CS 4120 – MACHINE LEARNING, DATA MINING PROJECT PROPOSAL

AUTHORS:

- **CHIZURUM EWELIKE**
- **LAYO FALOSEYI**

WINE QUALITY

Section 1.0: Overview

Expert tasters often judge wine quality, but their evaluations can be subjective and time-consuming. By building machine learning models, we can create a reproducible, data-driven way to evaluate wine quality, helping wine producers maintain consistency, supporting automated quality control, and benefiting consumers who want reliable indicators of wine quality.

Section 2.0: Dataset Description

This project uses the Wine Quality Dataset from the UCI Machine Learning Repository (Cortez et al., 2009), released under a Creative Commons license. The Wine Quality dataset consists of 6,497 Portuguese "Vinho Verde" wine samples, including 1,599 red and 4,898 white wines, each described by 11 physicochemical input variables such as acidity, residual sugar, pH, sulphates, and alcohol. The target variable is a sensory quality score between 0 and 10, rated by wine experts. No missing values are reported, and there are no sensitive attributes since the data is purely chemical and sensory.

Section 3.0: Tasks

This project will address a classification and a regression problem using the Wine Quality dataset.

- **Classification Task:** The wine quality score (0–10) will be converted into a binary label: High Quality (≥7) vs. Low Quality (<7). This derivation follows standard practice in prior studies, balancing interpretability with the class distribution.
- **Regression Task:** The original quality score (0–10) will be predicted directly as a continuous variable, based on expert sensory ratings.

Both tasks are feasible because the dataset provides a numerical target that can be framed as a categorical outcome or a continuous score. This allows us to explore and compare supervised learning approaches on the same dataset. We frame this as <P, T, E>: Performance (P) = Accuracy/F1 & MAE/RMSE, Task (T) = predict label and score, Experience (E) = the wine physicochemical measurements.

Section 4.0: Metrics Plan

We will use standard metrics for classification and regression tasks to evaluate model performance. We will use a stratified train/validation/test split with feature scaling for models that require it and k-fold cross-validation for model selection; final metrics will be reported on a held-out test set.

- **Classification:** Models will be assessed using Accuracy (overall proportion of correct predictions) and F1-score (harmonic mean of precision and recall, which is helpful for imbalanced classes). We will primarily use Accuracy and F1, since the classes are imbalanced. ROC-AUC may also be reported as a supplementary metric.
- **Regression:** Models will be evaluated using Mean Absolute Error (MAE), which captures average prediction error in the same units as the target score, and Root Mean Squared Error (RMSE), which penalizes larger errors more heavily.

Section 5.0: Baseline Plan (Classical ML Models)

We will implement at least two baseline models for each task from the required set of classical machine learning methods.

- **Classification:** We will use Logistic Regression and a Decision Tree Classifier. Logistic Regression provides a strong linear baseline, while Decision Trees can capture nonlinear relationships and feature interactions.

- **Regression:** We will use Linear Regression and a Decision Tree Regressor. Linear Regression offers an interpretable baseline, and Decision Trees provide flexibility in modelling complex variable effects.

These baselines will be trained and evaluated at the Midpoint stage, as the foundation for comparing neural network models in the Final report. We will implement a small multilayer perceptron (2 hidden layers, 32–64 units, ReLU) with Adam optimizer and early stopping on validation loss. Inputs will be standardized.

Section 6.0: Reproducibility Plan

We will use a GitHub repository with a clear folder structure and a requirements.txt file that pins exact dependency versions to ensure reproducibility. In addition, we will use MLflow tracking to log model parameters, metrics, and outputs across experiments. This setup will allow our results to be consistently reproduced from the repository.

Section 7.0: Tables

Table 1 – Dataset Snapshot

Rows	Columns	Target Descriptions	% Missing (top 5)	Class distribution
6,497 total (1,599 red + 4,898 white)	12 (11 features + 1 target)	Classification: High (≥7) vs Low (<7) Quality; Regression: Quality Score (0–10)	0% across all columns	Low (<7): 80.3%; High (≥7): 19.7%

Table 2 – Planned Models and Metrics

Task	Baseline models	Metrics (Midpoint)	Metrics (Final)
Classification	Logistic Regression; Decision Tree	Accuracy; F1	Accuracy; F1
Regression	Linear Regression; Decision Tree Regressor	MAE; RMSE	MAE; RMSE

Section 8.0: References

- Cortez, P., Cerdeira, A., Almeida, F., Matos, T., & Reis, J. (2009). Modelling wine preferences by data mining from physicochemical properties. Decision Support Systems, 47(4), 547–553. https://www.sciencedirect.com/
- UCI Machine Learning Repository. (2009). Wine Quality Data Set. University of California, Irvine. http://archive.ics.uci.edu/ml/datasets/Wine+Quality
- GitHub Repository: https://github.com/LAYO200/Wine-Quality-Prediction-models
- Use of AI - ChatGPT and Grammarly were used to check grammar and improve language clarity. No AI tools were used for analysis, interpretation, or generation of core content.