Wine Quality Prediction

Onur Alaçam, *Department of Computer Science, Ozyegin University*
Uygar Kaya, *Department of Computer Science, Ozyegin University*
Tuna Tuncer, *Department of Computer Science, Ozyegin University*

***Abstract* — The many qualities produced by the various stages of the winemaking process have a significant effect on the development of wine culture. As a result, it is critical to assess the wine's quality before to consumption. To come up with an efficient classification technique, machine learning methods utilizing physicochemical variables such as pH, density, citric acid, and other factors to assess the wine's quality. In this article, we analyze classification and regression of wine quality using Linear Regression, Logistic Regression, K-Nearest Neighbor (KNN), and Multilayer Perceptron (MLP) in the WineQuality dataset. We also compare the findings of this study to those of other studies in the literature and give a brief discussion of comparative benefits.**

***Index Terms* — Classification, Regression, K-Nearest Neighbor (KNN), Linear Regression, Logistic Regression, Multilayer Perceptron (MLP), Single Layer Perceptron (SLP), Wine Quality.**

I. Introduction

WINE is the most widely consumed beverage on the world, and its cultural aspects are highly valued. In today's competitive market, the quality of wine is always vital for customers and, more importantly, for producers to increase income. Traditionally, wine quality was evaluated by testing at the conclusion of the manufacturing process; to get to that point, one had already invested a significant amount of time and money. If the quality is poor, numerous procedures must be done from the start, which is quite expensive. Because everyone has their unique taste preferences, determining a quality based on a person's preferences is difficult. Manufacturers began to rely on various equipment for testing throughout development phases as technology advanced. As a result, they will have a better understanding of wine quality, which will save them both money and time. Furthermore, this aided in the collection of a large amount of data on numerous aspects such as the quantity of various chemicals used during production, the pH utilized, and the quality of the wine produced. With the rise of ML techniques and their success in the past decade, there have been various efforts in determining wine quality by using the available data [1] [2]. During this procedure, the factors that directly affect the wine quality may be fine-tuned. This allows the maker to fine-tune the wine's quality by adjusting various aspects during the development process. Furthermore, this might result in wines with a variety of flavors, as well as a new brand. As a result, it's critical to examine the basic characteristics that define wine quality. ML may be used to discover the most critical elements that govern wine quality in addition to humanitarian initiatives. On the WineQuality dataset, this article tested classification and regression using four different algorithms: Linear Regression, Logistic Regression, Multilayer Perceptron, and K-Nearest Neighbor, and attempted to predict the quality from thirteen common wines quality. While analyzing the classification and regression methods in our study, we obtained the most successful accuracy rate in K-Nearest Neighbor - (KNN) for classification. Also, we get the most successful accuracy rate in Multilayer Perceptron.

II. Related Work

Today, a growing number of customers are interested in wine. In order to support this growth, the wine industry is looking into new advances in both winemaking and offering structures [1]. Wine confirmation is evaluated using physicochemical and tactile tests [2]. The complexity and heterogeneity of its headspace indicate that wine segregation is not a simple procedure. The placement of wines is significant for a variety of reasons. These reasons include the financial estimation of wine items, the security and assurance of wine quality, the prevention of wine corruption, and the control of refreshment preparation [3]. Wine quality has been planned using data mining innovations. Machine learning techniques, such as various applications, are used to create models from data in order to predict wine quality. In 1991, a "Wine" informational index was given to the UCI store to order three cultivars from Italy [4]. The index contains 178 occurrences with estimations of 13 distinct synthetic constituents, such as alcohol and magnesium. This data has been widely used as a benchmark for new information mining classifiers because it is extremely easy to separate. Principal Component Analysis (PCA) was performed and published for wine characterization as indicated by geological area [5]. The information they used in their study included 33 Greek wines with physicochemical factors. Another study of wine classification used physicochemical data. This information is linked to wine odor chromatograms calculated with a Fast GC Analyzer [6]. Three portrayal methods, such as Naive Bayes, Random Forest, and Support Vector Machines (SVM), are contrasted agreeing and their exhibition in a two-organized architecture in the last investigation. A couple of applications of data mining frameworks to wine quality assessment have been proposed. A taste desire framework was proposed by Cortez et al. [1]. A Support Vector Machine, Naive Bayes, and a Random Forest

were used to engineer wine analysis in their taste expectation framework. Shanmuganathan's method involved predicting the effects of season and climate on wine yields and quality [7]. Chen et al. [8] presented a Wine informatics framework that depicted the flavor and characteristics of wine based on typical language audits. They used progressive clustering and association rules. The authors of a research paper [9] compared different machine learning algorithms such as Naive Bayes, Decision Trees, and Support Vector Machines on cardiotocography data to determine which one was the best.

III. Methods

A. The Dataset

Wine Quality Dataset [10] consists of 6498 rows and 12 columns, and these columns are fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total dioxide, density, pH, sulphates, alcohol, and quality. The quality column is made up of integer values ranging from 0 to 10. Even though the dataset contains 1168 duplicate data, we opted to maintain them. We reasoned that they may be separate wines if they shared all 11 characteristics. After that, we used one-hot encoding on the column type to create two new columns called type red and type white. We decided to solve our problem in 2 different ways by turning it into both regression and classification problems. Afterwards, we sorted all the columns in the dataset according to their correlations according to quality and created 4 different datasets with this information. While we included all columns in our first dataset, we included the 6 columns with the highest correlation (alcohol, density, volatile acidity, chlorides, type red, and type white) in our second dataset. For our third dataset, we selected the columns with the lowest correlation values (fixed acidity, free sulfur dioxide, total sulfur dioxide, sulphates, residual sugar, and pH), and for the 4th and last dataset, we created a 6-column dataset by taking the three highest (alcohol, density, volatile acidity, sulphates, residual sugar, and pH) and the three lowest correlations. Furthermore, we utilized the Pandas [11] package in our project to read the data and perform various analyses on it since the Wine Quality dataset is stored as a CSV. Apart from that, we used the Matplotlib [12] and Seaborn [12] libraries to make a more in-depth analysis of the data and visualize them.

B. Software

Code is mainly written in Python. Scikit-Learn was used to use learning models in this research. The implementation and datasets are available on GitHub.

C. Models

Linear Regression was used as a regression model for wine quality prediction. Logistic Regression was used as classification model. K-Nearest Neighbor and Multilayer Perceptron were used as both regression and classification models.

1) Linear Regression: Linear regression analysis is used to predict the value of a variable based on the value of another variable. The variable you want to predict is called the dependent variable. The variable you are using to predict the other variable's value is called the independent variable [13].

$$Y = B0 + BiXi + \text{epsilon } i = 0 - n$$

where n represents the number of samples in the dataset, y represents the output variable, B0 is the intercept term, Bi slope term, xi is the input variable, and epsilon is the error term.

2) Logistic Regression: Logistic regression estimates the probability of an event occurring, such as voted or didn't vote, based on a given dataset of independent variables. Since the outcome is a probability, the dependent variable is bounded between 0 and 1 [14].

Logistic Regression uses Logit Function.

$$\log\left(\frac{p}{1-p}\right) = \beta_o + \beta(\text{Age}) \tag{1}$$

Logistic Regression uses Linear equations just like Linear Regression. But the difference between them is that by inserting the result of the linear equation of Logistic Regression into Logit Function, a result for classification is obtained.

3) K-Nearest Neighbor: The K-Nearest Neighbors 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 [15].

Distance metrics are applied to all pairs of vectors to find the closest neighbors. The label is predicted by taking the mod of the occurrences of labels of K nearest neighbors after the distances to the neighbors have been calculated.

The Minkowski metric, which is a generalization of the Euclidean and Manhattan distances, is also used in computations. The Manhattan distance between two vectors is calculated using the following formula:

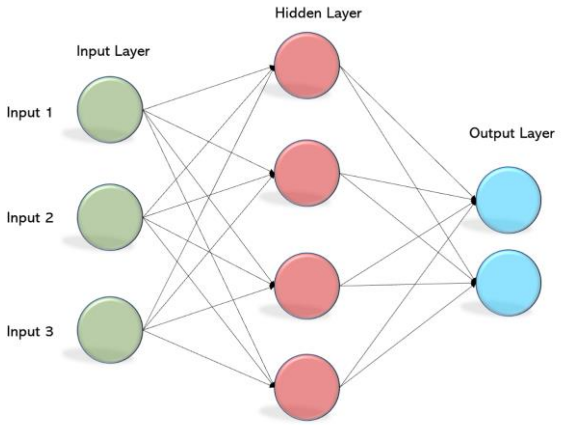
$$d(\vec{x}, \vec{y}) = (\vec{x} - \vec{y})^T (\vec{x} - \vec{y}) \tag{2}$$

as well as the Minkowski metric, which is calculated as

$$d(\vec{x}, \vec{y}) = (\sum_{i=1}^{n} |\vec{x}_i - \vec{y}_i|^p)^{1/p} \tag{3}$$

It is clear that when p equals 1, the distance transforms to Manhattan distance. When p equals 2, the distance is Euclidean.

4) Multilayer Perceptron: A multilayer perceptron (MLP) is a fully connected class of feedforward artificial neural network (ANN). An MLP consists of at least three layers of nodes: an input layer, a hidden layer, and an output layer. While the forward method is used when going from the input layer to the output layer, the backpropagation method is used when coming back from the output layer. The purpose of backpropagation here is to minimize error. In the hidden layer, the activation function is applied. The most common activation functions are the Sigmoid Function and the rectifier linear unit (ReLU).



D. Algorithm

1) Min-Max Scaler: The python Scikit-learn library's MinMaxScaler alters features by scaling them to a specified range. Values are scaled using Equation Q. MinMaxScaler is generally used in regression problems since we have it in our project for the models that are used for regression.

$$X = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{4}$$

2) Standard Scaler: The python Scikit-learn library's Standard Scaler alters features by scaling them using the mean and the standard deviation of the data. It is generally used in classification problems since we have it in our project for the models that are used for classification.

$$z = \frac{x - \mu}{\sigma} \tag{5}$$

3) Principal Component Analysis: PCA is a popular dimensionality reduction technique that is used to represent important pieces of information with smaller dimensions.

E. Evaluation Metrics

While we used score, mean-squared error, and mean-absolute error to assess regression models, we utilized accuracy, weighted precision, and weighted f1 score to evaluate classification models. Weighted f1 score and weighted precision are generally used to evaluate imbalanced datasets because they also use the support of each class during calculation. Since our dataset is also imbalanced, we decided to use them.

IV. Results & Discussions

In this section, we will present our results for both classification and regression and then discuss them.

Regression:
Linear Regression:

	Score	MSE	MAE
Dataset 1	0.2969	0.5513	0.5732
Dataset 2	0.2657	0.5758	0.5872
Dataset 3	0.0270	0.7630	0.6784
Dataset 4	0.2748	0.5687	0.5817

According to the table, the 1st dataset gave the best score which is %29.7, while the 3rd dataset gave the worst score which is %2.7. The reason the third dataset turned out so badly is that we used the 6 features with the lowest correlation. When we look at the literature, we could not find a study that makes wine quality prediction with Linear Regression.

K Nearest Neighbors:

	Score	MSE	MAE
Dataset 1	0.3304	0.5251	0.5399
Dataset 2	0.2474	0.5901	0.5701
Dataset 3	0.1767	0.6456	0.6164
Dataset 4	0.2955	0.5524	0.5580

The first dataset had the best score which is %33, while the third dataset had the worst score which is %17.7, according to the table. We used the 6 features with the lowest correlation in the third dataset, which is why it turned out so badly. We looked through the literature and couldn't find a study that used K-Nearest Neighbors Regressor to predict wine quality.
Multi-Layer Perceptron:

	Score	MSE	MAE
Dataset 1	0.3383	0.5189	0.5577
Dataset 2	0.2633	0.5776	0.5889
Dataset 3	0.1411	0.6735	0.6456
Dataset 4	0.2748	0.5687	0.5832

According to the table, the 1st dataset gave the best score which is %33.8, while the 3rd dataset gave the worst score which is %14.1. The reason the third dataset turned out so badly is that we used the 6 features with the lowest correlation. When we look at the literature, we could not find a study that makes wine quality prediction with Multi-Layer Perceptron Regressor.

Classification:
Logistic Regression:

	Accuracy	Weighted Precision	Weighted F1 Score
Dataset 1	0.9305	0.9	0.9
Dataset 2	0.9298	0.89	0.9
Dataset 3	0.9292	0.86	0.9
Dataset 4	0.9292	0.86	0.9

According to the table, Logistic Regression gave better results in the 1st dataset. Also, we got 93% accuracy, while a similar dataset (Trivedi, Sehrawat) [x] has 76% accuracy. This may be because we have more samples in our test dataset.

K Nearest Neighbors:

	Accuracy	Weighted Precision	Weighted F1 Score
Dataset 1	0.9298	0.91	0.91
Dataset 2	0.9286	0.9	0.9
Dataset 3	0.9311	0.89	0.9
Dataset 4	0.9317	0.91	0.91
Dataset 5*	0.9292	0.91	0.91

** Dataset 5: It is generated applying PCA with n_components=6 to Dataset 1.*

According to the table, we got the best accuracy value in the 4th dataset. With an accuracy of 93.2%, we passed Ting Wei [x] 80.4%. We think that this is because it uses only the first 60 training data before making predictions.

Multi-Layer Perceptron:

	Accuracy	Weighted Precision	Weighted F1 Score
Dataset 1	0.9274	0.9	0.91
Dataset 2	0.9286	0.88	0.9
Dataset 3	0.9274	0.89	0.9
Dataset 4	0.9274	0.87	0.9

According to the table, we got the highest accuracy value in the second dataset. While we got 92.9% accuracy in 1000 epoch and 200 batch size, (Agrawal, Kang) [x] 200 epoch vs 66 batch size, they got 53% accuracy red wine dataset and 48% accuracy white wine dataset. This may be because the hyperparameters and the dataset are different.

V. Conclusion

In this study, the classification and regression of wine quality data into 13 different features from WineQuality dataset was implemented. Selected features were trained and tested using three different classification and regression models namely Linear Regression, Logistic Regression, K-Nearest Neighbors, and Multilayer Perceptron. While K-Nearest Neighbor (KNN) performed the best of the three models for all of the evaluated metrics, with an accuracy of 93.17 %, Logistic Regression had an accuracy of 93.05 %, and Multilayer Perceptron had an accuracy of 92.74 % for classification. While Multilayer Perceptron scored 33.83 % for all of the specified assessment criteria, K-Nearest Neighbor - (KNN) scored 33.04 %, and Linear Regression scored 29.64 % for regression. The results reveal that the implementations produced outcomes that were comparable to earlier studies.

Acknowledgment

For his support and advice throughout our term project, we would like to express our thanks and gratitude to our advisor Mahir Atmış.

References

[1] P. Cortez, A. Cerderia, F. Almeida, T. Matos, and J. Reis, “Modelling wine preferences by data mining from physicochemical properties,” In Decision Support Systems, Elsevier, 47 (4): 547-553. ISSN: 0167-9236.
[2] S. Ebeler, “Linking Flavour Chemistry to Sensory Analysis of Wine,” in Flavor Chemistry, Thirty Years of Progress, Kluwer Academic Publishers, 1999, pp. 409-422.
[3] V. Preedy, and M. L. R. Mendez, “Wine Applications with Electronic Noses,” in Electronic Noses and Tongues in Food Science, Cambridge, MA, USA: Academic Press, 2016, pp. 137-151.
[4] A. Asuncion, and D. Newman (2007), UCI Machine Learning Repository, University of California, Irvine, [Online]. Available: http://www.ics.uci.edu/~mlearn/MLRepository.html
[5] S. Kallithraka, IS. Arvanitoyannis, P. Kefalas, A. El-Zajouli, E. Soufleros, and E. Psarra, “Instrumental and sensory analysis of Greek wines; implementation of principal component analysis (PCA) for classification according to geographical origin,” Food Chemistry, 73(4): 501-514, 2001.
[6] N. H. Beltran, M. A. Duarte- MErmound, V. A. S. Vicencio, S. A. Salah, and M. A. Bustos, “Chilean wine classification using volatile organic compounds data obtained with a fast GC analyzer,” Instrum. Measurement, IEEE Trans., 57: 2421-2436, 2008.
[7] S. Shanmuganathan, P. Sallis, and A. Narayanan, “Data mining techniques for modelling seasonal climate effects on grapevine yield and wine quality,” IEEE International Conference on Computational Intelligence Communication Systems and Networks, pp. 82-89, July 2010.
[8] B. Chen, C. Rhodes, A. Crawford, and L. Hambuchen, “Wineinformatics: applying data mining on wine sensory reviews processed by the computational wine wheel,” IEEE International Conference on Data Mining Workshop, pp. 142-149, Dec. 2014.
[9] K. Agrawal and H. Mohan, "Cardiotocography Analysis for Fetal State Classification Using Machine Learning Algorithms," 2019 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, Tamil Nadu, India, 2019, pp. 1-6.
[10] R. Palmar, Wine Quality, (2018), https://www.kaggle.com/datasets/rajyellow46/wine-quality
[11] Reback, J., McKinney, W., jbrockmendel, den Bossche, J. V., Augspurger, T., Cloud, P., gfyoung, Hawkins, S., Sinhrks, Roeschke, M., Klein, A., Petersen, T., Tratner, J., She, C., Ayd, W., Naveh, S., Garcia, M., Schendel, J., patrick, Hayden, A., Saxton, D., Jancauskas, V., McMaster, A., Gorelli, M., Battiston, P., Seabold, S., Dong, K., chris-b1, h-vetinari, and Hoyer, S.: Pandas-Dev/Pandas: Pandas 1.2.2, Zenodo [code], https://doi.org/10.5281/zenodo.4524629, 2021.
[12] Droettboom M., Hunter J., Caswell T. A. et al. 2016 matplotlib: matplotlib, v1.5.1 doi: 10.5281/zenodo.44579
[13] About Linear Regression. IBM. (n.d.). Retrieved June 7, 2022, from https://www.ibm.com/topics/linear-regression#:~:text=Resources-,What%20is%20linear%20regression%3F,is%20called%20the%20independent%20variable.
[14] What is Logistic Regression? IBM. (n.d.). Retrieved June 7, 2022, from https://www.ibm.com/topics/logistic-regression#:~:text=Logistic%20regression%20estimates%20the%20probability,bounded%20between%200%20and%201.
[15] What is the K-Nearest Neighbors Algorithm? IBM. (n.d.). Retrieved June 7, 2022, from https://www.ibm.com/topics/knn#:~:text=The%20k%2Dnearest%20neighbors%20algorithm%2C%20also%20known%20as%20KNN%20or,of%20an%20individual%20data%20point.

Appendix

You can find the source codes written in this term project from the links below.
https://github.com/UygarKAYA/WineQualityPrediction

CONFUSION MATRICES FOR CLASSIFICATOIN

1) LOGISTIC REGRASSION

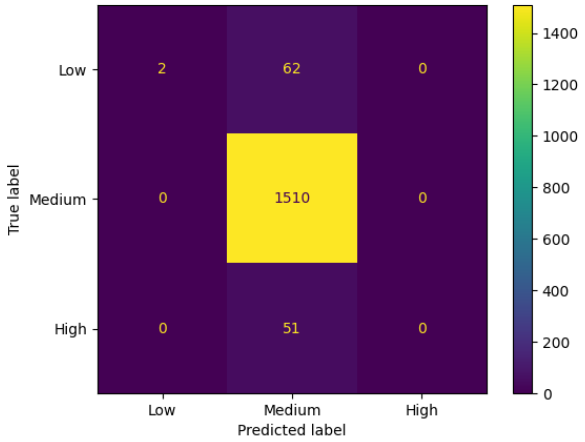


Figure 1: Confusion Matrix of Logistic Regression model with Dataset 1

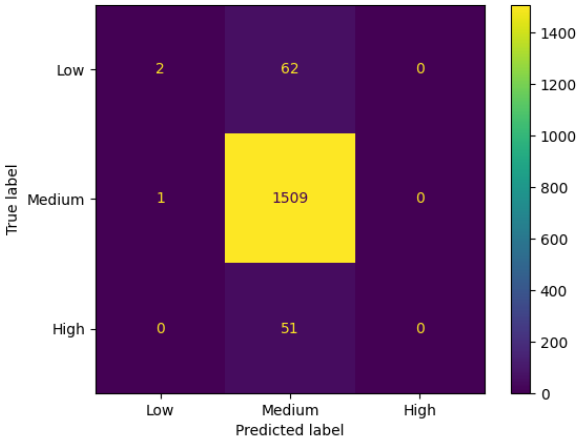


Figure 2: Confusion Matrix of Logistic Regression model with Dataset 2

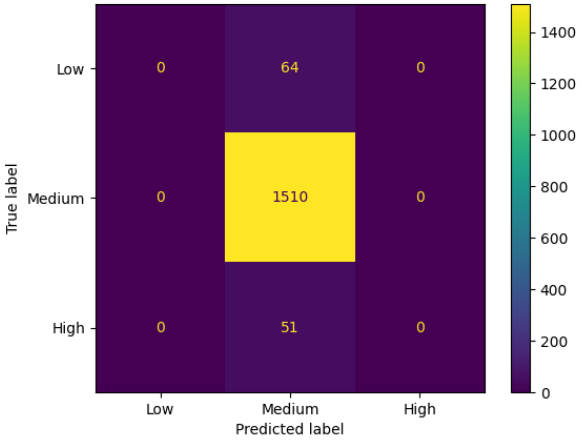


Figure 3: Confusion Matrix of Logistic Regression model with Dataset 3

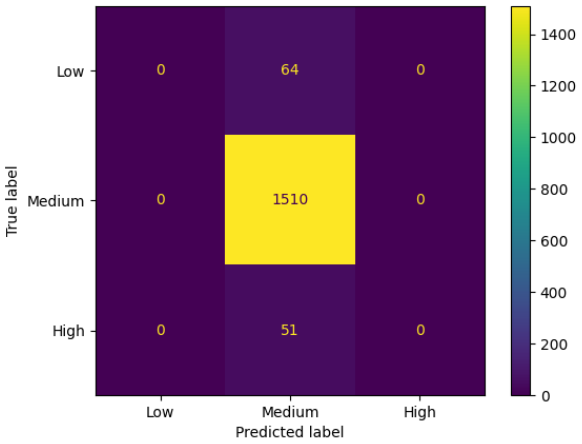


Figure 4: Confusion Matrix of Logistic Regression model with Dataset 4

2) K NEAREST NEIGHBORS

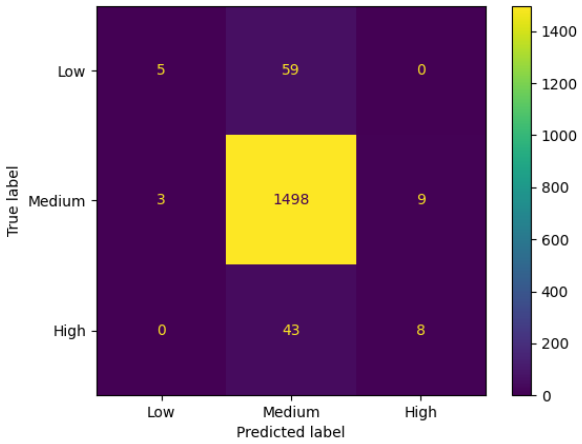


Figure 5: Confusion Matrix of K Nearest Neighbor model with Dataset 1

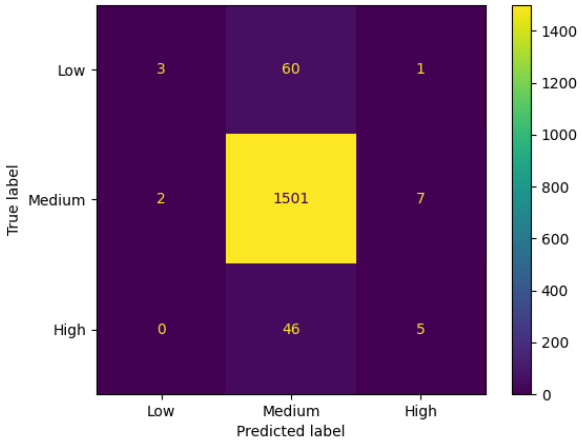


Figure 5: Confusion Matrix of K Nearest Neighbor model with Dataset 2

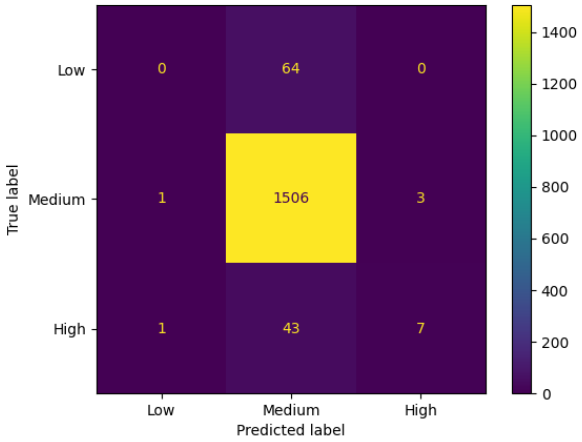


Figure 6: Confusion Matrix of K Nearest Neighbor model with Dataset 3

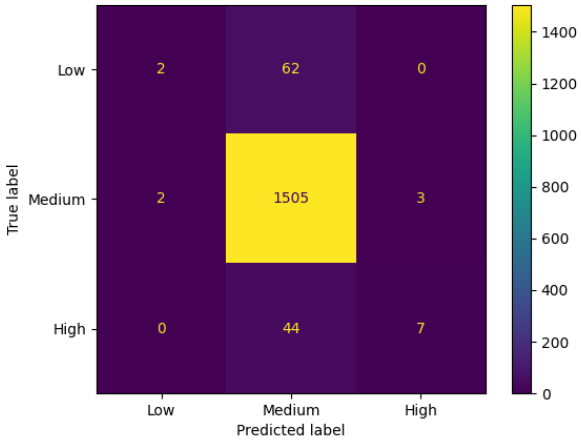


Figure 7: Confusion Matrix of K Nearest Neighbor model with Dataset 4

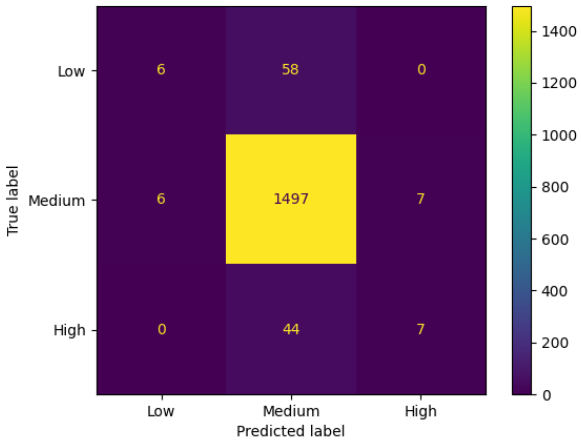


Figure 8: Confusion Matrix of K Nearest Neighbor model with Dataset 5

3) MULTI-LAYER PERCEPTRON

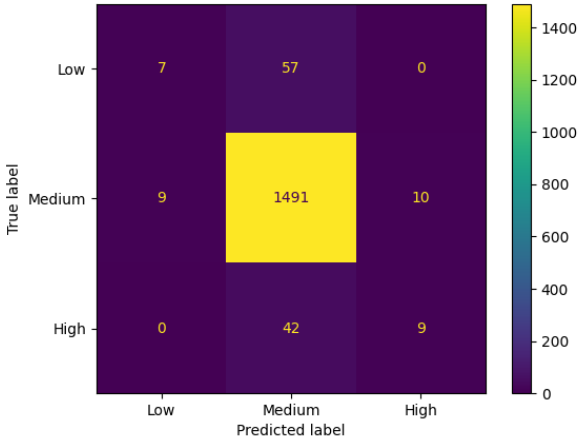


Figure 9: Confusion Matrix of Multi-Layer Perceptron model with Dataset 1

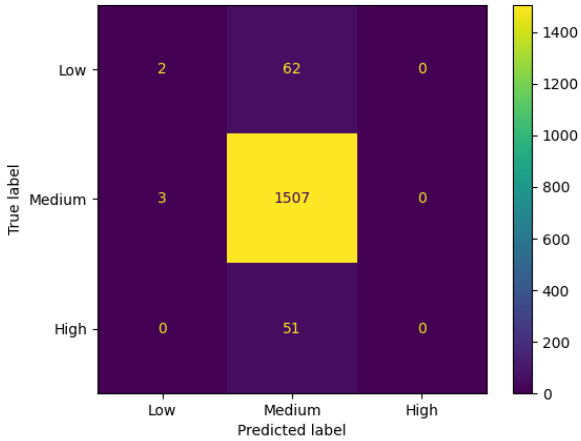


Figure 10: Confusion Matrix of Multi-Layer Perceptron model with Dataset 2

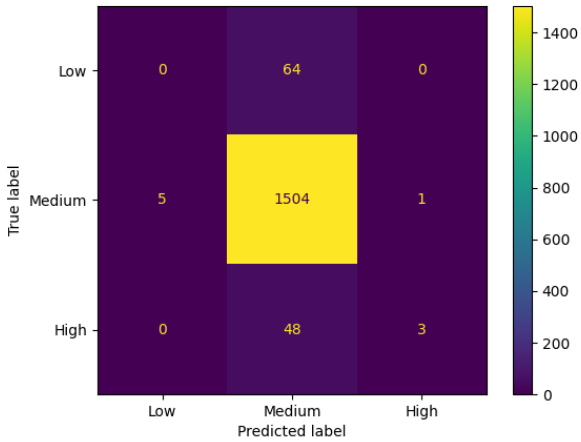


Figure 11: Confusion Matrix of Multi-Layer Perceptron model with Dataset 3

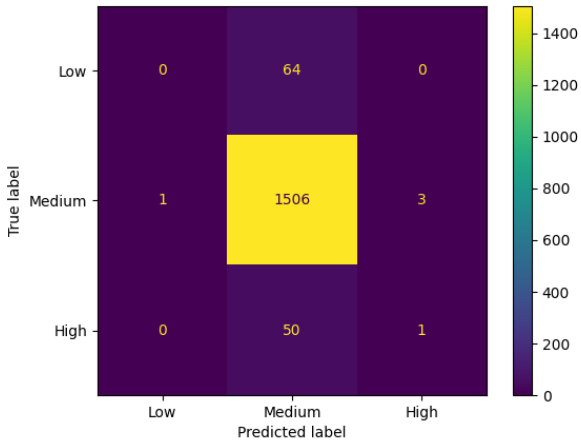


Figure 12: Confusion Matrix of Multi-Layer Perceptron model with Dataset 4