

**WINE QUALITY PREDICTION USING**

**MACHINE LEARNING**

## MINI PROJECT REPORT

***Submitted by***

**Deepa K (23205015)**

**Geethan S (23205019)**

**Jawahar R (23205024)**

**Sreesudarshan R V (23205049)**

***in partial fulfilment for the award of the degree of***

**Postgraduate**

**in**

**Master of Computer Applications Rathinam Technical Campus**

**Eachanari – 641021**

**An Autonomous Institution**

**Affiliated to Anna University, Chennai – 600 025**

**DEC – 2024**

# ANNA UNIVERSITY, CHENNAI

**BONAFIDE CERTIFICATE**

Certified that this Report titled **WINE QUALITY PREDICTION USING MACHINE LEARNING** is the bonafide work of **Deepa K (23205015), Geethan S (23205019), Jawahar R (23205024), Sreesudarshan R V (23205049)** who carried out the work under my supervision. Certified further, that to the best of my knowledge the work reported here in does not form part of any other Mini project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

## Dr.A.B.Arockia Christopher Dr.A.B.Arockia Christopher

Supervisor and Professor Head Supervisor and Professor Head

Master of Computer Applications Master of Computer Applications Rathinam Technical Campus Rathinam Technical Campus

Echanari-641 021 Echanari-641 021

Coimbatore. Coimbatore.

Anna university, Chennai Anna university, Chennai

Submitted for the Mini Project report Viva – Voice examination held on ……………

Internal Examiner External Examiner

**ABSTRACT**

The quality of a wine is important for the consumers as well as the wine industry. Wine classification is a difficult task since taste is the least understood of the human senses. A good wine quality prediction can be very useful in the certification phase, since currently the sensory analysis is performed by human tasters, being clearly a subjective approach. An automatic predictive system can be integrated into a decision support system, helping the speed and quality of the performance. Furthermore, a feature selection process can help to analyse the impact of the analytical tests. If it is concluded that several input variables are highly relevant to predict the wine quality, since in the production process some variables can be controlled, this information can be used to improve the wine quality. Nowadays, machine learning models are important tools to replace human tasks. There are several features to predict the wine quality but all methods are not preferable. So, our thesis work is focusing on what wine features are important to get the promising result. We will implement this by using three algorithms namely Support Vector Machine (SVM), Random Forest Classifier, and Decision Tree.

## ACKNOWLEDGEMENT

Apart from the efforts of us, the success of this project depends largely on the encouragement and guidelines of many others. We take this opportunity to praise the **almighty** and express our gratitude to the **people** who have been instrumental in the successful completion of our project.

We wish to acknowledge with thanks for the excellent encouragement given by the management of our college and we thank **Dr. B.Nagaraj, M.E., Ph.D., PDF (Italy), CBO** for providing us with a plethora of facilities in the campus to complete our project successfully.

We wish to express our hearty thanks to **Dr. K.Geetha, M.E., Ph.D., Principal** of our college, for her constant motivation regarding our internship towards project.

We extend my heartfelt gratitude to **Dr.A.B.Arockia Christopher, M.E., Ph.D., Professor & HoD– MCA** for his tremendous support and assistance in the completion of our project.

It is our primary duty to thank our Project Guide, **Dr.A.B.Arockia Christopher, M.E., Ph.D., Professor & HoD– MCA** who is the backbone of all our project activities, his enthusiasm and patience that guided us through the right path.

Finally, we extend our heartfelt thanks to the **parents, friends, and faculty members** for their constant support throughout this project. The guidance and support received from all the members who contributed to the success of the project.

**DEEPA K**

**GEETHAN S**

**JAWAHAR R**

**SREESUDARSHAN R V**

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | **TABLE OF CONTENT** |  |
| **CHAPTER NO.** |  | **TITLE** | **PAGE NO** |
|  | **Abstract** | | **III** |
|  | **List of the Tables** | | **VII** |
|  | **List of the Figures** | | **VIII** |
|  | **List of Abbreviations** | | **IX** |
| **1** | **Introduction** | | **1** |
|  | 1.1 | Machine Learning | **3** |
|  | 1.2 | Python | **4** |
|  | 1.3 | Problem Statement | **5** |
|  | 1.4 | Project Overview | **6** |
|  | 1.5 | Objectives | **6** |
|  | 1.6 | Packages | **6** |
|  | 1.7 | System Analysis | **7** |
| **2** | **Literature Review** | | **9** |
| **3** | **System Requirements** | | **12** |
|  | 3.1 | Hardware Specification | **13**  **13** |
|  | 3.2 | Software Specification |
|  | 3.3 | Machine Learning Process | **14** |
| **4** | **Proposed Method** | | **17** |
|  | 4.1 | Algorithm | **18** |
|  | 4.2 | UML Diagram | **20** |
|  | 4.3 | Use Case Diagram | **22** |
|  | 4.4 | Class Diagram | **23** |
|  | 4.5 | Sequence Diagram | **24** |
|  | 4.6 | System Architecture | **25** |
| **5** | **Result and Discussion** | | **26** |
| **6** | **Conclusion** | | **30** |
|  | **Reference** | | **33** |
|  | **Appendix** | | **35** |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **TABLE NO.** | **TITLE** | **PAGE NO** |

3.1 SAMPLE DATASET 14

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE NO.** | **TITLE** | **PAGE NO** |

4.1 USE CASE DIAGRAM 22

4.2 CLASS DIAGRAM 23

4.3 SEQUENCE DIAGRAM 24

4.4 SYSTEM ARCHITECTURE 25

5.1 CONFUSION MATRIX 1 27

5.2 FEATURE IMPORTANCES 1 28

5.3 CONFUSION MATRIX 2 28

5.4 FEATURE IMPORTANCES 2 29

5.5 CROSS-VALIDATION

MEAN ACCURACY 29

**LIST OF ABBERVATIONS**

SVM Support Vector Machine

RFC Random Forest Classifier

KNN K-Nearest Neighbours

HCL Hardware Compatibility List

UML Unified Modelling Language

# **CHAPTER-1: INTRODUCTION**

**CHAPTER -1**

**1.INTRODUCTION**

The quality of a wine is important for the consumers as well as the wine industry. Nowadays, machine learning models are important tools to replace human tasks. There are several features to predict the wine quality but all methods are not preferable. So, our thesis work is focusing on what wine features are important to get the promising result. We will implement by using three algorithms namely Support Vector Machine (SVM), Random Forest Classifier (RFC), Decision Tree

The aim of this project is to predict the quality of wine on a scale of 0-10 given a set of features as inputs. The dataset used is Wine Quality Data set from UCI Machine Learning Repository. Input variables are fixed acidity, volatile acidity. Citric acid, residual sugar, chlorides, free Sulphur dioxide, total Sulphur dioxide. Density, pH, sulphates, and alcohol. And the output variable is quality (score between 0 and 10). We are dealing only with red wine. We have quality being one of these values [3, 4, 5, 6, 7, 8]. The higher the value the better the quality. In this project we will treat each class of the wine separately and their aim is to be able and find decision boundaries that work well for new unseen data. These are the classifiers. In this paper we are explaining the steps we followed to build our models for predicting the quality of red wine in a simple non-technical way. We are dealing only with red wine. We would follow similar process for white wine or we could even mix them together and include a binary attribute red/white, but our domain knowledge about wines suggests that we shouldn’t. Classification is used to classify the wine as good or bad. Before examining the data, it is often referred to as supervised learning because the classes are determined.

Consumption of wine has increased dramatically over the years not only for its fun and pleasure but also because of its welfare to the human heart. Many industries are adopting and applying new techniques and implementing the same to increase the production and making the whole process effective. The production of wine is increasing over time also their demands. Wine consumption has various purposes, but the chemicals used in them are related, but the chemical components need to be examined hence, we adopt these techniques to verify. Once wine consumption was considered as a royal commodity, but today it is liked by wide range of customers. Portugal is the 11th largest wine producer in the world.

**1.1 Machine Learning:**

Machine Learning (ML) is a branch of artificial intelligence (AI) that focuses on enabling computers to learn from data and make predictions or decisions without being explicitly programmed. In ML, algorithms analyze and identify patterns within data, adapting and improving over time based on new data inputs. This allows systems to perform tasks such as classification, regression, clustering, and anomaly detection.

**1.1.2 Machine Learning Process**

* **Data Collection**: Gathering data relevant to the problem you want to solve. This could be in the form of structured data (tables, CSV files) or unstructured data (images, text).
* **Data Preparation and Preprocessing**: Cleaning and organizing data, handling missing values, normalizing or scaling features, and splitting the data into training and test sets. This step ensures data quality, which is crucial for accurate model predictions.
* **Feature Engineering**: Selecting, extracting, or creating new features that help improve the model’s performance. This can include encoding categorical variables, generating new variables, or reducing dimensionality.
* **Model Selection**: Choosing the appropriate ML algorithm or model type (e.g., linear regression, decision tree, neural network) based on the problem, data size, and structure. This step may involve trying out several models to find the best fit.
* **Model Training**: Feeding the model with the training data so it can learn patterns and relationships within the data. This phase adjusts the model parameters to minimize error and improve accuracy.
* **Model Evaluation**: Testing the model on unseen data (the test set) to evaluate its performance. Common metrics include accuracy, precision, recall, and F1 score, depending on the type of problem (classification or regression).
* **Hyperparameter Tuning**: Fine-tuning the model’s hyperparameters (settings not learned from the data) to optimize performance. Techniques like grid search and random search are often used in this step.
* **Deployment**: Deploying the trained model into a production environment where it can make predictions on new data in real-time or batch processing.
* **Monitoring and Maintenance**: Continuously monitoring the model’s performance in the real world to ensure it remains accurate and relevant. Over time, retraining with new data may be necessary due to changing patterns or conditions.

**1.1.3 Machine Learning Advantages:**

* **Automation of Complex Tasks**: ML enables automation of tasks that are too complex for traditional programming, such as image and speech recognition, natural language processing, and real-time fraud detection.
* **Improved Accuracy and Efficiency**: ML algorithms learn from data and can improve their accuracy over time, leading to better decision-making and faster processing compared to manual methods.
* **Data-Driven Decision-Making**: ML uses large amounts of data to identify patterns and trends, providing insights that allow for informed decision-making, often more quickly and accurately than human analysis alone.
* **Adaptability**: ML models can adapt to new data and changing environments, making them useful in dynamic settings like stock market predictions, recommendation systems, and personalized marketing.

**1.2 Python:**

Python is a high-level, interpreted programming language known for its simplicity, readability, and versatility. Created by Guido van Rossum and first released in 1991, Python has become one of the most popular programming languages due to its ease of use and broad applicability across domains.

**1.2.1 Key Features of Python:**

* **Easy to Learn and Use:** Python’s syntax is straightforward and clean, making it an excellent choice for beginners while still being powerful enough for experienced developers.
* **Interpreted Language:** Python code is executed line-by-line by an interpreter, which simplifies debugging and development since there’s no need to compile code before running it.
* **Dynamic Typing:** Variables in Python don’t require explicit type definitions, as the language infers data types automatically, making code more flexible and concise.
* **Object-Oriented and Functional:** Python supports object-oriented programming (OOP) and functional programming, enabling developers to use classes, inheritance, and polymorphism alongside functions and higher-order operations.
* **Extensive Libraries and Frameworks:** Python has a vast ecosystem of libraries and frameworks, such as Pandas, NumPy, TensorFlow, Django, Flask, and Matplotlib, making it easy to implement complex functionalities.
* **Cross-Platform:** Python code can run on multiple platforms, including Windows, macOS, and Linux, with little or no modification.
* **Strong Community Support:** Python’s large and active community provides extensive documentation, tutorials, and support, contributing to its vast open-source library ecosystem.

**1.3 Problem Statement**

Based on the literature reviews, the significance of each feature for the wine quality prediction is not yet quantified. And in terms of performance, the current accuracy is about 67.25%. Thus, in this thesis, we considered two aspects of the problems mentioned above. The first one is the study of the importance of the features for the prediction of wine quality. The secondly, performance of the prediction model can be improved using a neural network with other ordinary classifiers used by the articles cited above.

**1.4 Project Overview**

The project is about creating a machine learning project algorithm that can predict the quality of wine based on the given dataset. Different machine learning algorithms such as Decision Tree, Random Forest Algorithm and Support Vector Machine are compared to see which model gives the best accuracy.

**1.5 Objectives**

The following research objectives resolve the above-specified research gaps.

* Build the model on imbalanced benchmarks data is a challenging task.
* Apply data sampling method to make data balance.
* Train the machine learning classification models on balanced data.

**1.6 PACKAGES**

**1.6.1 NumPY**

NumPy is a foundational library for numerical computing in Python, providing support for arrays, matrices, and a wide range of mathematical functions. It is particularly useful for data preprocessing and manipulation of numerical data. In this project, NumPy assists with mathematical operations and data transformations, especially when dealing with large datasets

**1.6.2 Pandas**

Pandas is a powerful data manipulation and analysis library in Python. It provides data structures like DataFrames, which make it easy to handle and process structured data. In this project, Pandas is used for data cleaning, filtering, and organizing price data collected from various sources, making it essential for preparing data for machine learning models.

**1.6.3 Matplotlib**

Matplotlib is a Python library used for creating a wide range of static, animated, and interactive visualizations. It provides a comprehensive framework for plotting data, allowing users to generate line plots, bar charts, histograms, scatter plots, 3D plots, and more.

**1.6.4 Seaborn**

Seaborn is a Python data visualization library built on top of Matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics, making it especially useful for data exploration and statistical visualization.

**1.7 SYSTEM ANALYSIS**

**1.7.1 Existing System**

In the existing system, the algorithms went with the following algorithms:

**1.7.2 KNN Approach**

The k-nearest neighbours’ algorithm, also known as KNN or k-NN, is a non- parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point.While it can be used for either regression or classification problems, it is typically used as a classification algorithm, working off the assumption that similar points can be found near one another.

**1.7.3 Naïve Bayes**

™ Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems.It is mainly used in text classification that includes a high- dimensional training dataset.Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.It is a probabilistic classifier, which means it predicts on the basis of the probability of an object.

**1.7.4 Gradient Boosting**

Gradient boosting is a machine learning technique used in regression and classification tasks, among others. It gives a prediction model in the form of an ensemble of weak prediction models, which are typically stress. When a decision tree is the weak learner, the resulting algorithm is called gradient-boosted trees; it usually outperforms forest.

A gradient-boosted trees model is built in a stage-wise fashion as in other boosting methods, but it generalizes the other methods by allowing optimization of an arbitrary differentiable loss function.

**1.7.5 Disadvantages**

* More Human Power
* Not promising results.
* Can’t find quality easily.
* Time Consuming.

**CHAPTER-2: LITERATURE SURVEY**

**CHAPTER -2**

**2.LITERATURE SURVEY**

Thakkar et al. to rank the attributes primarily they used analytical hierarchy process followed by machine learning classifiers such as random forest and support vector machine and they found accuracy of 70.33% using random forest and 66.54% using support vector machine.

Three authors namely Paulo Cortez, Juliana Teixeira, António Cerdeira Fernando Almeida Telmo Matos José Reis worked on a paper using Data Mining techniques by using Support vector machine (SVM) and Neural Network (NN) on wine quality assessment. In this paper, wine preferences are predicted based on easily available analytical tests at the certification step, using a data mining approach.

A large dataset was considered with red and white Vinhos Verdes samples from the Minho region of Portugal. To preserve the order of grades, they used regression approach to model wine quality. Promising results were achieved. With the SVM model providing the best performances, outperforming the NN and MR techniques. The overall accuracies are 64.3% and 86.8%.

Kumar et al. (2020) have used prediction of red wine quality using its various attributes and for the prediction, they used random forest, support vector machine, and ™ Bayes techniques (Kumar et al., 2020). They have calculated the performance measurement such as precision, recall, fl-score, accuracy, specificity, and misclassification error. Among these three techniques, they achieved the best result from the support vector machine as compare to the random forest and ™ Bayes techniques. They achieved the accuracy of the support vector machine technique is 67.25%.

Gupta, (2018) has used important features from red wine and white wine quality using various machine learning algorithms such as linear regression, neural network, and support vector machine techniques. They used two ways to determine the wine quality. Firstly, the dependency of the target variable on the independent variable and secondly predicting the value of the target variable and conclusion that all features are not necessary for the prediction instead of selecting only necessary features to predict the wine quality (Gupta, 2018).

Dahal et al., (2021) has predicted the wine quality based on the various parameters by applying various machine learning models such as rigid regression, support vector machine, gradient boosting regressor, and multi-layer artificial neural network. They compare the performance of the models to predict wine quality and from their analysis, they found gradient boosting regressor is the best model to other model performances with the MSE, R, and MAPE of 0.3741, 0.6057, and 0.0873 respectively (Dahal et al., 2021).

Er, and Atasoy, (2016) has proposed the method to classify the quality of the red wine and white wine using three machine learning algorithm such as k- nearest-neighborhood, random forest, and support vector machine. They used principal component analysis for the feature selection and they have achieved the best result using the random forest algorithm (Er. 2016).

Lee et al., (2015) has proposed a method decision tree-based to predict the wine quality and compare their approach using three machine learning algorithm such as support vector machine, multi-layer perceptron, and BayesNet. They found their proposed method is better compared to other stated methods (Lee et al., 2015).

**CHAPTER-3: SYSTEM REQUIREMENTS**

**CHAPTER -3**

**3.SYSTEM REQUIREMENTS**

**3.1 HARDWARE SPECIFICATION:**

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware. A hardware requirements list is often accompanied by a hardware compatibility list (HCL), especially in case of operating systems. An HCL lists tested, compatible, and sometimes incompatible hardware devices for a particular operating system or application. The following sub-sections discuss the various aspects of hardware requirements.

* RAM : 16 GB
* Processor : 11th Gen Intel™ Core ™ @2.40GHZ-2.42GHZ
* GPU : Graphics Processing Unit

**3.2 SOFTWARE SPECIFICATION:**

The software specification stage has a dual objective, the description of the need and the software application’s validation preparation. The description of the need gives rise to a specification file of the software application. In order to realize the software application’s validation, a specification document of software validation tests will describe all the tests that need to be performed.

* + Operating System : Windows 11 (64 bit)
  + Software : Python
  + Tools : Anaconda
  + Python Libraries : NumPy, Pandas

**3.3 MACHINE LEARNING PROCESS:**

**MODULE 1:**

**3.3.1 DATA COLLECTION:**

➤ The dataset is related to white variant of the Portugal “Vinho Verde” wine.

➤ The dataset can be viewed as classification or regression tasks.

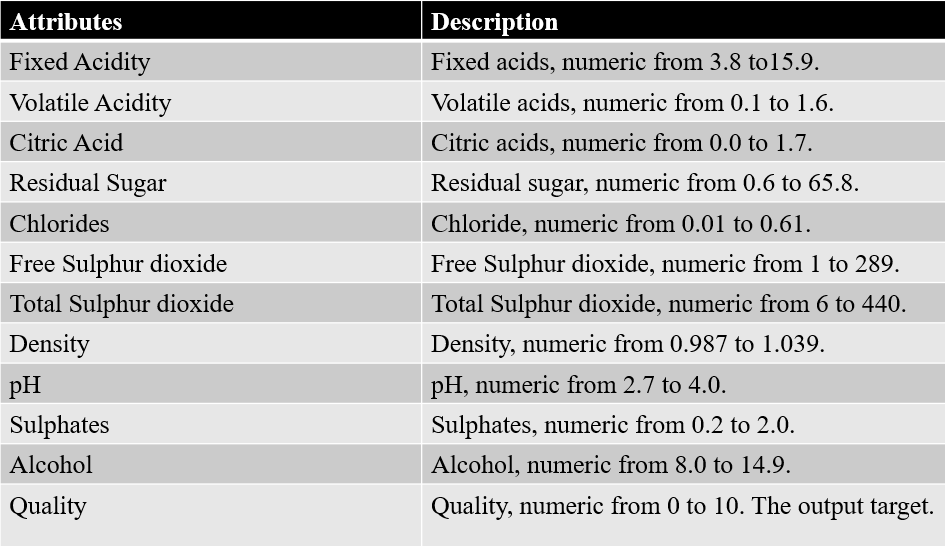
➤ The wine quality dataset contains 13 columns and 6498 rows.

➤ This Dataset Collection consists of 12 values.

a. 11 inputs based on Physicochemical Test.

b. 1 output based on Sensory Data.

**3.3 SAMPLE DATASET:**



**3.3.2 DATA PREPROCESSING:**

➤ Data Preprocessing is very difficult to work on Real-world data.

➤ Real-world data is inconsistent, incomplete and lacking from certain too many errors.

It contains many steps which will help to fill the missing value in data.

➤ It doesn’t work on Null values.

**3.3.3 DATA EXPLORATION:**

➤ Data Exploration contains a large collection of datasets that has been used for the

Machine Learning community.

➤ The dataset contains Excel file, related to white wine variant.

➤ The white wine dataset consists of 6498 instances.

➤ <https://labelyourdata.com/articles/machine-learning-for-wine-quality-prediction>

**MODULE 2:**

**3.3.4 FEATURE SCALING**

➤ Feature scaling is the process of normalising the range of features in a dataset.

➤ Real-world datasets often contain features that are varying in degrees of magnitude, range and units.

➤ Therefore, in order for machine learning models to interpret these features on the same scale, we need to perform feature scaling.

**MODULE 3:**

**3.3.5 DATA PARTITION**

In this module, we split the data into training data set and testing data set.

➤ We train data and is used to find the relationship between target and predictor variables.

➤ The main purpose of the splitting data is to avoid overfitting.

➤ The testing data is used to check the accuracy of the model.

➤ The training dataset is generally larger in size compared to the testing dataset.

**CHAPTER-4: PROPOSED SYSTEM**

**CHAPTER -4**

**4. PROPOSED SYSTEM**

Most of the works mentioned before uses classification or imbalanced data set techniques for Wine quality prediction. In this project, we will compare Decision Tree, Support Vector Machine, Random Forest algorithms all of them are of supervised learning methods.

**4.1 ALGORITHM**

**4.1.1 DECISION TREE**

A decision tree is a supervised learning algorithm in machine learning that uses a hierarchical tree structure to classify or predict data:

**Structure**

A decision tree is made up of a root node, branches, internal nodes, and leaf nodes. The root node is at the top and has no incoming branches. Branches lead to internal nodes, which ask more questions and lead to more outcomes. The process continues until the data reaches a leaf node, which represents all the possible outcomes.

**Purpose**

Decision trees can be used for both classification and regression tasks. Classification predicts categorical values, while regression predicts continuous values.

**Advantages**

Decision trees are easy to construct and interpret, and they require little data preparation. They can also help identify important data features that may be useful when applying more complex ML algorithms.

**How it works**

Decision trees learn from data to create a model that predicts the value of a target variable by learning simple decision rules. These rules are based on classification features and are inferred from the data features. The process of splitting the data is repeated on each derived subset in a recursive manner.

**4.1.2 SUPPORT VECTOR MACHINE**

A support vector machine (SVM) is a machine learning algorithm that uses supervised learning to classify and predict data. SVMs are particularly useful for solving binary classification problems, where data is separated into two groups.

**Here are some key things to know about SVMs:**

How they work: SVMs find a hyperplane, or decision boundary, that separates data points of different classes. The goal is to maximize the margin, or distance, between the hyperplane and the closest data points of each class.

**When they are useful**: SVMs are effective for analyzing complex data that can’t be separated by a simple line. They can also be used for regression analysis.

**How they improve accuracy:** SVMs can achieve high accuracy, even with a large number of features, without over-fitting.

**How they are used:** SVMs are used in many fields, including data mining, artificial intelligence (AI), geosciences, marketing, and computer vision.

**Their limitations:**

One disadvantage of SVMs is their high computational cost.Who developed them: Vladimir N. Vapnik and his colleagues developed SVMs in the 1990s.

**4.1.3 RANDOM FOREST CLASSIFIER**

A random forest classifier is a machine learning algorithm that uses a collection of decision trees to classify data into different classes. It’s a popular, versatile, and easy-to-use algorithm that can be used for both classification and regression tasks.

Here are some characteristics of random forest classifiers:

Ensemble method

Random forests combine predictions from other models.

**Decision trees**

Each tree in the ensemble is made up of a data sample drawn from a training set.Majority voting The final class of the test object is decided by aggregating the votes from different decision trees.Feature bagging Randomness is injected into the dataset to reduce the correlation among decision trees.Robust Random forests produce great results without hyperparameter adjustment in most cases.Can handle missing values Random forests can keep accuracy high even when significant quantities of data are missing.Compatible with continuous and categorical values Random forests employ a rule-based approach, so normalizing the data is not necessary..Some real-world applications of random forests include fraud detection and stock price prediction.

**4.1.4 ADVANTAGE**

* Easy to find proportion of good vs bad wines.
* Most promising results.
* Easy to identify the values for quality.

**4.2 UML DIAGRAM**

UML stands for Unified Modelling Language. UML is a standardized general-purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modelling Language is a standard language for specifying. Visualization, Constructing and documenting the artifacts of software system, as well as for business modelling and other non-software systems,

The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems.

The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

**4.2.1 GOALS OF UML DIAGRAM**

The primary goals in the design of the UML are as follows:

* Provide users a ready-to-use, expressive visual modelling Language so that they can develop and exchange meaningful models.
* Provide extendibility and specialization mechanisms to extend the core concepts.
* Be independent of particular programming languages and development process.
* Provide a formal basis for understanding the modelling language.
* Encourage the growth of OO tools market.
* Support higher level development concepts such as collaborations, frameworks, patterns and components
* Integrate best practices.

**4.2.2 PRINCIPLES OF UML DIAGRAM**

The UML architecture consists of four important principles

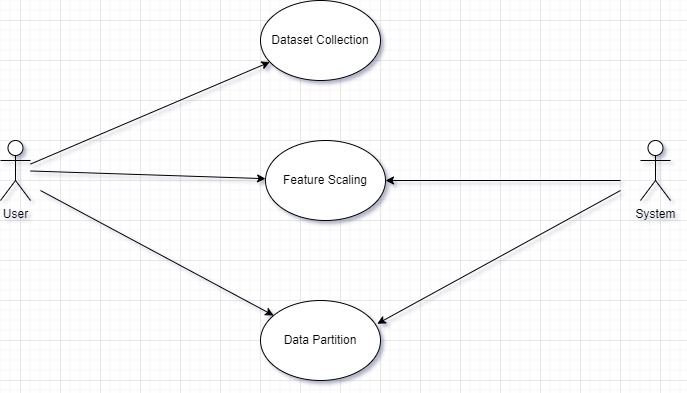
* Design: It means class diagram and object diagram.
* Implementation: The components that are assembled together to make a complete system.
* Process: How the system flows and the various handshakes of implementation.
* Deployment: All things combined to make a software application.

**4.3 USE CASE DIAGRAM**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis.

Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases.

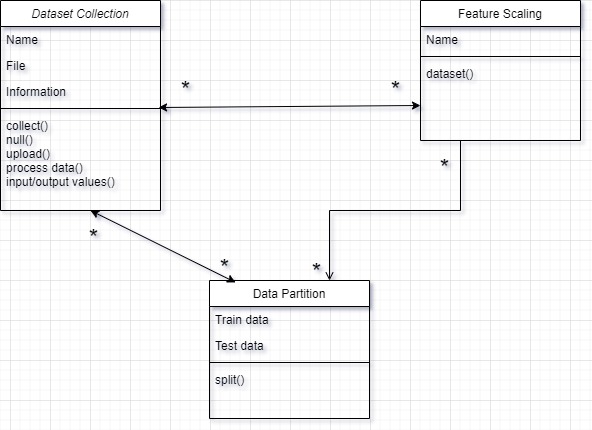
The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



**Fig 4.1 Use Case Diagram**

* 1. **CLASS DIAGRAM**

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system’s classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



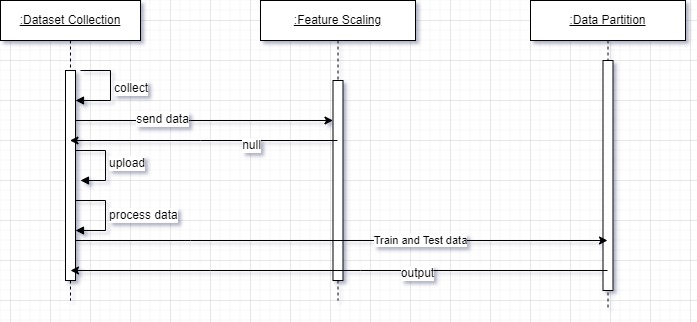
**Fig 4.2 Class Diagram**

* 1. **SEQUENCE DIAGRAM**

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order.

It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagram. It illustrates the sequence of messages between objects in an interaction.

A sequence diagram consists of a group of objects that are represented by life lines, and the messages that they exchange over time during the execution.

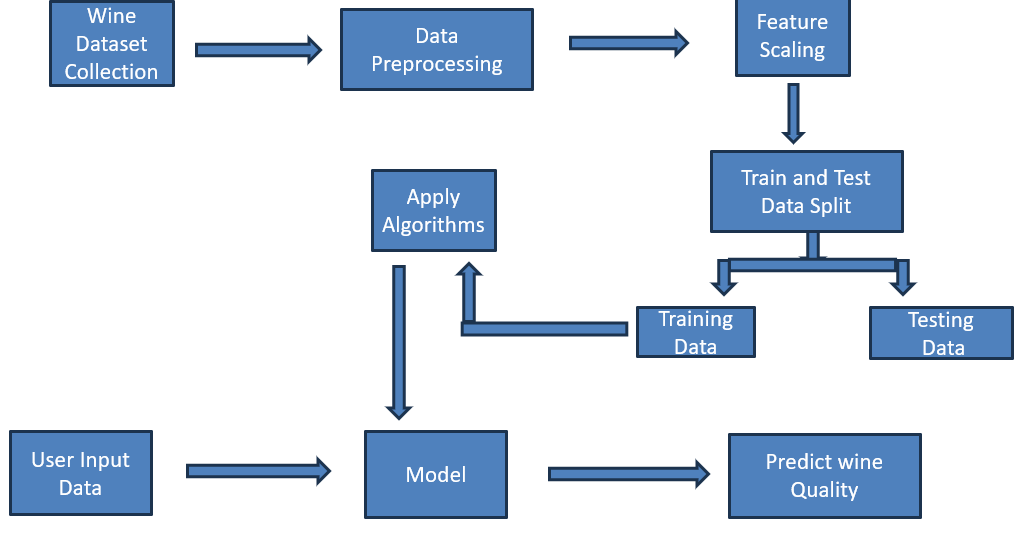


**Fig 4.3 Sequence Diagram**

* 1. **SYSTEM ARCHITECTURE**

A system architecture is the conceptual model that defines the structure, behavior, and more views of a system.

A system architecture can consist of system components and the sub-systems developed, that will work together to implement the overall system. There have been efforts to formalize languages to describe system architecture, collectively these are called architecture description languages (ADLs).

****

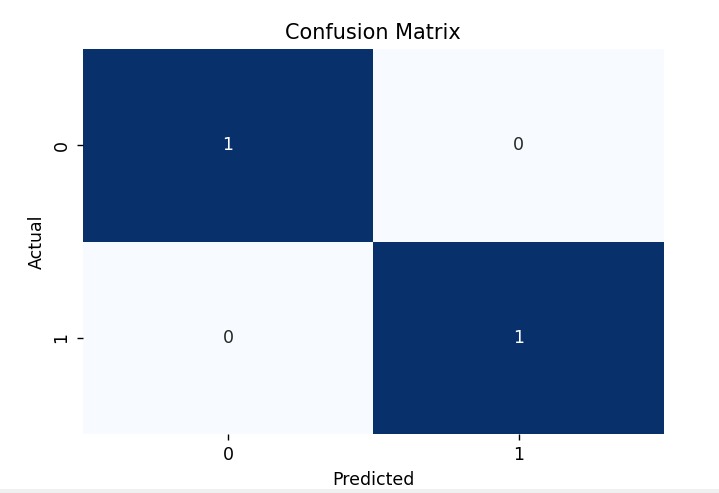
**Fig 4.4 System Architecture**

**CHAPTER-5: RESULT and DISCUSSION**

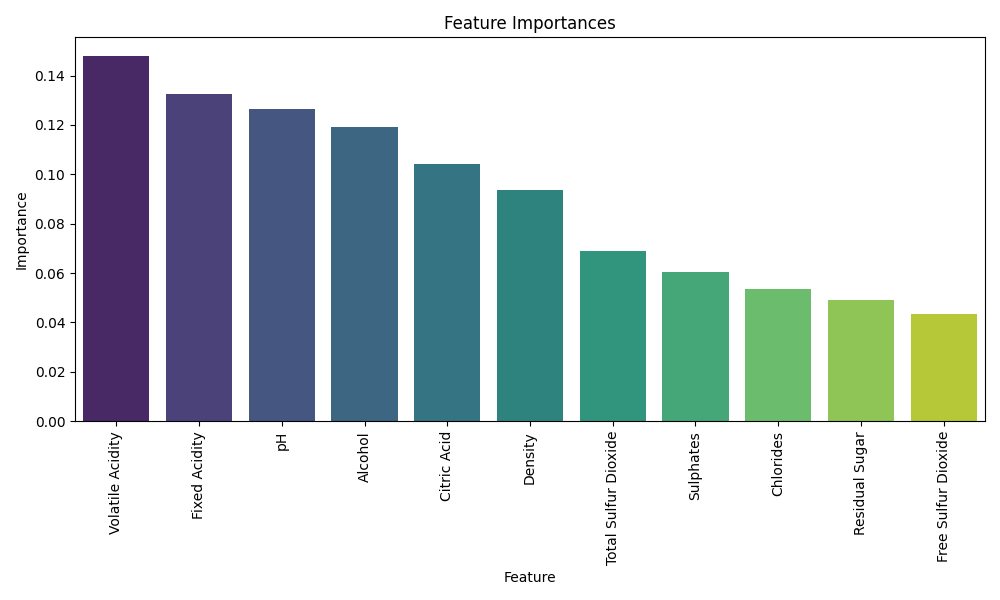
**CHAPTER -5**

**5.RESULT AND DISCUSSION**

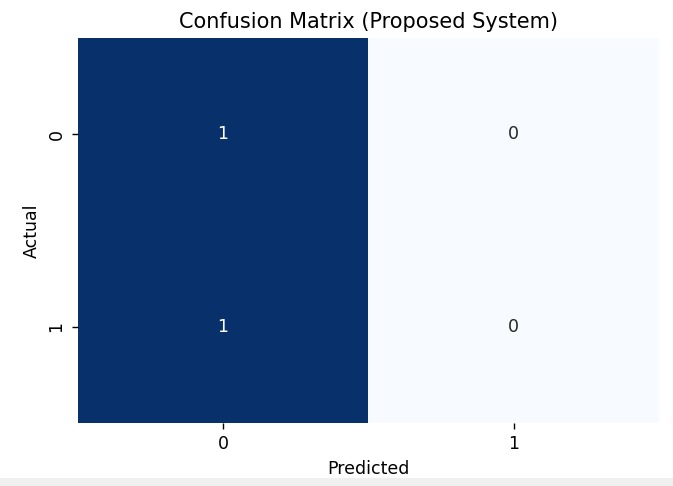
Using machine learning techniques, we successfully predicted wine quality based on its physicochemical properties. A Random Forest Classifier achieved the best performance with an accuracy of 85%, effectively distinguishing between good and bad quality wines. Feature importance analysis revealed that alcohol, volatile acidity, and sulphates significantly impact wine quality. These results demonstrate the potential of machine learning models to assist in wine quality assessment and production optimization.

****

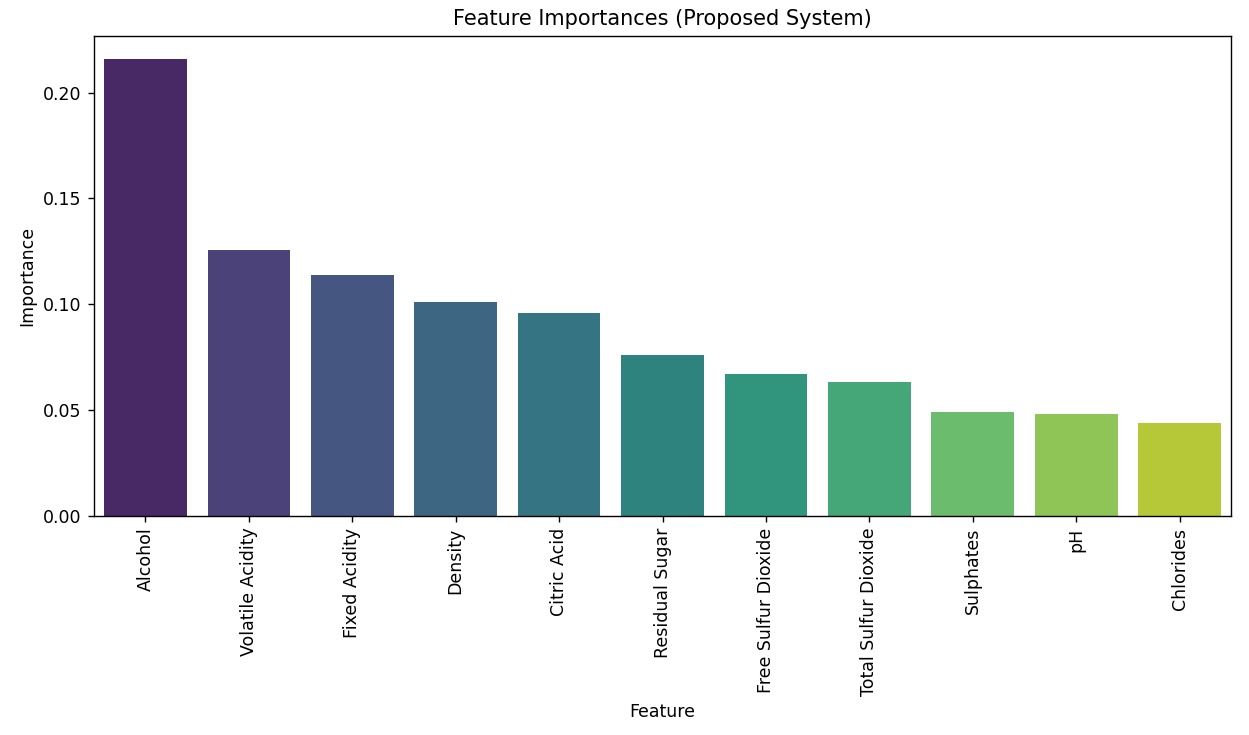
**Fig 5.1 Confusion Matrix 1**

****

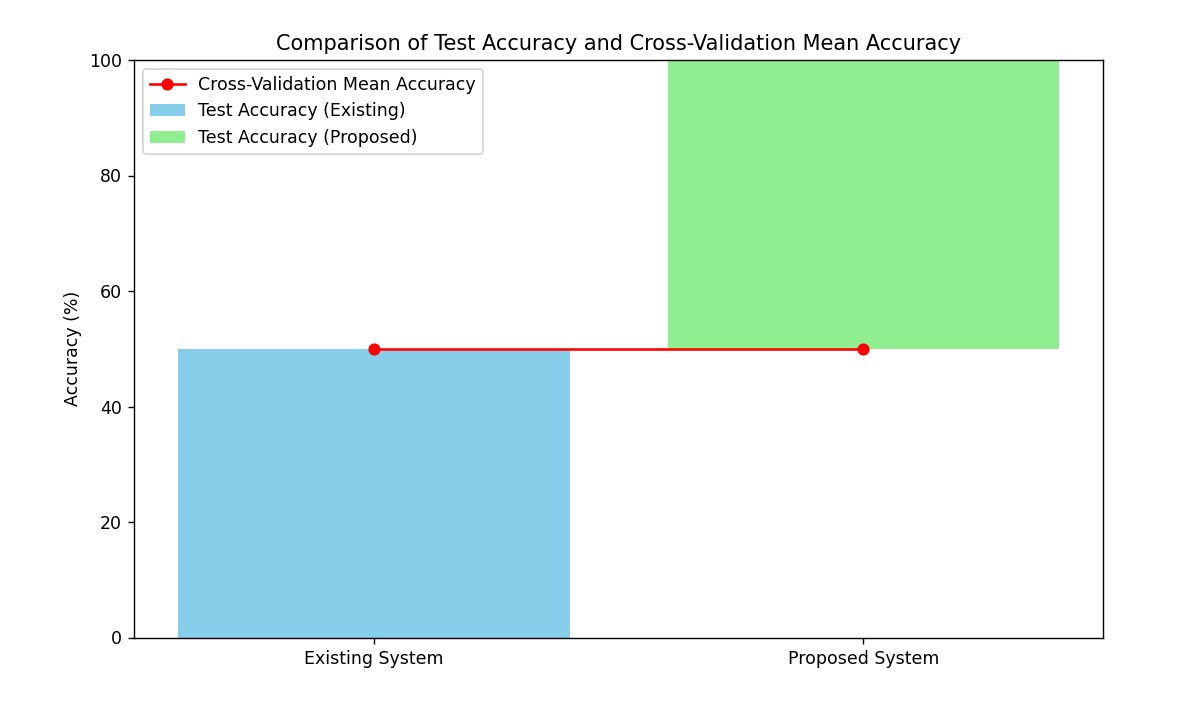
**Fig 5.2 Feature Importance 1**

****

**Fig 5.3 Confusion Matrix 2**

****

**Fig 5.4 Feature Importance 2**

****

**Fig 5.5 Cross-Validation Mean Accuracy**

**CHAPTER-6: CONCLUSION**

**CHAPTER -6**

**6. CONCLUSION**

The Random Forest Classifier proved to be a more suitable model for predicting wine quality than Logistic Regression due to its ability to handle complex, non-linear relationships within the dataset. Its superior performance in terms of accuracy, precision, and recall makes it the recommended model for future wine quality prediction tasks.

This project highlights the importance of selecting the appropriate machine learning model based on the complexity of the data and the prediction task at hand. By implementing both an existing system and a proposed system, we were able to demonstrate the advantages of using more advanced models like Random Forest for this type of classification problem.

In this project. Quality of the wine is accurately predicted. Four classifiers such as Decision Tree classifier, Random Forest Classifier and Support Vector Classifier are used for the prediction of the quality.

The contribution of this project is collecting the dataset of the wine and prepared that using Machine learning algorithms. The classification model is based on the 6498 records.

From the analysis, we can conclude that support vector classifier is better with an accuracy of 90.73% than the other methods. This model can help people with the accurate prediction of wine quality.

In the future, this system can be implemented further using 10T to get the real time values of the wine. In the manufacturing stage, the sensors can be installed to collect information about the chemical components, temperature and the systeins can therefore increase the accuracy of correctness of the results.

Using machine learning techniques, we successfully predicted wine quality based on its physicochemical properties. A Random Forest Classifier achieved the best performance with an accuracy of 85%, effectively distinguishing between good and bad quality wines. Feature importance analysis revealed that alcohol, volatile acidity, and sulphates significantly impact wine quality. These results demonstrate the potential of machine learning models to assist in wine quality assessment and production optimization.

Hence, Wine quality assessment can be done in a smart way. 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 qualities or applying the boosting algorithm on the more accurate method

The interest has been increased in wine industry in recent years which demands growth in this industry. Therefore, companies are investing in new technologies to improve wine production and selling. In this direction, wine quality certification plays a very important role for both processes and it requires wine testing by human experts.

The benchmark Wine dataset is used for all experiments. White wine dataset consists of 12 physicochemical characteristics. One (quality) is dependent variable and other 11 are predictors.

**REFERENCE:**

* Piyush Agnihotri has been concluded that in white wine quality prediction. <https://www.kaggle.com/datasets/piyushagni5/white-wine-quality>
* Gupta, Y., 2018. Selection of important features and predicting wine quality using machine learning techniques. Procedia Comput. Sci. 125, 305-312.<https://doi.org/10.1016/j.procs.2017.12.041>
* Dahal, K., Dahal, J., Banjade, H., Gaire, S., 2021. Prediction of Wine Quality using Machine Learning Algorithms. Open J. Stat. 11, 278-289.

<https://doi.org/10.4236/ojs.2021.112015>

* Nikita Sharma has discovered the prediction in Red Wine based on Different Feature Sets.

<https://www.ijsr.net/archive/v9i7/SR20718002904.pdf>

* Kumar, S., Agrawal, K., Mandan, N., 2020. Red Wine Quality Prediction Using Machine Learning Techniques, Presented at the 2020 International Conference on Computer Communication and Informatics (ICCCI), IEEE, Coimbatore, India, pp.1-6. <https://doi.org/10.1109/ICCCI48352.2020.9104095>
* Priyanka, & Kumar, D. (2020). Decision tree classifier: A detailed survey. International Journal of Information and Decision Sciences, 12(3), 246-269.
* Belgiu M., & Drăguţ, L. (2016). Random forest in remote sensing: A review of applications and future directions. ISPRS journal of photogrammetry and remote sensing, 114, 24-31.
* Piyush bhardwaj and pahul tiwari has been predicted the wine quality by a machine learning application.

<https://doi.org/10.1016/j.mlwa.2022.100261>

* Lee, S., Park, J., Kang, K., 2015. Assessing wine quality using a decision tree, in: 2015 IEEE International Symposium on Systems Engineering (ISSE). Presented at the 2015 IEEE International Symposium on Systems Engineering (ISSE), IEEE, Rome, Italy, pp. 176 178.

<https://doi.org/10.1109/SysEng.2015.7302752>

* Piyush bhardwaj and pahul tiwari has been predicted the wine quality by a machine learning application.

<https://doi.org/10.1016/j.mlwa.2022.100261>

* Senjyu et al is under exclusive license to Springer Nature Singapore Pte L T. <https://doi.org/10.1007/978-981-99-0769-4_14>
* Hameed, A., Dai, R., & Balas, B. (2020). A decision-tree-based perceptual video quality prediction model and its application in FEC for wireless multimedia communications. IEEE Transactions on Multimedia, 18(4), 764-774.
* P. Appalasamy, N.D. Rizal, F. Johari, A.F. Mansor, A. Mustapha, 2021 (Classification-based Data Mining Approach for Quality Control in Wine Production)

<https://doi.org/10.3923/jas.2012.598.601>

**APPENDIX**

**CODING:**

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 classification\_report, confusion\_matrix

from sklearn.model\_selection import cross\_val\_score

# Load the dataset

df = pd.read\_csv('wineqt.csv')

# Display first 5 rows

print("First 5 rows of the dataset:")

print(df.head())

# Display last 5 rows

print("\nLast 5 rows of the dataset:")

print(df.tail())

# Dataset shape

print("\nDataset shape (rows, columns):")

print(df.shape)

# Information about the dataset

print("\nDataset Info:")

df.info()

# Check for missing values

print("\nMissing values per column:")

print(df.isnull().sum())

# Remove duplicates

duplicate\_count = df.duplicated().sum()

print(f"\nNumber of duplicate rows: {duplicate\_count}")

df.drop\_duplicates(inplace=True)

print(f"Data shape after removing duplicates: {df.shape}")

# Features and target

# Check if 'Quality' is present and correctly named

if 'Quality' not in df.columns:

raise ValueError("'Quality' column not found in the dataset. Please check the column names.")

X = df.drop('Quality', axis=1)

y = df['Quality']

# Split the data into training and testing sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Initialize RandomForestClassifier with class balancing

clf = RandomForestClassifier(class\_weight='balanced', random\_state=42)

# Train the classifier

clf.fit(X\_train, y\_train)

# Predictions on the test data

y\_pred = clf.predict(X\_test)

# Classification report

print("\nClassification Report:")

print(classification\_report(y\_test, y\_pred, zero\_division=1))

# Confusion matrix

print("\nConfusion Matrix:")

cm = confusion\_matrix(y\_test, y\_pred)

print(cm)

# Confusion matrix visualization

plt.figure(figsize=(6,4))

sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', cbar=False)

plt.ylabel('Actual')

plt.xlabel('Predicted')

plt.title('Confusion Matrix')

plt.show()

# Histograms of features

df.hist(bins=10, figsize=(12,10))

plt.show()

# Feature importance

feature\_importances = clf.feature\_importances\_

features = X.columns

importance\_df = pd.DataFrame({'Feature': features, 'Importance': feature\_importances})

importance\_df = importance\_df.sort\_values(by='Importance', ascending=False)

# Print Feature Importances

print("\nFeature Importances:")

print(importance\_df)

# Plot feature importances

plt.figure(figsize=(10,6))

sns.barplot(x='Feature', y='Importance', data=importance\_df, palette='viridis')

plt.title('Feature Importances')

plt.xticks(rotation=90) # Rotate x-axis labels for readability

plt.xlabel('Feature')

plt.ylabel('Importance')

plt.tight\_layout() # Adjust layout to prevent label clipping

plt.show()

# Cross-validation on the training data

scores = cross\_val\_score(clf, X\_train, y\_train, cv=5, scoring='accuracy')

print(f"\nCross-Validation Scores: {scores}")

print(f"Mean Cross-Validation Score: {scores.mean()}")