CSE422

LAB PROJECT REPORT

WINE QUALITY PREDICTION USING MACHINE LEARNING

Submitted By

Group: 07

Ariyan Hossain, 20101099
Ahanaf Hannan, 20101079
Nowreen Tarannum Rafa, 20101329
Zaki Zawad Mahmood, 20101102

INTRODUCTION

The most popular beverage consumed worldwide is wine, and its values are considered important in society. For customers and producers to increase profits in the current competitive market, wine quality is always crucial. Testing was traditionally performed to assess the quality of wine towards the conclusion of production and to get there, one already invests a lot of time and money. If the quality is poor, numerous procedures must be created from scratch, which is quite expensive. It is difficult to determine a quality based on someone's taste because everyone has their own preferences. As technology advanced, manufacturers began to rely more and more on various equipment for testing during the development process. So that they may save a ton of money and time and have a better understanding of wine quality. A number of initiatives have been made to assess wine quality utilizing the available data since the development of ML methods. One can adjust the variables that directly affect the quality of the wine throughout this procedure. This offers the producer a better understanding of how to adjust various parameters during the development process in order to improve the wine quality. Analysis of the fundamental factors that affect wine quality is therefore crucial. ML may be used as an alternative to find the most crucial factors affecting wine quality. In this study, we have demonstrated how ML may be used to find the optimum parameter that influences wine quality and predict wine quality. 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 and in this paper, we are explaining the steps we followed to build our models for predicting the quality of the wine.

METHODOLOGY

I. Dataset Description

In this study, we use the publicly available wine quality dataset obtained from the UCL Machine Learning Repository, which contains a large collection of datasets that have been widely used by the machine learning community. The datasets are related to both red wine and white variants of the Portuguese "Vinho Verde" wine. These datasets can be viewed as classification or regression tasks. The classes are ordered and not balanced. with 11 physiochemical properties: fixed acidity (g[tartaric acid]/dm3), volatile acidity (g[acetic acid]/dm3), citric acid (g/dm3), residual sugar

(g/dm3), chlorides (g[sodium chloride]/dm3), free sulfur dioxide (mg/dm3), total sulfur dioxide (mg/dm3), density (g/cm3), pH level, sulphates (g[potassium sulphate]/dm3), and alcohol (vol%). The red wine dataset contains 1599 instances and the white wine dataset contains 4898 instances. 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) and labelled as quality in the datasets.

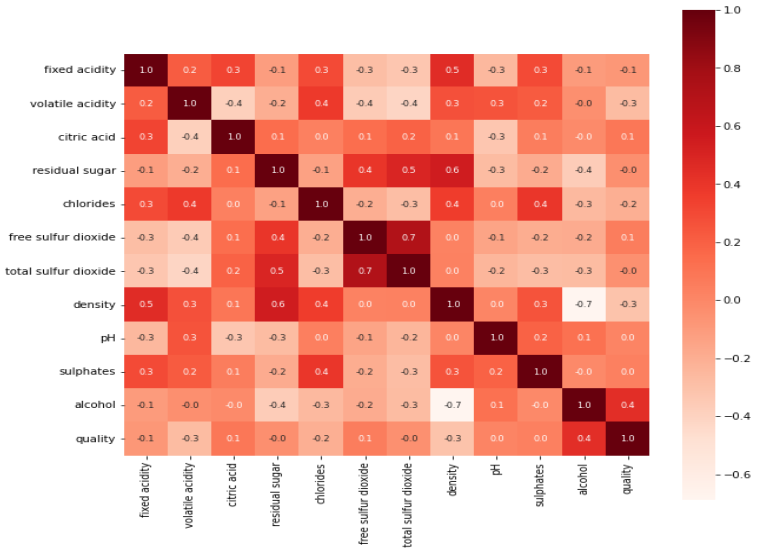
II. Pre-Processing Techniques

Handling Missing Values:

Finding and fixing faulty or incomplete data that are included in the dataset is known as data cleaning. Dealing with the values that are missing from the dataset is one of these processes that is required. Dealing with missing values is a crucial step since many datasets in real life will have many of them. Most of the machine learning models will provide an error if you pass NaN values are passed into it. So to deal with this, we replaced the missing values with the mean values of each features. Initially, in the dataset, the features – 'fixed acidity', 'volatile acidity', 'citric acid', 'residual sugar', 'chlorides', 'pH' and 'sulphates' had missing values.

Feature Selection:

Only a small portion of the dataset's variables may be used to create a machine learning model; the others are either redundant or useless. The overall performance and accuracy of the model may decrease if all these redundant and pointless features are included in the dataset. In order to remove the unnecessary or less significant features from the data, it is crucial to discover and choose the most appropriate features from the data, which is accomplished with the aid of feature selection in machine learning. Feature selection is the method of selection of the best subset of features that will be used for classification. In this study, for a better understanding of the features and to examine the correlation between the features, the Pearson correlation coefficient is calculated for each feature which shows the pairwise person correlation coefficient. The range of the correlation coefficient from -1 to1. Point 1 value implies linear equation is describes the correlation between X and Y strong positive, which is all data points are lying on a line for Y increases as X increases. Point -1 value indicates that strong negative correlations between data points. All data points lie on a line in which Y decreases as X increases and point 0 indicates that there is an absence of correlation between the points.



If two features are correlated, we can predict one from the other. Therefore, if two features are correlated, the model only really needs one of them, as the second one does not add additional information. We considered 0.7 as the threshold for selecting variable that means if the correlation coefficient value of two features are more than 0.7, they will be considered highly correlated and one of the features will be dropped. We found out 'total sulfur dioxide' and 'free sulfur dioxide' are highly correlated and hence we dropped the feature 'total sulfur dioxide'.

Encoding Categorical Features:

In machine learning, datasets sometimes features that have data in words to make the data understandable or in human-readable form. It is challenging to determine how to use these data in the analysis. Many machine learning algorithms can support categorical values without further manipulation but there are many more algorithms that do not. So these data needs to be converted into numeric form so machine learning algorithms can then decide in a better way how those labels must be operated. In the dataset we used, 'type' feature is categorical which represents the type of the wine – red or white. Using Label Encoder, we converted type 'white' to '1' and type 'red' to '0'.

Feature Scaling:

Feature scaling is the final step of data preprocessing in machine learning. It is a technique to standardize the independent variables of the dataset in a specific range. Machine learning models learn a mapping from input variables to an output variable. As such, the scale and distribution of the data drawn from the domain may be different for each variable. Input variables may have different units (e.g. feet, kilometers, and hours) that, in turn, may mean the variables have different scales. Differences in the scales across input variables may increase the difficulty of the problem being modeled. In feature scaling, we put our variables in the same range and in the same scale so that no any variable dominate the other variable. In our study, we used Minmax Scaler to normalize our data. Normalization is a rescaling of the data from the original range so that all values are within the new range of 0 and 1.

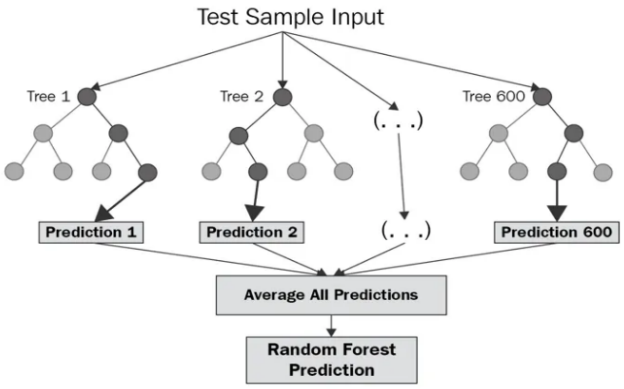
A value is normalized as follows:

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

III. Applied Models

Random Forest Classifier:

Random forest is a supervised machine learning algorithm commonly used in classification and regression problems. It builds decision trees on different samples taken from datasets and takes their majority vote for classification and takes average. One of the most important feature of random forest algorithm is its ability to process datasets containing continuous variables as in regression and datasets containing categorical variables as in classification. Because a random forest combines multiple trees to predict classes in a dataset, it is possible that some decision trees predict the correct output and others do not. But together all the trees predict the correct output. Random forest algorithm takes less training time compared to other algorithms and it also predicts with higher accuracy even for the larger datasets which sometimes miss large proportions of data. This algorithm is capable of performing both classification and regression tasks and it can also handle datasets with higher volume containing high dimensionality. Random forest as the name already suggests, consists of a large number of individual decision trees. Each individual tree in the forest gives a prediction or results and the results with most of the votes becomes the final result of the model itself. In short, random forest algorithm sort of takes the wisdom of the crowd and gets the average best possible voted result and gives it to the machine.

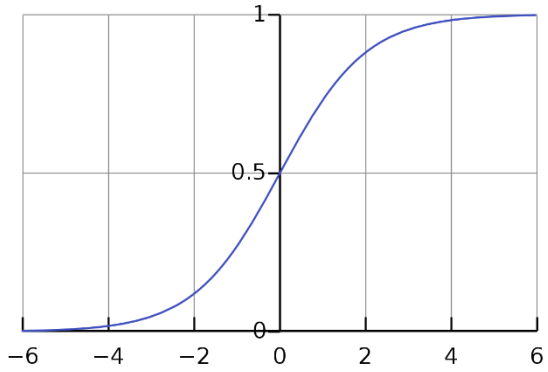


Logistic Regression:

Logistic regression is a supervised learning classification algorithm used to predict the probability of a target variable. Generally logistic regression indicates binary regression which targets binary target variables but there are other types of target variables that can be predicted by it. In binary logistic regression, a dependent variable always has two possible types either 0 or

1 meaning these variable repression either success or failure. In multinomial, a dependent variable may have 3 or more unordered types, meaning they can represent type 1 or 2 or 3. Moving on, the last one is ordinal logistic regression where a dependent variable can have 3 or more ordered types having quantitative significance for example, good, bad, excellent etc. Logistic regression is very much similar to linear regression except how they are implemented or used in real life. Linear regression is mainly used for regression problems whereas logistic regression is used for classification problems. Logistic regression model has a couple of advantages over other algorithms. It is easier to implement and very efficient to train, and it also makes no assumptions about the distribution of classes in future space, it can easily extend to multiple classes and it not only shows how accurate and predictor really is but also shows the direction of prediction.

$$P(Y|X) = \frac{e^{a+bX}}{1 + e^{a+bX}}$$



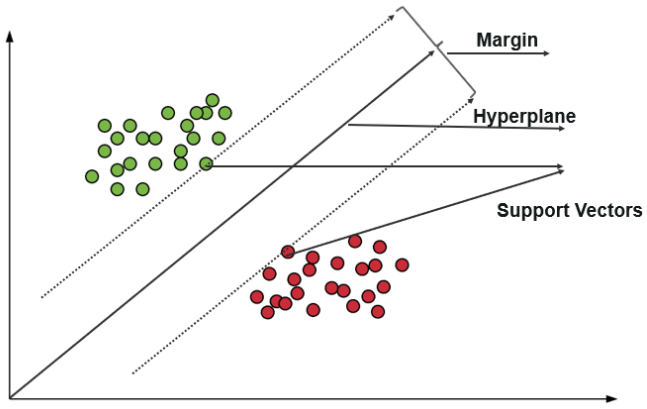
Naive Bayes Classifier:

Naive Bayes algorithm is a supervised learning algorithm which takes the concept of Bayes Theorem and uses it to solve classification problems. It is one of the simplest yet most effective classification algorithms that helps build fast machine learning models that can predict fast predictions. The fundamental Naive Bayes assumption is that each feature makes an independent and equal contribution to the outcome. The assumptions made by Naive Bayes are not generally correct in real-world situations. In-fact, the independence assumption is never correct but often works well in practice. Advantages of Naive Bayes classifier includes the following: fast and easy ML algorithms to predict class of datasets, can be used for binary or multi class classification, performs well in multiclass classification compared to other algorithms, most popular for text classification problems. There are three types of Naive Bayes model. In our work, we used Gaussian, which follows a normal distribution meaning it predicts continuous values instead of discrete. The likelihood of the features is assumed to be Gaussian, hence, conditional probability is given by:

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} exp\left(-\frac{(x_i-\mu_y)^2}{2\sigma_y^2}\right)$$

Support Vector Classifier:

SVM also known as support vector machines are supervised machine learning algorithms used for both classification as well as regressions models. It is primarily suited for classification problems in machine learning. The main objective of the SVM algorithm is to find a hyperplane or a best fit line in a n-dimensional space that uniquely classifies the data points. SVM chooses the extreme vectors or data points that help create a hyperplane, these extreme cases are otherwise known as support vectors and thus the algorithm itself is known as a support vector machine. SVM is divided into two types, linear SVM which is used for linearly separable data, which means that the datasets can be classified into two classes by using a straight line. On the other hand, Non-Linear SVM meaning it is used for non-linearly separable data that is the dataset cannot be divided into two parts by using a straight line. SVM is very much effective in high dimensional cases. It is memory efficient as it uses support vectors and also different kernel functions can be specified for the decision function.



RESULTS

Accuracy:

The proportion of accurately predicted observations to all observations is known as the accuracy ratio. Accuracy can be determined by dividing the total number of predictions by the number of right predictions. We can calculate Accuracy as,

$$Accuracy = \frac{True\ Positive + True\ Negative}{True\ Positive + False\ Positive + False\ Negative + True\ Negative}$$

Precision:

The proportion of accurately predicted positive observations to all expected positive observations is referred to as precision. We can calculate Precision as,

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Recall:

The proportion of accurately predicted positive observations to all of the actual class observations is known as recall. We can calculate Recall as,

$$Recall = \frac{True\ Postive}{True\ Positive + False\ Negative}$$

F1 Score:

The F1 score is a weighted harmonic mean of precision and recall such that the best score is 1.0 and the worst is 0.0. F1 scores are lower than accuracy measures as they embed precision and recall into their computation. As a rule of thumb, the weighted average of F1 should be used to compare classifier models, not global accuracy. We can calculate F1 Score as,

$$F1\ Score = 2 * \frac{Recall * Precision}{Recall + Precision}$$

Support:

Support is the number of actual occurrences of the class in the specified dataset. Support points to structural flaws in the classifier's reported scores. As the requirement for stratified sampling or rebalancing can be indicated by unbalanced Support in the training data. Support remains constant across models.

Result from Random Forest Classifier:

Random Forest classifier had an accuracy of 88%

	Precision	*Recall*	*F1 Score*	*Support*
0	0.90	0.96	0.93	1047
1	0.76	0.58	0.66	253

Result from Logistic Regression:

Logistic Regression classifier had an accuracy of 82%

	Precision	*Recall*	*F1 Score*	*Support*
0	0.84	0.96	0.90	1047
1	0.63	0.25	0.35	253

Result from Naive Bayes:

Naive Bayes classifier had an accuracy of 73%

	Precision	*Recall*	*F1 Score*	*Support*
0	0.90	0.75	0.82	1047
1	0.39	0.66	0.49	253

Result from Support Vector Classifier:

Support Vector classifier had an accuracy of 82%

	Precision	*Recall*	*F1 Score*	*Support*
0	0.83	0.98	0.90	1047
1	0.66	0.18	0.28	253

When we observe all the results we can see that using Random Forest, we were able to obtain a maximum accuracy of 88%. This had the highest accuracy amongst all the other classifiers. The accuracy of Logistic Regression was 82%. Support Vector Classifier also gave an accuracy of 82%. Lastly, Naïve Bayes had the least accuracy, as it had only 73% accuracy.

REFERENCES

1. https://www.kaggle.com/datasets/rajyellow46/wine-quality
2. https://www.geeksforgeeks.org/wine-quality-prediction-machine-learning/
3. https://www.javatpoint.com/data-preprocessing-machine-learning
4. https://www.scirp.org/journal/paperinformation.aspx?paperid=107796
5. https://www.diva-portal.org/smash/get/diva2:1574730/FULLTEXT01.pdf
6. https://www.javatpoint.com/machine-learning-random-forest-algorithm
7. https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm
8. https://www.javatpoint.com/logistic-regression-in-machine-learning
9. https://www.geeksforgeeks.org/naive-bayes-classifiers/
10. https://www.analyticsvidhya.com/blog/2021/04/wine-quality-prediction-using-machine-learning/