Wine quality prediction project

Project of predicting wine quality based on physicochemical properties.

By: Juan Vicente Peluso

GitHub notebook

Table of Contents

1. Introduction

About *Vinho Verde*

Vinho Verde refers to Portuguese wine that originated in the historic Minho province in the far north of the country. The modern-day Vinho Verde region, originally designated in 1908, includes the old Minho province plus adjacent areas to the south. In 1976, the old province was dissolved.

Vinho Verde is not a grape variety, it is a DOC for the production of wine. The name means *green wine* but translates as *young wine*, with wine being released three to six months after the grapes are harvested. They may be red, white, or rosé and they are usually consumed soon after bottling.

For further information, you can go to their official web.

The dataset, features and target value

This dataset is available from the UCI machine learning repository (*source*). The inputs include objective tests (e.g. PH values) and the output is based on sensory data (median of at least 3 evaluations made by wine experts). Each expert graded the wine quality between 0 (very bad) and 10 (very excellent). The features in the dataset are:

- **fixed acidity**: Most acids involved with wine or fixed or nonvolatile (do not evaporate readily).
- **volatile acidity**: The amount of acetic acid in wine, which at too high levels can lead to an unpleasant, vinegar taste.
- **citric acid**: Found in small quantities, citric acid can add 'freshness' and flavor to wines.
- **residual sugar**: The amount of sugar remaining after fermentation stops.
- **chlorides**: The amount of salt in the wine.
- **free sulfur dioxide**: The free form of SO2 exists in equilibrium between molecular SO2 (as a dissolved gas) and bisulfite ion; it prevents microbial growth and the oxidation of the wine.
- **total sulfur dioxide**: Amount of free and bound forms of S02; in low concentrations, SO2 is mostly undetectable, but at free SO2 concentrations over 50 ppm, SO2 becomes evident.
- **density**: The density of the wine is close to that of water.
- **pH**: Describes how acidic or basic a wine is on a scale from 0 (very acidic) to 14 (very basic.
- **sulphates**: A wine additive which can contribute to S02 levels, acting as an antimicrobial and antioxidant.
- **alcohol**: The alcohol percentage present in the wine.
- **quality**: Output/target variable (based on sensory data, score between 0 and 10).

Problem description

As we know, a ***Sommelier*** or wine expert, is a trained and knowledgeable wine professional, who specializes in all aspects of wine service, as well as wine and food pairing. The quality of the wine record of this dataset has been given by sommeliers, using their sensory expertise.

Can we match the Sommelier sensory scoring with a machine learning algorithm, saying if a wine is good or not? Can the physicochemical properties of the wine dictate its quality, and can we develop a model that supports it?

If a winegrower could know beforehand, which approximate quality his wine will have, it could present innumerable advantages, such as:

- If a is ***not good***, the fees for having Sommeliers can be saved, have your marketing team working on an idea on how to place it in the market, knowing it won't be an outstanding product.
- If a batch **is *good***, you can have top Sommeliers and raise the budget for the marketing campaign, knowing the product is outstanding, and that the investment will translate into profit.

With the data available, we will develop a machine learning model, that will try to predict whether the ***wine quality is good or not good***, based on the features available.

2. Data quality check and target feature creation

The dataset has 1599 records (wines analyzed) with no missing or NaN values. As our hypothesis is to create a model to predict either a wine is good or bad, we've updated the values of the target variable ***Quality*** to **0** for wines labeled from 3 to 6, considered as *not good*, and to **1** to those labeled 7 or 8, considered as *good*.

After updating the target value, we've found that **240** duplicate records exist, meaning that, in the original dataset 2 wines with the same values are labeled with a different value. This might impact the model predictive power, let's keep that in mind.

3. Descriptive statistics

General statistics

At first sight, after computing the basic statistics of the dataset, we can draw the following conclusions:

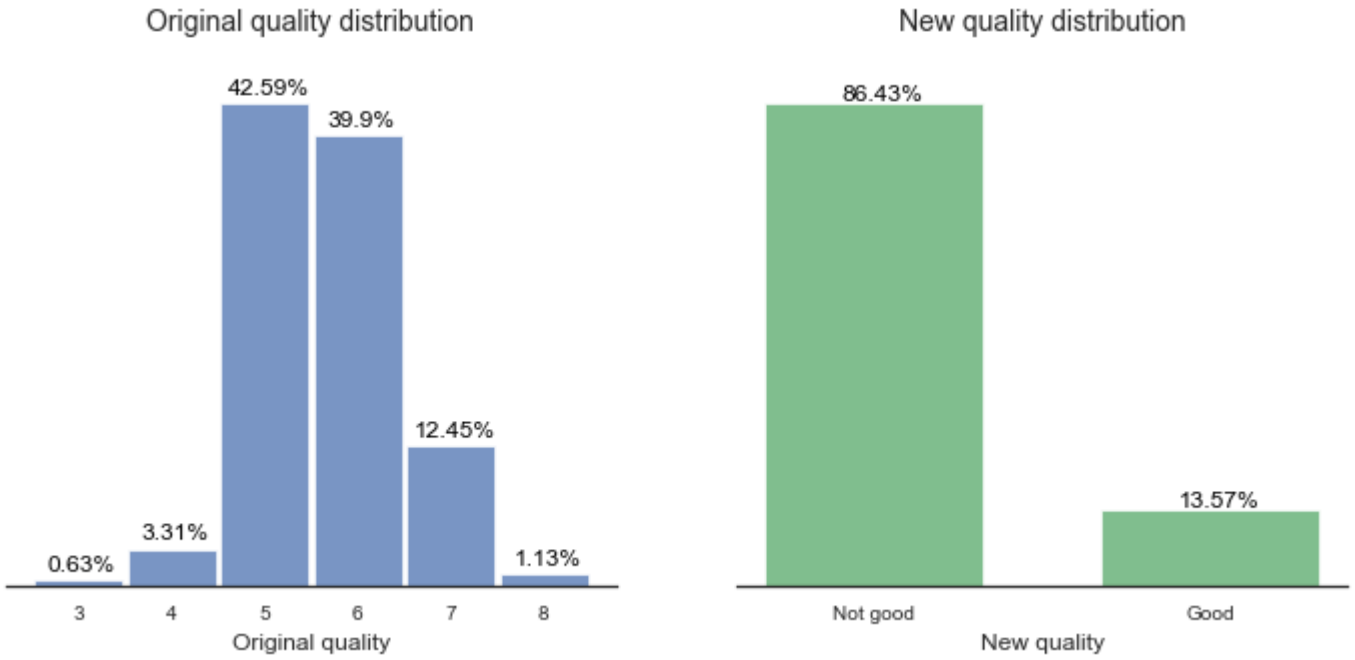
- The *Quality* feature has a mean of ***0.135***, indicating a high percentage of the records are labeled *0* (not good wines). At least 3/4 of the records are labeled as bad, as the 3rd quartile value (*75%*) is still 0.

- Comparing the values of the 3rd quartile (*75%*) and the maximum (*max*) value of the features, we see that in some features (*Residual sugar*, *Chlorides*, *Free so2* and *Total so2*) we will have to handle possible ***outliers***.

- In fact, *Residual sugar* and *Chlorides* have ***9.7%*** and ***6.45%*** respectively, of upper-range outliers, while *Free so2* and *Total so2* don't exceed ***4%***. It's worth noting, that there is no significant number of lower-range outliers.

Target feature distribution

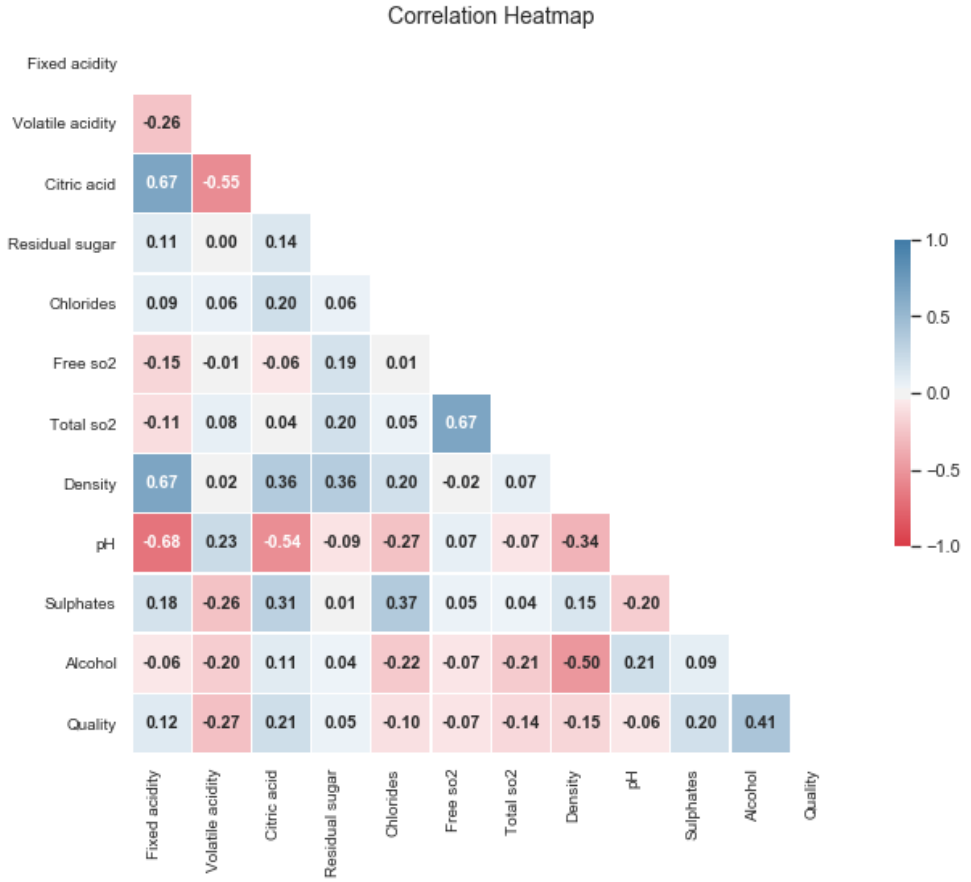
As expected, we are dealing with an *imbalanced classification problem*, having ***86.43%*** of the records are labeled as **not good** (*0*), implying a ratio of 1:6 approximately.



It is important to have in mind, that almost 40% of the records in the original quality distribution belong to the label **6**, which is the threshold in the new quality distribution between labels.

Correlations

The target value isn't strongly correlated to any of the features, being *Alcohol* the highest (*0.41*), but there are strong correlations between some of them, especially, *Fixed acidity* is strongly correlated with other 3 features. And another strong and *logic* correlation is between the total and free *CO2*.



Fixed acidity and *pH* **-0.68**
Fixed acidity and *Citric acid* **0.67**
Fixed acidity and *Density* **0.67**
Total co2 and *Free co2* **0.67**

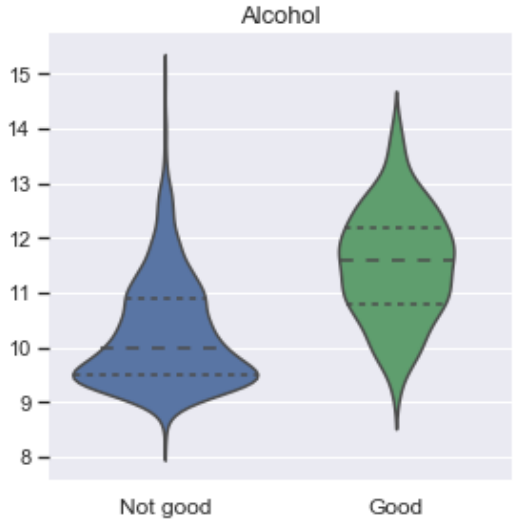
4. Exploratory data analysis

Features statistics per target label

When we calculating the principal statistics (mean and STD) of the features by target label, we get interesting insights on some features:

- ***Fixed acidity:*** *Good* wines have a **higher mean** value than the *not good* ones, but its data points are more scattered.
- ***Volatile acidity:*** *Good* wines have a **lower mean** value than the *not good* ones, and its data points are closer to each other.
- ***Sulphates:*** *Good* wines have a **higher mean** value than the *not good* ones, and its data points are closer to each other.
- ***Alcohol:*** *Good* wines have a **higher mean** value than the *not good* ones, but its data points are more scattered.
- ***Citric acid:*** *Good* wines have a **higher mean** value than the *not good* ones, and are slightly more scattered.

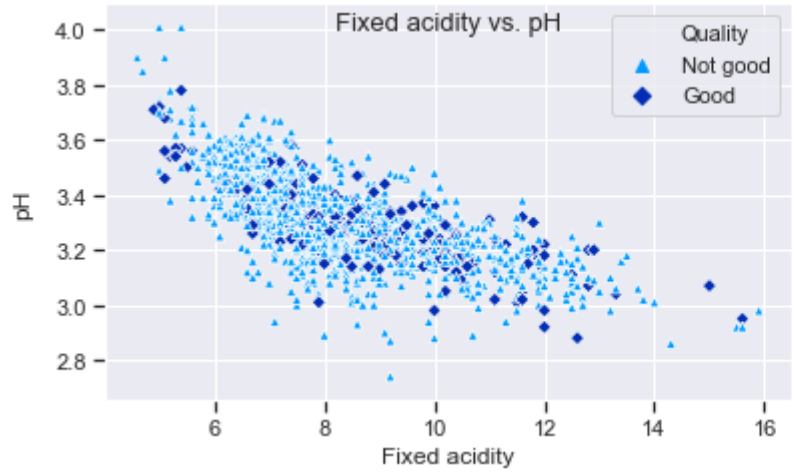
Here, you can see the KDE and Violin plots for the *Alcohol* feature, all the other plots can be seen in the notebook.

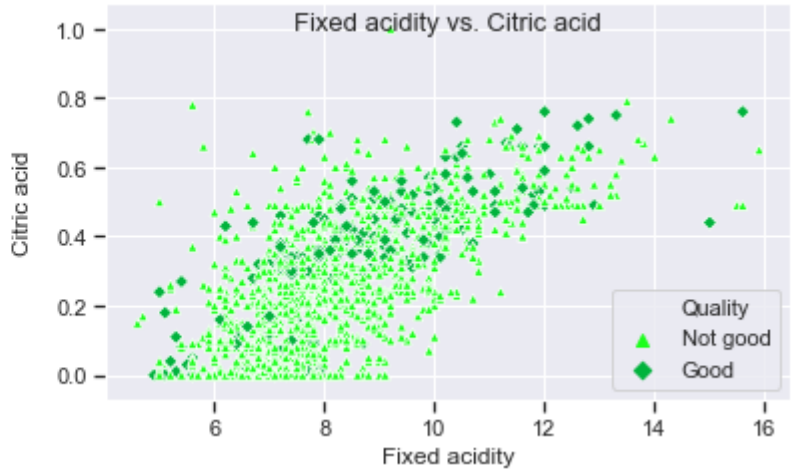


Correlated features analysis

In general, the correlation of the target feature with the rest of the values is weak, except for Alcohol (0.41). The feature Fixed acidity is highly correlated with other 3 features (pH, Citric acid and Density), and all present a linear trend, especially strong in Density, but in a very small range of values (Density varies from 0.99 to 1.01).

The other 2 correlated features are Total so2 and Free so2, that present a strong trend when their values are small but is lost as both values grow. In the notebook, 4 scatter plots graph for the four cases, in all cases, we don't see any clear clusters of data points by Quality label. All the other plots can be seen in the notebook.





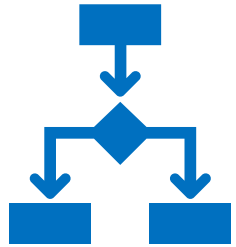
5. Data pre-processing

The data is split into a 7:3 train/test ratio. We'll fit the models with the train set and leave the test set for the last evaluation.

We've normalized the data with the **Z Normalization** technique. We created new datasets (normalized train/test) and kept the originals, to verify whether better results are obtained with the original data or the standardized data.

Being this an imbalanced classification problem, we had to approach this issue to ensure the models be fitted with the best data possible.

With a small dataset (1599 rows), downsampling would leave us with very few records, which would not guarantee the model would work properly. We've upsampled the train data with SMOTE (Synthetic Minority Oversampling Technique).



6. Model development

Models selection and baselines

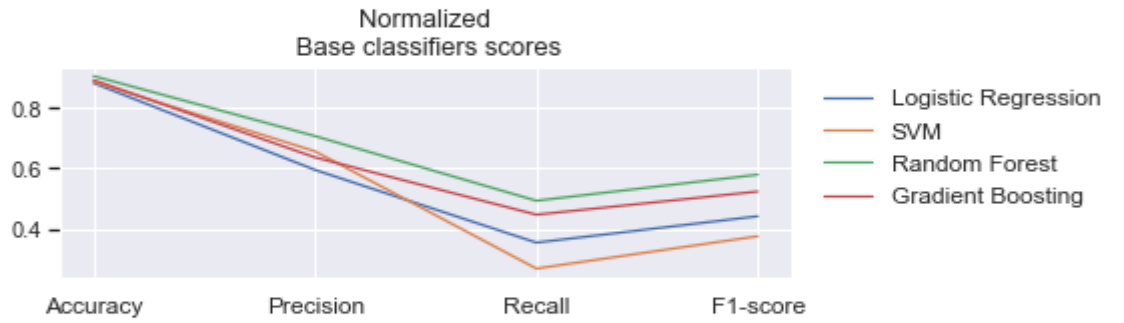
For this classification task, 4 different machine learning algorithms were chosen, two *linear* and two *ensembles*, to see which performs better for the problem:

- ***Logistic Regression***
- ***Support Vector Machine (SVM)***
- ***Random Forest***
- ***Extreme Gradient Boosting (XGBoost)***

To begin, we fitted the algorithms on their default state, for each of the 3 different datasets created (original, normalized and normalized-oversampled) to check its performance, then to select the one with best results for all the models, and have a baseline to compare.

To measure the models' effectiveness, in this first step we've used 4 of the most popular classification scoring methods. *Accuracy, Precision, Recall* and *F1*. For the following steps, we'll focus on the **F1 score**, being this an imbalanced classification problem, this score will give us the harmonic mean between Precision and Recall.

By far, the *normalized-oversampled* dataset had the best performance on all 4 models, the best the Random Forest model, the worst Logistic Regression.



Model tuning

To correctly tune the models, a Grid Search Cross-validation was made, with a selected range of the most important hyperparameters of each algorithm, doing a 5 folder cross-validation to avoid overfitting. The hyperparameters with the best F1 scores were:

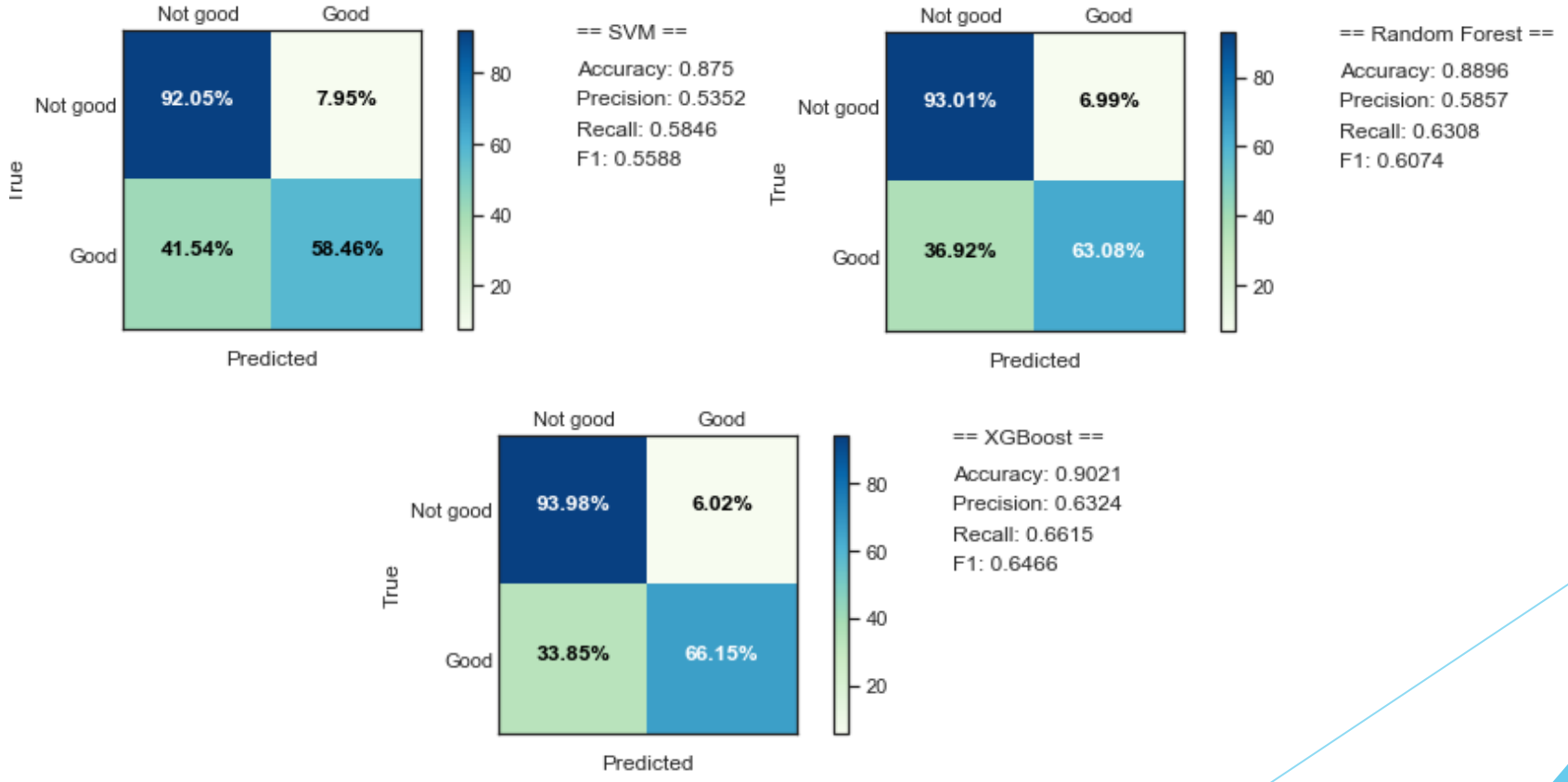
Classifier	Parameters	F1 score
SVM	C = 1.5 - gamma = 1	0.9694
XGBoost	gamma = 0.005 - learning_rate = 0.25 - max_depth = 6 - max_features = [0.6, 0.7, 0.8]	0.9441
Random Forest	max_depth = 14 - max_features = 0.4 - min_samples_leaf = 2	0.9354
Logistic Regression	C = 0.01	0.8393

At this point, we dropped the *Logistic Regression* model (lowest F1 score) and continued with the other three. After a more careful tuning of the values obtained for the hyperparameters (shown above), the best hyperparameters for the 3 models are:

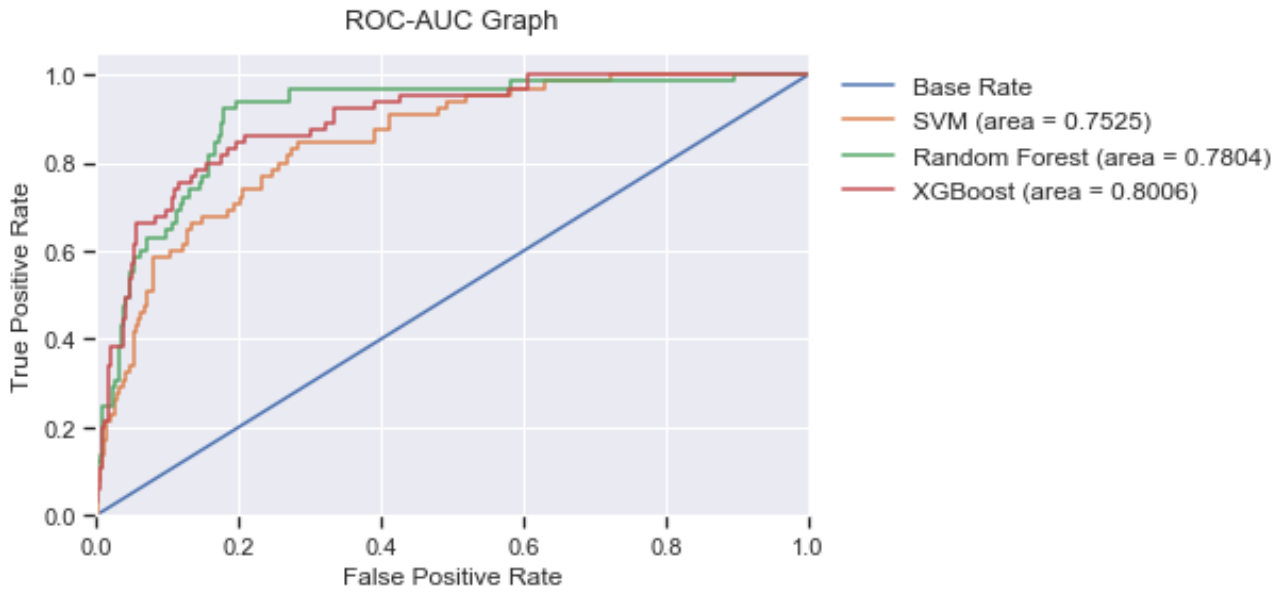
Classifier	Parameters
SVM	C = 1.35 - gamma = 0.2
XGBoost	gamma = 0.02 - learning_rate = 0.25 - max_depth = 6 - max_features = 0.7
Random Forest	max_depth = 10 - max_features = 0.4 - min_samples_leaf = 2

Model validation

We've made the predictions with the validation data, and evaluated the performance with a confusion matrix and the scoring methods used at the beginning. The higher percentage of correct predictions for the minority class (***66.15%***) and the best F1 score (***0.6466***) was obtained by the *XGBoost* model. Here we can see the results in the *Confusion matrices.*

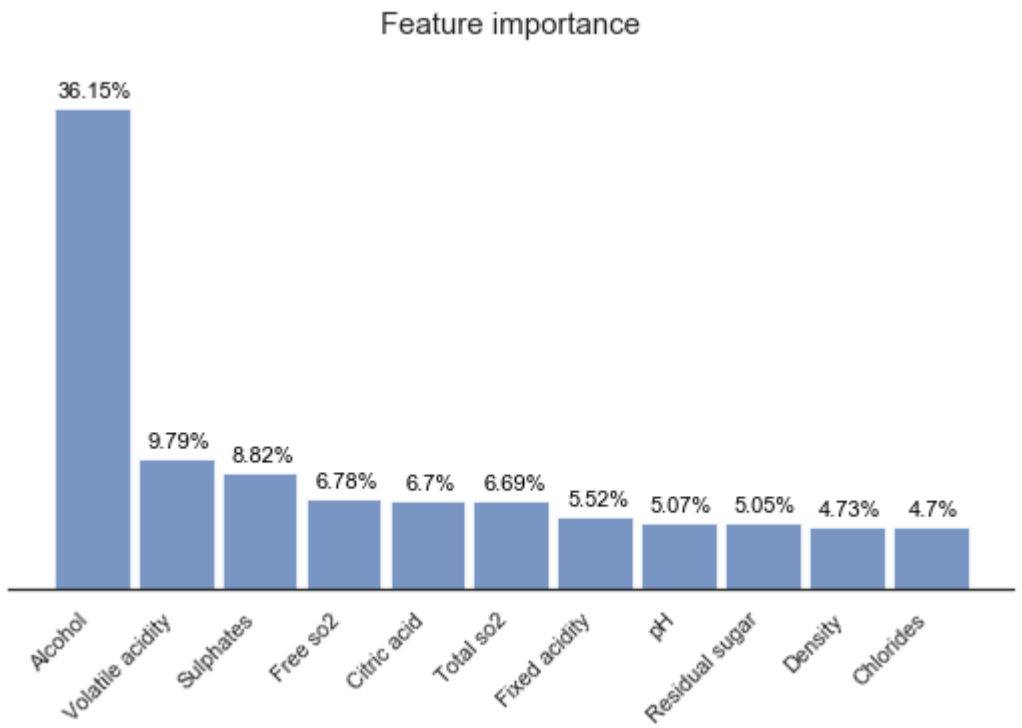


Here, we can see the *ROC-AUC curve* of all classifiers.



Feature importance and predictions

Finally, the predictions for the unseen wine records were made and exported into a CSV file. And to end, the feature importance generated by the model are:



7. Conclusion and recommendations

Conclusion

Having the final results, we can confirm that the best model that can predict very accurately if a wine is *not good*, but doesn't repeat performance on the *good* wines. If we remember, the original distribution of the target feature, are levels from 3 through 8; the threshold was between labels **6** and **7**. Now, the records labeled **6** represented almost 40% of the records, against 12% of those labeled **7**.

The quality labels in the dataset are the mean of the votes (minimum of 3) of the sommeliers, meaning that a wine with votes *6, 7, 7, 6, 6*, will be labeled as quality **6**, which lead us to the conclusion that the labeling process, being an average of scores based on sensory appreciation, there's no clear definition of a threshold between labels.

Besides, remember the 240 duplicates we've found, what happens is that a record labeled *6 (not good)* in the dataset, can have a *twin* record but labeled as *7 (good)*, mainly because one sommelier vote declined the balance one way or the other.

Final recommendations

- Perhaps, with more data, the models will be able to predict more accurately. As example, information about the exact vineyard location, the grape varieties used, the exact time of fermentation, etc.

- Standardize the labeling process, having a minimum number of votes, and make the sommelier group consistent in time, as the sensory appreciations are different from one person to another, having a *closed group* will allow the thresholds to be more consistent.

- Of course, the balance between the profit than can be generated with the model in production, should be contrasted against the cost of gathering either the extra data and the sommeliers' group.

With no clear forecasted benefits, the process should remain the same as it has been to avoid losses.