WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

A Study on Red Wine Quality Detection Using Machine Learning

Mrs. Ramya B N[1], Adithi Shankar[2], Kruthika R[3]

[2](1JT20AI003), [3](1JT20AI018)
[1]Department of AIML, Jyothy Institute of Technology
[1]ramyabn@jyothyit.ac.in, [2]1jt20ai003@jyothyit.ac.in, [3]1jt20ai018@jyothyit.ac.in

ABSTRACT –

As a branch of artificial intelligence (AI), machine learning (ML) aims to understand data structures and incorporate them into models for later use on unknown data in order to achieve particular goals. It is widely used in many different fields, including astronomy, business, medicine, and other sciences. Inspired by ML's successes in these domains, we take advantage of its potential to predict wine quality according to many factors. We evaluate the prediction capabilities of a multi-layer Artificial Neural Network (ANN), Ridge Regression (RR), Support Vector Machine (SVM), and Gradient Boosting Regressor (GBR) in this study. We examine a variety of factors that affect wine quality. As can be seen from the MSE, R, and MAPE values of 0.3741, 0.6057, and 0.0873, respectively, GBR performs better than the other models. This study demonstrates the ability of statistical analysis to reveal

INTRODUCTION

Wine quality holds significant importance for both consumers and manufacturing industries. Industries strategically boost their sales through product quality certifications, a practice particularly crucial in a global market where wine is a commonly consumed beverage. Certifying product quality has become integral for enhancing market value. Traditionally, assessing product quality occurred post-production, a time-consuming and resource-intensive process involving various human experts. The subjectivity inherent in human opinions made this method both challenging and expensive.

Recognizing that not all features contribute equally to accurate predictions of wine quality, our research focuses on identifying the essential wine features that yield promising results. We employ machine learning classification algorithms such as Support Vector Machine (SVM), Naïve Bayes (NB), and Artificial Neural Network (ANN) using the publicly available wine quality dataset from the UCI machine learning repository (Cortez et al., 2009). This dataset comprises red and white wine variants of the Portuguese "Vinho Verde" wine, featuring 1599 instances for red wine and 4898 instances for white wine. Both datasets consist of 11 input features based on physicochemical tests and one output feature scaling sensory data into 11 quality classes from 0 to 10 (0-very bad to 10-very good).

A crucial step in data preprocessing is feature selection, a widely adopted practice (Wolf and Shashua, 2005). This process involves choosing a subset of relevant features based on their weighted relevance. Features with lower weights are eliminated, simplifying the model, reducing training time, and enhancing overall performance (Panday et al., 2018). Our study emphasizes the importance of feature selection, and to assess our model's effectiveness, we rely on key indicators such as accuracy, precision, recall, and F1 score.

I. LITERARURE SURVEY

In their study, Kumar et al. (2020) employed random forest, support vector machine, and naive Bayes techniques to predict red wine quality based on various attributes. They evaluated the performance using metrics such as precision, recall, f1-score, accuracy, specificity, and misclassification error. Among these techniques, the support vector machine yielded the highest accuracy at 67.25%, outperforming random forest and naive Bayes.

Gupta (2018) focused on determining important features for red and white wine quality using machine learning algorithms like linear regression, neural network, and support vector machine. They assessed wine quality in two ways: evaluating the target variable's dependency on independent variables and predicting the target variable value. Their conclusion emphasized that not all features are necessary for prediction; instead, only essential features should be selected.

Dahal et al. (2021) predicted wine quality by employing various machine learning models, including rigid regression, support vector machine, gradient boosting regressor, and multi-layer artificial neural network. Through performance analysis, they identified the gradient boosting regressor as the best model, surpassing others with MSE, R, and MAPE values of 0.3741, 0.6057, and 0.0873, respectively.

Er and Atasoy (2016) proposed a method to classify red and white wine quality using k-nearest-neighborhood, random forest, and support vector machine algorithms. They utilized principal component analysis for feature selection, achieving the best results with the random forest algorithm.

Lee et al. (2015) introduced a decision tree-based method to predict wine quality and compared it with support vector machine, multi-layer perceptron, and BayesNet algorithms. Their study revealed that their proposed method outperformed other approaches.

P. Appalasamy et al. (2012) predicted wine quality based on physiochemical data, utilizing decision tree and naive Bayes algorithms on both red and white wine datasets. Their comparative analysis concluded that a classification approach could enhance wine quality during production.

II.METHODOLOGY

Under this section, the methods used for the execution of the study and implementation of the algorithms have been discussed. The diagram below shows the flowchart of the methodology.

1. Data Description

The UCI machine learning repository, which has a sizable collection of datasets utilized by the machine learning community, is where the red and white wine datasets used in this paper were found. Two Excel files pertaining to red and white wine variations of the Portuguese "Vinho Verde" wine are included in the dataset (Cortez et al., 2009). There are 1599 examples in the red wine dataset and 4898 cases in the white wine dataset. Eleven input variables—fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulfates, and alcohol—are present in both datasets (based on physicochemical testing), and one output variable—quality—is based on sensory data. Eleven quality classifications of sensory data are rated from 0.

Table1.Attribute description

Attributes	Description
fixed acidity	Fixed acids, numeric from 3.8 to 15.9
volatile acidity	Volatile acids, numeric from 0.1 to 1.6
citric acid	Citric acids, numeric from 0.0 to 1.7
residual sugar	residual sugar, numeric from 0.6 to 65.8
chlorides	Chloride, numeric from 0.01 to 0.61
free sulfur dioxide	Free sulfur dioxide, numeric: from 1 to 289
total sulfur dioxide	Total sulfur dioxide, numeric: from 6 to 440
density	Density, numeric: from 0.987 to 1.039
pH	pH, numeric: from 2.7 to 4.0
sulfates	Sulfates, numeric: from 0.2 to 2.0
alcohol	Alcohol, numeric: from 8.0 to 14.9
quality	Quality, numeric: from 0 to 10, the output target

2. Feature Selection

The process of choosing the optimal subset of features to be utilized for classification is known as feature selection (Fauzi et al., 2017). The majority of feature selection methods consist of two parts: a filter and a wrapper. The filter uses the public features separately from the learning algorithm, while the wrapper assesses the features and selects attributes based on an accuracy estimate using a particular learning model and search algorithm (Onan and Korukoğlu, 2017). This study looks at the correlation between the qualities in order to better comprehend them. Table 1 displays the pairwise person correlation coefficient P, which is determined using the formula below, for each attribute for which the Pearson correlation coefficient has been calculated (Dastmard, 2013).

$$\mathrm{Px,y} = \frac{\mathrm{cov}\,(\mathrm{X,Y})}{\sigma\mathrm{X},\sigma\mathrm{Y}}$$

Where cov is the covariance and σ is the standard deviation of the features X and Y. The correlation coefficient's range is -1 to 1. The linear equation implied by point 1 indicates a significant positive correlation between X and Y, meaning that all data points are on a line where Y grows as X increases. When there are significant negative correlations between data points, the point -1 value is indicated. Every data point is located on a line where Y falls off as X rises. Furthermore, point 0 denotes a lack of association between the points (Dastmard, 2013).

3. Hyper Parameter tuning

Grid search serves as a fundamental technique for hyperparameter tuning, involving a comprehensive exploration of the user-specified hyperparameter set. This method proves effective for scenarios with multiple hyperparameters and a confined search space. The algorithm is straightforward and, with ample resources, can yield the most accurate predictions, enabling users to identify the optimal combination (Joseph, 2018). Parallel execution of grid search is straightforward, as each test operates independently of the time series, ensuring that one experiment's results do not influence others. This allows flexible allocation of computing resources. Furthermore, grid search is adaptable to a limited sampling range, as an excessive number of settings is not conducive. In practical applications, grid search is typically favored when users possess adequate knowledge of the hyperparameters, enabling the definition of a narrow search space. It is particularly suitable when adjusting no more than three hyperparameters simultaneously. Despite the existence of alternative search algorithms with potentially more features, grid search remains the most widely utilized method due to its mathematical simplicity (Yu and Zhu, 2020).

4. Evaluation

The evaluation of performance involves calculating and assessing techniques to gauge the effectiveness and efficiency of the model. Four criteria determine the accuracy of predictions:

- True Positive: The count of samples predicted as positive that are genuinely positive.

- False Positive: The count of samples predicted as positive that are actually negative.

- False Negative: The count of samples predicted as negative that are truly positive.

- True Negative: The count of samples predicted as negative that are genuinely negative.

We employ the methods indicated below to evaluate the model.

1. Accuracy - The ratio of accurately predicted observations to total observations is known as accuracy. By dividing the total number of guesses by the number of right predictions, one can readily calculate the accuracy.

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

2. Precision - Precision, denoted by Precision, is a performance metric in machine learning that measures the accuracy of the positive predictions made by a classification model. It is calculated as the ratio of true positive predictions to the sum of true positive and false positive predictions.

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

3. Recall - Recall, also known as Sensitivity or True Positive Rate, is a performance metric in machine learning that measures the ability of a classification model to correctly identify all relevant instances from the total actual positive instances. It is particularly useful in scenarios where the emphasis is on minimizing false negatives. Recall is calculated as the ratio of true positive predictions to the sum of true positives and false negatives. The formula for recall is expressed as:

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

4. F1-score - The F1 score is a metric in machine learning that combines both precision and recall into a single value, providing a balanced assessment of a classification model's performance. It is particularly useful when there is an uneven class distribution or when false positives and false negatives carry different levels of importance. The F1 score is calculated using the harmonic mean of precision and recall. The formula for F1 score is expressed as:

$$F1 - score = 2 * \frac{\text{Recall} * \text{Precision}}{\text{Recall} * \text{Precision}}$$

For the majority of conventional applications, accuracy is the most often utilized evaluation metric. However, many experts have noticed that for very skewed class distributions, the recall rate for minority classes is often 0, which means that no classification rules are formed for the minority class. As a result, the accuracy rate is not appropriate for analysing imbalanced data sets. The minority categories' precision and recall are significantly worse than those of the majority class, to use terms from information retrieval. It is difficult for the classifier to perform well in the minority class because accuracy assigns greater weight to the majority class than to the minority class.More measures are becoming widely used for this purpose (Guo et al., 2008).According to Estabrooks and Japkowicz (2001), the F1 score is a widely used assessment matrix for the imbalanced class problem. The F1 score is the product of two matrices: recall and precision. Recall indicates the opposite of the proportion of incorrectly classified cases, whereas precision indicates how well the model predicted a certain class. Considering that different classes have different F1 scores. By calculating our final score using

the F1 scores' unweighted mean. Our goal is for our models to become optimal in order to classify cases that fall into the minority, such wine quality ratings of 3, 8, or 9, in an equally good manner as the other attributes that are represented in higher numbers.

III. ARCHITECTURE

1.Unbalanced Data:

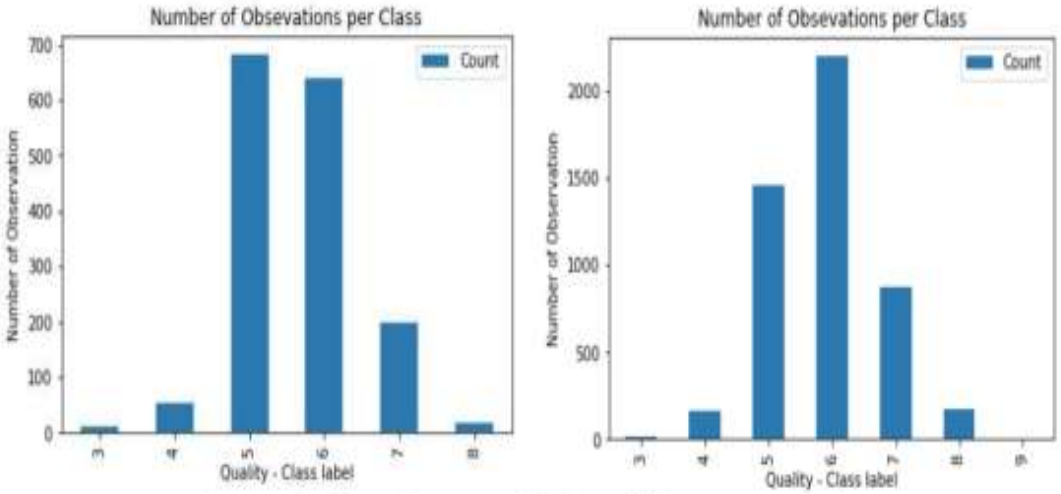

Figure 3: Distribution of Red & White wine quality

The disparate classes' unequal representation in the datasets is the unbalanced distribution of red and white wine. Alphabets that are overfit or underfit may result from these unbalanced data.

Class 5 red wines had 681 examples of the greatest quality, while class 6 white wines have 2198 instances. The number of instances in each of the two datasets is out of balance; for red wine, it ranges from 5 in the minority class to 681, and for the majority class, it is 1297. Rarely do the middle classes and the highest quality ratings correspond. It is possible to remedy this issue by resampling; this can be done either by over-sampling, which involves adding duplicate cases from the underrepresented class, or by utilizing other methods.

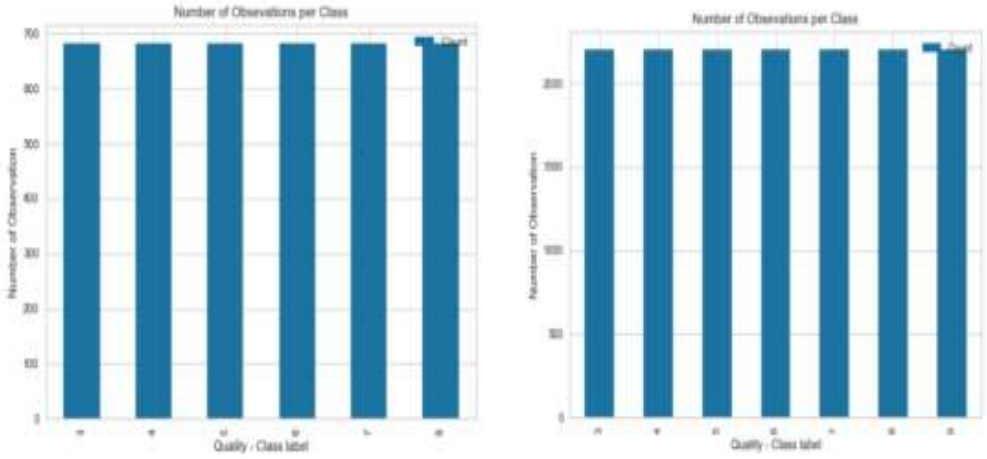

Figure 4: Effect of balancing dataset

2. Feature Selection

Figure 5: correlation matrices red wine

The features of Figure 5 red wine correlation matrix, such as "alcohol," "volatile acidity," "sulphates," "citric acid," "total sulfur dioxide," "density," "chlorides," "fixed acidity," "pH," "free sulfur dioxide," and "residual sugar," were ranked based on the high correlation values to the quality class.

Similarly, from Figure 6's white wine correlation matrix, we ranked the characteristics—such as those related to alcohol, density, chlorides, volatile acidity, total sulfur dioxide, fixed acidity, pH, residual sugar, sulphates, citric acid, and free sulfur dioxide—according to the high correlation values to the quality class.

Figure 6: correlation matrices white wine

3. Data Standardization

The most recent machine learning algorithm for both supervised and unsupervised problems is integrated into the Python module Scikit-learn (Pedregosa et al., 2011).

By applying the data standardization technique to every numeric feature and then dividing the data by standard derivation, it can scale the features between 0 and 1. This will be helpful for learning the model (Pedregosa et al., 2011). Thus, we standardize the data using this method.

$$Zi = xi - u\ \sigma$$

is the standard derivation, xi is each value, and u is the array x's mean value. This is the formula for standardization.

4. Data Separation

A training and testing set of data is being created using the scikit-learn library. In order to split the dataset, we set the test size to 0.2. In order to prevent the sample data from being divided in an invisible way, the train test split method randomly divides the data into the testing set and the training set.

5. Hyper Parameter tuning

Table 2: Hyperparameter tuning for SVM Model

Parameter	Observations	Red wine outcome	White wine outcome
C	0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, 2, 3, 4, 5, 10	3	5
kernel	'linear, 'rbf', 'sigmoid'	'rbf'	'rbf'
gamma	0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, 2, 3, 4, 5, 10	0.4	2.2

Table 3: Hyperparameter tuning for ANN Model

Parameter	Observations	Red wine outcome	White wine outcome
hiden_layer_sizes	[100, 50], [200, 100], [300, 200], [400, 200]	[200, 100]	[400, 200]
activation	'tanh', 'relu', 'logistic'	'tanh'	'tanh'
solver	'lbfgs', 'adam', 'sgd'	'adam'	'adam'
Max_iter	200, 300, 400, 500, 700, 1000	300	400
random_state	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10	4	6
learning_rate_init	0.001, 0.002, 0.003, …., 0.01	0.006	0.006

6. Model and Evaluation

We implemented the model using machine learning techniques, including artificial neural networks (ANN), naïve Bayes (NB), and support vector machines (SVM). We use the scikit-learn Python machine learning libraries to access Adobe algorithms (scikit-learn, 2021).

Every application of the computed classification algorithm produced the evaluation results. as stated in the subsection on evaluation.

IV. RESULT

1. Feature Selection result

Results were obtained by using a Pearson correlation coefficient technique to assess each feature's performance. Over red wine White wine and Figure 5 Figure 6 illustrates the significance of each feature and ranks them based on their strong correlation with quality.

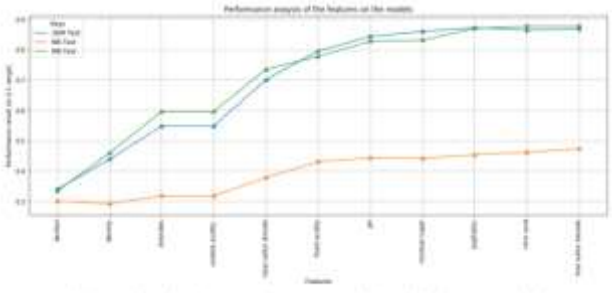

Figure 8: white wine performance analysis of the feature model

As a result, the study of feature groups is carried out from left to right, as seen in Figures 7 and 8 from both datasets. Of these, the first 21 features are chosen, 10 are excluded, and the final feature is not included since it does not improve the model's performance.

'Residual sugar' and 'free sulfur dioxide' features from the white and red wine datasets are not included in the final model implementation. The aforementioned Figures 7 and 8 from the red and white wine performance analyses provide evidence that the prediction models performed better when using the 10 features they chose.

2. Model Results

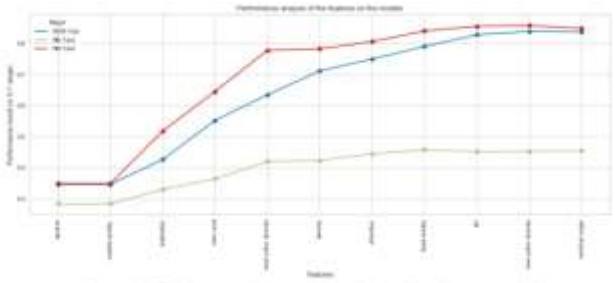

Figure 7: Red wine performance analysis of the feature model

The features' relative importance is determined, and the top 10 features from each dataset were chosen, with the final feature being eliminated, as shown in the performance analysis of red wine. White wine performance analysis and Figure 7 The accuracy performance is displayed in Figure 8.

First, the unbalanced classes were subjected to the implementation of these chosen features. Figure 3 depicts the unbalanced classes, and Tables 4 and 5 express the prediction model's performance in terms of accuracy, precision, recall, and F1 score.

Table 4: 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

Table 6: 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	

Following the implementation of these chosen features on the balanced class, Tables 6 and 7 of red and white wine illustrate how each class's balancing and the prediction model's performance are evaluated in terms of accuracy, precision, recall, and f1 score (Figure 4).

3. Discussion

"What wine features are important to get the promising result?" is the research question that guides our study. We use the Pearson coefficient correlation matrices to implement the answer to this question and determine the relationship among all the characteristics, as shown in Figure 5's red wine correlation matrices and the correlation matrices for white wine (Figure 6). The features were then ranked according to how well they correlated with the quality feature. The implementation of the analysis of feature groups from left to right is demonstrated the first ten features from each dataset in Figures 7 and 8 are chosen, and the final feature is not included since there enhancement and it is lowering the model's performance. 'Residual sugar' and 'free sulfur dioxide' features from the white and red wine datasets are not included in the final model implementation.

We begin putting the model into practice after determining how important the features are. First, we used the original data to implement the model and assess its performance (unbalanced class), as seen in Figure 3, and subsequently put into practice the Figure 4 depicts a model of the balance class that balances each class.

Table 4 red wine and Table 5 white wine performance analysis results for the prediction model's accuracy, precision, recall, and f1 score analyses show the results of the analysis. Each model's imbalanced classes are looked at, and Table 6 red wine and Table 7 presents the balanced white wine performance analysis results classes for every model are looked at.

We were able to improve all of the models' performance results on the balanced class by using these unbalancing and balancing classes.

Out of the three algorithms, the artificial neural network (ANN) algorithm outperformed the support vector machine (SVM) and naïve Bayes (NB) algorithms in both the red and white wine datasets.

Although they are not directly related to this project, the other related works were mentioned in section 2.2.

The paper by Kumar (2020) bears similarities in that it employs comparable performance metrics and machine learning algorithms, including support vector machines and naïve Bayes. The distinction is that all features were employed for the model's prediction, and the model was trained on uneven classes. Er and Atasoy (2016) achieved the best accuracy result from the random forest on 69.90% in the red wine and 71.23% in the white wine datasets, and they used the principal components analysis technique for feature selection. In terms of performance analysis, they achieved the best of 67.25% accuracy from the support vector machine on the red wine dataset. Gupta, (2018) It has been suggested that, rather than choosing just the features that are required to predict wine quality, all features may not be necessary for the prediction. They did this by using linear regression to ascertain the target variable's dependencies. In contrast, our model's support vector machine accuracy was 67.83% in the white wine dataset and 69.06% in the red wine dataset.

Following the model's training on balanced data and the selection of the optimal hyperparameters, the model's performance improved to achieve 83.52% accuracy in the red wine and 86.86% accuracy in the white wine. Furthermore, by using the Pearson coefficient correlation matrices for feature selection, our model outperformed the artificial neural network model with the best accuracy of 85.16% in the red wine and 88.28% in the white wine.

V.CONCLUSION

This report utilizes two datasets, red and white, derived from Portuguese "Vinho Verde" wine to predict wine quality based on physicochemical properties. Initially, we employed oversampling during data preprocessing to balance the dataset, optimizing model performance. Subsequently, we identified features for enhanced prediction results by using Pearson coefficient correlation matrices, ranking features based on high correlations. The performance of the model improved after applying balanced datasets, highlighting the general enhancement gained from eliminating irrelevant features.

In summary, the removal of irrelevant dataset features significantly boosted the performance of the classification model. The challenge of minority class representation in a dataset was addressed by employing oversampling and under sampling techniques to achieve balance across class representations.

The accuracy of the Support Vector Machine (SVM) algorithm reached 83.52% for red wine and 86.86% for white wine. The Naïve Bayes (NB) algorithm achieved accuracy rates of 46.33% for red wine and 46.68% for white wine. Meanwhile, the Artificial Neural Network (ANN) yielded accuracy rates of 85.16% for red wine and an impressive 88.28% for white wine. Notably, the best accuracy results were obtained from the Artificial Neural Network (ANN) among the three machine learning algorithms for both red and white wine datasets.

Hence, in classification algorithms, the careful selection of features and the implementation of data balancing techniques prove pivotal in improving model performance.

VI. ACKNOWLEDGEMENT

I would like to express my appreciation to my mentors, colleagues, and peers who have provided guidance, insights, and support throughout the exploration of this topic. Their feedback, discussions, and collaborations have been invaluable in shaping my understanding and enhancing the quality of the work.

The progress made was a collective effort, and it is through the contributions and collaboration of numerous individuals and institutions that we have been able to deepen our understanding and achieve advancements in this field.

VII. REFERENCES

[1] Chawla, N.V. (2005). Data Mining for Imbalanced Datasets: An Overview. In Maimon, O., Rokach, L. (Eds.), Data Mining and Knowledge Discovery Handbook. Springer US, Boston, MA, pp. 853–867. https://doi.org/10.1007/0-387-25465-X_40

[2] Cortez, P., Cerdeira, A., Almeida, F., Matos, T., Reis, J. (2009). Modeling wine preferences by data mining from physicochemical properties. Decis. Support Syst. 47, 547–553. https://doi.org/10.1016/j.dss.2009.05.016

[3] 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

[4] Er, Y., Atasoy, A. (2016). The Classification of White Wine and Red Wine According to Their Physicochemical Qualities. Int. J. Intell. Syst. Appl. Eng. 4, 23–26. https://doi.org/10.18201/ijisae.265954

[5] Estabrooks, A., Japkowicz, N. (2001). A Mixture-of-Experts Framework for Learning from Imbalanced Data Sets. In: Hoffmann, F., Hand, D.J., Adams, N., Fisher, D., Guimaraes, G. (Eds.), Advances in Intelligent Data Analysis, Lecture Notes in Computer Science. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 34–43. https://doi.org/10.1007/3-540-44816-0_4

[6] Fauzi, M., Arifin, A.Z., Gosaria, S., Prabowo, I.S. (2017). Indonesian News Classification Using Naïve Bayes and Two-Phase Feature Selection Model. Indones. J. Electr. Eng. Comput. Sci. 8, 610–615. https://doi.org/10.11591/ijeecs.v8.i3.pp610-615

[7] Gandhi, R. (2018). Support Vector Machine — Introduction to Machine Learning Algorithms. Medium. URL https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47 (accessed 6.6.21).

[8] Guo, X., Yin, Y., Dong, C., Yang, G., Zhou, G. (2008). On the Class Imbalance Problem. In: 2008 Fourth International Conference on Natural Computation. https://doi.org/10.1109/ICNC.2008.871

[9] 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

[10] Hewahi, N.M., Abu Hamra E. (2017). A Hybrid Approach Based on Genetic Algorithm and Particle Swarm Optimization to Improve Neural Network Classification. J. Inf. Technol. Res. JITR 10, 48–68. https://doi.org/10.4018/JITR.2017070104

[11] Joseph, R. (2018). Grid Search for model tuning. Medium. URL https://towardsdatascience.com/grid-search-for-model-tuning-3319b259367e (accessed 6.6.21).

[12] Kumar, S., Agrawal, K., Mandan, N. (2020). Red Wine Quality Prediction Using Machine Learning Techniques. In: 2020 International Conference on Computer Communication and Informatics (ICCCI). https://doi.org/10.1109/ICCCI48352.2020.9104095

[13] Lee, S., Park, J., Kang, K. (2015). Assessing wine quality using a decision tree. In: 2015 IEEE International Symposium on Systems Engineering (ISSE). https://doi.org/10.1109/SysEng.2015.7302752

[14] Mhatre, M.S., Siddiqui, D.F., Dongre, M., Thakur, P. (2015). Review paper on Artificial Neural Network: A Prediction Technique 6, 3.

[15] Onan, A., Korukoğlu, S. (2017). A feature selection model based on genetic rank aggregation for text sentiment classification. J. Inf. Sci. 43, 25–38. https://doi.org/10.1177/0165551515613226

[16] P. Appalasamy, N.D. Rizal, F. Johari, A.F. Mansor. (2012). Classification-based Data Mining Approach for Quality Control in Wine Production. https://doi.org/10.3923/jas.2012.598.601

[17] Panday, D., Cordeiro de Amorim, R., Lane, P. (2018). Feature weighting as a tool for unsupervised feature selection. Inf. Process. Lett. 129, 44–52. https://doi.org/10.1016/j.ipl.2017.09.005

[18] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D. (2011). Scikit-learn: Machine Learning in Python. Mach. Learn. PYTHON 6.

[19] Rish, I. (2001). An Empirical Study of the Naïve Bayes Classifier. IJCAI 2001 Work Empir Methods Artif Intell 3.

[20] Shafei, D. (2020). Artificial Neural Network - Applications, Algorithms and Examples. TechVidvan. URL https://techvidvan.com/tutorials/artificial-neural-network/ (accessed 6.6.21).

[21] scikit-learn, developer (2021). scikit-learn: machine learning in Python — scikit-learn 0.24.2 documentation. URL https://scikit-learn.org/stable/ (accessed 5.31.21).

[22] Wolf, L., Shashua, A. (2005). Feature Selection for Unsupervised and Supervised Inference: The Emergence of Sparsity in a Weight-Based Approach 33.

[23] Yu, T., Zhu, H. (2020). Hyper-Parameter Optimization: A Review of Algorithms and Applications. ArXiv200305689 Cs Stat.

[24] Zhang, P. (2000). Neural Networks for Classification: A Survey. Syst. Man Cybern. Part C Appl. Rev. IEEE Trans. On 30, 451–462. https://doi.org/10.1109/5326.897072