Exploration of Wine Quality Datasets using Connectionist methods

CONCEPTS IN AI

Mwaka Namwila | DAS5001 | December 2025

Table of Contents

Introduction

In this report we were tasked with creating models that correctly predict the quality of red wines and white wines using the dataset given to us in the brief. In this report I will go over the attributes, classes and distribution of the datasets, exploring and analysing them and giving descriptions of the challenges faced while working with this dataset as well as the solutions I came up with and justifications for the solutions used.

Exploratory Data Analysis (EDA)

CHALLENGES FACED BY DATASET AND SOLUTIONS

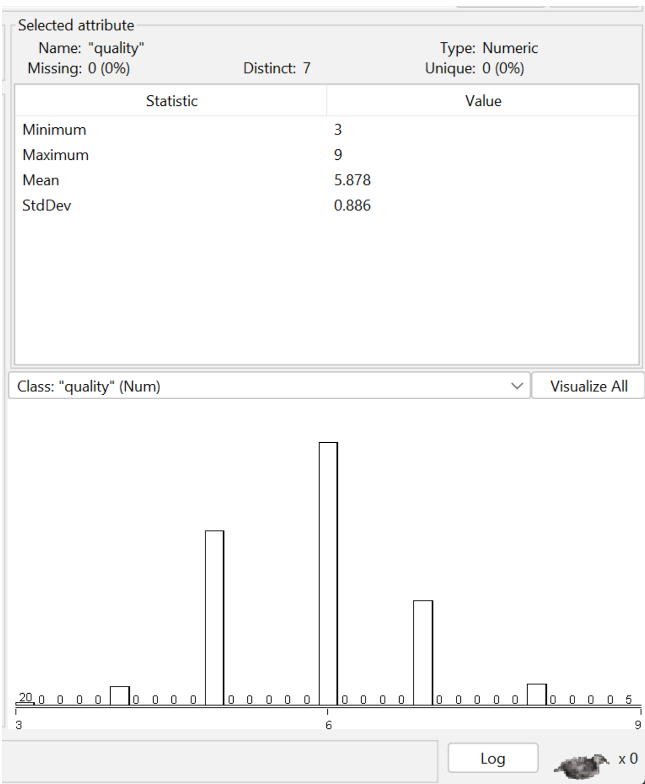


Figure 1 White wine dataset in WEKA

The Wine Quality datasets contain chemical and physical properties of both red and white Portuguese wine. Each instance of the wines has these 11 attributes that including fixed and volatile acidity, citric acid, residual sugar, chlorides, free sulphur dioxide, total sulphur dioxide, density, pH, sulphates and alcohol.
The 12th attribute being the target class, quality, which is an integer score and has a maximum of 9 and a minimum of 3.

The datasets show that there is a large proportion of the dataset that leans heavily towards the middle of the quality integer scale as shown in Figure 1 and Figure 2.

The class distribution is unbalanced. This poses a problem for training the model using this dataset. Without many other samples that are on either end of the scale the model will become 'lazy' and score the quality of majority of the wine samples it receives as being in the 5-6 range even thought this may not be the case in reality.

Even though the model may seem to be predicting the quality with high accuracy, the output becomes deceptive in the end.(*How to Handle Imbalanced Classes in Machine Learning*, 17:39:26+00:00)

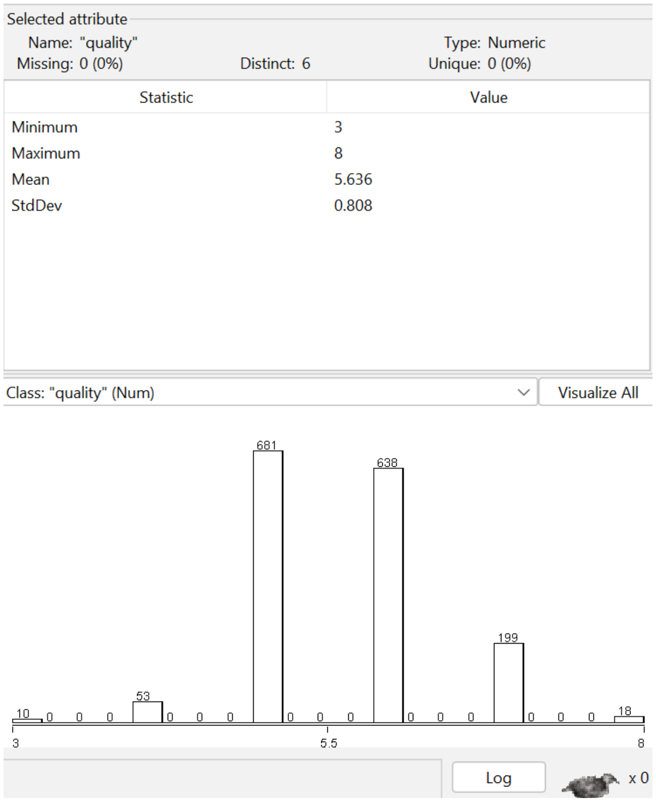


Figure 2 Red Wine Dataset in WEKA

Instead of treating this as a regression, problem where the model is trained to predict the numerical value of the quality of the wine samples, the values will be binned into class labels.

That being 'low', 'medium', 'high'. Where 'low' is less than or equal to 5, 'medium' is equal to 6 and 'high' is equal to or more than 7.

This changes the training models that we'll be using and the outcome of the training. Binning the values into 3 categories creates classes that are less imbalanced than when we were using continuous values to predict the quality of the wines.

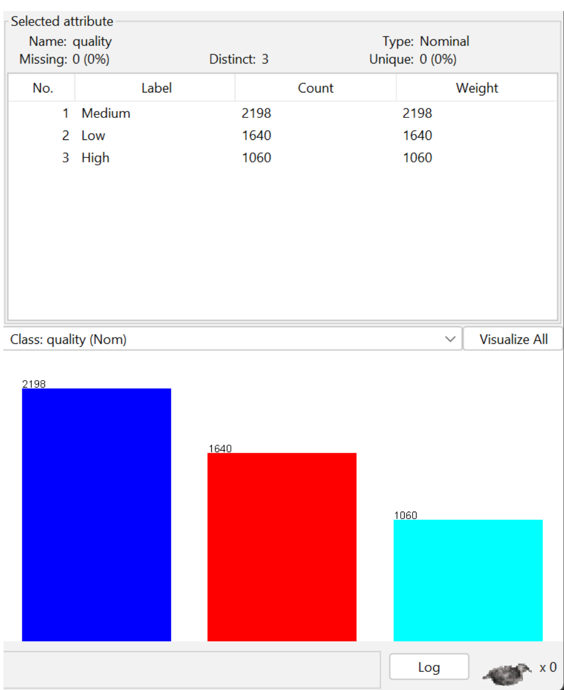


Figure 3 White wine binned quality class

In Figure 3 is a screenshot of the dataset after binning the values together into categories.
The class is still moderately imbalanced with the data skewing in favour of the 'Medium' group and the 'High' class having the least (making up about 21% of the group).

In Figure 4 there dataset shows a larger imbalance than with the white wine dataset even after binning the values into categories of low, medium and high.

Pearson Correlation Evaluation

Examining the relationship between the input features within the datasets to find the feature that correlates the least with the target attribute is another method of removing the 'noise' within a dataset.

The features that correlates the least with the target attribute is the most likely to add to the noise when training a model. However, this method only works if the relationship between two features is linear and if they aren't then the results may not be accurately captured.

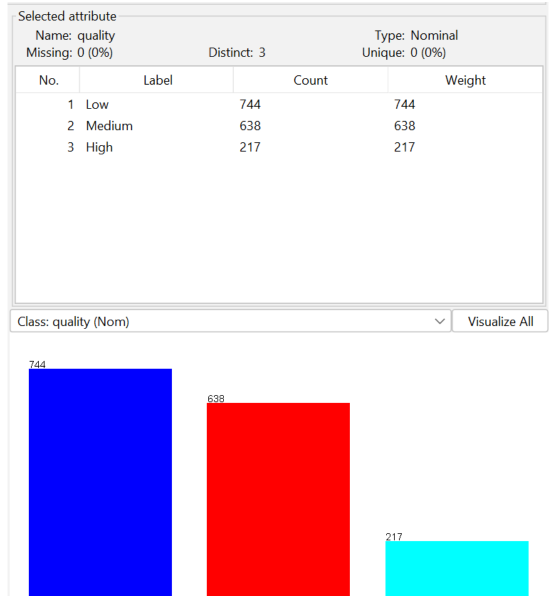


Figure 4 Red Wine binned quality class

Pearson correlation can be used to create heatmaps that show visually which features are most useful while also showing which ones contribute the least to the prediction of the target attribute.

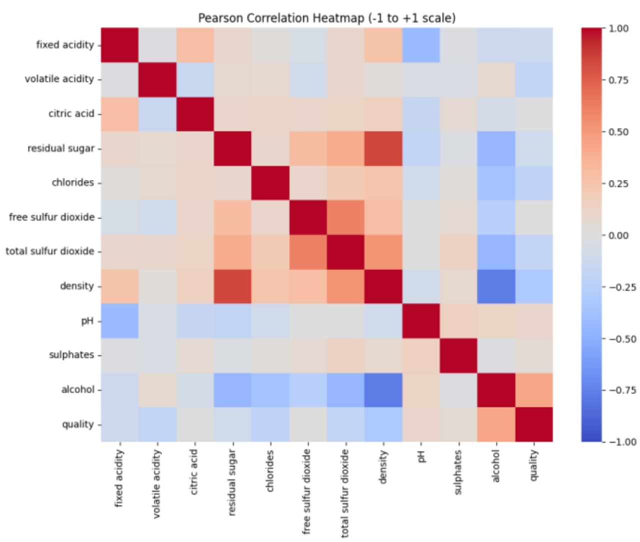


Figure 5 Pearson Correlation Heatmap for White Wine Dataset

Most of the input features seem to correlate negatively to the target attribute while 'alcohol' has the strongest positive correlation relationship and 'density' has the strongest negative correlation relationship.

The input feature with the least correlation to the target attribute in the white wine dataset seems to be 'citric acid' and 'free sulphur dioxide'.

Figure 6 is a bar graph that shows the correlation relationship between the target attribute and the other input features individually.

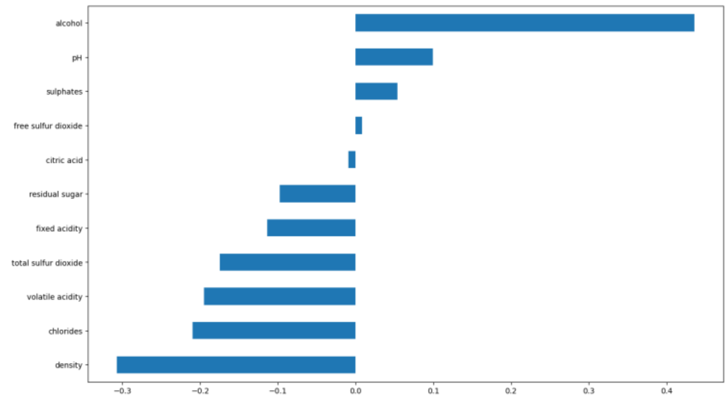


Figure 6 White wine correlation Bar graph

The results in the red wine dataset differ. The feature that correlates the least to the target attribute is 'Residual Sugar' and 'Free Sulphur dioxide.

The strongest correlation is similar to that of the white wine. That being 'alcohol' in the positive direction but 'volatile acidity' in the negative direction.

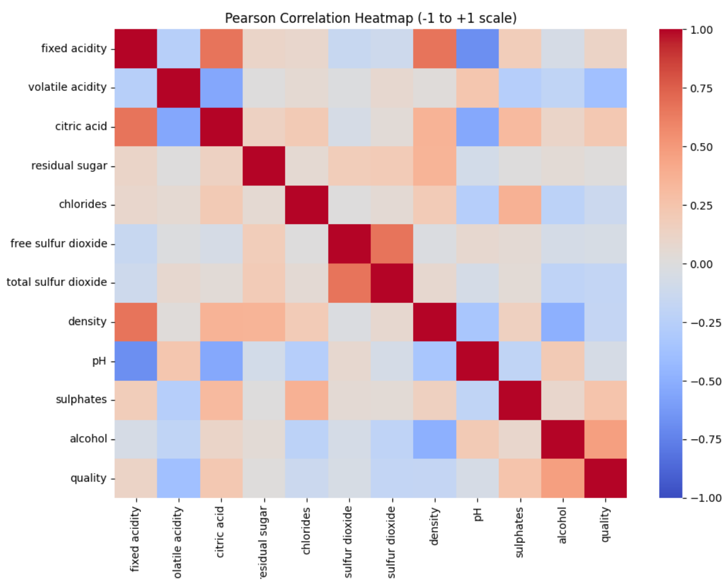
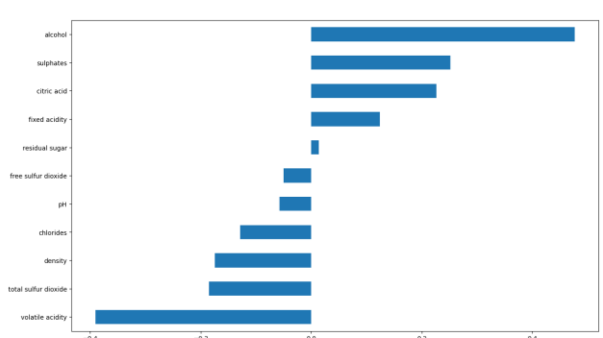


Figure 7 Pearson Correlation Heatmap for Red Wine Dataset

Even though this gives us an idea of what features strongly contribute to the predictability of the quality of wine samples, it doesn't give the full picture of the usefulness of some features and how much noise others add to the dataset.

Figure 8 Red wine correlation Bar graph



ANOVA F-test

ANOVA F test F and P interpretation

F-value range	Meaning	Feature usefulness
F > 1000	Exceptionally strong separation	Extremely informative
100–1000	Strong separation	Very useful feature
10–100	Moderate separation	Likely helpful
1–10	Weak separation	Possibly low-value
< 1	No meaningful separation	Likely useless

p-value	Interpretation	Keep the feature?
< 0.001	Extremely significant	Yes
0.001 – 0.01	Strongly significant	Yes
0.01 – 0.05	Statistically significant	Probably yes
0.05 – 0.1	Marginal	Caution
> 0.1	Not significant	Likely remove

Figure 9 F and P value interpretations

This test measures how influential input features are on the class output. The results come in two values, F and P, where F calculates how strong the relationship between the input feature and the class output is and P shows the confidence of that relationship.

F ranges between 0 and 1000, P ranges between 0 and 1.

The general rule of thumb for a good predictor feature is a larger F value and smaller P value will indicate a predictor that's very useful.

feature	F_score	p_value
alcohol	670.5277538	4.24E-258
density	306.7878701	2.99E-126
volatile acidity	131.9444824	1.54E-56
chlorides	126.2959859	3.30E-54
total sulfur dioxide	103.2474692	1.20E-44
residual sugar	41.08419187	2.02E-18
pH	28.50513193	4.92E-13
fixed acidity	26.35061825	4.14E-12
sulphates	8.854343881	0.00014506
citric acid	3.502183659	0.030207037
free sulfur dioxide	1.500438624	0.223134871

Figure 10 ANOVA results on White wine features

feature	F_score	p_value
alcohol	279.5798637	8.01E-105
volatile acidity	119.2912569	5.22E-49
sulphates	54.49441031	1.28E-23
total sulfur dioxide	48.37548056	4.01E-21
citric acid	45.0808542	9.01E-20
density	28.98090187	4.33E-13
fixed acidity	14.12579013	8.30E-07
chlorides	12.75336028	3.20E-06
free sulfur dioxide	5.290263283	0.005129204
pH	2.925720534	0.053913719
residual sugar	2.190839432	0.112159025

Figure 11 ANOVA results on Red wine dataset

Residual Sugar within the red wine dataset and free sulphur dioxide within the white wine dataset both have very low F-values and high P-values.

Along with the scores they got from the Pearson correlation heatmap, this shows that they are likely to add noise to the datasets and are candidates for removal from the list of features.

WEKA Modelling

DECISION TREE (J48)

The accuracy of the correctly classified instances is similar for both the red and white wine datasets, white wine being 64% and red wine being 66%.

```
Time taken to build model: 0.16 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        3135               64.0057 %
Incorrectly Classified Instances      1763               35.9943 %
Kappa statistic                          0.4404
Mean absolute error                      0.2585
Root mean squared error                  0.4537
Relative absolute error                 60.6129 %
Root relative squared error             98.2504 %
Total Number of Instances             4898

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.614    0.275    0.645      0.614    0.629      0.341    0.692     0.619     Medium
                 0.692    0.173    0.668      0.692    0.680      0.514    0.785     0.611     Low
                 0.613    0.119    0.588      0.613    0.600      0.487    0.790     0.502     High
Weighted Avg.    0.640    0.207    0.640      0.640    0.640      0.431    0.744     0.591

=== Confusion Matrix ===

    a    b    c   <-- classified as
 1350  481  367 |    a = Medium
  416 1135   89 |    b = Low
  326   84  650 |    c = High
```

Figure 12 White wine Decision Tree

The red wine is slightly more accurate in classifying, however the red wine dataset has a lot fewer instances than the white wine dataset.

The macro-f1 score of the white wine dataset is 0.636 vs the red wine's 0.626. This shows that the white wine model may have a lower accuracy but performs better when classifying across all classes.

```
Time taken to build model: 0.06 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        1054               65.9162 %
Incorrectly Classified Instances       545               34.0838 %
Kappa statistic                          0.4322
Mean absolute error                      0.2495
Root mean squared error                  0.436
Relative absolute error                 61.7542 %
Root relative squared error             97.0165 %
Total Number of Instances             1599

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.770    0.261    0.720      0.770    0.744      0.508    0.787     0.702     Low
                 0.578    0.248    0.608      0.578    0.593      0.334    0.683     0.572     Medium
                 0.516    0.061    0.571      0.516    0.542      0.476    0.803     0.455     High
Weighted Avg.    0.659    0.228    0.655      0.659    0.656      0.434    0.748     0.617

=== Confusion Matrix ===

   a   b   c   <-- classified as
 573 154  17 |   a = Low
 202 369  67 |   b = Medium
  21  84 112 |   c = High
```

Figure 13 Red wine Decision Tree

The f1 scores for the medium and high score are much lower in the red wine dataset which would decrease the macro-f1 score.

The white wine dataset took longer to train, 0.16 seconds as opposed to 0.06 seconds in the red wine, likely because it's a larger dataset.

LOGISTIC REGRESSION

```
Time taken to build model: 0.16 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        2803               57.2274 %
Incorrectly Classified Instances      2095               42.7726 %
Kappa statistic                          0.3046
Mean absolute error                      0.3586
Root mean squared error                  0.4237
Relative absolute error                 84.0923 %
Root relative squared error             91.7529 %
Total Number of Instances             4898

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.679    0.486    0.532      0.679    0.597      0.194    0.629     0.551     Medium
                 0.577    0.159    0.647      0.577    0.610      0.431    0.799     0.655     Low
                 0.343    0.069    0.579      0.343    0.431      0.338    0.793     0.513     High
Weighted Avg.    0.572    0.286    0.581      0.572    0.565      0.305    0.722     0.578

=== Confusion Matrix ===

    a    b    c   <-- classified as
 1493  470  235 |    a = Medium
  664  946   30 |    b = Low
  649   47  364 |    c = High
```

Figure 14 White wine Logistic Regression

With this model, the accuracy of prediction is higher with the red wine dataset than with the white wine dataset, 63.4% compared with 57.2%.

The macro-f1 scores of the white wine and red wine being 0.546 and 0.575 respectively. The red wine model performs better overall in terms of accuracy and classifying.

```
Time taken to build model: 0.05 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        1013               63.3521 %
Incorrectly Classified Instances       586               36.6479 %
Kappa statistic                          0.3772
Mean absolute error                      0.317
Root mean squared error                  0.3998
Relative absolute error                 78.4562 %
Root relative squared error             88.9595 %
Total Number of Instances             1599

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.773    0.276    0.709      0.773    0.740      0.496    0.815     0.762     Low
                 0.572    0.307    0.553      0.572    0.562      0.264    0.678     0.544     Medium
                 0.336    0.040    0.570      0.336    0.423      0.374    0.870     0.490     High
Weighted Avg.    0.634    0.256    0.628      0.634    0.626      0.387    0.768     0.638

=== Confusion Matrix ===

   a   b   c   <-- classified as
 575 163   6 |   a = Low
 224 365  49 |   b = Medium
  12 132  73 |   c = High
```

Figure 15 Red wine Logistic Regression

The time taken to build the model is a lot shorter, 0.05 seconds, while the white wine model is 0.16 seconds.

MULTILAYER PERCEPTRON

```
Time taken to build model: 3.85 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        2823               57.6358 %
Incorrectly Classified Instances      2075               42.3642 %
Kappa statistic                          0.313
Mean absolute error                      0.3459
Root mean squared error                  0.4225
Relative absolute error                 81.11   %
Root relative squared error             91.5046 %
Total Number of Instances             4898

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.671    0.472    0.537      0.671    0.596      0.200    0.628     0.552     Medium
                 0.593    0.165    0.644      0.593    0.618      0.438    0.806     0.653     Low
                 0.354    0.069    0.587      0.354    0.441      0.348    0.815     0.529     High
Weighted Avg.    0.576    0.282    0.584      0.576    0.570      0.312    0.728     0.581

=== Confusion Matrix ===

    a    b    c   <-- classified as
 1475  491  232 |    a = Medium
  635  973   32 |    b = Low
  639   46  375 |    c = High
```

Figure 17 White wine Multilayer Perceptron

```
Time taken to build model: 1.11 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         984               61.5385 %
Incorrectly Classified Instances       615               38.4615 %
Kappa statistic                          0.3532
Mean absolute error                      0.3047
Root mean squared error                  0.4056
Relative absolute error                 75.4232 %
Root relative squared error             90.2514 %
Total Number of Instances             1599

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.722    0.253    0.713      0.722    0.717      0.469    0.815     0.761     Low
                 0.577    0.340    0.529      0.577    0.552      0.234    0.673     0.530     Medium
                 0.364    0.052    0.523      0.364    0.429      0.365    0.856     0.459     High
Weighted Avg.    0.615    0.260    0.614      0.615    0.612      0.361    0.764     0.628

=== Confusion Matrix ===

   a   b   c   <-- classified as
 537 200   7 |   a = Low
 205 368  65 |   b = Medium
  11 127  79 |   c = High
```

Figure 16 Red wine Multilayer Perceptron

The last models both took a lot longer to build than the previous 2 models did. 3.85 seconds for the white wine model and 1.11 seconds for the red wine model. This indicates that the last model uses a lot more resources to build.

The white wine model has a lower accuracy than the red wine model as well. The macro-f1 of the models shows that the red wine model still outperforms the white wine model (0.566 and 0.552 respectively).

STANDARD NEURAL NETWORKS

Learning Curves

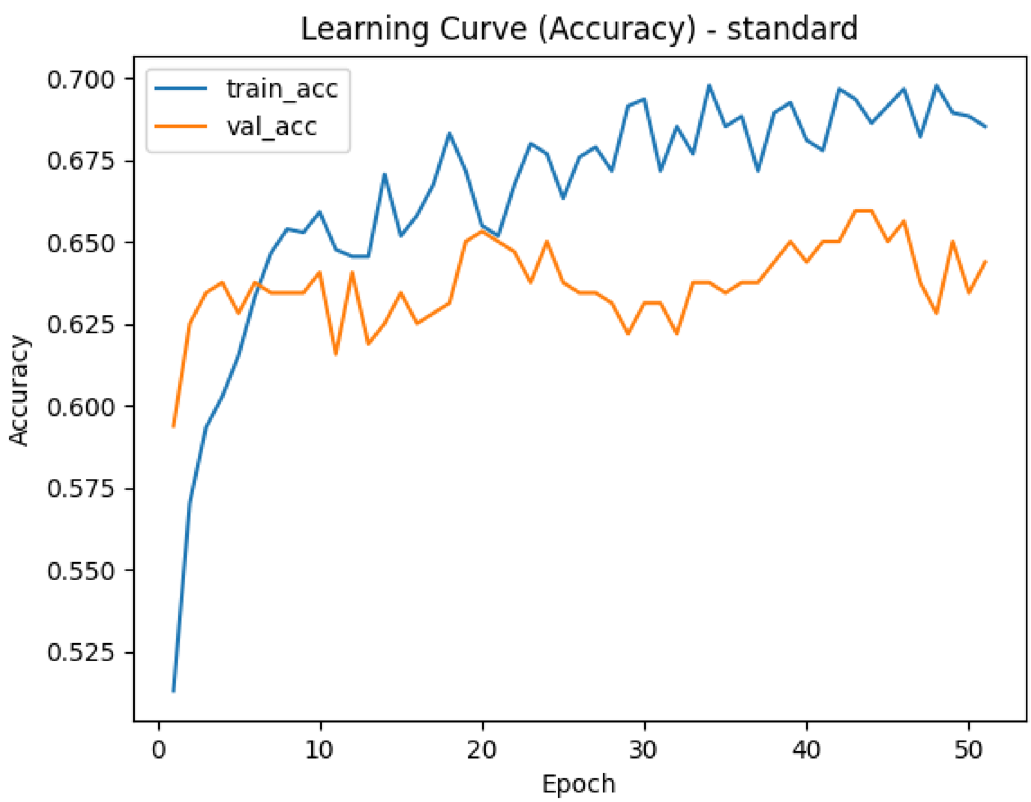


Figure 18 Standard NN Learning Curve Red Wine

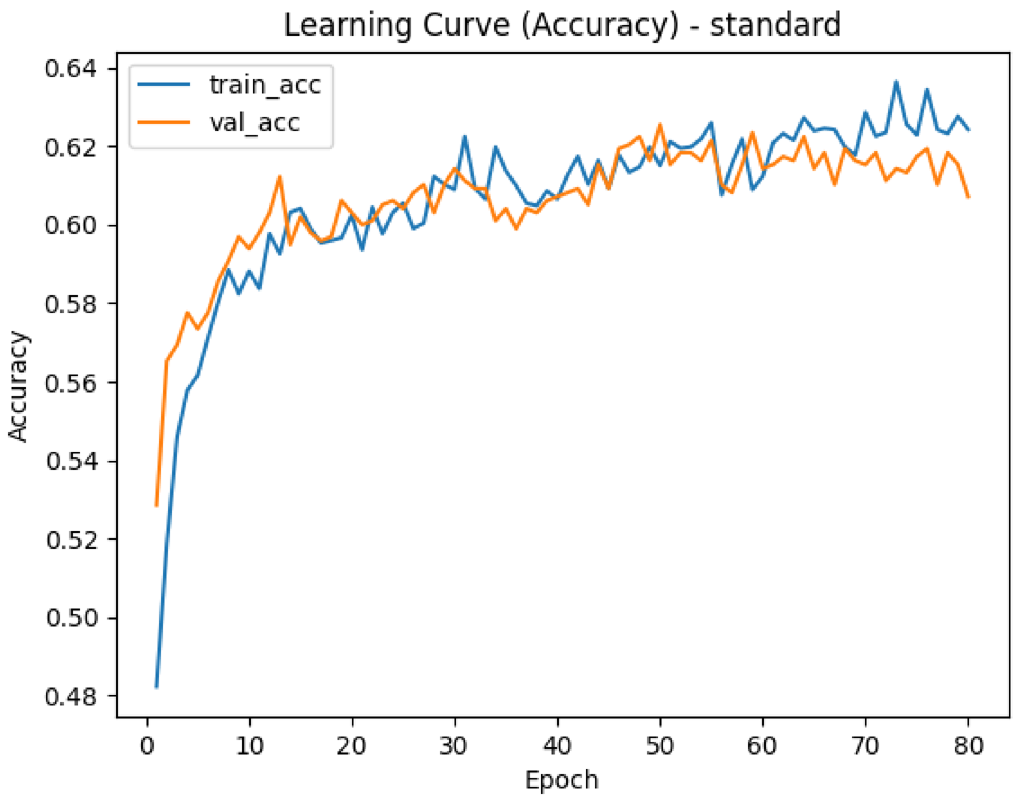


Figure 19 Standard NN Learning Curve White Wine

Confusion Matrices

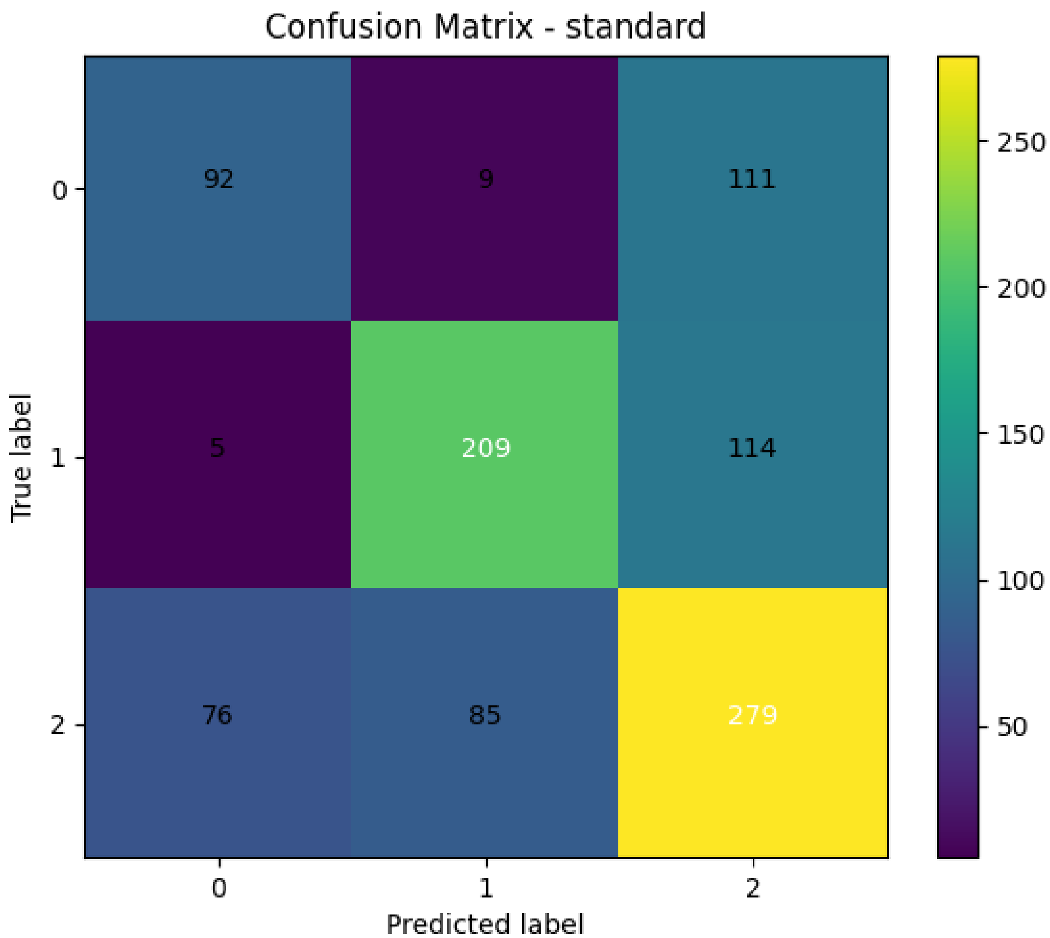


Figure 20 Standard NN Confusion Matrix White Wine

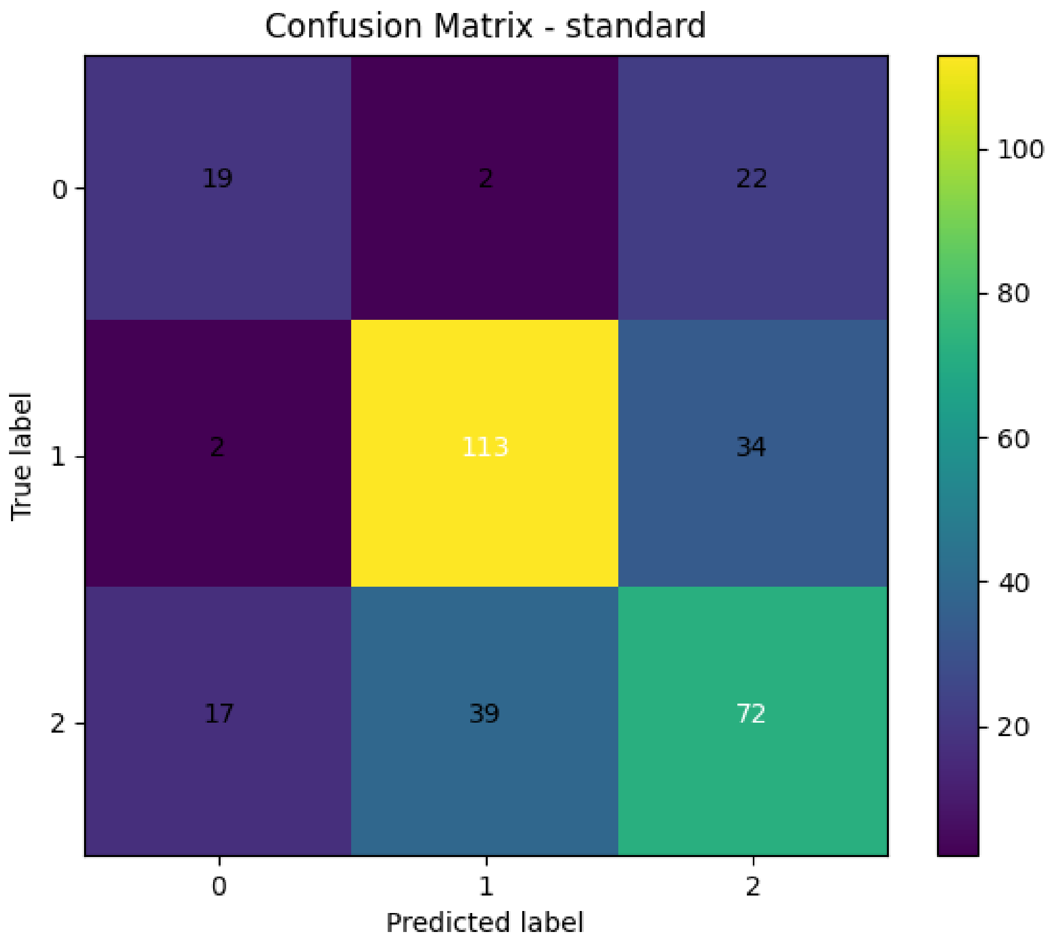


Figure 21 Standard NN Confusion Matrix Red Wine

Performance Statistics

=== Standard RESULTS ===

Accuracy: 0.5918

Precision (macro): 0.5917

Recall (macro): 0.5684

F1 (macro): 0.5772

	Precision	Recall	F-1 Score
High	0.5318	0.4340	0.4779
Medium	0.6898	0.6372	0.6624
Low	0.5536	0.6341	0.5911

Figure 22 White Wine Performance Metrics

=== Standard RESULTS ===

Accuracy: 0.5938

Precision (macro): 0.5364

Recall (macro): 0.5305

F1 (macro): 0.5326

	Precision	Recall	F-1 Score
High	0.3947	0.3488	0.3704
Medium	0.7063	0.7584	0.7314
Low	0.5082	0.4844	0.4960

Figure 23 Red Wine Performance Metrics

Learning Curves

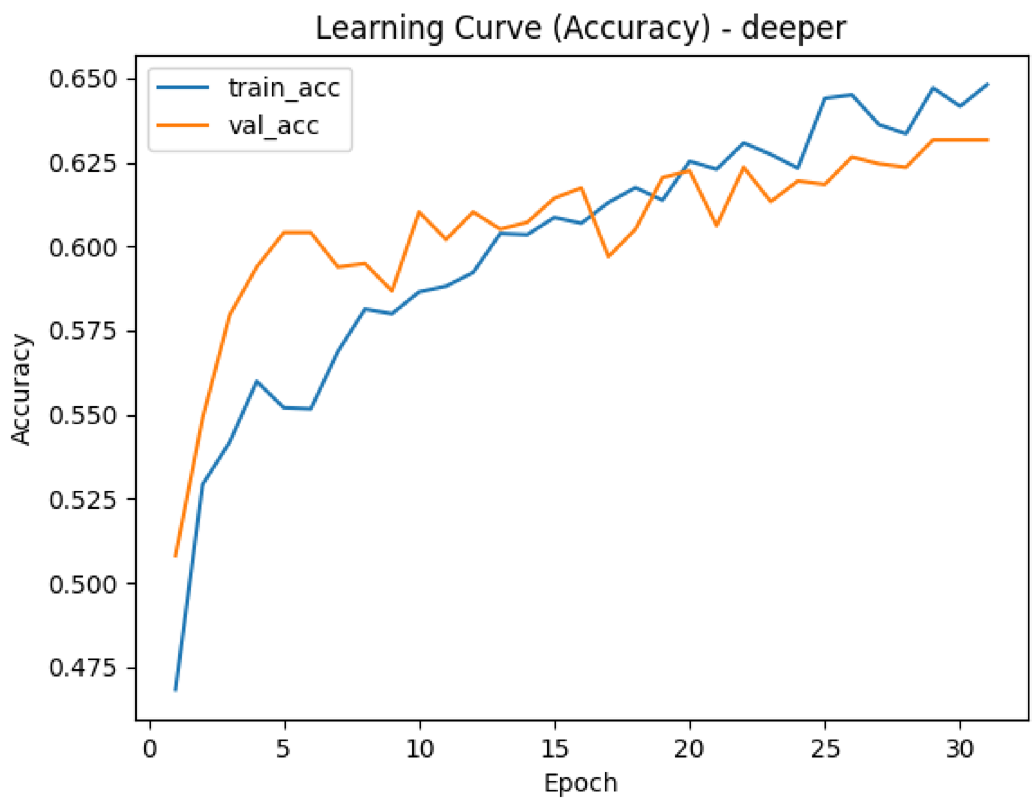


Figure 24 White Wine Deep NN Learning Curve

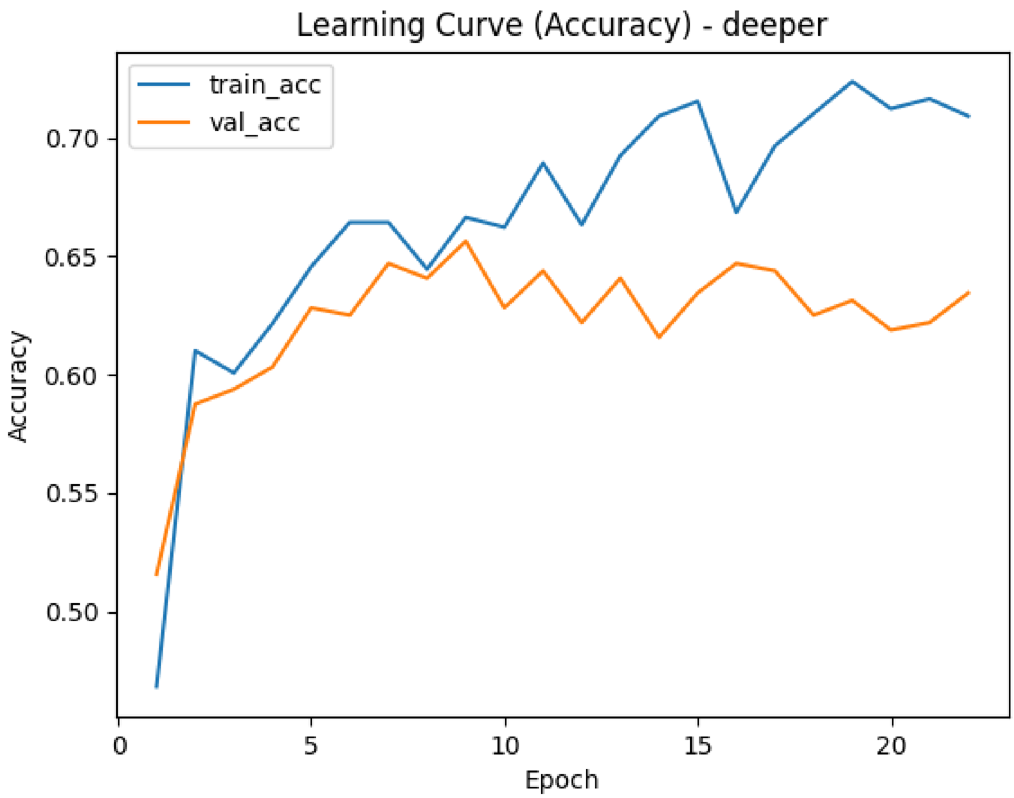


Figure 25 Red Wine Deep NN Learning Curve

Confusion Matrices

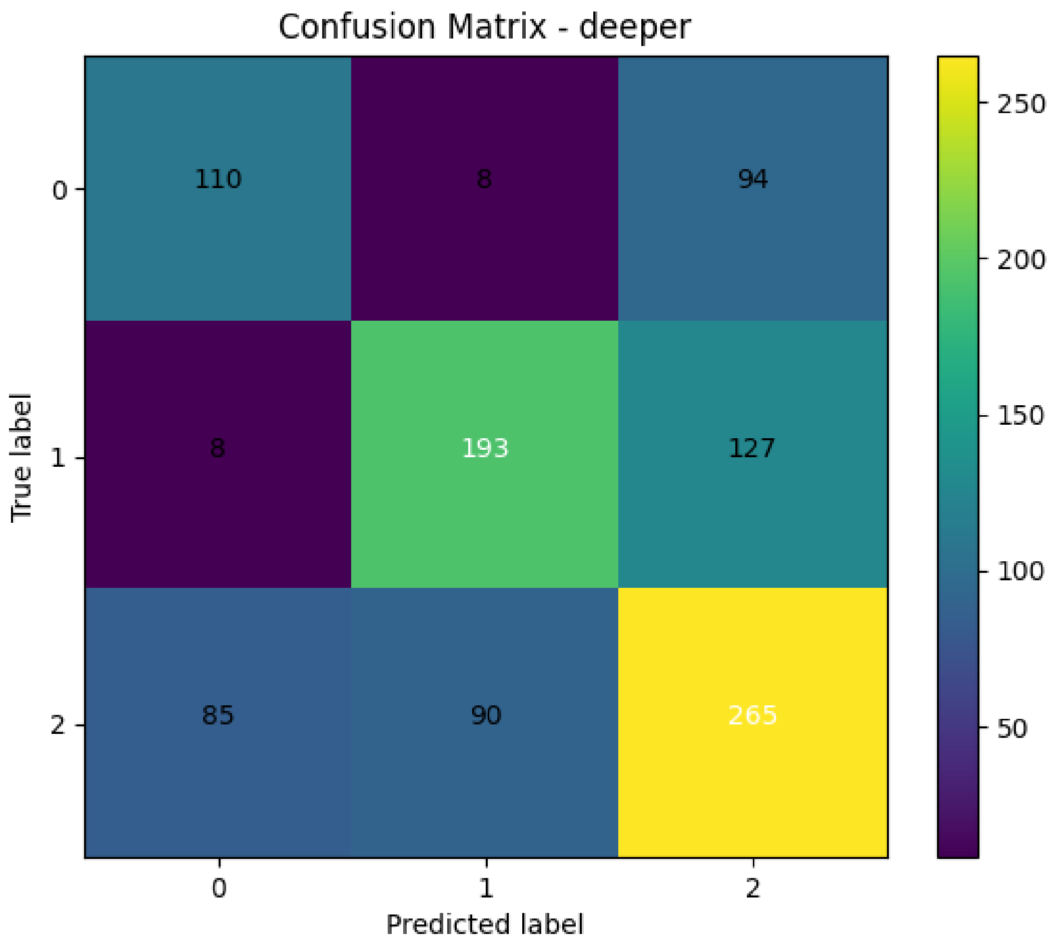


Figure 26 White Wine Deep NN Confusion Matrix

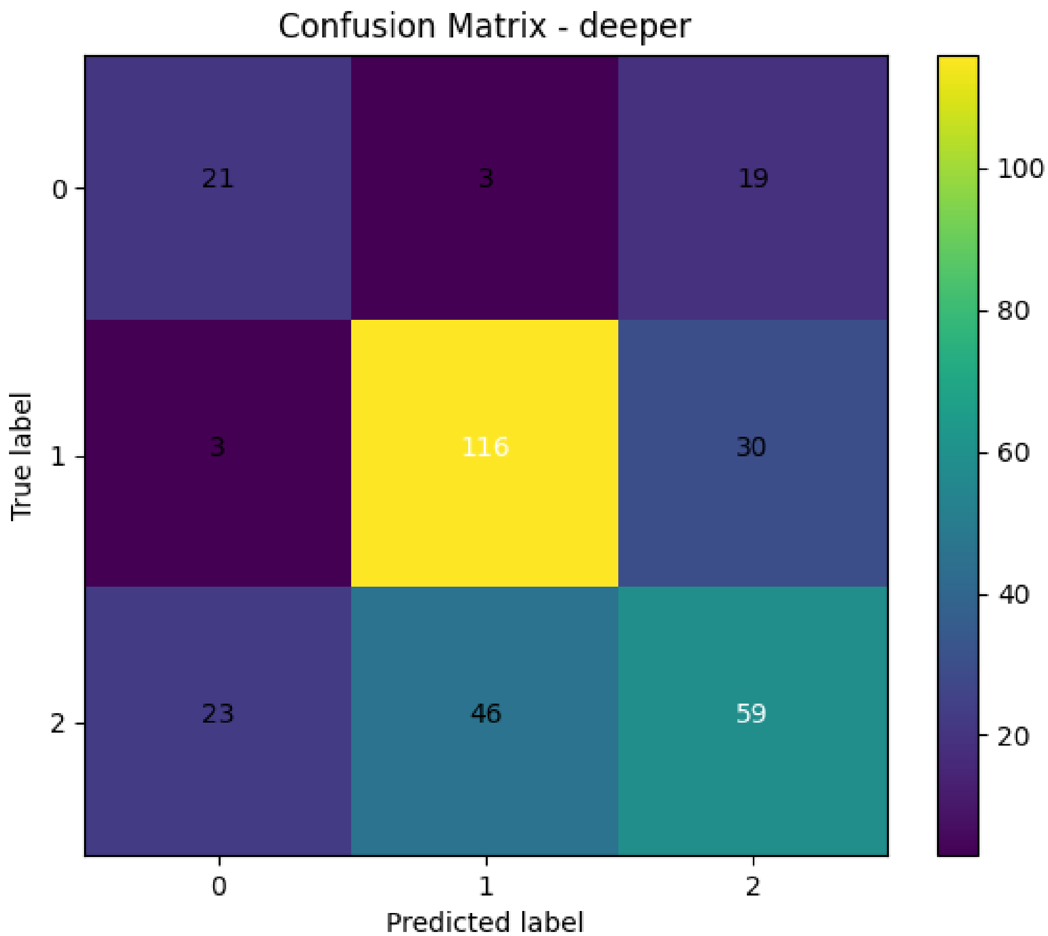


Figure 27 Red Wine Deep NN Confusion Matrix

Performance Metrics

=== Deeper RESULTS ===

Accuracy: 0.5796

Precision (macro): 0.5835

Recall (macro): 0.5699

F1 (macro): 0.5754

	Precision	Recall	F-1 Score
High	0.5419	0.5189	0.5301
Medium	0.6632	0.5884	0.6236
Low	0.5453	0.6023	0.5724

Figure 28 White Wine Performance Metrics

=== Deeper RESULTS ===

Accuracy: 0.6500

Precision (macro): 0.6049

Recall (macro): 0.6031

F1 (macro): 0.6040

	Precision	Recall	F-1 Score
High	0.4762	0.4651	0.4706
Medium	0.7434	0.7584	0.7508
Low	0.5952	0.5859	0.5906

Figure 29 Red Wine Performance Metrics

Model Performance Reflection

Overall, both the standard neural network and the deep neural network achieved moderate performance, with accuracies generally ranging between 58% and 65%. The models consistently performed best on the Medium wine quality class, as shown by higher precision, recall, and F1-scores, while High and Low classes were more difficult to classify. This imbalance is also visible in the confusion matrices, where misclassifications between adjacent quality classes were common. Comparing architectures, the deep neural network showed a noticeable improvement for the red wine dataset, achieving the highest accuracy (65.0%) and improved macro-averaged metrics, suggesting that the deeper model was better able to capture more complex patterns in that dataset. However, for white wine, the deep model did not significantly outperform the standard network, indicating diminishing returns from increased model complexity.

Use of AI Assistance

AI assistance was used primarily to generate the initial neural network code, including model architecture definitions, training loops, and evaluation metrics such as accuracy, precision, recall, F1-scores, learning curves, and confusion matrices. This allowed me to focus more on interpreting results and understanding model behaviour rather than implementing boilerplate code from scratch.