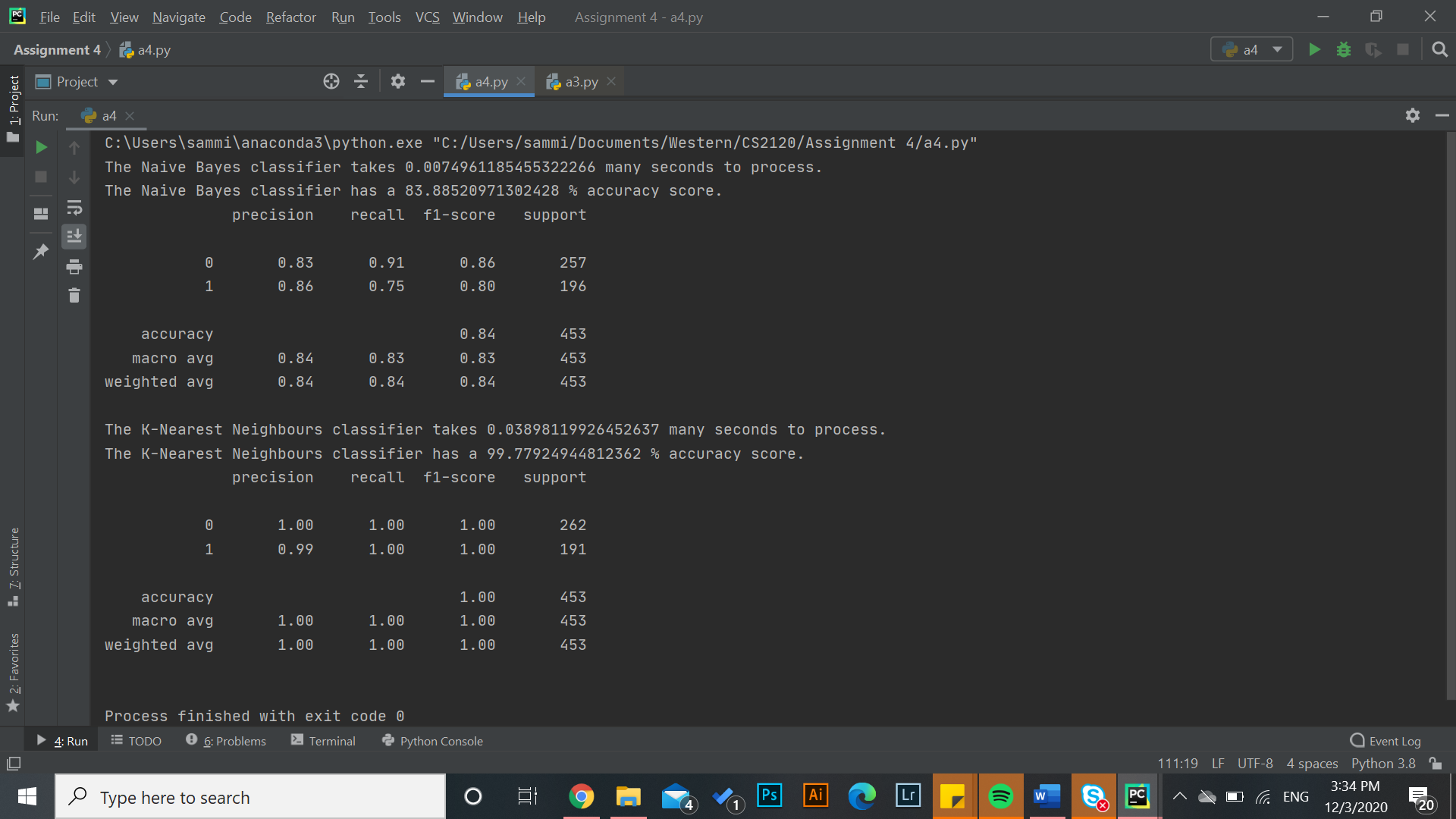
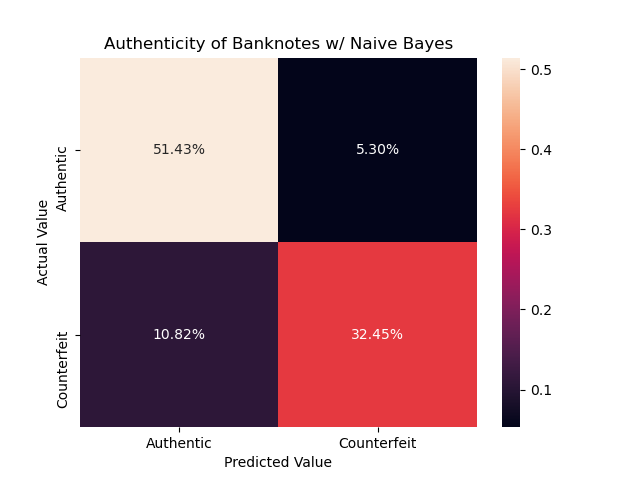
**Dataset**

The dataset used is the banknote authentication data set from UCI’s Machine Learning Repository. It can be found here: <http://archive.ics.uci.edu/ml/datasets/banknote+authentication>. In creating this data set, the researchers took photos of authentic and counterfeit banknotes using an industrial, print-inspection camera. The photos were in gray-scale and of 600 dpi resolution. The researchers then used a Wavelet Transform tool to elicit the features from the image; the Wavelet Transform tool helps analyze signals and reduce noise in a time and frequency domains (Liu, 2016). This dataset has 1372 instances and 5 attributes (1 class and 4 features). The four features analyzed in combination to determine if a banknote was authentic or counterfeit were the banknote image’s variance, skewness, curtosis, and entropy. Variance involves analysis of the discrete Fourier transform coefficients and discrete wavelet transform coefficients at the first and second decomposition levels (Dhanya, 2016). Skewness focuses on how the examined banknote’s specifications differ from normal distribution of authentic banknotes. Curtosis is similar and focuses on if the signals and noise levels of the examined banknote produces few or many outliers (Natrella, 2013). Entropy measures the degree of randomness based on the texture of the banknote image (Gonzalez, 2003). The machine learning approaches applied to this dataset were Naïve Bayes and K-Nearest Neighbours.

**Machine Learning Approach #1: Naïve Bayes**

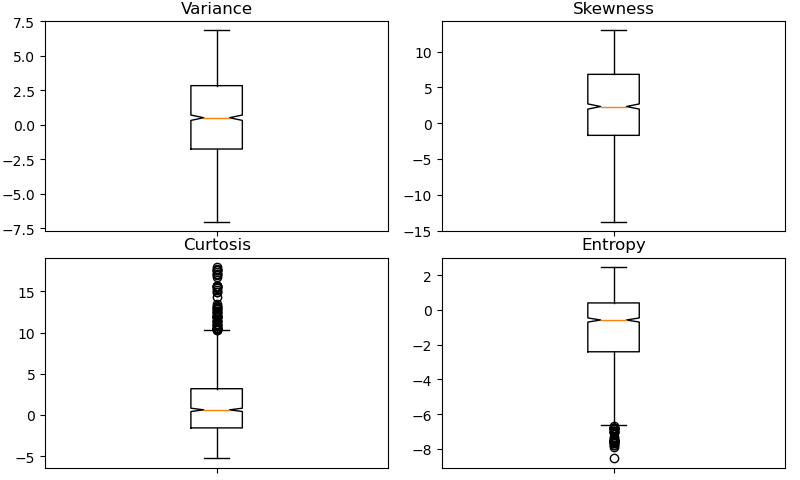
Using the Naïve Bayes classifier with a testing sample size of 33% resulted in an approximately 84% correct prediction rate of banknote authenticity. This approach is suitable for binary classification, which is why it was chosen for application to this dataset (either 0 or 1 to represent a banknote being either real or fake). This classifier had a fast run time, but Naïve Bayes is not perfectly suitable for the dataset as it assumes conditional independence between the features of variance, skewness, curtosis, and entropy (Verma, 2019). In reality, it is rarely the case that features involved in predicting the class are 100% independent.

****

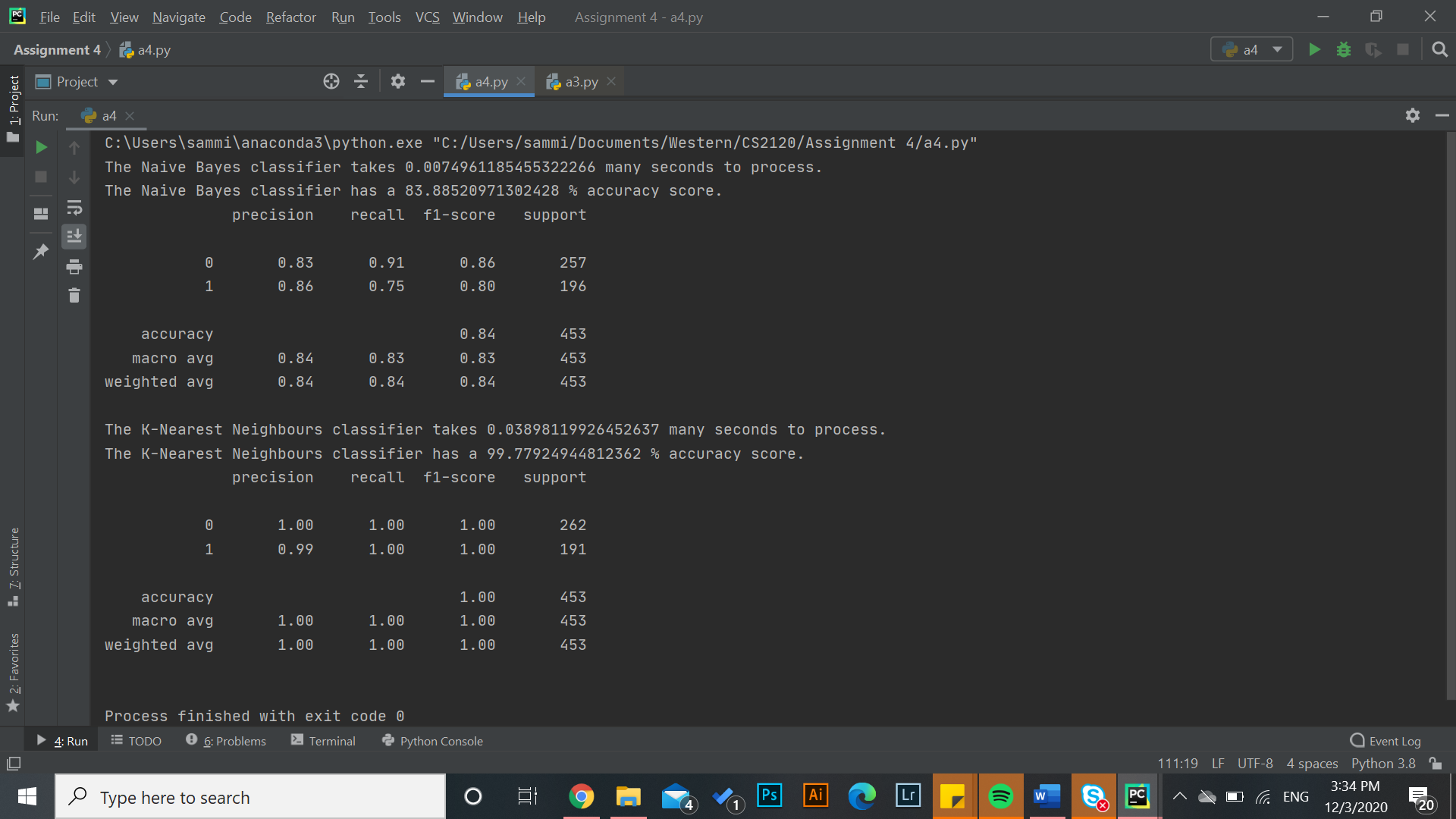
This confusion matrix visualizes the results from the Naïve Bayes classifier. It illustrates that there was approximately a 16% total error rate, with 11% coming from predictions of the banknote being counterfeit when it was authentic and 5% stemming from predictions of the banknote being authentic when it was counterfeit.

**Machine Learning Approach #2: K-Nearest Neighbours (KNN)**

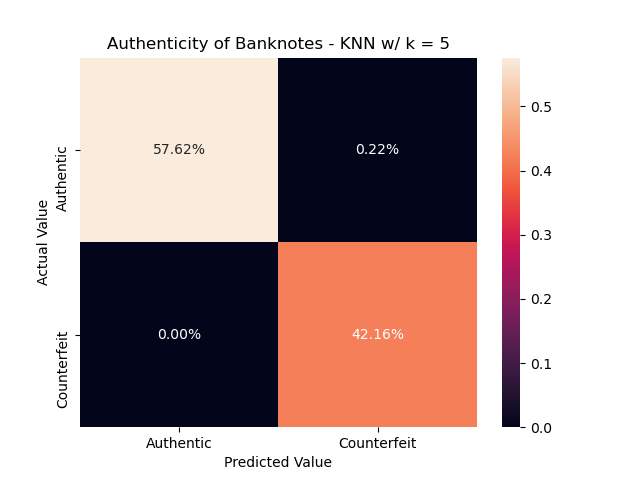
As the K-Nearest Neighbours classifier relies on plotted distances to assume similarity, group points and classify an input entry, it is sensitive to feature transformations and thus needs to be scaled (Dernoncourt, 2016). The Naïve Bayes classifier did not need to be scaled because its model was graphically based.



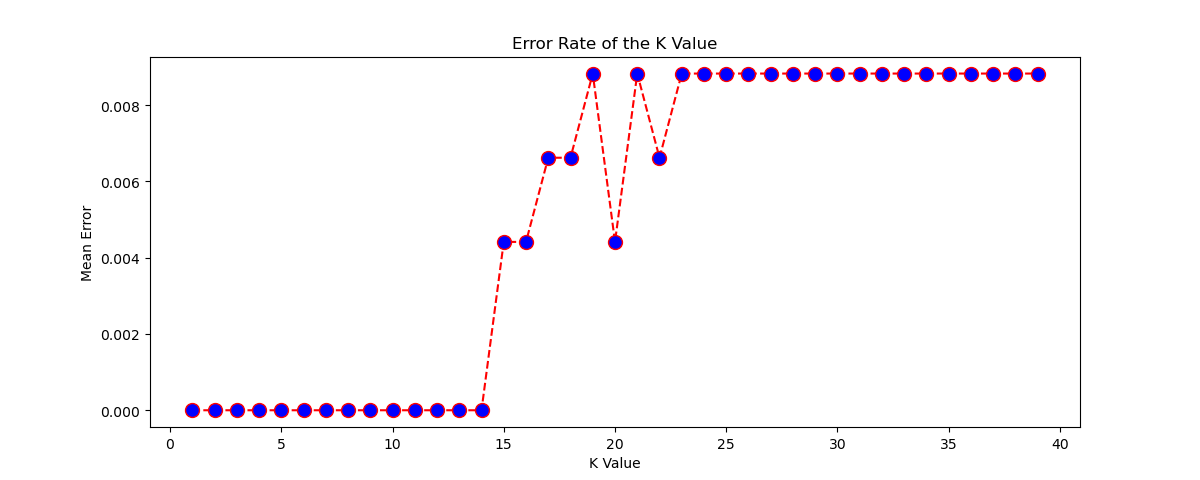
This diagram above shows how the features of curtosis, entropy, and skewness are not normally distributed prior to standard scaling. There are also many outliers for the curtosis and entropy features.



Using the KNN classifier post-scaling with a testing sample size of 33%, a near-perfect accuracy rate (99.8%) was achieved. Because the KNN algorithm does not make assumptions about the data’s distribution, it is suitable to be used for pattern recognition (Verma, 2019).



This confusion matrix illustrates the results from the KNN classifier. It shows that there was a 0% error rate, with correct predictions being comprised of 58% authentic banknotes and 42% counterfeit banknotes.



The K value is crucial in determining the KNN classifier’s accuracy rate because the K value is equivalent to the number of points considered in proximity to be of similar classification. In this approach, a K value of 5 was used as it is one of the most common values to use; it was a coincidence that this K value has a 0% mean error rate.

**Comparison and Contrast**

The Naïve Bayes classifier was five times faster than the KNN classifier at training, fitting, and predicting data, but the KNN classifier is considered better due to its 16% higher, near perfect accuracy rate. Factors that may have affected this accuracy rate would be the standardized scaling applied to the KNN classifier’s features. Also, because the KNN classifier is non-parametric, its decision boundary can take on any form and, therefore, it is the more flexible classifier of the two. Furthermore, if a K value above 20 was selected, the accuracy rate of the KNN classifier would be much lower. Additionally, both classifiers had a decreased accuracy rate for testing sample sizes greater than 40%. This is because most data is supposed to be used for training rather than testing.

**Citations**

Dernoncourt, F. (2016, December 27). What algorithms need feature scaling, beside from SVM? Retrieved December 04, 2020, from https://stats.stackexchange.com/questions/244507/what-algorithms-need-feature-scaling-beside-from-svm

Dhanya, S., & Kumari Roshni, V. (2016, January 20). Comparison of various texture classification methods using multiresolution analysis and linear regression modelling. Retrieved December 04, 2020, from <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4720626/>

Gonzalez, R. (2003). Entropy of grayscale image. Retrieved December 04, 2020, from https://www.mathworks.com/help/images/ref/entropy.html

Liu, Z. (2016, October 19). Wavelet Analysis. Retrieved December 04, 2020, from <https://www.sciencedirect.com/topics/earth-and-planetary-sciences/wavelet-analysis>

Natrella, M. (2013). 1.3.5.11. Measures of Skewness and Kurtosis. Retrieved December 04, 2020, from <https://www.itl.nist.gov/div898/handbook/eda/section3/eda35b.htm>

Verma, A. (2019, October 04). Machine Learning Classification - 8 Algorithms for Data Science Aspirants. Retrieved December 04, 2020, from https://data-flair.training/blogs/machine-learning-classification-algorithms/