Analysis of Computation Time vs Performance Trade-Off in a Portuguese Wine Quality Classification Problem

Miguel Silvério
Universidade de Évora
m55661@uevora.alunos.pt

Abstract

This study aims to explore the trade-offs between computational efficiency and predictive performance in classifying wine quality using physicochemical attributes. Using a dataset sourced from the UCI Machine Learning Repository, three classification algorithms were employed to evaluate the impact of dataset size, feature selection, and algorithm complexity on model outcomes. Decision Trees, Random Forests, and Support Vector Machines (SVMs) were selected for evaluation based on their performance in terms of classification and, training and inference time. The analysis revealed a expected relationship between dataset size and model performance, concluding that larger datasets generally lead to more accurate classifications but at the expense of longer computation times. This study highlights the importance of balancing computational and predictive considerations in the deployment of machine learning models for wine quality assessment and similar classification problems.

1 Introduction

Wine quality evaluation has traditionally relied on human tasters who assess sensory attributes such as taste, aroma, and texture. However, human evaluation is inherently subjective, prone to inconsistency. To address these limitations, researchers have explored using physicochemical properties of wine to model and predict sensory preferences, taking advantage of data mining techniques.

The study by Cortez et al. "Modeling wine preferences by data mining from physicochemical properties" [2] proposed a data mining approach to predict human wine preferences based on physicochemical tests conducted during wine certification. Their work highlighted three main regression techniques: multiple regression (MR), neural networks (NN), and support vector machines (SVM). Their findings demonstrated that SVM outperformed other models in accuracy due to its robustness and ability to handle non-linear relationships. They also emphasized the importance of variable selection in model performance.

Building on the work of Cortez et al. [2], this study investigates the quality prediction problem as a classification problem instead of regression and also does it using different well known classification algorithms like decision trees and random forests besides SVMs.

Using a dataset sourced from the UCI Machine Learning Repository, this study aims to evaluate the computational efficiency and predictive accuracy of these algorithms. The study also investigates the impact of feature selection, dataset size, and data imbalance on model performance, ultimately contributing to a better understanding of the trade-offs between computation time and predictive performance in wine quality classification.

2 Data

2.1 Dataset Description

The Wine Quality dataset [1], sourced from the UCI Machine Learning Repository, provides physicochemical properties of wines produced in the north region of Portugal. The dataset consists of two

subsets: red wine, comprising 1,599 samples, and white wine, with 4,898 samples. Each sample is characterized by 11 numerical physicochemical attributes and a quality score ranging from 0 (low) to 10 (high). These quality scores are derived from sensory evaluations conducted by wine tasters, making this dataset specially suited for classification tasks. [3]

The physicochemical attributes include:

- **Fixed Acidity**: Natural acids in wine, such as tartaric, malic, citric, and succinic acids, which contribute to the wine's acidity and flavor profile. Most of these acids originate from grapes, except succinic acid, which is produced during fermentation.
- **Volatile Acidity**: Fatty acids, primarily acetic acid, that impart a vinegar-like smell and influence the wine's sensory characteristics.
- **Citric Acid**: A mild organic acid used to enhance acidity, freshness, and prevent cloudiness, though it may also encourage microbial growth.
- **Residual Sugar**: The leftover grape sugars that remain unfermented, contributing to the sweetness of the wine.
- **Chlorides**: The sodium chloride content in wine, which adds a slight saltiness to its taste.
- **Free Sulfur Dioxide**: The active form of sulfur dioxide, acting as a preservative and antioxidant to maintain the wine's freshness.
- **Total Sulfur Dioxide**: The sum of free and chemically bound sulfur dioxide, which works to protect the wine during storage and aging.
- **Density**: A measure of the wine's mass relative to its volume, affected by the balance of sugar and alcohol content.
- **pH**: A measure of the wine's acidity level, with lower pH values indicating higher acidity.
- **Sulphates**: Compounds like potassium sulphate that contribute to wine preservation and its flavor complexity.
- **Alcohol**: The ethanol content in wine, which significantly impacts its structure, body, and character.

2.2 Relevance of the Dataset

This dataset is particularly significant for machine learning applications due to its diverse set of attributes and practical relevance, making it an excellent resource for developing and testing quality prediction models.

In addition to the work by Cortez et al. [2], other researchers have utilized this dataset in their studies. For example, Qingwen Zeng explored ensemble learning techniques to predict wine quality in the article titled "Prediction of Wine Quality Using Ensemble Learning Approach of Machine Learning" [4]. Similarly, Nuriel Shalom Mor used the dataset in the study "Wine Quality and Type Prediction from Physicochemical Properties Using Neural Networks for Machine Learning: A Free Software for Winemakers and Customers" [3], focusing on predicting both wine quality and type using neural network models. These diverse applications highlight the dataset's versatility and its value for advancing research in wine quality prediction.

2.3 Data Preprocessing

The raw dataset was of high quality and required minimal preprocessing. It contained no missing values, making it ready for analysis with minimal adjustments. Initially, the red and white wine subsets were merged into a single dataset, and a new ”wine type” column was introduced to differentiate between the two types. This column was encoded numerically, with 0 representing red wine and 1 representing white wine.

The distribution of wine quality scores was highly imbalanced, with some extreme values being sparsely represented. To preserve the diversity of the dataset and maintain representation across all quality categories, no data samples were removed, even though some could be considered outliers. This decision aimed to retain as much information as possible for classification purposes.

2.4 Exploratory Data Analysis

For the Exploratory Data Analysis several methods were utilized to understand the relationships between physicochemical properties and wine quality. Descriptive statistics provided information into feature distributions, while correlation matrices analyzed the strength and direction of relationships between features and wine quality. Line plots were also used to show how features varied with wine quality with logarithmic ”y” scales to highlight subtle differences between features. Random Forest feature importance was used to quantify the predictive power of each feature, offering a hierarchical ranking for model selection.

Red and white wines were analyzed separately due to their distinct chemical compositions.

For red wines, alcohol had the strongest positive correlation with quality (0.48), followed by sulphates (0.25) and citric acid (0.23). Negatively correlated features included volatile acidity (-0.39) and chlorides (-0.13). Features like free and total sulfur dioxide showed minimal correlation with quality. The selected features for red wine models were alcohol, sulphates, volatile acidity, total sulfur dioxide, density, fixed acidity, and chlorides.

For white wines, alcohol again showed the strongest positive correlation (0.44), while density (-0.31), volatile acidity (-0.19), and chlorides (-0.21) had moderate negative correlations. Residual sugar, free sulfur dioxide, and pH had negligible impacts. Selected features for white wine models included alcohol, density, volatile acidity, total sulfur dioxide, chlorides, residual sugar, and pH.

Redundant or low-impact features like free sulfur dioxide were excluded, as its contribution was encapsulated by total sulfur dioxide.

The final combined feature set emphasized alcohol, sulphates, volatile acidity, density, total sulfur dioxide, and chlorides, with pH and residual sugar selectively retained for white wines.

3 Algorithms and Methodology

This study was conducted using Jupyter Notebooks and utilized several Python libraries, including pandas, numpy, seaborn, matplotlib, sklearn, and imblearn. Three classification algorithms — Decision Tree, Random Forest, and Support Vector Machines (SVMs) — were selected for evaluation based on their performance in terms of classification and, training and inference time. Decision Tree and Random Forest were chosen for their popularity and to explore methods not examined in the study by Cortez et al. SVMs were included due to their strong performance in the original study by Cortez et al.

To examine the effect of dataset size on model performance, the red and white wine datasets were processed separately, and subsets comprising 50% of the original sizes were also used. Additionally, the impact of feature selection was tested by training and evaluating the models on both the full feature set and the reduced set of selected features.

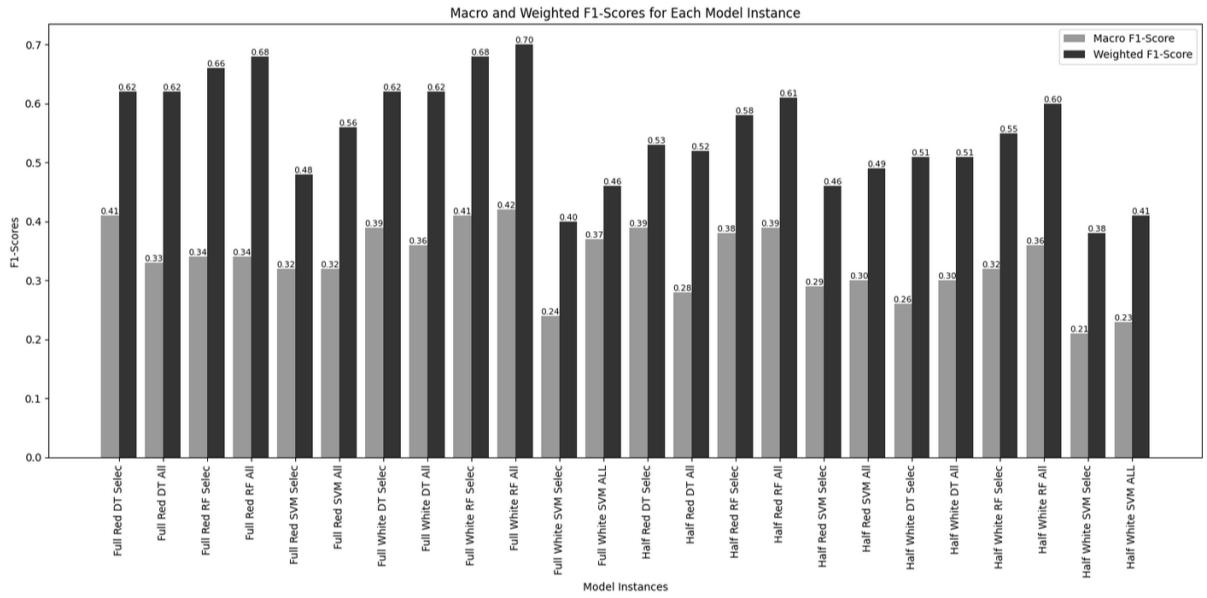

Figure 1: F1-Scores per model, categorized by Dataset Size (Full or Half), Wine Type (Red or White), Algorithm (Decision Tree, Random Forest, or SVM), and Feature Set (Selected or All).

The dataset was divided into training and testing sets, with 80% of the data allocated for training and 20% for testing. This split was performed using the train_test_split function, with the arguments test_size=0.2, random_state=1, and stratify=y to ensure the stratification of the unbalanced dataset.

Confusion matrices and classification reports were generated for all models, along with measurements of training and inference times. The Decision Tree classifiers were configured with random_state=1 and class_weight='balanced' to address class imbalances. For Random Forest, parameters were set as n_estimators=500, random_state=1, and class_weight='balanced'.

Since SVMs are sensitive to the scale of the input data, the features were standardized to a range between 0 and 1 using the StandardScaler from scikit-learn. To mitigate the effects of class imbalance further, SMOTE (Synthetic Minority Oversampling Technique) was applied to over-sample underrepresented classes. As the classification task involved multiple classes, the SVM models were configured with decision_function_shape='ovo' and class_weight='balanced'.

The primary performance metric used was the F1-score, which provides a more balanced measure of performance in the presence of class imbalance. A simple train-test split with stratification was used for validation. The 5x cross-validation used on the Cortez et. al. [2] study was not applied in this study to simplify and because of time constraints.

For computational performance analysis, a custom function was implemented to measure the time taken for model training and inference. Timers were activated at the start of training or inference and stopped on completion, providing data about the computational demands of each algorithm.

4 Results and Discussion

The F1-scores for the models were obtained from their classification reports and are illustrated in Figure 1.

From the analysis, it is evident that weighted F1-scores consistently score higher than macro F1-scores across all models. This outcome is expected given the dataset's imbalance, where weighted scores account for the prevalence of each class. However, the disparity between weighted and macro F1-scores

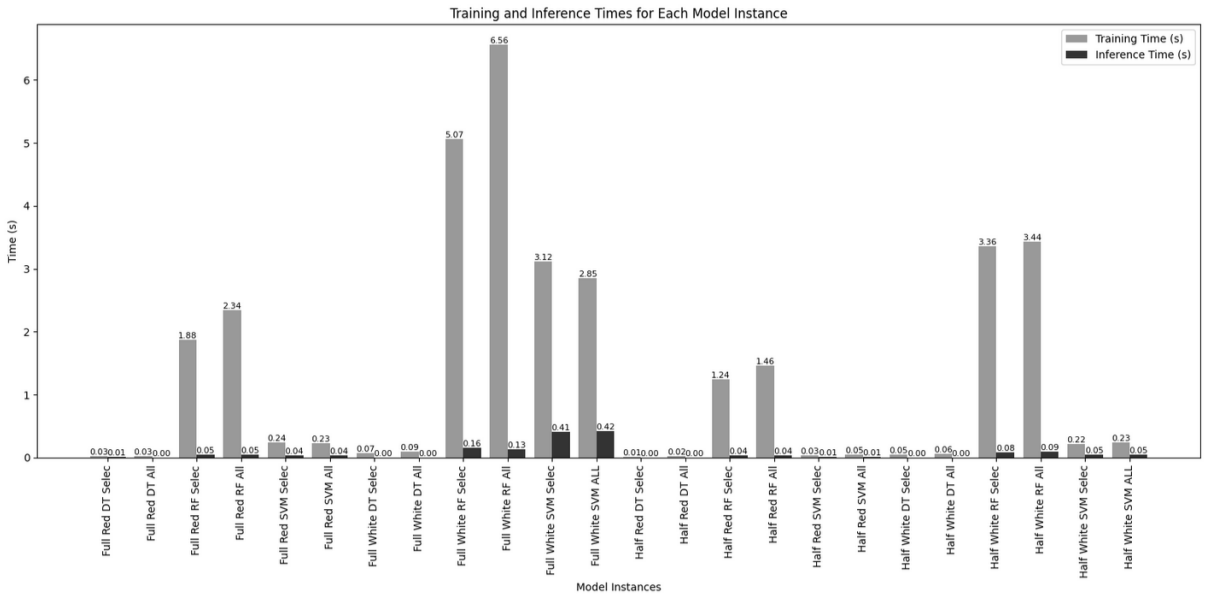

Figure 2: Training and Inference Times for all models, categorized by Dataset Size (Full or Half), Wine Type (Red or White), Algorithm (Decision Tree, Random Forest, or SVM), and Feature Set (Selected or All).

in some models suggests that they struggled to accurately classify underrepresented classes. This difficulty might come from factors such as the dataset's imbalance and the standardization process applied in SVM models.

Comparing results from the full dataset to those from the reduced (half) dataset, the F1-scores are consistently lower for the smaller dataset, confirming that a larger dataset generally improves classification performance. From all the models, Random Forest classifiers demonstrated the highest F1-scores for both red and white wine datasets, even outperforming SVMs in several cases.

When comparing the impact of feature selection, Decision Trees performed better with selected features in three out of four cases. In contrast, Random Forests and SVMs achieved superior results when utilizing all features. This suggests that Decision Trees benefit from a curated feature set, whereas Random Forests and SVMs take advantage of the additional information from a bigger feature set more effectively.

Comparing the algorithms overall, SVMs unexpectedly underperformed, achieving the lowest F1-scores. This could be attributed to the dataset's uneven class distribution, which negatively affected SVM performance. Random Forests emerged as the top-performing algorithm, highlighting their effectiveness in handling unbalanced datasets.

In terms of computational performance, illustrated in Figure 2, Random Forests required the most computation time, which is understandable given their ensemble nature and the use of 500 estimators. SVMs followed in computational time, while Decision Trees were the most efficient. A significant increase in computation time was observed when moving from the red wine subset to the larger white wine subset, emphasizing the effect of dataset size on processing time. This trend was further corroborated by the comparison between the full and half datasets.

To evaluate efficiency in terms of performance relative to computation time, Figure 3 was analyzed. Decision Trees were identified as the most efficient, offering the best balance of performance per computation time. On the other hand, Random Forests and SVMs were less efficient, with Random Forests providing the highest F1-scores but requiring considerably more computational resources.

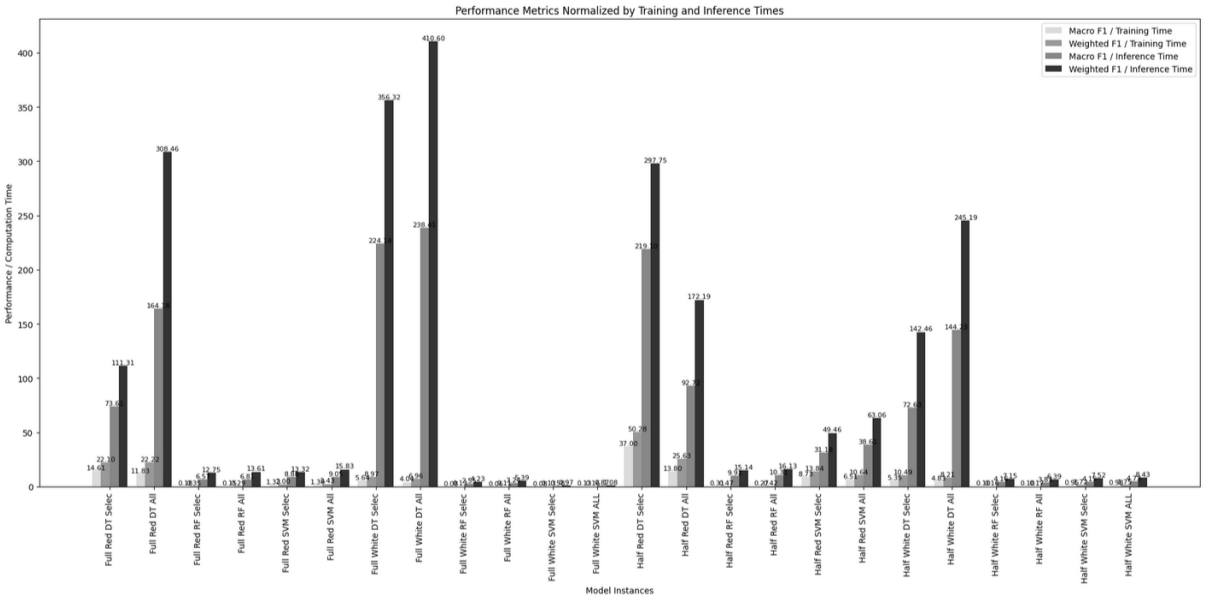

Figure 3: Performance vs. Computation Time, categorized by Dataset Size (Full or Half), Wine Type (Red or White), Algorithm (Decision Tree, Random Forest, or SVM), and Feature Set (Selected or All).

For scenarios where computational efficiency is critical, Decision Trees are the most suitable choice. However, when high performance is the primary objective, Random Forests are the preferred option. SVMs proved less practical due to their computational time results relative to their performance.

5 Conclusion

This study explored the trade-offs between computational efficiency and predictive performance in classifying wine quality using physicochemical attributes. Decision Trees, Random Forests, and Support Vector Machines (SVMs) were employed to evaluate the impact of dataset size, feature selection, and algorithm complexity on model outcomes.

The results highlighted that Random Forests offered the highest predictive performance, particularly for unbalanced datasets, highlighting their suitability for complex classification tasks. However, this performance came at the cost of significant computational time, especially when using larger datasets. Decision Trees, while less accurate than Random Forests, turned out to be the most computationally efficient option, making them a good choice for applications where efficiency is critical. Surprisingly, SVMs underperformed in this study, largely due to the dataset's imbalance and the algorithm's sensitivity to class distribution.

Feature selection proved to be a crucial factor in improving the computational and predictive efficiency of Decision Trees, while its impact was less pronounced for Random Forests and SVMs. Additionally, the analysis revealed a expected relationship between dataset size and model performance, concluding that larger datasets generally lead to more accurate classifications but at the expense of longer computation times.

In conclusion, the choice of algorithm for wine quality classification depends on the specific requirements of the task. For high accuracy, Random Forests are recommended, while Decision Trees are more suitable scenarios where resources are constrained.

This study highlights the importance of balancing computational and predictive considerations in the deployment of machine learning models for wine quality assessment and similar classification problems.

Future work could focus on several areas. First, hyper-parameter optimization could be explored, since the algorithms in this study were implemented with relatively standard configurations. Experimenting with alternative algorithms could also address their suitability for this classification problem. Additionally, the trained models could be validated on other datasets to assess their generalization ability. Employing cross-validation would further improve the results, addressing the dataset imbalance. Finally, a more in-depth exploratory data analysis could be conducted to uncover and better understand the relationships between the features, potentially leading to improved model performance.

References

[1] Cerdeira A. Almeida F. Matos T. Cortez, Paulo and J. Reis. Wine Quality. UCI Machine Learning Repository, 2009. DOI: https://doi.org/10.24432/C56S3T.

[2] Paulo Cortez, António Cerdeira, Fernando Almeida, Telmo Matos, and José Reis. Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47(4):547–553, 2009. Smart Business Networks: Concepts and Empirical Evidence.

[3] Nuriel Mor. Wine quality and type prediction from physicochemical properties using neural networks for machine learning: A free software for winemakers and customers, 02 2022.

[4] Qingwen Zeng. *Prediction of Wine Quality Using Ensemble Learning Approach of Machine Learning*, pages 770–774. 01 2023.