A COMPARATIVE ANALYSIS ON DETERMINING WATER SAFETY USING MULTILAYER PERCEPTRON AND SUPPORT VECTOR MACHINES

Gulce Alper

Gulce.Alper@city.ac.uk

Abstract

This paper covers the comparison of Multilayer Perceptron and Support Vector Machine algorithms on binary classification problem regarding water safety. The best model is tried to be achieved by using various hyperparameters for each of the algorithms. Selected models are evaluated with various performance metrics such as ROC, training time and confusion matrix. It is concluded that the Multilayer Perceptron has a better ability to separate classes.

1. Introduction

There is no doubt that water is one of the most basic needs of life. "*Water is a vital resource for human survival. Safe drinking water is a basic need for good health, and it is also a basic right of humans."* [1] Although it is an indispensable factor in terms of life cycle, the density of some substances in water can negatively affect its quality. Measuring water quality and determining whether it is suitable for use has become critical in order to prevent poisonings, deaths and even epidemics.

In this paper, the effects of a group of ingredients (elements, bacteria and viruses) in water on water quality will be examined and whether the water is suitable for use will be analyzed as to be a binary outcome. In this context, two neural network methods, Multilayer Perceptron (MLP) and Support Vector Machines (SVM), will be compared. *"Classification is one of the important machine learning operations. It is the operation that enables organizations to discover patterns in large or complex data sets."* [2]

The workflow of the study is planned as follows; Giving brief explanations about the operation of the methods in question, examining the data set and performing Exploratory Data Analysis (EDA), defining the architecture and hyperparameters of both methods, comparing the two methods and evaluating the results regarding the gains.

1.1. Multilayer Perceptron (MLP)

Multilayer perceptron (MLP) is a feed-forward decision-making method that is frequently used in the field of artificial neural networks. "*MLP neural networks consist of units arranged in layers. Each layer is composed of nodes and in the fully connected networks considered in this paper each node connects to every node in subsequent layers. Each MLP is composed of a minimum of three layers consisting of an input layer, one or more hidden layer(s) and an output layer.*" [3]

1.2. Support Vector Machines (SVM)

Support Vector Machines (SVM) is one of the most common supervised learning methods that can be used in both regression and classification problems. "*A special property of SVM is SVM simultaneously minimize the empirical classification error and maximize the geometric margin. So SVM called Maximum Margin Classifiers. SVM is based on the Structural risk Minimization (SRM).*" [4]

1.3. Advantages and Disadvantages of MLP and SVM

The advantages and disadvantages of both methods are summarized below:

Multilayer Perceptron		Support Vector Machines	
✚	▬	✚	▬
Can manage missing values	Overfitting if the data is too complicated	More powerful in high dimensions	Not effective in large datasets
Good generalization ability	Time consuming	Memory efficient	Cannot handle noisy data effectively
Can handle complex data	May be unclear about how the network works	Less risk of overfitting	Difficult to interpret

Figure 1. Advantages and Disadvantages of the Methods

2. Dataset

The data set consisting of 21 columns and 7999 rows used in the study was taken from Kaggle. *"This is a set of data created from imaginary data of water quality in an urban environment."* [5] It has been determined that the data set is suitable for obtaining sufficient and effective results for the study area.

2.1. Exploratory Data Analysis

When the data set was examined, it was seen that 20 variables (ingredients) that make up the features were continuous except one ("*ammonia*") that was corrected later. Additionally, the target variable "*is_safe*" column takes the values 0 (unsafe) and 1 (safe) and has been changed to an integer to accommodate the binary format. Then, it was checked whether there were null values in the data set and no null values were found. Additionally, descriptive statistics were calculated to get an idea about the basic characteristics and distribution of the variables.

In order to facilitate the analysis and support it visually, the distribution of each variable on the target variable was drawn with a histogram. (Figure 2) When the histogram was examined, it was clearly seen that there is a class imbalance and the class 0 (unsafe) is dominant in all variables. Besides, it was also noticed that there was skewness in some variables. By applying SMOTE, class imbalance was eliminated and the estimator was prevented from being biased. The class distribution before and after applying SMOTE is shown in Figure 3.

Since the values were not between 0 and 1, normalization was also needed. Firstly, the Z-score Standardization method was preferred because the values were relatively high. After the first normalization process, it was seen that the values were still not between 0 and 1, but close. Therefore, Min-Max Scaling was used as the second normalization method to ensure that the values were between 0 and 1.

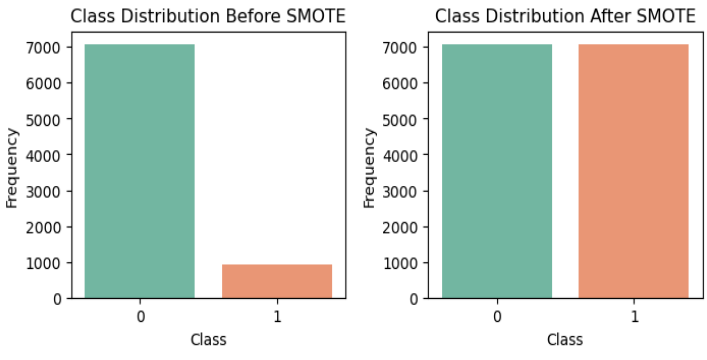

Figure 3. Bar Plot of Class Imbalance After SMOTE

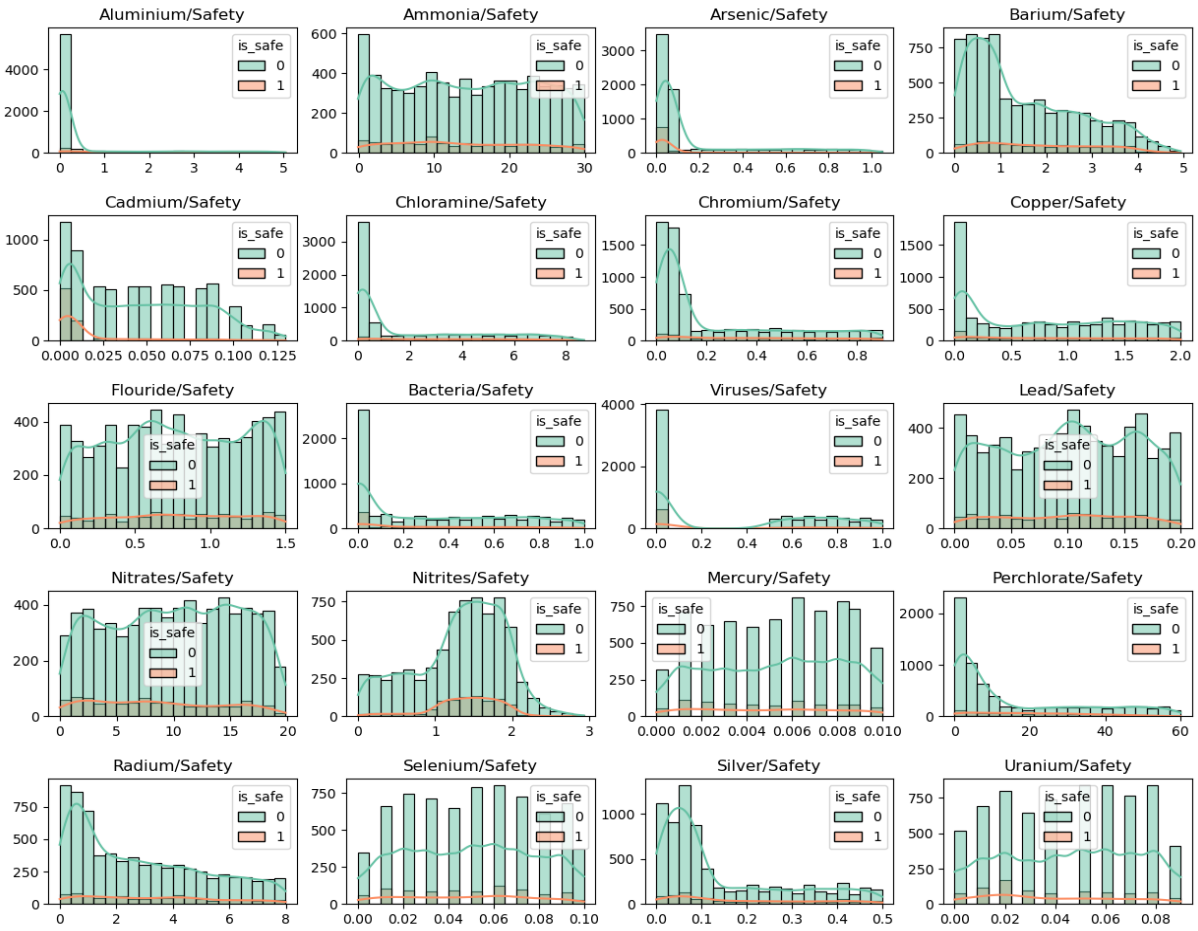

Figure 2. Histogram of Features on Target Classes

3. Methodology

Data was divided at the same rates for both methods, 20% of the data was used for testing and the remaining 80% was used for training. Likewise, 20% of the training set was used for the validation. The "*random_state*" parameter was defined to ensure that the division was deterministic and reproducible for subsequent uses. Detailed descriptions of the operational steps and hyperparameters used for both methods will be explained in the next steps.

3.1. Procedures Followed for MLP

Pytorch, an open source library for machine learning, was used in the architecture of the MLP model. After the data set was divided into test, training and validation sets, all sets were converted into multidimensional arrays called tensors. This step is a mandatory step for models created using Pytorch. In the next step, 3 dataset objects (test, training and validation) were created and these objects were converted into data loaders that would be more effective in both the training and testing stages.

The progression procedure for the training and testing phases was determined by creating two functions named *train* and *evaluate*. As for the train function, it took parameters such as *model, criterion, optimizer, train_loader, num_epochs* and aimed to achieve the outcome by applying forward pass through the model. Additionally, it was planned to update the model parameters with the help of optimizer.step().

Similar to the train function, the evaluate function took parameters such as *model* and *data_loader* and was used to evaluate models in both validation and test datasets. Apart from the optimizer, its general operation is parallel to the train function.

Before training stage, the learning rate and the number of epochs remained constant, taking the values of 0.01 and 150, respectively. Three hyperparameters of MLP were used for hyperparameter tuning, and the aim was to reach the best model by combining different values of these hyperparameters.

Hyperparameter	Values
Hidden size	(10, 15), (20, 10, 15)
Momentum	0.9, 0.95, 0.99
Weight decay	0.001, 0.0001, 0.00001

Figure 4. Hyperparameters for MLP

3.2. Procedures Followed for SVM

Another popular machine learning library, Scikit-learn, was used to implement the method. The 4 hyperparameters of SVM used for hyperparameter tuning and the different values they took are as follows:

Hyperparameter	Values
Regularization parameter(C)	0.1, 1, 10
Kernel	linear, rbf, poly
Gamma (if kernel is rbf)	scale, auto
Degree (if kernel is poly)	2, 3, 4

Figure 5. Hyperparameters for SVM

4. Results & Learnings

4.1. Model Selection

The validation accuracy and hyperparameters obtained through training for both models are as follows.

MLP				SVM				
Hyperparameter			Validation Accuracy	Hyperparameter				Validation Accuracy
Hidden size	Momentum	Weight decay		C	Kernel	Gamma	Degree	
[20, 10, 15]	0.9	0.0001	96.38%	10	rbf	scale	2	96.51%
Test Accuracy				Test Accuracy				
95.27%				96.93%				

Figure 6. Best Model Results

Looking at the results, increasing the number of hidden layers in MLP also increased the validation accuracy, as expected. It is also seen that the momentum and weight loss values for the best model are 0.9 and 0.0001, respectively. When we look at SVM, the values of C, kernel, gamma and Degree for the best model are 10, rbf, scale and 2 respectively. The validation accuracies of both models are very close to each other, and it is possible to say that they generalize well on unseen (test) data.

4.2. Comparison of Methods

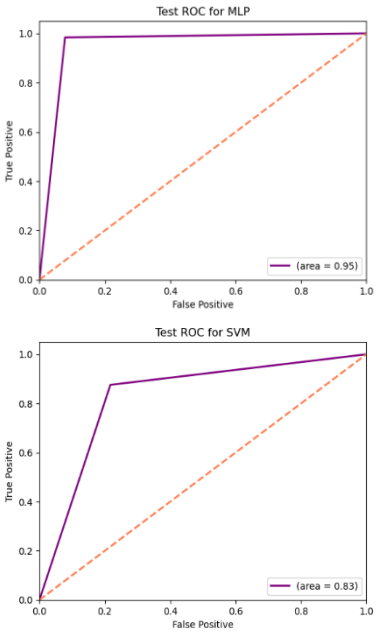

Figure 7. Test ROCs for Methods

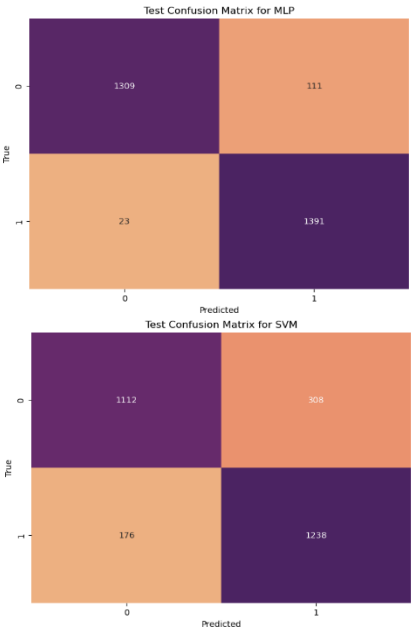

Figure 8. Test Confusion Matrix for Methods

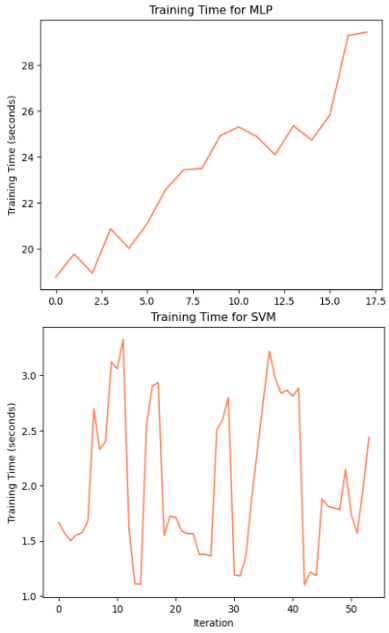

Figure 9. Training Time for Methods

The graphs containing the performance evaluations of the models are as above.

When we look at Figure 7, we see the ROC graph of the testing phase of both models. It is clearly seen that the AUC (Area Under the Curve) of MLP (0.95) is higher than that of SVM (0.83), which means that MLP has better discrimination.

As for Figure 8, we can observe confusion matrices for testing phase. If we calculate Precision values using these matrices,

For MLP: 1391 / (1391 + 111) = 0.92

For SVM: 1238 / (1238 + 308) = 0.80

Looking at the values, it can be said that MLP is more sensitive than SVM and therefore makes more accurate predictions when deciding whether water is safe or not.

In Figure 9 we can see the graphs of training times for each model. An increasing trend can be seen in the MLP chart, which accounts for the longer training period than that of SVM. This may be caused by noisy gradients. On the contrary, fluctuations are observed in the SVM graph, which indicates instability.

5. Conclusion

To summarize the analysis,

It can be said that both models showed high performance in determining water safety. In other words, both models are effective models in binary classification problems. However, the fact that mlp has higher AUC and precision values than SVM makes it a more powerful, although its test accuracy is slightly lower. Additionally, the higher training time of MLP than SVM may be due to the fact that the structure of MLP is relatively more complex. Finally, it is obvious that both models are sufficient and powerful models in line with their own strategies and working principles.

6. References

The source of the dataset and some of the websites used for help are also among the references.

[1]

Meride, Yirdaw, ve Bamlaku Ayenew. “Drinking Water Quality Assessment and Its Effects on Residents Health in Wondo Genet Campus, Ethiopia”. Environmental Systems Research, c. 5, sy 1, Aralık 2016, s. 1. DOI.org (Crossref), https://doi.org/10.1186/s40068-016-0053-6.

[2]

Zanaty, E. A. “Support Vector Machines (SVMs) versus Multilayer Perception (MLP) in Data Classification”. Egyptian Informatics Journal, c. 13, sy 3, Kasım 2012, ss. 177-83. DOI.org (Crossref), https://doi.org/10.1016/j.eij.2012.08.002.

[3]

Delashmit, Walter H., and Michael T. Manry. "Recent developments in multilayer perceptron neural networks." Proceedings of the seventh annual memphis area engineering and science conference, MAESC. Vol. 7. 2005.

[4]

Bhavsar, Himani, and Mahesh H. Panchal. "A review on support vector machine for data classification." International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) 1.10 (2012): 185-189.

[5]

https://www.kaggle.com/datasets/mssmartypants/water-quality

[6]

https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html

[7]

https://pytorch.org/tutorials/beginner/blitz/neural_networks_tutorial.html