WINE QUALITY ANALYSIS IN VINHO VERDE WINE

By

Paul Parks, Alden Caterio, Mayank Bhatt

AAI-500-02-SU23

Probability and Statistics for Artificial Intelligence

A REPORT

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

In Applied Artificial Intelligence

UNIVERSITY OF SAN DIEGO

SHILEY-MARCOS SCHOOL OF ENGINEERING

Summer 2023

Table of Contents

[Introduction 3](#_Toc138523226)

[Data Cleaning and Preparation 4](#_Toc138523227)

[Exploratory Data Analysis 5](#_Toc138523228)

[General Descriptive Statistics 5](#_Toc138523229)

[Quantifying Outliers 8](#_Toc138523230)

[Correlation 9](#_Toc138523231)

[Model Selection 10](#_Toc138523232)

[Model Analysis 10](#_Toc138523233)

[Red Wine Model Analysis 10](#_Toc138523234)

[White Wine Model Analysis 11](#_Toc138523235)

[Summary 12](#_Toc138523236)

[Conclusion 12](#_Toc138523237)

[Recommendation 13](#_Toc138523238)

[Appendix: Github 15](#_Toc138523239)

[Appendix: Code Output 15](#_Toc138523240)

[Datasets 15](#_Toc138523241)

[Boxplot all data to view outliers 17](#_Toc138523242)

[Correlation and Variation 21](#_Toc138523243)

[Probability of Scores 23](#_Toc138523244)

[Remove all outliers 24](#_Toc138523245)

[Generalized Linear Model Regression 25](#_Toc138523246)

[Predictions 30](#_Toc138523247)

[Bibliography 39](#_Toc138523248)

Table of Figures

[Figure 1: Boxplots for Red Wine Attributes 7](#_Toc138522808)

[Figure 2: Boxplots for White Wine Attributes 7](#_Toc138522809)

[Figure 3: Correlation Matrix for Red Wine Attributes 9](#_Toc138522810)

[Figure 4: Correlation Matrix for White Wine Attributes 10](#_Toc138522811)

[Figure 5: Red Wine Model Regression Results 11](#_Toc138522812)

[Figure 6: White Wine Model Regression Results 12](#_Toc138522813)

[Figure 7: Wine Simulated 1 Quality Increase 13](#_Toc138522814)

[Figure 8: Wine pH and alcohol summary 13](#_Toc138522815)

[Figure 9: Red Wine alcohol/pH quality correlation 14](#_Toc138522816)

[Figure 10: White Wine alcohol/pH quality correlation 14](#_Toc138522817)

Table of Tables

[Table 1: Physicochemical Attributes 5](#_Toc138522837)

[Table 2: General Descriptive Statistics 5](#_Toc138522838)

[Table 3: Total Outliers 8](#_Toc138522839)

# Introduction

Wine quality is known to be a very subjective topic. Wine taste, smell, and consistency will vary depending on many factors. Type of fruit, sugar content, age, temperature, pH level, additives, and more can affect how a glass of wine tastes to the consumer. Additionally, the consumer's taste pallet, aversion to alcohol content, and other external factors will further change how the wine is perceived. Pinpointing what "quality" truly means in terms of wine is a difficult task, but it is heavily studied for wine companies to produce higher quality products. Analyzing data gathered from wine studies can give insight into the subjectivity.

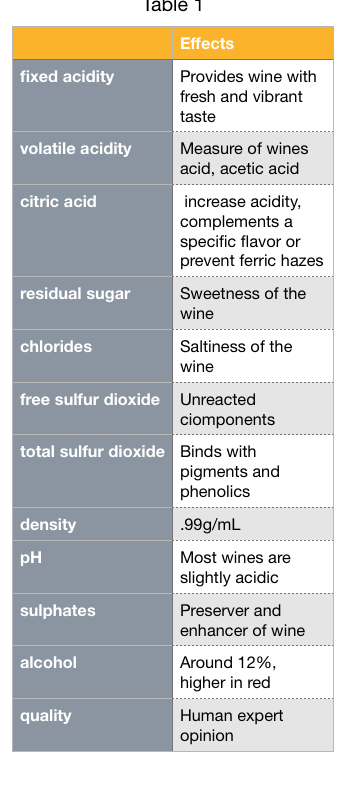
In this report, two datasets of wine from a popular Portuguese wine company called Vinho Verde are analyzed. Two types of wine are represented: red wine and white wine. They are separated since they are drastically different in taste and composition. Each dataset contains 12 attributes. There are 11 physicochemical attributes, measured and recorded by an official certification entity dealing with wine quality and marketing. The last attribute is Quality, attained through blind tastes, and rated on a scale from 0, meaning very bad, to 10, meaning excellent.

In our models, we focus on 2 attributes: pH and alcohol. The "pH" attribute quantifies the acidity or basic levels of the sample wine on a scale from 0 (very acidic) to 14 (very basic). The "alcohol" attribute describes how much alcohol is in the wine, measured in percent per volume. Through statistical inference, predictions about wine quality and the effects of these attributes can be made with a degree of confidence. Our models indicate that lowering pH levels while increasing alcohol content can lead to an increase in wine quality. Knowing this may be desirable for wine companies and their constituents because understanding the factors that increase overall wine quality can lead to improvements in production processes and profitability.

# Data Cleaning and Preparation

Wines are often identified by their physicochemical attributes and sensory test to characterize the overall composition. Physicochemical attributes such as fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, alcohol in relation to a quality sensory test by individuals determine the quality of a wine. Much of the raw data is gathered through physicochemical laboratory test to quantify each attribute. To develop accurate and precise analysis the raw data provided from these tests were cleaned by developing boxplots regarding each variable to remove any anomalies. Additionally, published papers regarding the same dataset state, “during the preprocessing stage, the database was transformed to include a distinct wine sample (with all tests) per row. To avoid discarding examples, only the most common physicochemical tests were selected”. Therefore, much of the data was optimized by researchers conducting the tests resulting in the raw dataset provided. By removing outliers from our data set we can interpret more accurate conclusion and develop an accurate model. The table below shows the significance of each physicochemical attribute on the composition of wine. Having a better understanding of these variables will allow for a better understanding of the data.

Table : Physicochemical Attributes



# Exploratory Data Analysis

## General Descriptive Statistics

The data to be analyzed consists of two different datasets: red wine and white wine. The datasets were divided into red and white wine due to the difference in taste and appearance. The red wine dataset contains 1599 samples, and the white wine dataset contains 4898 samples. In both datasets, there are 12 characteristics: 11 physicochemical attributes and 1 quality attribute. The physicochemical attributes are independent variables, and the quality attribute is the dependent variable. Table 1 shows general descriptive statistics of each dataset’s attributes.

Table 2: General Descriptive Statistics

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | RED WINE (1599 samples) | | | | White Wine (4898 samples) | | | |
|  | **Mean** | **Std** | **Min** | **Max** | **Mean** | **Std** | **Min** | **Max** |
| Fixed Acidity | 8.32 | 1.74 | 4.60 | 15.90 | 6.86 | 0.84 | 3.80 | 14.20 |
| Volatile Acidity | 0.53 | 0.18 | 0.12 | 1.58 | 0.28 | 0.10 | 0.08 | 1.10 |
| Citric Acid | 0.27 | 0.19 | 0.00 | 1.00 | 0.33 | 0.12 | 0.00 | 1.60 |
| Residual Sugar | 2.54 | 1.41 | 0.90 | 15.50 | 6.39 | 5.07 | 0.60 | 65.80 |
| Chlorides | 0.09 | 0.05 | 0.01 | 0.61 | 0.04 | 0.02 | 0.01 | 0.35 |
| Free Sulfur Dioxide | 15.87 | 10.46 | 1.00 | 72.00 | 35.08 | 17.01 | 2.00 | 289.00 |
| Total Sulfur Dioxide | 46.47 | 32.90 | 6.00 | 289.00 | 138.36 | 42.50 | 9.00 | 440.00 |
| Density | 1.00 | 0.002 | 0.99 | 1.00 | 0.99 | 0.003 | 0.99 | 1.03 |
| pH | 3.31 | 0.15 | 2.74 | 4.01 | 3.19 | 0.15 | 2.72 | 3.82 |
| Sulphates | 0.66 | 0.17 | 0.33 | 2.00 | 0.49 | 0.11 | 0.22 | 1.08 |
| Alcohol | 10.42 | 1.07 | 8.40 | 14.9 | 10.51 | 1.23 | 8.00 | 14.2 |
| Quality | 5.64 | 0.81 | 3.00 | 8.00 | 5.88 | 0.89 | 3.00 | 9.00 |

In the red wine dataset, the maximum value for a handful of attributes falls more than 3 standard deviations away from their respective mean, representing an outlier. The same is observed in the white wine dataset. “Outliers can have adverse effects on the model analysis if unaccounted for. Boxplots for both datasets are graphed and shown in Figure 1 and Figure 2 in order to visualize the general descriptive statistics and identify outliers that may affect the models.

In Figure 1, the red wine samples for fixed acidity, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, and sulphates are skewed to the right, possessing low-valued means and many outliers greater than the maximum values. Each box plot contains at least a few outliers. In Figure 2, the white wine samples for fixed acidity, volatile acidity, citric acid, chlorides, free sulfur dioxide, and sulphates seem skewed to the right. All attributes except alcohol contain outliers.

Figure : Boxplots for Red Wine Attributes

A picture containing text, diagram, parallel, plan

Description automatically generated

Figure : Boxplots for White Wine Attributes

A picture containing text, diagram, parallel, plan

Description automatically generated

## Quantifying Outliers

The boxplots reveal many outliers in certain attributes. Table 2 quantifies the outliers and the percentage relative to the total number of samples for each dataset. For red wine, the percentages are high for chlorides and sugar. For white wine, there are double-digit percentages in volatile acidity, citric acid, and chlorides. The high number of outliers increases variability which may decrease the statistical accuracy of our models. Techniques such as nonparametric tests and certain bootstrapping techniques can assist with handling outliers.

When comparing the outliers between both datasets, there are many outliers for residual sugar in red wine while there are only some in white wine. Additionally, there are many outliers in citric acid for white wine and very few in red wine. The difference in variability further confirms the need to separate the datasets.

Table 3: Total Outliers

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | RED WINE (1599 samples) | | WHITE WINE (4898 samples) | |
|  | **Total Outliers** | **Percentage** | **Total Outliers** | **Percentage** |
| Fixed Acidity | 49 | 3.06 | 119 | 7.44 |
| Volatile Acidity | 19 | 1.19 | 186 | 11.63 |
| Citric Acid | 1 | 0.06 | 270 | 16.89 |
| Residual Sugar | 155 | 9.39 | 7 | 0.44 |
| Chlorides | 112 | 7 | 208 | 13.01 |
| Free Sulfur Dioxide | 30 | 1.88 | 50 | 3.13 |
| Total Sulfur Dioxide | 55 | 3.44 | 19 | 1.19 |
| Density | 45 | 2.81 | 5 | 0.31 |
| pH | 35 | 2.19 | 75 | 4.69 |
| Sulphates | 59 | 3.69 | 124 | 7.75 |
| Alcohol | 13 | 0.81 | 0 | 0 |
| Quality | 28 | 1.75 | 200 | 12.51 |

The number of outliers in chlorides range from moderate in red wine to high in white wine. Chlorides in wine are influenced by both terroir and type of grape (Coli et al, 2015). The type of grape can be controlled, but the terroir, or entire natural environment in which the wine is produced, is highly variable. Soil composition, natural weather phenomenon, humidity, temperature, and other extraneous factors can alter the terroir as often as every day. This variability may explain the variability of the data and the high number of outliers. Volatile acidity and citric acid also have a high number of outliers in white wine, but a low number in red wine. This might be explained by the way each type of wine is produced. Red wines are fermented with the grape seeds and skins while white wines are not. Mineral elements from the environment are absorbed through the roots of the vine and are mainly present in the skin, seeds, and pulp of the grape (Coli et al, 2015). The presence of these minerals in red wine may alter the chemical composition by decreasing acidity. White wine is generally more acidic than red wine, and this difference is evident in the taste.

## Correlation

Another important statistic to analyze is correlation between individual attributes. High correlation between explanatory variables can lead to multicollinearity which causes large standard errors and overfitting a model. Figure 1 and Figure 2 show a correlation matrix for red wine and white wine respectively. Analyzing the red wine correlation matrix in Figure 3, fixed acidity is positively correlated with citric acid. Same with total sulfur dioxide and free sulfur dioxide. For the white wine correlation matrix in Figure 4, total sulfur dioxide and residual sugar are moderately correlated. For both datasets, residual sugar and density have a strong positive correlation, and density and alcohol have a strong negative correlation. Additionally, alcohol has a moderate positive correlation with quality. Alcohol is one of the attributes that will be analyzed in the model analysis section.

Figure : Correlation Matrix for Red Wine Attributes

A picture containing text, screenshot, colorfulness, diagram

Description automatically generated

Figure : Correlation Matrix for White Wine Attributes

A picture containing text, screenshot, number, square

Description automatically generated

# Model Selection

The model chosen for the Wine dataset is the Generalized Linear Model (GLM) from the statsmodels library. The Generalized Linear Model (GLM) is an improved version of linear regression that can handle different types of data, including those that do not follow a normal distribution, by incorporating various types of regression like multiple linear, logistic, and Poisson. In this specific case, we are using the Gaussian model. The response or dependent variable is 'quality'; all other dataset features are treated as independent variables. After removing outliers from the dataset, the GLM model is fitted with these independent variables to predict the response variable 'quality'. The process is done separately for red wine and white wine data. Two models are created and used for each type of wine, red and white.

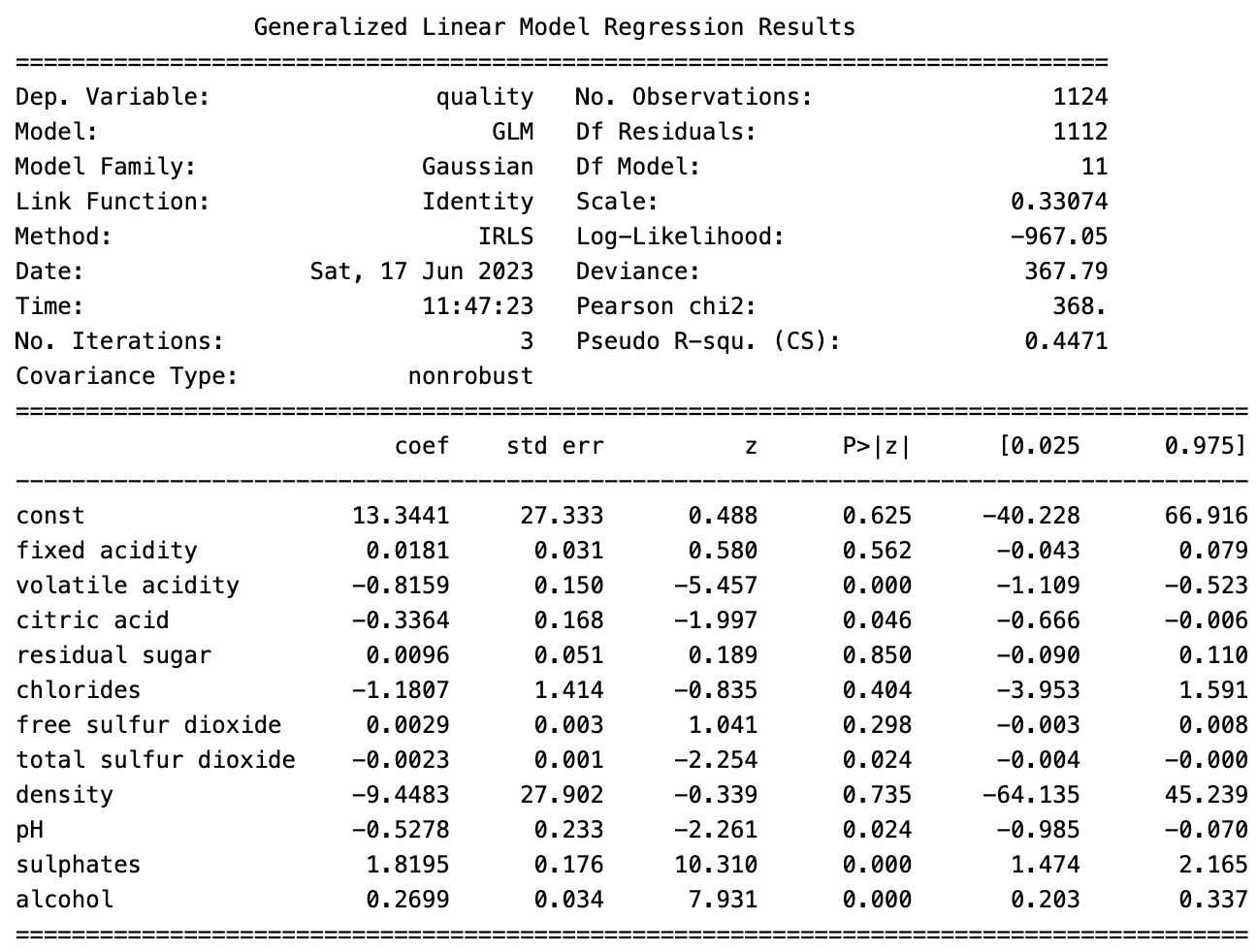
# Model Analysis

## Red Wine Model Analysis

The Generalized Linear Model (GLM) for the red wine quality prediction has been trained on 1,124 observations and includes 11 predictor variables. The model summary's coefficients provide insight into each variable's impact on wine quality. For instance, 'sulphates' and 'alcohol' appear to have a significant positive effect on wine quality, as suggested by their respective positive coefficients (1.8195 and 0.2699) and small p-values (< 0.05). Similarly, 'volatile acidity', 'citric acid', 'total sulfur dioxide', and 'pH' also significantly influence but negatively.

In contrast, some variables like 'fixed acidity', 'residual sugar', 'chlorides', 'free sulfur dioxide', and 'density' exhibit large p-values (> 0.05), suggesting that these predictors might not be statistically significant in explaining the variation in wine quality. Particularly, 'density' has a very large standard error compared to its coefficient, further indicating that it may not be a reliable predictor.

Figure : Red Wine Model Regression Results



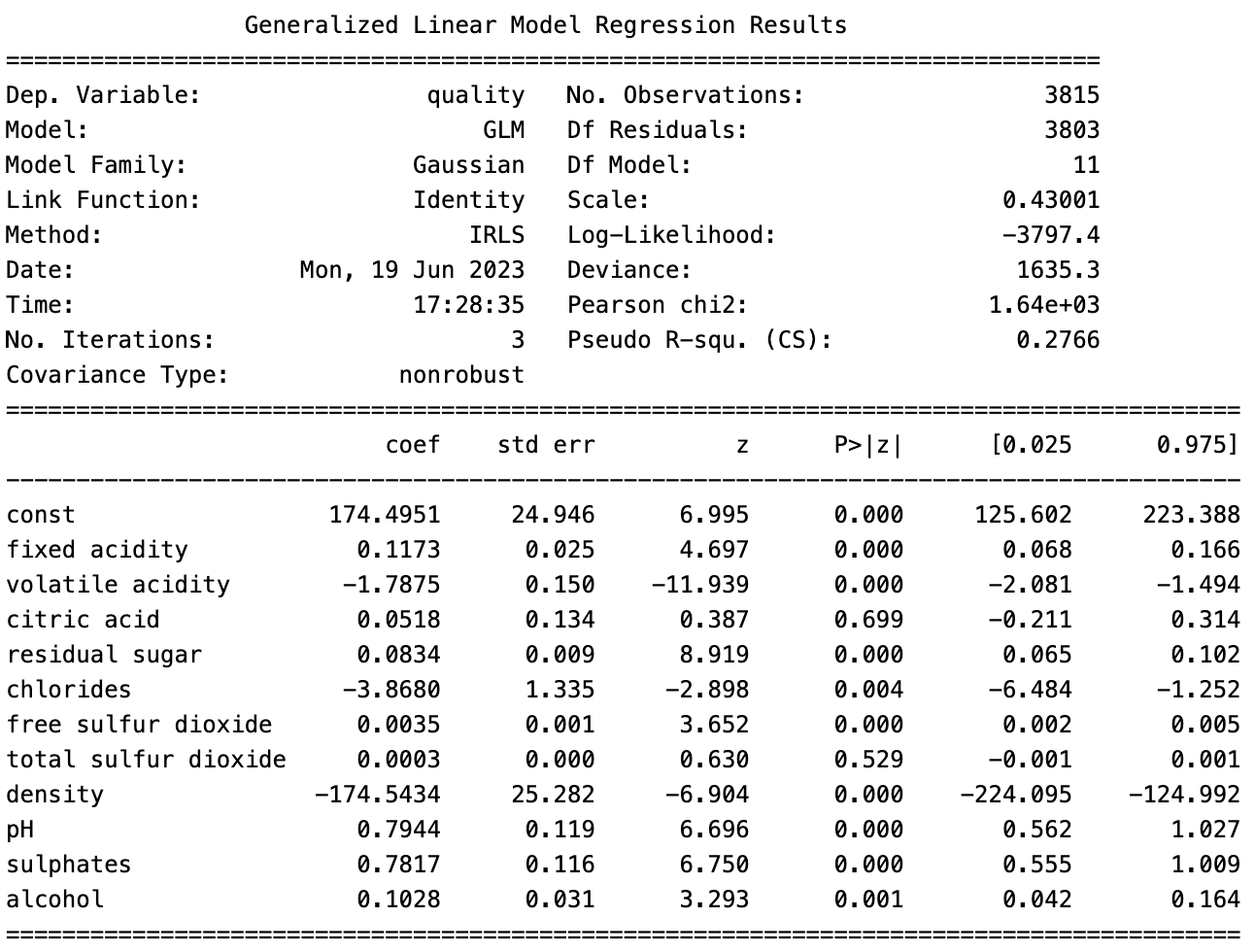
The Pseudo R-squared value is approximately 0.4471, which indicates that the model explains around 44.71% of the variability in wine quality, leaving a substantial portion unexplained. This suggests there may be other factors not included in the model that could contribute to red wine's quality, or non-linear relationships that this GLM does not capture.

## White Wine Model Analysis

This Generalized Linear Model (GLM) for white wine quality prediction has been trained on a larger set of 3,815 observations. It reveals that the statistically significant factors influencing white wine quality include 'fixed acidity', 'volatile acidity', 'residual sugar', 'chlorides', 'free sulfur dioxide', 'density', 'pH', 'sulphates', and 'alcohol'.

Among these, 'density' and 'volatile acidity' exhibit a substantial negative relationship with quality. Interestingly, unlike red wine, 'residual sugar' and 'pH' positively influences white wine quality. 'Citric acid' and 'total sulfur dioxide', however, do not have a statistically significant impact, indicating that they may not be crucial in predicting white wine quality.

Figure : White Wine Model Regression Results



Despite the model accounting for these factors, it explains only about 27.66% of the variability in white wine quality (as denoted by the Pseudo R-squared value), suggesting other important factors or non-linear relationships might not be captured in the model. While the deviance of 1635.3 indicates a reasonable model fit, the model may benefit from further enhancements such as considering additional predictors or refining variable interactions.

## Summary

In summary, while the model’s reveal some interesting patterns and appears to predict some quality variance, the relatively low R-squared value and high p-values for certain predictors suggest room for further refinement.

# Conclusion

While the predictive models developed in this study do not achieve perfect accuracy, they successfully identify the attributes that correlate with higher wine quality. These insights can prove invaluable to experienced winemakers, enabling them to refine their products and enhance their wine ratings through simple modifications.

Our model identified two modifiable attributes post-fermentation: ' pH' and 'alcohol'. The model suggests that wines with a higher alcohol content generally receive higher quality scores. In contrast, red wines exhibiting lower pH values tend to achieve higher quality ratings. White wines exhibiting higher pH achieve higher quality scores. Based on the current red and white wine datasets, our model predicts that a change in pH by 1.5 units, coupled with an increase in alcohol by 1.5 units, could potentially increase the quality score by one unit across all wines.

Figure : Wine Simulated 1 Quality Increase

A picture containing text, line, plot, diagram

Description automatically generatedA picture containing text, line, plot, diagram

Description automatically generated

## Recommendation

Our model predicts winemakers can increase quality scores by 1 unit by changing pH by 1.5 units and increasing alcohol by 1.5 units. To accomplish this, we recommend using an acidifying agent to decrease pH or using carbonate salts to increase pH. To increase alcohol content, we recommend back-adding higher alcohol wine or increasing fermentable sugars. Commonly used acidifiers in beer and wine include phosphoric acid and lactic acid. We recommend using phosphoric acid to decrease pH levels in the wine since it does not contain adverse flavors that the lactic acid may contain. To increase alcohol, we recommend blending a higher-alcohol wine or adding additional fermentable sugars such as dextrose. pH can be brought down to a low of 2.74 units for red wine and raised to 3.82 units for white wine. Alcohol can be raised to 14.9 units in red wine and 14.2 units in white wine.

Figure : Wine pH and alcohol summary

A picture containing text, font, screenshot, number

Description automatically generated A picture containing text, screenshot, font, number

Description automatically generated

Figure : Red Wine alcohol/pH quality correlation

A picture containing text, screenshot, line, diagram

Description automatically generated

Figure : White Wine alcohol/pH quality correlation

A picture containing text, screenshot, line, plot

Description automatically generated

# Appendix: Github

<https://github.com/p-parks/AAI-500-Team-4-Final-Project/tree/main>

# Appendix: Code Output

In [ ]:

# imports

import pandas as pd

from scipy import stats

import statsmodels.api as sm

import random

import numpy as np

import seaborn as sns

import matplotlib.pyplot as plt

from scipy.stats import norm

from sklearn.metrics import mean\_absolute\_error

from tabulate import tabulate

## Datasets[¶](" \l "Datasets)

In [ ]:

wine\_white = pd.read\_csv('../Dataset/wine+quality/winequality-white.csv', sep=';')

wine\_white.describe()

Out[ ]:

|  | **fixed acidity** | **volatile acidity** | **citric acid** | **residual sugar** | **chlorides** | **free sulfur dioxide** | **total sulfur dioxide** | **density** | **pH** | **sulphates** | **alcohol** | **quality** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **count** | 4898.000000 | 4898.000000 | 4898.000000 | 4898.000000 | 4898.000000 | 4898.000000 | 4898.000000 | 4898.000000 | 4898.000000 | 4898.000000 | 4898.000000 | 4898.000000 |
| **mean** | 6.854788 | 0.278241 | 0.334192 | 6.391415 | 0.045772 | 35.308085 | 138.360657 | 0.994027 | 3.188267 | 0.489847 | 10.514267 | 5.877909 |
| **std** | 0.843868 | 0.100795 | 0.121020 | 5.072058 | 0.021848 | 17.007137 | 42.498065 | 0.002991 | 0.151001 | 0.114126 | 1.230621 | 0.885639 |
| **min** | 3.800000 | 0.080000 | 0.000000 | 0.600000 | 0.009000 | 2.000000 | 9.000000 | 0.987110 | 2.720000 | 0.220000 | 8.000000 | 3.000000 |
| **25%** | 6.300000 | 0.210000 | 0.270000 | 1.700000 | 0.036000 | 23.000000 | 108.000000 | 0.991723 | 3.090000 | 0.410000 | 9.500000 | 5.000000 |
| **50%** | 6.800000 | 0.260000 | 0.320000 | 5.200000 | 0.043000 | 34.000000 | 134.000000 | 0.993740 | 3.180000 | 0.470000 | 10.400000 | 6.000000 |
| **75%** | 7.300000 | 0.320000 | 0.390000 | 9.900000 | 0.050000 | 46.000000 | 167.000000 | 0.996100 | 3.280000 | 0.550000 | 11.400000 | 6.000000 |
| **max** | 14.200000 | 1.100000 | 1.660000 | 65.800000 | 0.346000 | 289.000000 | 440.000000 | 1.038980 | 3.820000 | 1.080000 | 14.200000 | 9.000000 |

In [ ]:

wine\_red = pd.read\_csv('../Dataset/wine+quality/winequality-red.csv', sep=';')

wine\_red.describe()

Out[ ]:

|  | **fixed acidity** | **volatile acidity** | **citric acid** | **residual sugar** | **chlorides** | **free sulfur dioxide** | **total sulfur dioxide** | **density** | **pH** | **sulphates** | **alcohol** | **quality** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **count** | 1599.000000 | 1599.000000 | 1599.000000 | 1599.000000 | 1599.000000 | 1599.000000 | 1599.000000 | 1599.000000 | 1599.000000 | 1599.000000 | 1599.000000 | 1599.000000 |
| **mean** | 8.319637 | 0.527821 | 0.270976 | 2.538806 | 0.087467 | 15.874922 | 46.467792 | 0.996747 | 3.311113 | 0.658149 | 10.422983 | 5.636023 |
| **std** | 1.741096 | 0.179060 | 0.194801 | 1.409928 | 0.047065 | 10.460157 | 32.895324 | 0.001887 | 0.154386 | 0.169507 | 1.065668 | 0.807569 |
| **min** | 4.600000 | 0.120000 | 0.000000 | 0.900000 | 0.012000 | 1.000000 | 6.000000 | 0.990070 | 2.740000 | 0.330000 | 8.400000 | 3.000000 |
| **25%** | 7.100000 | 0.390000 | 0.090000 | 1.900000 | 0.070000 | 7.000000 | 22.000000 | 0.995600 | 3.210000 | 0.550000 | 9.500000 | 5.000000 |
| **50%** | 7.900000 | 0.520000 | 0.260000 | 2.200000 | 0.079000 | 14.000000 | 38.000000 | 0.996750 | 3.310000 | 0.620000 | 10.200000 | 6.000000 |
| **75%** | 9.200000 | 0.640000 | 0.420000 | 2.600000 | 0.090000 | 21.000000 | 62.000000 | 0.997835 | 3.400000 | 0.730000 | 11.100000 | 6.000000 |
| **max** | 15.900000 | 1.580000 | 1.000000 | 15.500000 | 0.611000 | 72.000000 | 289.000000 | 1.003690 | 4.010000 | 2.000000 | 14.900000 | 8.000000 |

In [ ]:

columns = [

'fixed acidity',

'volatile acidity',

'citric acid',

'residual sugar',

'chlorides',

'free sulfur dioxide',

'total sulfur dioxide',

'density',

'pH',

'sulphates',

'alcohol',

'quality'

]

## Boxplot all data to view outliers[¶](" \l "Boxplot-all-data-to-view-outliers)

In [ ]:

def do\_boxplot(data):

# fig, axes = plt.subplots(nrows=3, ncols=4, figsize=(15,10))

# 6/19/23 ACaterio: Lowering figsize to fit into screenshot for EDA in the report

fig, axes = plt.subplots(nrows=3, ncols=4, figsize=(10,8))

axes = axes.ravel()

for i, column in enumerate(columns):

axes[i].boxplot(data[column])

axes[i].set\_title(f'Boxplot of {column}')

plt.tight\_layout()

plt.show()

In [ ]:

print('BoxPlots Red Wine')

do\_boxplot(wine\_red)

BoxPlots Red Wine

A picture containing text, diagram, plan, parallel

Description automatically generated

In [ ]:

print('BoxPlots White Wine')

do\_boxplot(wine\_white)

BoxPlots White Wine

A picture containing text, diagram, plan, parallel

Description automatically generated

In [ ]:

wine\_red\_n = len(wine\_red)

wine\_white\_n = len(wine\_white)

def countOutliers(df, df\_str, n):

Q1 = df.quantile(0.25)

Q3 = df.quantile(0.75)

IQR = Q3 - Q1

outliers = ((df < (Q1 - 1.5 \* IQR)) | (df > (Q3 + 1.5 \* IQR))).sum()

outliers = outliers.tolist()

arr = []

for i in range(len(outliers)):

arr.append([])

outlier\_perc = round(outliers[i]/len(wine\_red)\*100,2)

arr[i].append(columns[i])

arr[i].append(outliers[i])

arr[i].append(outlier\_perc)

print(tabulate(arr, headers=['Attribute', 'Total Outliers', 'Percentage'], tablefmt="fancy\_grid"))

In [ ]:

countOutliers(wine\_red, "Red Wine", wine\_red\_n)

╒══════════════════════╤══════════════════╤══════════════╕

│ Attribute │ Total Outliers │ Percentage │

╞══════════════════════╪══════════════════╪══════════════╡

│ fixed acidity │ 49 │ 3.06 │

├──────────────────────┼──────────────────┼──────────────┤

│ volatile acidity │ 19 │ 1.19 │

├──────────────────────┼──────────────────┼──────────────┤

│ citric acid │ 1 │ 0.06 │

├──────────────────────┼──────────────────┼──────────────┤

│ residual sugar │ 155 │ 9.69 │

├──────────────────────┼──────────────────┼──────────────┤

│ chlorides │ 112 │ 7 │

├──────────────────────┼──────────────────┼──────────────┤

│ free sulfur dioxide │ 30 │ 1.88 │

├──────────────────────┼──────────────────┼──────────────┤

│ total sulfur dioxide │ 55 │ 3.44 │

├──────────────────────┼──────────────────┼──────────────┤

│ density │ 45 │ 2.81 │

├──────────────────────┼──────────────────┼──────────────┤

│ pH │ 35 │ 2.19 │

├──────────────────────┼──────────────────┼──────────────┤

│ sulphates │ 59 │ 3.69 │

├──────────────────────┼──────────────────┼──────────────┤

│ alcohol │ 13 │ 0.81 │

├──────────────────────┼──────────────────┼──────────────┤

│ quality │ 28 │ 1.75 │

╘══════════════════════╧══════════════════╧══════════════╛

In [ ]:

countOutliers(wine\_white, "White Wine", wine\_white\_n)

╒══════════════════════╤══════════════════╤══════════════╕

│ Attribute │ Total Outliers │ Percentage │

╞══════════════════════╪══════════════════╪══════════════╡

│ fixed acidity │ 119 │ 7.44 │

├──────────────────────┼──────────────────┼──────────────┤

│ volatile acidity │ 186 │ 11.63 │

├──────────────────────┼──────────────────┼──────────────┤

│ citric acid │ 270 │ 16.89 │

├──────────────────────┼──────────────────┼──────────────┤

│ residual sugar │ 7 │ 0.44 │

├──────────────────────┼──────────────────┼──────────────┤

│ chlorides │ 208 │ 13.01 │

├──────────────────────┼──────────────────┼──────────────┤

│ free sulfur dioxide │ 50 │ 3.13 │

├──────────────────────┼──────────────────┼──────────────┤

│ total sulfur dioxide │ 19 │ 1.19 │

├──────────────────────┼──────────────────┼──────────────┤

│ density │ 5 │ 0.31 │

├──────────────────────┼──────────────────┼──────────────┤

│ pH │ 75 │ 4.69 │

├──────────────────────┼──────────────────┼──────────────┤

│ sulphates │ 124 │ 7.75 │

├──────────────────────┼──────────────────┼──────────────┤

│ alcohol │ 0 │ 0 │

├──────────────────────┼──────────────────┼──────────────┤

│ quality │ 200 │ 12.51 │

╘══════════════════════╧══════════════════╧══════════════╛

## Correlation and Variation[¶](" \l "Correlation-and-Variation)

In [ ]:

def createCorrMatr(df, df\_str, color):

cols\_df = df.corr().nlargest(len(columns), 'quality')['quality'].index

correl = df[cols\_df].corr()

plt.figure(figsize=(10,8))

plt.title(f"Correlation Matrix: {df\_str}")

sns.heatmap(correl, annot=True, cmap = color)

createCorrMatr(wine\_red, 'Red Wine', 'plasma')

createCorrMatr(wine\_white, 'White Wine', 'GnBu')

A picture containing text, screenshot, colorfulness, diagram

Description automatically generated

A picture containing text, screenshot, number, square

Description automatically generated

## Probability of Scores[¶](" \l "Probability-of-Scores)

In [ ]:

def get\_probability(df):

df.sort\_values(by=['quality'], inplace=True)

df\_mean = np.mean(df["quality"])

df\_std = np.std(df["quality"])

pdf = stats.norm.pdf(df["quality"], df\_mean, df\_std)

plt.xlabel('Quality')

plt.ylabel('Probability')

plt.title('PDF of Quality')

plt.plot(df["quality"], pdf)

In [ ]:

get\_probability(wine\_red)

A diagram of quality

Description automatically generated with low confidence

In [ ]:

get\_probability(wine\_white)

A picture containing line, diagram, plot, slope

Description automatically generated

## Remove all outliers[¶](" \l "Remove-all-outliers)

In [ ]:

def remove\_all\_outliers(data\_source):

data = data\_source.copy()

for column in columns:

Q1 = data[column].quantile(0.25)

Q3 = data[column].quantile(0.75)

IQR = Q3 - Q1

data = data[(data[column] >= Q1 - 1.5\*IQR) & (data[column] <= Q3 + 1.5\*IQR)]

return data

In [ ]:

wine\_red\_cleaned = remove\_all\_outliers(wine\_red)

wine\_white\_cleaned = remove\_all\_outliers(wine\_white)

## Generalized Linear Model Regression[¶](" \l "Generalized-Linear-Model-Regression)

In [ ]:

def create\_glm\_fitted\_model(df):

X = df.drop('quality', axis=1)

y = df['quality']

X = sm.add\_constant(X)

# Create the model

model = sm.GLM(y, X)

return model.fit()

In [ ]:

wine\_red\_results = create\_glm\_fitted\_model(wine\_red)

print(wine\_red\_results.summary())

Generalized Linear Model Regression Results

==============================================================================

Dep. Variable: quality No. Observations: 1599

Model: GLM Df Residuals: 1587

Model Family: Gaussian Df Model: 11

Link Function: Identity Scale: 0.41992

Method: IRLS Log-Likelihood: -1569.1

Date: Sat, 24 Jun 2023 Deviance: 666.41

Time: 18:16:51 Pearson chi2: 666.

No. Iterations: 3 Pseudo R-squ. (CS): 0.4286

Covariance Type: nonrobust

========================================================================================

coef std err z P>|z| [0.025 0.975]

----------------------------------------------------------------------------------------

const 21.9652 21.195 1.036 0.300 -19.575 63.506

fixed acidity 0.0250 0.026 0.963 0.336 -0.026 0.076

volatile acidity -1.0836 0.121 -8.948 0.000 -1.321 -0.846

citric acid -0.1826 0.147 -1.240 0.215 -0.471 0.106

residual sugar 0.0163 0.015 1.089 0.276 -0.013 0.046

chlorides -1.8742 0.419 -4.470 0.000 -2.696 -1.052

free sulfur dioxide 0.0044 0.002 2.009 0.045 0.000 0.009

total sulfur dioxide -0.0033 0.001 -4.480 0.000 -0.005 -0.002

density -17.8812 21.633 -0.827 0.408 -60.281 24.519

pH -0.4137 0.192 -2.159 0.031 -0.789 -0.038

sulphates 0.9163 0.114 8.014 0.000 0.692 1.140

alcohol 0.2762 0.026 10.429 0.000 0.224 0.328

========================================================================================

The variables 'volatile acidity', 'chlorides', 'free sulfur dioxide', 'total sulfur dioxide', 'pH', 'sulphates', and 'alcohol' are statistically significant predictors of wine quality because their p-values are less than 0.05.

In [ ]:

wine\_white\_results = create\_glm\_fitted\_model(wine\_white)

print(wine\_white\_results.summary())

Generalized Linear Model Regression Results

==============================================================================

Dep. Variable: quality No. Observations: 4898

Model: GLM Df Residuals: 4886

Model Family: Gaussian Df Model: 11

Link Function: Identity Scale: 0.56454

Method: IRLS Log-Likelihood: -5543.7

Date: Sat, 24 Jun 2023 Deviance: 2758.3

Time: 18:16:51 Pearson chi2: 2.76e+03

No. Iterations: 3 Pseudo R-squ. (CS): 0.3240

Covariance Type: nonrobust

========================================================================================

coef std err z P>|z| [0.025 0.975]

----------------------------------------------------------------------------------------

const 150.1928 18.804 7.987 0.000 113.337 187.048

fixed acidity 0.0655 0.021 3.139 0.002 0.025 0.106

volatile acidity -1.8632 0.114 -16.373 0.000 -2.086 -1.640

citric acid 0.0221 0.096 0.231 0.818 -0.166 0.210

residual sugar 0.0815 0.008 10.825 0.000 0.067 0.096

chlorides -0.2473 0.547 -0.452 0.651 -1.318 0.824

free sulfur dioxide 0.0037 0.001 4.422 0.000 0.002 0.005

total sulfur dioxide -0.0003 0.000 -0.756 0.450 -0.001 0.000

density -150.2842 19.075 -7.879 0.000 -187.670 -112.899

pH 0.6863 0.105 6.513 0.000 0.480 0.893

sulphates 0.6315 0.100 6.291 0.000 0.435 0.828

alcohol 0.1935 0.024 7.988 0.000 0.146 0.241

========================================================================================

The variables 'volatile acidity', 'residual sugar', 'free sulfur dioxide', 'density', 'pH', 'sulphates', and 'alcohol' are statistically significant predictors of wine quality because their p-values are less than 0.05.

In [ ]:

wine\_red\_cleaned\_results = create\_glm\_fitted\_model(wine\_red\_cleaned)

print(wine\_red\_cleaned\_results.summary())

Generalized Linear Model Regression Results

==============================================================================

Dep. Variable: quality No. Observations: 1124

Model: GLM Df Residuals: 1112

Model Family: Gaussian Df Model: 11

Link Function: Identity Scale: 0.33074

Method: IRLS Log-Likelihood: -967.05

Date: Sat, 24 Jun 2023 Deviance: 367.79

Time: 18:16:51 Pearson chi2: 368.

No. Iterations: 3 Pseudo R-squ. (CS): 0.4471

Covariance Type: nonrobust

========================================================================================

coef std err z P>|z| [0.025 0.975]

----------------------------------------------------------------------------------------

const 13.3441 27.333 0.488 0.625 -40.228 66.916

fixed acidity 0.0181 0.031 0.580 0.562 -0.043 0.079

volatile acidity -0.8159 0.150 -5.457 0.000 -1.109 -0.523

citric acid -0.3364 0.168 -1.997 0.046 -0.666 -0.006

residual sugar 0.0096 0.051 0.189 0.850 -0.090 0.110

chlorides -1.1807 1.414 -0.835 0.404 -3.953 1.591

free sulfur dioxide 0.0029 0.003 1.041 0.298 -0.003 0.008

total sulfur dioxide -0.0023 0.001 -2.254 0.024 -0.004 -0.000

density -9.4483 27.902 -0.339 0.735 -64.135 45.239

pH -0.5278 0.233 -2.261 0.024 -0.985 -0.070

sulphates 1.8195 0.176 10.310 0.000 1.474 2.165

alcohol 0.2699 0.034 7.931 0.000 0.203 0.337

========================================================================================

In [ ]:

wine\_white\_cleaned\_results = create\_glm\_fitted\_model(wine\_white\_cleaned)

print(wine\_white\_cleaned\_results.summary())

Generalized Linear Model Regression Results

==============================================================================

Dep. Variable: quality No. Observations: 3815

Model: GLM Df Residuals: 3803

Model Family: Gaussian Df Model: 11

Link Function: Identity Scale: 0.43001

Method: IRLS Log-Likelihood: -3797.4

Date: Sat, 24 Jun 2023 Deviance: 1635.3

Time: 18:16:51 Pearson chi2: 1.64e+03

No. Iterations: 3 Pseudo R-squ. (CS): 0.2766

Covariance Type: nonrobust

========================================================================================

coef std err z P>|z| [0.025 0.975]

----------------------------------------------------------------------------------------

const 174.4951 24.946 6.995 0.000 125.602 223.388

fixed acidity 0.1173 0.025 4.697 0.000 0.068 0.166

volatile acidity -1.7875 0.150 -11.939 0.000 -2.081 -1.494

citric acid 0.0518 0.134 0.387 0.699 -0.211 0.314

residual sugar 0.0834 0.009 8.919 0.000 0.065 0.102

chlorides -3.8680 1.335 -2.898 0.004 -6.484 -1.252

free sulfur dioxide 0.0035 0.001 3.652 0.000 0.002 0.005

total sulfur dioxide 0.0003 0.000 0.630 0.529 -0.001 0.001

density -174.5434 25.282 -6.904 0.000 -224.095 -124.992

pH 0.7944 0.119 6.696 0.000 0.562 1.027

sulphates 0.7817 0.116 6.750 0.000 0.555 1.009

alcohol 0.1028 0.031 3.293 0.001 0.042 0.164

========================================================================================

## Predictions

In [ ]:

def quality\_histogram(X, y, results):

predicted\_scores = []

actual\_scores = []

for row\_iter in range(len(X)):

row = X.iloc[row\_iter]

predicted\_quality = results.predict(row)

predicted\_scores.append(predicted\_quality[0])

actual\_scores.append(y.iloc[row\_iter])

sns.kdeplot(predicted\_scores, label='Predicted Score')

sns.kdeplot(actual\_scores, label="Actual Score")

plt.legend()

plt.show()

def predict\_wine\_using\_df(df\_source, results):

df = df\_source.copy()

get\_mse\_predictions(df, results)

X = df.drop('quality', axis=1)

X = sm.add\_constant(X)

y = df['quality']

index = random.randint(0, len(df))

row = X.iloc[index]

predicted\_quality = results.predict(row)

print('Predicted wine quality:', predicted\_quality[0])

print('Predicted wine quality rounded:', round(predicted\_quality[0]))

print('Actual wine quality:', y.iloc[index])

quality\_histogram(X, y, results)

In [ ]:

def get\_mse\_predictions(df, results):

X = df.drop('quality', axis=1)

X = sm.add\_constant(X)

y = df['quality']

predictions = results.predict(X)

mae = mean\_absolute\_error(y, predictions)

print(f'Mean Absolute Error: {mae}')

In [ ]:

def predict\_simulated\_best\_wine(data\_source, results):

print('Take the best scoring wine in the dataset and make it even better.')

# new\_wine = {

# 'const': [1],

# 'fixed acidity': [8.5],

# 'volatile acidity': [0.8],

# 'citric acid': [0.56],

# 'residual sugar': [1.8],

# 'chlorides': [0.077],

# 'free sulfur dioxide': [10.0],

# 'total sulfur dioxide': [37.0],

# 'density': [0.9968],

# 'pH': [3.2],

# 'sulphates': [0.68],

# 'alcohol': [9.8]

# }

data = data\_source.copy()

# get the best scoring wine in the real dataset

X = sm.add\_constant(data)

max\_quality\_index = X['quality'].idxmax()

max\_quality\_row = X.loc[max\_quality\_index]

actual\_score = max\_quality\_row['quality']

print(f'Actual quality: {actual\_score}')

max\_quality\_row = max\_quality\_row.drop('quality')

# Statistically significant values for both red and white wines

# tldr how to get a 11/10 wine

max\_quality\_row['alcohol'] = 15 #high alcohol

max\_quality\_row['sulphates'] = 2 #high sulphates

max\_quality\_row['volatile acidity'] = 0.1 #low volatile acidity

max\_quality\_row['total sulfur dioxide'] = 30 # low total sulfur dioxide

max\_quality\_row['pH'] = 2 # low pH

print(max\_quality\_row)

predicted\_quality = results.predict(max\_quality\_row)

print(f'\nPredicted wine quality: {round(predicted\_quality[0])}\n')

In [ ]:

print('\nRed Wine prediction: \n')

predict\_wine\_using\_df(wine\_red, wine\_red\_results)

Red Wine prediction:

Mean Absolute Error: 0.500489963564491

Predicted wine quality: 6.4713673724387

Predicted wine quality rounded: 6

Actual wine quality: 6

A picture containing text, plot, line, diagram

Description automatically generated

In [ ]:

print('\nWhite Wine prediction: \n')

predict\_wine\_using\_df(wine\_white, wine\_white\_results)

White Wine prediction:

Mean Absolute Error: 0.5836349500279457

Predicted wine quality: 5.6392830408493975

Predicted wine quality rounded: 6

Actual wine quality: 6

A picture containing text, plot, diagram, line

Description automatically generated

In [ ]:

print('\nRed Wine Cleaned prediction: \n')

predict\_wine\_using\_df(wine\_red\_cleaned, wine\_red\_cleaned\_results)

Red Wine Cleaned prediction:

Mean Absolute Error: 0.4593529800397901

Predicted wine quality: 5.2065795461938

Predicted wine quality rounded: 5

Actual wine quality: 5

A picture containing text, plot, line, diagram

Description automatically generated

In [ ]:

print('\nWhite Wine Cleaned prediction: \n')

predict\_wine\_using\_df(wine\_white\_cleaned, wine\_white\_cleaned\_results)

White Wine Cleaned prediction:

Mean Absolute Error: 0.5287723085610089

Predicted wine quality: 6.218434120248082

Predicted wine quality rounded: 6

Actual wine quality: 6

A graph of blue and orange lines

Description automatically generated with low confidence

In [ ]:

print('\nRed Wine prediction: \n')

predict\_simulated\_best\_wine(wine\_red, wine\_red\_cleaned\_results)

Red Wine prediction:

Take the best scoring wine in the dataset and make it even better.

Actual quality: 8.0

const 1.0000

fixed acidity 5.5000

volatile acidity 0.1000

citric acid 0.0300

residual sugar 1.8000

chlorides 0.0440

free sulfur dioxide 28.0000

total sulfur dioxide 30.0000

density 0.9908

pH 2.0000

sulphates 2.0000

alcohol 15.0000

Name: 1269, dtype: float64

Predicted wine quality: 11

In [ ]:

print('\nWhite Wine prediction: \n')

predict\_simulated\_best\_wine(wine\_white, wine\_white\_cleaned\_results)

White Wine prediction:

Take the best scoring wine in the dataset and make it even better.

Actual quality: 9.0

const 1.000

fixed acidity 9.100

volatile acidity 0.100

citric acid 0.450

residual sugar 10.600

chlorides 0.035

free sulfur dioxide 28.000

total sulfur dioxide 30.000

density 0.997

pH 2.000

sulphates 2.000

alcohol 15.000

Name: 774, dtype: float64

Predicted wine quality: 7

In [ ]:

def predict\_simulated\_best\_wine\_only\_modify\_pH\_and\_alcohol(data\_source, results):

predicted\_scores\_original = []

predicted\_scores\_with\_modifications = []

score\_diff = []

data = data\_source.copy()

X = sm.add\_constant(data)

for row\_iter in range(len(data)):

row = X.loc[row\_iter]

row = row.drop('quality')

predicted\_quality = results.predict(row)

row['alcohol'] = row['alcohol'] + 1.5

row['pH'] = row['pH'] - 1.5

predicted\_quality\_modified = results.predict(row)

predicted\_scores\_original.append(predicted\_quality[0])

predicted\_scores\_with\_modifications.append(predicted\_quality\_modified[0])

score\_diff = predicted\_quality\_modified[0] - predicted\_quality[0]

sns.kdeplot(predicted\_scores\_original, label='Predicted Score (Original)')

sns.kdeplot(predicted\_scores\_with\_modifications, label="Predicted Score (+1.5% abv -1.5%pH)")

plt.legend()

plt.show()

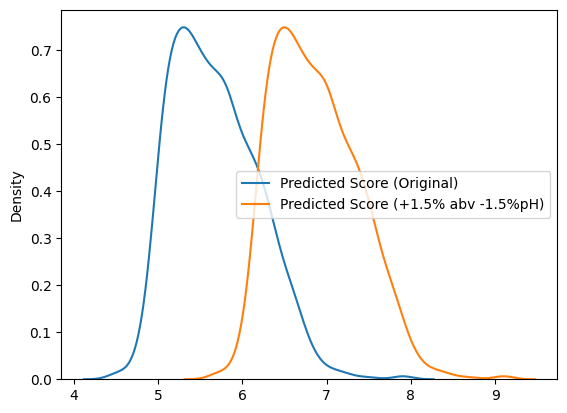
print(f'Average Score difference (Score point out of 10): {np.mean(score\_diff)}')

In [ ]:

# pH and Alcohol and both easily adjustable post fermentation.

# What would happen to our wine scores if we increased alcohol and decreased pH?

predict\_simulated\_best\_wine\_only\_modify\_pH\_and\_alcohol(wine\_red, wine\_red\_cleaned\_results)



Average Score difference (Score point out of 10): 1.1965024634214307

In [ ]:

predict\_simulated\_best\_wine\_only\_modify\_pH\_and\_alcohol(wine\_white, wine\_white\_cleaned\_results)

A picture containing text, line, diagram, plot

Description automatically generated

Average Score difference (Score point out of 10): -1.0374215502692428

Increasing alcohol percentage by 1.5 and lowering pH by 1.5 gains an average of 1 whole score point.

In [ ]:

subset = wine\_red[["pH", "alcohol"]]

description = subset.describe()

print('Red Wine pH and alcohol summary')

print(description)

Red Wine pH and alcohol summary

pH alcohol

count 1599.000000 1599.000000

mean 3.311113 10.422983

std 0.154386 1.065668

min 2.740000 8.400000

25% 3.210000 9.500000

50% 3.310000 10.200000

75% 3.400000 11.100000

max 4.010000 14.900000

In [ ]:

subset = wine\_white[["pH", "alcohol"]]

description = subset.describe()

print('White Wine pH and alcohol summary')

print(description)

White Wine pH and alcohol summary

pH alcohol

count 4898.000000 4898.000000

mean 3.188267 10.514267

std 0.151001 1.230621

min 2.720000 8.000000

25% 3.090000 9.500000

50% 3.180000 10.400000

75% 3.280000 11.400000

max 3.820000 14.200000

In [ ]:

def do\_regression\_and\_plot(df, param, axs, label):

X = df[[param]]

y = df['quality']

# Add a constant to the independent value

X = sm.add\_constant(X)

# Perform regression

model = sm.GLM(y, X)

results = model.fit()

axs.scatter(X[param], y, alpha=0.5)

# Fitted line

y\_pred = results.predict(X)

axs.plot(X[param], y\_pred, color='red')

axs.set\_xlabel(param)

axs.set\_ylabel('Quality')

axs.set\_title(label + ' Wine Quality vs ' + param + ' Content')

def create\_scatterplot(df, label):

# Create two subplots side by side

fig, axs = plt.subplots(1, 2, figsize=(10, 5)) # 1 row, 2 columns

# Scatter plot for first set of data

do\_regression\_and\_plot(df, 'alcohol', axs[0], label)

# Scatter plot for second set of data

do\_regression\_and\_plot(df, 'pH', axs[1], label)

# Display the plots

plt.show()

In [ ]:

create\_scatterplot(wine\_red, 'Red')

A picture containing text, screenshot, line, diagram

Description automatically generated

In [ ]:

create\_scatterplot(wine\_white, 'White')

A picture containing text, screenshot, line, plot

Description automatically generated

# Bibliography

1. <https://archive.ics.uci.edu/dataset/186/wine+quality>
2. <http://www3.dsi.uminho.pt/pcortez/wine5.pdf>
3. <http://www3.dsi.uminho.pt/pcortez/wine5.pdf>
4. <https://www.scielo.br/j/cta/a/HQsrPrPMNZYgRzSKtrjHyHh/?format=pdf>
5. <https://r4ds.had.co.nz/exploratory-data-analysis.html>
6. <https://towardsdatascience.com/exploratory-data-analysis-8fc1cb20fd15>
7. <https://www.vinhoverde.pt/en/homepage>
8. <https://www.vinhoverde.pt/en/homepage>