Int. J. Math. And Appl., 9(4)(2021), 59–64
ISSN: 2347-1557
Available Online: http://ijmaa.in/

International Journal of *Mathematics And its Applications*

Application of Machine Learning to Predict Water Potability

Ramya Nataraj[1,*]

[1] Green Hope High School, Carpenter Upchurch Rd, Cary, NC, United States.

Abstract: Determining water potability is an important area of research with heavy implications for human health and viability of certain living quarters. In this endeavor, I applied machine learning techniques to predict the potability of water. Based on a data set including chemical and physical water parameters, supervised learning programming is applied to this binary classification problem. Neural network mechanisms are applied to determine the significant parameters that impact potability and accuracy in predicting water potability. Through this research, the significant parameters that affect water potability were identified and used to determine whether a particular combination makes the water potable. Additionally, accuracy of in predicting the potability of the water was found to increase when hidden layers were increased and not a significant impact was realized when increasing the epoch value.

Keywords: Machine Learning, Water Potability, Neural network.

1. Introduction

Access to safe drinking water is essential for the survival and health of people. Determining water potability has been a very important topic in environmental science and is supported by national agencies like the Environment Protection Agency (EPA) [5]. Quality of water and guidelines for maintaining water quality have also been an important topic for World Health Organization (WHO) [6]. There are various factors that impact water potability. In various research, selecting different physical and chemical parameters that affect potability [7] has been an important approach. Some of these factors include water acidity level (measured by pH), hardness of water, dissolved solids found in the water, organic carbon, turbidity, etc. Many such factors can be used to determine whether sampled water is potable or not. Given the vast data sets available capturing various water samples and the complex relationships between different chemical and physical parameters affecting potability, machine learning techniques hold great potential in increasing efficiency and accuracy.

2. Understanding Machine Learning

Machine learning [1] involves computers discovering how they can perform tasks without being explicitly programmed to do so. It uses data to develop computational models that make predictions and help make decisions. In supervised learning, the machine already knows the output of an algorithm before it starts working on or learning it. With the output of the algorithm known, all that a system needs to do is to work out the steps or process needed to reach from the input to the

---

[*] *E-mail: ramyanataraj25@gmail.com*

output [2]. In supervised learning, we are given a set of examples. An example contains the features and a target (also called label). The goal in supervised learning is to develop a model that can predict the target if we are given the features of a new example. A binary classification problem [3] is one in which the target value is placed into two possible categories. In the case of water potability, the features used by machine learning algorithm include chemical and physical parameters like acidity, hardness and dissolved solids, and the label (target) is to determine whether the water is potable or not. Artificial neural networks [4] are a subset of machine learning. Their name and structure are inspired by the human brain and are based on how biological neurons signal one another. Artificial neural networks are composed of node layers, containing an input layer, one or more hidden layers, and an output layer. Each node, or artificial neuron, connects to another and has an associated weight and threshold. If the output of any individual node is above the specified threshold value, that node is activated, sending data to the next layer of the network. Otherwise, no data is passed along to the next layer of the network.

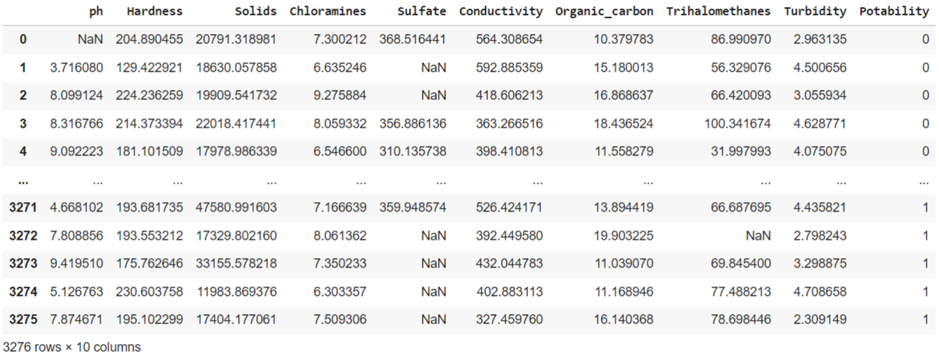
3. Applying Machine Learning to Water Potability

3.1. Objective & approach

The primary research objective was to apply machine learning techniques to predict the potability of water with increased accuracy. Given the available data set including chemical and physical parameters, I implemented supervised learning programming using Python and Pandas and applied that to this binary classification problem. In this approach, neural networks were used to determine the significant parameters that impact potability and to increase the accuracy in predicting it.

3.2. Data set

For this research, I selected a water potability data set [8] containing data nine parameters - pH, hardness, solids, chloramines, sulfate, conductivity, organic carbon, trihalomethanes, and turbidity - that affect potability of water. In machine learning, the parameters (x - variables) are called features, and the target result (y - variable) is called label. In the given data set, containing examples, each row is available data where the values of the features and labels are captured as columns. In each of the examples in the data set covered by this research, the target result is ‘potability’, which is a binary value - potable is 1 and not-potable is 0. The dataset used for this research has 3276 rows - meaning 3276 data samples that reflect water potability. The label of my dataset is shown in binary numbers - either as 1 (the water is potable), or 0 (the water is not safe to consume/is not potable). Figure 1, below shows a sample view of the data set used in this research effort.



	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability
0	NaN	204.890455	20791.318981	7.300212	368.516441	564.308654	10.379783	86.990970	2.963135	0
1	3.716080	129.422921	18630.057858	6.635246	NaN	592.885359	15.180013	56.329076	4.500656	0
2	8.099124	224.236259	19909.541732	9.275884	NaN	418.606213	16.868637	66.420093	3.055934	0
3	8.316766	214.373394	22018.417441	8.059332	356.886136	363.266516	18.436524	100.341674	4.628771	0
4	9.092223	181.101509	17978.986339	6.546600	310.135738	398.410813	11.558279	31.997993	4.075075	0
...	...	...	...	...	...	...	...	...	...	...
3271	4.668102	193.681735	47580.991603	7.166639	359.948574	526.424171	13.894419	66.687695	4.435821	1
3272	7.808856	193.553212	17329.802160	8.061362	NaN	392.449580	19.903225	NaN	2.798243	1
3273	9.419510	175.762646	33155.578218	7.350233	NaN	432.044783	11.039070	69.845400	3.298875	1
3274	5.126763	230.603758	11983.869376	6.303357	NaN	402.883113	11.168946	77.488213	4.708658	1
3275	7.874671	195.102299	17404.177061	7.509306	NaN	327.459760	16.140368	78.698446	2.309149	1

3276 rows × 10 columns

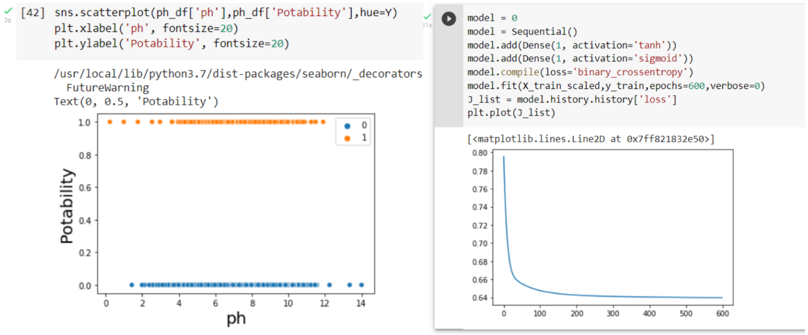
Figure 1. **Sample view of the data set containing features and potability result**

3.3. Cleaning the data set

Data cleaning is a critically important step in any machine learning project [9]. Using the dataset on water potability, I first examined the dataset and recognized null values for some of the features - i.e, numbers/data points do not exist for some features in certain data samples. One technique for data cleaning is to eliminate samples that contain null values. Using Python and Pandas, I cleaned the data set so that any rows that contain null values in any of the features were deleted. This in turn decreased the number of data samples available for research. After implementing data cleaning, the shape of the dataset decreased from (3276 rows, 10 columns), to (2011, 10). In other words, there were about 700 examples within the dataset that contained null values in at least one of the features (columns).

3.4. Feature set that impacts potability

An important aspect of supervised learning is to determine the subset of features that influence the determination of the resultant classification. First, I experimented with select features to determine if any of them have direct correlation to determine the resultant prediction (i.e. whether water is potable or not). When I compressed the dataset to only ph and potability, I noticed that ph had no impact on potability. The scatter plot depicted in Figure 2 represents the data when the ‘ph’ was the only parameter applied to determine the ‘potability’ label. As shown in the Figure 2, even for a given pH value, potability could result in both 0 and 1, rendering it non-deterministic. After applying all 9 features, the resulting graph revealed a cluster of potability dots each pertaining to 0 or 1, thus indicating that pH is not a determining factor.



```
[42] sns.scatterplot(ph_df['ph'],ph_df['Potability'],hue=Y)
     plt.xlabel('ph', fontsize=20)
     plt.ylabel('Potability', fontsize=20)

     /usr/local/lib/python3.7/dist-packages/seaborn/_decorators
       FutureWarning
     Text(0, 0.5, 'Potability')
```

```
model = 0
model = Sequential()
model.add(Dense(1, activation='tanh'))
model.add(Dense(1, activation='sigmoid'))
model.compile(loss='binary_crossentropy')
model.fit(X_train_scaled,y_train,epochs=600,verbose=0)
J_list = model.history.history['loss']
plt.plot(J_list)

[<matplotlib.lines.Line2D at 0x7ff821832e50>]
```

Figure 2. **Relationship between pH and Potability**

When the impact of the ‘hardness’ feature was analyzed on potability, it was similar to the scatterplot and conclusions of the impact of ‘pH’ as shown in Figure 3.

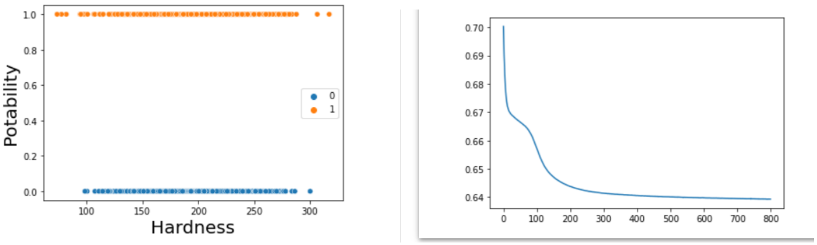


Figure 3. **Relationship between Hardness and Potability**

When more than one feature was experimented with (e.g. pH and water hardness), the scatterplot that resulted was a cluster of orange and blue dots. Orange dots represent a potability value of 1 and blue dots represent potability value of 0.

Such a distribution of is similar to when nine features were experimented with potability. This is shown in Figure 4.

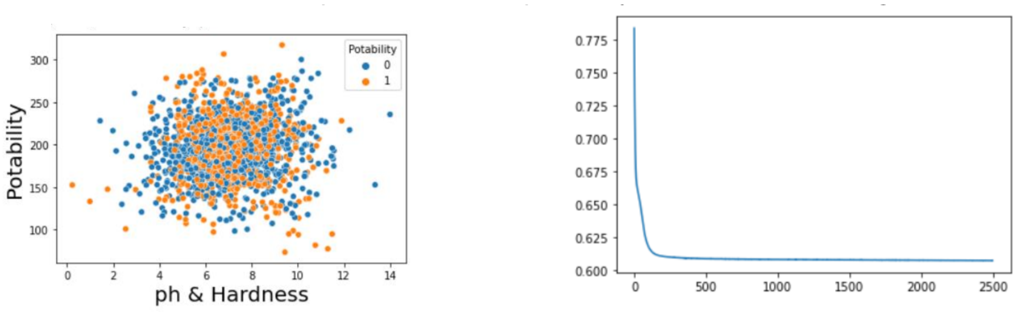
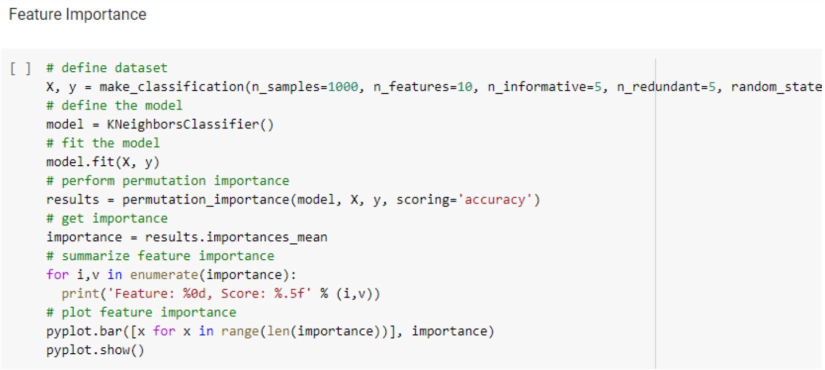


Figure 4. **Measure potability with two features - pH and Hardness**

3.5. Feature Importance

Feature importance refers to techniques that assign a score to input features based on how useful they are at predicting a target variable [10]. Given the feature importance score plays an important role in predictive modeling projects, I applied it in selecting key features. Code for that determination is shown in Figure 5.

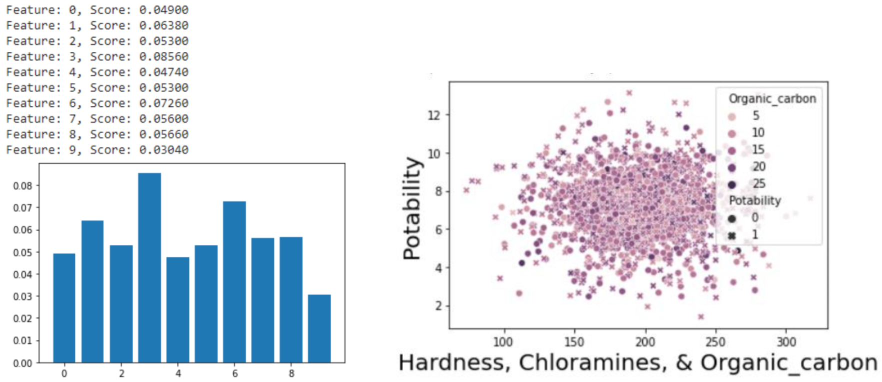


Feature Importance

```
[ ] # define dataset
    X, y = make_classification(n_samples=1000, n_features=10, n_informative=5, n_redundant=5, random_state
    # define the model
    model = KNeighborsClassifier()
    # fit the model
    model.fit(X, y)
    # perform permutation importance
    results = permutation_importance(model, X, y, scoring='accuracy')
    # get importance
    importance = results.importances_mean
    # summarize feature importance
    for i,v in enumerate(importance):
      print('Feature: %0d, Score: %.5f' % (i,v))
    # plot feature importance
    pyplot.bar([x for x in range(len(importance))], importance)
    pyplot.show()
```

Figure 5. **Code to determine feature importance amongst 8 features**

As outlined in Figure 6 below, upon applying this technique, the bar graph shows that the three features that scored the highest are #3, #6, and #8, which pertains to Hardness, Chloramines, and Organic_carbon respectively. These were the three features used to train and validate the neural network.

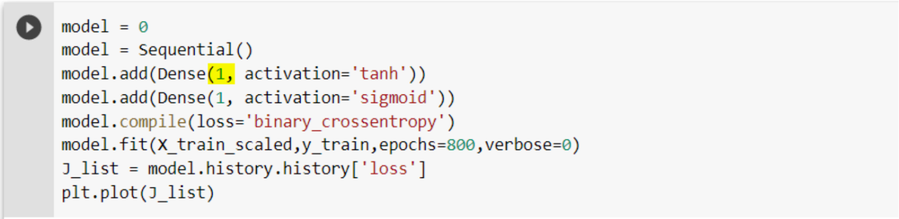


```
Feature: 0, Score: 0.04900
Feature: 1, Score: 0.06380
Feature: 2, Score: 0.05300
Feature: 3, Score: 0.08560
Feature: 4, Score: 0.04740
Feature: 5, Score: 0.05300
Feature: 6, Score: 0.07260
Feature: 7, Score: 0.05600
Feature: 8, Score: 0.05660
Feature: 9, Score: 0.03040
```

Figure 6. **Predicting potability with three important features**

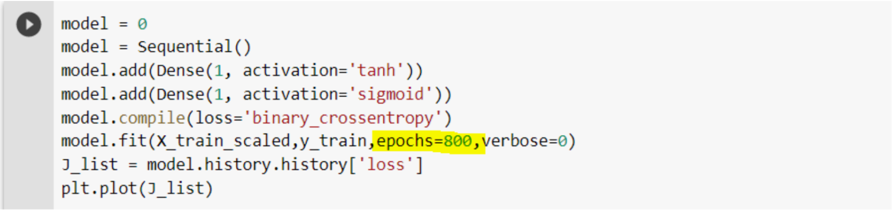
3.6. Improving accuracy with hidden layers & epoch

“Performance of the neural networks depends on the number of layers and number of neurons in each layer” [11]. “It is an art in machine learning to decide the number of epochs sufficient for a network” [12]. Using the three features, I varied the number of hidden layers and epoch values to increase accuracy of prediction. Figures 7 and 8 highlight the code where the hidden layers and epoch values were changed.



```
model = 0
model = Sequential()
model.add(Dense(1, activation='tanh'))
model.add(Dense(1, activation='sigmoid'))
model.compile(loss='binary_crossentropy')
model.fit(X_train_scaled,y_train,epochs=800,verbose=0)
J_list = model.history.history['loss']
plt.plot(J_list)
```

Figure 7. **Code to change hidden layer**



```
model = 0
model = Sequential()
model.add(Dense(1, activation='tanh'))
model.add(Dense(1, activation='sigmoid'))
model.compile(loss='binary_crossentropy')
model.fit(X_train_scaled,y_train,epochs=800,verbose=0)
J_list = model.history.history['loss']
plt.plot(J_list)
```

Figure 8. **Code to change epoch value**

By selecting the three influencing features and varying the hidden layers and epoch values, the models were trained and validated. As shown in the Table 1 below, increasing the epoch with a constant hidden layer did not have an impact; however, changing the hidden layers increased accuracy of prediction. In the experiments done to date, applying 3 hidden layers on the three influencing features, even for 1000 epochs, resulted in increased accuracy of 69% during training and 66% during validation.

Features selected	No. of hidden layers	Epoch	Accuracy (Training / Validation)
Hardness, Chloramines, Organic_carbon	1	1000	66 / 61
Hardness, Chloramines, Organic_carbon	1	2000	66 / 61
Hardness, Chloramines, Organic_carbon	2	1000	69 / 65
Hardness, Chloramines, Organic_carbon	2	2000	69 / 64
Hardness, Chloramines, Organic_carbon	3	1000	69 / 66

Table 1. **Tuning hidden layers and epoch to increase accuracy**

4. Summary

In this research, I applied machine learning techniques to predict potability of water. Based on the data set that includes chemical and physical parameters, supervised learning programming was applied to this binary classification problem. Artificial neural network mechanisms are applied to determine the significant parameters that impact potability and accuracy in predicting water potability. Among the nine features in the data set, three of the features had higher scores - which pertained to Hardness, Chloramines, and Organic_carbon. Those were selected to train the models. Experiments were done by increasing the hidden layers and increasing the epochs. Accuracy of predicting the potability of the water was found to increase when hidden layers were increased and not a significant impact was realized when increasing the epoch value.

Acknowledgements

I would like to sincerely thank Prof. Guillermo Goldsztein from Georgia Institute of Technology, USA (Georgia Tech) for teaching me the fundamental concepts of machine learning, helping me select a research topic, and guiding me throughout the entirety of this project. I would also like to sincerely thank Ms. Davida Kollmar from New York University for assisting me with programming models.

References

[1] ——, *Wikipedia Machine Learning*, https://en.wikipedia.org/wiki/Machine_learning

[2] ——, *Big Data Made Simple: Machine learning explained: Understanding supervised, unsupervised, and reinforcement learning by Ronald Van Loon*, https://bigdata-madesimple.com/machine-learning-explained-understanding-supervised-unsupervised-and-reinforcement-learning

[3] ——, *Machine Learning Mastery: 4 Types of Classification Tasks in Machine Learning*, https://machinelearningmastery.com/types-of-classification-in-machine-learning

[4] ——, *IBM Cloud Learn Hub: What are Neural Networks?*, https://www.ibm.com/cloud/learn/neural-networks

[5] ——, *Water Research, Environmental Protection Agency*, https://www.epa.gov/water-research

[6] ——, *Guidelines for Drinking Water Quality, WHO*, https://www.who.int/water_sanitation_health/dwq/gdwq0506.pdf

[7] Garcia-Avilia, *Evaluation of water quality and stability in the drinking water distribution network in the Azogues city, Ecuador*, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5996164

[8] ——, *Water Quality: Drinking water potability, Aditya Kadiwal*, https://www.kaggle.com/adityakadiwal/water-potability

[9] ——, *How to Perform Data Cleaning for Machine Learning with Python, Jason Brownie*, https://machinelearningmastery.com/basic-data-cleaning-for-machine-learning

[10] ——, *How to calculate feature importance with Python, Jason Brownie* https://machinelearningmastery.com/calculate-feature-importance-with-python

[11] ——, *The Math Behind Training of a Neural Network, Sergen Cansiz*, https://towardsdatascience.com/adventure-of-the-neurons-theory-behind-the-neural-networks-5d19c594ca16

[12] https://deepai.org/machine-learning-glossary-and-terms/epoch