Predicting Style and Quality of Vinho Verde from its Physical Properties

Grigor Vardanyan MS student

Yerevan State University

Faculty Statistics: Applied Statistics and Data Science

Introduction

Nowdays there are lot of Sommelier (wine testing experts), who is manually test wines and tells us about wine quality. It is mean there is some functional dependency between properties of wines and quality. In this paper I will introduce some machine learning models for predicting wine quality based on its properties/features as well we will try distinguishing between red and white wines.

According to the International Organization of Vine and Wine (OIV) during a year is produced millions of tons of wine and there is a need to understand quality of each separate wine. There are some researches to create sensors which based on chemical or physical properties try to understand quality of wine.

The dataset has 6497 rows. The dataset is unevenly split between two styles: 75% of examples are of white wines (4898) and 25% are of reds (1599) (Fig 1). From point of quality we here also have unbalanced split of data between 7 qualities: 3 - 30, 4 - 214 ,5 - 2128, 6 - 2820, 7 - 1074, 8 - 192, 9 – 5 (Fig 2).

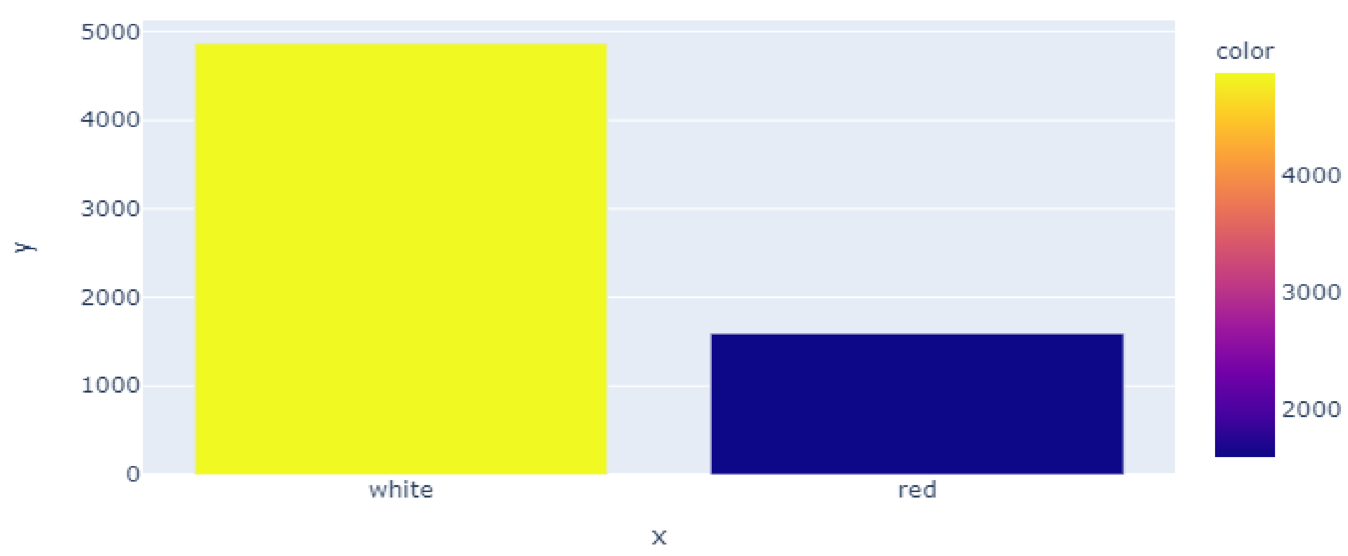

Fig. 1. Count of training data for each type of wine. Left white wine. Right red wine.

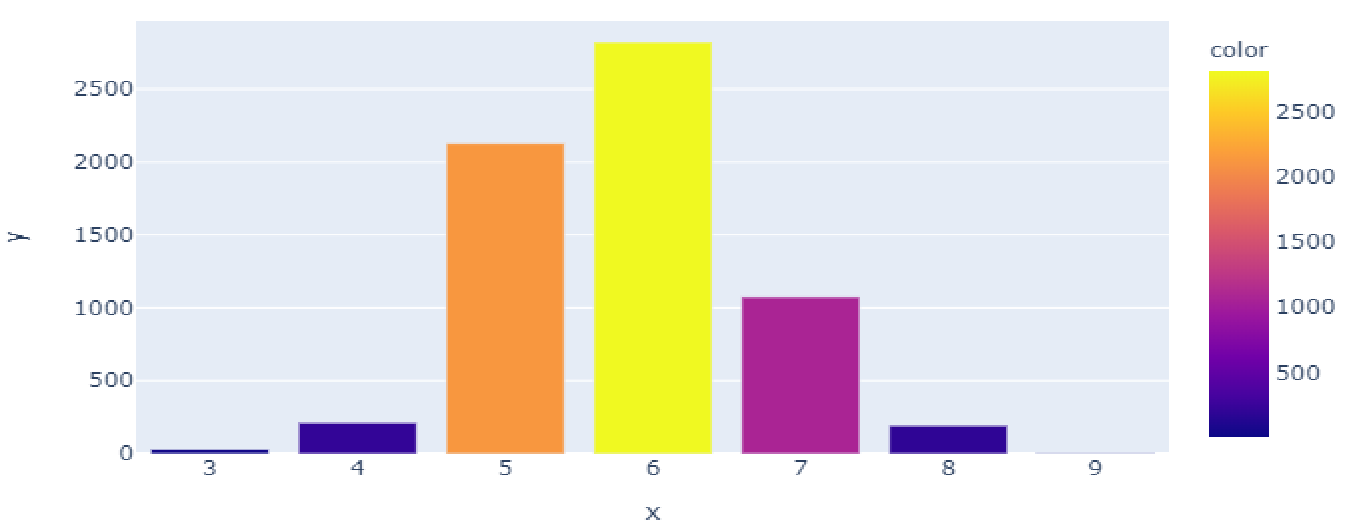

Fig. 2. Count of training data for each quality of wine.

Methods

Data exploration: During data exploration we have found that data is well separable for wine type classification (Fig 3).

Fig. 3. Visualization for type of wine in 2D, where axes are first 2 components.

Data preprocessing: For data preprocessing I removed missing values from dataset and scaled values with Standard Scalar of Sckit-learn. There was an idea also try to reduce dimensionality of data with PCA, but such trick was done in Kaggle, so I did not repeat it.

Models: For Vine type prediction I used Logistic Regression [1] and XGBoost [2]. For quality prediction I used Random Forest [6] and XGBoost [2] classifiers. In results section we will see all results of experiments. The reason of using XGBoost [2] is, as we have less data from 3, 4, 8, 9 qualities, if previous tree did not guess current wine quality, current tree will try to fit on residual of previous one.

Results

Wine type prediction: I have found that wine type predictions is not so hard task. Both classes is very linearly separable even though in high dimensional space. You can see evaluation of Logistic regression [1] model on test data (Fig 4) and evaluation of XGBoost [2] model (Fig 5).

	precision	recall	f1-score	support
white	0.99	0.99	0.99	529
red	1.00	1.00	1.00	1604
accuracy			1.00	2133
macro avg	0.99	0.99	0.99	2133
weighted avg	1.00	1.00	1.00	2133

Fig. 4. Logistic Regression evaluation

	precision	recall	f1-score	support
white	0.99	0.99	0.99	529
red	1.00	1.00	1.00	1604
accuracy			1.00	2133
macro avg	0.99	0.99	0.99	2133
weighted avg	1.00	1.00	1.00	2133

Fig. 5. XGBoost evaluation

Wine quality prediction: Wine quality prediction is more hard task than type prediction. From Fig 2 is obvious how big is difference of frequency between classes . For this approach we used Random Forest [6] and Xgboost [2] model.
From Fig 6 we see that Xgboost [2] was not able to predict lowest frequency classes.

Next thing I tried to group data between low (3 and 4 quality), middle (5, 6 and 7 quality) and high (8 and 9). With this trick we get more good results on low quality and middle quality (Fig 7).

	precision	recall	f1-score	support
3	0.00	0.00	0.00	8
4	0.37	0.10	0.16	67
5	0.71	0.69	0.70	721
6	0.63	0.76	0.69	912
7	0.66	0.51	0.57	356
8	0.87	0.29	0.44	68
9	0.00	0.00	0.00	1
accuracy			0.66	2133
macro avg	0.46	0.34	0.37	2133
weighted avg	0.66	0.66	0.65	2133

Fig. 6. Evaluation of XGBoost model for quality prediction

	precision	recall	f1-score	support
middle	0.77	0.57	0.65	425
high	0.36	0.05	0.09	75
low	0.86	0.95	0.90	1633
accuracy			0.84	2133
macro avg	0.66	0.52	0.55	2133
weighted avg	0.82	0.84	0.83	2133

Fig. 7. Evaluation of XGBoost model for quality prediction

Results of Randm Forest [2] was not so high. More details is written in paper.
There was an idea to balance data or reduce most frequent classes and train model on balanced data.

	precision	recall	f1-score	support
3	0.00	0.00	0.00	11
4	0.55	0.48	0.51	79
5	0.49	0.52	0.51	77
6	0.34	0.31	0.33	64
7	0.31	0.38	0.34	61
8	0.46	0.53	0.49	64
9	0.00	0.00	0.00	2
accuracy			0.43	358
macro avg	0.31	0.32	0.31	358
weighted avg	0.42	0.43	0.43	358

Fig. 8. Evaluation of XGBoost model on reduced data

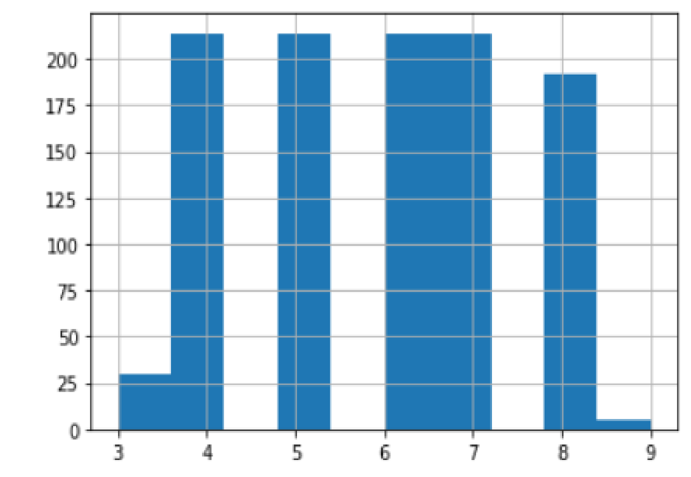

Fig. 9. Reduced data frequency

Conclusion

For type classification best model for us is Logistic regression based on precision and recall score (Fig 4). Same result we received from Xgboost [2], but as Logistic regression [1] is more simple(based on properties compared to XGBoost [2], hyperparameters e.t.c) model, we choose it rather than XGBoost [2].

For quality prediction, decision of choosing model is more subjective. If we want predict very well low and middle quality ,in this case we should use XGBoost [2] with target variable transformation trick(grouping some qualities into one group). But if we want to predict very well quality 8 in this case we can choose XGBoost [2] model without transformation trick. So decision of using model depends from business needs.

Reduction of classes, which has high frequency, did not helped us a lot. Still we have less data (e.g from 9 class 5 samples vs 6900), which describes low accuracy on this class. So in the future we will not use for this problem reduction trick.

References

1. . D. R. Cox, “The Regression Analysis of Binary Sequences,” Journal of the Royal Statistical Society. Series B (Methodological), vol. 20, no. 2, pp. 215–242
2. Tianqi Chen, Carlos Guestrin “XGBoost: A Scalable Tree Boosting System”
3. S. Tempere, S. Per´ es, A. F. Espinoza, P. Darriet, E. Giraud-H ` eraud, ´ and A. Pons, “Consumer preferences for different red wine styles and repeated exposure effects,” Food Quality and Preference, vol. 73, pp. 110–116, apr 2019\
4. Jerome H. Friedman – “Greedy function approximation: A Gradient Boosting Machine”.
5. Petro Liashchynskyi, Pavlo Liashchynskyi - Grid Search, Random Search, Genetic Algorithm: A Big Comparison for NAS
6. Leo Breiman - RANDOM FORESTS