Wine Quality Prediction Model

I. Introduction

In this report, I developed and analyzed multiple machine learning models with the goal of accurately predicting wine quality using chemical data. The models that I tested include K-Nearest Neighbors and Decision Trees to identify the best-performing model, while also including a standard Linear Regression model. At the end of the report, I will compare all three final models against a baseline estimator to determine which performed most effectively on new wine data.

II. Data

This table below displays the first five rows of the dataset, which contains various chemical properties of different wines along with their quality ratings.

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality	color
0	7.6	0.23	0.64	12.9	0.033	54.0	170.0	0.99800	3.00	0.53	8.8	5	white
1	NaN	0.75	0.01	2.2	0.059	11.0	18.0	0.99242	3.39	0.40	NaN	6	red
2	7.4	0.67	0.12	1.6	0.186	5.0	21.0	0.99600	3.39	0.54	9.5	5	red
3	6.4	0.18	0.74	NaN	0.046	54.0	168.0	0.99780	3.58	0.68	10.1	5	white
4	6.7	0.35	0.32	9.0	0.032	29.0	113.0	0.99188	3.13	0.65	12.9	7	white

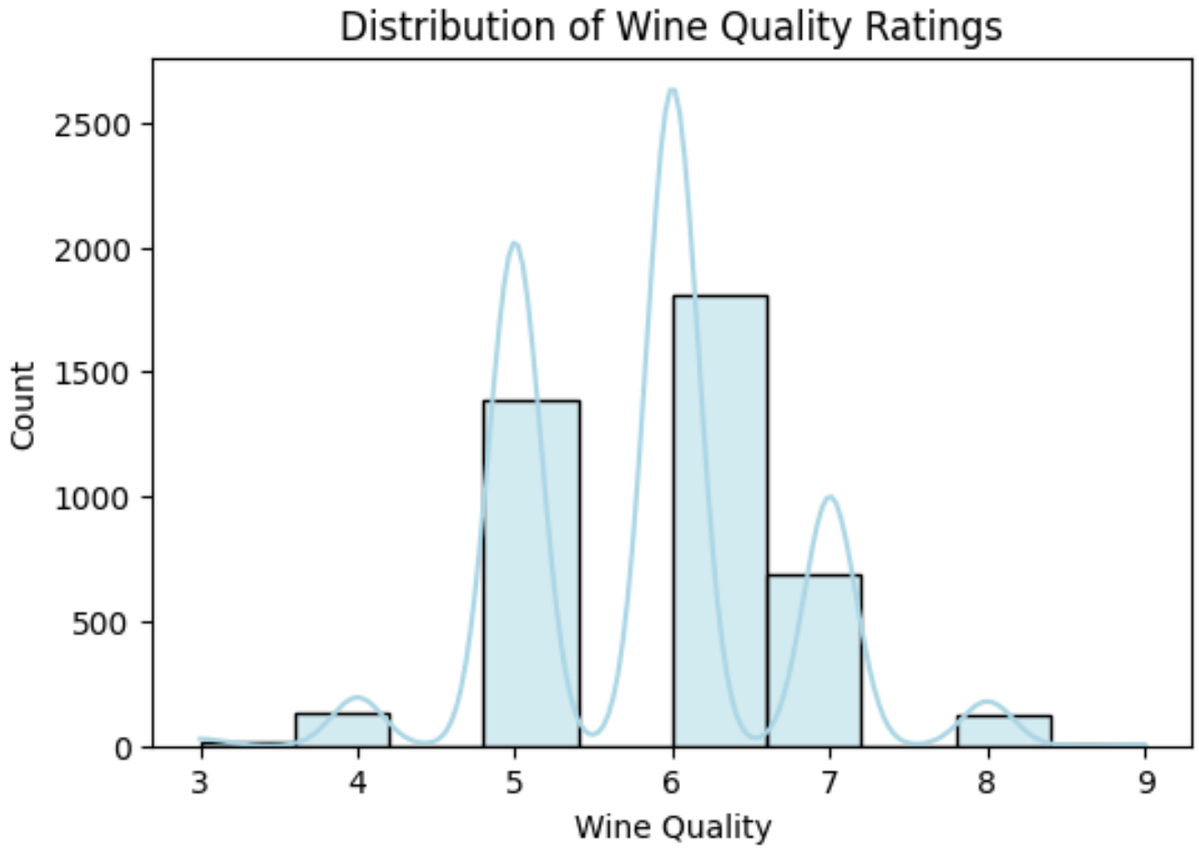
The target variable is `quality`, which is a score ranging from 0 to 10 that assesses the wine's overall quality. Each wine sample was rated by human blind-taste testers, who graded it from 0 (very bad) to 10 (excellent).

The feature variables include: `fixed acidity`, `volatile acidity`, `citric acid`, `residual sugar`, `chlorides`, `free sulfur dioxide`, `total sulfur dioxide`, `density`, `pH`, `sulphades`, `alcohol`, and `color`.

III. Exploratory Data Analysis

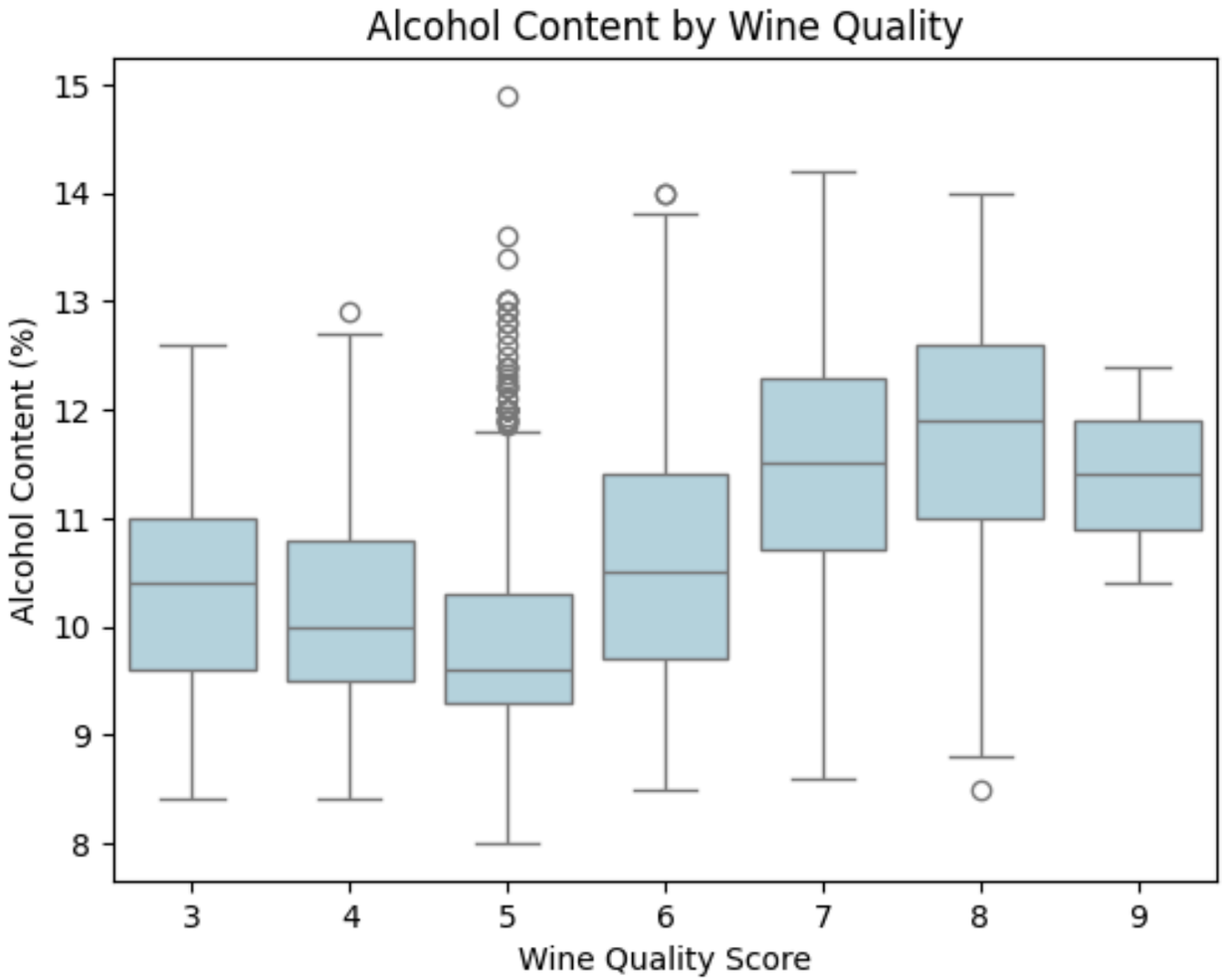
The bar plot and chart below display the distribution and proportion of wine quality ratings in the dataset. Something worth nothing is that the data has a significant class imbalance, with over 75% of the samples belonging to ratings 5 and 6. Unfortunately, there isn't better additional data available to address this issue, and like always, a more balanced dataset could improve the predictive performance of the models.

	Quality Rating	Count	Proportion
0	3	19	0.005
1	4	133	0.032
2	5	1385	0.333
3	6	1810	0.435
4	7	686	0.165
5	8	122	0.029
6	9	2	0.000



Alcohol Content / Wine Quality Graph

The box plot below visualizes the relationship between the alcohol content and wine quality scores. From the data, we can see a clear seperation between quality levels, and there even appears to be a potential upward trend, which might indicate that alcohol content may be a key variable used in predicting the wine's quality.



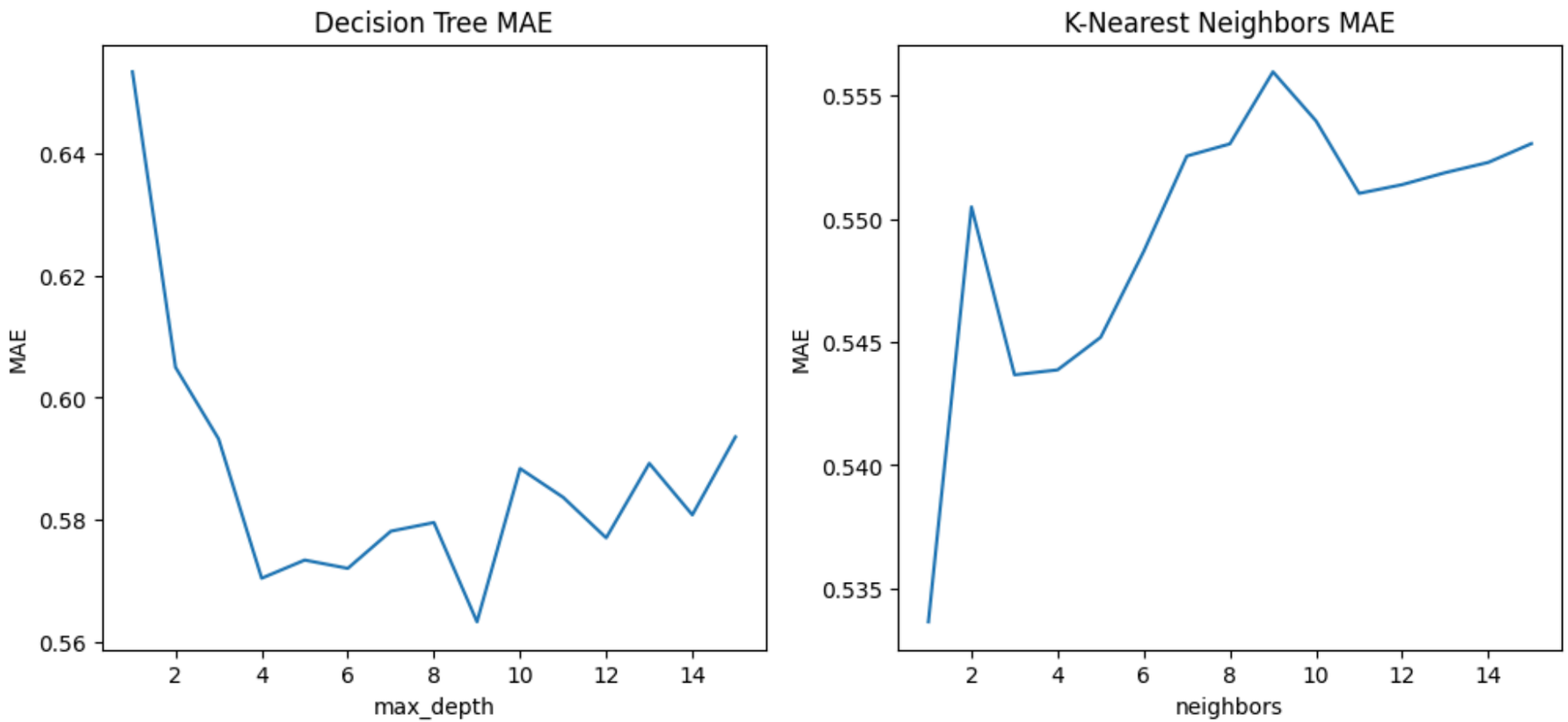
IV. Modeling

For Modeling, I began by preprocessing the data by encoding and imputing my categorical variables, as well as scaling the numeric variables to ensure that the data could be used efficiently for machine learning. Next, I trained K-Nearest Neighbors

and Decision Tree models, looping through their tuning parameters with k = 1 though k = 15, and evaluated how well they performed by calculating the `Mean Absolute Error (MAE)` on validation test sets.

Additionally, I trained a simple Linear Regression model without feature engineering, which performed worse than both the KNN and Decision Tree model.

The graph below shows the overall results, where we can see that the K-Nearest Neighbors model with n_neighbors = 1 was the top performing model and achieved a MAE of `0.5288`



Baseline Model

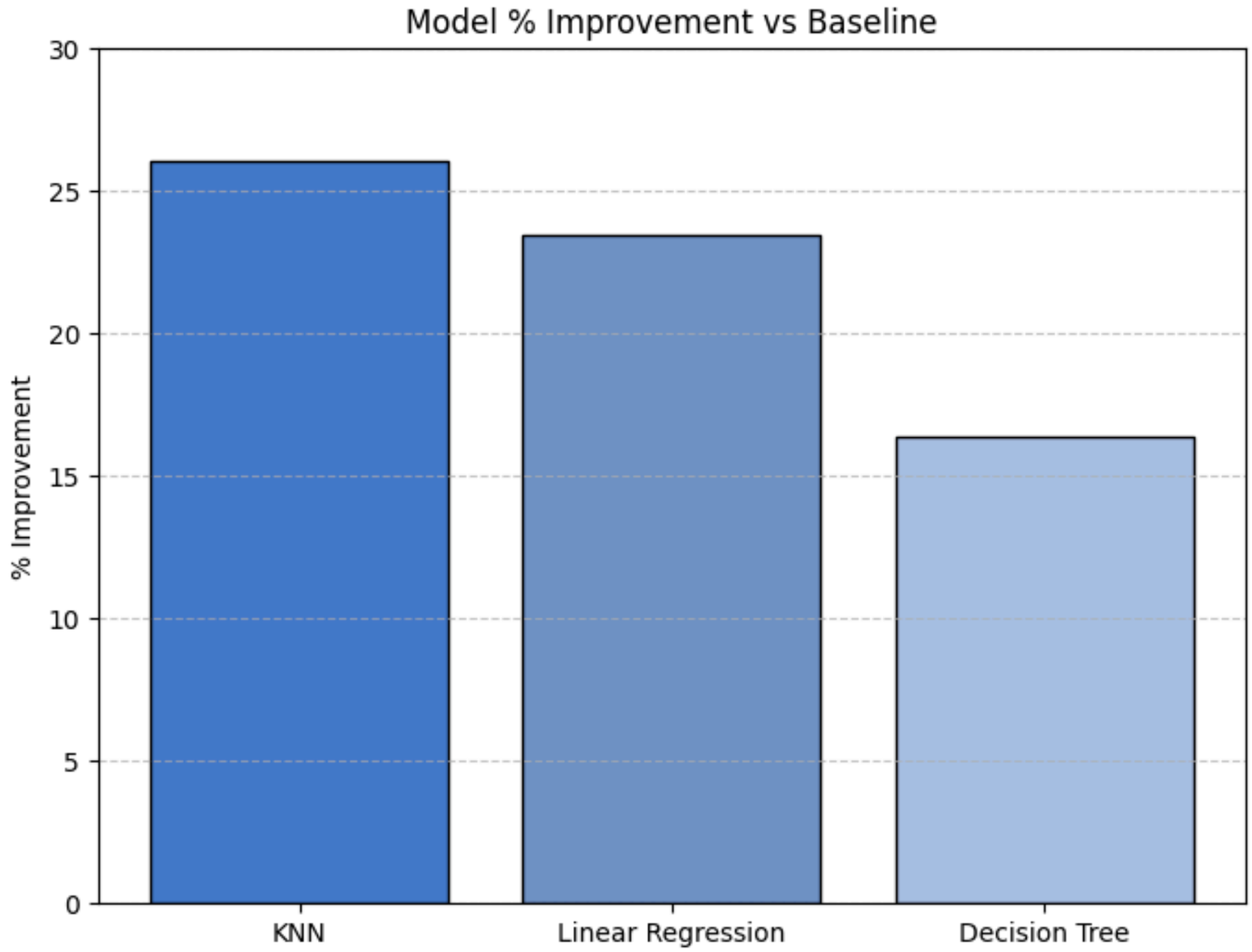
After testing the model performance, I trained a `DummyRegressor()` model to establish a baseline MAE that could be used to assess the genuine improvement in predictive accuracy. As shown below, the DummyRegressor() model simply predicts the mean of the overall data for every value. This model had a MAE of `0.7149`, which is fortuately higher than the MAEs of the other models I tested, indicating that our tuned models performed significantly stronger than this baseline.

	DummyRegressor() Prediction
0	5.810827
1	5.810827
2	5.810827
3	5.810827
4	5.810827

V. Results

```
K-Nearest Neighbors Model Improvement over baseline model: 26.03%
Decision Tree Model Improvement over baseline model: 23.45%
Linear Regression Model Improvement over baseline model: 16.39%
```

All three of the best models outperformed the baseline, which shows improved predictive accuracy. The KNN model with n_neighbors = 1 achieved the highest performance, with a MAE that was 26.03% lower than the baseline MAE. The bar chart below summarizes the improvement for each model compared to the baseline.


Model % Improvement vs Baseline
% Improvement
30
25
20
15
10
5
0
KNN
Linear Regression
Decision Tree

VI. Conclusion

The goal of this project was successfully achieved, as I developed a K-Nearest Neighbors model that significantly outperformed the baseline.

However, a MAE of 0.5288 still means that the model's performance is far from perfect. This means that on average, the predictions for the wine quality were off by roughly .5288.

I think that the biggest reason for this is because of the datasets imbalance, with over 75% of the samples having a quality score of 5 or 6. This caused a limitation which likely caused our model to have a hard time predicting more extreme values.

While the model is useful and works as a valuable tool for assessing wine quality, I don't believe that it's good enough to fully replace a wine steward.

In the future, I would love to revisit this project to test out more advanced models, such as Random Forests or Gradient Boosting to see how far I can improve predictive accuracy!