Quality white wine predictor

Contents

Summary

This report uses the white wine database from "vinho verde" to predict the quality based on physicochemical properties. Quality is a subjective measure, given by the average grade of three experts.

Before starting the predictions, the report performs an exploratory data analysis (EDA) to look for features that may provide good prediction results, and also makes an short explanation about the metrics used in the models. In data preparation, the database are downloaded and processed in python. In this phase, the training and testing sets are created and they will be used during the model building.

There's a brief explanation of the models used in this report. Other important machine learning concepts, such as ensemble and cross validation, are also discussed.

The results section presents the best model for predicting quality and discuss why it was chosen for this purpose.

Goal

This project aims to determine a model to predict wine quality given measurable wine features.

Introduction

According to experts, wine is differentiated according to its smell, flavour, and colour, but most people are not wine experts to say that wine is good or bad. The quality of the wine is determined by many variables including, but not limited to, the ones mentioned previously. The quality of a wine is important for the consumers as well as the wine industry. For instance, industry players are using product quality certifications to promote their products. However, this is a time-consuming process and requires the assessment given by human experts, which makes this process very expensive. Nowadays, machine learning models are important tools to replace human tasks and, in this case, a good wine quality prediction can be very useful in the certification phase. For example, an automatic predictive system can be integrated into a decision support system, helping the speed and quality of the performance.

Data

The wine quality dataset is publicly available on the UCI machine learning repository (check the links below). The dataset has two files, red wine and white wine variants of the Portuguese "Vinho Verde" wine. It contains a large collection of datasets that have been used for the machine learning community. The red wine dataset contains 1599 instances and the white wine dataset contains 4898 instances. Both files contain 11 input features and 1 output feature. Input features are based on the physicochemical tests and output variable based on sensory data is scaled in 11 quality classes from 0 to 10 (0-very bad to 10-very good)

- UCI repository
- White wine database

Input variables:

1. **Alcohol**: the amount of alcohol in wine
2. **Volatile acidity**: are high acetic acid in wine which leads to an unpleasant vinegar taste
3. **Sulphates**: a wine additive that contributes to SO2 levels and acts as an antimicrobial and antioxidant
4. **Citric Acid**: acts as a preservative to increase acidity (small quantities add freshness and flavor to wines)
5. **Total Sulfur Dioxide**: is the amount of free + bound forms of SO2
6. **Density**: sweeter wines have a higher density
7. **Chlorides**: the amount of salt in the wine
8. **Fixed acidity**: are non-volatile acids that do not evaporate readily
9. **pH**: the level of acidity
10. **Free Sulfur Dioxide**: it prevents microbial growth and the oxidation of wine
11. **Residual sugar**: is the amount of sugar remaining after fermentation stops. The key is to have a perfect balance between — sweetness and sourness (wines > 45g/ltrs are sweet)

Methods

How to analyze the data Our task here is to focus on what white wine features are important to get the promising result. For the purpose of regression model and evaluation of the relevant features, we are using the following algorithms to perform this task:

1. **DummyRegressor**: is one that takes only the value 0 or 1 to indicate the absence or presence of some categorical effect that may be expected to shift the outcome.
2. **Ridge**: is a model tuning method that is used to analyse any data that suffers from multicollinearity. This method performs L2 regularization. When the issue of multicollinearity occurs, least-squares are unbiased, and variances are large, this results in predicted values to be far away from the actual values.
3. **Random Forest**: is an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees. For regression tasks, the mean or average prediction of the individual trees is returned
4. **KNN**: is an approach to data classification that estimates how likely a data point is to be a member of one group or the other depending on what group the data points nearest to it are in.The k-nearest-neighbor is an example of a "lazy learner" algorithm, meaning that it does not build a model using the training set until a query of the data set is performed.
5. **Bayes**: is an algorithm that uses Bayes' theorem to classify objects. Naive Bayes classifiers assume strong, or naive, independence between attributes of data points. Popular uses of naive Bayes classifiers include spam filters, text analysis and medical diagnosis.

6. **SVM**: is machine learning algorithm that analyzes data for classification and regression analysis. SVM is a supervised learning method that looks at data and sorts it into one of two categories. An SVM outputs a map of the sorted data with the margins between the two as far apart as possible.

Analysis

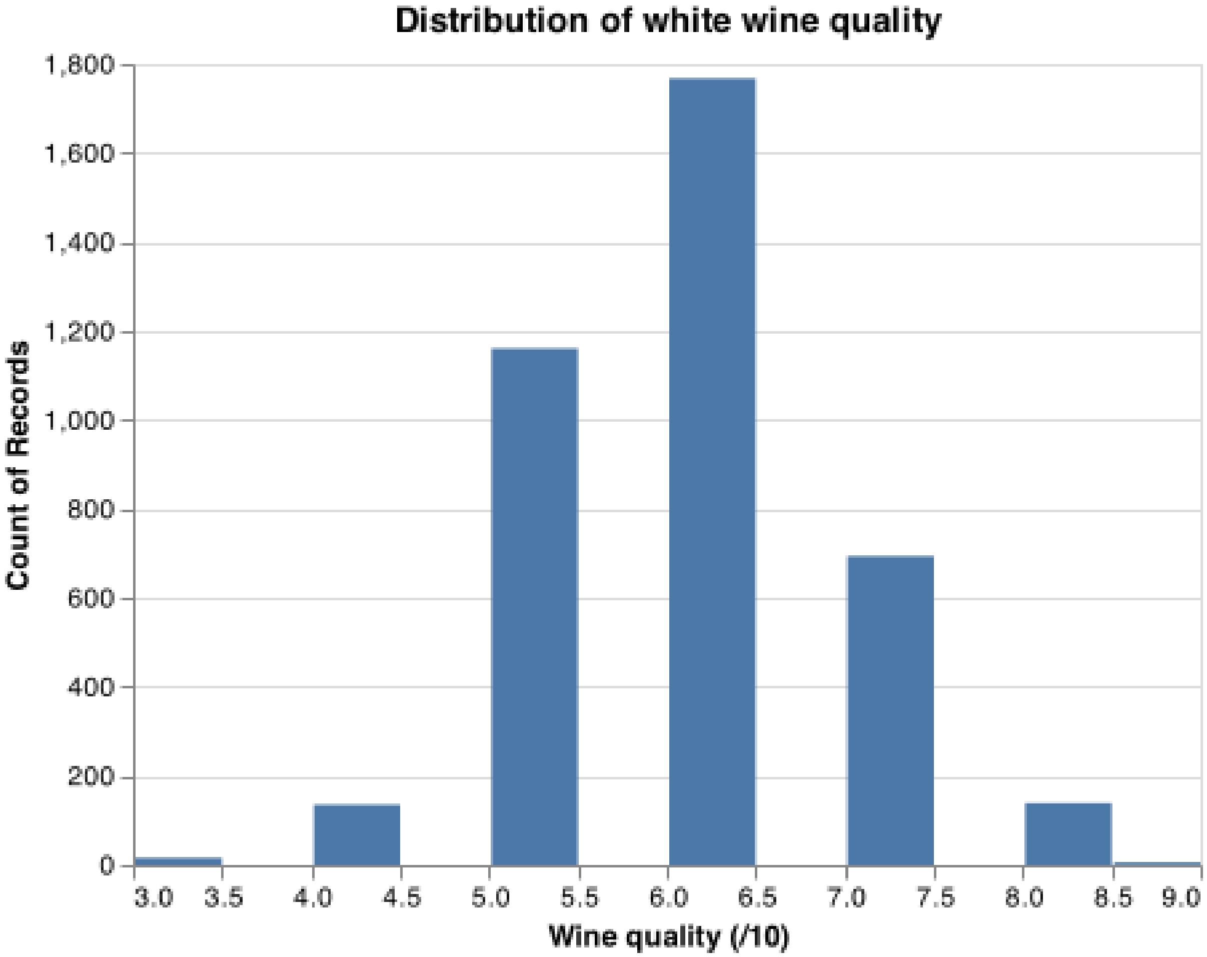
Data Cleaning and Preprocessing

The first step clean and prepare the data for analysis. First, It is necessary to checked the data types focusing on numerical and categorical to simplify the correlation's computation and visualization. Second, it's necessary to identify any missing values existing in our data set. Last, it's relevant to research each column/feature's statistical summary to detect any problem like outliers and abnormal distributions.

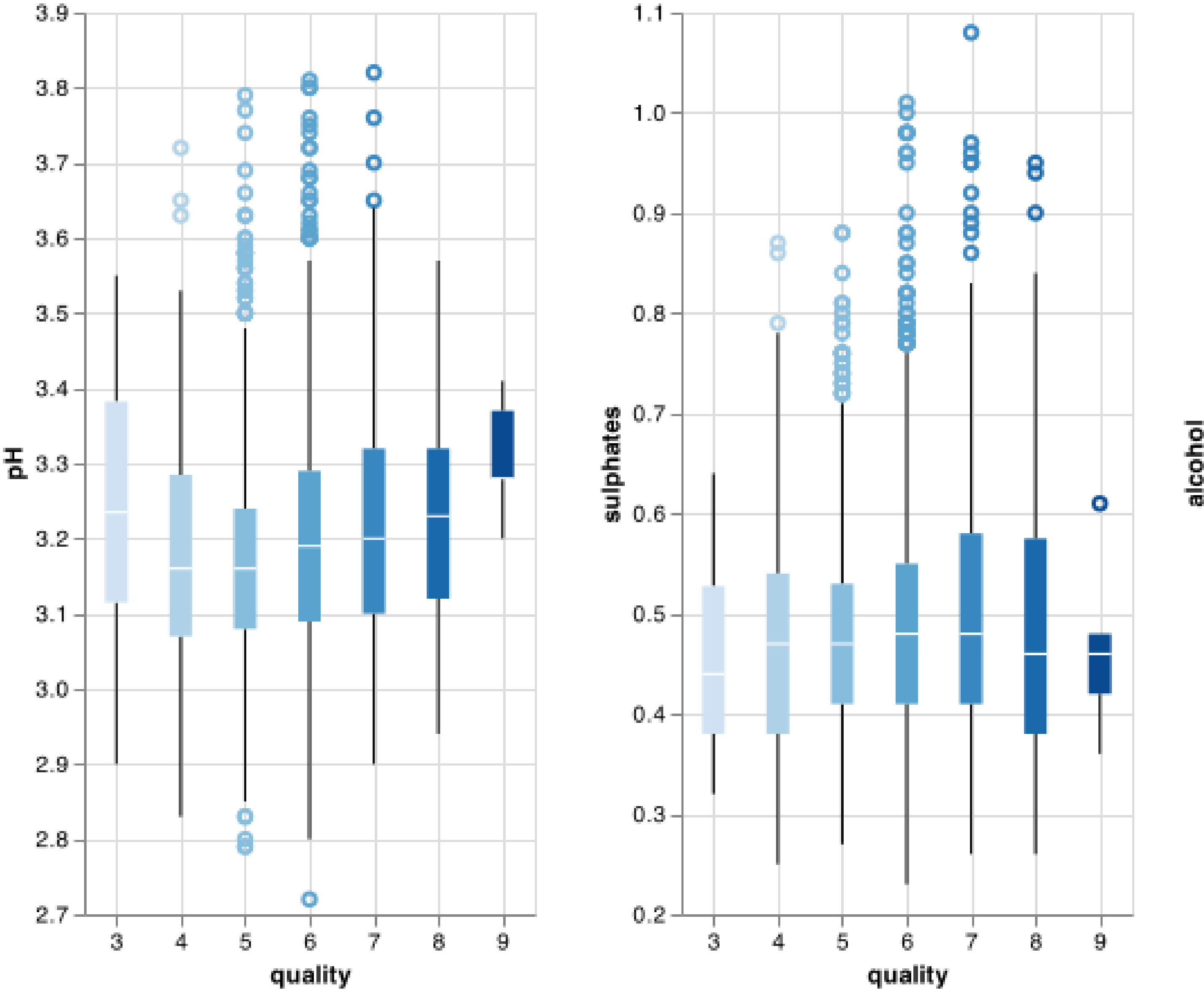
After, data preprocessing is crucial in any data mining process as they directly impact success rate of the project. This reduces complexity of the data under analysis as data in real world is unclean. we split the dataset in two parts, one for training and one for testing. The model building and tuning is done in the training set, and then we use the test set to predict new values and evaluate the results.

EDA first conclusions

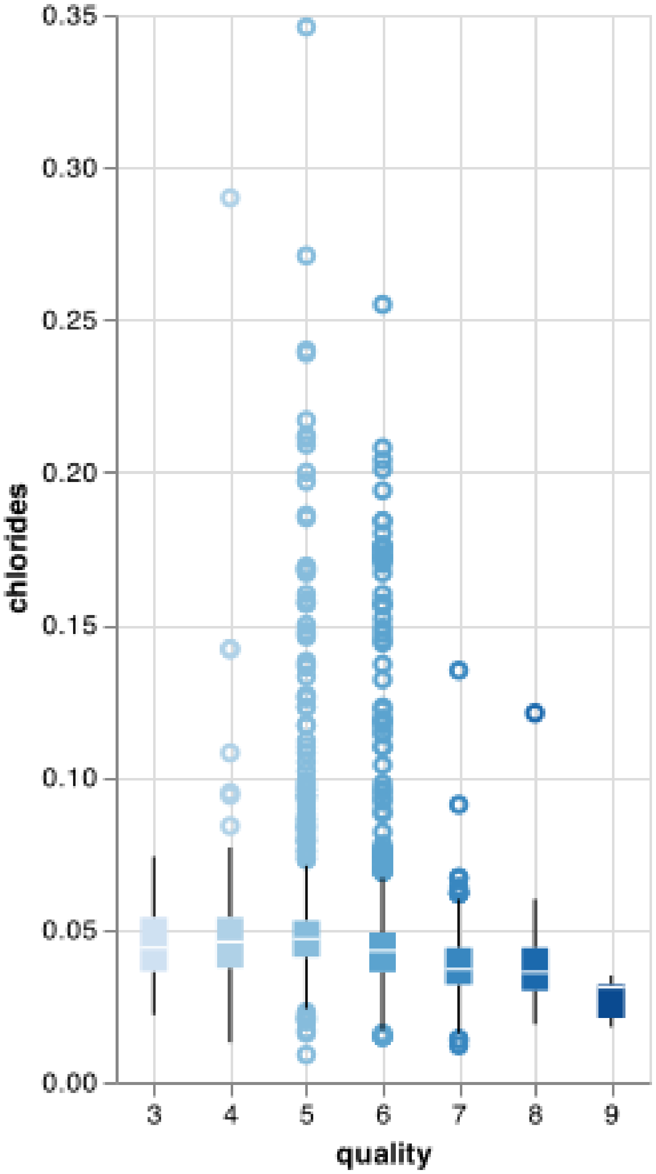
According to our first EDA, we do not have a balanced database, our wines are concentrated around quality 5 and 7.5 (around 80% of data points). The image below can clear shows up this first finding:

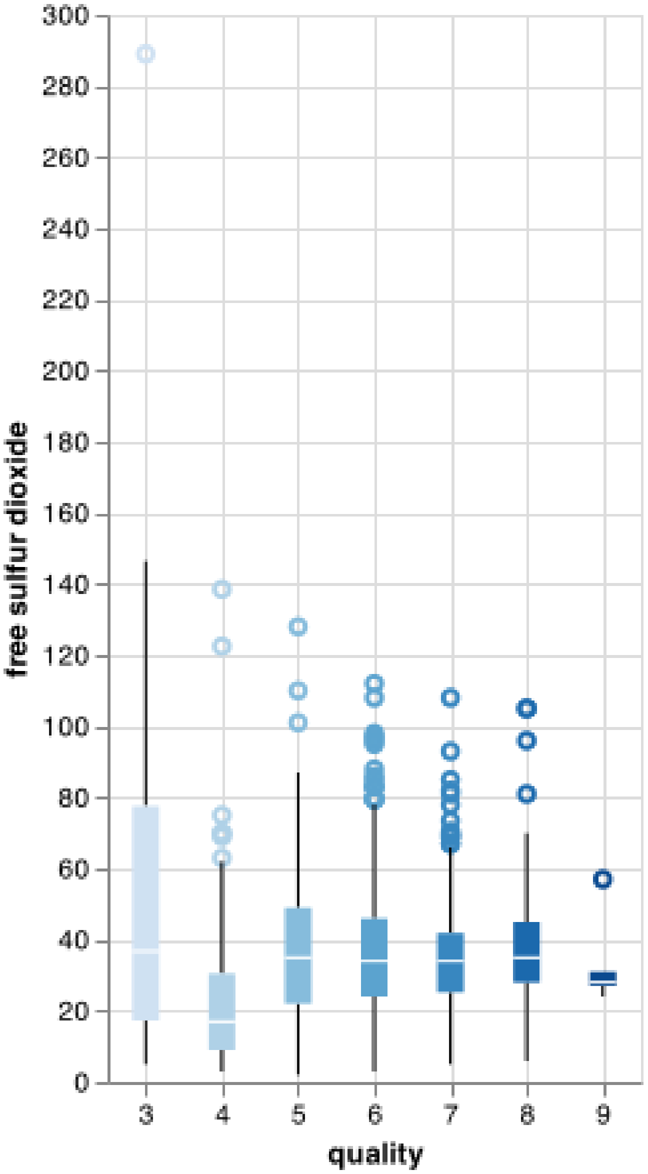


Besides, we have a couple of signs about some variables. For instance, it appears that the higher the alcohol, Sulphates and ph levels the better the wine quality. We can check these information on the chart below:

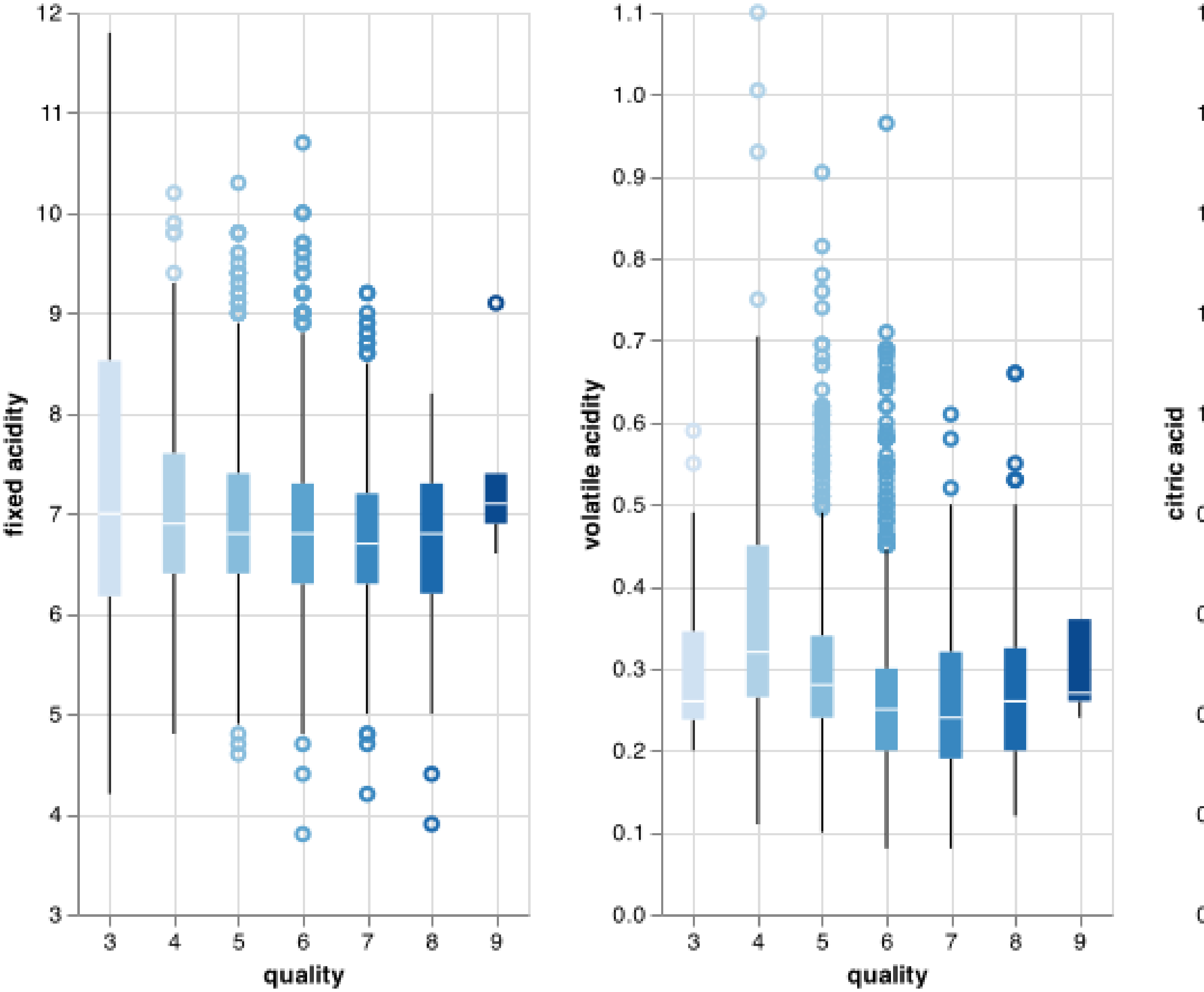


Following the same idea, we there are some evidences that the smaller the chlorides, Free Sulfur Dioxide, total sulphur dioxide and density levels, the better the wine quality. The chart below ilustrates these findings:





For some variables like Fixed acidity, Volatile acidity, Citric Acid and Residual sugar seem do not influence wine quality on their own. However, we need to take care with this statement since when they are combining with other variables above, they might indeed influence wine quality. The chart below can show up these findings:



Cross Validation

In order to evaluate the best model, we will use the cross validation approach to support our decision. The original dataset is partitioned in the training set used to train the model, and the test set used to predict the values with the trained model. Cross validation partitions the training set in the same way and performs the training and prediction several times. Then, the result with the best R^2 will be selected.

Hyperparameter tuning

In this analysis, we also performed the Hyperparameter optimization since the same kind of machine learning model can require different constraints, weights or learning rates to generalize different data patterns. Hyperparameter optimization finds a tuple of hyperparameters that yields an optimal model which minimizes a predefined loss function on given independent data.

Packages

A relevant point to be mentioned is what libraries are we using in this analysis. We are using (Pedregosa et al. 2011), (Van Rossum and Drake 2009), (team 2020), (Xie, n.d.), (Harris et al. 2020) and (Virtanen et al. 2020), (Römer and Kraska 2007), (Sievert 2018) and (Joblib Development Team 2020)

Results & Discussion

Evaluation

After running the models, we used the test-score metrics to evaluate our model prediction performance. As we can assess into the table below, model KNN have a $R^2 = 0.54$ which is the best value in comparison with the other models

Table 1: Table 1:

model	r2_score	mse_score	rmae_score	mae_score	mse_log_score	mae_log_score
KNN	0.5453724	0.3537442	0.594764	0.3923278	0.0081026	0.2672034

In the context of our business question focusing on the prediction of white wine quality, it is reasonable that KNN gives us superior "predictions."

Discussion

By analyzing the physicochemical tests samples data of white wines from the north of Portugal, we were able to create a model that can help industry producers, distributors, and sellers predict the quality of white wine products and have a better understanding of each critical and up-to-date features. the KNN Model performed better than others.

It is relevant to mention that there are some limitations for this analysis. First, the main problem came from the fact that our data set was unbalanced. A majority of the quality values were around 5 and 6, which made no significant contribution to finding an optimal model. These values might made it harder to identify each factor's different influence on a "high" or "low" quality of the wine, which was the main focus of this analysis. In order to improve our predictive model, we need more balanced data.

Another limitation is that we have only 12 attributes that can narrow down the accuracy of our predicting quality of white wine. The solution for this is to include more relevant data features, like the year of harvest, brew time, etc. So that our R^2 (0.54) could be improved, since the value that we obtained is not close to 0.7 or 0.8. Those are considered, as a rule of thumb, relevant R^2 values for a regression.

References

Harris, Charles R., K. Jarrod Millman, Stéfan J van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, et al. 2020. "Array Programming with NumPy." *Nature* 585: 357–62. https://doi.org/10.1038/s41586-020-2649-2.

Joblib Development Team. 2020. *Joblib: Running Python Functions as Pipeline Jobs.* https://joblib.readthedocs.io/.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, et al. 2011. "Scikit-Learn: Machine Learning in Python." *Journal of Machine Learning Research* 12: 2825–30.

Römer, F, and T Kraska. 2007. "Homogeneous Nucleation and Growth in Supersaturated Zinc Vapor Investigated by Molecular Dynamics Simulation." *The Journal of Chemical Physics* 127 (23): 234509.

Sievert, Jacob VanderPlas AND Brian E. Granger AND Jeffrey Heer AND Dominik Moritz AND Kanit Wongsuphasawat AND Arvind Satyanarayan AND Eitan Lees AND Ilia Timofeev AND Ben Welsh AND Scott. 2018. "Altair: Interactive Statistical Visualizations for Python." *The Journal of Open Source Software* 3 (32). http://idl.cs.washington.edu/papers/altair.

team, The pandas development. 2020. *Pandas-Dev/Pandas: Pandas* (version latest). Zenodo. https://doi.org/10.5281/zenodo.3509134.

Van Rossum, Guido, and Fred L. Drake. 2009. *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace.

Virtanen, Pauli, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, et al. 2020. "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python." *Nature Methods* 17: 261–72. https://doi.org/10.1038/s41592-019-0686-2.

Xie, Yihui. n.d. *Knitr: A General-Purpose Package for Dynamic Report Generation in r*. https://yihui.org/knitr/.