Analysis on White Wine Quality Dataset

Final Report

Qi Zhao

DATA-1030 Fall 2022

Github: https://github.com/kevinz8866/1030project.gits

DSI Brown University

12.02.2022

Introduction

The dataset I will be working on is ”Wine Quality Data Set” from UCI Machine Learning Repositories. The original dataset contains a red wine dataset and a white wine dataset, and I choose to focus on the white wine dataset for this project. I am interested in this dataset because I had many experiences with wine and always had trouble distinguishing good wines from bad ones. There are a lot of people drinking wine everyday so it is very useful to come up with a quantitative measure on wine quality.

This data set contains 4898 instances and 11 columns of explanatory variables and one column of target variables. The input features are continuous variables which represent numerical values from the lab results. The output variable is a numerical score based on the median of at least 3 evaluations from wine experts). Each expert graded the wine quality between 0 (very bad) and 10 (very excellent).

The target variable is "quality", which indicates the quality score of the predicted wine. Because quality score is an ordinal variable, this can be both a regression problem and a classification problem. There are order relations between labels but they are also discrete integers suitable for classification. For this project, I will first train my models using the regression metrics RMSE, by minimizing the loss between my output values and the true scores. Then on the best model, I will compute an accuracy score after rounding the model’s predictions to the nearest integers.

The author of this dataset adopted a regression approach. They used regression metrics to train their models. Then, model outputs are rounded to the closest integer and correlates with a class prediction. They found that SVM achieved the best results as compared to other models. When rounding to the nearest integer, the overall accuracy is 64.6% on white wine.

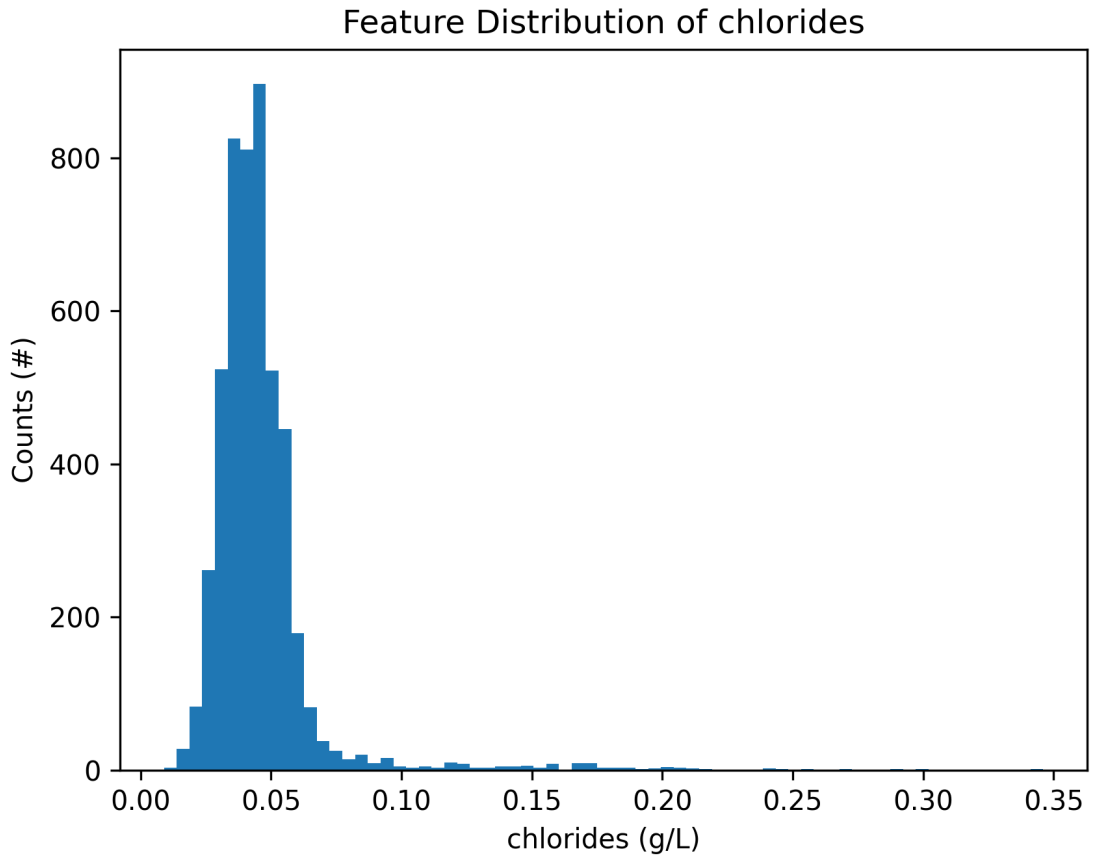
A 2021 study "Analysis of white wine using machine learning algorithms" performed similar research on the dataset using many models including SVM, Random Forests, and etc. Researchers reported regression metrics including root mean square error(RMSE) as well as classification metrics including accuracy. Their results are promising, in which many models reached accuracy around 0.999 and RMSE around. But they neither revealed any hyperparameter choices nor described their pipeline in detail, which makes it hard to replicate and to compare to.

Some other previous work includes "Assessing wine quality using a decision tree"

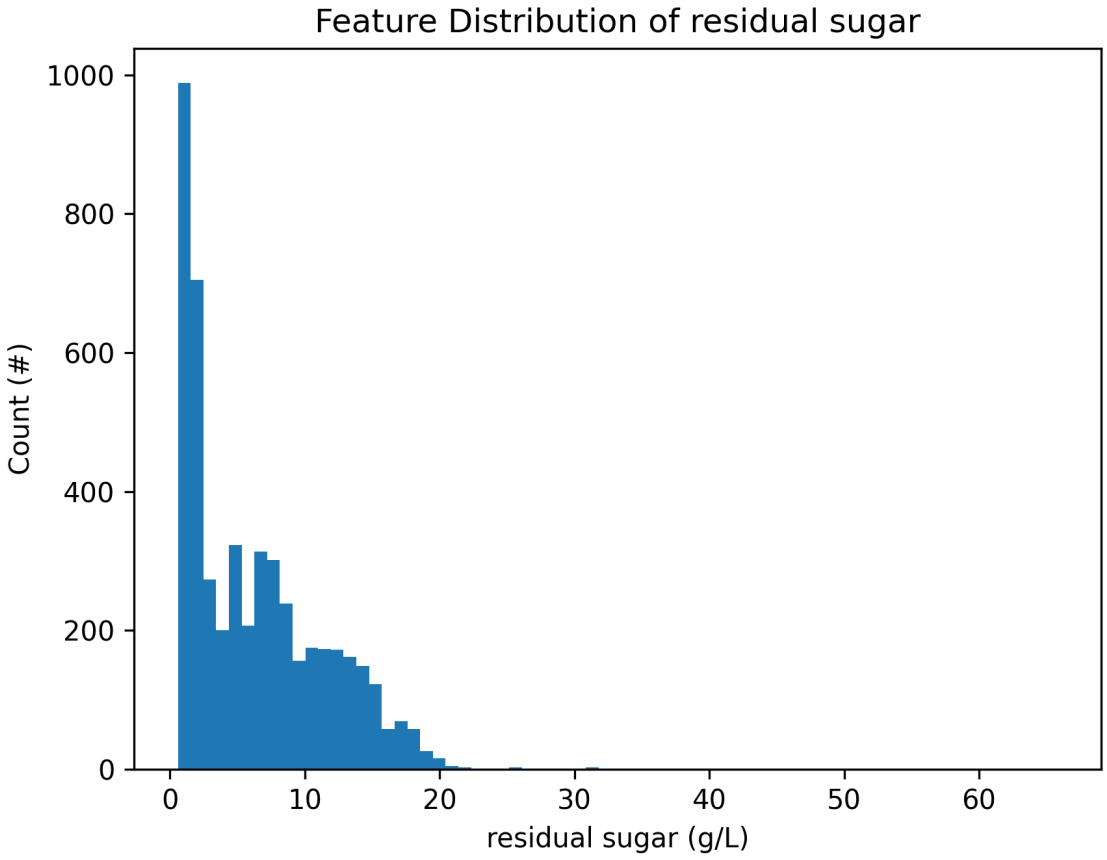
by Lee, Park and Kang. They used decision tree models and reached accuracy of 60.4%. In "Wine Quality Analysis Using Machine Learning Algorithms" by Gupta et al researchers reported a RMSE of 0.6430 for random forest alone and a combined algorithm of RF and KNN reached RMSE of 0.54.

Explantionary Data Analysis

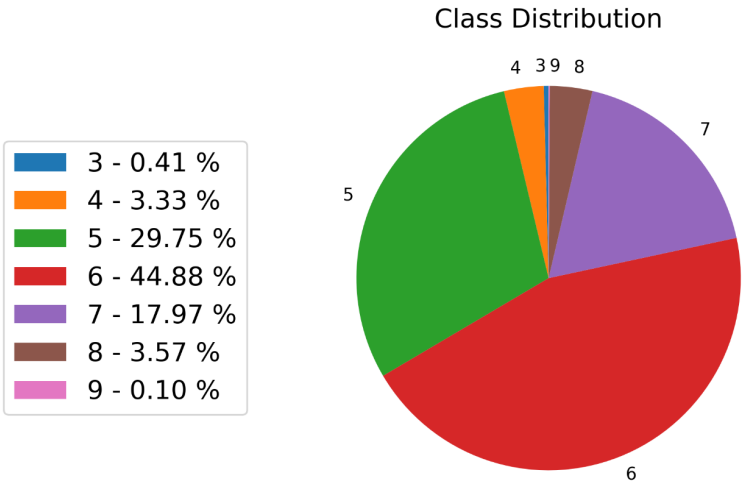
The data has 0% missing values. All of my input features are continuous variables, and I have studied their distribution, skewness and shape. Most of my input features are positive numbers, and nine of them are normally distributed with a non-zero mean with some outliers three std away from the mean. These include: fixed acidity, volatile acidity, citric acid, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, and sulfates. I will histogram to plot chlorides to show an example of these types of features.

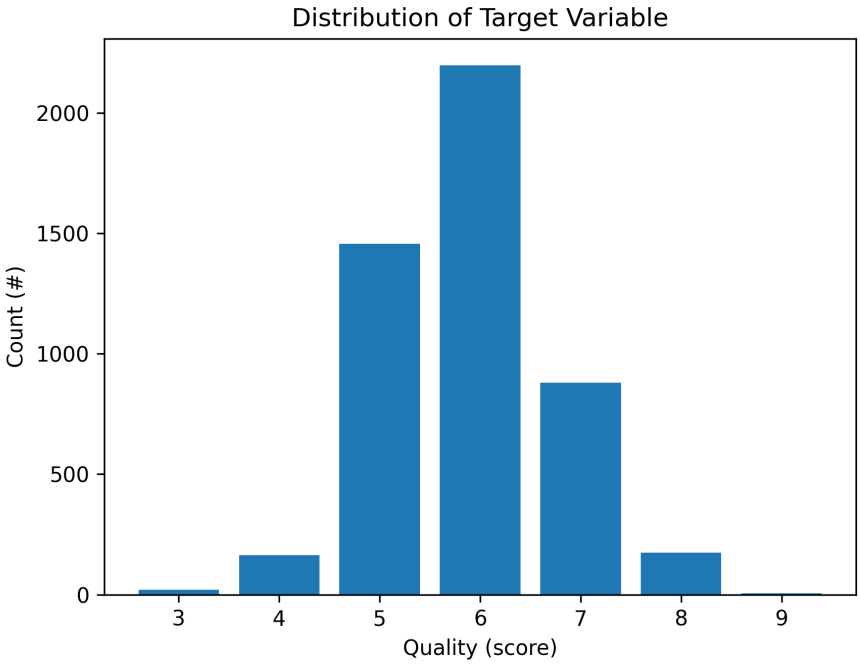


Two features have skewness and are not distributed normally. These include residual sugar and alcohol. I also use histogram to plot them. We can see that most of the data is concentrated near zero but there are also many on the right. The reason for these skewed data is partly because the lab testing results in a positive number. Many wine samples have very little residual sugar, but there are also sweet wines that contain a lot of sugar in the solution.

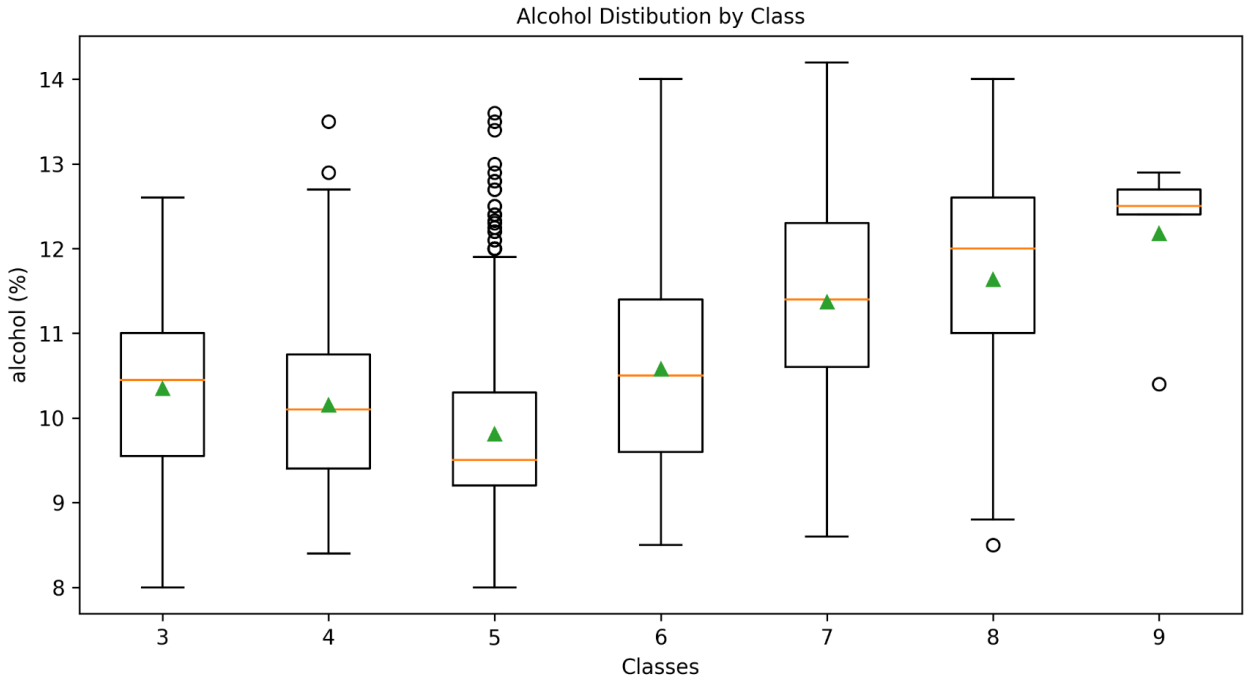


Our target variable is an ordinal variable so it has both regression features as well as class distribution. The distribution of our target variable is normally distributed with a very small standard deviation and no skewness. A quality score of 3 and 9 could be considered outliers in this distribution because they are three std away from the mean. The class distribution confirms that we have an imbalanced class distribution, score 6 is the majority class and 3,4,8,9 have very little presence as compared to 5, 6, 7.





After conducting F-statistics analysis, I found that the input feature alcohol has the largest correlation with the target variable. So I plot the distribution of input feature alcohol in each target variable class. Surprisingly, alcohol also has a non-linear correlation with the target variable as well. We can see that the mean of alcohol tends to increase with quality score after quality score is greater than 5, but decrease with quality score when quality score is smaller than 5.



Methods

My dataset is independent and identically distributed without group structure. So I will split my dataset into training, validation, and testing of a 60%-20%-20% ratio with stratification. The reason for this arrangement is that the class in the target variable is not balanced. For example, there are only 5 instances with target variable equals to 9. So I need the 60%-20%-20% ratio to ensure that there is at least 1 instance in all of training, validation, and testing with target variable equals to 9.

I use MinMaxScaler for all my input features because I am dealing with some skewed distribution and also some features have very large mean and standard deviation. Using MinMaxScaler, I could have all input features in the range of (0,1). After processing, I have 11 features in the input data.

Then, I will select my models based on five different random splits of the data. In this way, I can account for the variance occurring in splitting my data. For each different split, the first step is to run a grid search on the splitted and preprocessed data to find the best hyperparameters for the model, using the training and validation dataset. For non-deterministic models like random forest, I will also run different initialization to account for model randomness. I will select my model from four algorithms: linear regression, support vector machine, random forest, k-nearest neighbor, and xgboost. This decision is motivated by the fact that these algorithms tend to perform well for other regression problems and they are frequently used and recommended in other papers on this topic as well.

The hyperparameters searched for different models are as follows. For linear regression, I searched optimal parameter choices for alpha. For support vector machine, l searched optimal parameter choices for kernel and degree. For random forest, l searched optimal parameter choices for n_estimators, max_depth, and max_features. For k-nearest neighbors, l searched optimal parameter choices for n_neighbors, and weights. For xgboost, l searched optimal parameter choices for n_estimators, and max_depth.

Then, for the best performing model on training and validation, I will run the test sets on it and take the average of the testing scores to compare it with other models. Since in this problem our target variable is an ordinal variable, the quality scores on each wine sample, I will train and evaluate my models using a regression approach and the selected metric is RMSE. This choice of metrics is motivated by the fact that RMSE is one of the most general metrics for regression problems, and it is reported in many

papers related to this dataset. After I trained and evaluated my models using the regression approach, I can also evaluate them from a classification point of view. After the model outputs predicted scores for each instance, I can round them to the nearest integer and check if it matches the correct score label. I can then calculate the accuracy of my model by dividing the number of correctly predicted instances over the total number of instances.

Results

The naive baseline RMSE of this problem is 0.886, which uses the mean of the target variable as my model prediction. The corresponding baseline accuracy is 0.4 when I round predictions to the nearest integer. The baseline RMSE of my model is 0.74, which comes from a linear regression model without any hyperparameter tuning. The corresponding baseline accuracy is 0.4 when I round predictions to the nearest integer.

The mean and standard deviation of the test scores of the best models is illustrated in the table below. Ridge is able to reach a mean RMSE of 0.745 with standard deviation of 0.011, which is 12.2 times standard deviation above the baseline. Support vector machine is able to reach a mean RMSE of 0.701 with standard deviation of 0.006, which is 27.75 times standard deviation above the baseline. Random forest is able to reach a mean RMSE of 0.617 with standard deviation of 0.007, which is 38.28 times standard deviation above the baseline. K-nearest neighbors is able to reach a mean RMSE of 0.645 with standard deviation of 0.012, which is 19.8 times standard deviation above the baseline. Xgboost is able to reach a mean RMSE of 0.671 with standard deviation of 0.008, which is 25.0 times standard deviation above the baseline.

Algorithm	Ridge	SVM	RF	KNN	Xgboost
Mean	0.745	0.701	0.617	0.645	0.671
Std	0.011	0.006	0.007	0.012	0.008

Random forest is the most predictive model for this problem. The optimal parameter choice of random forest is max_depth=100, max_features=0.25, n_estimators=1000, and other parameters are as default. It reached a RMSE of 0.607 and

accuracy of 66.6% after rounding its predictions to the nearest integers.

For this best model, I studied the global feature importance for all features by calculating their SHAP values. It turns out that alcohol is the most important feature, volatile acidity and density are the second and third. They have the largest impact on the model prediction. Sulfates is the least important feature relatively. To study the local feature importance, I picked 10 individual points from the test set and created their force plots. These points include representatives from the ones closer to the mean like 5,6,7, and from the ones far away from the mean like 3,4,8,9.

It turns out that on points closer to the mean and the outliers with low quality score, feature alcohol usually negatively contributes to the prediction. But for outliers with high quality scores, feature alcohol usually positively contributes to the prediction. This also echoes my findings in EDA where I pointed out that feature alcohol has a non-linear correlation with the target variable. Another interesting finding is that for wines that have a very low quality score, a very low value of free sulfur dioxide is usually present. This suggests that this feature might have a correlation with wines that have low quality scores.

Outlook

My best model shows satisfying performance above the baseline and compared to some other research. However, it is not perfect especially for that it does not do well on predicting the outliers. Since the data is highly imbalanced, the model tends to make predictions closer to the mean of the target variable. I could have set higher sample weights to those points that have a label far away from the mean, but doing so does not usually improve the overall performance of the model. If I set the sample weight too large, then the model sacrifices its performance on the instances whose labels are closer to the mean.

If I could collect more data on the wines that have either very high quality scores, or very low quality scores to make the dataset more balanced, then I think my model would have a better performance. Also, my model could also make better predictions if I could get more features on the wine samples, for example other metrics like color, sediment, etc. The reason is that wine samples that score 5,6,7 consist of more than 80% of the total samples, but my model could not reach a perfect regression on these points, which means there is some randomness among these points if we only look at the currently available features.

Word count: 1996

References

1. Cortez, Paulo. “ Wine Quality Data Set.” UCI Machine Learning Repository: Wine Quality Data Set, 7 Oct. 2009, https://archive.ics.uci.edu/ml/datasets/Wine+Quality.

2. P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis. Modeling wine preferences by data mining from physicochemical properties. In Decision Support Systems, Elsevier, 47(4):547-553, 2009.

3. Koranga, Manisha, et al. "Analysis of white wine using machine learning algorithms." Materials Today: Proceedings 46 (2021): 11087-11093.

4. Gupta, Ujjawal, et al. "Wine quality analysis using machine learning algorithms." Micro-Electronics and Telecommunication Engineering. Springer, Singapore, 2020. 11-18.

5. Lee, Seunghan, Juyoung Park, and Kyungtae Kang. "Assessing wine quality using a decision tree." 2015 IEEE International Symposium on Systems Engineering (ISSE). IEEE, 2015.