Machine Learning for Wine Color and Quality Classification

Frederik Stihler
Department of Statistics
University of California Berkeley

Stat154/254 (Fall Semester 2023)
Modern Statistical Prediction and Machine Learning
Instructor: Nikita Zhivotovskiy

Abstract

This project explores the predictive capabilities of linear regression, logistic regression, neural networks, and random forests in determining wine color and quality. Leveraging machine learning techniques, it aims to model and analyze the relationships between key attributes, offering insights into effective predictive models for wine characteristics.

1 Introduction

The wine industry, valued at 172.7 billion USD in 2023 [Statista], stands as a cornerstone of tradition and heritage. Introducing state-of-the-art technology an industry with such a rich history presents an intriguing prospect. This project aims to forecast wine color and quality exclusively based on physicochemical variables, not taking into account data concerning grape types, wine brands, or selling prices.

By leveraging various machine learning models, the project's focal point lies in comparing and evaluating the accuracy of different algorithms for classification tasks. Additionally, it seeks to contrast the outcomes of classification and regression techniques in predicting wine quality.

2 Dataset and features

The dataset was downloaded from the Kaggle website. It originates from the UCI Machine Learning Repository. The dataset is related to red and white variants of the Portuguese "Vinho Verde" wine. Due to privacy and logistic issues, only physicochemical variables such as "fixed acidity", "chlorides", "residual sugar", etc. are available. There is no data on e.g. grape types, wine brand or wine selling price.

There are 11 covariates in the dataset with physicochemical numerical values. The response variables are "Type", which is either white or red, and "Quality", which can take on integer values between 0 and 10. During the pre-processing stage, the data of the covariates was normalized using the StandardScaler function from ScikitLearn. The "Type" variable was encoded as a binary variable with white=0 and red=1.

After dropping empty values, the dataset has 6,463 entries. We first look at the distribution of red and white wine:

Template: Conference on Neural Information Processing Systems (NeurIPS 2023).

Figure 1: Distribution of wine types (Red and White).

Notice that only around a quarter of the wines are red. Next, we analyze the distribution of the wine quality classes. Only values between 3 and 9 are present (higher score indicates higher quality). Moreover, the classes are not balanced (e.g. there are many more normal wines than excellent or poor ones).

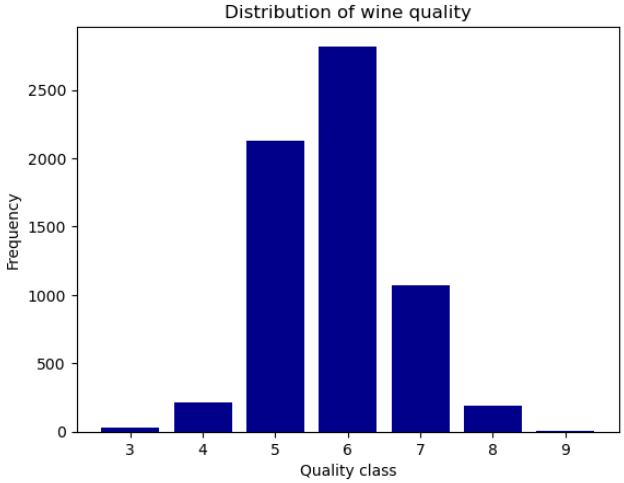

Figure 2: Distribution of wine quality (0-9).

3 Predicting wine color

The first classification task we want to perform is predicting the wine color based on the 11 covariates (physicochemical features). To do this, we split that data into training (70 %) and test data (30 %).

3.1 Logistic regression

We perform a binary logistic regression as our first classification method. We then calculate the Hamming loss to evaluate the performance. It is the fraction of labels that are incorrectly predicted. Let n be the number of data points predicted. Moreover, let y_i be the true label and $\hat{y}_i$ the predicted label. Then the Hamming loss is:

$$L_{hamming} = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{y_i \neq \hat{y}_i\}$$

The logistic regression model achieves a loss of approx. 0.4 % on the test data. This corresponds to an accuracy of approx. 99.6 %, which is almost perfect.

In addition to that, we calculate the log-loss (with true label k and p_{ik} the predicted probability of the true label for the i-th row):

$$L_{log} = \frac{1}{n} \sum_{i=1}^{n} log(\frac{1}{p_{ik}})$$

The log-loss turns out to be 1.5 % for our model.

3.2 Neural network

A neural network with one hidden layer of size 10 was chosen. The learning rate was selected to be 0.0005. ReLU was used as activation function in the hidden layer. The sigmoid function was applied to the final output of the network. The model was implemented using PyTorch. After training, the network achieved a Hamming loss of approx. 0.9% on the test data (corresponds to an accuracy of approx. 99.1%).

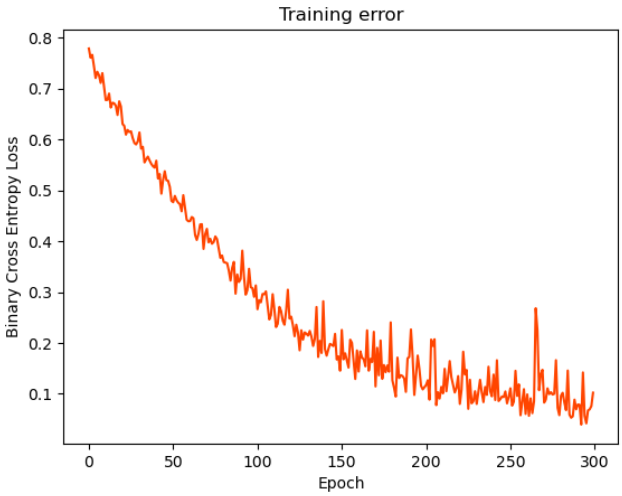

Figure 3: Binary cross entropy loss during training.

This is slightly worse than e.g. the logistic regression. Further tuning of the hyperparameters could improve the accuracy of the model.

3.3 Feature importance

Lastly, we compare our results to a Random Forest Classifier and use it to create an importance ranking of the features. The Random Forest Classifier achieved an accuracy of 99.7% on the test data.

Notice that, according to the model, the two most important features for predicting wine color is total sulfur dioxide and chlorides. If we fit a Random Forest Classifier only on these two features, we can already achieve a test accuracy of 97.8%. To investigate this further, we plot the two features against each other and visualize the corresponding wine color. The plot reveals a clear clustering, where it seems that red wines have higher chlorides and lower total sulfur dioxide compared to white wines.

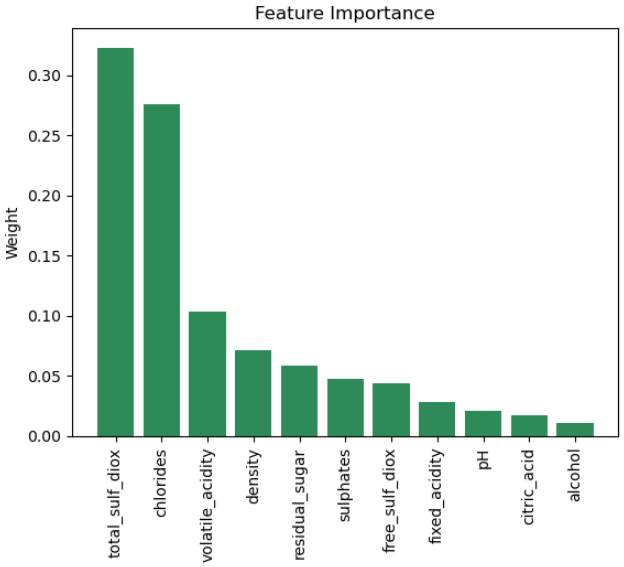

Figure 4: Feature importance based on Random Forest Classifier.

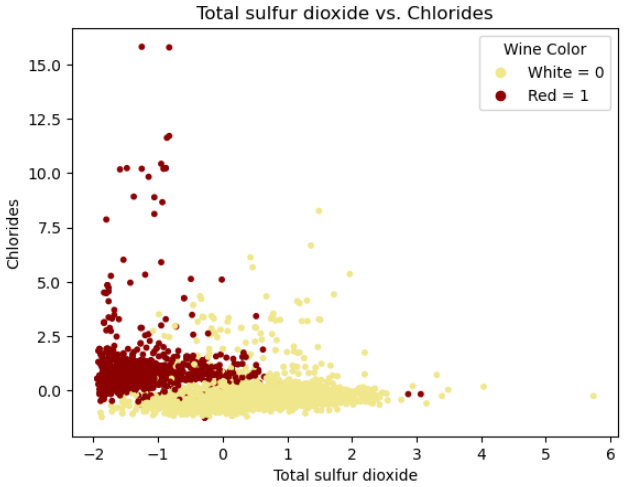

Figure 5: Pairwise plot of two most important features - total sulfur dioxide and chlorides.

4 Predicting wine quality

In our second experiment, we aim to predict the wine quality based on the covariates. The wine quality is expressed as an integer between 0 and 9. This represents an ordered categorical variable. It lies in between a pure categorical and a continous variable. Hence we want to compare how regression models perform versus classification models.

4.1 Linear Regression

First, we use linear regression. To convert the outputs to the integer classes of wine quality, we simply round the predictions of the linear regression. With this method, we achieve a Hamming loss of approx. 46.6% (corresponds to accuracy of 53.4%). That means that we have correctly classified around half of the test data points. However, we have no measure on how bad our prediction was on the misclassified data (interpreting the quality scale as a continous variable). To get a sense for this, we also computed the mean squared error:

$$L_{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

For linear regression, we achieve an MSE of approx. 0.59 on the test data.

4.2 Logistic regression

We repeat the experiment of wine quality classification with logistic regression, which resulted in a Hamming loss of approx. 45.1% (corresponds to accuracy of 54.9%). This is slightly higher better than our result with linear regression.

Yet, when we calculate the MSE, we notice that for logistic regression with approx. 0.62 it is slightly larger then for linear regression, which suggest that some of the predictions are a little further away from the true label.

Overall, we notice that even though the accuracy scores for both models are only around one half, the rather low MSE suggests that the predictions are not very far off from the truth.

4.3 Random Forests

As a final experiment, we compare the performance of a regression random forest against the performance of a classification random forest (rounding the results of the regression random forest to integers). Both achieve a very similar result of approx. 32% Hamming loss.

Finally, notice that this is significantly better compared to the results of linear regression and logistic regression.

5 Conclusion

In this project, we observed variations in performance among different machine learning models when predicting wine characteristics. Predicting wine color proved to be relatively straightforward across models. However, accurately predicting wine quality presented greater challenges, possibly due to inherent complexities or limitations in the scoring method for quality assessment. Introducing an error tolerance could notably enhance prediction outcomes. Lastly, our findings suggest that regression and classification methods exhibit comparable performance within this context, highlighting their similar efficacy for wine quality prediction.

References

Market Insights Statista. Worldwide wine revenue. URL https://www.statista.com/outlook/cmo/alcoholic-drinks/wine/worldwide.

Machine Learning Repository UCI. Wine quality data. URL https://archive.ics.uci.edu/dataset/186/wine+quality.