School of Engineering & Technology

Asian Institute of Technology

Computer Programming for Data Science and Artificial Intelligence

Date: <2023/11/25>

Project Proposal Group 6(Wine Quality Prediction)

Submitted To
Dr. Chantri Polprasert

Submitted By: (st123187, st123392)
Lakash Maharjan, Jiewen Shen

Contents

Introduction

Overview

Wine quality prediction is an interesting machine learning project in the domain of oenology. Evaluation of quality wine is very subjective; it is based on the expertise of sommeliers and wine enthusiasts. However, machine learning techniques can provide valuable insights and predictions based on quantitative data. This can be very useful for different wine makers, sommeliers, vineyards etc. We aim to build a model that leverages the power of data analysis and predictive modeling to assess and predict wine quality based on various wine attributes. Wine quality prediction combines technology and tradition to measure what makes a great wine. In a long-standing industry, giving consistent quality is one of the major challenges faced by the wine industry. Our goal in using machine learning is to provide data-driven insights to winemakers so they may improve their operations and provide customers with more information to make wise decisions.

Problem Statement

Ensuring the quality of wine production is a critical endeavor for any winery. Traditionally, the evaluation of wine quality occurs only after the production process is complete, leading to potential costly setbacks if expectations are not met. The subjectivity of individual tastes further complicates the quest for a standardized quality measure. In contemporary trends, the assessment of wine quality relies heavily on the subjective judgments of sommeliers and enthusiasts, resulting in varying opinions and challenges in maintaining consistent quality. The primary objective of our project is to develop a predictive model for a wine company or enthusiasts to assess the quality of wine based on various chemical components such as acidity, sugar content, pH value, and more. The key challenge lies in achieving high predictive accuracy while avoiding overfitting, ensuring that the model not only performs well on existing data but also generalizes effectively to new, previously unseen data. Striking this balance is essential to produce reliable and meaningful predictions that can aid in maintaining and enhancing wine quality standards.

Related Work

- **A Machine Learning Based Approach for Wine Quality Prediction:** Using standard datasets of Portuguese "Vinho Verde" wine, this research provides an automatic assessment of wine quality, classified as good or terrible, using machine learning techniques including neural networks, logistic regression, and support vector machines.

- **Prediction of Red Wine Quality Using One-dimensional Convolutional Neural Network:** The one-dimensional convolutional neural network (1D-CNN) model proposed in this study uses physicochemical parameters to predict red wine quality. With a 92.5% accuracy rate and outperforms other methods such as SVM, RF, KNN, DNN, and LR.

- **Wine Quality Prediction - Machine Learning3**: This article shows how to use various machine learning models such as XGBoost, SVM, and Logistic Regression to predict the quality of wine on the basis of given features. It also shows how to perform exploratory data analysis and data preprocessing on the wine quality dataset.

Methodology

Data Description

Data Source and Description

The data used in this project is open-source data downloaded from Kagel. This data is related to the chemical composition of red wine. There are total of 11 physiochemical properties They are as follow:

- Fixed acidity
- Volatile acidity
- Citric acid
- Residual sugar
- Chlorides
- Free sulfur dioxide
- Total sulfur dioxide
- Density
- pH
- Sulphates
- Alcohol

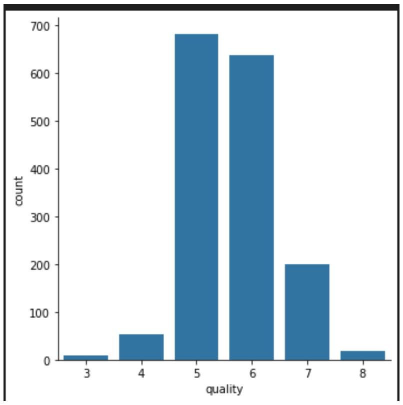
In addition to these characteristics, a sensory score was obtained from several blind taste testers, who assigned a number to each wine sample, 0 representing low quality and 10 representing great quality. There are 1600 records. The statistical analysis was done to understand the nature of the dataset.

Variable Name	Count	Mean	Standard Deviation	Min	max
Fixed acidity	1599	8.319637	1.47	4.6	15.9
Volatile acidity	1599	0.52	0.17	0.12	1.58
Citric Acid	1599	0.27	0.19	0	1.0
Residual sugar	1599	2.53	1.40	0.00	15.5
chlorides	1599	0.08	0.047	0.90	0.61
Free sulfur dioxide	1599	15.87	10.46	1.00	72.00
Total sulfur dioxide	1599	46.46	32.89	6.00	289.00
Density	1599	0.99	0.001887	0.99	1.00
Ph	1599	3.31	0.15	2.74	4.010
Sulphate	1599	0.65	0.16	0.33	2.00
Alcohol	1599	10.42	1.0	8.4	17.90

Exploratory Data Analysis (EDA)

Range of Quality

We were able to identify the range of wine quality using our dataset. Figure 1 shows the five categories into which wine is divided. It is evident from the bar plot in Figure that this dataset has many wines with quality ratings of 5 and 6. And relatively few wines are classified as being between grades 3 and 8.



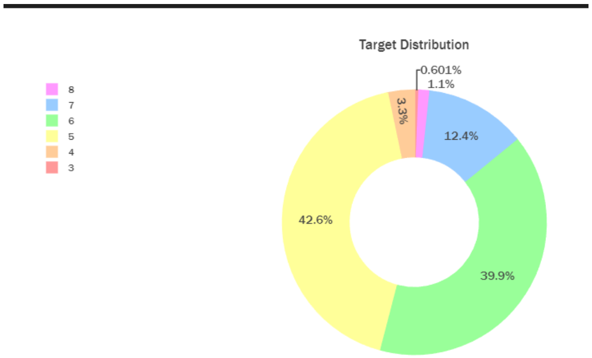


Figure 1 Range of wine

Data Distribution

In figure 2 we can see histogram to view the distribution of the data with all values in columns of dataset.

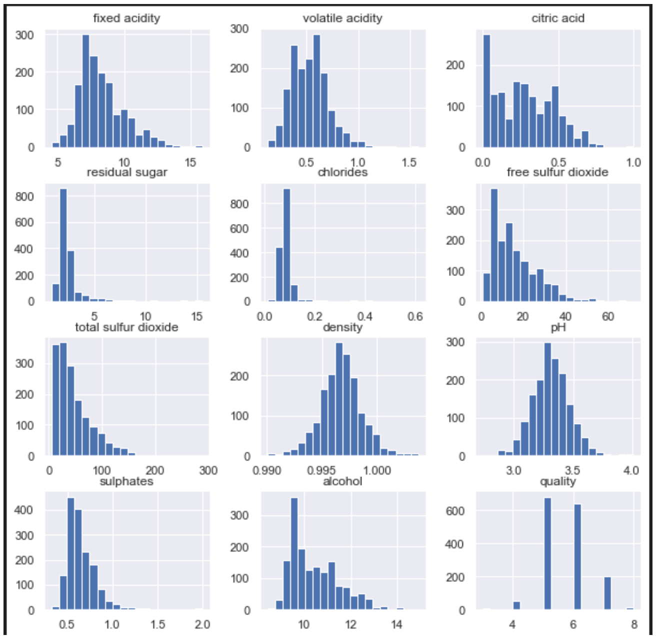


Figure 2 data distribution graph

Outliers Detection

In our dataset we saw there were lots of outliers.

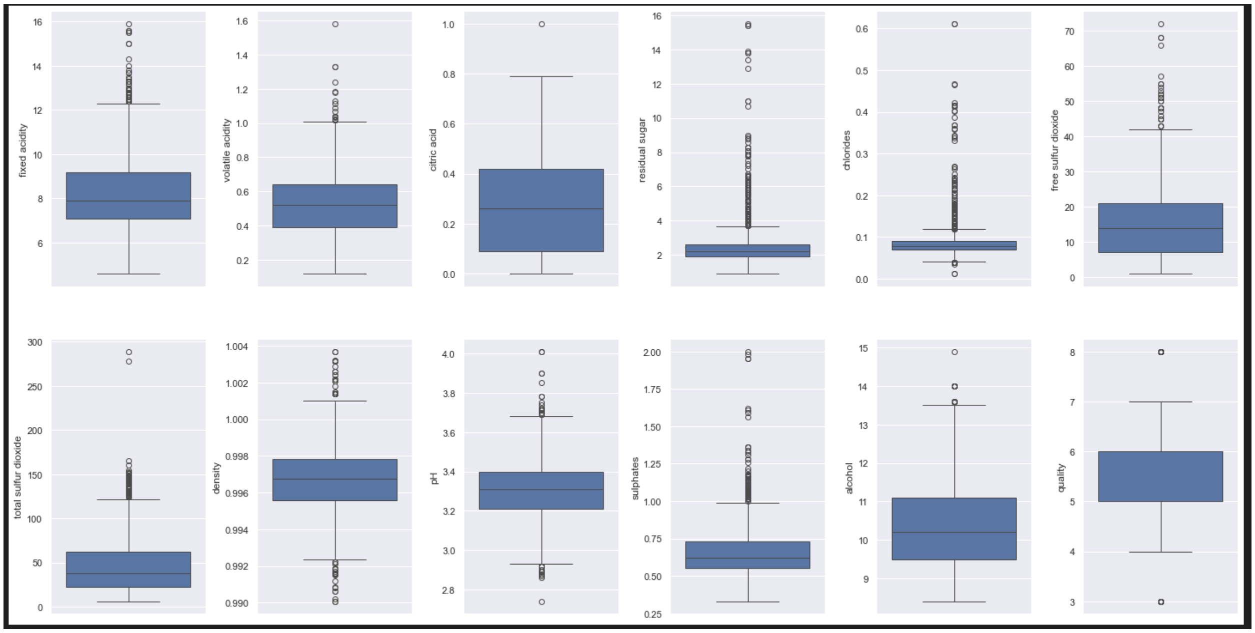


Figure 3 Outlier Detection

We have noticed that our dataset has many outliers. Data points that substantially differ from the majority are known as outliers, and they can have a big influence on machine learning models and statistical studies. The robustness and dependability of our analysis depend on our capacity to recognize and deal with these outliers. We may use a variety of strategies, such statistical approaches, or visualizations, to identify and evaluate the impact of outliers. Then, suitable techniques, such data removal or transformation, can be used to lessen their effects and improve the overall integrity of our dataset. In addition to being essential for correct modeling, addressing outliers helps us understand the underlying patterns and trends in our wine quality data.

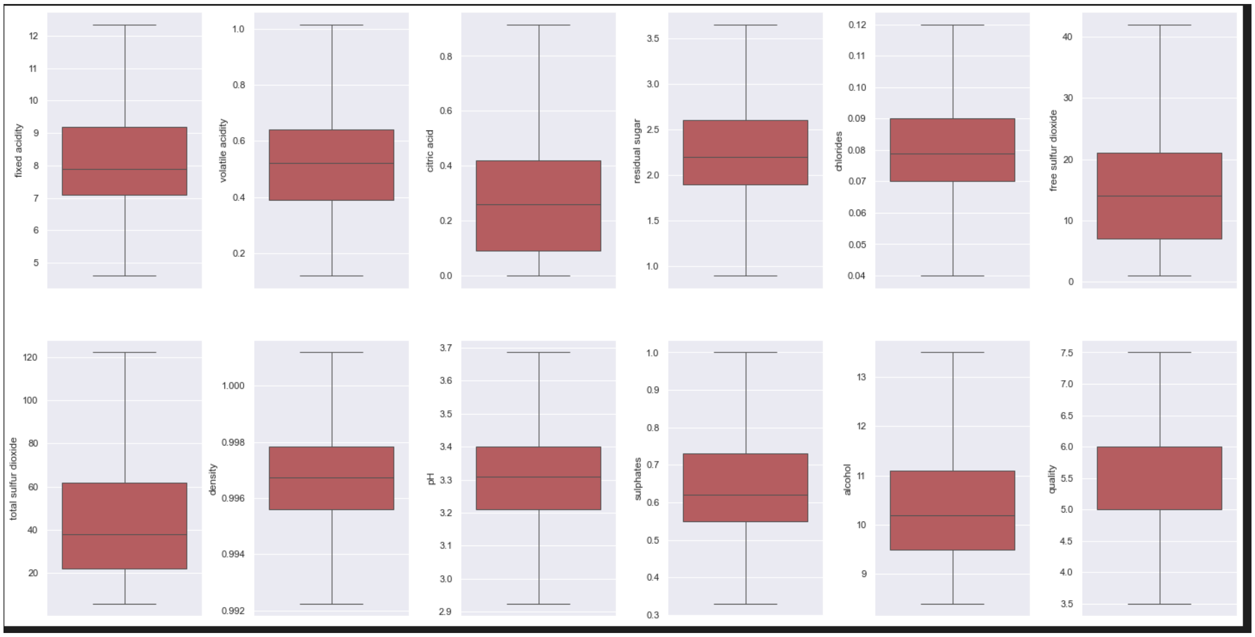


Figure 4 Removal of Outlier

It's evident from Figure 4 that we were able to eliminate every anomaly from our data. Our data now appears much more organized because to the clever method we used to handle outliers. As a result, the image now depicts a smoother, more reliable dataset. This goes beyond appearances, by eliminating these anomalies, we can prepare our data for further analysis and gain more confidence in our ability to uncover the true nature of our wines' excellence.

Finding Correlation

Insightful patterns emerge from the correlations between different characteristics in our red wine sample. Remarkably, alcohol concentration and wine quality show a positive correlation, indicating that better quality red wines are typically linked with higher alcohol content. In addition, there is a minor positive association between alcohol and pH, suggesting that a larger alcohol level may be somewhat more acidic. Further investigation reveals that density and citric acid have a strong positive connection with fixed acidity, indicating a cooperative relationship between these elements. On the other hand, density, citric acid, sulfates, and fixed acidity all show a negative connection with ph.

This extensive network of connections allows us to gain a more detailed knowledge of how these chemical properties interact and contribute to the overall quality and composition of our red wine samples.

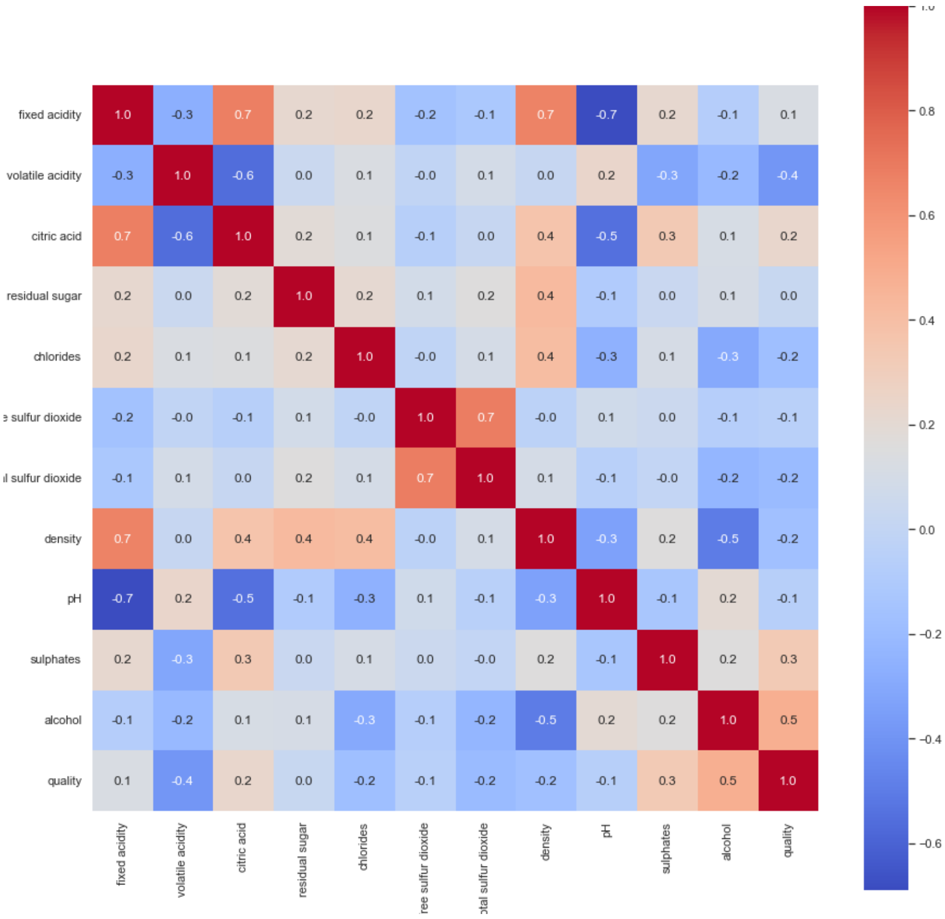


Figure 5 Correlation Heat map

When we look at the distribution of alcohol concentration in our red wine dataset, we can see that it is positively skewed in proportion to wine quality. This skewness suggests that higher-quality red wines have alcohol levels that is biased to the higher end of the curve. In simple terms, most top-quality red wines in our sample had higher-than-average alcohol level, contributing to a rightward shift in the distribution. This finding is consistent with the idea that higher alcohol levels are related with higher quality red wines, providing a clear picture of the prevalent tendency in our dataset.

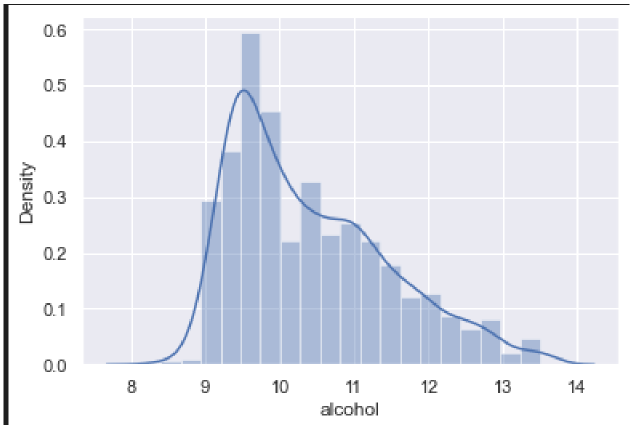


Figure 6 Positively Skewed

To check the relationship between alcohol and quality we plot a box plot. There are some outliers in shown the figure 7 that is around wine quality 5 and 6. A systematic strategy to refining the graph and reducing the visual effect of outliers entails exploiting the option to eliminate outliers. This may be accomplished by passing the “showoutliers=False” parameter. By employing this option, the graph becomes cleaner, concentrating on the main distribution while excluding the prominent outliers.

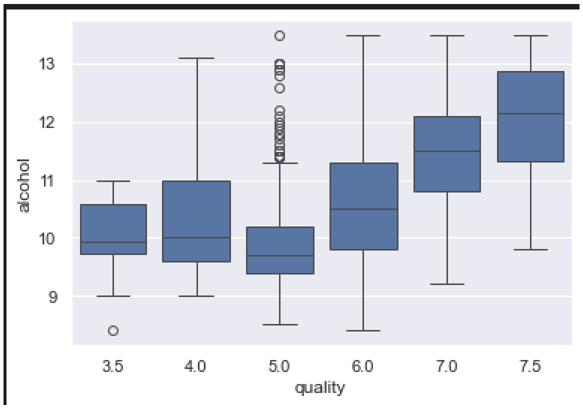


Figure 8 alcohol vs quality

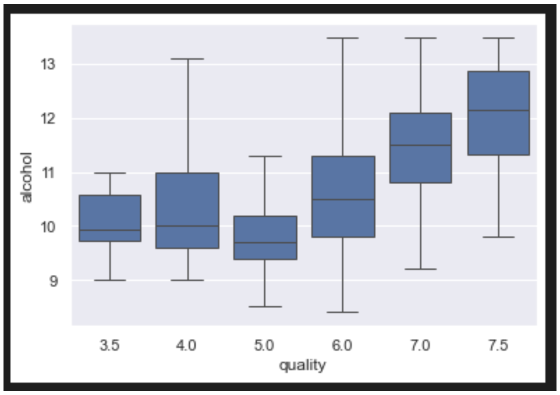


Figure 7Outlire Removal

The association between alcohol content and wine quality demonstrates a definite trend: the greater the alcohol concentration, the higher the wine's quality. This positive association implies that wines with higher alcohol content are more likely to relate to higher quality ratings. In essence, alcohol content appears as a crucial component influencing wine quality perception. This conclusion is consistent with the wine industry's larger awareness that higher alcohol concentration is frequently indicative of a more robust and well-received wine flavor.

To find out what other features affect the quality of the wine we checked some features and quality relationships with reference to our correlation graph.

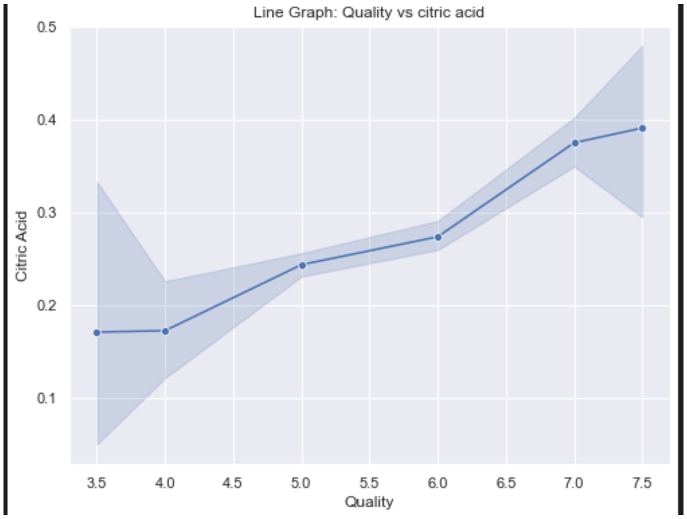


Figure 9 Citric Acid Vs Quality

The correlation between citric acid and wine quality is positively proportional, in contrast with volatile acidity. A rise in citric acid concentrations is accompanied by an improvement in wine quality. Higher quality ratings are often given to wines with higher citric acid concentration. This positive link implies that citric acid contributes positively to elevating the wine's perceived quality. The information emphasizes the role that citric acid plays in a wine's increased perceived value and possible superiority.

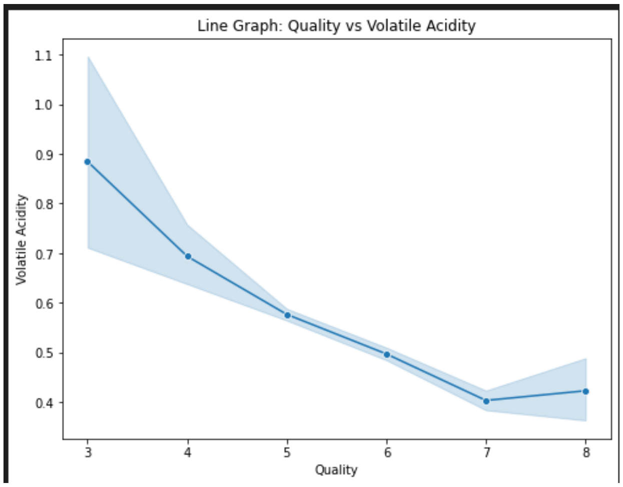
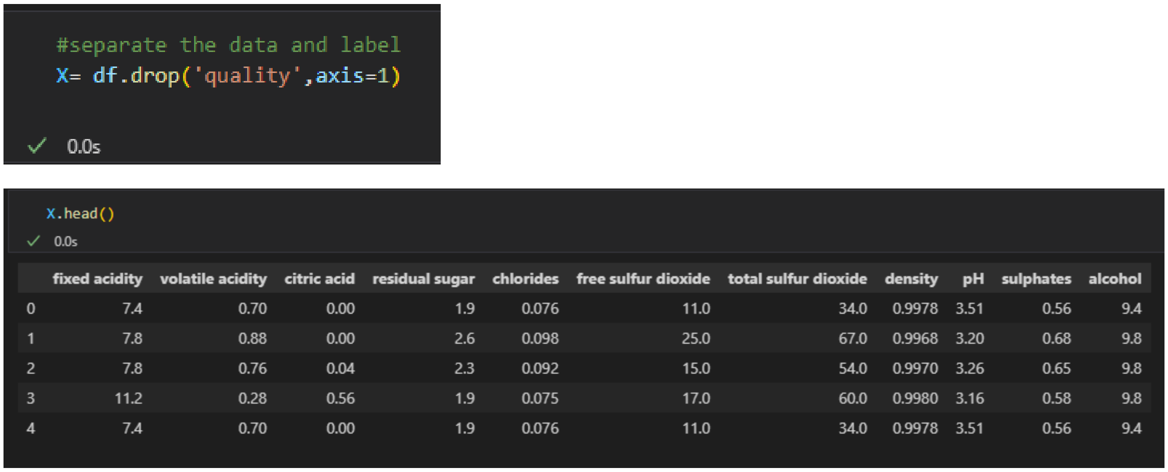


Figure 10 Volatile Acidity vs Quality

The link between volatile acidity and wine quality is inverse, and this relationship is important to note. Wine quality clearly decreases as volatile acidity rises, with lesser quality wines usually having greater volatile acidity levels. This is especially important when trying to get the best possible quality rating; for volatile acidity, a range of about 0.4 turns out to be a good cutoff point. Essentially, the evidence points to the increase of wine quality as a result of keeping volatile acidity at or around 0.4. The importance of controlling volatile acidity levels to produce and maintain greater perceived quality in wines is shown by this inverse relationship.

Data Preprocessing

For data preprocessing we separated the target variable in our case quality from the main dataset and stored the main dataset in a new variable.



```
#separate the data and label
X= df.drop('quality',axis=1)
```

✓ 0.0s

```
X.head()
```

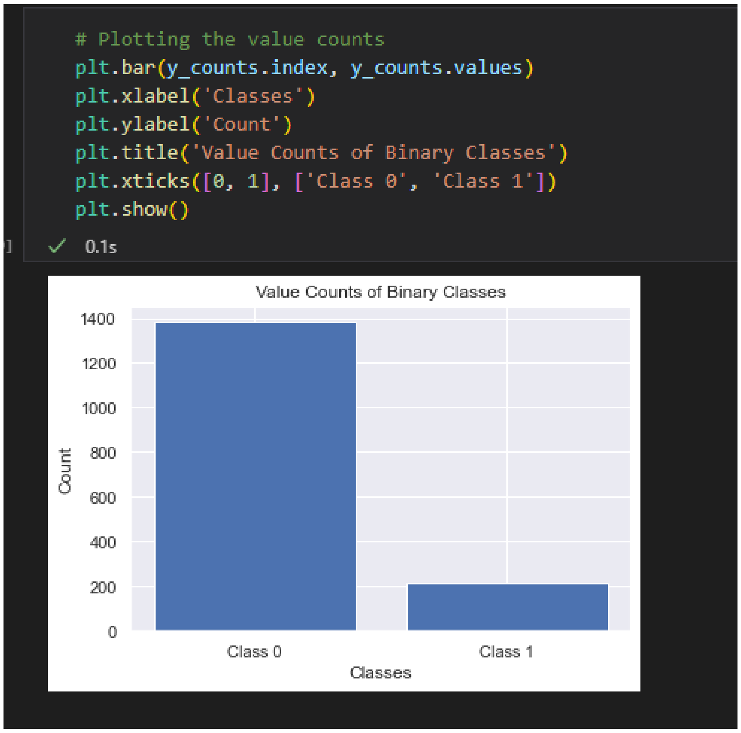
✓ 0.0s

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	0.68	9.8
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	0.65	9.8
3	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	0.58	9.8
4	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4

Figure 11 removing target variable

We strategically stored the target variable in a new variable to make the categorization of various wine kinds easier. After that, we used label binarization. By converting categorical data into a binary format, this procedure makes categorization easier to understand and more effective.

We used a clear criteria to distinguish between low and high-quality wines in our categorization effort. Wines with a quality rating of less than 5 were considered low-quality, while those with a grade of 5 or more were considered high-quality. This binary classification schema makes discriminating between wines of differing grade levels easier, and it serves as a practical and easy foundation for our classification model. This strategy approach establishes the framework for training a machine learning model to categorize wines into discrete quality classes, therefore expediting the evaluation and assessment process.



```
# Plotting the value counts
plt.bar(y_counts.index, y_counts.values)
plt.xlabel('Classes')
plt.ylabel('Count')
plt.title('Value Counts of Binary Classes')
plt.xticks([0, 1], ['Class 0', 'Class 1'])
plt.show()
```

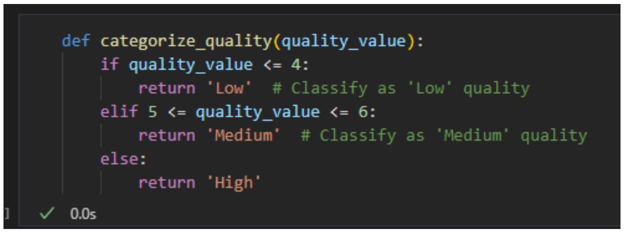
```
✓ 0.1s
```

Figure 12 Label Binarization

The identification of class imbalance in our dataset is a critical finding. The present ratio of around 6:1 between low- and high-quality wines demonstrates a large discrepancy in class distribution. This imbalance may provide difficulties during model training since the model may become skewed toward the majority class, potentially resulting in inferior performance in forecasting the minority class.

In machine learning, dealing with class imbalance is critical. Techniques like oversampling the minority class, under sampling the majority class, or utilizing specialist algorithms developed to deal with unbalanced datasets can be investigated. The capacity of the model to generalize well across both low and high-quality wine categories is dependent on striking a balance in class representation.

To address the issue of class imbalance, a strategic strategy is to shift from a binary to a multi-class categorization system. We want to give a more subtle and complete representation of the dataset by expanding our categorization approach to include many quality categories. This change enables a more equal allocation of instances across various quality levels, reducing the impact of class imbalance on model training. As a result, the model gets higher precision in distinguishing and classifying wines into separate quality groups, encouraging more robust and accurate forecasting performance.



```
def categorize_quality(quality_value):
    if quality_value <= 4:
        return 'Low'  # Classify as 'Low' quality
    elif 5 <= quality_value <= 6:
        return 'Medium'  # Classify as 'Medium' quality
    else:
        return 'High'
```

```
✓ 0.0s
```

Figure 13 Applying multi class Classifiaciton

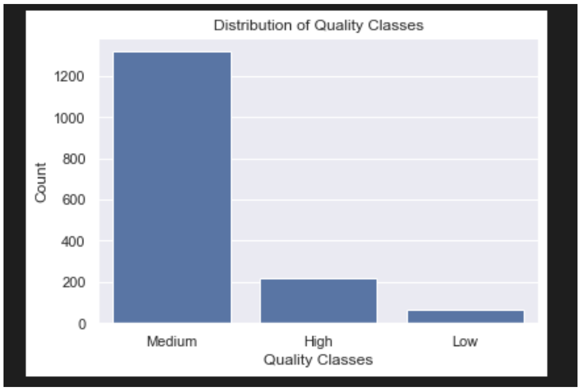


Figure 14 Multiclass Classification

We used the Synthetic Minority Over-sampling Technique (SMOTE) to address the ongoing issue of class imbalance. SMOTE is an effective strategy for balancing the class distribution by generating synthetic samples for the minority class. We want to offer the machine learning model with a fairer representation of both low and high-quality wines by oversampling the minority class.

SMOTE contributes to overcoming the obstacles provided by unbalanced datasets by improving the model's capacity to learn from the minority class and produce more accurate predictions. This intentional use of SMOTE helps to a more balanced training set, ultimately increasing the model's overall performance and dependability in wine quality classification.

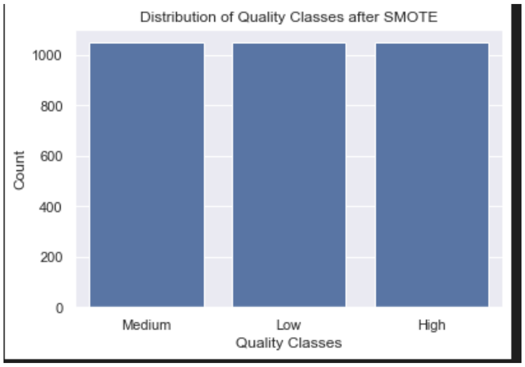
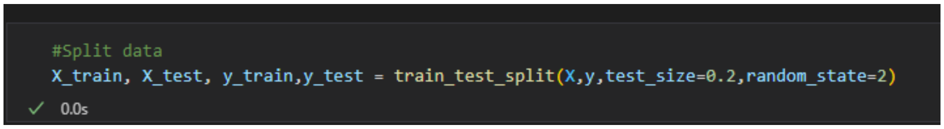


Figure 15 Distribution after oversampling

Modelling Machine Learning Algorithms

Train test split

We split our data into train test split. To be precise, we first split the data and then we applied oversampling techniques on out train data.



```
#Split data
X_train, X_test, y_train,y_test = train_test_split(X,y,test_size=0.2,random_state=2)
```

Figure 16 Trian test Split

Cross validation

We received separate performance ratings for each model in our cross-validation investigation across three different models — Logistic Regression, Random Forest Model, and Support Vector Classification. Here are the outcomes:

Logistic Regression Analysis:

[0.689, 0.648, 0.711, 0.692, and 0.695]

The average score is 0.687.

Standard deviation: 0.021 (lower is better).

Model of the Random Forest:

[0.946, 0.925, 0.929, 0.929, 0.941] Scores

The average score is 0.934.

0.008 standard deviation (lower is preferable).

Classification of the Support Vector:

[0.595, 0.548, 0.594, 0.579, 0.617] Scores

The average score is 0.587.

Standard deviation: 0.023 (lower is better).

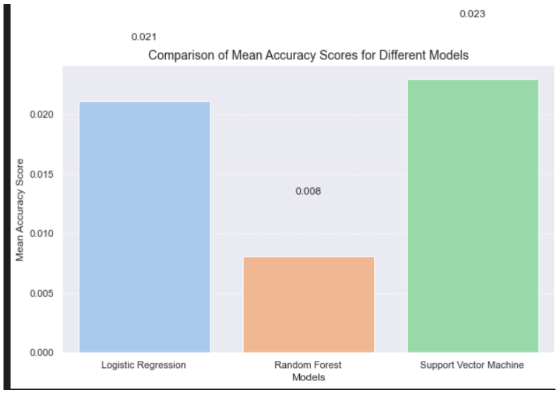


Figure 17 Cross Validation

These scores give information about each model's performance during cross-validation. The mean score represents the average accuracy over folds, whereas the standard deviation reveals performance variability. Lower standard deviations are preferable in this scenario since they indicate more consistent model performance. The findings show that the Random Forest Model outperformed the other two models, with the greatest mean score and the lowest standard deviation. This data helps us choose the best model for future improvement and deployment in our wine quality classification work.

Accuracy testing

We performed a grid search for hyperparameter tuning on a machine learning model. The "Best Parameters" and "Best Accuracy Score" outputs give useful information about the results of the grid search for hyperparameter tuning:

The best parameters are:

- maximum_depth: 0,
- min_leaf_samples: 1,
- min_samples_split:2
- n_estimators: 300

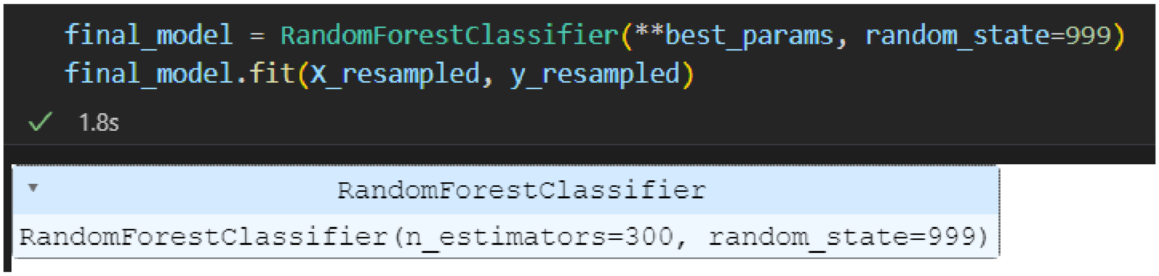
These are the ideal hyperparameter values selected by the grid search as having the maximum accuracy on the training data. Best Accuracy Rating: **0.9273015873015872** .

This shows the model's accuracy when utilizing the best collection of hyperparameters. It represents the percentage of accurate predictions based on the training data.

With these findings, you have a well-tuned Random Forest model with the stated hyperparameter configuration that outperforms the default values in terms of predictive performance. This collection of parameters can be used to train the final model for deployment and additional testing on unobserved data.

Model fit.

After testing the model, we fit our model using random forest, as it shows the best accuracy.



```
final_model = RandomForestClassifier(**best_params, random_state=999)
final_model.fit(X_resampled, y_resampled)
```

✓ 1.8s

```
▾                  RandomForestClassifier
RandomForestClassifier(n_estimators=300, random_state=999)
```

Figure 18 Model Fit

Results And Discussion

Accuracy testing on test data

After finalizing the training process and preparing the Random Forest model for deployment, rigorous testing with the dedicated test dataset was conducted. The evaluation metrics provide valuable insights into the model's performance across various aspects:

- **Accuracy:** 0.84375
    - The proportion of correctly predicted instances among the total instances in the test dataset.
- **Precision:** 0.8659630924858572

    - The accuracy of positive predictions, indicating how well the model performs when it predicts a wine quality level to be "High," "Low," or "Medium."
- **Recall:** 0.84375
    - The ability of the model to correctly identify and capture all relevant instances of "High," "Low," or "Medium" wine quality levels.
- **F1 Score:** 0.8525477398711991
    - A balanced metric that considers both precision and recall, providing a comprehensive measure of model performance.

The detailed classification report breaks down these metrics for each class ("High," "Low," "Medium"). Observing the precision, recall, and F1-score for each class offers a more granular understanding of the model's strengths and areas for improvement.

Class-wise Evaluation

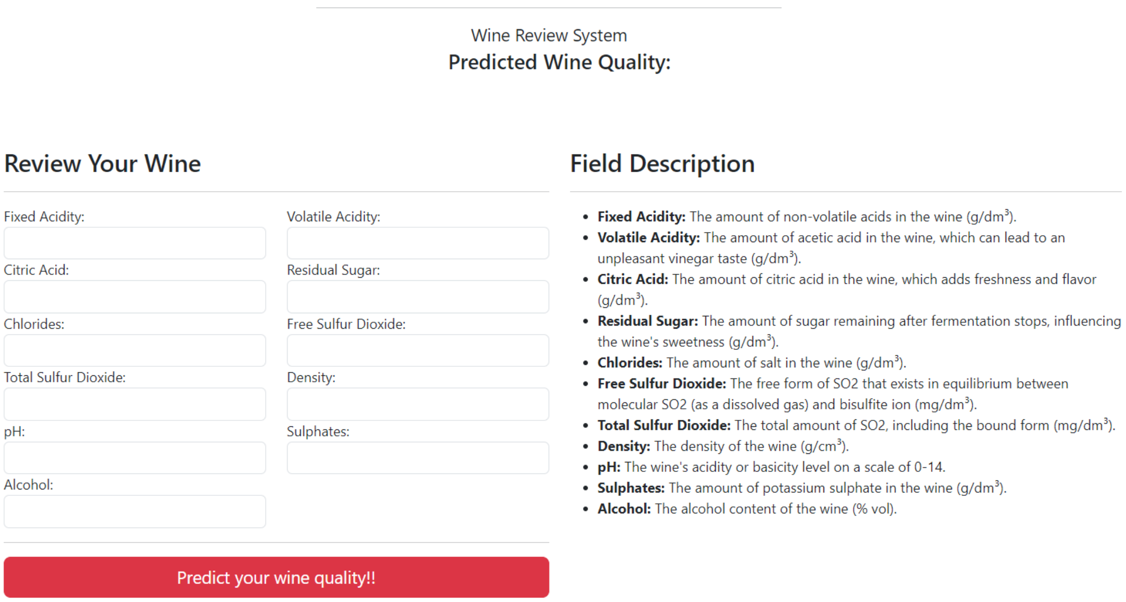
- High Quality: Precision (0.60), Recall (0.78), F1-Score (0.68)
- Low Quality: Precision (0.14), Recall (0.20), F1-Score (0.17)
- Medium Quality: Precision (0.93), Recall (0.88), F1-Score (0.90)

Results

- The model demonstrates strong performance in predicting wines with "Medium" quality.
- Predictions for "High" quality wines show good precision and recall, though there is room for improvement.
- Predictions for "Low" quality wines exhibit lower precision and recall.

Deployment

We successfully developed and tested our wine quality classification model, and then we used a Flask API to deploy it. In figure 19, it shows the landing page for our model. After filling all the fields on the application it successfully predicts the quality of wine.



Wine Review System

Predicted Wine Quality:

Review Your Wine

Fixed Acidity:

Volatile Acidity:

Citric Acid:

Residual Sugar:

Chlorides:

Free Sulfur Dioxide:

Total Sulfur Dioxide:

Density:

pH:

Sulphates:

Alcohol:

Field Description

- **Fixed Acidity:** The amount of non-volatile acids in the wine (g/dm^3).
- **Volatile Acidity:** The amount of acetic acid in the wine, which can lead to an unpleasant vinegar taste (g/dm^3).
- **Citric Acid:** The amount of citric acid in the wine, which adds freshness and flavor (g/dm^3).
- **Residual Sugar:** The amount of sugar remaining after fermentation stops, influencing the wine's sweetness (g/dm^3).
- **Chlorides:** The amount of salt in the wine (g/dm^3).
- **Free Sulfur Dioxide:** The free form of SO2 that exists in equilibrium between molecular SO2 (as a dissolved gas) and bisulfite ion (mg/dm^3).
- **Total Sulfur Dioxide:** The total amount of SO2, including the bound form (mg/dm^3).
- **Density:** The density of the wine (g/cm^3).
- **pH:** The wine's acidity or basicity level on a scale of 0-14.
- **Sulphates:** The amount of potassium sulphate in the wine (g/dm^3).
- **Alcohol:** The alcohol content of the wine (% vol).

Figure 19 Deployment

Conclusion And Future Work

Conclusion

This research, which classifies wine quality, represents an in-depth review of data, preprocessing, and model building. Using Random Forest as the selected technique, our model performed well on the test dataset, obtaining an accuracy of 84.38%.

Our investigation revealed the importance of several chemical characteristics in defining wine quality, with citric acid, volatile acidity, and alcohol concentration emerging as major determinants. We deliberately switched from binary to multi-class classification after realizing the difficulty of class imbalance in quality ratings, and we used the Synthetic Minority Over-sampling Technique (SMOTE) to effectively mitigate the problem.

Robust hyperparameter adjustment is crucial, as demonstrated by the Random Forest model's effectiveness. The grid search method found the best settings to maximize prediction performance overall and accuracy.

Finally, we deployed our model using flask API.

Future Work

- Examine other chemical characteristics that can affect the quality of wine, keeping in mind that adding new characteristics would improve the model's comprehension.
- Try experimenting with more sophisticated machine learning algorithms than just Random Forest to see if there are any other models that can predict wine quality more accurately.
- To make an interface that is more user-friendly, get user input to evaluate usability and satisfaction. Then, introduce features or make improvements depending on user experiences.

- Provide a means for dynamic model updates so that the system may adjust and develop in response to changes in wine quality trends over time.
- To verify robustness and generalizability, validate the model's performance on a variety of datasets representing various wine varieties or areas.
- Develop a mobile application that would allow users to conveniently get wine quality estimates while on the go, appealing to a wider audience.
- Work along with sommeliers or wine specialists to add domain-specific information to the model, which might improve its accuracy.
- Install ongoing monitoring tools to find any drift in data patterns and make sure that models are updated on time to maintain their relevance.
- To enhance the model's performance even further, investigate specific methods for addressing imbalances in multi-class settings.

Project Link

Git hub link: https://github.com/lakash56/Wine_Predition_Project/tree/main