Melchizedek De Castro

CSC 59866

Yunhua Zhao

Assignment 1 – Summary

**Part A: Model Code**

The first part of this assignment requires the writing of multiple necessary functions that would be used for later parts. Functions such as the calculation of *Euclidean* and *Manhattan* distances, *accuracy* and *generalization* *error*, *precision*, *recall*, and *F1* *score*, *confusion* *matrix*, the generation of the *Receiver Operating Characteristic* (ROC) curve, the computation of the *Area Under Curve* (AUC) for the ROC curve, and the generation of a *precision-recall curve*. The final step of Part A also requires to implement our own KNN\_Classifier model class which consists of the methods *fit*which stores the relevant values needed for the KNN algorithms as instance variables, and most importantly – *predict* which predicts the labels for the given data sample.

During my construction of these functions, I mostly relied on comparing my results to sklearn’s corresponding library functions for reference and found it to be fairly accurate on most functions.

**Part B: Data Processing**

The second part of this assignment involves reading a csv file named *winequality-white* as a Pandas data frame. Initially, the original downloaded file had all of its contents on a single column and I had troubles parsing through them, so I had to do some unrelated work for separating the contents for future ease. After then, the ‘quality’ column needed to be converted into a two-category variable where ‘good’ has quality value of more than 5, and ‘bad’ has quality value of less than 6. The target vector should then have 0s to represent the ‘bad’ quality wines and 1s to represent the ‘good’ ones. To summarize each of the variables in the dataset, I had to use the pandas function *describe()* which generates the dataset variable mean, standard deviation, and quartiles. The next task was to shuffle the dataset, and generate the pair plot using seaborn. By having a visual representation of the dataset, I could identify which features were redundant which led to my conclusion that features ‘volatile acidity’ and ‘sulphates’ were redundant due to its similarities to their neighboring features. I then had to write a *partition* function to split the data into training and testing sets. Lastly, I had to naively run my KNN\_Classifier model on the training dataset with the neighbors set to 5. I then had to use accuracy and F1 score to compare my prediction to the expected labels, which outputs the accuracy of 0.8333333333333334 and F1 score of 0.0. Next is to standardize each feature of the training set using the mean and standard deviation of values for each of the features, then rerun the KNN\_Classifier model and find the new accuracy and F1 score. The output that I got for the new standardized data accuracy was 0.03592814371257485, which is very low. I’m not sure what caused that, which ate most of my time. Same thing for the last task which is the get the accuracy of the standardized data with inverse distance weight which also yield a number similar to the previous which is 0.037016875340228635.

**Part C: Model Evaluation**

The last part of this assignment requires us to evaluate the performance of our model over each combination of the *k* (1,5,9,11], *distance metrics* (Euclidean, Manhattan), and *weights* (uniform, distance)*.* We also had to report our performance with measures such as precision, recall, F1 score, Confusion Matrix, and accuracy.

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A computer screen shot of a computer code

Description automatically generated

A screenshot of a computer code

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer code

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screen shot of a computer code

Description automatically generated

A screenshot of a computer code

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer code

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A white background with black text

Description automatically generated

A grid of blue dots

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer code

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A computer screen shot of a program

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

