Wine Quality Data Analysis

Exploring Insights from Wine Quality Datasets

Presented by -
Aakiff Panjwani
Sai Kurra
Srija Anasuri

Project Motivation and Objectives

- Why analyse wine quality?

  Wine quality is an important aspect of the wine industry, impacting both consumer satisfaction and market value.

- Identify and analyze the relationships between the physical/chemical features of wines and wine quality ratings.
- Build and compare various Machine Learning models to predict wine quality.
- To assess how dimensionality reduction techniques like PCA (Principal Component Analysis) can enhance the performance of machine learning models.

Dataset Overview

The dataset includes red and white wine samples of the Portuguese "Vinho Verde" wine, with input features based on chemical tests and output based on sensory evaluations from wine experts, who rated quality on a scale from 0 to 10.

- Number of Instances
  - Red wine - 1599
  - White wine - 4898
- Input variables
  - fixed acidity
  - volatile acidity
  - citric acid
  - residual sugar
  - chlorides
  - free sulfur dioxide
  - total sulfur dioxide
  - density
  - pH
  - sulphates
  - alcohol
- Output/Target variable
  - quality (score between 0 and 10)

Dataset Source - UCI Machine Learning Repository
https://archive.ics.uci.edu/dataset/186/wine+quality

Research Questions

- **What are the correlations between physicochemical properties such as fixed acidity, volatile acidity, pH, and alcohol content of wine towards its quality?**
  - Explore how these physicochemical features influence wine quality and identify significant correlations.
- **How effective are various classification models in predicting wine quality based on available features in a dataset?**
  - Assess the performance of different classification models in predicting wine quality, such as wine quality, using chemical and other feature data.
- **How can employing dimensionality reduction techniques like PCA enhance the accuracy and performance of the ML models in predicting wine quality?**
  - PCA reduces the number of features, helping to prevent overfitting and multicollinearity while highlighting the most important information, leading to better model accuracy and efficiency.

Data Analysis Steps

Collection

Cleaning

Exploratory

Statistical Modeling

Visualization

Insights

Dataset Preprocessing

- **Data Loading and Integration**
  - Red and white wine datasets are loaded into separate DataFrames, and a ‘WineType’ column is added to distinguish between them before concatenating into a single DataFrame for analysis.
- **Initial Data Exploration**
  - **Shape and Structure** - The combined DataFrame consists of 6,497 rows and 13 columns, indicating a well-structured dataset for analysis.

```
In [351]: # Shape of the Wine dataset
          Wine.shape

Out[351]: (6497, 13)

In [352]: # Features of the Wine dataset
          Wine.columns

Out[352]: Index(['fixed acidity', 'volatile acidity', 'citric acid', 'residual sugar',
                 'chlorides', 'free sulfur dioxide', 'total sulfur dioxide', 'density',
                 'pH', 'sulphates', 'alcohol', 'quality', 'WineType'],
                dtype='object')
```

- **Descriptive Statistics** - Descriptive statistics are generated to provide insights into the distribution and central tendencies of the features.

In [388]:
```
# Summary of the Wine dataset
Wine.describe()
```

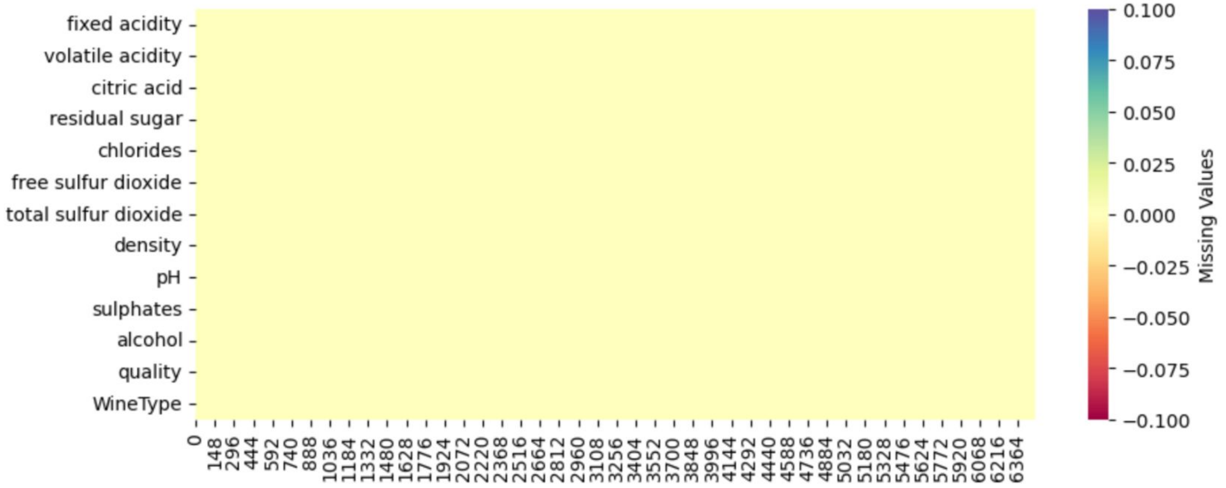
Out[388]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	
count	6497.000000	6497.000000	6497.000000	6497.000000	6497.000000	6497.000000	6497.000000	6497.000000	6497.000000	6497.000000	6497.000000	6497
mean	7.215307	0.339666	0.318633	5.443235	0.056034	30.525319	115.744574	0.994697	3.218501	0.531268	10.491801	5
std	1.296434	0.164636	0.145318	4.757804	0.035034	17.749400	56.521855	0.002999	0.160787	0.148806	1.192712	0
min	3.800000	0.080000	0.000000	0.600000	0.009000	1.000000	6.000000	0.987110	2.720000	0.220000	8.000000	3
25%	6.400000	0.230000	0.250000	1.800000	0.038000	17.000000	77.000000	0.992340	3.110000	0.430000	9.500000	5
50%	7.000000	0.290000	0.310000	3.000000	0.047000	29.000000	118.000000	0.994890	3.210000	0.510000	10.300000	6
75%	7.700000	0.400000	0.390000	8.100000	0.065000	41.000000	156.000000	0.996990	3.320000	0.600000	11.300000	6
max	15.900000	1.580000	1.660000	65.800000	0.611000	289.000000	440.000000	1.038980	4.010000	2.000000	14.900000	9

- **Data Cleaning**
  - A check for missing values reveals no null entries across all columns, ensuring data completeness for analysis.

```
In [354]: Wine.isnull().sum()

Out[354]: fixed acidity           0
          volatile acidity        0
          citric acid             0
          residual sugar          0
          chlorides               0
          free sulfur dioxide     0
          total sulfur dioxide    0
          density                 0
          pH                      0
          sulphates               0
          alcohol                 0
          quality                 0
          WineType                0
          dtype: int64
```

```
In [390]: plt.figure(figsize=(10,4))
          sns.heatmap(Wine.isna().transpose(),
                      cmap="Spectral",
                      cbar_kws={'label': 'Missing Values'})
Out[390]: <Axes: >
```



Visual representation of missing values

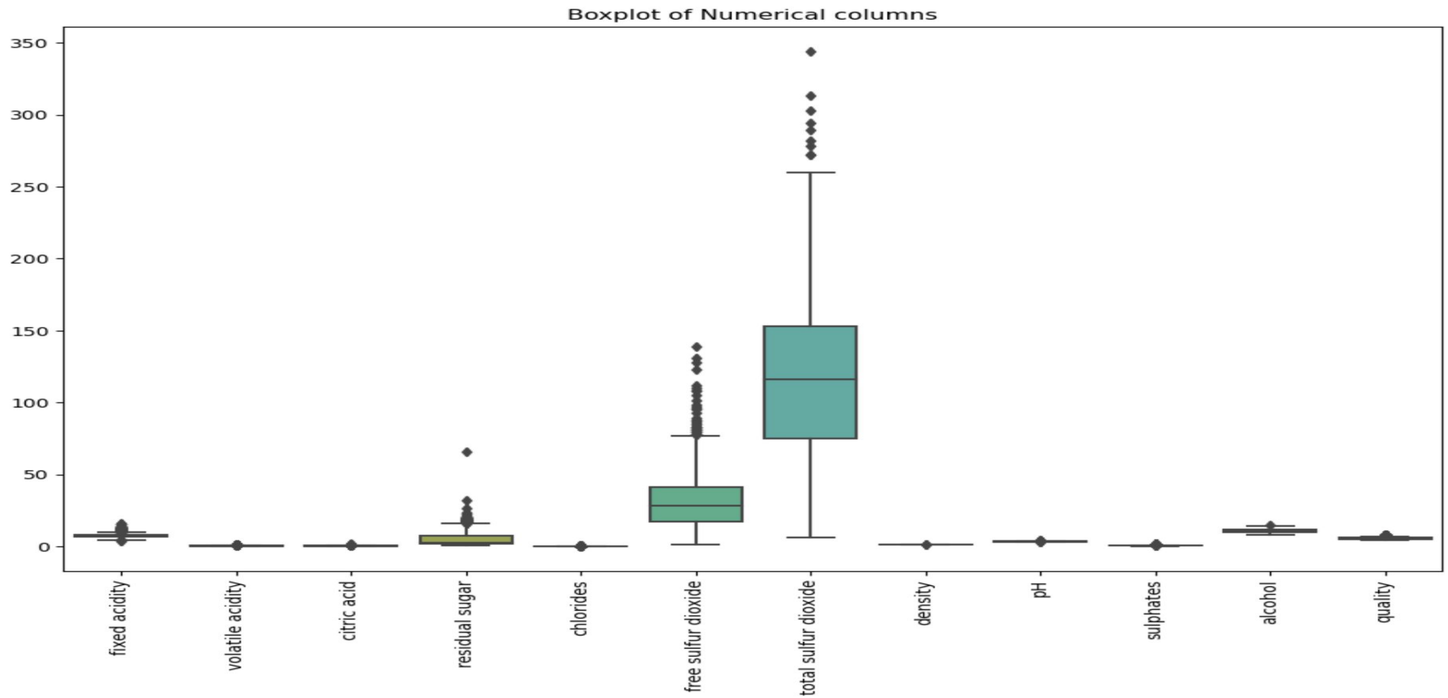
- Duplicate rows are identified using the duplicated() method. This step is vital as duplicates have the potential to distort analysis results and impact model performance. Specific duplicate entries for both red and white wines were found, showcasing identical values across all features. Duplicates are dropped using drop_duplicates(), resulting in a cleaned dataset without redundancy.

2. Checking for duplicate values.

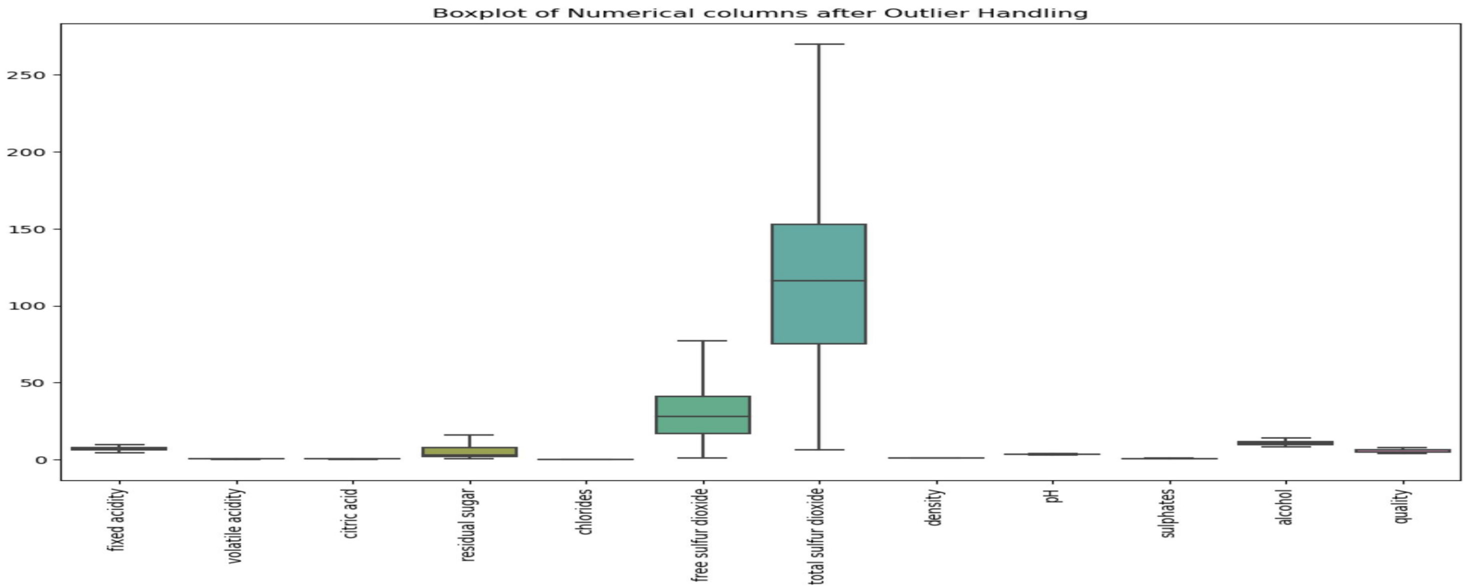
```
In [391]: duplicate_Rows = Wine[Wine.duplicated(keep=False)]
          #print("Duplicate Rows - \n", duplicate_Rows)
          print("Duplicate Rows Count - ", Wine.duplicated().sum())
          New_Wine = Wine.drop_duplicates(keep = "first")
          print("Checking if duplicate rows are removed from the dataset - ", New_Wine.duplicated().sum())

          Duplicate Rows Count -  1177
          Checking if duplicate rows are removed from the dataset -  0
```

- **Label Encoding**
    - Label encoding assigns a unique integer to each category. In this case, we encoded "red" as 0 and "white" as 1.

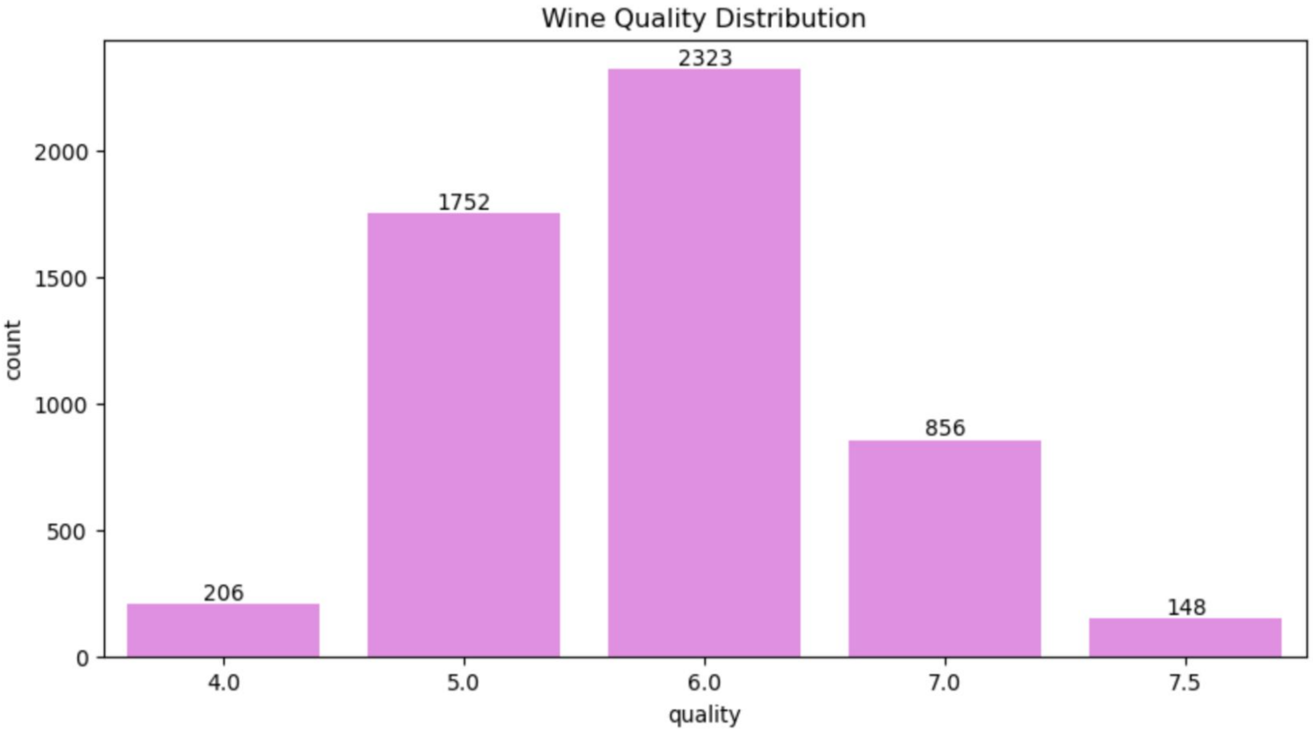
- **Outlier Detection -**
    - Used boxplots to visualize the distribution of the data and identify potential outliers.



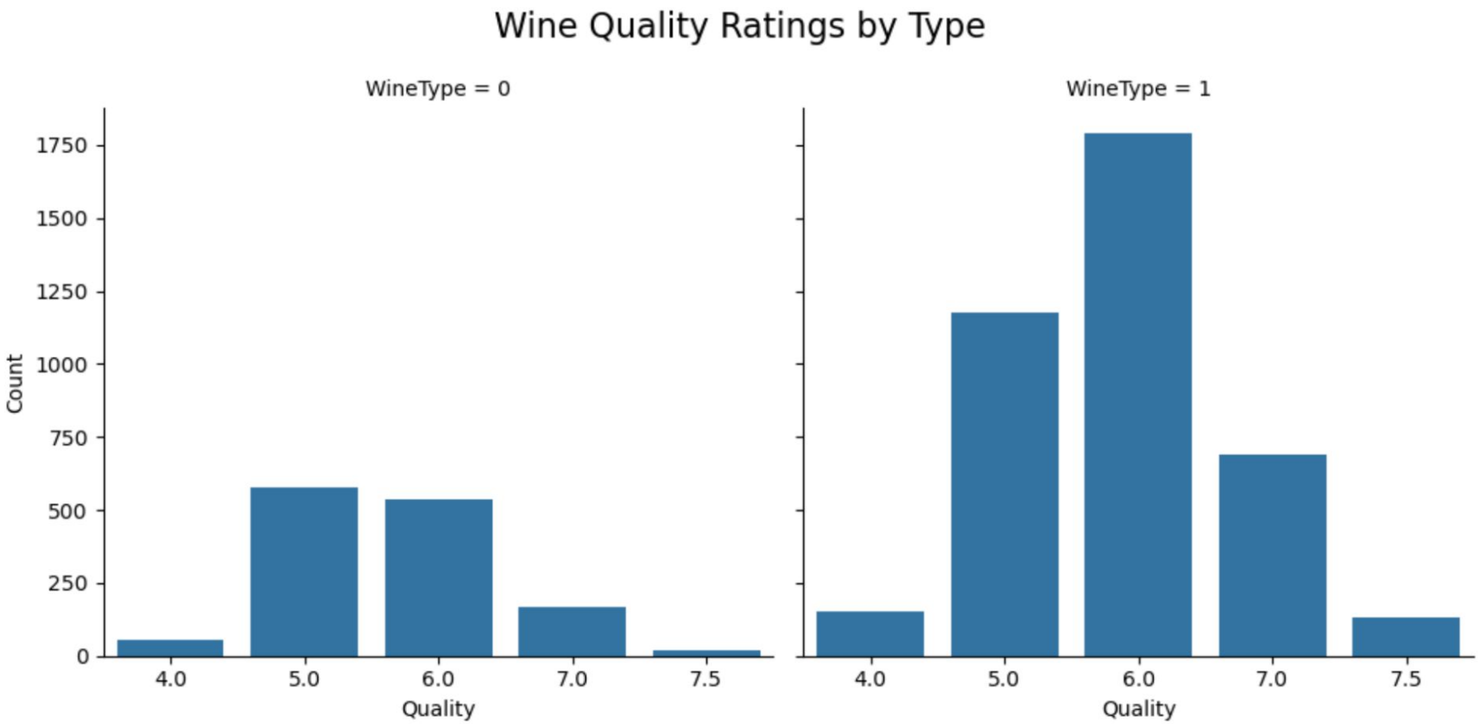
- For our dataset, we used Interquartile Range (IQR) to handle outliers because it is a robust method that is less sensitive to extreme values compared to methods like mean and standard deviation. This makes IQR well-suited for handling skewed data.
- Imputation replaces these outliers rather than removing them, which helps retain as much data as possible while addressing extreme values.



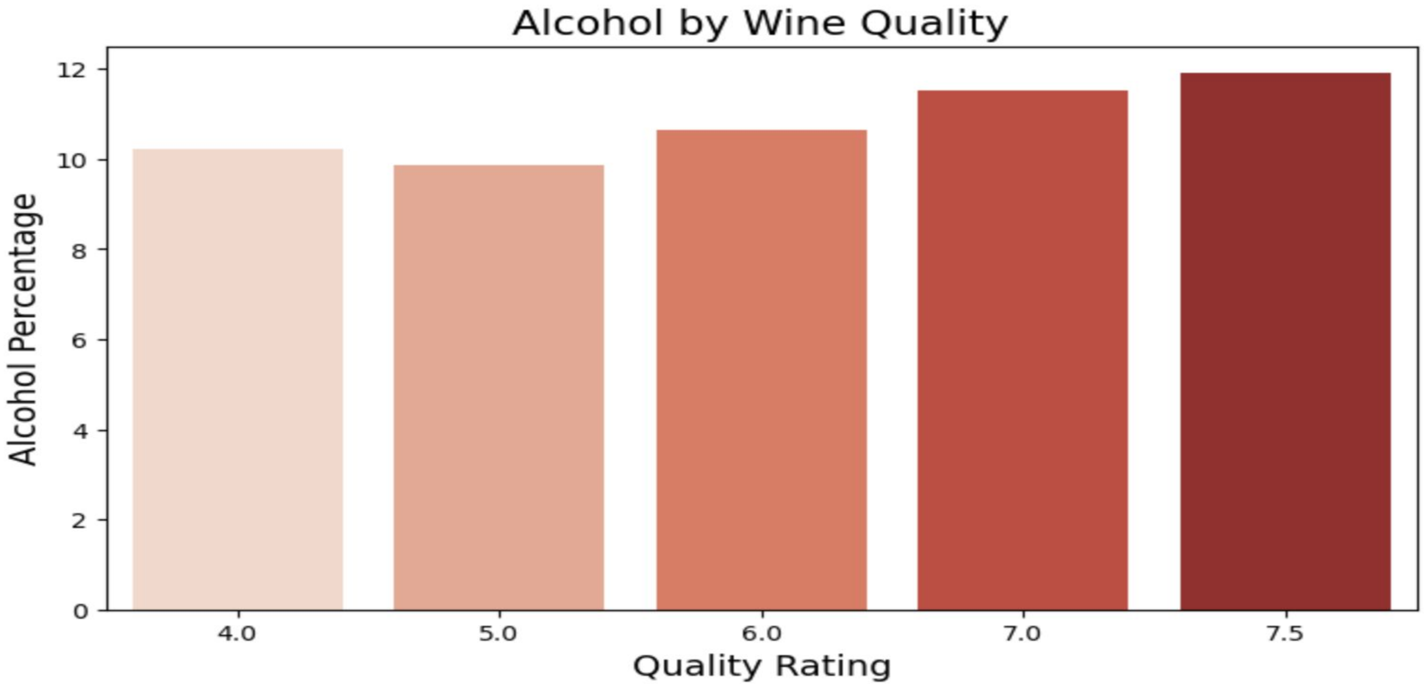
EDA and Findings



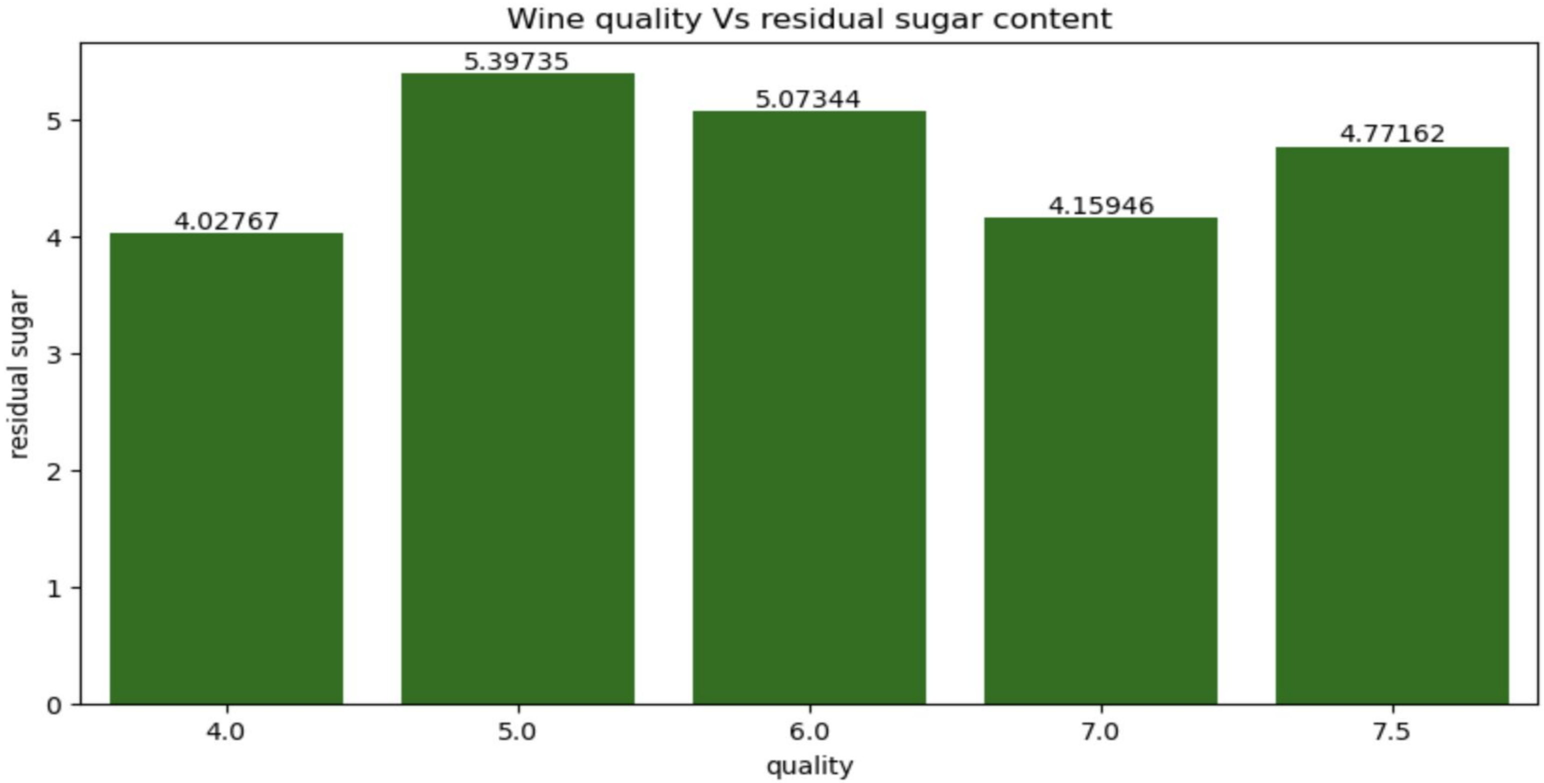
Based on the wine quality distribution data, it is evident that the majority of the wine samples are rated as 6 in terms of quality, while there are notably fewer samples with a quality rating of 4 and 7.5.



Red wines (0) typically achieve a rating of 5 and 6. The majority of white wines (1) tend to receive a quality rating of 5, 6 and 7.



Visual of how important alcohol content is, from the graph it is evident that more the alcohol percentage the better rating the wine got.



Wines with quality 5 have the highest residual sugar, while quality 4 wines have the lowest. Residual sugar slightly decreases in premium wines (quality 7+), reflecting balanced sweetness. Moderate sugar levels seem key to mid-quality wines.

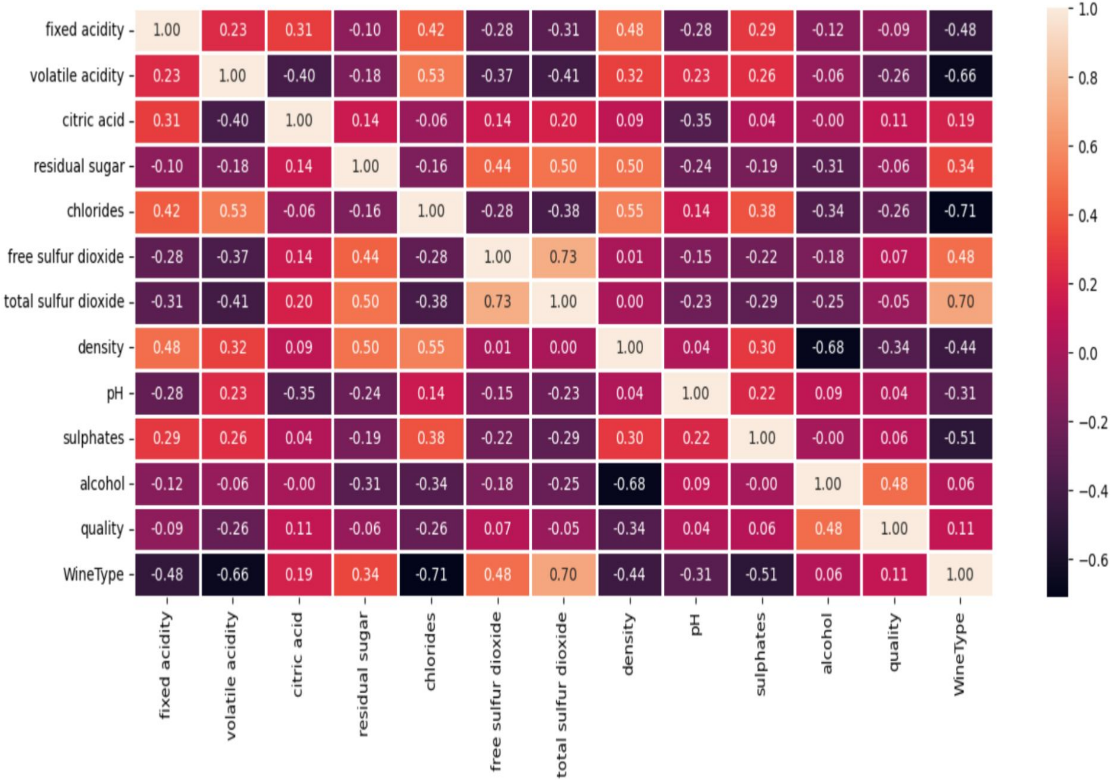
Research Question 1

- **What are the correlations between physicochemical properties such as fixed acidity, volatile acidity, pH, and alcohol content of wine towards its quality?**
  - Exploring how these physicochemical features influence wine quality and identify significant correlations.

The relationship between physicochemical properties such as fixed acidity, volatile acidity, pH, and alcohol content and wine quality can be explored using a correlation matrix. This statistical tool helps identify how strongly these features are linearly related to wine quality. The correlation matrix highlights these relationships and helps pinpoint the most influential properties affecting wine quality.

```
In [468]: corr_matrix = New_Wine.corr()

# Sorting the correlations with respect to the 'quality' column in descending order
corr_quality = corr_matrix['quality'].sort_values(ascending = False)
print(corr_quality)
```

```
quality                 1.000000
alcohol                 0.478052
WineType                0.113073
citric acid             0.106081
free sulfur dioxide     0.070752
sulphates               0.059916
pH                      0.044203
total sulfur dioxide   -0.051300
residual sugar         -0.058827
fixed acidity          -0.094370
chlorides              -0.260154
volatile acidity       -0.261492
density                -0.335902
Name: quality, dtype: float64
```


fixed acidity 1.00 0.23 0.31 -0.10 0.42 -0.28 -0.31 0.48 -0.28 0.29 -0.12 -0.09 -0.48
volatile acidity 0.23 1.00 -0.40 -0.18 0.53 -0.37 -0.41 0.32 0.23 0.26 -0.06 -0.26 -0.66
citric acid 0.31 -0.40 1.00 0.14 -0.06 0.14 0.20 0.09 -0.35 0.04 -0.00 0.11 0.19
residual sugar -0.10 -0.18 0.14 1.00 -0.16 0.44 0.50 0.50 -0.24 -0.19 -0.31 -0.06 0.34
chlorides 0.42 0.53 -0.06 -0.16 1.00 -0.28 -0.38 0.55 0.14 0.38 -0.34 -0.26 -0.71
free sulfur dioxide -0.28 -0.37 0.14 0.44 -0.28 1.00 0.73 0.01 -0.15 -0.22 -0.18 0.07 0.48
total sulfur dioxide -0.31 -0.41 0.20 0.50 -0.38 0.73 1.00 0.00 -0.23 -0.29 -0.25 -0.05 0.70
density 0.48 0.32 0.09 0.50 0.55 0.01 0.00 1.00 0.04 0.30 -0.68 -0.34 -0.44
pH -0.28 0.23 -0.35 -0.24 0.14 -0.15 -0.23 0.04 1.00 0.22 0.09 0.04 -0.31
sulphates 0.29 0.26 0.04 -0.19 0.38 -0.22 -0.29 0.30 0.22 1.00 -0.00 0.06 -0.51
alcohol -0.12 -0.06 -0.00 -0.31 -0.34 -0.18 -0.25 -0.68 0.09 -0.00 1.00 0.48 0.06
quality -0.09 -0.26 0.11 -0.06 -0.26 0.07 -0.05 -0.34 0.04 0.06 0.48 1.00 0.11
WineType -0.48 -0.66 0.19 0.34 -0.71 0.48 0.70 -0.44 -0.31 -0.51 0.06 0.11 1.00
fixed acidity
volatile acidity
citric acid
residual sugar
chlorides
free sulfur dioxide
total sulfur dioxide
density
pH
sulphates
alcohol
quality
WineType
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6

The summary of the correlations between wine features -

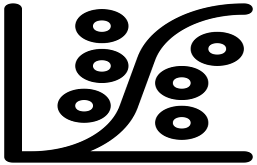
- WineType and Volatile Acidity - Strong negative correlation (-0.66)
- WineType and Total Sulfur Dioxide - Positive correlation (0.70)
- Alcohol and Quality - Moderate positive correlation (0.47)
- Density and Alcohol - Strong negative correlation (-0.68)
- Free Sulfur Dioxide and Total Sulfur Dioxide - Very strong positive correlation (0.73)

Research Question 2

- **How effective are various classification models in predicting wine quality based on available features in a dataset?**
    - Assess the performance of different classification models in predicting wine quality, using chemical and other feature data.

To predict wine quality, Logistic Regression, SVM, and Random Forest can be used. Logistic Regression works well for linear relationships but may struggle with complex data. SVM is effective for high-dimensional data and clear class separations but is computationally expensive. Random Forest excels at capturing non-linear relationships and feature interactions, typically offering the best performance. Random Forest outperforms the other models in accuracy, while Logistic Regression is a good baseline, and SVM is effective for well-separated classes.

Machine Learning and Data Mining Models



Logistic Regression

Random Forest

Support Vector Machine

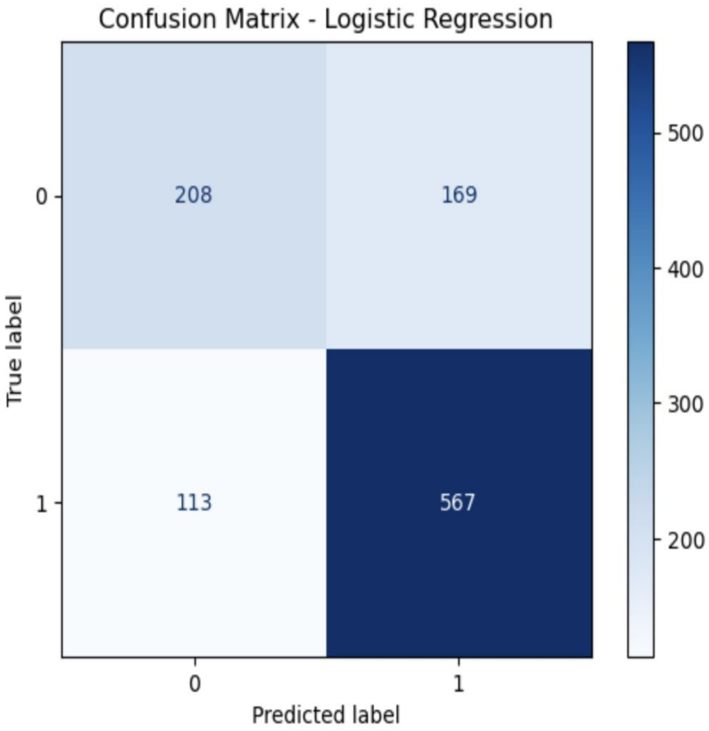
Logistic Regression

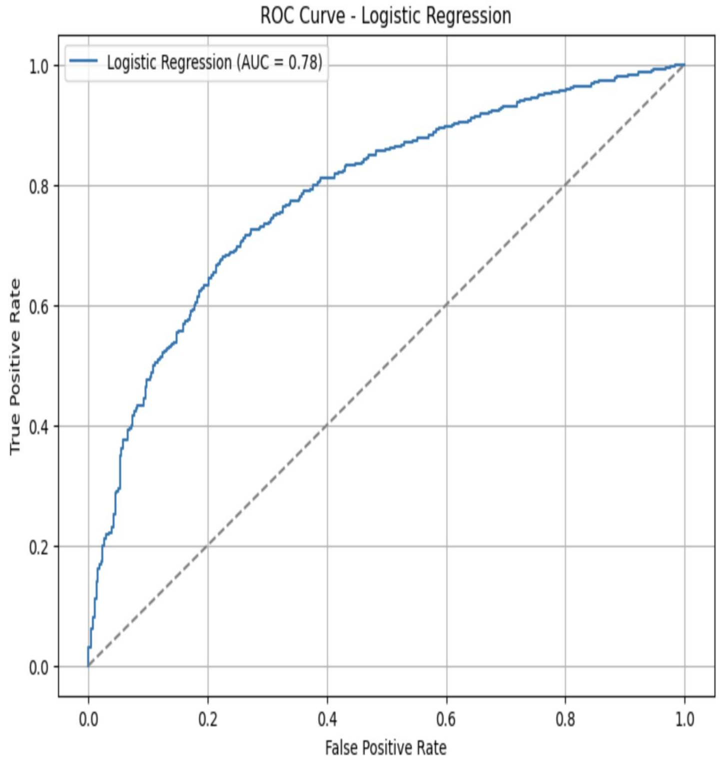
Logistic Regression Classification Report -

	precision	recall	f1-score	support
0	0.65	0.55	0.60	377
1	0.77	0.83	0.80	680
accuracy			0.73	1057
macro avg	0.71	0.69	0.70	1057
weighted avg	0.73	0.73	0.73	1057

Accuracy - 0.7332071901608326

ROC-AUC Score - 0.7841199875175535


Confusion Matrix - Logistic Regression
208
169
113
567
0
1
True label
Predicted label
500
400
300
200

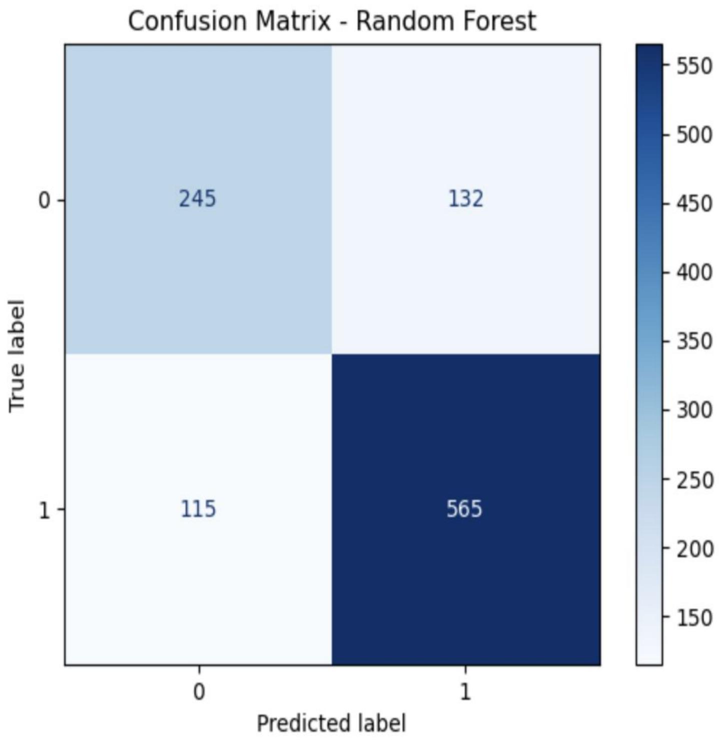

ROC Curve - Logistic Regression
Logistic Regression (AUC = 0.78)
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0

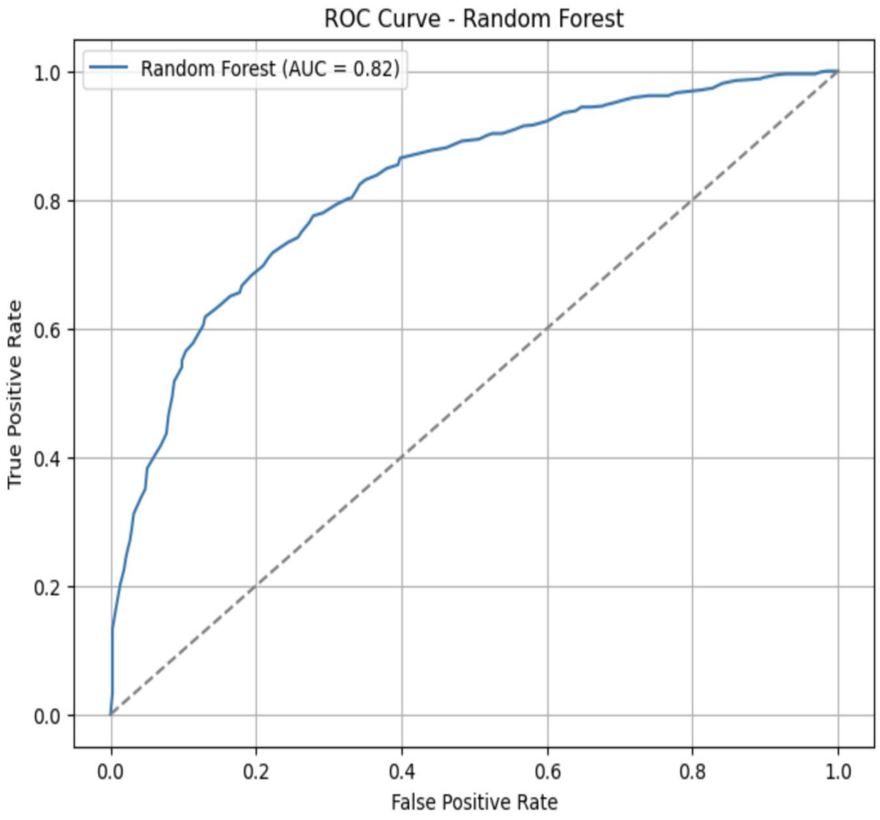
Random Forest

Random Forest Classification Report -

	precision	recall	f1-score	support
0	0.68	0.65	0.66	377
1	0.81	0.83	0.82	680
accuracy			0.77	1057
macro avg	0.75	0.74	0.74	1057
weighted avg	0.76	0.77	0.77	1057

Accuracy - 0.7663197729422895
ROC-AUC Score - 0.8213917927913871


Confusion Matrix - Random Forest
245
132
115
565
0
1
True label
0
1
Predicted label
550
500
450
400
350
300
250
200
150

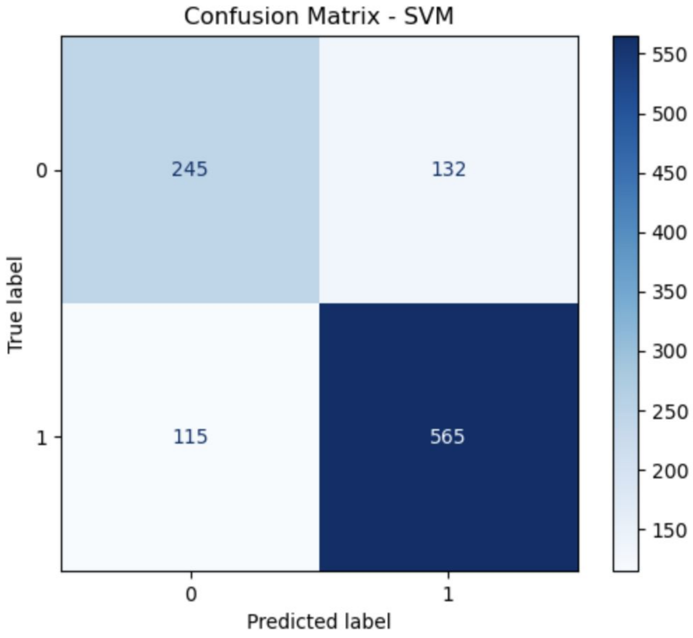

ROC Curve - Random Forest
Random Forest (AUC = 0.82)
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0

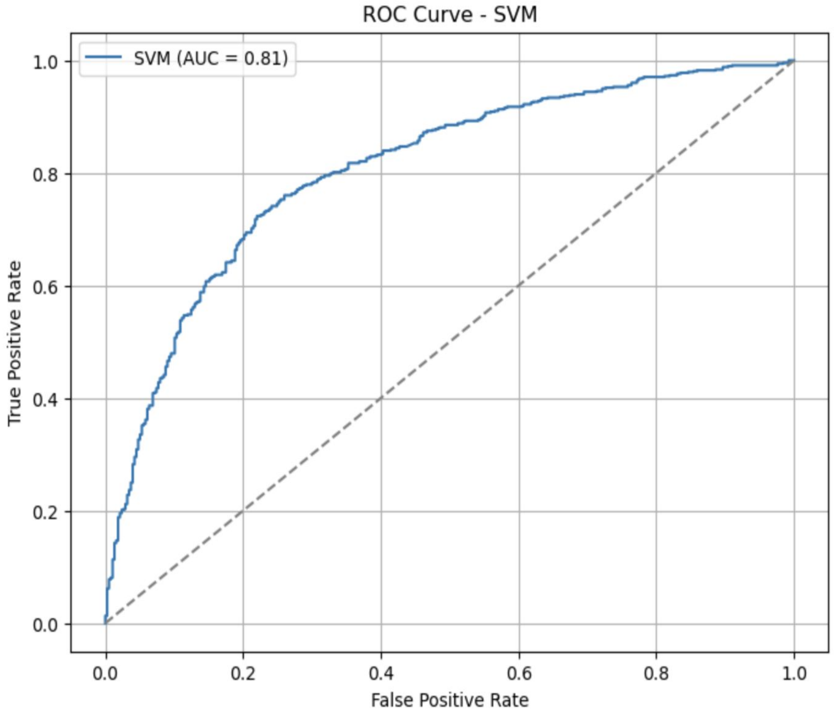
Support Vector Machine

SVM Classification Report:

	precision	recall	f1-score	support
0	0.67	0.58	0.62	377
1	0.78	0.84	0.81	680
accuracy			0.75	1057
macro avg	0.73	0.71	0.72	1057
weighted avg	0.74	0.75	0.74	1057

Accuracy: 0.7483443708609272
ROC-AUC Score: 0.8068848494304883


Confusion Matrix - SVM
245
132
115
565
0
1
True label
Predicted label
550
500
450
400
350
300
250
200
150

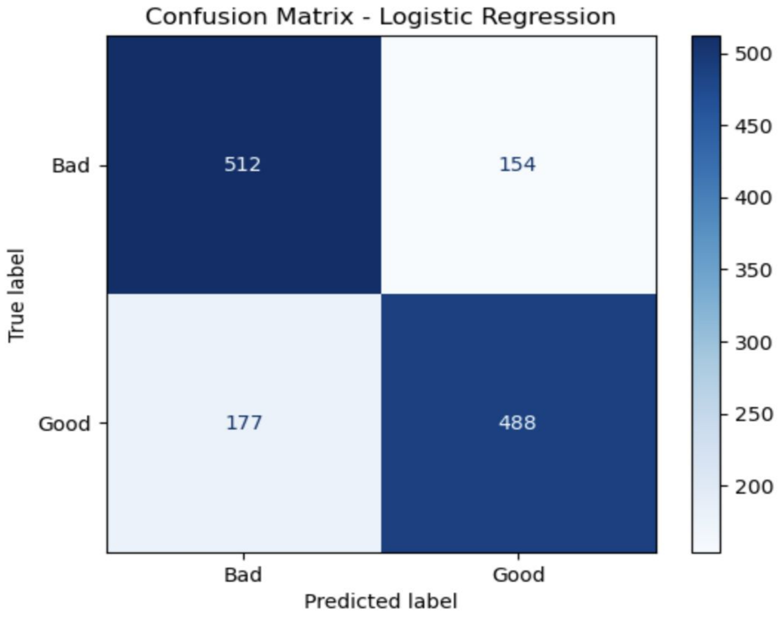

ROC Curve - SVM
SVM (AUC = 0.81)
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0

Modeling after applying SMOTE for class imbalance

Logistic Regression Classification Report -

	precision	recall	f1-score	support
0	0.74	0.77	0.76	666
1	0.76	0.73	0.75	665
accuracy			0.75	1331
macro avg	0.75	0.75	0.75	1331
weighted avg	0.75	0.75	0.75	1331

Logistic Regression Accuracy - 0.7513148009015778
Logistic Regression ROC-AUC - 0.818717966086387

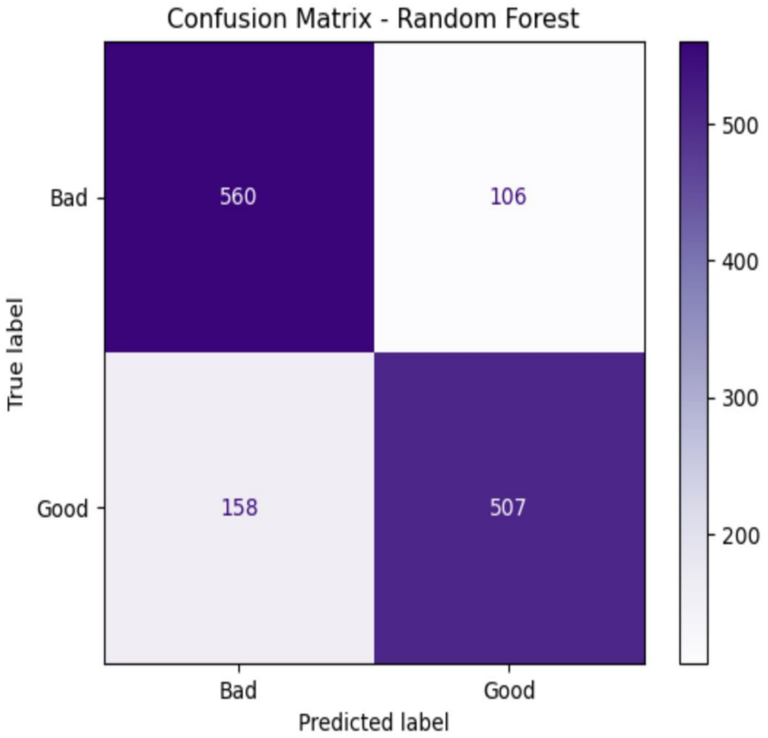


Random Forest Classification Report -

	precision	recall	f1-score	support
0	0.78	0.84	0.81	666
1	0.83	0.76	0.79	665
accuracy			0.80	1331
macro avg	0.80	0.80	0.80	1331
weighted avg	0.80	0.80	0.80	1331

Random Forest Accuracy - 0.8016528925619835

Random Forest ROC-AUC - 0.8863340784393416

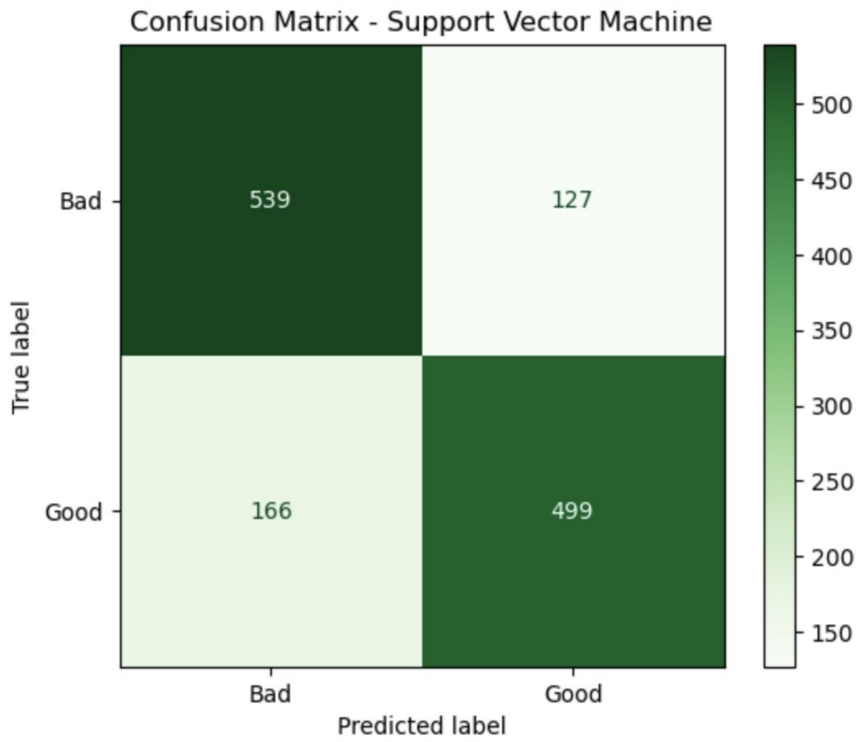


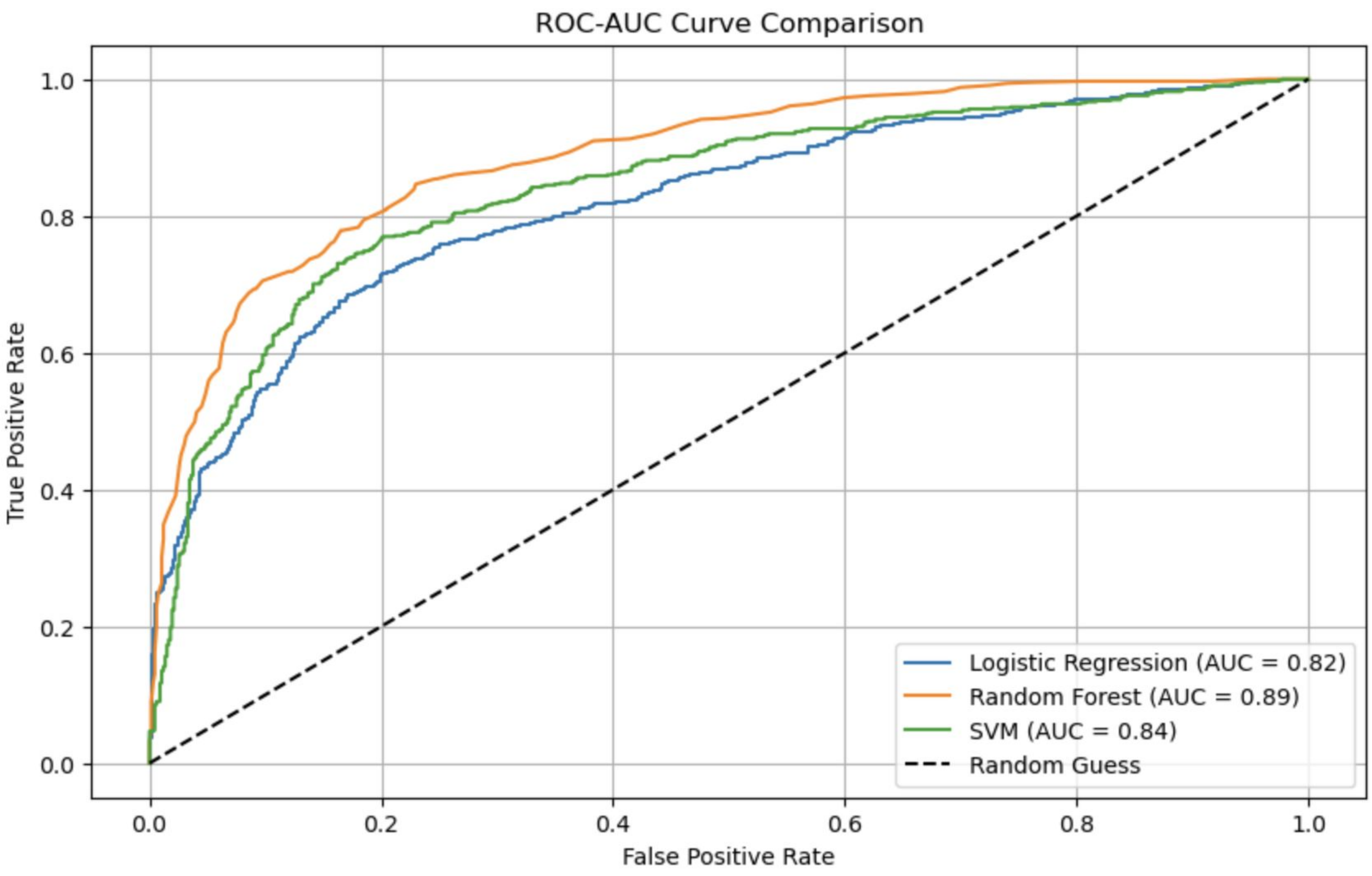
SVM Classification Report -

	precision	recall	f1-score	support
0	0.76	0.81	0.79	666
1	0.80	0.75	0.77	665
accuracy			0.78	1331
macro avg	0.78	0.78	0.78	1331
weighted avg	0.78	0.78	0.78	1331

SVM Accuracy - 0.7798647633358378

SVM ROC-AUC - 0.8411592043170991





ROC-AUC Curve Comparison
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
Logistic Regression (AUC = 0.82)
Random Forest (AUC = 0.89)
SVM (AUC = 0.84)
Random Guess

After applying SMOTE, Random Forest is the best performing model with:

- Accuracy: 80.17%
- ROC-AUC: 0.886
- Balanced precision and recall for both classes.

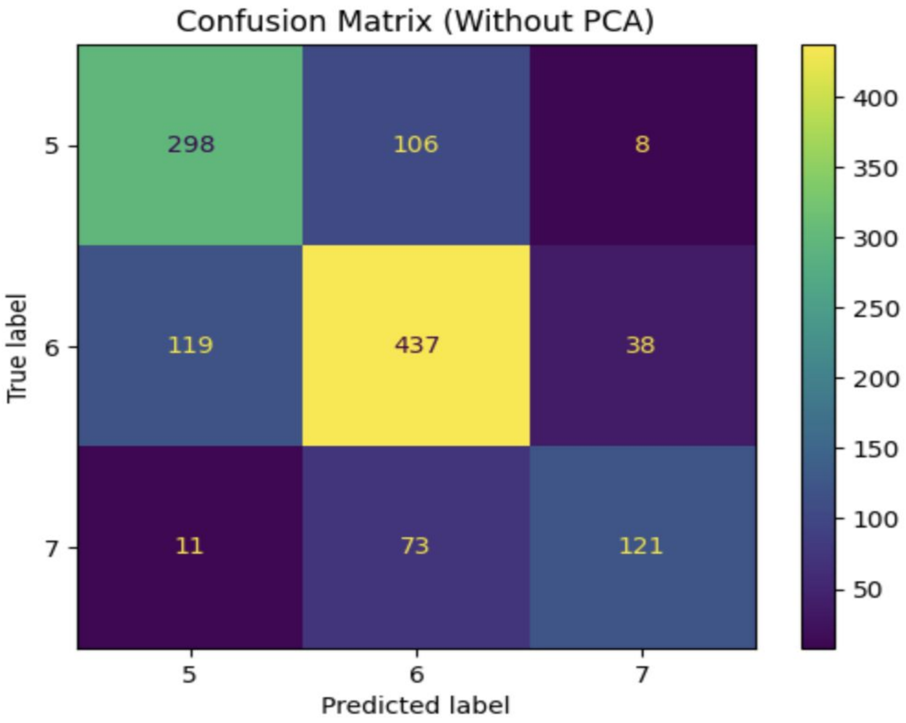
It outperforms Logistic Regression (accuracy = 75.13%, ROC-AUC = 0.819) and SVM (accuracy = 77.99%, ROC-AUC = 0.841) in terms of both accuracy and ROC-AUC, making it the most reliable model for this task.

PCA

After modelling and hyperparameter-tuning, it was now time to apply some dimensionality reduction techniques.

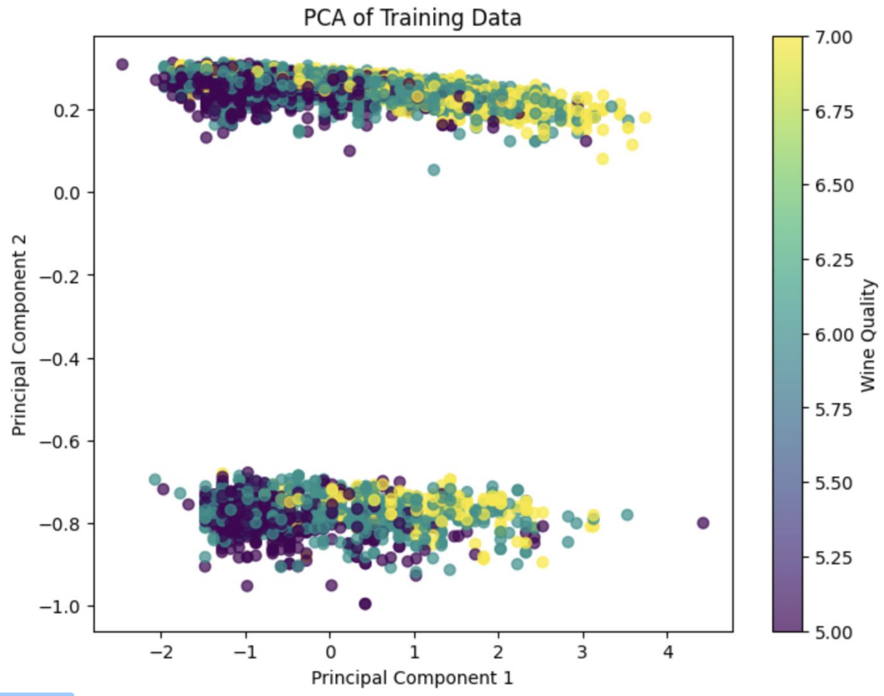
Heatmap

With top 5 features, selected using Correlation Analysis.



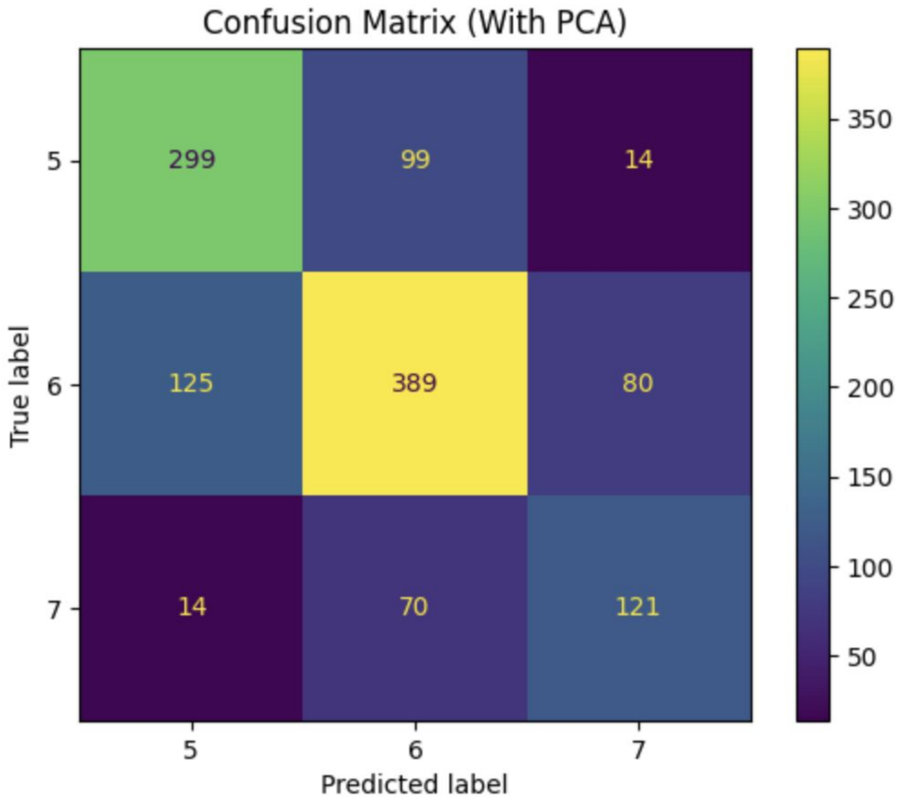
Employing RF

RF model with top 5 features comes out with an accuracy of 71%.



PCA

Applying PCA on the top 5 features and modelling the same RF model using the first 2 Principle Components



RF model on 2 PCs

RF model with top 2 PCs comes out with an accuracy of 67%, which is lower.

Potential reasons for decline in Accuracy

1. Loss of Information.
2. Misaligned Components.
3. Oversimplification.
4. Scaling issues.
5. Noise amplification.

In conclusion, while Principal Component Analysis (PCA) is a powerful technique for dimensionality reduction, its effectiveness is highly context-dependent. In our case, applying PCA did not yield the desired results and, in fact, led to a reduction in model accuracy. This outcome highlights the importance of carefully evaluating the impact of dimensionality reduction methods on specific datasets and models, as their applicability may vary based on the underlying data structure and the problem at hand.

Conclusion

Key physicochemical properties, such as alcohol content and volatile acidity, showed significant correlations with wine quality. Among the models tested, Random Forest was the most effective, outperforming Logistic Regression and SVM in prediction accuracy. While PCA is a powerful dimensionality reduction technique, its application in this case reduced model accuracy, highlighting the importance of evaluating its impact based on the dataset and specific problem context.

Contributions and References

- Whole Team - Data Acquisition
- Data Cleaning, EDA and Results - Sai & Srija
- Data Modeling & Reporting - Srija & Aakiff

References -

[1] Dahal, K. R., Dahal, J. N., Banjade, H., & Gaire, S. (2021). Prediction of Wine Quality Using Machine Learning Algorithms. Open Journal of Statistics, 11(2), 278–289. https://doi.org/10.4236/ojs.2021.112015

[2] 1.4. Support Vector Machines. (2024). Scikit-Learn. https://scikit-learn.org/1.5/modules/svm.html

[3] Brownlee, J. (2020, April 7). 4 Types of Classification Tasks in Machine Learning. Machine Learning Mastery. https://machinelearningmastery.com/types-of-classification-in-machine-learning/

THANKS!