Using Predictive Modelling to Assess Water Safety

Yash Jain, Ting-Lun Hsu, YaYuan Yuan, Yaun Liu
Boston University
yashjain@bu.edu, hsutingl@bu.edu, yayuany@bu.edu, yuanl1@bu.edu

Abstract

Water safety is a very pressing problem faced by many people and governments around the world. Machine Learning may offer a scalable solution in assessing the safety of water obtained from various water sources and filtering methods. Our study showed promising results with K-Nearest Neighbours.

1 Understanding the Dataset

The dataset has samples from 3276 different sources, for each sample we are provided 9 different features related to the potability of the water:

- pH
- Hardness
- Solids
- Chloramines
- Sulfate
- Conductivity
- Organic Carbon
- Trihalomethanes
- Turbidity

1.1 Missing Data

1065 samples had at least one missing value in one of the 10 features, potability was not a missing value for any sample in the dataset, but one or more related features were missing for the 1065 samples mentioned.

ph	491
Hardness	0
Solids	0
Chloramines	0
Sulfate	781
Conductivity	0
Organic Carbon	0
Trihalomethanes	162
Turbidity	0
Potability	0

Table 1: Number of Missing Values for each feature

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

1.2 Variable of Measure

The potability is a binary classification, one for water which is potable, zero for water which is not potable. The fraction of water that was deemed potable in the dataset was 0.39, the fraction of the water deemed not potable in the dataset was 0.61. The data was imbalanced towards water that was not potable.

2 Literature Review

Other researchers have implemented different prediction models on this dataset. We found analysis on the following models through previous implementations: Logistic Regression, Support Vector Machine, K-Nearest Neighbors, Decision Trees, Naive Bayes, Random Forest Classifier, XGBoost Classifier, and Artificial Neural Networks.

The highest prediction accuracy on this dataset was 0.79 using a Random Forest Classifier. This model used mean imputation after finding that the standard deviation for all the features is quite low. Another model that was able to generate a relatively high prediction accuracy was K-Nearest Neighbors, 0.73. Across different papers, the highest accuracy was coming from a Random Forest Classifier predictive model.

3 Preprocessing the Data

Since all our features were numerical, we preprocessed for missing values and normalization.

3.1 Missing Values

Having observed a low standard deviation across all the features, mean imputation was used to impute missing data.

3.2 Normalization

After having filled missing data using mean imputation, the data was normalized with a standard z-score scaling.

4 Logistic Regression

The first predictive model we ran on the data was a Logistic Regression. Logistic Regression find a line of best for the data and then use the line as probabilistic boundary to spilt the data into two classes. It felt ideal in setting a baseline since we did have a binary variable of measure.

This established our baseline accuracy from which we could then improve upon with other models. Tuning for the parameter we were able to get the following results.

	precision	recall	f1-score	support
0	0.64	1.00	0.78	828
1	0.82	0.02	0.04	483
accuracy			0.64	1311
macro avg	0.73	0.51	0.41	1311
weighted avg	0.70	0.64	0.50	1311

Figure 1: The performance of our logistic regression model

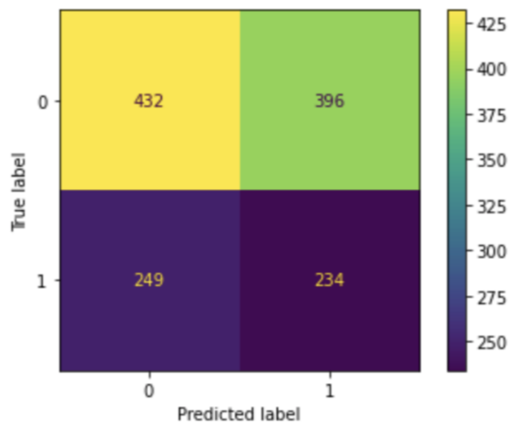


Figure 2: Confusion matrix of results

5 Support Vector Machine

The second predictive model we ran was Support Vector Machine. SVM tries to find the boundary of the largest margin in a dataset to separate the data into two binary classes

5.1 Linear Kernel

Implementing and SVM with a linear kernel, we tuned for the hyper-parameter C to try and obtain the best accuracy in classification.

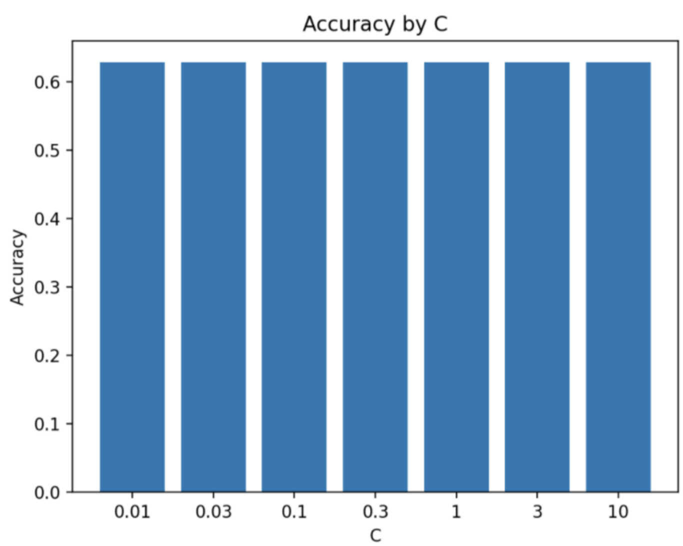
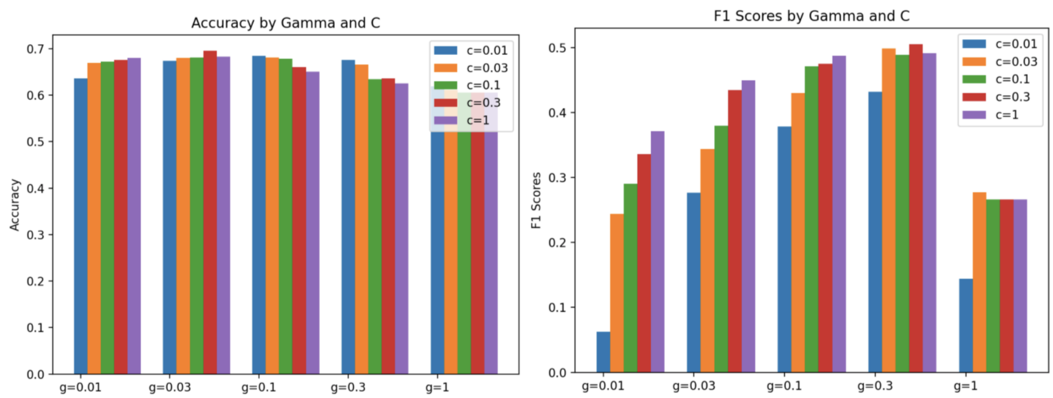


Figure 3: The accuracy of the SVM with linear kernel

5.2 rbf Kernel

SVM performed better with a Radial Based Function kernel, here we had to tune for the hyper-parameters C and γ.



(a) Accuracy (b) F1 Score

Figure 4: The performance of the SVM with rbf kernel

6 K-Nearest Neighbours

The final predictive model we used was K-Nearest Neighbours(knn) which classifies data by considering the classification of its k nearest neighbors, where k is a parameter we specify to the algorithm.

6.1 Balancing the Data

As we discussed earlier the dataset was imbalanced. We had 0.39 of the data as potable water and 0.61 as not potable.

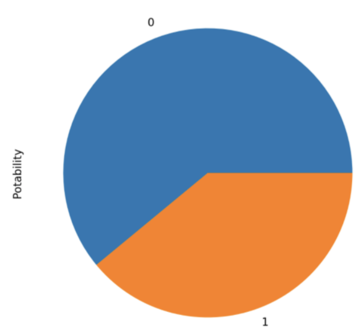


Figure 5: The imbalance of the dataset

This imbalance affects the performance of knn, hence we balanced the data by picking a random number of fixed samples from each of the classification categories. This alone improved the accuracy from 0.633 to 0.796 with K fixed.

6.2 Picking K

We then tuned to see which K would yield the best performance.

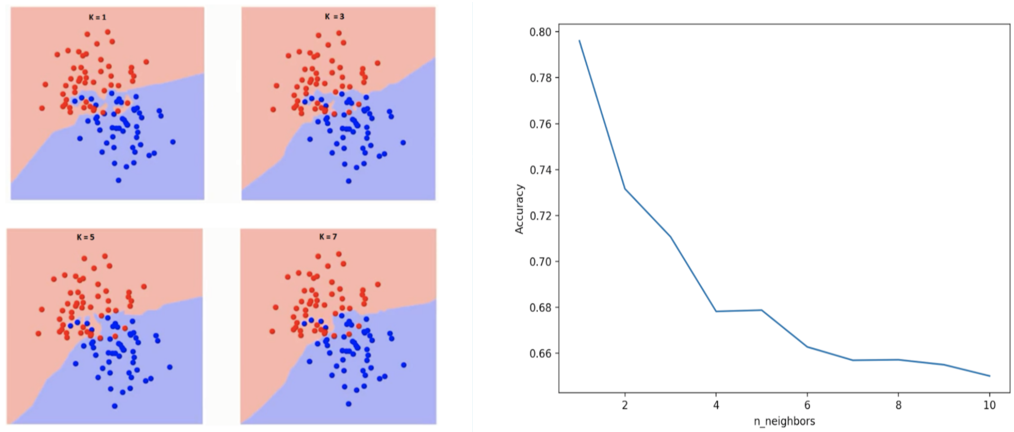


Figure 6: The performance of KNN for different values of K

7 Outcome

We were able to replicate the performance accuracy that we saw in our literature review with the K-Nearest Neighbours Algorithm. However, given more time to understand the nuances of the dataset and create a more sophisticated model with the algorithms we used, we could see major improvements in the performance of our models. Machine Learning, as it does in many fields, possesses the potential to create a solution that would quickly allow us to test water anywhere in the world withing seconds, and work towards a safer future where water-borne sicknesses can be more easily prevented.