University of Hertfordshire

(Affiliated with Nebula Institute of Technology, Sri Lanka)

Module Name	:	Programming
Module Code	:	5FTC2150
Assignment Title	:	CW2 – Hand in Assignment
Submitted by:		
Student Name	:	S. A. L. M. Mihiliya Jayasiri
Student ID	:	23113610
Degree Programme	:	BSc (Hons) Data Science
Department	:	Physics, Engineering and Computer Science
Academic Year	:	Year 2 - 2025/26
Submitted to:		
Lecturer	:	Ms. Thushari Senevirathne
Date of Submission	:	24/10/2025

Table of Contents

Table of Figures

Table of Tables

1 Dataset Selection

Dataset Title: Wine Quality – Red Wine Variant (winequality-red)

Source: UCI Machine Learning Repository

Link: https://archive.ics.uci.edu/dataset/186/wine+quality

1.1 Dataset Structure

The dataset contains 1599 records of Portuguese *Vinho Verde* red wine samples. Each record includes 11 continuous physicochemical features: fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, and alcohol content and a discrete quality score.

The heatmap in Figure 1 below helps identify the relationship of each feature with the wines quality.

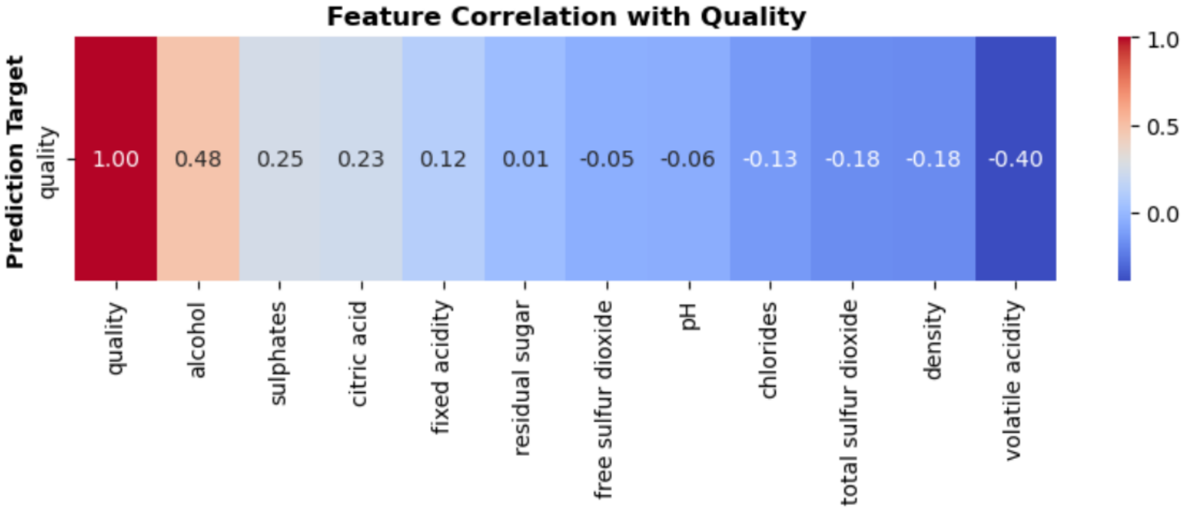


Figure 1: Heatmap: Feature Correlation with Quality

The Table 1 below provides concise definitions, their roles in wine quality, and the observed correlation direction.

Feature	Definition	Role in Wine Quality	Correlation with Wine Quality
Fixed Acidity	Tartaric acid contributing to sourness	- Enhances freshness. - Excessive levels can be harsh.	Slight positive
Volatile Acidity	Acetic acid responsible for vinegar aroma	- High levels indicate spoilage or oxidation.	Strong negative
Citric Acid	Adds freshness and flavour complexity	- Balances sweetness and acidity. - Improves flavour profile.	Positive
Residual Sugar	Sugar remaining after fermentation	- Affects sweetness and mouthfeel. - Excess may lower red wine quality.	Weak
Chlorides	Salt content in wine	- High levels reduce taste and stability.	Slight negative
Free Sulfur Dioxide	Active preservative preventing microbial growth	- Important for shelf life.	Weak
Total Sulfur Dioxide	Sum of free and bound SO_2	- Excess can suppress flavour and aroma.	Negative
Density	Related to sugar and alcohol content	- Lower density implies higher alcohol	Negative
pH	Measures acidity strength	- Lower pH enhances freshness and stability.	Weak
Sulphates	Antioxidant and antimicrobial properties	- Moderate levels improve preservation and flavour.	Positive
Alcohol	Ethanol concentration	- Enhances body, aroma, and richness.	Strongest positive

Table 1: Dataset Feature Descriptions, Roles & Correlation with Wine Quality

The discrete target variable is quality, which is a sensory score ranging from 0 to 10 assigned by professional tasters. It serves as the dependent variable for both rule-based and regression prediction models.

1.2 Relevance to the Prediction Problem

The dataset is numeric, interpretable and well-suited for both rule-based and statistical approaches. Correlations between chemical attributes and wine quality make it ideal for evaluating how human-interpretable rules compare to machine-learned predictions.

2 Data Cleaning and Preparation

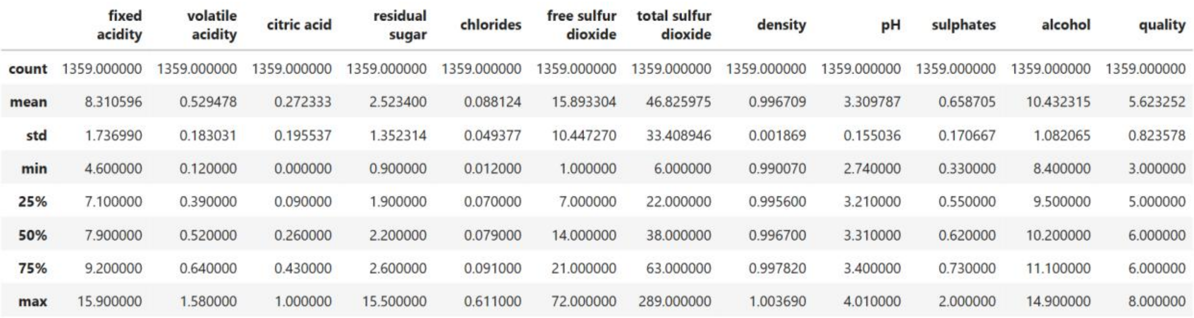
Data preparation was performed in Python using the pandas library. Table 2 below show how the dataset was validated, cleaned, and formatted with the results.

Cleaning Step	Description	Result
Import	Loaded dataset using pandas	1 599 records, 12 columns
Validation	Used .info() and .describe()	All numeric types
Missing Values	None found	Complete dataset
Duplicates	240 duplicates removed	1 359 unique rows
Data Types	Float64 / Int64	Compatible with ML

Table 2: Data Cleaning & Structuring Process with Descriptions & Results

Exception handling was added to manage file and type errors.

Below in Figure 3 is the summary/descriptive statistics of the cleaned dataset.



	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
count	1359.000000	1359.000000	1359.000000	1359.000000	1359.000000	1359.000000	1359.000000	1359.000000	1359.000000	1359.000000	1359.000000	1359.000000
mean	8.310596	0.529478	0.272333	2.523400	0.088124	15.893304	46.825975	0.996709	3.309787	0.658705	10.432315	5.623252
std	1.736990	0.183031	0.195537	1.352314	0.049377	10.447270	33.408946	0.001869	0.155036	0.170667	1.082065	0.823578
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	0.996700	3.310000	0.620000	10.200000	6.000000
75%	9.200000	0.640000	0.430000	2.600000	0.091000	21.000000	63.000000	0.997820	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

Table 3: Dataset Summary Statistics After Cleaning

3 Implementation of Prediction Methods

3.1 Rule-Based Prediction

Deriving the Rules

1. Correlation study

   The single target (only quality based) heatmap revealed that;

   - Alcohol – strong positive correlation (+0.48)
   - Sulphates – moderate positive correlation (+0.25)
   - Volatile acidity – strong negative correlation (-0.40)

2. Logical interpretation

   - Features with positive correlation raise quality when large

- Features with negative correlation reduce quality when large

3. Threshold design

The line chart in Figure 2 below, visualizes how the normalized averages of three key chemical features alcohol, sulphates, and volatile acidity change with increasing wine-quality scores.

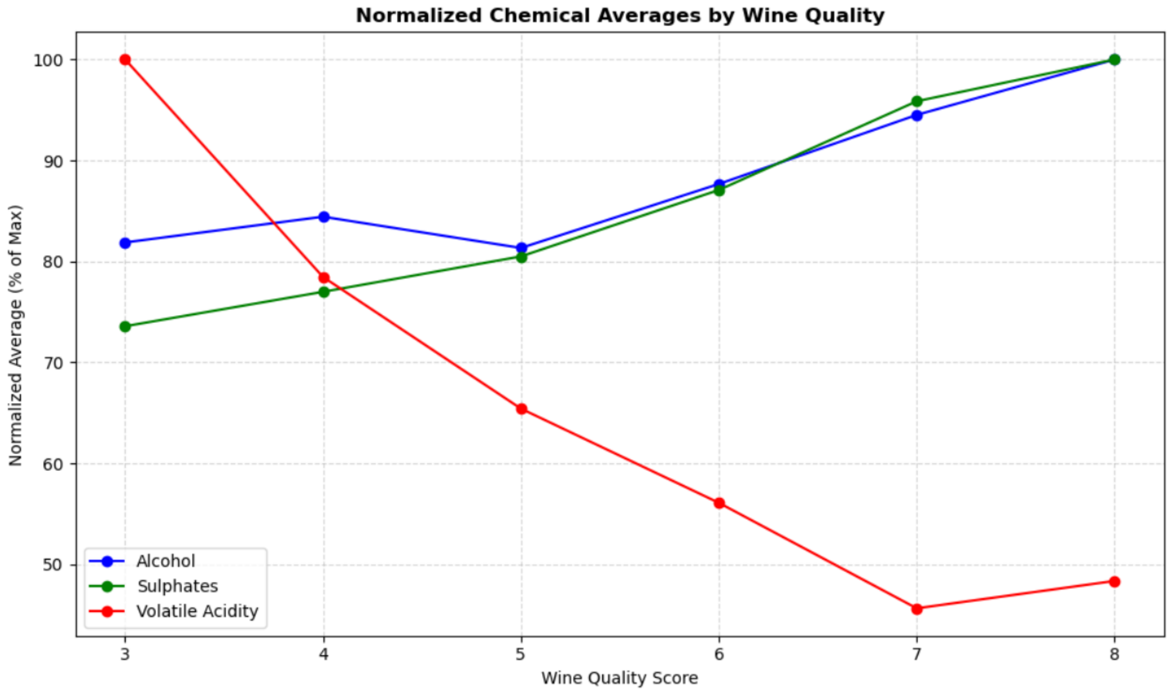


Figure 2: Normalized Chemical Averages by Wine Quality

- Alcohol (blue line) and sulphates (green line) both show positive upward trends, confirming that higher concentrations of these compounds are associated with improved wine quality.
- Volatile acidity (red line) shows a steady decline as quality increases, indicating that lower acidity levels correspond to higher sensory scores.

These trends reinforce the correlation results from the heatmap and provide clear visual evidence for the rule-based thresholds, which is that higher alcohol and sulphate content means higher quality and higher volatile acidity means lower quality.

Thresholds were determined using both histograms and boxplots to confirm that the chosen cut-off points accurately represent natural groupings in the data.

- Alcohol > 11 → likely high quality; 10 – 11 → medium; < 10 → low.

- Sulphates > 0.7 supports high quality.
- Volatile acidity > 0.7 strongly indicates low quality; < 0.5 is medium quality.

Histograms in Figure 3 below show how often each feature value occurs.

- Alcohol: Most values lie between 9-12%, with a clear decline after 11%, indicating that higher alcohol levels are linked with better quality.
- Sulphates: Most samples cluster below 0.7g/L and values beyond this point are less frequent but associated with higher quality.
- Volatile acidity: Most samples fall around 0.4 - 0.6 g/L and frequencies drop sharply above 0.7g/L, confirming that high acidity corresponds lower quality.

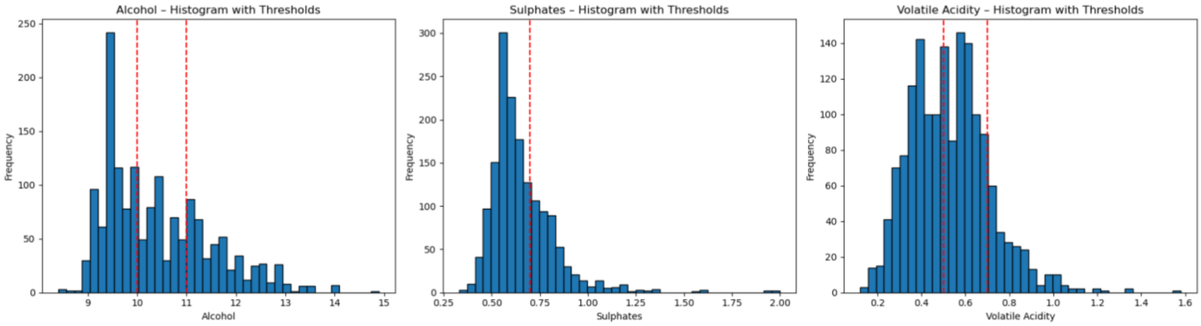


Figure 3: Histograms of Alcohol, Sulphates, and Volatile Acidity with Thresholds

Boxplots in Figure 4 below visually validate these boundaries.

- The alcohol boxplot shows a median near 10.5% and an upper quartile above 11% reinforcing 11% as a practical high quality threshold.
- The sulphates boxplot shows moderate dispersion with outliers above 0.7, confirming this as a reasonable cut-off for high quality.
- The volatile-acidity boxplot places the lower quartile near 0.4 - 0.5, consistent with the acceptable range of medium quality wines.

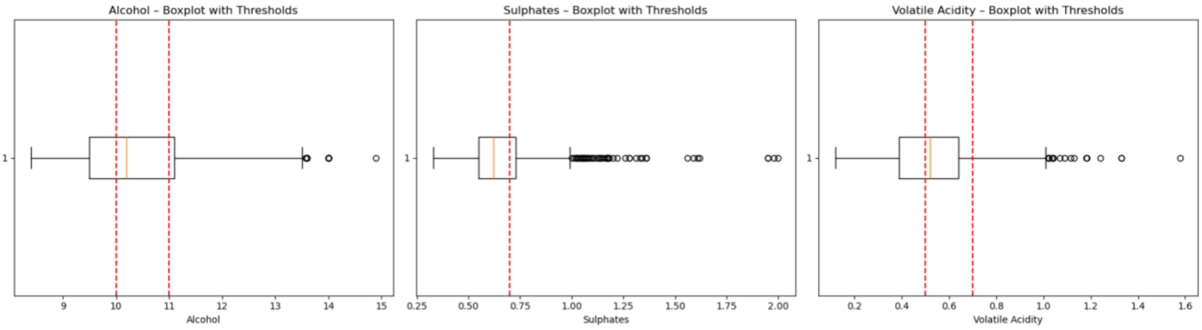


Figure 4: Boxplots of Alcohol, Sulphates, and Volatile Acidity with Thresholds

Together, the histograms identify where values naturally cluster, and the boxplots confirm that these thresholds align with actual data spread supporting that the selected rules are statistically and visually justified.

4. Rule logic construction

   Table 3 below shows how the conditions were assigned to predicted categories.

Predicted Category	**Conditions**
High	Alcohol > 11 and Sulphates > 0.7
Medium	10 ≤ Alcohol ≤ and Volatile Acidity < 0.5
Low	Alcohol < 10 or Volatile Acidity > 0.7
Fallback	Medium (default)

5. Results

The bar chart in Figure 5 below shows that most wines with actual quality scores from 3 to 5 were predicted low, while scores around 6 to 7 were mostly predicted as medium, and highest score 8 were predicted as high. This pattern confirms that the rule-based model follows the intended threshold logic.

Confusion Matrix in Figure 6 below shows that the model performs well for low quality wines, correctly classifying most of them. However, there is some overlap between medium and high categories, as their chemical values are closer together. Overall, the rule-based model gives a clear, interpretable trend, though it is less precise for borderline cases.

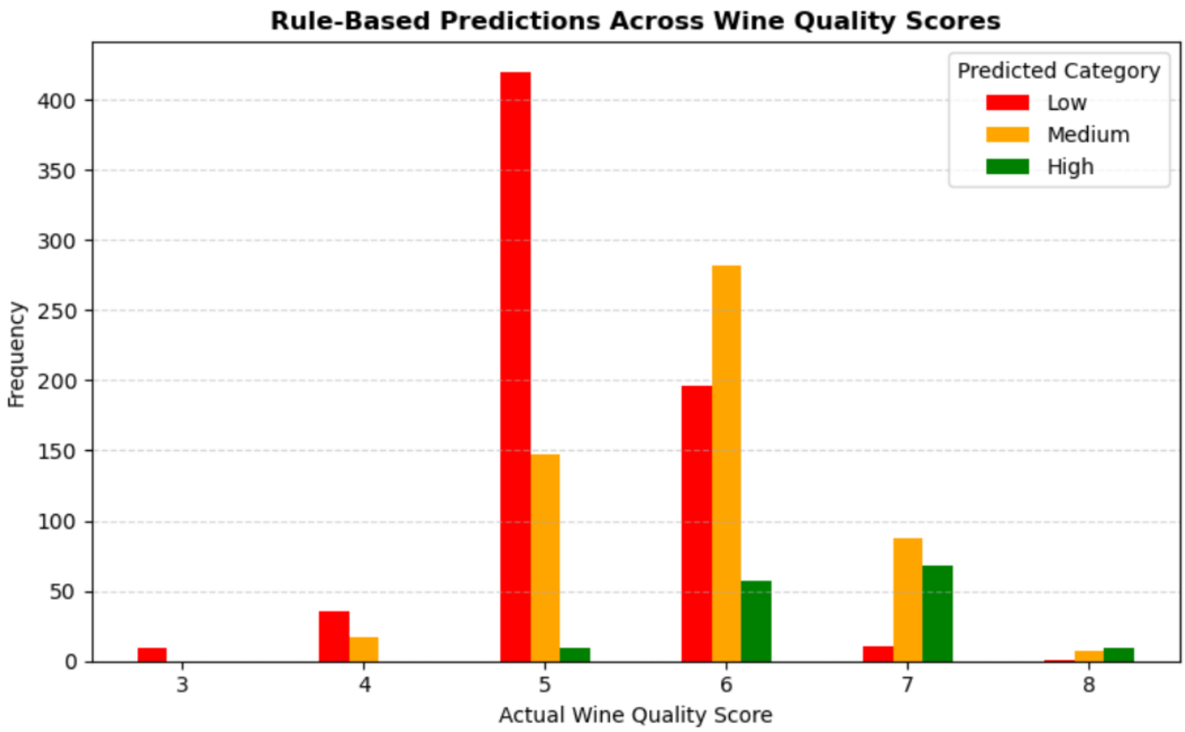


Figure 5: Rule-Based Predictions Across Wine Quality Scores

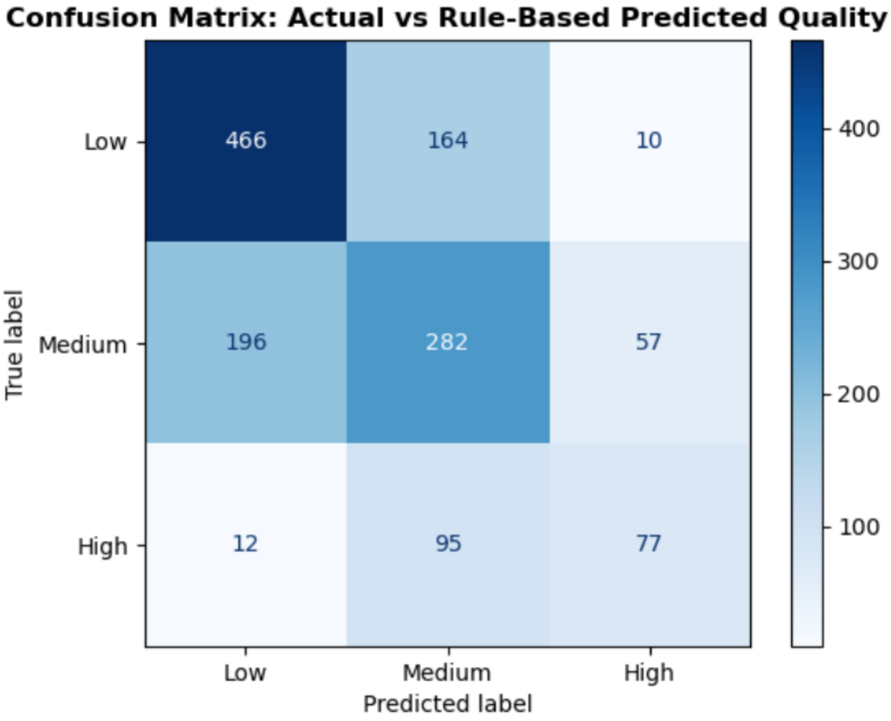


Figure 6: Confusion Matrix: Actual vs Rule-Based Predicted Quality

3.2 Statistical Prediction – Ridge Regression

A Ridge Regression model was implemented using scikit learn to improve the ordinary Linear Regression through L2 regularisation, which penalises excessively large coefficients and helps to reduce overfitting.

Workflow

1. Features: All 11 physicochemical variables were used as predictors.
2. Normalization: Standardization was applied to equalize feature ranges before model fitting.
3. Data split: The dataset was divided into 80% training and 20% testing set.
4. Model training: A ridge regression model was trained using different values of regularisation parameter α to identify the one that minimised prediction error. The optimal value, α = 59, produced the lowest MSE on the test data.
5. Evaluation: Model performance was assessed using Mean Squared Error (MSE).

Results

- Mean Squared Error (MSE) – 0.42750813

The Ridge Regression model achieved a slightly lower MSE compared with the Linear Regression model (MSE = 0.43100091), indicating a minor improvement in predictive accuracy and better generalisation stability.

Testing multiple α values confirmed that α = 59 provided the best balance between model complexity and performance, as it yielded the smallest error.

This result suggests that while dataset is already well-behaved and exhibits limited multicollinearity, applying L2 regularisation further stabilises the regression coefficients without significantly altering the model’s overall behaviour.

The scatter plot in Figure 7 below compares predicted and actual wine quality scores.

Most data points cluster closely around the red dashed diagonal line, which represents perfect prediction (predicted = actual). This alignment confirms that the ridge regression model successfully captures the general trend of the data, although some minor deviations indicate slight under or over predictions for certain samples.

Overall, the plot visually supports the numerical results, showing that the model predicts wine quality with reasonable accuracy and consistency.

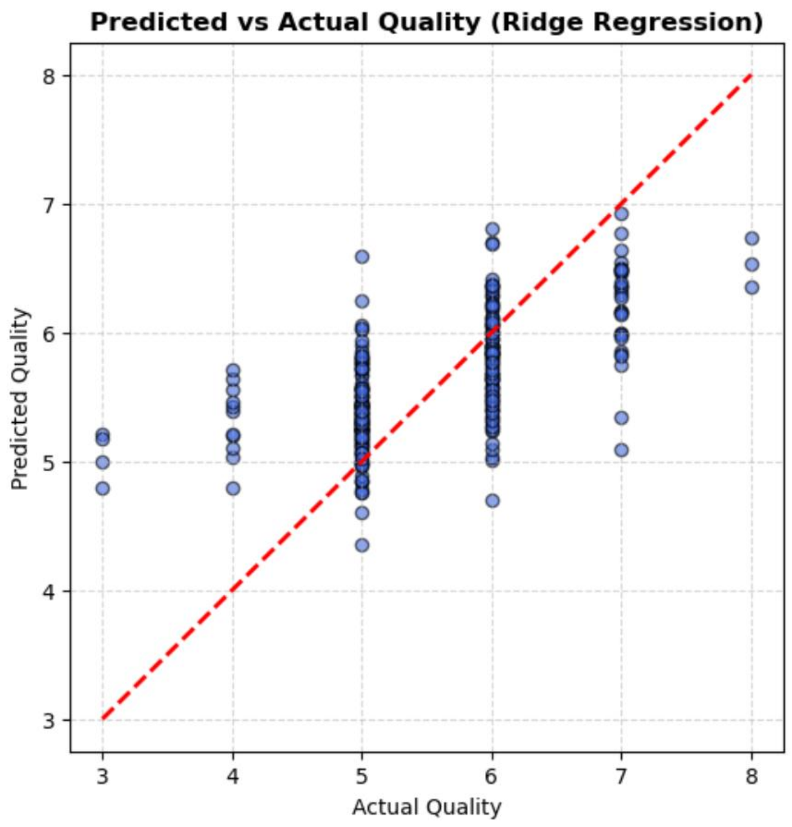


Figure 7: Predicted vs Actual Quality (Ridge Regression)

4 Comparative Evaluation

Criterion	Rule-Based Model	Ridge Regression Model
Interpretability	Very high – easy to understand and manually explain how each feature affects wine quality through clear threshold logic.	Moderate – regression coefficients indicate feature influence but require statistical interpretation.
Predictive Accuracy	Qualitative – follows general trends correctly but lacks precision for borderline cases.	Quantitative – achieved a lower MSE (0.4275) compared to the linear regression model (0.4310), showing slightly better accuracy and generalization.
Complexity	Low – uses simple conditional rules derived from data visualisation.	Moderate – involves feature scaling, parameter tuning (α = 59), and regularisation.

Transparency	Full – every decision is traceable and interpretable by humans.	Partial – model is explainable through coefficients but involves mathematical abstraction.
Scalability	Easy to extend to new datasets by adjusting threshold values.	Requires retraining with new data to maintain performance consistency.

Table 4: Evaluative Comparison of Rule-Based vs Statistical Models

The comparative evaluation in Table 4 above highlights that while the Rule-Based model provides simplicity, interpretability, and ease of explanation, it performs better for clearly defined categories. However, it struggles with borderline cases due to its fixed thresholds.

In contrast, the Ridge Regression model offers slightly improved predictive performance by capturing precise linear relationships among multiple features simultaneously. Although it is less intuitive, it generalizes better to unseen data and reduces overfitting through L2 regularisation.

Overall, Ridge Regression demonstrates stronger predictive reliability, whereas the Rule-Based approach excels in clarity and educational value.

5 Conclusion

This report presented a complete workflow for predicting wine quality, from dataset selection and cleaning to model development, evaluation, and comparison. It combined rule-based logic, grounded in data exploration, with statistical modelling using Ridge Regression to achieve both interpretability and measurable accuracy.

Key Findings

- Alcohol content is the strongest positive determinant of quality.
- Sulphates provide secondary support for higher quality.
- High volatile acidity lowers perceived quality.

The Rule-Based model effectively demonstrates how logical conditions can classify wines using intuitive chemical thresholds, while the Ridge Regression model refines these relationships mathematically, achieving a slightly lower MSE of 0.4275 and more consistent generalization.

This balance between explainability and accuracy illustrates the complementary roles of human-defined rules and machine learning models in data driven decision making.

Limitations and Future Work

- The dataset is relatively balanced but limited in range, having additional samples could have strengthened generalization.
- Future work could explore non-linear models.
- Automating rule extraction from statistical models could bridge the gap between interpretability and predictive power.

In summary, this coursework demonstrates that rule-based and statistical models have their own advantages and limitations. However, developing an approach that effectively blends the interpretability of rule-based systems with the predictive power of statistical models could be a potential game changer.

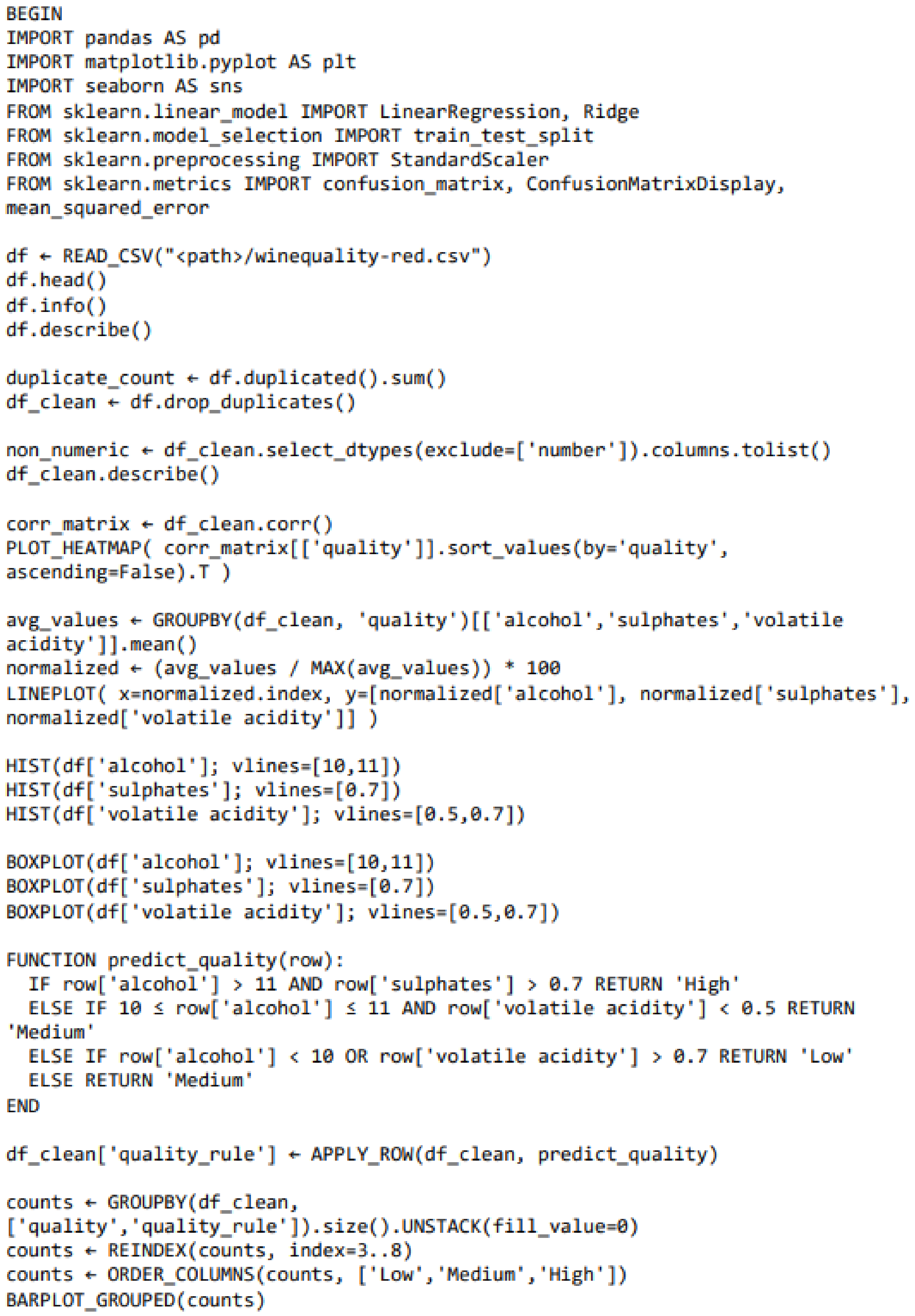
AI Assistance Disclosure

AI tools were used only for language refinement and grammar correction throughout the preparation of this report.

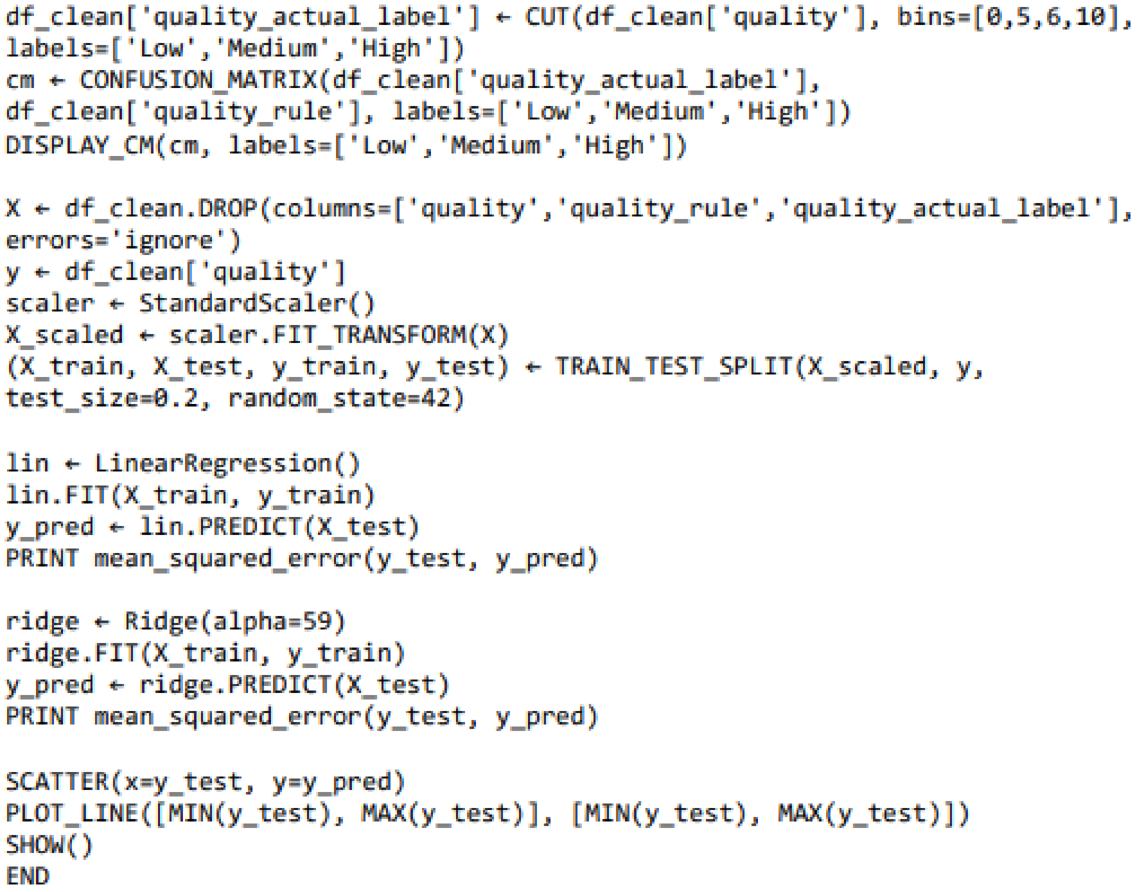
6 References

Cortez, P. &. (2009). *Wine Quality Data Set*. Retrieved from UCI Machine Learning Repository.: https://archive.ics.uci.edu/dataset/186/wine+quality

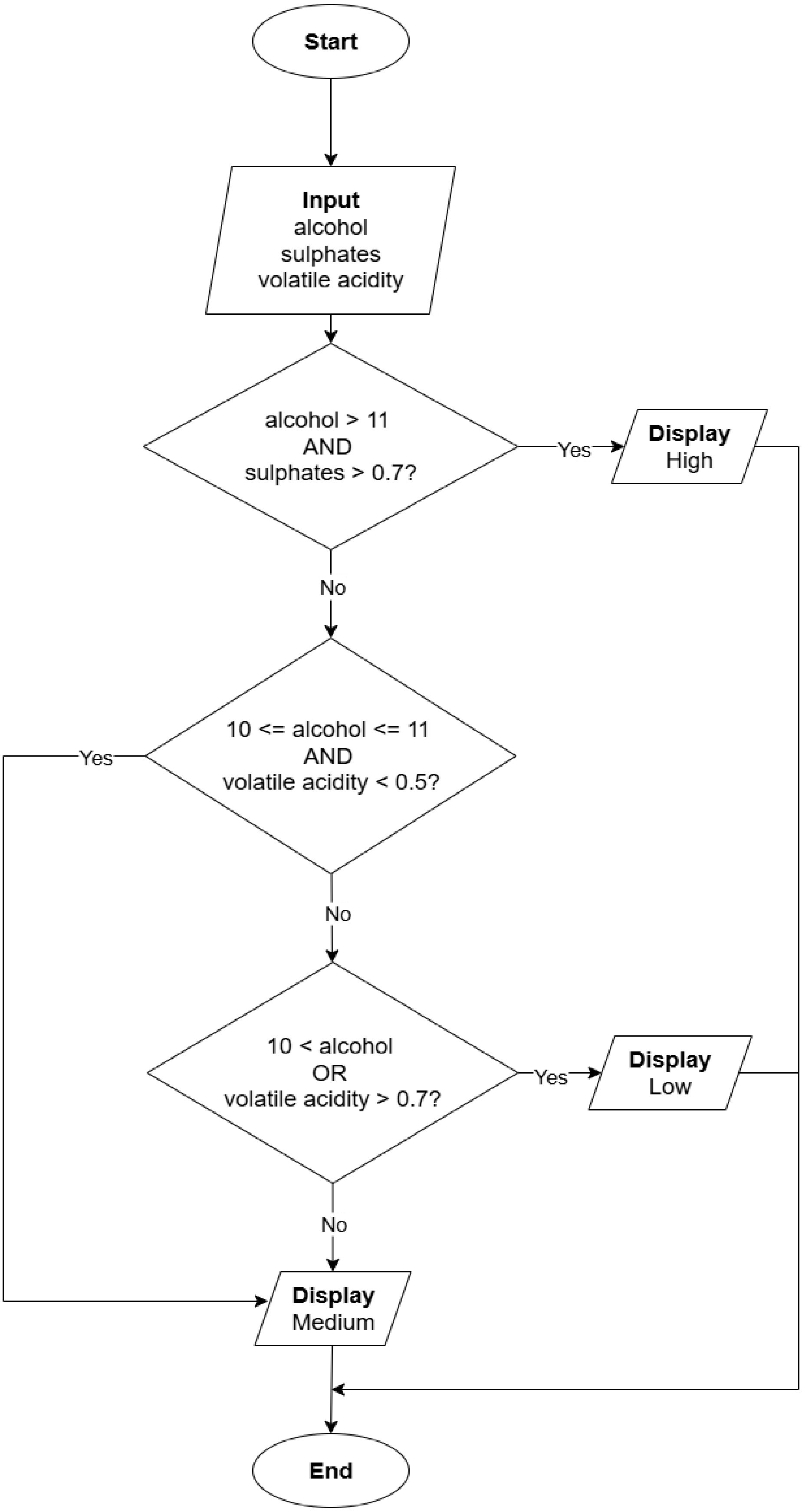
7 Program Logic Pseudocode and Rule-Based Model Flowchart



```
BEGIN
IMPORT pandas AS pd
IMPORT matplotlib.pyplot AS plt
IMPORT seaborn AS sns
FROM sklearn.linear_model IMPORT LinearRegression, Ridge
FROM sklearn.model_selection IMPORT train_test_split
FROM sklearn.preprocessing IMPORT StandardScaler
FROM sklearn.metrics IMPORT confusion_matrix, ConfusionMatrixDisplay,
mean_squared_error

df ← READ_CSV("<path>/winequality-red.csv")
df.head()
df.info()
df.describe()

duplicate_count ← df.duplicated().sum()
df_clean ← df.drop_duplicates()

non_numeric ← df_clean.select_dtypes(exclude=['number']).columns.tolist()
df_clean.describe()

corr_matrix ← df_clean.corr()
PLOT_HEATMAP( corr_matrix[['quality']].sort_values(by='quality',
ascending=False).T )

avg_values ← GROUPBY(df_clean, 'quality')[['alcohol','sulphates','volatile
acidity']].mean()
normalized ← (avg_values / MAX(avg_values)) * 100
LINEPLOT( x=normalized.index, y=[normalized['alcohol'], normalized['sulphates'],
normalized['volatile acidity']] )

HIST(df['alcohol']; vlines=[10,11])
HIST(df['sulphates']; vlines=[0.7])
HIST(df['volatile acidity']; vlines=[0.5,0.7])

BOXPLOT(df['alcohol']; vlines=[10,11])
BOXPLOT(df['sulphates']; vlines=[0.7])
BOXPLOT(df['volatile acidity']; vlines=[0.5,0.7])

FUNCTION predict_quality(row):
  IF row['alcohol'] > 11 AND row['sulphates'] > 0.7 RETURN 'High'
  ELSE IF 10 ≤ row['alcohol'] ≤ 11 AND row['volatile acidity'] < 0.5 RETURN
'Medium'
  ELSE IF row['alcohol'] < 10 OR row['volatile acidity'] > 0.7 RETURN 'Low'
  ELSE RETURN 'Medium'
END

df_clean['quality_rule'] ← APPLY_ROW(df_clean, predict_quality)

counts ← GROUPBY(df_clean,
['quality','quality_rule']).size().UNSTACK(fill_value=0)
counts ← REINDEX(counts, index=3..8)
counts ← ORDER_COLUMNS(counts, ['Low','Medium','High'])
BARPLOT_GROUPED(counts)
```



```
df_clean['quality_actual_label'] ← CUT(df_clean['quality'], bins=[0,5,6,10],
labels=['Low','Medium','High'])
cm ← CONFUSION_MATRIX(df_clean['quality_actual_label'],
df_clean['quality_rule'], labels=['Low','Medium','High'])
DISPLAY_CM(cm, labels=['Low','Medium','High'])

X ← df_clean.DROP(columns=['quality','quality_rule','quality_actual_label'],
errors='ignore')
y ← df_clean['quality']
scaler ← StandardScaler()
X_scaled ← scaler.FIT_TRANSFORM(X)
(X_train, X_test, y_train, y_test) ← TRAIN_TEST_SPLIT(X_scaled, y,
test_size=0.2, random_state=42)

lin ← LinearRegression()
lin.FIT(X_train, y_train)
y_pred ← lin.PREDICT(X_test)
PRINT mean_squared_error(y_test, y_pred)

ridge ← Ridge(alpha=59)
ridge.FIT(X_train, y_train)
y_pred ← ridge.PREDICT(X_test)
PRINT mean_squared_error(y_test, y_pred)

SCATTER(x=y_test, y=y_pred)
PLOT_LINE([MIN(y_test), MAX(y_test)], [MIN(y_test), MAX(y_test)])
SHOW()
END
```


Start
Input
alcohol
sulphates
volatile acidity
alcohol > 11
AND
sulphates > 0.7?
Yes
Display
High
No
10 <= alcohol <= 11
AND
volatile acidity < 0.5?
Yes
No
10 < alcohol
OR
volatile acidity > 0.7?
Yes
Display
Low
No
Display
Medium
End