Red Wine Quality

ABSTRACT-

This study leverages the Wine Quality Dataset from the UC Irvine Repository to predict red wine quality using Random Forest and SVM algorithms, focusing on 11 physiochemical properties. It demonstrates the effectiveness of machine learning in classifying wine quality, with the Random Forest model showing superior performance, highlighting the potential of data-driven approaches in viticulture.

Keywords- Red Wine Quality, Machine Learning Techniques, Random Forest Algorithm, Support Vector Machine (SVM), UC Irvine Machine Learning Repository, Wine Quality Dataset, Physiochemical Properties, Classification of Wine Quality, Hyperparameter Tuning, Model Evaluation and Performance

I. INTRODUCTION

This report explores the intricate relationship between the chemical composition of red wine and its perceived quality, utilizing the Wine Quality Dataset from the UC Irvine Machine Learning Repository. By applying advanced machine learning techniques, specifically Random Forest and Support Vector Machine algorithms, the study aims to classify wine quality based on 11 distinct physiochemical properties. This research not only offers insights into the factors that contribute to wine quality but also demonstrates the potential of data-driven approaches in enhancing wine production and quality assessment in the viticulture industry.

II. PROBLEM STATEMENT

Problem address in this research is the classification of wine quality based on various chemical attributes. Utilizing a dataset with 11 chemical features, the challenge is to accurately predict wine quality on a scale of 3 to 8. This endeavor bridges the gap between subjective taste and objective analysis, with significant implications for the wine industry's production and quality assessment standards.

III. DATA COLLECTION

Dataset used is the Wine Quality Dataset is available on UC Irvine Machine Learning repository[1]. It consist of 11 different features representing different chemical properties along with one target value ‘quality’ that depends on values of chemical properties. Target variable for this problem is Quality who’s value range from 3 to 8. Data does not have any missing value. Below is the class distribution of Quality attribute.

Quality levels	No. of entries
5	681
6	638
7	199
4	53
8	18
3	10

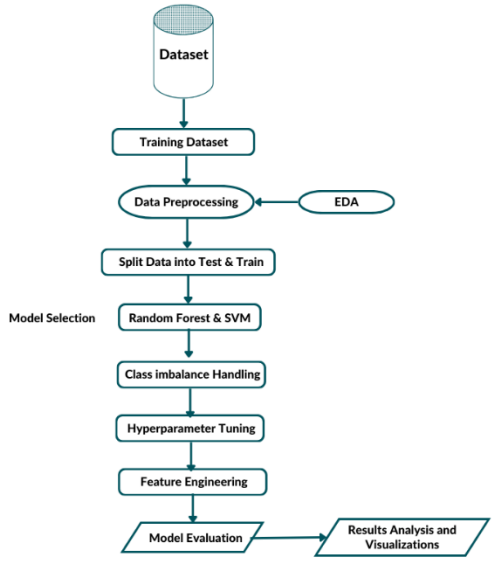
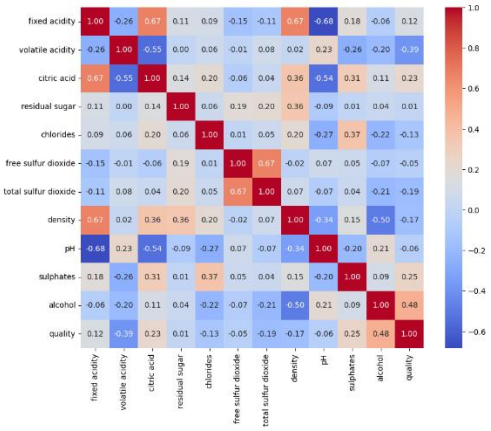


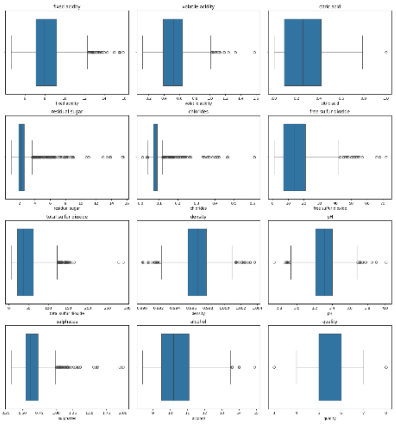
Fig. 1. Machine Learning Workflow [4]

IV. DATA EXPLORATION

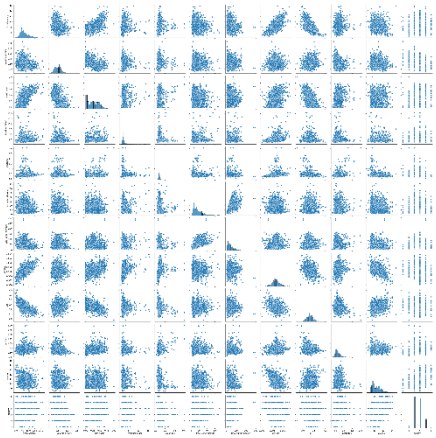


This Fig. shows correlation plot

Positive High Correlation between Alcohol and Quality of wine.



This Fig. shows outliers in wine dataset.



This Fig. shows pair plot of wine dataset

V. Algorithms and Motivation

Solving the wine quality classification problem, two powerful and widely used algorithms, Random Forest and Support Vector Machine (SVM), were selected for model development.

- Random Forest is an ensemble learning method that builds multiple decision trees during training and outputs the mode of the classes of the individual trees. Dataset contains multiple features representing chemical properties of wines. Random Forest provides a natural way to assess the importance of each feature in the classification process, aiding in feature selection and interpretation of results. Random Forest can effectively capture non-linear relationships in the data, making it well-suited for datasets with complex patterns that may not be linearly separable the reason for selecting this model is that random forest can handle non-linear relationships as shown in the pair plot and the second reason is its robustness to overfitting.[2].
- Support Vector Machine (SVM) is a powerful supervised learning algorithm commonly used for classification tasks. The decision boundary created by SVM aims to maximize the margin between different classes, making it suitable for datasets with distinct class separations. The dataset contains multiple features, and SVM is known for its effectiveness in high-dimensional spaces. It is less sensitive to outliers in the data. One of the reason for selecting SVM is its effectiveness in high dimensional spaces [3].

VI. Experimental Setup

In this experiment, the 'winequality-red.csv' dataset was divided into training and testing sets, ensuring that the models were trained on a subset of the data and evaluated on unseen data to gauge their generalization capabilities.

- **Data Preparation**: The data was preprocessed, checked for missing values in the dataset, and standardized. The outliers were also removed using mean value replacement except for non-numeric data and "quality" column because quality column is for classification of wines and non-numeric data is categorical data. The features were also standardized using Standard Scaling to ensure an equal contribution to the model training by normalizing them to a common scale. Additionally, the class weights were adjusted, and the data was split into features and a target variable ('quality').

Model Training and Evaluation:

- **Random Forest Classifier**: The model was first trained with its default parameters. To handle potential class imbalances, the class weights were adjusted to 'balanced'. GridSearchCV was then used for hyperparameter tuning to optimize the model's performance. A new feature was engineered from existing data to explore its impact on model performance.
- **Support Vector Machine (SVM)**: An SVM model with balanced class weights was trained. Similar to the Random Forest, this model also underwent hyperparameter tuning using GridSearchCV. The impact of the newly engineered feature on the SVM model's performance was evaluated.

Evaluation Metrics: The primary metric for evaluating the classifiers was accuracy. This metric was chosen for its ability to provide a balanced assessment of the model's performance across different quality classes, aligning with the objective of predicting overall wine quality. The classification report, including precision, recall, and F1-score, was used for a more detailed assessment.

Visualization and Analysis: For the Random Forest model, a feature importance plot was generated to visualize the relative importance of each feature in the dataset. Confusion matrices were also generated for both models to visually assess their performance in accurately classifying each class.

VII. Model Evaluation Results

The evaluation of the Random Forest and SVM models on the wine quality dataset yielded the following results:

Random Forest Model:

Initial Model:

Accuracy: 0.659375

Classification Report:

	precision	recall	f1-score	support
3	0.00	0.00	0.00	1
4	0.00	0.00	0.00	10
5	0.72	0.75	0.73	130
6	0.63	0.69	0.66	132
7	0.63	0.52	0.57	42
8	0.00	0.00	0.00	5
accuracy			0.66	320
macro avg	0.33	0.33	0.33	320
weighted avg	0.63	0.66	0.64	320

Accuracy: 65.9375%.

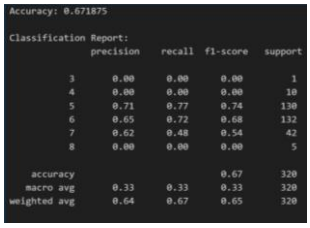
Performance: Showed moderate precision and recall, particularly effective in classes 5 and 6.

Class Breakdown:

Class 5: TP = 98, FP = 39, FN = 32, TN = 151.

Class 6: TP = 91, FP = 54, FN = 41, TN = 134.

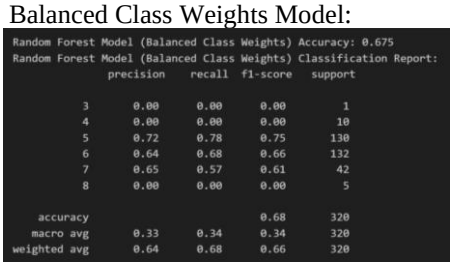
After Removing Outliers:



	precision	recall	f1-score	support
3	0.00	0.00	0.00	1
4	0.00	0.00	0.00	10
5	0.71	0.77	0.74	130
6	0.65	0.72	0.68	132
7	0.62	0.48	0.54	42
8	0.00	0.00	0.00	5
accuracy			0.67	320
macro avg	0.33	0.33	0.33	320
weighted avg	0.64	0.67	0.65	320

Accuracy: 67.1875%.

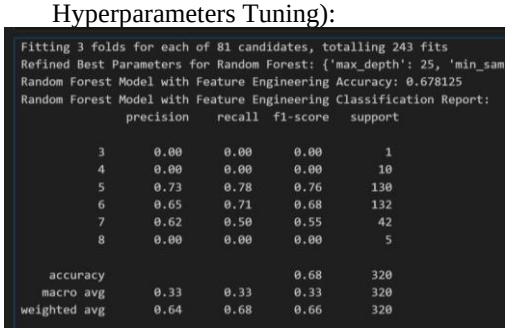
- Balanced Class Weights Model:



	precision	recall	f1-score	support
3	0.00	0.00	0.00	1
4	0.00	0.00	0.00	10
5	0.72	0.78	0.75	130
6	0.64	0.68	0.66	132
7	0.65	0.57	0.61	42
8	0.00	0.00	0.00	5
accuracy			0.68	320
macro avg	0.33	0.34	0.34	320
weighted avg	0.64	0.68	0.66	320

Accuracy: 67.5%.
Performance: Slight improvements, especially in recall for classes 5 and 6.

- Refined Model (Feature Engineering and Hyperparameters Tuning):

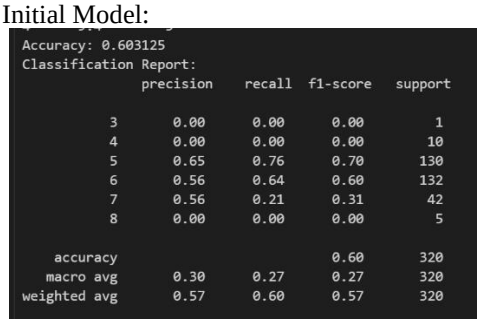


	precision	recall	f1-score	support
3	0.00	0.00	0.00	1
4	0.00	0.00	0.00	10
5	0.73	0.78	0.76	130
6	0.65	0.71	0.68	132
7	0.62	0.50	0.55	42
8	0.00	0.00	0.00	5
accuracy			0.68	320
macro avg	0.33	0.33	0.33	320
weighted avg	0.64	0.68	0.66	320

Accuracy: 67.812%.
Hyperparameters: max_depth = 25, min_samples_leaf = 1, min_samples_split = 3, n_estimators = 480.
Performance: Improved precision and recall for some classes; significant performance in class 5 (TP = 98) and class 6 (TP = 91).

SVM Model:

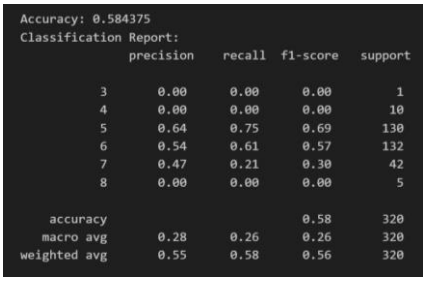
- Initial Model:



	precision	recall	f1-score	support
3	0.00	0.00	0.00	1
4	0.00	0.00	0.00	10
5	0.65	0.76	0.70	130
6	0.56	0.64	0.60	132
7	0.56	0.21	0.31	42
8	0.00	0.00	0.00	5
accuracy			0.60	320
macro avg	0.30	0.27	0.27	320
weighted avg	0.57	0.60	0.57	320

Accuracy: 60.3125%.
Performance: Moderate precision and recall; struggled with less represented classes.

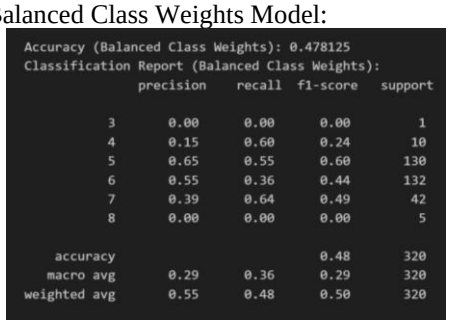
After Removing Outliers:
Initial Model:



	precision	recall	f1-score	support
3	0.00	0.00	0.00	1
4	0.00	0.00	0.00	10
5	0.64	0.75	0.69	130
6	0.54	0.61	0.57	132
7	0.47	0.21	0.30	42
8	0.00	0.00	0.00	5
accuracy			0.58	320
macro avg	0.28	0.26	0.26	320
weighted avg	0.55	0.58	0.56	320

Accuracy:58.43%

- Balanced Class Weights Model:

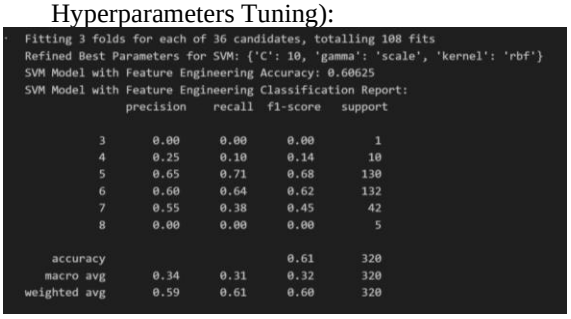


	precision	recall	f1-score	support
3	0.00	0.00	0.00	1
4	0.15	0.60	0.24	10
5	0.65	0.55	0.60	130
6	0.55	0.36	0.44	132
7	0.39	0.64	0.49	42
8	0.00	0.00	0.00	5
accuracy			0.48	320
macro avg	0.29	0.36	0.29	320
weighted avg	0.55	0.48	0.50	320

Accuracy: 47.812%.

Performance: Increase in recall for minority classes, although overall accuracy decreased.

- Refined Model (Feature Engineering & Hyperparameters Tuning):



	precision	recall	f1-score	support
3	0.00	0.00	0.00	1
4	0.25	0.10	0.14	10
5	0.65	0.71	0.68	130
6	0.60	0.64	0.62	132
7	0.55	0.38	0.45	42
8	0.00	0.00	0.00	5
accuracy			0.61	320
macro avg	0.34	0.31	0.32	320
weighted avg	0.59	0.61	0.60	320

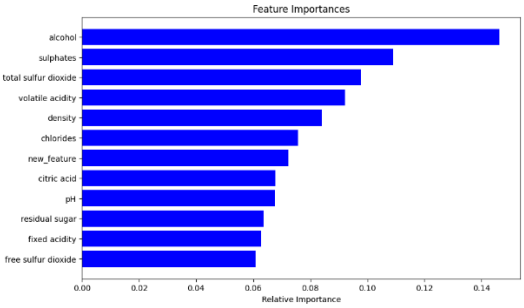
Accuracy: 60.3125%.
Hyperparameters: C = 10, gamma = 'scale', kernel = 'rbf'.

Performance: Similar to the initial model, with slight improvements. Notably, class 5 and 6 showed relatively better precision and recall.
Both models demonstrated strengths and weaknesses, with the Random Forest model showing a better balance in accuracy and class-wise performance, particularly after hyperparameter tuning and balancing class weights. The SVM model, while effective in certain aspects, faced challenges in handling class imbalances, which was partially addressed with balanced class weights. The feature engineering step contributed to the refined analysis in both models,

although its impact was more pronounced in the Random Forest model.

VIII. Visualizations

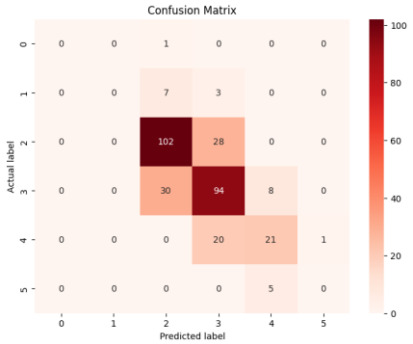
Feature Importance Plot:



This plot is generated to show the importance of each feature in a red wine dataset.

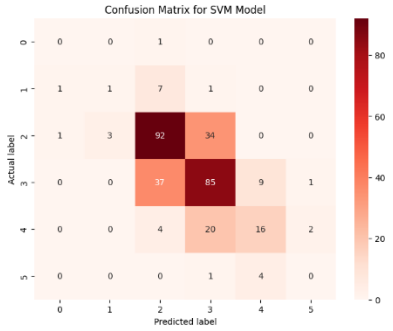
- **Alcohol (0.1463)** is the most important feature, suggesting it has the greatest impact on the model's predictions.
- **Sulphates (0.1090)** and **Total Sulfur Dioxide (0.0977)** are also significant, but less so than alcohol.
- Features like **Free Sulfur Dioxide (0.0608)** and **Fixed Acidity (0.0628)** have some impact but are less important compared to others.

Confusion Matrix Plot:
Random Forest:



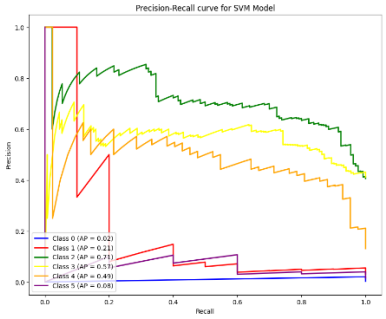
- True Positives (TP): 217
- True Negatives (TN): 0
- False Positives (FP): 102
- False Negatives (FN): 49

Confusion Matrix Plot:
SVM:



- True Positives (TP): 194
- True Negatives (TN): 0
- False Positives (FP): 46
- False Negatives (FN): 62

Precision-Recall curve:
SVM:



IX. Summary of the Project

The project involved developing machine learning models to predict the quality of red wine based on various chemical attributes. Utilizing the Wine Quality Dataset from the UC Irvine Machine Learning repository, two models were employed: Random Forest and Support Vector Machine (SVM). These models were chosen for their robustness in handling complex datasets. The dataset was preprocessed, including standard scaling of features, and the models were trained and evaluated using accuracy, precision, recall, and F1-score. Both models underwent hyperparameter tuning, and a new feature was engineered to assess its impact on performance. The analysis concluded that the Random Forest model, particularly with balanced class weights, generally outperformed the SVM in predicting wine quality.

Limitations

1. **Class Imbalance**: A significant imbalance in the dataset's class distribution likely impacted the models' ability to predict minority classes effectively.
2. **Performance on Minority Classes**: Both models struggled with accurate predictions for less represented quality classes, indicating a need for better handling of class imbalances.
3. **Feature Dynamics**: The specific influence of each chemical attribute on different quality levels was not deeply explored, which could provide further insights into predictive patterns.
4. **Risk of Overfitting**: The complexity of the models, especially the Random Forest, poses a risk of overfitting, potentially compromising their performance on new datasets.

Improvement Strategies

1. **Handling Class Imbalance**: Implementing techniques like SMOTE or further adjusting class weights could improve model performance on minority classes.
2. **Enhanced Feature Engineering**: More in-depth feature engineering might reveal additional predictive relationships within the data.
3. **Broader Model Exploration**: Experimenting with different models or ensemble techniques

could uncover more effective predictive strategies.

4. **Enhanced Validation Techniques**: Employing rigorous cross-validation and regularization methods would help in preventing overfitting and ensuring the models' generalization ability.

Lessons Learned

1. **Complex Nature of Wine Quality Prediction**: The project highlighted the intricate challenge of predicting wine quality based on chemical attributes, emphasizing the complexity inherent in such tasks.
2. **Critical Role of Data Preprocessing**: The importance of thorough data preprocessing, particularly in dealing with class imbalances and feature scaling, was a key takeaway.
3. **Importance of Model Selection and Tuning**: The project underscored the need for careful model selection and hyperparameter tuning in handling complex, imbalanced datasets.
4. **Relevance of Evaluation Metrics**: It stressed the importance of selecting appropriate evaluation metrics that accurately reflect the model's performance in the context of an imbalanced dataset.

X. Self Evaluation

If the study were to be redone, the focus would be on addressing class imbalances using techniques like SMOTE, exploring advanced feature engineering, experimenting with a broader range of models to improve predictions, especially for minority classes, and implementing rigorous cross-validation to prevent overfitting and enhance model generalization. Additionally, a deeper analysis of the impact of individual chemical attributes on wine quality would be conducted for more precise insights.

XI. Conclusion

In conclusion, the analysis of the wine quality prediction models reveals that Random Forest models, particularly when trained with balanced class weights, outperform Support Vector Machines in predicting wine quality based on various chemical attributes. Feature engineering, along with hyperparameter tuning, moderately enhances model performance, although challenges persist in accurately classifying certain wine quality classes. Despite efforts to address imbalanced classes, SVM models with balanced weights exhibit suboptimal accuracy. Recommendations include further exploration of feature engineering techniques, continued hyperparameter tuning, and addressing data imbalances to optimize model performance for the specific task of wine quality prediction.

XII. References

[1] UCI Machine Learning Repository. (n.d.). "Wine Quality Data Set." [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Wine+Quality.

[2] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.

[3] C. Cortes and V. Vapnik, "Support-Vector Networks," Machine Learning, vol. 20, no. 3, pp. 273–297, 1995.

[4] A. C. Müller and S. Guido, "Introduction to Machine Learning with Python: A Guide for Data Scientists," O'Reilly Media, 2017.