Wine Quality Prediction

Arnav Agrawal
IIIT Delhi
arnav22097@iiitd.ac.in

Aryan Singla
IIIT Delhi
aryan22112@iiitd.ac.in

Harsh Rajput
IIIT Delhi
harsh22201@iiitd.ac.in

Kshitij Gupta
IIIT Delhi
kshitij22257@iiitd.ac.in

Abstract

The purpose of this project is to develop a machine learning pipeline to predict the quality of wines based on physicochemical features. Wine quality prediction is an important problem for producers, as it allows for quality assurance and consistency. In this project, we aim to build a model that can predict wine quality with high accuracy using various regression and classification techniques. Additionally, we plan to explore the usage of machine learning operations (MLOps) techniques to streamline the workflow and manage the model lifecycle.

1. Introduction

Wine quality assessment is an important factor in maintaining consistency and value in the wine industry. Accurate prediction of wine quality, based on physicochemical features, helps wine producers ensure that their product meets certain standards. Traditional methods of wine quality assessment are time-consuming and subjective. This project aims to develop an automated machine learning pipeline to predict wine quality based on features such as acidity, alcohol content, and residual sugar.

Wine quality is rated on a discrete scale from 3 to 8, making this an ordinal classification problem. The objective is to use various machine learning techniques, including regression and classification models, to predict these scores. Additionally, this project explores MLOps practices for managing the model lifecycle.

2. Related Works

Previous studies on wine quality prediction have applied a variety of machine learning models [1, 2]. Methods like linear regression, decision trees, and logistic regression have been explored, with non-linear models such as Random Forest and Gradient Boosting consistently outperforming linear models. These ensemble methods capture complex feature interactions and non-linearities, which are essential for improving prediction accuracy.

Recent advances in MLOps practices have provided tools for automating model deployment, tracking experiments, and ensuring reproducibility. This project integrates these practices to streamline the model lifecycle.

3. Dataset

3.1. Dataset Acquisition

The Wine Quality dataset was sourced from the Kaggle Machine Learning Repository. It contains physicochemical features of red and white wine samples, with quality scores ranging from 3 to 8. The dataset contains 12 attributes related to the wine's composition, including fixed acidity, volatile acidity, residual sugar, chlorides, and alcohol content.

3.2. Data Preprocessing

Initial exploration revealed no missing values in the dataset. The features were standardized using **StandardScaler** to ensure equal contribution to the models. Since the dataset showed a class imbalance, where most wines were rated between 5 and 6, we applied **SMOTE** (Synthetic Minority Over-sampling Technique) to generate synthetic samples for the minority classes and balance the dataset.

4. Methodology and Model Details

4.1. Models Used

We implemented both regression and classification models to predict wine quality. However, wine quality prediction is inherently a classification problem, as the target variable is a discrete quality score (3 to 8). Below is a summary of the models used:

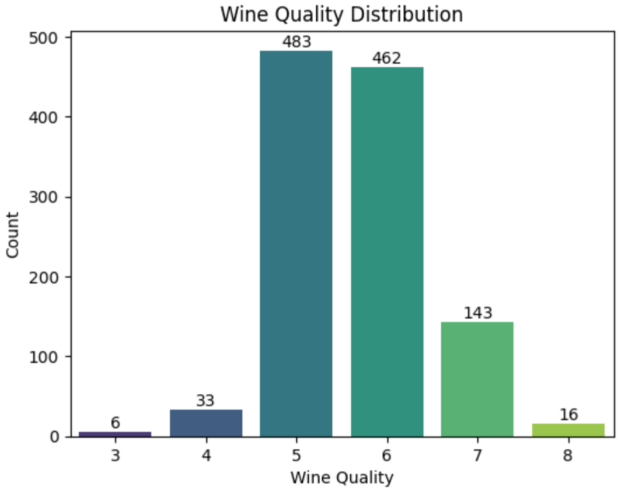


Figure 1. Class imbalance in the dataset.

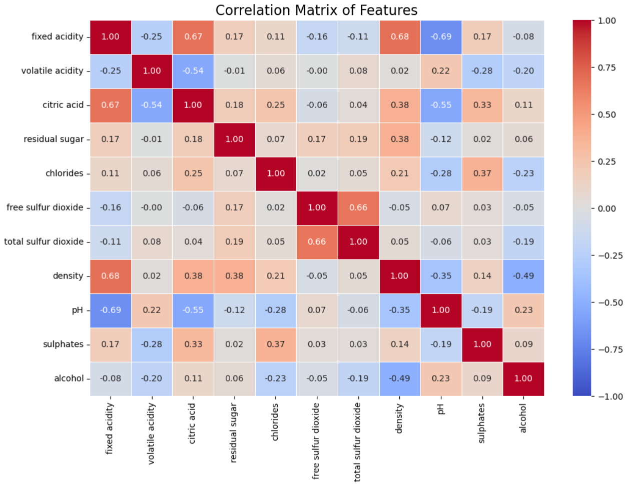


Figure 2. Correlation matrix of the dataset.

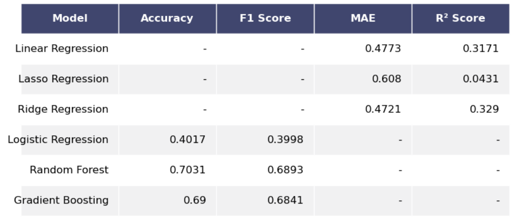
- **Linear Regression**: Predicts continuous approximations of wine quality scores. This method is suboptimal because it predicts continuous values that must be rounded, introducing errors.

- **Ridge and Lasso Regression**: Add regularization to avoid overfitting, but face the same limitations as Linear Regression for classification tasks.

- **Logistic Regression**: A linear classifier that models the probability of each class but struggles with non-linear relationships.

- **Random Forest**: An ensemble method that builds multiple decision trees to capture non-linear relationships in the data. Performs well due to its ability to model complex feature interactions.

- **Gradient Boosting**: Builds decision trees sequentially, correcting errors from the previous model. Provides high accuracy, comparable to Random Forest.



Model	Accuracy	F1 Score	MAE	R² Score
Linear Regression	-	-	0.4773	0.3171
Lasso Regression	-	-	0.608	0.0431
Ridge Regression	-	-	0.4721	0.329
Logistic Regression	0.4017	0.3998	-	-
Random Forest	0.7031	0.6893	-	-
Gradient Boosting	0.69	0.6841	-	-

Figure 3. Comparison of model performance.

4.2. Regression Models on Classification Problems

Regression models like **Linear Regression**, **Ridge Regression**, and **Lasso Regression** were applied, but their continuous predictions required rounding, leading to poor classification results.

Why Regression Fails:

- **Continuous Output**: Regression models predict continuous values that must be rounded, introducing errors.

- **Lack of Decision Boundaries**: Regression models do not define clear boundaries between classes, which is crucial for classification tasks.

- **Linear Assumptions**: These models assume linear relationships between features and target, which is not the case with wine quality data.

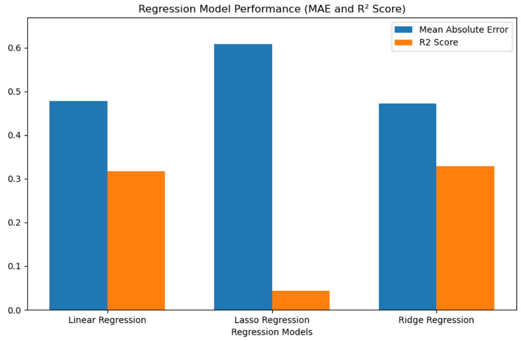


Figure 4. Regression models perform poorly on classification tasks.

4.3. Classification Model Results

Classification models performed significantly better, with **Random Forest** and **Gradient Boosting** outperforming logistic regression due to their ability to capture non-linear patterns.

Classification Model Performance:

- **Logistic Regression**: Accuracy = 40.17%, F1 Score = 0.3998.
- **Random Forest**: Accuracy = 70.31%, F1 Score = 0.6893.
- **Gradient Boosting**: Accuracy = 69.00%, F1 Score = 0.6841.

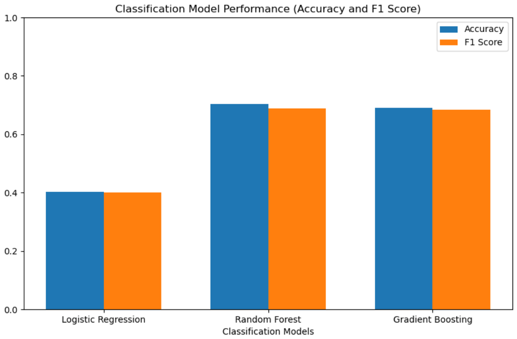


Figure 5. Performance of classification models.

4.4. MLOps Pipeline

We implemented an MLOps pipeline to manage the model lifecycle, including data ingestion, preprocessing, model training, and evaluation. Experiment tracking was handled through version control, ensuring reproducibility. In future work, we aim to containerize the models using Docker for easier deployment and scaling.

5. Results and Analysis

5.1. Regression Models

Regression models struggled with predicting discrete quality scores. After rounding, the performance was significantly worse compared to classification models.

- **Linear Regression**: $R^2 = 0.3171$, Mean Squared Error = 0.4773.
- **Lasso Regression**: R^2 = 0.0431, Mean Squared Error = 0.6080.
- **Ridge Regression**: R^2 = 0.3290, Mean Squared Error = 0.4721.

5.2. Classification Models

- **Logistic Regression**: Accuracy = 40.17%, F1 Score = 0.3998.
- **Random Forest**: Accuracy = 70.31%, F1 Score = 0.6893.
- **Gradient Boosting**: Accuracy = 69.00%, F1 Score = 0.6841.

6. Conclusion

In this project, we explored both regression and classification models for predicting wine quality. Classification models, particularly ensemble methods like Random Forest and Gradient Boosting, performed significantly better than regression models. This is because the classification models are better suited to handle non-linear relationships between features and discrete target classes, which is essential for predicting wine quality effectively.

We successfully implemented a basic MLOps pipeline to manage the model lifecycle, covering data ingestion, preprocessing, model training, and evaluation. The initial experiment tracking was also integrated with version control, ensuring reproducibility and experiment transparency.

6.1. Future Work

In the future, we aim to implement a full end-to-end MLOps pipeline to automate and streamline the machine learning workflow. This includes the following:

- **Hyperparameter Tuning**: Applying advanced techniques like grid search and randomized search to optimize model hyperparameters for the best performance.
- **Model Deployment**: Containerizing the models using Docker and deploying them to cloud environments (such as AWS, GCP, or Azure) for scalability and accessibility.
- **Continuous Integration/Continuous Deployment (CI/CD)**: Setting up automated pipelines for retraining, testing, and deploying models based on new data or performance degradation, ensuring that the model stays up-to-date and efficient.
- **Model Monitoring**: Implementing real-time model monitoring to track the performance of models in production, enabling quick response to performance drops or data drift.
- **Data Versioning and Experiment Tracking**: Using tools like DVC (Data Version Control) and MLflow to maintain organized version control for datasets and experiments, further enhancing reproducibility and collaboration.

By building a fully automated and scalable MLOps framework, we can continuously improve the model's accuracy and reliability while minimizing manual intervention and reducing time-to-deployment.

6.2. Contribution

- **Arnav Agrawal:** EDA, Model implementation
- **Aryan Singla:** EDA, Inference
- **Harsh Rajput:** Preprocessing, Model implementation
- **Kshitij Gupta:** Preprocessing, Inference

7. References

References

[1] K. R. Dahal, J. N. Dah-al, H. Banjade, and S. Gaire. Prediction of wine quality using machine learning algorithms. *Open Journal of Statistics*, 11:278–289, 2021.

[2] K. Jain, K. Kaushik, S.K. Gupta, et al. Machine learning-based predictive modelling for the enhancement of wine quality. *Scientific Reports*, 13(1):17042, 2023.