

Wine Quality Grading from Physicochemical Features: Comparing SVM and Random Forest with PCA/t-SNE Visualization

Fuquan Gong*

Jinan University-University of Birmingham Joint Institute, Jinan University, Xingye Avenue, Guangzhou, China

**Corresponding author: Fuquan Gong.*

Abstract

This study investigates multi-class prediction of wine quality scores (3–8) using physicochemical measurements from a Kaggle dataset. After removing the identifier field and standardizing features, we examine the data structure with correlation analysis and low-dimensional embeddings. PCA and t-SNE visualizations reveal substantial overlap among adjacent quality levels, implying gradual transitions rather than sharply separable classes. We then benchmark an RBF-kernel Support Vector Machine (SVM) and a Random Forest (RF) using an 80/20 stratified train/test split and 5-fold stratified cross-validation, tuning hyperparameters using Macro-F1 to account for class imbalance. Cross-validation slightly favors RF (Macro-F1 0.3271) over SVM (0.3078). On the held-out test set, RF achieves higher overall performance (Accuracy 0.7118, Macro-F1 0.3442, Weighted-F1 0.6897) than SVM (Accuracy 0.6332, Macro-F1 0.3072, Weighted-F1 0.6157). However, the relatively low Macro-F1 indicates uneven per-class performance, with limited effectiveness at rare quality levels, as reflected in confusion patterns that concentrate between neighboring grades (e.g., 5↔6 and 6↔7). RF feature importance highlights alcohol, sulphates, and volatile acidity as key predictors, offering interpretable evidence for the dominant decision cues.

Keywords

wine quality grading, SVM, random forest, PCA, t-SNE

1. Introduction

Recent years have seen growing use of machine learning for wine-quality assessment and broader oenology-related prediction tasks, motivated by the cost and subjectivity of sensory evaluation and the availability of routine physicochemical measurements. For example, Jain et al. [1] developed and compared multiple machine learning models on the classic red-wine physicochemical dataset and reported that ensemble tree methods, such as Random Forest and gradient-boosting variants, can achieve strong predictive performance, highlighting the practical value of data-driven quality forecasting in wine analytics. Beyond model comparisons, more recent work has proposed dedicated computational frameworks for wine-quality prediction using modern learners and emphasizes systematic evaluation protocols for multi-class quality labels. Related studies also apply machine learning to readily accessible wine analytical data from public sources and

stress that the relationship between physicochemical attributes and perceived quality is “complex and vague,” motivating robust modeling and careful interpretation. In addition, the current literature continues to benchmark classical classifiers and tree ensembles for wine-quality prediction and compare their behavior across different feature sets and settings. These findings collectively motivate our design choices: benchmarking a margin-based classifier against a tree-ensemble method, while using dimensionality-reduction visualizations to inspect class overlap and interpret error patterns in the ordered quality scores.

Motivated by these considerations, this study focuses on the wine-quality dataset from Kaggle. It formulates wine quality prediction (quality scores 3–8) as a multi-class classification task based on physicochemical features. Under a consistent train/test split, we benchmark two representative models: Support Vector Machine (SVM), which seeks maximum-margin decision boundaries in feature space, and Random Forest, which captures nonlinear feature interactions and offers feature-importance insights. To better understand the structure of the high-dimensional feature space and class separability, we further employ dimensionality reduction techniques—Principal Component Analysis (PCA) and t-SNE—to produce two-dimensional embeddings for visualization, thereby supporting the interpretation of performance differences and confusion patterns. Overall, our goal is to present a reproducible baseline pipeline that combines model comparison, metric-driven evaluation, and visualization-based explanation, and to inform future improvements such as imbalance-aware training and hyperparameter optimization.

2. Methods

2.1 Dataset Description

This study uses a wine quality dataset from Kaggle to investigate wine quality level prediction. The dataset contains 1,143 samples and 13 columns, including physicochemical measurements and the corresponding quality score labels.

2.2 Train/Test Split

Because the quality classes are imbalanced, we apply a stratified train/test split to preserve class proportions in both subsets. We use an 80/20 split: 80% for training and 20% for testing. The test set is used only for final reporting and is not involved in model selection or hyperparameter tuning.

For model selection and hyperparameter tuning, we perform 5-fold stratified cross-validation on the training set only, using Macro-F1 as the primary metric (with Weighted-F1 and Accuracy reported as secondary metrics). This protocol provides a fairer assessment across classes under imbalance and reduces variance caused by a single split.

2.3 Data Preprocessing

To predict wine quality levels from physicochemical measurements, we represent the dataset as a feature matrix X and a target vector y . Specifically, quality is treated as the target variable y , while the remaining physicochemical columns constitute the input features X . The Id column is an identifier and does not convey physicochemical information; therefore, it is removed before modeling to avoid adding non-informative noise.

Because physicochemical variables differ substantially in scale, feature scaling is required for scale-sensitive algorithms and dimensionality-reduction methods. We apply z-score standardization so that each feature has a mean of 0 and a variance of 1. To prevent data leakage, the scaler parameters are fitted on the training set only and then applied to both training and test sets; likewise, during cross-validation, preprocessing is fitted within each fold using only the corresponding training portion.

2.4 Dimensionality Reduction: PCA & t-SNE

To better understand the structure of the high-dimensional physicochemical feature space and the separability among wine quality levels, we apply PCA and t-SNE to produce two-dimensional visualizations after performing a stratified train/test split and z-score standardization. Standardization is essential because the physicochemical variables differ substantially in scale; without scaling, features with larger numeric ranges may dominate the embedding, reducing stability and interpretability.

PCA is a linear dimensionality reduction technique that projects data onto orthogonal principal components that capture the maximum variance. In this study, we first project standardized features onto the first two principal components (PCA-2D) and color points by quality to inspect the distribution and overlap of different quality levels, while also reporting cumulative explained variance to quantify how much information is retained by the leading components.

t-SNE (t-distributed Stochastic Neighbor Embedding) is a nonlinear embedding method that preserves local neighborhood relationships, making it suitable for visualizing cluster structure and class overlap at a local scale. We use t-SNE exclusively for visualization and error interpretation rather than as model input, since its embeddings are sensitive to hyperparameters and random initialization, and global distances in the 2D map are not strictly interpretable. For reproducibility, we run t-SNE on standardized features with `n_components=2`, `perplexity=30`, `learning_rate='auto'`, `init='pca'`, `n_iter=1200`, and `random_state=42`.

2.5 Classification Models: SVM & Random Forest

We formulate wine quality prediction as a multi-class supervised learning task, using the physicochemical feature matrix X to predict the label $y = \text{quality}$. After a stratified train/test split, we benchmark two representative classifiers: Support Vector Machine (SVM) and Random Forest. These models reflect two complementary modeling paradigms—maximum-margin discriminative learning and tree-ensemble nonparametric learning—allowing us to compare performance under different inductive biases.

For SVM, since the decision function is built on distances and margins, the model is sensitive to feature scaling. We therefore standardize the inputs via z-score normalization and train a multi-class SVM within a single preprocessing pipeline. In practice, we use a Pipeline that chains `StandardScaler` and SVM to ensure the scaler is fitted only to the training data, preventing data leakage. To account for potential nonlinear class boundaries among quality levels, we primarily use an RBF kernel and select hyperparameters using 5-fold stratified cross-validation (`StratifiedKFold` with `n_splits=5`, `shuffle=True`, `random_state=42`). The fixed search space is: penalty $C \in \{0.1, 1, 10, 100\}$ and kernel parameter $\gamma \in \{\text{scale}, 0.01, 0.1, 1\}$. Macro-F1 is used as the primary selection metric, with Weighted-F1 and Accuracy reported as secondary metrics to provide a fair assessment under class imbalance.

Random Forest captures nonlinear relationships and feature interactions by aggregating many decision trees and is typically robust to noise and outliers. Unlike SVM, Random Forest is not scale-sensitive, so it can be trained directly on the original features; for consistency, we still compare it using the same feature set and evaluation protocol. Hyperparameters are tuned using the same 5-fold stratified cross-validation, with the fixed search space $n_estimators \in \{200, 500\}$, $max_depth \in \{\text{None}, 10, 20\}$, $min_samples_leaf \in \{1, 2\}$, and $class_weight \in \{\text{None}, \text{"balanced"}\}$. Under our setting, the best Random Forest uses $n_estimators = 200$, $max_depth = 20$, $min_samples_leaf = 1$, $class_weight = \text{None}$, with $random_state = 42$ for reproducibility.

2.6 Evaluation Metrics

We treat quality as a multi-class label, so we evaluate model performance using a confusion matrix and a classification report. The confusion matrix directly shows how true classes map to predicted classes, which helps identify dominant error patterns—particularly misclassifications between adjacent quality levels (e.g., $5 \leftrightarrow 6$ and $6 \leftrightarrow 7$). The classification report provides class-wise precision, recall, and F1-score, making it easier to detect whether the model performs well only on frequent classes while underperforming on minority classes.

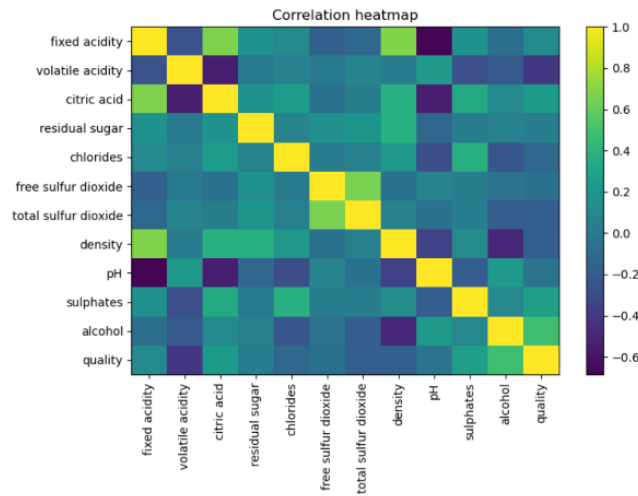
Because the class distribution is imbalanced, Macro-F1 is used as the primary metric for model comparison. Macro-F1 averages the F1-scores across classes with equal weight, providing a fairer reflection of performance across all quality levels. We also report Weighted-F1 and Accuracy as secondary metrics: Weighted-F1 accounts for class frequencies and reflects overall performance under the observed distribution, while Accuracy indicates the proportion of correct predictions but can be overly optimistic under imbalance and is therefore not used as the sole criterion.

3. Results

3.1 Exploratory Analysis & Dimensionality Reduction

From a correlation perspective, the heatmap (Figure R1) reveals clear dependencies among physicochemical variables. For example, free sulfur dioxide is strongly positively correlated with total sulfur dioxide; fixed acidity is positively correlated with citric acid; and density shows a clear negative association with alcohol. Such non-independence suggests noticeable collinearity in the dataset, motivating the use of PCA for linear dimensionality reduction. Moreover, the heatmap indicates directional relationships between quality and several key features: alcohol is positively associated with quality, while volatile acidity is negatively associated (as reflected by the corresponding color patterns in Figure R1), implying that these variables are likely informative for quality prediction.

Figure R1: Correlation heatmap



Regarding separability, the PCA 2D projection (Figure R2) shows substantial overlap among different quality levels in the first two principal components, suggesting that a simple linear 2D view cannot cleanly separate the classes. This aligns with the nature of the task—adjacent quality scores often differ only subtly—and suggests that most errors may occur between neighboring classes (e.g., 5↔6 and 6↔7), a claim confirmed later with confusion matrices. The cumulative explained variance curve (Figure R3) indicates that most variance can be captured with a limited number of components; around 8 principal components already retain nearly 95% of the variance, implying some redundancy in the original 11 features and supporting PCA-based compression from an information-retention standpoint (even though the 2D visualization still exhibits overlap).

Figure R2: PCA 2D scatter

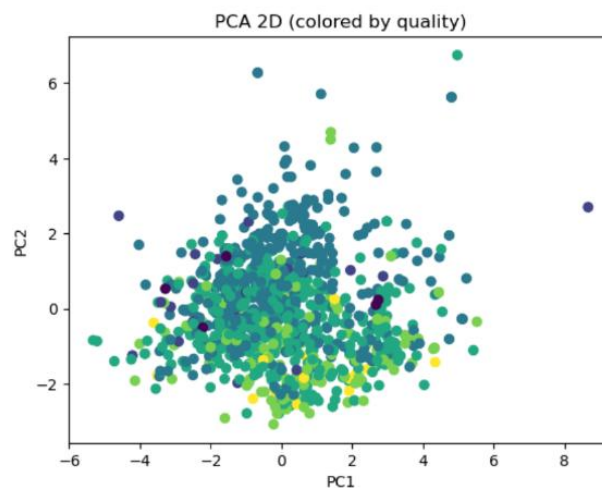
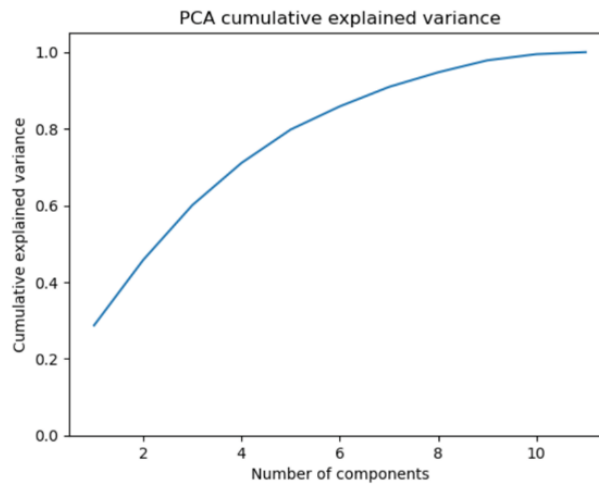
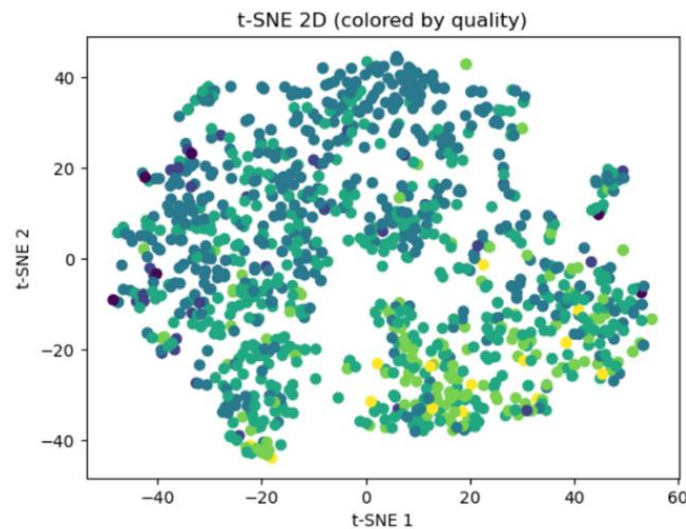


Figure R3: PCA cumulative explained variance

The t-SNE embedding (Figure R4) highlights local cluster structure more clearly than PCA, yet quality labels remain heavily mixed within clusters rather than forming clean, label-specific groups. This suggests that even with nonlinear embeddings, quality levels still overlap significantly in feature space. Consequently, classifiers must operate under ambiguous boundaries between adjacent classes, providing an intuitive explanation for the performance limitations that appear later, particularly for rare classes and neighboring-grade confusions.

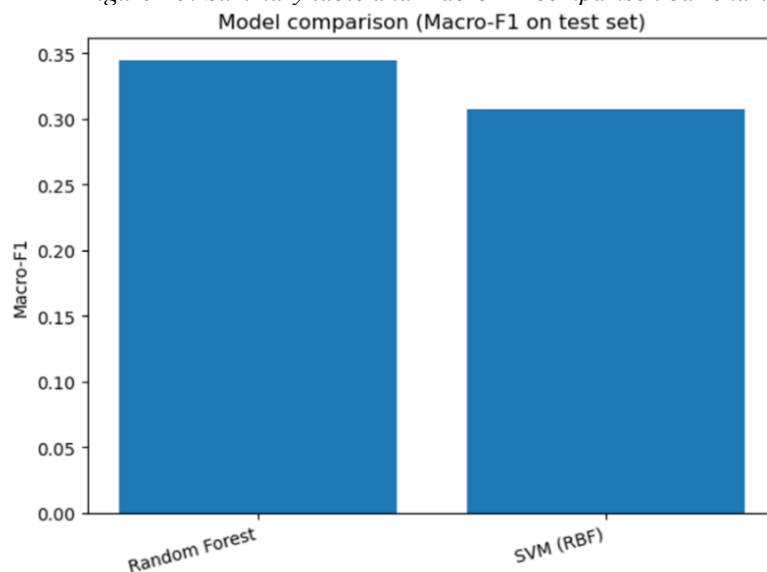
Figure R4: t-SNE 2D scatter

3.2 Model Selection & Predictive Performance

During model selection, we performed a grid search with 5-fold stratified cross-validation and used Macro-F1 as the optimization criterion. The best SVM (RBF) configuration is $C=10$, $\gamma=0.1$, with a CV Macro-F1 of 0.3078, while the best Random Forest uses $\text{max_depth}=20$, $\text{n_estimators}=200$, $\text{min_samples_leaf}=1$ ($\text{class_weight}=\text{None}$), and achieves a higher CV Macro-F1 of 0.3271. This indicates that Random Forest is slightly stronger than SVM in balanced, class-aware performance under the same CV protocol.

On the held-out test set, the overall metrics further confirm this trend. SVM achieves $\text{Accuracy}=0.6332$, $\text{Macro-F1}=0.3072$, $\text{Weighted-F1}=0.6157$, whereas Random Forest reaches $\text{Accuracy}=0.7118$, $\text{Macro-F1}=0.3442$, $\text{Weighted-F1}=0.6897$. Random Forest outperforms SVM across all three metrics, and the Macro-F1 comparison plot shows a clear advantage for Random Forest (Figure R5). Therefore, Random Forest is selected as the best-performing model under our experimental setting.

Figure R5: Summary table and Macro-F1 comparison bar chart



Class-wise reports show that both models perform relatively well on the dominant mid-range classes (5 and 6), but fail on the rare classes (3 and 8): precision/recall/F1 are all zero for classes 3 and 8 in both models. This is consistent with the extreme scarcity of these classes and highlights class imbalance as a primary limitation. Compared to SVM, Random Forest yields higher recalls for the main classes (e.g., class 5 recall improves from 0.722 to 0.845; class 6 recall from 0.674 to 0.728), contributing to its stronger overall performance.

Confusion matrices reveal that most errors occur between adjacent quality levels. For SVM, true class 5 is mostly predicted as 5 or 6 (70→5, 25→6), true class 6 is mostly predicted as 6 or 5 (62→6, 24→5), and true class 7 is often predicted as 6 (14→6) or 7 (13→7) (Figure R6). Random Forest shows the same neighbor-confusion structure but with stronger diagonal counts: class 5 correct predictions increase to 82 (vs. 70 in SVM), class 6 correct predictions increase to 67 (vs. 62), and class 7 slightly improves as well (Figure R7). This suggests that Random Forest better exploits nonlinear feature interactions, improving separation for major classes, while the inherent ambiguity between neighboring grades still drives most misclassifications.

Figure R6: Confusion matrix for the SVM (RBF) classifier

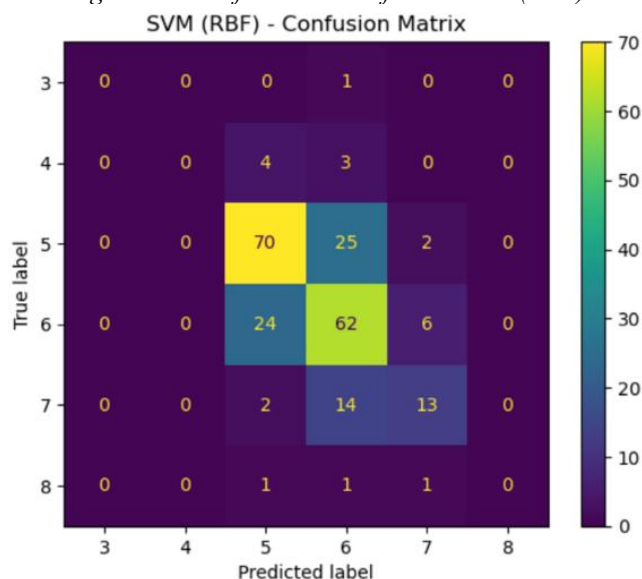
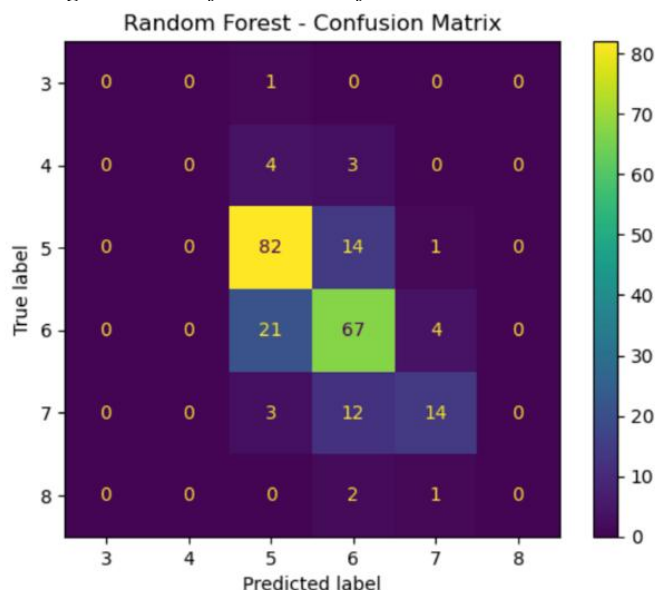


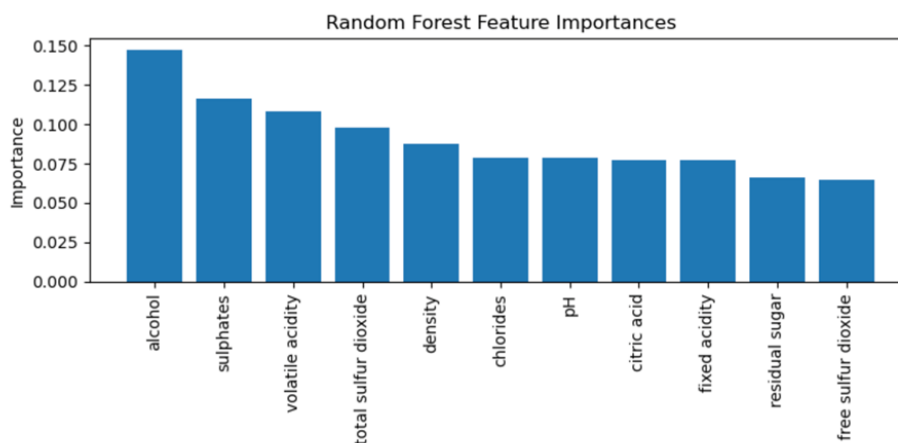
Figure R7: Confusion matrix for the Random Forest classifier



3.3 Model Interpretability and Error Pattern Interpretation

To better understand why the models achieve the observed performance and under what conditions predictions become less certain, we interpret the results by combining Random Forest feature importance, correlation structure, and confusion-matrix evidence. The feature-importance plot (Figure R8) shows that alcohol is the most influential predictor, followed by sulphates, volatile acidity, total sulfur dioxide, and density. This suggests that model decisions are primarily driven by physicochemical signals closely related to body/flavor properties, with alcohol and acidity-related cues playing a central role.

Figure R8: Random Forest feature-importance bar chart



4. Conclusion

The importance rankings are consistent with the correlation heatmap (Figure R1). The heatmap indicates that quality is positively associated with alcohol and negatively associated with volatile acidity, supporting the interpretation that these two variables provide strong predictive cues in this dataset. The heatmap also shows a clear correlation structure among certain feature pairs, such as the strong positive relationship between free sulfur dioxide and total sulfur dioxide. Under such feature dependence, predictive information can be distributed across correlated variables, which explains why sulfur-related variables collectively appear among the more important features.

From a behavioral perspective, the confusion matrices show that the dominant uncertainty arises between adjacent quality levels. The strongest confusion occurs between 5 and 6, and class 7 is also frequently confused with its neighbor class 6. This pattern aligns with the dimensionality-reduction visualizations: both PCA (Figure R2) and t-SNE (Figure R4) reveal substantial overlap between quality levels in feature space, especially among mid-range grades, indicating a more continuous transition rather than sharply separated clusters. Therefore, neighbor-level confusion is an expected consequence of the dataset geometry and the subtle differences between adjacent grades.

Finally, the classification reports and confusion matrices suggest that predictions are most stable for the dominant mid-range classes (5 and 6). In contrast, predictions for very rare grades (e.g., 3 and 8) tend to concentrate toward more frequent neighboring levels. This reflects the natural influence of class frequency on learning: when certain grades have limited examples, the model relies more heavily on patterns supported by larger sample sizes. This observation naturally motivates future enhancements, such as imbalance-aware training (class weights), resampling, or ordinal modeling to better reflect the ordered nature of quality scores.

References

- [1] Jain, K., Kaushik, K., Gupta, S. K., Mahajan, S., & Kadry, S. (2023). Machine learning-based predictive modelling for the enhancement of wine quality. *Scientific Reports*, 13, 17042.
- [2] Xiao, X.-Y., Yang, M., Huang, Y., Zhang, L., & Wang, Q. (2023). Research on Red Wine Quality Prediction Model Based on Deep Learning Architecture. In *Proceedings of the 2023 6th International Conference on Big Data Technologies (ICBDT '23)* (pp. 80–85). Association for Computing Machinery.
- [3] Li, X. (2024). Machine Learning-Based Wine Quality Predictive Modelling. In *Proceedings of the 1st International Conference on Innovations in Applied Mathematics, Physics and Astronomy (IAMPA 2024)* (pp. 86–91). SciTePress.
- [4] Chen, Z. (2025). Wine Quality Prediction with Ensemble Trees: A Unified, Leak-Free Comparative Study. In *Proceedings of the 2nd International Conference on Image Processing, Machine Learning, and Pattern Recognition (IPMLP '25)* (pp. 162–170). Association for Computing Machinery.

Funding

This research received no external funding.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgment

We would like to thank our course instructor and teaching assistants for their guidance and feedback throughout this project. We also acknowledge the contributors who made the wine quality dataset publicly available on Kaggle, which enabled the empirical study in this work.

Appendix

Data source: <https://www.kaggle.com/datasets/yasserh/wine-quality-dataset/data>

Table 1: SVM (RBF) Best Hyperparameters, Cross-Validation Score, and Test-Set Overall Metrics

Item	Value
Best SVM params	C=10, gamma=0.1
Best SVM CV Macro-F1	0.307810
Test Accuracy	0.633188
Test Macro-F1	0.307190
Test Weighted-F1	0.615661

Table 2: SVM (RBF) Test-Set Classification Report (Per-Class Metrics)

Class	Precision	Recall	F1-score	Support
3	0.000	0.000	0.000	1
4	0.000	0.000	0.000	7
5	0.693	0.722	0.707	97
6	0.585	0.674	0.626	92
7	0.591	0.448	0.510	29
8	0.000	0.000	0.000	3
Accuracy			0.633	229
Macro avg	0.311	0.307	0.307	229
Weighted avg	0.603	0.633	0.616	229

Table 3: Random Forest Best Hyperparameters, Cross-Validation Score, and Test-Set Overall Metrics

Item	Value
Best RF params	max_depth=20, n_estimators=200, min_samples_leaf=1, class_weight=None
Best RF CV Macro-F1	0.327137
Test Accuracy	0.711790
Test Macro-F1	0.344192
Test Weighted-F1	0.689679

Table 4: Random Forest Test-Set Classification Report (Per-Class Metrics)

Class	Precision	Recall	F1-score	Support
3	0.000	0.000	0.000	1
4	0.000	0.000	0.000	7
5	0.739	0.845	0.788	97
6	0.684	0.728	0.705	92
7	0.700	0.483	0.571	29
8	0.000	0.000	0.000	3
Accuracy			0.712	229
Macro avg	0.354	0.343	0.344	229
Weighted avg	0.676	0.712	0.690	229

Open Access

This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

