Credit Card Fraud Detection

Mohammed Alduris
Shannon Groth
Khaled Hossain

Contents

Introduction

Classifying a credit card activity as fraudulent or legitimate can be done in different approaches.

Multiple algorithm were applied to help obtain the most accurate results.

Why is fraud detection important?

Credit card fraud costs billions of dollars from the consumers end and from the financial companies end as well.

A fraud detection system might not be 100% effective, however it is a necessity when it comes to banks and other financial institutions.

Dataset - creditcard.csv

Users of credit cards and credit card companies want to avoid being charged for something they did not purchase through fraudulent charge.

The credit card dataset, creditcard.csv, contains 284,807 transactions made by European credit card users from September 2013 over a two-day period.

Algorithms

1. KNN
2. Perceptron
3. K-means
4. Naive Bayes
5. Random Forest
6. SVM
7. Passive Aggressive

Perceptron

The Perceptron classifier applies weights, w, and bias, b, to an input vector, x.
The Perceptron classifier uses a learning rate to determine the magnitude of change for weights during each step of training.
The classifier trains inputs and makes a prediction on the inputs.

K-means

The K-Means classifier uses a predetermined number of clusters in a dataset. In our code, the cluster number chosen is 2, since there are two classes in our dataset, 1 and 0.

SVM

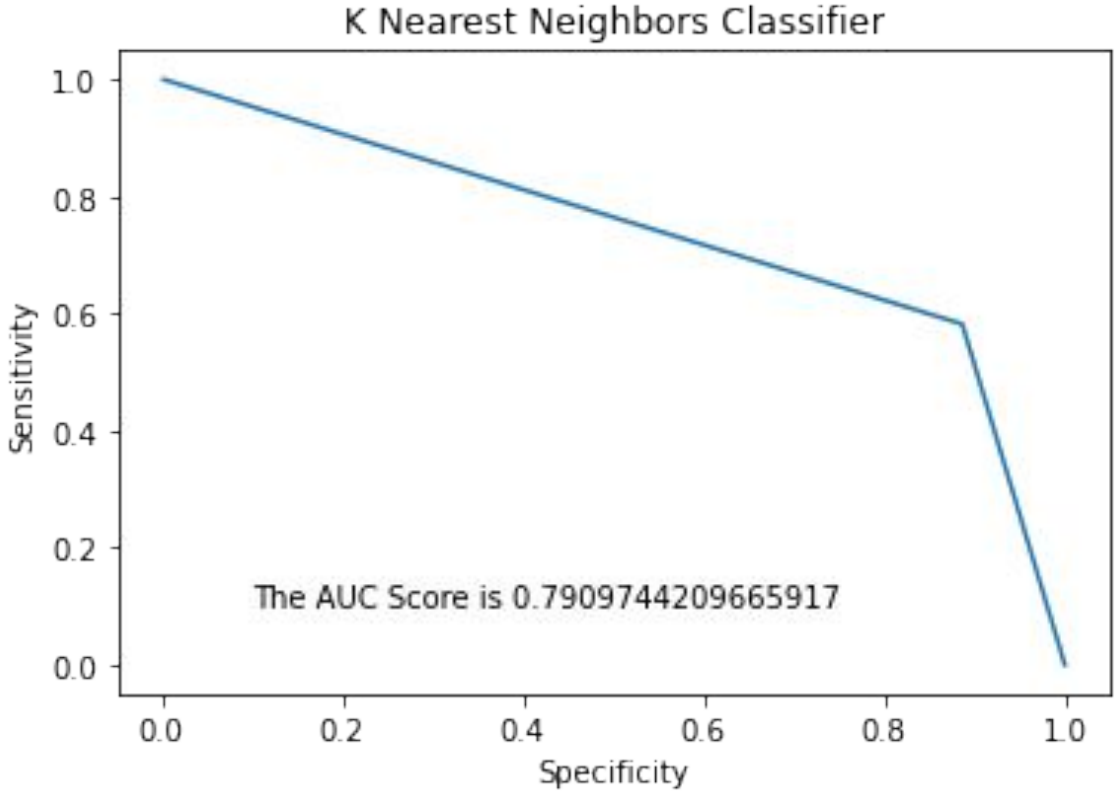
The Support Vector Machine(SVM) classifier is effective in high dimensional spaces and uses a subset of training points in the decision function so it is memory efficient.
The SVM takes the output of the linear function and separates the output into two classes depending on if the output is –1 or greater than 1.

Passive Aggressive Classifier

The Passive Aggressive Classifier uses a hyperplane to separate positive instances from negative instances and does not require a learning rate. If the classification prediction is correct, the model is kept. If the prediction is incorrect, the model is changed.

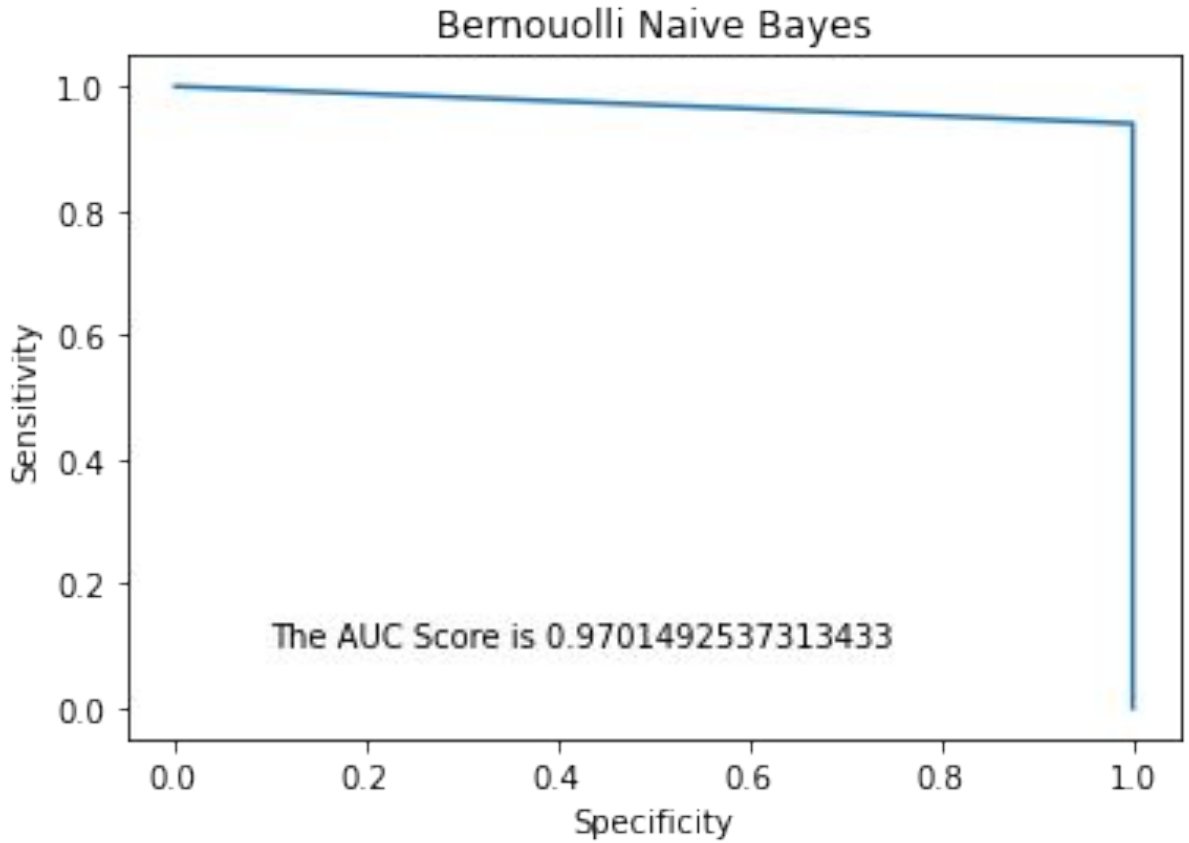
KNN

- The K-Nearest Neighbors(KNN) classifier finds a predefined number of training samples closest in distance to the new point and predicts the label from these.
- Neighbors-based classification stores instances of the training data rather than attempt to construct a general internal model.
- In our code, the number of neighbors chosen is 5, the generally default number for neighbors


K Nearest Neighbors Classifier
Sensitivity
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
The AUC Score is 0.7909744209665917

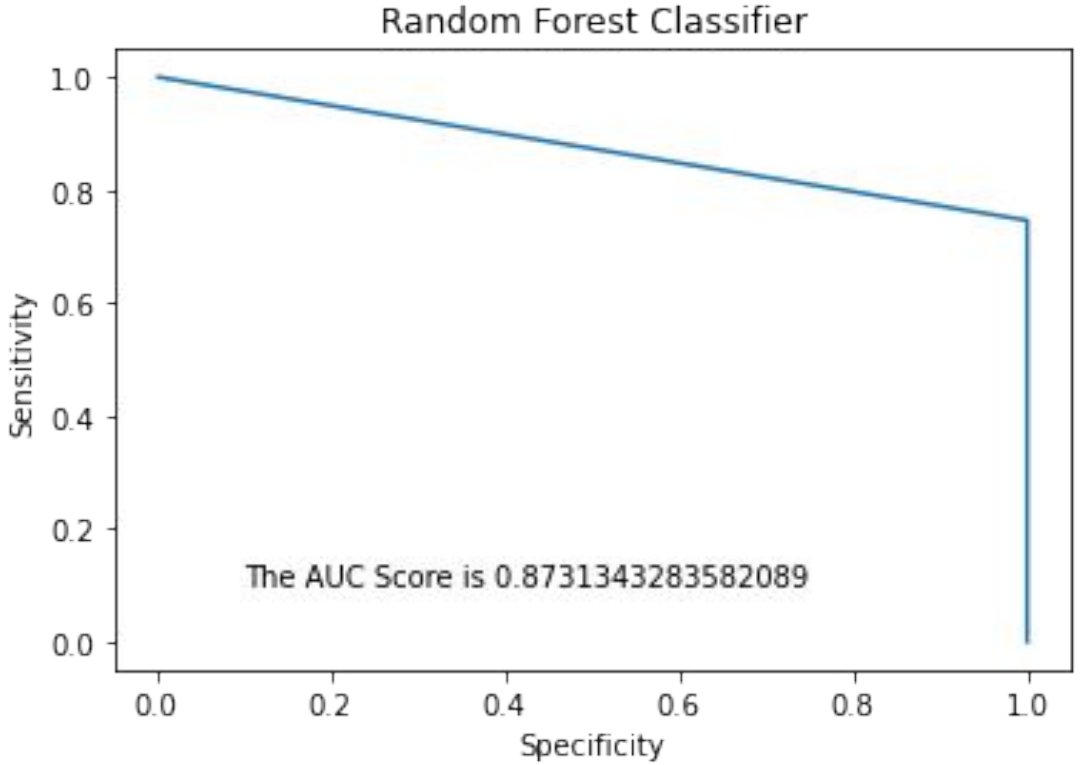
Naive Bayes

- A variant of Naïve Bayes, is ideal for a binary or Boolean dataset, of which the credit card dataset is since it is classified as either 1 or 0 with the y_test.unique() line of code.
- The Bayes Rule is a way of going from P(X|Y), known from the training dataset, to find P(Y|X).
- The Bernoulli Naive Bayes ROC AUC Score is: 0.9701492537313433
- The Bernoulli Naive Bayes AP Score is: 0.940410863857169 9
- Number of Bernoulli Naive Bayes mislabeled points out of a total 35601 points : 4


Bernouolli Naive Bayes
1.0
0.8
0.6
0.4
0.2
0.0
Sensitivity
The AUC Score is 0.9701492537313433
0.0
0.2
0.4
0.6
0.8
1.0
Specificity

Random Forest

- The Random Forest classifier consists of multiple decision trees where the most common decision tree output is the prediction.
- The Random Forest classifier picks n random records from the dataset to build a decision tree
- Number of Random Forest Classifier mislabeled points out of a total 35601 points : 17


Random Forest Classifier
Sensitivity
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
The AUC Score is 0.8731343283582089

Results - Which algorithms perform the best?

- Through using different classification algorithms, we were able to determine if credit card data contained a fraudulent transaction or not.
- The Passive Aggressive Classifier predicted the fraudulent vs genuine charges in the credit card dataset with the most accuracy of the algorithms implemented.

References

Cesarsouza, /. (2010, March 17). César Souza. Retrieved December 01, 2020, from http://crsouza.com/2010/03/17/kernel-functions-for-machine-learning-applications/

Countz, T. (2018, April 06). 19-line Line-by-line Python Perceptron. Retrieved December 01, 2020, from https://medium.com/@thomascountz/19-line-line-by-line-python-perceptron-b6f113b161f3

Gandhi, R. (2018, July 05). Support Vector Machine - Introduction to Machine Learning Algorithms. Retrieved December 01, 2020, from https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47

Passive Aggressive Classifiers. (2020, July 17). Retrieved December 01, 2020, from https://www.geeksforgeeks.org/passive-aggressive-classifiers/

References

Passive Aggressive Classifiers. (2020, July 17). Retrieved December 01, 2020, from https://www.geeksforgeeks.org/passive-aggressive-classifiers/

Real Python. (2020, November 07). K-Means Clustering in Python: A Practical Guide. Retrieved December 01, 2020, from https://realpython.com/k-means-clustering-python/

ULB, M. (2018, March 23). Credit Card Fraud Detection. Retrieved December 01, 2020, from https://www.kaggle.com/mlg-ulb/creditcardfraud

Thank you!