordinal, Discrete
2. Binarization

Number of Children @ home:

1. Ordinal, Discrete
2. Binarization

English Education:

1. Nominal, Discrete
2. Binarization

Spanish Education:

1. Nominal, Discrete

French Education:

1. Nominal, Discrete

English Occupation:

1. Nominal, Discrete

Spanish Occupation:

1. Nominal, Discrete

French Occupation:

1. Nominal, Discrete

House Owner Flag:

1. Nominal, Discrete

Number of Cars Owned:

1. Ratio, Discrete
2. Binarization->Normalization

Date first purchased:

1. Interval, Discrete
2. Binarization-> Normalization

Commute Distance:

1. Ratio, Continuous
2. Discretization -> Normalization

Age:

1. Ratio, Discrete
2. Discretization -> Normalization

Bike Buyer:

1. Nominal, Discrete

## Part 2

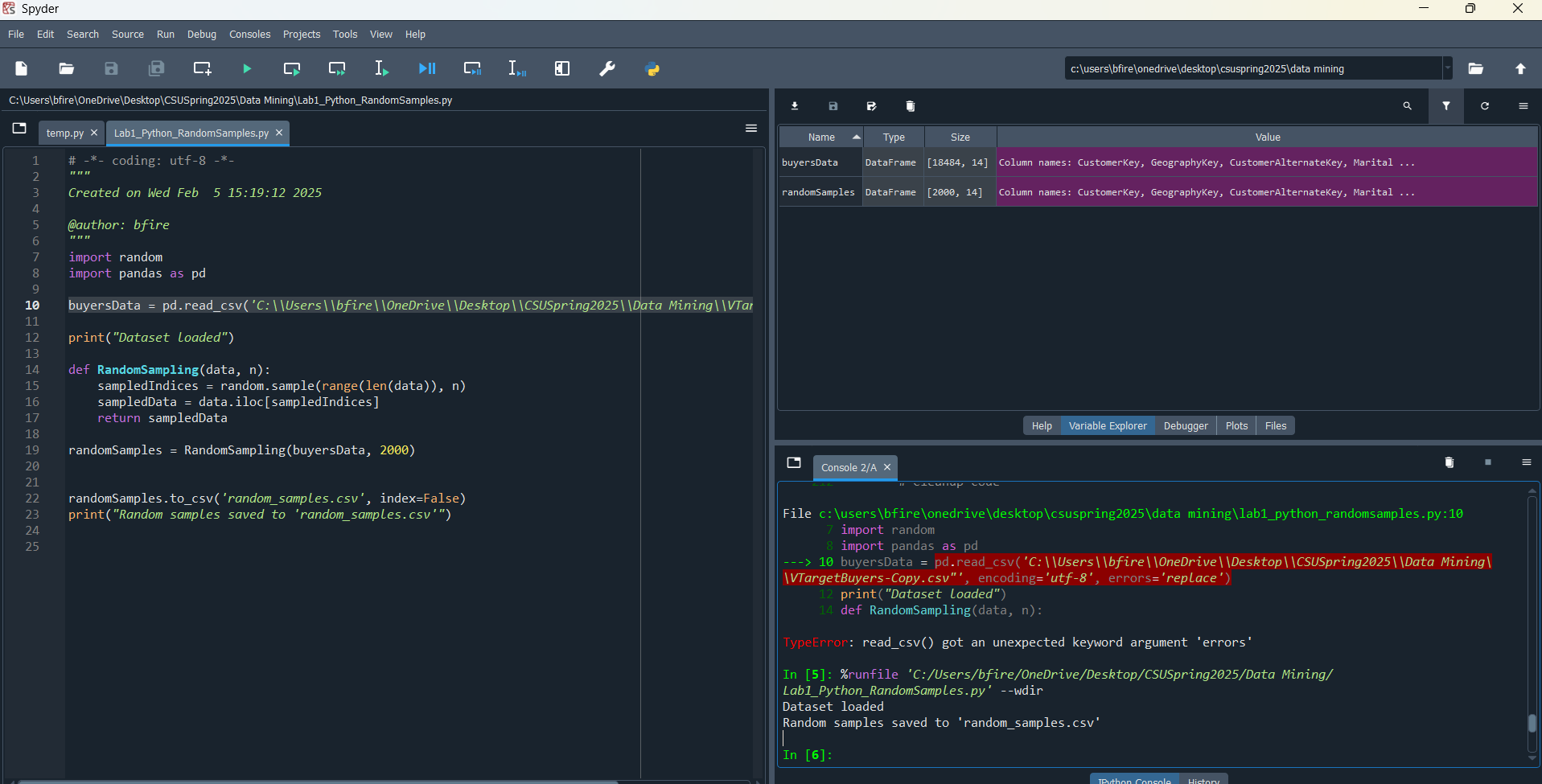
Initially, I started making the necessary preprocessing programs using Python (as it is what I am most familiar with) in Spyder. However, after reaching the part of the lab that requires data visualization, I instead swapped to Jupyter notebook. Using these two tools, I primarily used pandas for data processing, matplot for data visualization, scipi for ease of calculating z-scores, and sklearn for the 1HotEncoding and the standardization.

### Handling Null values to Replace:

Given we don’t know the exact details of the purpose of said data, we should replace with the weighted mean/average, normalizing according to the diet/class.

### Sampling Random Data:

Data selected saved in ‘Random Samples.csv’. Code is ‘Lab1\_Python\_RandomSamples.py’



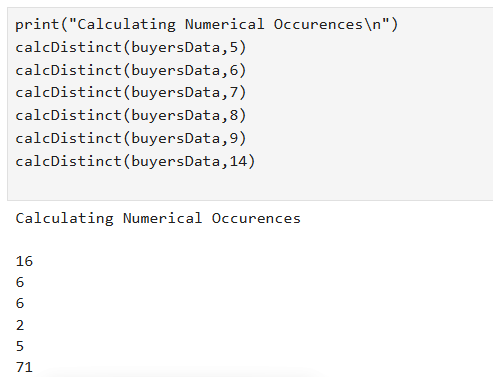
### Full Data Set Min, Max, Mean/Variance/Standard Deviation

A screenshot of a computer

Description automatically generated

Output of the statistical data for various Data collumns

### Number of distinct values



Number of distinct values in each column

### Box plot, Histogram, or Z-score Visualization

A diagram with a blue rectangle and a black line

Description automatically generated A graph of a column

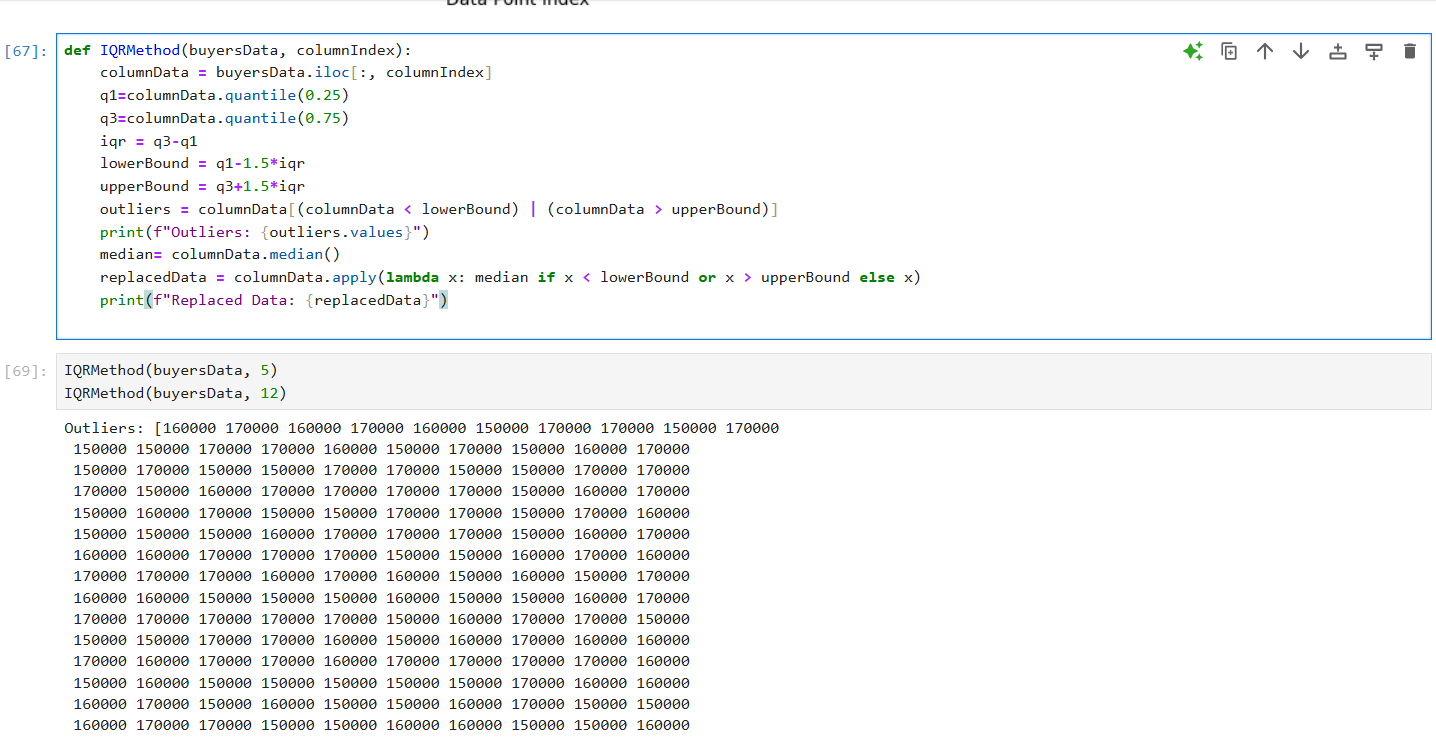
Description automatically generated A screen shot of a graph

Description automatically generated

An example histogram, box plot, and Z-score visual

### Calculating Outliers

Calculates the lower and upper bound to determine which values are outliers. Replaces data accordingly.



Screenshot of data outliers found.

### Discretionization

Sorts data passed according to the column index into bins.

A graph with numbers and text

Description automatically generated

An example of discretization with 5 bins.

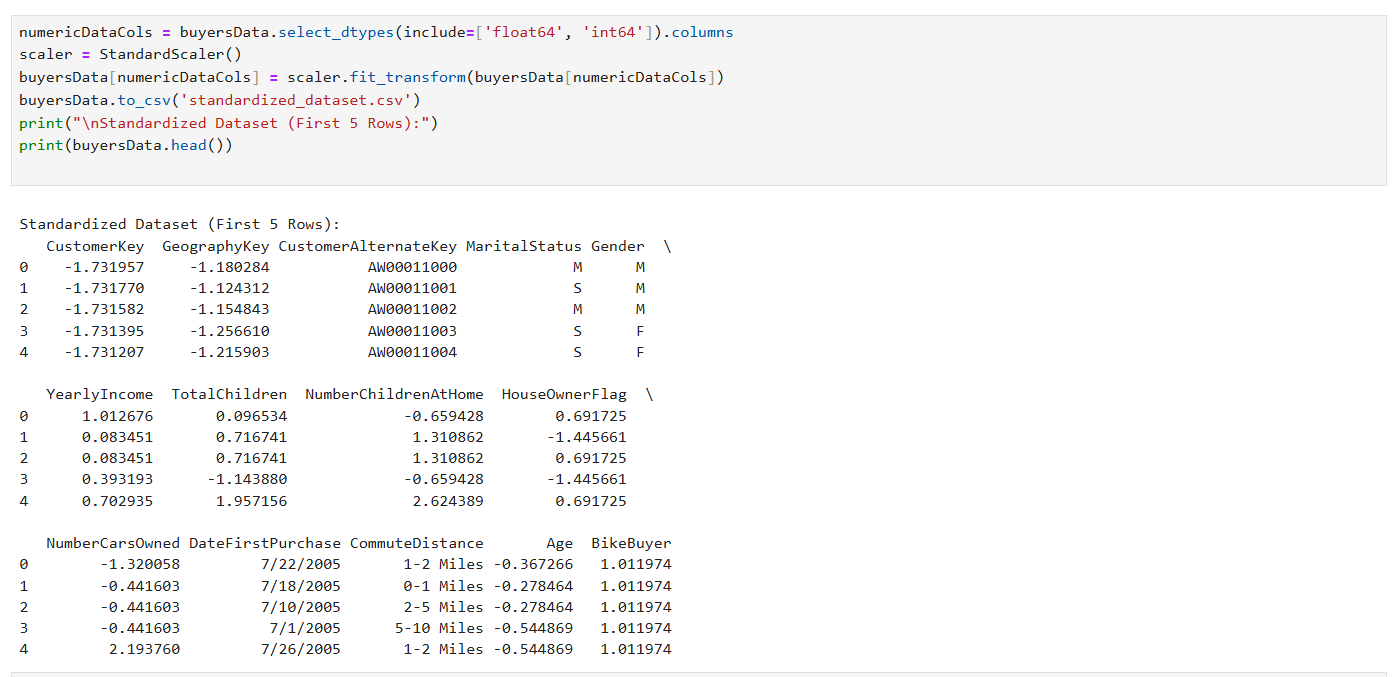
### Hot2Encoder

A screenshot of a computer

Description automatically generated

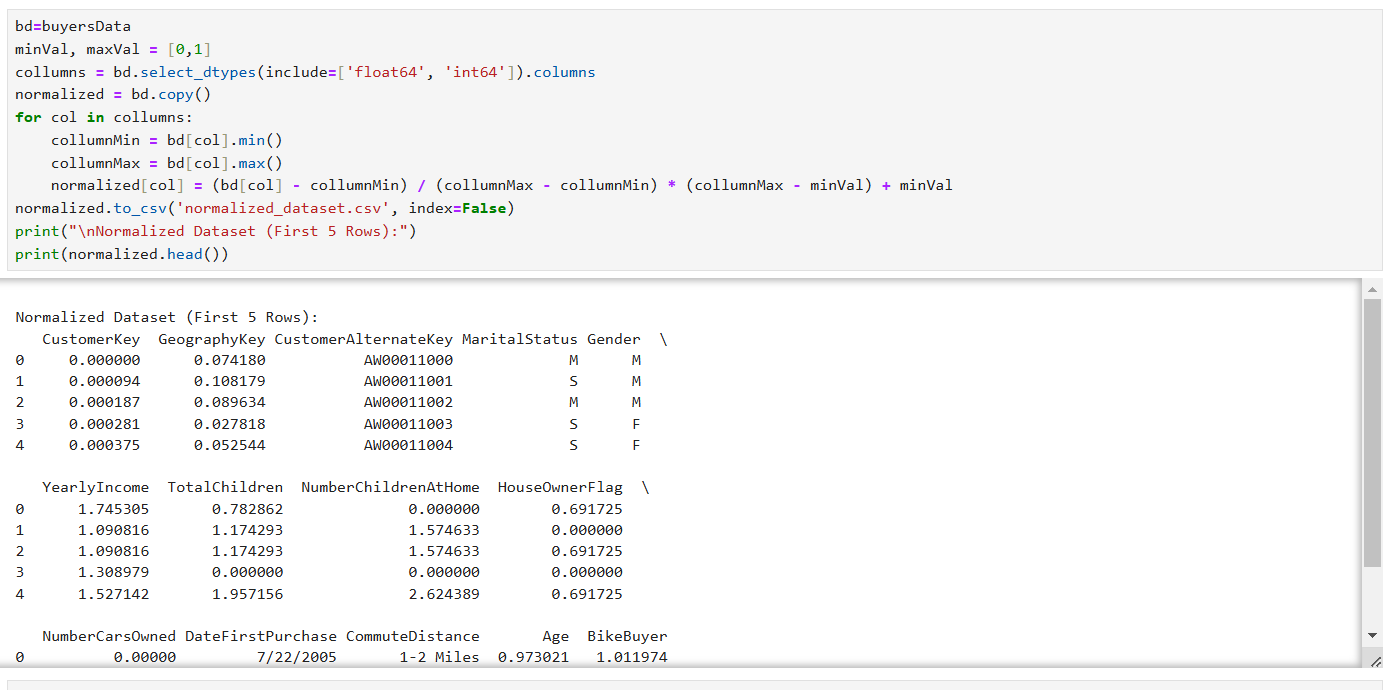
Example of the output for one-hot encoding.

### Column Standardization



Output of the first 5 rows as standardization.

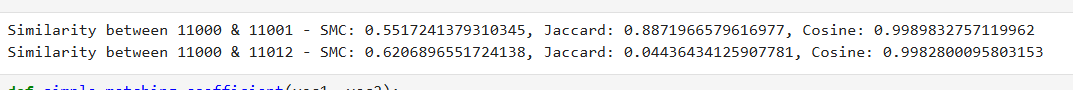
### Min-Max Normalization



Example of the new first five rows after min-max normalization.

## Part 3

After freshly normalizing the data, the code will look through the data set for a Customer ID that matches each of the passed ID’s and pull the vector for it. It will then compare using Simple Matching (just find the ratio of matching variables), Extended Jaccard Similarity (), and Cosine Similarity () to find the similarity between two customers.



## Part 4

Without the type of similarity being expressed in the assignment, I decided to keep it simple and use, well, a simple similarity matrix rather than Extended Jaccard or Cosine. To do this, I extracted the first 100 customers from the data set (leaving off the customer ID’s) and then created an array with length and height of 100. It will then use the simple similarity matrix function used in part 3 to find the similarity between each of the 100 customers, causing the lines of 1 down the diagonal (as a feature is 100% similar to itself) and causing similarity along said line, forming a reflection. This data is saved and a sample is show below.

