Project discussion: dataset on wine quality

Data Driven Decision project

M. Piccirilli G. Viozzi M.Caradio

June 6, 2025

Table of Contents

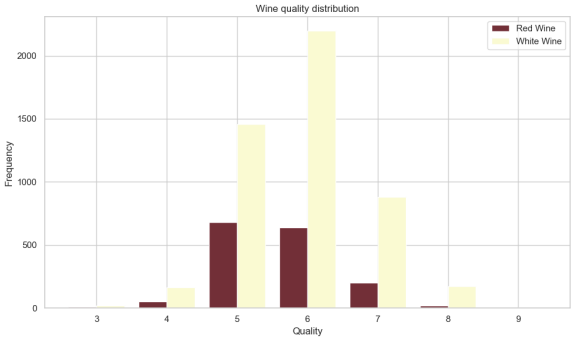
Studied problem

It is possible to predict wine quality (from 1 to 10) given some observations about it?

The problem can be interpreted as both a regression and **classification** problem.
We chose to study it as a classification problem because:

- With an imbalanced dataset, regression models tend to predict values near the common central classes, neglecting rare cases.
- Predictions are continuous values, which don't exist in the dataset. Rounding these values introduces additional errors.

Quality of wine distribution



- Red wine dimensions: 1599 rows, 12 columns.
- White wine dimensions: 4898 rows, 12 columns.
- Features: fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, alcohol.
- Target feature: quality.

We conducted preliminary experiments in order to determine the best dataset configuration to work on. The considered configurations were the following:

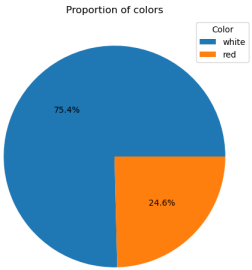
- Keeping the red and white datasets separated
- Joining the two datasets without balancing the red and white samples.
- Joining the datasets and applying oversampling
- Joining the datasets and applying undersampling

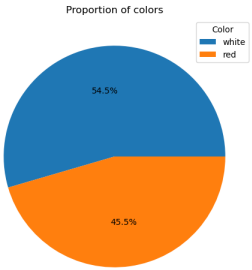
After training some simple models on the different configurations we got the following **accuracy** results:

Model	Separated	Joined (no balance)	Joined (oversample)	Joined (undersample)
Decision Tree	0.5864	0.6115	0.6956	0.5391
SVC	0.5822	0.5608	0.6194	0.5875
MLP	0.5935	0.5677	0.6450	0.5609

We made similar experiments using **balanced accuracy** with same conclusions.

Distribution of the colors

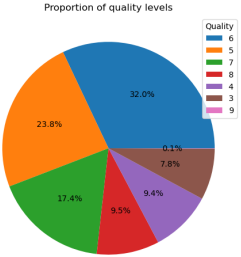


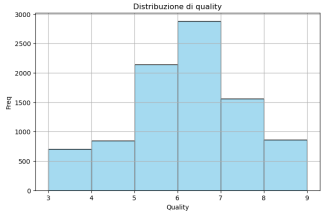


The pictures above show the proportions of samples for each color before and after we decided to oversample the joined dataset.

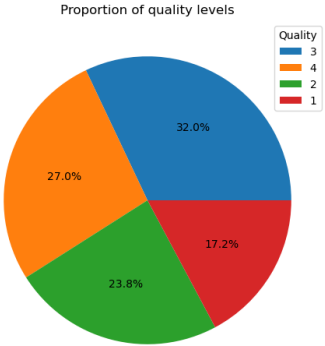
Distribution of the classes 1/2

Before the preprocessing of the classes, we can notice that they were pretty unbalanced, and some classes did not have any representation.





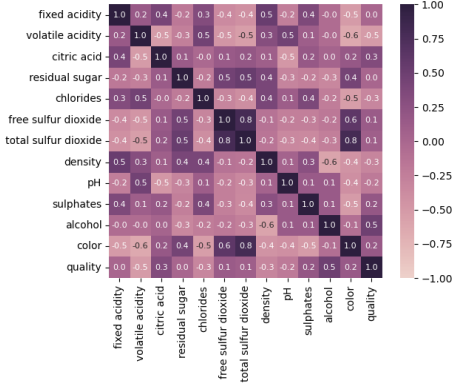
Distribution of classes 2/2



- We regrouped the classes as follows:
  - classes 3 and 4 become class 1
  - class 5 becomes class 2
  - class 6 becomes class 3
  - classes 7, 8 and 9 become class 4
- Then, since we still had minority classes, we also oversampled the classes with less representation

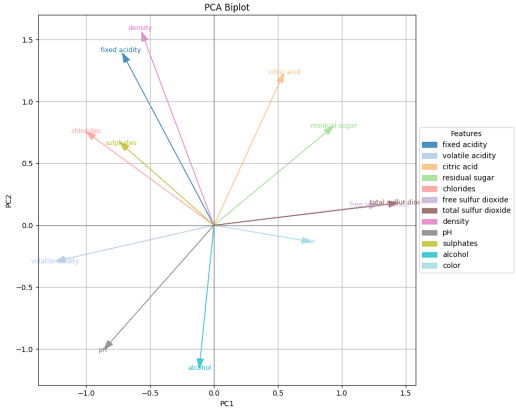
Note

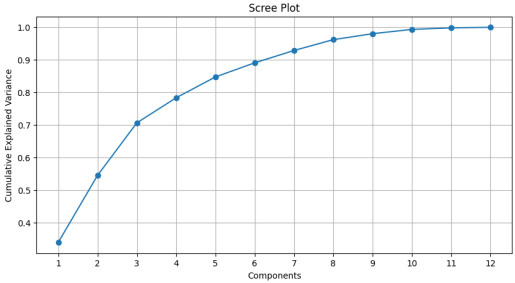
For the categorical values of the y target (quality) we did not use One Hot Encoding since they do follow some sort of ordering.



As we can see from the correlation matrix the only features that are strongly correlated are:

- *total sulfur dioxide* and *free sulfur dioxide*
- *total sulfur dioxide* and *color*


PCA Biplot
density
fixed acidity
citric acid
residual sugar
chlorides
sulphates
free sulfur dioxide
total sulfur dioxide
pH
alcohol
volatile acidity
color
PC1
PC2
Features
fixed acidity
volatile acidity
citric acid
residual sugar
chlorides
free sulfur dioxide
total sulfur dioxide
density
pH
sulphates
alcohol
color


Scree Plot
Cumulative Explained Variance
Components

We considered four possible models for this problem:

- Logistic Regression
- Support Vector Machine
- Decision Trees
- MLPClassifier (Multilayer Perceptron Classifier)

Results

After conducting preliminary experiments without any optimization techniques we realized that the **Decision Trees** and the **MLPClassifier** provided the best results in term of accuracy, thus we conducted a preliminary **Grid Search** on these models to determine the best model with the best hyperparameters.

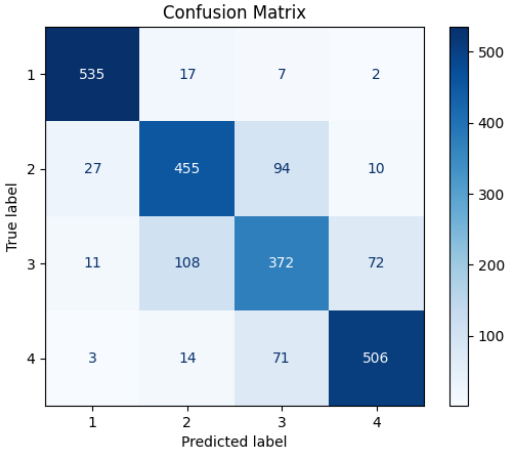
After the Grid search we came to the conclusion that the best model is the **MLPClassifier**, on which we conducted a wider Parameter-space Grid Search using the following hyperparameters:

- hidden_layer_sizes: [(128,), (128, 64), (128, 64, 32)]
- activation: ['relu', 'tanh']
- solver: ['adam', 'sgd']
- alpha: loguniform(1e-5, 1e-1)
- learning_rate: ['constant', 'adaptive']
- learning_rate_init': loguniform(1e-4, 1e-1)

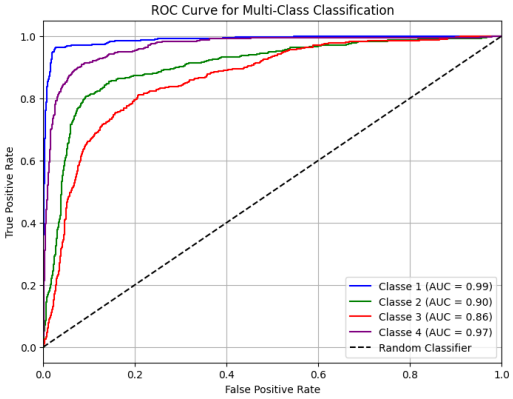
- Accuracy obtained: 0.8108.

Classe	Precision	Recall	F1-score	Support
1	0.93	0.95	0.94	561
2	0.77	0.78	0.77	586
3	0.68	0.66	0.67	563
4	0.86	0.85	0.85	594
Accuracy		0.81		2304
Macro avg	0.81	0.81	0.81	2304
Weighted avg	0.81	0.81	0.81	2304

- The model behaves pretty well on class 1 ($F1 : 0.94$) and 4 ($F1 : 0.85$), which represent the minority classes in the original dataset. This is due to the predictability introduced by oversampling.
- The model has lower performance on class 3 ($F1 : 0.67$), which represents the two majority classes in the original dataset. This is due to the spread of the original collected features.



- 94 samples of class 2 are predicted as class 3 and 108 for the opposite case, i.e. the 19% of samples which has 'quality' 2 or 3 are predicted erroneously.
- A very small number of class 1 (resp. class 4) samples are predicted as class 4 (resp. class 1) and this is very important because we want to avoid that low quality wine are classified as good one.



- All 4 classes show **AUC** above 0.86, with Classes 1 and 4 achieving near-perfect performance, but, anyways, all **ROC curves** are well separated from the random classifier line.
- Class 3 (UAC=0.86), originally class 6, the majority one, show the lowest performance in the group, suggesting it may be harder to separate from other classes.

Limitations

- We can see that class 3 shows inferior performances (F1 = 0.67), this suggests some difficulties in distinguishing central classes, probably due to not highly discriminating features.
- The classes have been regrouped, thus the final prediction is not fine-grained.
- The MLP model, is a **black box**, thus it does not leave much room for interpretability

Future Work

- Using advanced balancing techniques for the generation of **synthetic data**.
- Exeriment with **ensemble methods**. We could test models like Random Forest, XGBoost etc. for better robustness.
- Trying different techniques to guarantee model **explainability**.
- We could also give a try to **transfer learning** if partially labeled data are available.

Conclusions

- We obtained a model from a dataset where we did oversampling for classify the quality of a wine with accuracy of 0.81.
- We tested also the separated dataset, red and white wine, on both of which we obtained an accuracy of 0.80.

Therefore, both approaches are valid, however we prefer the strategy of using the combined dataset. This allows us to:

- Avoid building and maintaining two separate models.
- Simplify the deployment pipeline, without having to instantiate different models based on the color of the wine.