# Step C Machine Learning Project Report

**Abstract**

In this report, I encapsulate my journey through the Phase C machine learning project, focusing on the strategic development of a predictive model for wine quality assessment. I outline my technical choices, explain the rationale behind them, describe the analytical process of evaluating the model, and share the insights I gained from visual tools.

**Introduction**

My endeavor in Phase C was to construct a machine learning model to predict wine quality. This report documents my process, articulates my rationale for significant decisions, and analyzes the model's performance through quantitative measures and visualizations.

**Data Selection and Preparation**

* **Dataset**: I consistently utilized the wine quality dataset (ID 186).
* **Feature Engineering**: I crafted features to capture the interactions between alcohol, volatile acidity, and sulfur dioxide.
* **Preprocessing**: I employed standard scaling to normalize the feature values.

**Model Selection and Tuning**

The core of my model tuning process involved iterative experimentation with RandomForest and SVM algorithms, with a primary focus on the RandomForest classifier due to its effectiveness in handling feature interactions and providing interpretable outputs through feature importance scores.

For the RandomForest, I explored a range of values for several hyperparameters:

* **n\_estimators**: I began with a default of 10 trees and incrementally increased the number up to 200, noting substantial performance gains until the improvements plateaued.
* **max\_depth**: I tested depths ranging from shallow (5, 10) to allowing full growth (None), finding that 'None' significantly improved the model's capacity to discern data nuances.
* **min\_samples\_split**: I examined the impact of more restrictive splits (2, 4, 6, 10), discovering that a minimum of 2 samples allowed the model to learn finer patterns without overfitting.
* **min\_samples\_leaf**: I adjusted the parameter to find the best balance between bias and variance, ultimately finding that a single sample per leaf was optimal.

For the SVM, my focus was on the **C** parameter, balancing correct classification with the maximization of the decision function's margin. I experimented with values (0.1, 1, 10), and found that '1' provided the best compromise between margin maximization and error penalization.

In a detailed examination, I observed how alterations in these parameters influenced the ROC AUC score, precision, recall, and F1-score. For example, setting **min\_samples\_split** too high made the model overly conservative, missing essential patterns, as reflected by a reduced F1-score. Conversely, too low a value caused an overly complex model that did not generalize well, shown by a decrease in cross-validated ROC AUC scores.

Through diligent experimentation and careful comparison of cross-validation scores, I identified the optimal configuration for the RandomForestClassifier: **n\_estimators=200**, **max\_depth=None**, **min\_samples\_split=2**, and **min\_samples\_leaf=1**. This configuration produced a robust model with a high ROC AUC score, signaling strong predictive performance while maintaining generalizability.

The SVM, while initially promising, did not outperform the RandomForest model under the parameter settings I tested. It showed particular sensitivity to the **C** parameter, with lower values causing underfitting and higher values leading to overfitting, as noted in the ROC AUC scores from cross-validation.

**Handling Data Imbalance**

* **Techniques**: I implemented SMOTE and ADASYN, acknowledging their effectiveness in improving predictions for the minority class.

**Validation and Testing**

* **Cross-Validation**: I executed 10-fold Stratified K-Fold cross-validation to ensure the model's stability across different data subsets.

**Visualization and Interpretation**

Visual tools were pivotal in interpreting the model's performance, offering me an insightful lens through which to view the efficacy and precision of my predictions.

* **Feature Importance**: The first visualization provided a bar chart detailing feature importances, with 'free sulfur dioxide' standing out as the most influential predictor. The relative importance of features like 'volatile acidity' and 'alcohol' underscored the success of my feature engineering efforts to capture meaningful interactions affecting wine quality.
* **Predicted Probabilities**: The histogram of predicted probabilities displayed a skew towards higher probability scores. This suggested a strong confidence in the model’s predictions, predominantly favoring one class. It urged me to consider threshold adjustment to fine-tune the balance between sensitivity and specificity.
* **Precision-Recall Curve**: The precision-recall curve further elucidated the trade-off between precision and recall within the model. The curve started high, indicating a strong precision at the outset, which is essential when the cost of a false positive is high. Yet, the curve demonstrated a decline, a reminder of the intrinsic tension between capturing all relevant instances and maintaining precision.
* **Receiver Operating Characteristic (ROC)**: The ROC curve revealed the model's true positive rate against the false positive rate. The area under the curve (AUC) of 0.83 was an affirmation of the model's good discriminative ability, suggesting that the model has a high likelihood of correctly distinguishing between the classes.
* **Confusion Matrix**: The confusion matrix visual was a stark representation of the model's predictive power, with a notable concentration of true positives. However, the instances of false negatives and false positives highlighted areas where the model could potentially be improved, possibly through further data augmentation or feature engineering.

**Insights and Optimization**

* **Model Comparison**: I detailed the comparative performance of RandomForest and SVM, choosing the final model based on comprehensive metrics that went beyond the ROC AUC score.
* **Quantitative Impact**: I elaborated on how different steps, including feature engineering and class balancing, quantitatively influenced the model's performance.
* **Challenges**: I shared the challenges I faced during the model-building process and the strategies I employed to overcome them.
* **Future Directions**: I suggested potential avenues for future research to further improve the model's predictive capabilities.

**Conclusion**

This report details a data-driven approach to predicting wine quality. The project showcases my blend of rigorous analytical methods and strategic decision-making, culminating in a robust and interpretable machine learning model.

**Appendix**

* **Technical Details**: Included are code excerpts, output logs, and full model parameters for an in-depth technical review. The codebase is maintained on [GIT](https://github.com/neo050/Machine-learning-workshop.git).
* **Visualization Analysis**: I provide interpretations of feature importance plots, ROC curves, precision-recall curves, histograms of predicted probabilities, and heatmaps for confusion matrices. These visualizations appended at the end of the report for a comprehensive understanding of the model's performance.
* **Author:** Neoray Hagag ID 208090738  
    
    
  תמונה שמכילה טקסט, צילום מסך, תרשים, עיצוב

  התיאור נוצר באופן אוטומטי  
    
    
    
  תמונה שמכילה טקסט, קו, תרשים, עלילה

  התיאור נוצר באופן אוטומטי  
    
  תמונה שמכילה טקסט, תרשים, קו, עלילה

  התיאור נוצר באופן אוטומטי  
    
    
    
  תמונה שמכילה טקסט, תרשים, צילום מסך, עלילה

  התיאור נוצר באופן אוטומטי  
    
  תמונה שמכילה טקסט, צילום מסך, תרשים, עלילה

  התיאור נוצר באופן אוטומטי  
    
    
  