***Project Report on***

**Wine Quality Prediction Using Machine Learning**

Submitted in the partial fulfillment of the requirements for the award of Degree of B. Tech

***By***

**Yash Rastogi (2100680100355)**

Under the Supervision of:-

Mr. Md. Shahid

(Assistant Professor)



Department of Computer Science

Meerut Institute of Engineering and Technology,

Meerut – 250005

Dr. A.P.J. Abdul Kalam Technical University, U.P., Lucknow [Batch:2021-2025]

DECLARATION

I hereby declare that the work, which is being presented in the Project, entitled **“Wine Quality Prediction Using Machine Learning”** in partial fulfillment for the award of Degree of “Bachelor of Technology” in Department of Computer Science, and **submitted to the Department of Computer Science & Engineering**, Meerut Institute of *Engineering* and Technology, Meerut, affiliated to Dr. A.P.J Abdul Kalam Technical University, Uttar Pradesh, Lucknow is a record of my own investigations carried under the Guidance of **Mr.Md. Shahid, Assistant Professor**, Meerut Institute of Engineering and Technology. I have not submitted the matter presented in this Project anywhere for the award of any other Degree.

Name: **Yash Rastogi**

Roll No: **2100680100355**

Date: Signature:

# 

# 

# CERTIFICATE

I hereby certify that the work which is being presented in the project report entitled “Wine Quality Prediction Using Machine Learning” by “Yash Rastogi” in fulfillment of requirements for the award of the degree of B.Tech. (CSE) submitted in the Department of CSE at “Meerut Institute of Engineering and Technology” under A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY, LUCKNOW is an authentic record of my own work carried out under the supervision of Mr Md. Shahid.

.

Date: Signature of the Supervisor:

# ACKNOWLEDGEMENT

*It gives us a great sense of pleasure to present the report of the B. Tech Project undertaken during B.Tech. Final Year. We owe special debt of gratitude to our guide Mr.Md. Shahid, Department of Computer Science , Meerut Institute of Engineering and Technology, Meerut for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavors have seen light of the day.*

*We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.*

*Name: Yash Rastogi*

*Roll No: 2100680100355*

## *Date:*

*Signature:*

# ABSTRACT

*People generally want to live luxurious lives. They usually use the items for regular use or as a way to present their wealth. Wine consumption is very common these days. In order to protect human health, it became necessary to evaluate the quality of wine before consuming it. Therefore, this study is a step toward utilizing the wine's many properties for predicting its quality. The sources are used to obtain the dataset, and methods like Support Vector Machine, and Random Forest are used. The best of the three approaches is projected based on the training set results after a variety of measures are computed and the results are compared between the training and testing sets.*

*The quality of a wine is important for the consumers as well as the wine industry. The traditional (expert) way of measuring wine quality is time-consuming. Nowadays, machine learning models are important tools to replace human tasks. In this case, there are several features to predict the wine quality but the entire features will not be relevant for better prediction.*

*So, our thesis work is focusing on what wine features are important to get the promising result. For the purpose of classification model and evaluation of the relevant features, we used three algorithms namely support vector machine (SVM), naïve Bayes (NB), and artificial neural network (ANN). In this study, we used two wine quality datasets red wine and white wine. To evaluate the feature importance we used the Pearson coefficient correlation and performance measurement matrices such as accuracy, recall, precision, and f1 score for comparison of the machine learning algorithm. A grid search algorithm was* applied to improve the model accuracy. Finally, we achieved the artificial neural network (ANN) *algorithm has better prediction results than the Support Vector Machine (SVM) algorithm and the Naïve Bayes (NB) algorithm for both red wine and white wine datasets.*

# LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| S.NO | DESCRIPTION | PAGE NO. |
| Fig 1 | DISTRIBUTION OF RED AND WINE QUALITY | 16 |
| Fig 2 | EFFECT OF BALANCING DATA | 17 |
| Fig 3 | CORRELATION MATRICES OF RED WINE | 18 |
| Fig 4 | CORRELATION MATRICES OF WHITE WINE | 18 |
| Fig 5 | SUPPORT VECTOR MACHINE | 20 |
| Fig 6 | ARTIFICIAL NEURAL METWORK | 31 |
| Fig 7 | TO CHECK QUALITY WITH COUNT | 32 |
| Fig 8 | TO CHECK QUALITY WITH VOLATILE ACIDITY | 34 |
| Fig 9 | TO CHECK QUAKLITY WITH CITRIC ACID | 35 |
| Fig 10 | TO CHECK NEGATIVITY | 36 |
| Fig 11 | RED WINE PERFORMANCE ANALYSIS OF THE FEATURE MODEL | 37 |
| Fig 12 | WHITE WINE PERFORMANCE ANALYSIS OF THE FEATURE MODEL | 38 |
|  | TABLES |  |
| Fig 13 | TABLE 1 | 40 |
| Fig 14 | TABLE 2 | 40 |
| Fig 15 | TABLE 3 | 41 |
| Fig 16 | TABLE 4 | 41 |
|  | APPENDIX |  |

# TABLE OF CONTENTS

**Page No.**

**Declaration I**

**Certificate II**

**Acknowledgements III**

**Abstract IV**

**List of Figures V**

**List of Tables VI**

**CHAPTER 1: INTRODUCTION** 11

* 1. **INTRODUCTION** 1
  2. **SCOPE** 11
  3. **SOFTWARE DEVELOPMENT METHODOLOGY** 12

**CHAPTER 2: LITREATURE REVIEW 14**

**CHAPTER 3: METHODOLGY 16**

**CHAPTER 4: SOFTWARE REQUIREMENT SPECIFICATIONS 20**

* 1. **INTRODUCTION 20**
     1. **PURPOSE 21**
     2. **SCOPE 21**

v

* + 1. **DEFINITIONS 21**
  1. **SYSYTEM DESCRIPTION 21**
     1. **FEATURES 21**
  2. **SYSTEM INTERFACES 22**
     1. **USER INTERFACE 22**
     2. **HARDWARE INTERFACE 22**
     3. **SOFTWARE INTERFACE 22**
  3. **SYSTEM FUNCTIONALITY 22**

**4.4.1 INPUT 22**

**4.4.2 PREPROCESSING 22**

**4.4.3 MODEL TRAINING 22**

**4.4.4 PREDICTION 22**

**4.4.5 RESULT 22**

**4.5 CONSTRAINTS 23**

**4.6 EXTERNAL STIMULI 23**

**4.7 CONCLUSION 23**

**CHAPTER 5 SYSTEM DESIGN……………………………………….……………..…….24**

* 1. **INTRODUCTION 24**
  2. **SYASTEMRCHITECTURE 24**
     1. **HARDWARECOMPONENT…………………………………………….24**
     2. **SOFTWARECOMPONENT…………..………………………………….24**
     3. **SYSTEM WORKFLOW 25**

**CHAPTER -6 IMPLEMENTATION 27**

**CHAPTER 7 -TECHNOLOGY USED…………………………………………………29**

* + 1. **Support Vector Machine………………………………………...29**

**7.2 naïve Bayesian…………………………………………………………30**

**7.3 Artificial Neural Network……………………………………….30**

**CHAPTER 8 – EXPERIMENT AND RESULT ANALYSIS………………………………32**

**CONCLUSION AND FUTURE ENHANCEMENT………………………………………...43**

**REFERENCES………………………………………………………………………………...46**

**APPENDIX………………………………………………………………………………….….48**

**CHAPTER 1**

**INTRODUCTION**

## INTRODUCTION

The quality of the wine is a very important part for the consumers as well as the manufacturing industries. Industries are increasing their sales using product quality certification. Nowadays, all over the world wine is a regularly used beverage and the industries are using the certification of product quality to increases their value in the market. Previously, testing of product quality will be done at the end of the production, this is time taking process and it requires a lot of resources such as the need for various human experts for the assessment of product quality which makes this process very expensive. Every human has their own opinion about the test, so identifying the quality of the wine based on human experts it is a challenging task.

There are several features to predict the wine quality but the entire features will not be relevant for better prediction.

The research aims to what wine features are important to get the promising result by implementing the machine learning classification algorithms such as Support Vector Machine (SVM), Naïve Bayes (NB), and Artificial Neural Network (ANN), using the wine quality dataset.

The wine quality dataset is publically available on the UCI machine learning repository (Cortez et al., 2009). The dataset has two files red wine and white wine variants of the Portuguese “Vinho Verde” wine. It contains a large collection of datasets that have been used for the machine learning community. The red wine dataset contains 1599 instances and the white wine dataset contains 4898 instances. Both files contain 11 input features and 1 output feature. Input features are based on the physicochemical tests and output variable based on sensory data is scaled in 11 quality classes from 0 to 10 (0-very bad to 10-very good).

**1.2 SCOPE**

This involves gathering data on various attributes of wines such as chemical composition (e.g., acidity, pH, sugar content), sensory properties (e.g., aroma, flavor), and possibly external factors like weather conditions during grape growth. Data preprocessing may include handling missing values, normalizing or scaling features, and encoding categorical variables. This step involves selecting or creating relevant features that could influence wine quality prediction. It might include transforming existing features, extracting new features from the data, or even combining multiple features to create more informative ones.

Choosing an appropriate machine learning model for wine quality prediction depends on factors such as the nature of the data, the complexity of the problem, and computational resources available. Commonly used models for this task include decision trees, random forests, support vector machines, and neural networks. Once a model is selected, it needs to be trained on a labeled dataset containing examples of wines along with their corresponding quality ratings. During training, the model learns the underlying patterns and relationships between the input features and the target variable (wine quality).

* 1. **SOFTWARE DEVELOPMENT METHODOLOGY**

**Data:**

The dataset in this study was collected from Kaggle, these datasets can be viewed as classification or regression tasks. These datasets Contains only physicochemical (inputs) and sensory (the output) variables are available like fixed acidity, volatile acidity, citric acid residual sugar, chlorides, free sulphur dioxide, total sulphur dioxide, density, pH, sulphates, alcohol, quality.

**Preprocessing:**

One of the most important stages of machine learning (ML) is data preparation, which is preparing raw data for model training by cleaning and formatting it. It has a major impact on how well a machine learning model performs and how accurate it is. Data preparation involves a number of crucial procedures, including:

1. Handling Missing Data: Find and fix any missing information in the dataset. Delete, fill with default values, or use imputation methods such as mean, median, or mode are the available options.

2. Data Cleaning: Eliminate duplicates from the dataset and rectify data that is erroneous or inconsistent.

3. Data Exploration and Understanding: To understand the distribution, patterns, and linkages of the data, analyse and visualize it. This facilitates the making of wellinformed preprocessing decisions.

4. Feature Scaling: To bring numerical features to a consistent scale, standardize or normalize them. Standardization, or z-score normalization, or Min-Max scaling are common methods.

5. Feature Engineering: In order to give the model more information, develop new features. This may entail changing or combining current features.

6. Normalization: In particular, for algorithms that depend on distance metrics, normalize the data to provide constant feature sizes.

7. Data Transformation: To make the data better suited for modelling, apply transformations such as logarithmic or power transformations, especially when working with skewed distributions.

# CHAPTER 2 LITERATURE REVIEW

The authors proposed a data-driven method to evaluate and categorize various quality levels by using machine learning techniques to forecast the quality of red wine. Understanding the robustness and dependability of their predictions requires an understanding of the features taken into account, the dataset used in the study, and the particular of the machine learning algorithms used.

A focus on the junction of data mining, wine sensory reviews, and the application of a computational tool called the "Computational Wine Wheel" is suggested by the topic "Wine informatics: Applying Data Mining on Wine Sensory Reviews Processed by the Computational Wine Wheel." They Examine the particular data mining methods used in the Wine informatics environment. Clustering, classification, association rule mining, and sentiment analysis are a few examples of this. Examine the reasoning behind the use of these methods and how well they work to glean insightful information from sensory data related to wine.

The authors Dahal, K. R., J. N. Dahal, H. Ban jade, and S. Gaire research that ML is used to forecast the wine quality depending on a number of factors. In order to predict the quality of wine, they examine the efficacy of many machine learning models, including multi-layer artificial neural networks (ANN), Ridge Regression (RR), Support Vector Machine (SVM), and Gradient Boosting Regressor (GBR). The quality of the wine is determined by analysing several factors. Based on our analysis, GBR performs better than the other models, with MSE, R, and MAPPE values of 0.3741, 0.6057, and 0.0873, respectively.

The paper “Wine classification by taste sensors made from ultra-thin films and using neural networks” [describes a sensor array that can identify flavors and be used to categorize red wines. They demonstrate that discrete clusters may be distinguished in principal component analysis (PCA) plots for six different varieties of red wine by using impedance spectroscopy as the basis of detection. There are differences in red wine brands, vintages, and vineyards.

Furthermore, this "artificial tongue" can recognize wine samples kept in various settings if the data are processed using artificial neural networks (ANNs). The methods Standard Backpropagation and Backpropagation momentum in the ANNs could identify 900 wine samples with 100% accuracy. These samples were acquired with 30 measurements for each of the five bottles containing the six wines.

By merging a taste sensor array with lipid/polymer membranes and a smell sensor array with conducting polymer elements, the taste-smell sensory fusion was achieved. Responses to several wine brands were examined, and by analyzing the data from both types of sensors, it was possible to clearly distinguish between various samples. Studies were also conducted on the impact of wine aging on its quality. The technique was shown to be able to distinguish between samples of the same red wine that were aged differently. When both types of arrays are combined, the total amount of information available about the sample being measured is increased. The information offered by one type of array is independent of the other.

The main objective “A machine learning application in wine quality prediction”  of this project is to develop a machine learning model based on available experimental data gathered from various and diverse places around New Zealand, and to forecast wine quality by creating synthetic data. They used eighteen samples of Pinot noir wine. including 54 distinct attributes (7 chemical and 47 physiochemical properties). Using the SMOTE approach, they created 1381 samples from the original 12 samples; six samples were kept for model testing. We examined the results with four different feature selecting methods. To predict wine quality, significant characteristics also known as important variables that were demonstrated to be significant in at least three feature selection techniques were used. A holdout original sample was used to train and test seven machine learning algorithms.

The paper “A classification approach with different feature sets to predict the quality of different types of presents a novel approach that takes into account various feature selection algorithms, including Principal Component Analysis (PCA), Recursive Feature Elimination (RFE) approach, and nonlinear decision tree based classifiers for performance metrics analysis. Using the Random Forest classifier, we discovered accuracies ranging from 94.51% to 97.79% with various feature sets. The wine specialists will find this analysis useful in understanding the key elements to take into account when choosing a high-quality wine. In the paper “Assessing wine quality using a decision tree “ present a decision tree-based approach to evaluating wine quality, and they evaluate it against the wine-quality dataset from the UC

Irvine Machine Learning Repository. 60% of the results match those obtained using conventional assessment methods.

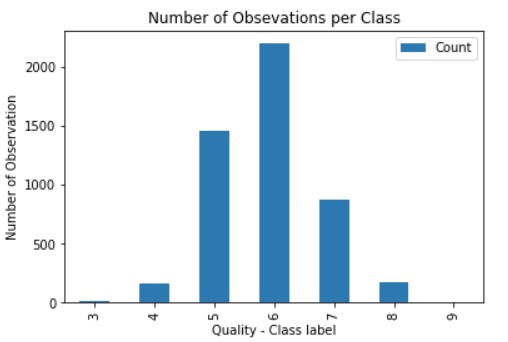
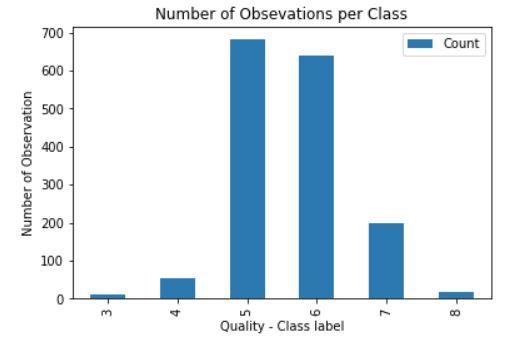
In the paper “Wine drinking is associated with increased heart rate variability in women with coronary heart disease”

# CHAPTER 3

# METHDOLOGY

## Unbalanced Data

Visualize the quality class label in the red wine and white wine dataset as follows:



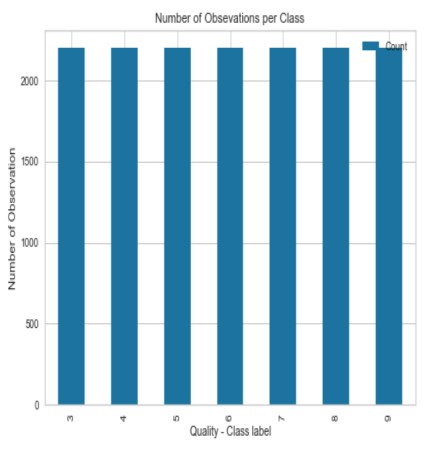
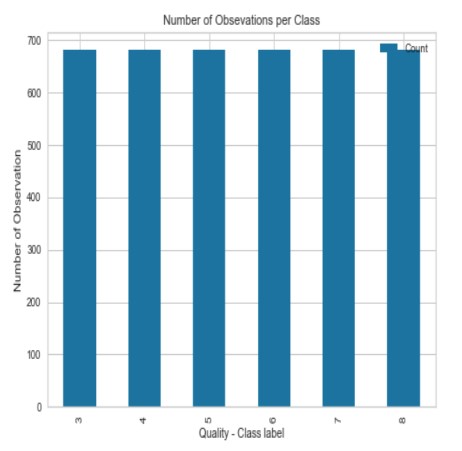
*Figure 1: Distribution of Red & White wine quality*

Figure 3 shows that the quality class of the red wine and white wine dataset shows that its distribution and we can see the most value is 5 in red wine and 6 in white wine, and all class values are in between 3 to 8 in red wine and 3 to 9 in white wine.

The datasets are the imbalanced distribution of red wine and white wine where the separate classes are not equally represented. This imbalanced data can lead to overfitting and underfitting algorithms. The red wine's highest quality class 5 instances are 681 and white wine highest quality class 6 instances are 2198. Both datasets are unbalanced with the number of instances ranging from 5 in the minority class up to 681 in red wine and ranging from 6 in the minority class up to 2198 in the majority class. The highest quality scores are rarely paralleled to the middle classes. By using resampling this problem can be solved, the resampling is by adding copies of examples from the under-represented class of unnaturally creating such instances (over-sampling) or either by removing from the overrepresented class (under-sampling). Mostly, it will be better to oversample unless you have sufficiently of data. However, there are some disadvantages to over-sampling it increases the instances of the dataset, so the processing time is increasing to build the model. Oversampling can lead to overfitting when putting the extremes

(Drummond and Holte, 2003). Therefore the resampling is preferred.

A good way to deal with the imbalanced datasets by applying the supervised synthetic minority oversampling technique (SMOTE) filter (Chawla, 2005). SMOTE is an over-sampling technique in which a lesser amount of classes in the training set is over-sampled and creating the new sample form to relieve the class imbalance. Therefore, to solve the data imbalanced problem we used the SMOTE technique.

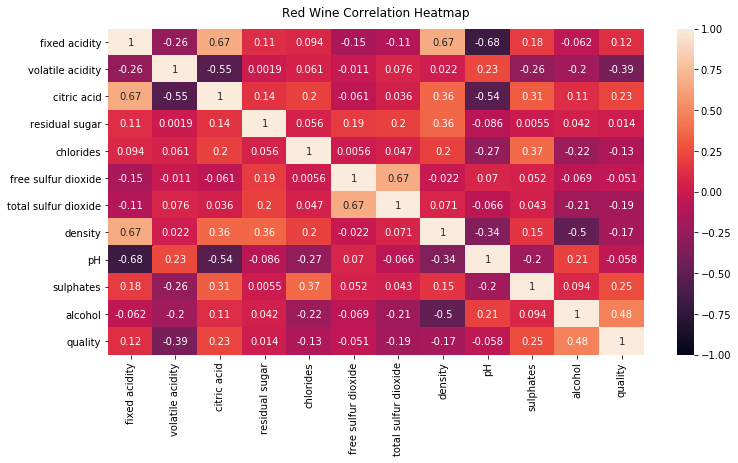


*Figure 2: Effect of balancing dataset*

After applying the SMOTE technique to balance the dataset as shown in Figure 4, the default and non-default amount of instances are the same, that is 681 instances in the red wine and 2198 instances in the white wine.

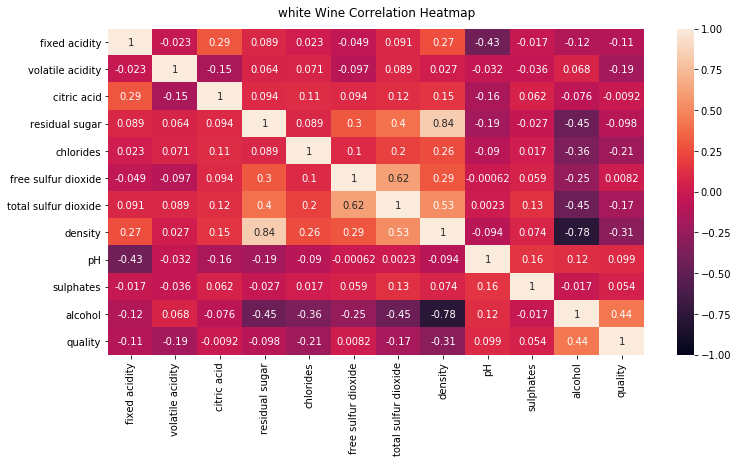
## Feature Selection

For a better understanding of the features and to examines the correlation between the features. We use the Pearson coefficient correlation matrices to calculate the correlation between the features.



*Figure 3: correlation matrices red wine*

From Figure 5 red wine correlation matrix we ranked the features according to the high correlation values to the quality class such as freatures are 'alcohol', 'volatile acidity', 'sulphates', 'citric acid', 'total sulfur dioxide', 'density', 'chlorides', 'fixed acidity', 'pH', 'free sulfur dioxide', 'residual sugar'.



*Figure 4: correlation matrices white wine*

Similarily, from Figure 6 white wine correlation matrix we ranked the features according to the high correlation values to the quality class such as freatures are 'alcohol', 'density', 'chlorides', 'volatile acidity', 'total sulfur dioxide', 'fixed acidity', 'pH', 'residual sugar', 'sulphates', 'citric acid', 'free sulfur dioxide’.

**CHAPTER 4**

**SOFTWARE REQUIREMENT AND SPECIFICATION**

**4.1. Introduction**

The Wine Quality Prediction System aims to leverage machine learning techniques to predict the quality of wine based on various input parameters. This Software Requirements Specification (SRS) document outlines the purpose, features, interfaces, functionalities, constraints, and reactions to external stimuli of the system.

**4.1.1 Purpose**

The purpose of this system is to provide users with accurate predictions of wine quality based on relevant features such as chemical composition, acidity, residual sugar, etc. These predictions will aid wine producers, connoisseurs, and enthusiasts in assessing and improving wine quality.

**4.1.2 Scope**

The system will accept input data containing various attributes of wine samples and provide predictions regarding the quality of the wine. It will not involve the actual production or distribution of wine but will serve as a tool for analysis and decision-making in the wine industry.

**4.1.3 Definitions, Acronyms, and Abbreviations**

SRS: Software Requirements Specification

ML: Machine Learning

* GUI: Graphical User Interface

**4.2. System Description**

The Wine Quality Prediction System will utilize machine learning algorithms to analyze input data and provide quality predictions. It will include the following features:

**4.2.1 Features**

Data Input: Accepts input data containing attributes of wine samples.

* Preprocessing: Cleans and preprocesses input data for analysis.

Model Training: Utilizes machine learning algorithms to train predictive models.

Prediction: Generates quality predictions for wine samples.

* Results Presentation: Presents prediction results to users through a graphical user interface (GUI).

**4.3. System Interfaces**

**4.3.1 User Interfaces**

The system will have a GUI for users to interact with. This interface will allow users to input data, initiate predictions, and view the results in a user-friendly manner.

**4.3.2 Hardware Interfaces**

The system will be compatible with standard hardware configurations, including computers and servers with sufficient processing power and memory for machine learning tasks.

**4.3.3 Software Interfaces**

The system will be developed using programming languages such as Python and libraries such as scikit-learn, TensorFlow, or PyTorch for machine learning implementations.

**4.4. System Functionality**

**4.4.1 Input**

Accept input data containing attributes of wine samples such as acidity, residual sugar, pH level, alcohol content, etc.

**4.4.2 Preprocessing**

Clean and preprocess input data to handle missing values, outliers, and inconsistencies.

**4.4.3 Model Training**

Utilize machine learning algorithms (e.g., Random Forest, Support Vector Machines) to train predictive models using the preprocessed data.

**4.4.4 Prediction**

Apply trained models to generate quality predictions for new wine samples.

**4.4.5 Results Presentation**

Present prediction results to users through a GUI, including predicted quality scores and confidence levels.

**4.5. Constraints**

* The accuracy of predictions may be limited by the quality and quantity of input data.
* Computational resources required for model training and prediction may be significant, especially for large datasets.

**4.6. External Stimuli**

* Changes in input data quality or distribution may impact the accuracy of predictions.
* Advances in machine learning techniques or algorithms may necessitate updates to the system.

**4.7. Conclusion**

The Wine Quality Prediction System described in this SRS document aims to provide users with accurate predictions of wine quality using machine learning techniques. By outlining the purpose, features, interfaces, functionalities, constraints, and reactions to external stimuli, this document serves as a comprehensive guide for system development and implementation.

**CHAPTER 5**

**SYSTEM DESIGN**

**5.1 INTRODUCTION**

Systems design is the process of defining elements of a system like modules, architecture, components and their interfaces and data for a system based on the specified requirements. It is the process of defining, developing and designing systems which satisfies the specific needs and requirements of a business or organization.

**5.2 SYSTEM ARCHITECTURE**

**5.2.1 HARDWARE COMPONENT**

**1 SENSOR**: Environmental sensors in the vineyard collect data on soil moisture, temperature, and humidity. Grape quality sensors measure pH levels of grapes.

**2 EDGE PROCESSING**: Edge devices process this data in real-time, performing preliminary analysis and sending relevant data to local or cloud servers via wireless communication.

**3 DATA STORAGE**: Data is stored securely in cloud storage for scalability and accessibility, with local backups maintained on NAS/SAN systems.

**4 COMPUTING HARDWARE FOR MACHINE LEARNING**: High-performance workstations or cloud servers equipped with GPUs/TPUs are used to train machine learning models on historical data. These models are then deployed to edge AI devices or local servers for real-time inference.

**5 USER INTERFACE DEVICE**: Winemakers use desktops, laptops, or mobile devices to access a user-friendly interface that displays data insights, predictions, and actionable recommendations generated by the machine learning models.

**5.2.2 SOFTWARE COMPONENT**

**DATA STORAGE:** Data is stored in a cloud data warehouse like Google Big Query.

**PYTORCH:** Another popular deep learning framework.

**MATPLOTLIB, SEABORN:** Libraries for data visualization in Python.

**TORCHSERVE:** For deploying PyTorch models.

**5.2.3 WORKFLOW**

**Data Collection:** Sensors in the vineyard and winery collect data on environmental conditions, grape quality, and fermentation parameters.

**Data Transmission:** Data is transmitted to a central database using IoT devices and gateways.

**Data Storage**: Data is stored in a cloud data warehouse like Google BigQuery.

**Data Preprocessing:** ETL processes clean and prepare the data using Pandas and custom scripts.

Feature Engineering: New features are created and selected using RFE and PCA.

**Model Development:** Models are developed and trained using Scikit-learn, TensorFlow, and hyperparameter tuning with Optuna.

**Model Deployment:** The trained model is containerized with Docker, deployed on Kubernetes, and served using TensorFlow Serving.

Real-Time Inference: The model provides real-time predictions on grape quality and fermentation outcomes.

**User Interaction:** Winemakers access predictions and recommendations through a web interface built with React.

**Monitoring:** System performance and model predictions are monitored using Prometheus and Grafana

**5.2.4 CHALLENGES**

**1.** **DATA COLLECTION AND QUALITY**

Variability in Data: Environmental conditions, grape varieties, and winemaking practices vary widely, leading to heterogeneous data that is difficult to standardize.

Data Completeness: Missing data points from sensor failures or inconsistent data collection practices can hamper the effectiveness of machine learning models.

Sensor Accuracy: The precision and calibration of sensors used to measure environmental and grape parameters can significantly affect data quality.

**2.** **FEATURE SELECTION AND ENGINEERING**

Complex Relationships: Wine quality is influenced by a multitude of factors, including weather conditions, soil composition, grape physiology, and fermentation processes, making it challenging to identify the most relevant features.

Non-linear Interactions: The relationships between features and wine quality are often non-linear and complex, requiring sophisticated feature engineering techniques.

**3. MODEL SELECTION AND TRAINING**

Model Complexity: Simple models may fail to capture the intricacies of the data, while complex models risk overfitting, especially with limited or noisy data.

Hyperparameter Optimization: Finding the optimal set of hyperparameters for machine learning models is computationally intensive and requires robust optimization techniques.

Computational Resources: Training advanced machine learning models, particularly deep learning models, requires significant computational power and resources.

**4. DATA INTEGRATION AND MANAGEMENT**

Scalability: Integrating and managing large volumes of data from various sources (vineyards, wineries, external databases) in real-time can be challenging.

Data Security and Privacy: Ensuring the security and privacy of sensitive data, such as proprietary vineyard practices or specific blend compositions, is critical.

**5. MODEL DEPLOYMENT AND MAINTENANCE**

Real-Time Processing: Implementing real-time data processing and prediction systems that are both reliable and scalable.

Model Drift: Over time, changes in vineyard conditions, grape varieties, and winemaking practices can lead to model drift, reducing the accuracy of predictions.

Continuous Improvement: Establishing an effective feedback loop for continuous model retraining and improvement based on new data and actual outcomes.

**6. INTERPRETABILITY AND TRUST**

Model Interpretability: Ensuring that machine learning models are interpretable and that the insights provided are understandable to winemakers, who may not have a technical background.

Trust in Predictions: Building trust in machine learning predictions among winemakers, who may be skeptical of relying on automated systems for decisions traditionally made by human expertise.

**7. ENVIRONMENTAL AND BIOLOGICAL FACTORS**

Climate Change: Ongoing changes in climate patterns introduce variability and uncertainty, making it harder to predict wine quality based on historical data.

Biological Variability: Natural variations in grape physiology and microbial populations during fermentation add layers of complexity to the prediction models.

**8. ECONOMIC AND PRACTICAL CONSTRAINTS**

Cost of Implementation: The initial cost of setting up sensor networks, data infrastructure, and computational resources can be prohibitive, especially for smaller vineyards and wineries.

Skill Gap: The lack of expertise in both machine learning and viticulture/winemaking within the same team can be a barrier to effective implementation and use of predictive models.

# CHAPTER 6

# IMPLEMENTATION

The performance measurement is calculated and evaluate the techniques to detect the effectiveness and efficiency of the model. There are four ways to check the predictions are correct or incorrect:

 True Positive: Number of samples that are predicted to be positive which are truly positive.

 False Positive: Number of samples that are predicted to be positive which are truly negative.

 False Negative: Number of samples that are predicted to be negative which are truly positive.

 True Negative: Number of samples that are predicted to be negative which are truly negative.

Below listed techniques, we use for the evaluation of the model.

1. Accuracy – Accuracy is defined as the ratio of correctly

predicted observation to the total observation. The accuracy

can be calculated easily by dividing the number of correct

predictions by the total number of predictions.

Accuracy = True Positive + True Negative

True Positive + False Positive + False Negative + True Negative

2. Precision – Precision is defined as the ratio of correctly predicted positive observations to the total predicted positive

observations.

Precision = True Positive

True Positive + False Positive

3. Recall – Recall is defined as the ratio of correctly predicted positive observations to all observations in the actual class. The recall is also known as the True Positive rate calculated as,

Recall = True Positive

True Positive + False Negative

4. F1 Score – F1 score is the weighted average of precision and recall. The f1 score is used to measure the test accuracy of the model. F1 score is calculated by multiplying the recall and precision is divided by the recall and precision, and the result is calculated by multiplying two.

F1 score = 2∗ Recall∗Precision

Recall + Precision

Accuracy is the most widely used evaluation metric for most traditional applications. But the accuracy rate is not suitable for evaluating imbalanced data sets, because many experts have observed that for extremely skewed class distributions, the recall rate for minority classes is typically 0, which means that no classification rules are generated for the minority class. Using the terminology in information retrieval, the precision and recall of the minority

categories are much lower than the majority class. Accuracy gives more weight to the majority class than to the minority class, this makes it challenging for the classifier to implement well in the minority class. For this purpose, additional metrics are coming into widespread usage (Guo et al., 2008).

The F1 score is the popular evaluation matric for the imbalanced class problem (Estabrooks and Japkowicz, 2001). F1 score combines two matrices: precision and recall. Precision state how accurate the model was predicting a certain class and recall state that the opposite of the regrate misplaced instances which are misclassified. Since the multiple classes have multiple F1 scores. By using the unweighted mean of the F1 scores for our final scoring. We want our models to get optimized to classify instances that belong to the minority side, such as wine quality of 3, 8, or 9 equally well with the rest of the qualities that are represented in a larger number.

# CHAPTER 7

# TECHNOLOGY USED

A wide range of machine learning algorithms is available for the learning process. This section describes the classification algorithms used in wine quality prediction and related work.

7.1 Support Vector Machine

The support vector machine (SVM) is the most popular and most widely used machine learning algorithm. It is a supervised learning model that can perform classification and regression tasks. However, it is primarily used for classification problems in machine learning (Gandhi, 2018). The SVM algorithm aims to create the best line or decision boundary that can separate n-dimensional space into classes so we can put the new data points easily in the correct groups. This best decision boundary is called a hyperplane.

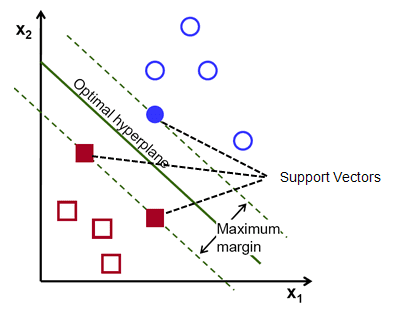


Fig 5 support vector machine

The support vector machine selects the extreme data points that helping to create the hyperplane. In Figure 1, two different groups are classified by using the decision boundary or hyperplane: The SVM model is used for both non-linear and linear data. It uses a nonlinear mapping to convert the main preparing information into a higher measurement. The model searches for the linear optimum splitting hyperplane in this new measurement. A hyperplane can split the data into two classes with an appropriate nonlinear mapping to suitably high measurements and for the finding, this hyperplane SVM uses the support vectors and edges (J. Han et al., 2012). The SVM model is a representation of the models as a point in space, the different classes are isolated by the gap to mapped with the aim that instances are wide as would be careful. The model can perform out a nonlinear form of classification (Kumar et al., 2020).

7.2 Naive Bayesian

The naive Bayesian is the simple supervised machine learning classification algorithm based on the Bayes theorem. The algorithm assumes that the feature conditions are independent of the given class (Rish, 2001). The naive Bayes algorithm helps to build fast machine learning models that can make a fast prediction. The algorithm finds whether a particular portion has a spot by a particular class it utilizes the probability of likelihood (Kumar et al., 2020).

7.3 Artificial Neural Network

The artificial neural network is a collection of neurons that can process information. It has been successfully applied to the classification task in several industries, including the commercial, industrial, and scientific filed (Zhang, 2000). The algorithm model is a connection between the neurons that are interconnected with the input layer, a hidden layer, and an output layer (Hewahi, 2017). 5 The neural network is constant because while an element of the neural network is failing, it can continue its parallel nature without any difficulties (Mhatre et al., 2015).

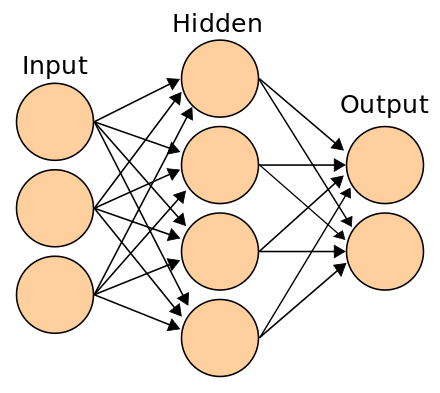


Fig 6 Ann

The implementation of the artificial neural network consists of three layers: input, hidden, and, output as shown in Figure 2. The function at the input layer is mapped the input attribute which passes input to

the hidden layer. The hidden layer is a middle layer where all input with the weights is received to each node in the hidden layer. The output layer is mapped to the predicted elements (says, 2020). The connection among the neurons is called weights, it has numerical values and this weight among the neurons are determining the learning ability of the neural network. The activation function is used to

standardize the output from the neurons and these activation functions are evaluate the output of the neural network in the mathematical equations. Each neuron has an activation function. The neural network is hard to understand without mathematical reasoning. Activation functions are also called the transmission function and also helps to standardize the output range between -1 to 1 or 0 to 1.

# CHAPTER 8

# EXPERIMENT AND RESULT ANALYSIS

## 

The various quality levels or categories are represented by the X-axis (Quality). It could take the form of continuous values (like a rating scale from 0 to 8) or discrete values (like low, medium, and high). The frequency or count of observations falling into each quality category is shown on the Y-axis (Count). It shows the number of instances or data points associated with each quality level. A positive connection on the graph indicates that more observations at greater counts or frequencies are related to higher quality levels. This would suggest that most observations have ratings of high quality. A negative correlation on the graph indicates that larger counts are related to poorer quality levels. This could suggest that the majority of observations have ratings that are too low. The lack of a discernible pattern or trend on the graph indicates that the number of observations does not change consistently across quality levels. Distribution A particular dataset's quality distribution or the overall quality of the data can be inferred from the distribution of counts across quality levels. A Synopsis of Statistics Further details regarding the count distribution's central tendency can be obtained by looking at the mean, median, and mode counts.

## 

## Fig 7 to check quality with count

## EXAMPLE:

Let's say you're analysing product quality data for a manufacturing company. The graph might show how many products fall into different quality categories (e.g., excellent, good, average, poor) based on quality assessments conducted during quality control processes. The graph could reveal that most products are rated as "good," with fewer products rated as "excellent" or "poor." This information could help the company identify areas for improvement in their manufacturing processes or product design.

The amounts of volatile acidity in the wine samples are represented by the Y-axis (Volatile Acidity). It could be expressed in terms of grams per litre, or g/L. Greater values denote higher acidity, and lower values denote lower acidity. The quality ratings given to the wine samples are displayed on the X-axis (Quality). It could be an ordinal category (bad, fair, good, exceptional), or a discrete scale (from 0 to 8). Greater perceived quality is indicated by higher values. Should the graph have a negative connection, it implies that wine quality tends to decline as volatile acidity rises. This is due to the fact that higher volatile acidity levels can lead to off-odours and flavours, which detract from the overall impression of quality. In the range of acidity levels seen in the dataset, volatile acidity may not have a substantial effect on wine quality if the graph displays no discernible pattern or trend. Given that lower quality wine is usually correlated with more volatile acidity, a positive connection on the graph would be unexpected in this particular case. Distribution: The way in which quality ratings are distributed among various volatile acidity levels might shed light on the interactions and interdependencies between these factors. Condensed Statistics: Further information on the central tendency of quality within particular acidity ranges can be obtained by examining the mean, median, and mode quality ratings at various volatile acidity levels.

## 

## Fig 8 to check quality with volatile acidity

## EXAMPLE:

In a graph depicting the relationship between quality and volatile acidity in wines, you might observe that wines with lower volatile acidity tend to receive higher quality ratings, indicating that acidity levels play a crucial role in determining perceived wine quality. This insight can be valuable for wine producers in managing the acidity levels during winemaking to produce wines of superior quality.

The citric acid content or level in the wine samples is shown by the Y-axis (Citric Acid). It could be expressed in terms of grams per litre, or g/L. Greater values correspond to higher citric acid concentrations. The quality ratings given to the wine samples are displayed on the X-axis (Quality). Either a discrete scale, like 1 to 10, or ordinal categories, like poor, fair, good, and exceptional, can be used to rate quality. Greater perceived quality is indicated by higher values. If the graph exhibits a positive association, it means that wine quality tends to get better as citric acid content rises. This could suggest that the wine's taste profile and overall sensory qualities are positively influenced by modest quantities of citric acid. In this case, it would be surprising if the graph displays a negative connection because higher citric acid levels are typically linked to increased

acidity and fresher flavor, both of which can improve wine quality. If there is no discernible pattern or trend in the graph, it is possible that, within the range of concentrations found in the dataset, the amount of citric acid may not significantly affect the quality of wine. Understanding how differences in citric acid levels affect perceived wine quality can be gained from examining the distribution of quality ratings across different amounts of citric acid. Condensed Statistics: Further information on the central tendency of quality within particular concentration ranges can be obtained by examining the mean, median, and mode quality ratings at various levels of citric acid concentration.

## 

## Fig 9 to check quality with citric acid

EXAMPLE: that wines with moderate levels of citric acid tend to obtain higher quality ratings in a graph showing the relationship between quality and citric acid in wines. This suggests that citric acid positively influences the impression of wine quality overall. This knowledge may help winemakers control the amount of citric acid added to their wines throughout the fermentation process, resulting in more harmonious and consumer-pleasing wines.

Correlation

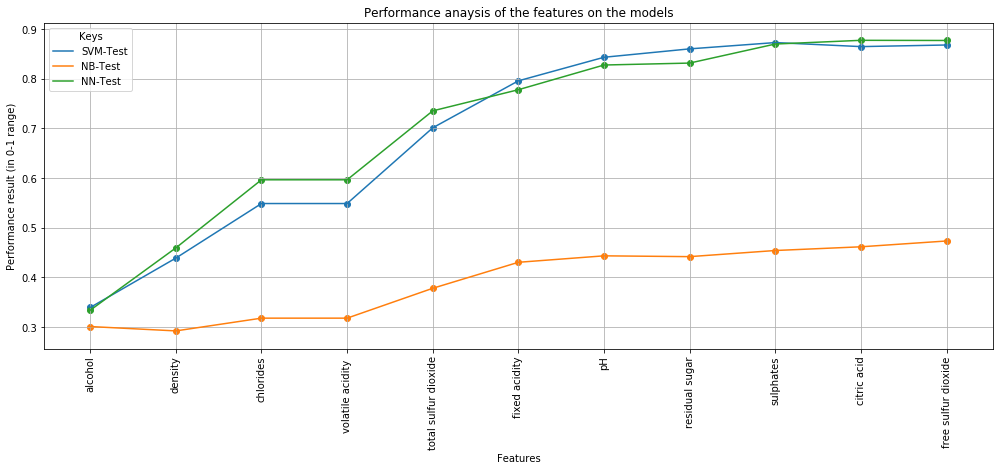
1. Positive Correlation
2. Negative Correlation

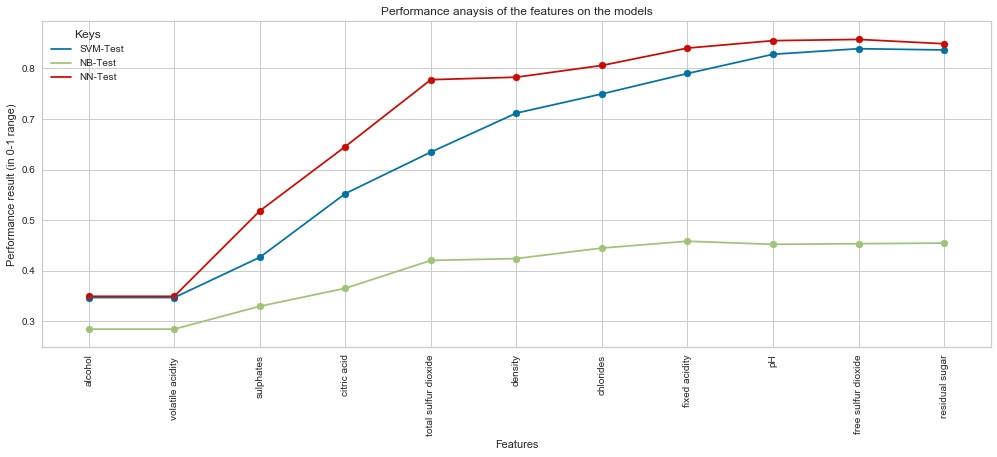
## 

## Fig 10 to check negativity

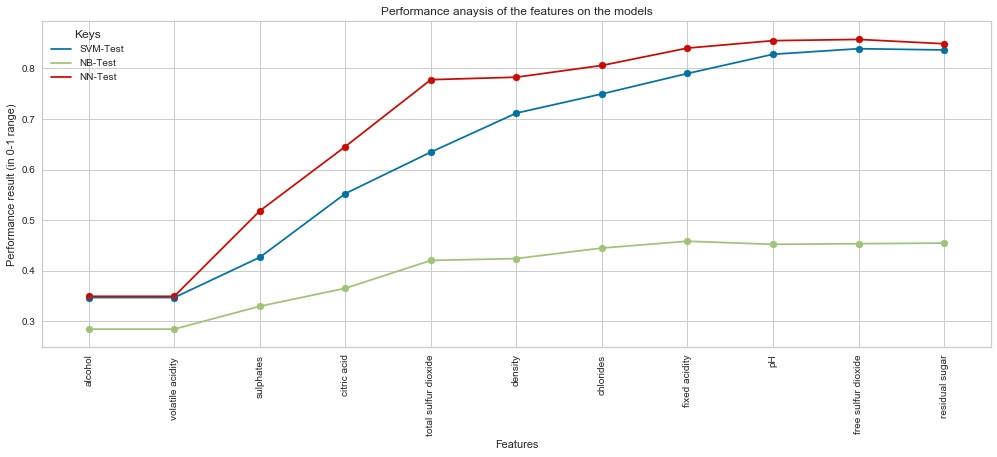
## Feature Selection result

To evaluate the performance of each feature, a Pearson correlation coefficient technique was implemented and obtained results. Above red wine Figure 5, and white wine Figure 6 shows the importance of each feature and according to the high relationship with the quality, the features were ranked.





*Figure 11: Red wine performance analysis of the feature model*



*Figure 12: white wine performance analysis of the feature model*

Therefore, the analysis of groups of features from left to right is implemented, shown in Figure 7 and Figure 8 from both datasets first 10 features are selected and the last feature is excluded because there is no improvement and it is decreasing the performance of the model. 'Residual sugar' feature from red wine datasets and 'free sulfur dioxide' feature from the white wine dataset is excluded for the final implementation of the models. The above red wine performance analysis Figure 7 and white wine performance analysis Figure 8 show a clue that the prediction models achieved better results with their selected 10 features.

## Model Results

The importance of the features are identified and from both dataset's first 10 features were selected and the last feature was excluded, above red wine performance analysis Figure 7 and white wine performance analysis Figure 8 shows that the performance in terms of accuracy.

Firstly, these selected features were implemented on the unbalanced classes, Figure 3 shows the unbalanced classes and the performance of the prediction model, in terms of accuracy, precision, recall, and F1 score is examined, as expressed in Table 4 red wine and Table 5 white wine.

*Table 1: Red wine Unbalanced class performance.*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **SVM** |  | **NB** | |  | **ANN** | |  |
| **Class** | Precision | Recall | F1    score | Precision | Recall | F1    score | Precision | Recall | F1    score |
| **3** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| **4** | 0.00 | 0.00 | 0.00 | 0.17 | 0.50 | 0.26 | 0.00 | 0.00 | 0.00 |
| **5** | 0.79 | 0.79 | 0.79 | 0.73 | 0.60 | 0.66 | 0.70 | 0.82 | 0.76 |
| **6** | 0.60 | 0.60 | 0.60 | 0.54 | 0.53 | 0.54 | 0.57 | 0.62 | 0.59 |
| **7** | 0.62 | 0.62 | 0.62 | 0.32 | 0.43 | 0.37 | 0.62 | 0.23 | 0.33 |
| **8** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| **Accuracy** |  | 69.06 |  | 54.06 | |  | 64.37 | |  |
|  |  |  |  |  | |  |  | |  |

*Table 2: White wine unbalanced class performance.*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **SVM** | |  | **NB** |  |  | **ANN** | |
| **Class** | Pr  ecision | Recall | F1    score | Precision | Recall | F1    score | Precision | Recall | F1    score |
| **3** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| **4** | 0.53 | 0.21 | 0.30 | 0.31 | 0.28 | 0.30 | 0.36 | 0.10 | 0.16 |
| **5** | 0.72 | 0.65 | 0.68 | 0.53 | 0.58 | 0.55 | 0.56 | 0.54 | 0.55 |
| **6** | 0.66 | 0.81 | 0.72 | 0.54 | 0.36 | 0.43 | 0.55 | 0.66 | 0.60 |
| **7** | 0.68 | 0.54 | 0.60 | 0.33 | 0.66 | 0.44 | 0.39 | 0.34 | 0.36 |
| **8** | 0.86 | 0.40 | 0.54 | 0.17 | 0.02 | 0.04 | 0.00 | 0.00 | 0.00 |
| **9** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| **Accuracy** |  | 67.83 | |  | 45.55 |  |  | 52.65 | |

Then these selected features were implemented on the balanced class, Figure 4 shows that the balancing of each class and the performance of the prediction model, in terms of accuracy, precision, recall, and f1 score is examined, as expressed in Table 6 red wine and Table 7 white wine.

*Table 3: Red wine balanced class performance.*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **SVM** |  | **NB** | |  | **ANN** | |  |
| **Class** | Precision | Recall | F1    score | Precision | Recall | F1    score | Precision | Recall | F1    score |
| **3** | 1.00 | 1.00 | 1.00 | 0.53 | 0.80 | 0.63 | 0.99 | 1.00 | 1.00 |
| **4** | 0.91 | 0.94 | 0.92 | 0.43 | 0.31 | 0.36 | 0.91 | 0.98 | 0.94 |
| **5** | 0.79 | 0.66 | 0.70 | 0.54 | 0.40 | 0.46 | 0.82 | 0.65 | 0.72 |
| **6** | 0.60 | 0.60 | 0.60 | 0.29 | 0.21 | 0.24 | 0.64 | 0.57 | 0.60 |
| **7** | 0.82 | 0.87 | 0.84 | 0.48 | 0.41 | 0.44 | 0.81 | 0.96 | 0.88 |
| **8** | 0.91 | 1.00 | 0.95 | 0.53 | 0.84 | 0.65 | 0.94 | 1.00 | 0.97 |
| **Accuracy** |  | 83.52 |  | 46.33 | |  | 85.16 | |  |

*Table 4: White wine balanced class performance.*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **SVM** | |  | **NB** |  |  | **ANN** | |
| **Class** | Precision | Recall | F1    score | Precision | Recall | F1    score | Precision | Recall | F1    score |
| **3** | 1.00 | 0.99 | 0.99 | 0.71 | 0.42 | 0.53 | 1.00 | 1.00 | 1.00 |
| **4** | 0.99 | 0.86 | 0.92 | 0.54 | 0.56 | 0.55 | 0.93 | 0.98 | 0.95 |
| **5** | 0.96 | 0.58 | 0.73 | 0.36 | 0.48 | 0.41 | 0.81 | 0.78 | 0.79 |
| **6** | 0.50 | 0.96 | 0.66 | 0.37 | 0.11 | 0.17 | 0.70 | 0.63 | 0.66 |
| **7** | 0.98 | 0.72 | 0.83 | 0.30 | 0.48 | 0.37 | 0.81 | 0.84 | 0.83 |
| **8** | 0.99 | 0.92 | 0.95 | 0.37 | 0.35 | 0.36 | 0.93 | 0.97 | 0.95 |
| **9** | 1.00 | 1.00 | 1.00 | 0.79 | 0.91 | 0.85 | 1.00 | 1.00 | 1.00 |
| **Accuracy** |  | 86.86 | |  | 46.68 |  |  | 88.28 | |

Based on our research question “what wine features are important to get the promising result?” To answer this question we implement the Pearson coefficient correlation matrices and calculate the relationship among all the features, as in red wine correlation matrices Figure 5 and white wine correlation matrices Figure 6. Then we ranked the features based on high correlation with the quality feature. The analysis of groups of features from left to right is implemented, shown in Figure 7 and Figure 8, and from both datasets first 10 features are selected and the last feature is excluded because there is no improvement and it is decreasing the performance of the model. 'residual sugar' feature from red wine datasets and 'free sulfur dioxide' feature from the white wine dataset is excluded for the final implementation of the models.

After identifying the importance of the features we start the implementation of the model. To analyze the performance of the model firstly, we implemented the model on the original data (unbalanced class), as shown in Figure 3, and then implemented the model on the balance class, shown in Figure 4, balancing each class.

In terms of the performance of the prediction model accuracy, precision, recall, and f1 score is examined, as expressed in Table 4 red wine and Table 5 white wine performance analysis results for unbalanced classes for each model is examined, and Table 6 red wine and Table 7 white wine performance analysis results for the balanced classes for each model is examined.

From these unbalancing and balancing classes, we achieved a better performance result on the balanced class for all the models.

Among the three algorithms, the artificial neural network (ANN) algorithm achieved the best performance result from both red and white wine datasets as compare to the support vector machine (SVM) and naïve Bayes (NB) algorithm. But they differ from this project in different ways.

Kumar, (2020) paper is similar in that they used similar performance measurements and similar machine learning algorithms such as support vector machine and naïve Bayes. The difference is that they trained the model on unbalanced classes and they used all features for the prediction of the model. In terms of performance analysis, they achieved the best of 67.25% accuracy from the support vector machine on the red wine dataset, Er and Atasoy, (2016) has been achieved the best accuracy result from the random forest on 69.90% in the red wine and 71.23% white wine datasets and use the principal components analysis technique for feature selection. Gupta, (2018) has been proposed that all features are not necessary for the prediction instead of selecting only necessary features to predict the wine quality. For that, they used linear regression for determining the dependencies of the target variable. Whereas our model achieved 69.06% accuracy in the red wine dataset and 67.83% accuracy in the white wine dataset from the support vector machine. Then after training, the model on the balanced data and selecting the best hyperparameters the performance of the model is improved and achieved 83.52% accuracy in the red wine and 86.86% accuracy in the white wine. In addition, our model achieved the best 85.16% accuracy in the red wine and 88.28% accuracy in the white wine from the artificial neural network model by applying the Pearson coefficient correlation matrices for the feature selection.

**CONCLUSION AND FUTURE ENHANCEMENT**

## CONCLUSION:

Predicting wine quality with machine learning techniques has showed promise recently. Through the use of several features, including chemical composition, sensory attributes, and ambient conditions, machine learning algorithms have demonstrated the ability to reliably forecast wine. quality ratings. These forecasts can help winemakers improve quality control procedures, streamline production operations, and ultimately raise the standard of their wines. Wine quality prediction challenges have seen the effective use of machine learning models, including random forests, support vector machines, and neural networks, which have demonstrated excellent levels of accuracy and generalization. The performance of these models can be greatly enhanced by feature engineering and selection, since the right feature selection can lower noise and increase prediction accuracy. But issues such the lack of data, unbalanced datasets, and model interpretability continue to be causes for concern. Subsequent investigations may concentrate on tackling these obstacles by gathering mor extensive and varied datasets, creating methods for managing unbalanced data, and improving the comprehensibility of machine learning models to facilitate improved decision making. The Wine Quality dataset from the UCI Machine Learning Repository is a popular dataset often used for predicting wine quality.

## FUTURE ENHANCEMENT:

In the future, to improve the accuracy of the classifier, it is clear that the algorithm or the data must be adjusted. We recommend feature engineering, using potential relationships between wine quality, or applying the boosting algorithm on the more accurate method.

In addition, by applying the other performance measurement and other machine learning algorithms for the better comparison on results. This study will help the manufacturing industries to predict the quality of the different types of wines based on certain features, and also it will be helpful for them to make a good product.

By adopting these advanced machine learning approaches, the wine industry can significantly enhance its ability to predict and improve wine quality, leading to more consistent and superior product.

**1**. **Data Collection and Integration**

Comprehensive Data Gathering: Collecting extensive data from vineyards, including soil composition, weather conditions, grape health, and vineyard management practices.

Sensor Networks: Deploying IoT devices and sensors in vineyards and wineries to continuously monitor environmental conditions, fermentation processes, and other critical parameters.

Integration of External Data: Utilizing external datasets such as market trends, consumer reviews, and competitive analysis to refine predictive models.

**2.** **Advanced Machine Learning Techniques**

Deep Learning: Applying deep learning models, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), to analyze complex patterns and interactions in large datasets.

Ensemble Methods: Combining multiple machine learning models (e.g., random forests, gradient boosting machines) to improve prediction accuracy and robustness.

Transfer Learning: Using pre-trained models from related domains and fine-tuning them with specific wine data to enhance predictive performance.

**3**. **Feature Engineering and Selection**

Advanced Feature Engineering: Identifying and creating new features from raw data that have a significant impact on wine quality. This may include biochemical markers, spectral data from grape analysis, and fermentation metrics.

Automated Feature Selection: Implementing algorithms such as Recursive Feature Elimination (RFE) and Principal Component Analysis (PCA) to automatically select the most relevant features for quality prediction.

**4**. **Model Training and Optimization**

Hyperparameter Tuning: Using techniques like grid search, random search, and Bayesian optimization to find the optimal hyperparameters for machine learning models.

Cross-Validation: Employing robust cross-validation methods to ensure models are not overfitting and can generalize well to new data.

Real-Time Learning: Implementing online learning algorithms that continuously update the model as new data becomes available.

**5**. **Explainable AI and Interpretability**

Model Interpretability: Developing interpretable machine learning models using techniques such as SHAP (Shapley Additive explanations) and LIME (Local Interpretable Model-agnostic Explanations) to provide insights into the factors influencing wine quality.

Transparent Predictions: Providing winemakers with clear, actionable insights and recommendations based on model predictions to facilitate informed decision-making.

**6.** **Deployment and Integration**

Cloud-Based Platforms: Utilizing cloud infrastructure to deploy scalable machine learning models that can be accessed by winemakers globally.

User-Friendly Interfaces: Developing intuitive dashboards and mobile applications to allow winemakers to easily interact with predictive models and visualize data insights.

Integration with Winery Systems: Seamlessly integrating predictive models with existing winery management systems for automated data collection and real-time quality monitoring.

**7.** **Continuous Improvement and Feedback Loops**

Feedback Mechanisms: Establishing feedback loops where winemakers can provide data on actual wine quality outcomes, which can be used to continuously improve model accuracy.

Iterative Model Refinement: Regularly updating models with new data and insights to ensure they remain accurate and relevant over time.

# 

# REFERENCES

[1] Kumar, Sunny, Kanika Agrawal, and Nels h a n Mandan. "Red wine quality prediction using machine learning techniques." 2020 International Conference on Computer Communication and Informatics (ICCCI). IEEE, 2020.

[2] Chen, Bernard, et al. "Wine informatics: Applying data mining on wine sensory reviews processed by the computational wine wheel." 2014 IEEE International Conference on Data Mining Workshop. IEEE, 2014.

[3] Dahal, K. R., J. N. Dahal, H. Ban jade, and S. Gaire. "Prediction of wine quality using machine learning algorithms." Open Journal of Statistics 11, no. 2 (2021): 278-289.

[4] Riu l Jr, A., de Sousa, H. C., Malme grim, R. R., dos Santos Jr, D. S., Carvalho, A. C., Fonseca, F. J., ... & Matto so, L. H. (2004). Wine classification by taste sensors made

[5] Baldacci, Stefano, Tetsuya Matsuno, Kiyoshi Toko, Rita Stella, and Danilo De Rossi. "Discrimination of wine using taste and smell sensors." Sens. Mater 10, no. 3 (1998): 185-200.

[6] Bhardwaj, Piyush, et al. "A machine learning application in wine quality prediction." Machine Learning with Applications 8 (2022): 100261.

[7] Aich, Satya brata, et al. "A classification approach with different feature sets to predict the quality of different types of wine using machine learning techniques." 2018 20th International conference on advanced communication technology (ICACT). IEEE, 2018.

[8] Lee, S., Park, J. and Kang, K., 2015, September. Assessing wine quality using a decision tree. In 2015 IEEE International Symposium on Systems Engineering (ISSE) (pp. 176-178). IEEE.

[9] Janszky, I., et al. "Wine drinking is associated with increased heart rate variability in women with coronary heart disease." Heart 91.3 (2005): 314-318.

[10] Cortez, P., & Morais, A. (2009). A Data Mining Approach to Predict Forest Fires using Meteorological Data. In J. Neves, M. F. Santos, & J. Machado Eds., New Trends in Artificial Intelligence (pp. 512-523). Springer Berlin Heidelberg. DOI: 10.1007/978-3-642- 11164-8\_49

[11] Pande, S. M., & Sawant, S. S. (2020). Wine Quality Prediction using Machine Learning Techniques. In 2020 International Conference on Artificial Intelligence and Signal Processing (AISP) (pp. 1-5). IEEE. DOI: 10.1109/AISP49506.2020.9155123

[12] UCI Machine Learning Repository. (n.d.). Wine Quality Data Set. Retrieved from <https://archive.ics.uci.edu/ml/datasets/wine+quality>

[13] Lima, L. N., Sousa, R. S., & Santos, J. A. (2021). Predicting Wine Quality using Random Forest and Decision Tree Algorithms. In 2021 IEEE Symposium Series on Computational Intelligence (SSCI) (pp. 1-7). IEEE. DOI: 10.1109/SSCI51683.2021.9569617

**APPENDIX**

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.model\_selection import train\_test\_split

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy\_score

Importing the dependencies

Data Collection

# loading the dataset to a Pandas DataFrame

wine\_dataset = pd.read\_csv('/content/winequality-red.csv')

# number of rows & columns in the dataset

wine\_dataset.shape

# first 5 rows of the dataset

wine\_dataset.head()

# checking for missing values

wine\_dataset.isnull().sum()

Data Analysis and Visualization

# statistical measures of the dataset

wine\_dataset.describe()

# number of values for each quality

sns.catplot(x='quality', data = wine\_dataset, kind = 'count')

# volatile acidity vs Quality

plot = plt.figure(figsize=(5,5))

sns.barplot(x='quality', y = 'volatile acidity', data = wine\_dataset)

# citric acid vs Quality

plot = plt.figure(figsize=(5,5))

sns.barplot(x='quality', y = 'citric acid', data = wine\_dataset)

Correlation

1. Positive Correlation

2. Negative Correlation

correlation = wine\_dataset.corr()

# constructing a heatmap to understand the correlation between the columns

plt.figure(figsize=(10,10))

sns.heatmap(correlation, cbar=True, square=True, fmt = '.1f', annot = True, annot\_kws={'size':8}, cmap = 'Blues')

# New section

Data Preprocessing

# separate the data and Label

X = wine\_dataset.drop('quality',axis=1)

print(X)

Label Binarization

Y = wine\_dataset['quality'].apply(lambda y\_value: 1 if y\_value>=7 else 0)

print(Y)

Train & Test Split

X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X, Y, test\_size=0.2, random\_state=3)

print(Y.shape, Y\_train.shape, Y\_test.shape)

Model Training:

Random Forest Classifier

model = RandomForestClassifier()

model.fit(X\_train, Y\_train)

Model Evaluation

Accuracy Score

# accuracy on test data

X\_test\_prediction = model.predict(X\_test)

test\_data\_accuracy = accuracy\_score(X\_test\_prediction, Y\_test)

print('Accuracy : ', test\_data\_accuracy)

Building a Predictive System

input\_data = (7.5,0.5,0.36,6.1,0.071,17.0,102.0,0.9978,3.35,0.8,10.5)

# changing the input data to a numpy array

input\_data\_as\_numpy\_array = np.asarray(input\_data)

# reshape the data as we are predicting the label for only one instance

input\_data\_reshaped = input\_data\_as\_numpy\_array.reshape(1,-1)

prediction = model.predict(input\_data\_reshaped)

print(prediction)

if (prediction[0]==1):

print('Good Quality Wine')

else:

print('Bad Quality Wine')