MQE Big Data & Forecasting
Kiersten Kochanowski
HW #1

Introduction

The best classifier to predict quality of wine based on chemical characteristics - such as acidity, sulfates, and chlorides - is the Gaussian Naive Bayes classifier. This method predicts high quality wines with an accuracy of 74% on average.

Description of Method

I took various steps in order to determine the best performing classifier.

Data Cleaning

The first step was to understand the characteristics of the provided sample data. This is necessary in order to start identifying which classification methods are suitable for the type of information collected. The values of our variables ruled out a potential classifier and led me in the direction of a couple more. I learned from the data visualization process that there were some bottles of wine that had characteristics that were extremely outside of the norm. I removed these bottles from our sample as they could have significantly influenced our model since they were drastically less representative than the large majority, and likely a product of typos. Next I determined how dispersed our sample bottles were around the average of each of our biochemical features and found that a few of our variables had “skewed distributions” where some bottles were either above or below the average instead of evenly dispersed around that average. Given dispersion is an important characteristic and a necessary condition in order to use the Gaussian Naive Bayes classifier, I transformed the necessary predictor variables to satisfy the condition. Finally, I looked at the makeup of our sample and how many bottles scored a quality of 3, 4, 5, 6, 7, and 8. I found that the majority of the bottles in our sample scored a 5 and a 6. This can cause problems for our model since it is harder to predict more extreme scores with fewer examples; in order to improve accuracy, I split our sample into two groups (high and low quality) instead of six.

Applying Classifiers

Once the data was cleaned I tested a couple different Naive Bayes models that were founded on the idea of calculating the likelihood of wine having a certain quality based on the prevalence of its biochemical characteristics. They both hold the assumption that characteristics are independent of one another. This is not the case for our data but ultimately it’s not harmful that we don’t meet this assumption. The best predictor was the Gaussian Naive Bayes classifier which works well with variables that can take on any number (including decimals), such as ours. In order to assess the performance of my models I measured how often they predicted the right quality level and tested their performance over a variety of samples with our data.

Results

As shared above, our classification method predicts the quality level of wine with 74% accuracy on average. One limitation of this classifier is that it can only predict the quality of wine as one of two levels: low or high. Although this model is not tailored to recognize specific quality scores (ranging from 3 to 8), you can be more confident that the model has correctly identified the quality level of a bottle than you otherwise could be confident that the model correctly identified the bottle’s specific quality score (from 3-8). This is in part due to the imbalanced representation of quality score in our sample data. By predicting quality level, our model has a better chance of accurate prediction on bottles outside of our sample moving forward.

How to Apply in the Future

In order to continually improve the performance of this model, it is best to add new data as it becomes available to the company. As mentioned earlier, it would be ideal to collect more information on bottles which score rather low (3-4) and high (7-8) in order to ensure our sample is representative. In order to use the model moving forward, you can simply use the “model.predict” function and insert your wine’s characteristics. You can also work with your machine learning engineering team to put this into practice. It will likely take the form of an Application Programming Interface (API) which will allow you to interact with the software calculating the prediction whilst you input the parameters (features).