# **Water quality prediction**

# **DATA 606**

# **Capstone in Data Science**

# **Project by: Rashmitha Challa(IZ48096)**

# [**rchalla1@umbc.edu**](mailto:rchalla1@umbc.edu)

# **Supervised by: Chaojie (Jay) Wang, D.Sc.**

# Table of Contents

1. Abstract .............................................................................................................................. 3
2. Introduction ....................................................................................................................... 4
3. Literature Review ............................................................................................................. 5
4. Data Description................................................................................................................ 8
5. Exploratory Data Analysis ............................................................................................. 12
6. Approach ......................................................................................................................... 19
7. Modeling Algorithms ...................................................................................................... 21
8. Results .............................................................................................................................. 22
9. Conclusions ...................................................................................................................... 30
10. References ........................................................................................................................ 33

# Abstract

Access to clean and safe drinking water is essential for the well-being of humans. However, ensuring access to safe drinking water remains a challenge, even in developed countries like Canada. Conventionally, water quality is tested through expensive and time-consuming laboratory analyses that can vary depending on the number of parameters being tested. This capstone project aims to explore the use of supervised machine learning algorithms to accurately predict water quality. The project will identify significant parameters required to predict potable water and explore various machine learning algorithms to classify water potability as safe or unsafe.

The project will use the Water Quality dataset, which contains nine parameters. By testing several machine learning algorithms on the identified predictors, the project aims to determine which parameters have the highest correlation with potability and compare the performance of the models. The six predictive modeling techniques used in Python are logistic regression, k-nearest neighbor regression, decision tree classifier, random forest classifier, support vector model classifier, and XGBoost algorithm. The results show that the Support Vector Machine Classifier had the best performance with an accuracy of 83.81% when modeled using train-test split and 87.98% using K-Fold cross-validation.

# Introduction

Water that is pure and suitable for drinking is said to be potable. Residents may not trust the water supply due to a variety of reasons, including geography and the distance of the reserves, ongoing underfunding that results in subpar treatment facilities, and previous government policies, for fear of elevated levels of heavy metals or toxins like E. coli. Consuming non-potable water can have a variety of negative effects on your health over the long term, depending on how severe the toxins or pathogens are in the water. Many First Nations communities, according to the Canadian government, do not currently have access to clean drinking water. According to the most recent update from the Government of Canada, 132 long-term drinking water advisories have been lifted since 2015; nevertheless, as of April 25, 2022, 33 long-term drinking water advisories will still be in place in 28 municipalities.

The turnaround time for a water sample submitted by a private citizen in the province of Ontario to a Public Health Ontario Laboratory is four days, and samples are only accepted Monday through Friday (Public Health Ontario, n.d.). Simple water quality evaluations can take time, be expensive, and involve some work to collect samples, ensure they are transported properly, and analyze them, excluding the cost of one sample kit and the lab report. Water is a necessity for all life on Earth, but it's also a fundamental human right to have access to clean water and sanitary facilities. Given that Canada is a water-rich nation, accounting for an estimated 7% of the world's renewable freshwater supply, the typical citizen shouldn't have to pay for their own access to safe drinking water. Because the expense of providing good quality control is much more than the impact on society and the local economy, water management is a crucial component of human survival everywhere in the globe. The strategy of using machine learning algorithms to forecast water potability with moderate accuracy could help guarantee that everyone has access to real-time water quality.

The data set was retrieved from:

https://www.kaggle.com/datasets/adityakadiwal/water-potability/

The raw data and processes for this study can be accessed from:

https://github.com/annsam0115/CIND820

# Literature Review

This study of supervised machine learning approaches will look at potential improvements in predicting water quality, especially for potability, or water that is safe to consume. This project will use six well-known machine learning techniques in an effort to support the notion of using AI to help regional water quality evaluations.

Multiple supervised and unsupervised machine learning methods have been used in earlier studies on water quality in an effort to forecast clean water with accuracy. In one study employing the Pakistan Council of Research in Water Resources (PCRWR) dataset, Ahmed et al.'s exploratory data analysis was able to clean the data by first filtering away laborious and pointless aspects before sub selecting only the most correlated attributes (Ahmed et al., 2019). To generate the water quality index (WQI), additional data processing methods like normalization were used. They used 10 classification algorithms and 5 different regression techniques in their investigation. Their methodology demonstrated that they were able to attain the greatest results for regression utilizing gradient boosting and polynomial regression with a mean absolute error (MAE) of 1.9642 and 2.7273, respectively, using four parameters: temperature, turbidity, pH, and total dissolved solids. The classifier was most effective when using multi-layer perception (MLP) classification, which had an accuracy of 85.07%..

Yoga Lakshmi & Mahalakshmi (2021) used linear, polynomial, and logistic regression on their dataset across 13 common water quality metrics to forecast water quality in Indian rivers using machine learning. The most intriguing thing they learned from their research was that different water toxins may theoretically undermine water quality data. They advised that in order to decrease the effects of contaminated water, it was essential to set up a more practical system for collecting water quality parameter data, particularly for the assessment of pH, turbidity, temperature, and TDS parameters. However, in rivers or lakes, the ecological health of the water source can significantly effect water quality at any particular instant, but may not necessarily represent the water quality as a whole. Data gathering is consistent in more controlled collecting sites, such as tanks or treatment facilities.

In Castillo et al.'s study from the year 2022, both classification and regression models were used, and it was discovered that multiple linear regression, when compared to other regression models, had a residual square error (RSE) of 3.262 on 15 of the 17 parameters. Additionally, 93% of the 17 parameters were correctly classified when the logistic regression model was used as a classification model. Curiously, their performance declines when the categorization model's parameters are reduced from 17 to 15. They introduced additional biological parameters, such as fats, oils, and grease, and biological oxygen demand, which were not present in other related studies that were examined, when comparing the parameters of their dataset with the one used in this study. They also included more chemical constituents as parameters.

In the study by Ubah et al. (2021), they employed Artificial Neural Network (ANN), a classification method comparable to linear regression, to anticipate water quality metrics for irrigation purposes. In contrast to machine learning algorithms, neural network algorithms were created to resemble the human brain by incorporating the concepts of bias and threshold into the modeling algorithm. The performance of the ANN algorithm used four parameters: pH, TDS, electrical conductivity, and sodium, with R-squared values ranging from 0.951 to as high as 0.989. There are three levels in artificial neural networks: the input, hidden, and output layers. These layers' architecture and the importance placed on their connections enable training using feed-forward back-propagation training techniques. Again, their research shows that a variety of factors, including sample site, season, and water quality indicators, can significantly affect the outcomes of water sampling.

A drinking water quality assessment study in Ethiopia's Wonda genet campus (Meride & Ayenw, 2016) used water sampling to determine water drinkability using three physio-chemical parameters and eight chemical constituent parameters, but it also revealed that additional testing of coliforms is required in conjunction with other indicators. This research went beyond machine learning studies related to water quality predictions. Coliforms are bacteria that come from animals and can be found in their faces as well as in plant and soil matter. Escherichia coli, sometimes known as E. coli, is a prominent species of fecal coliform that can cause serious sickness, even though signs of coliforms in water may not be disease-causing. Testing for coliforms is an important factor to consider when establishing appropriate water testing and predicting mechanisms, even though all the studies about water quality prediction mentioned previously use similar physio-chemical parameters, such as: turbidity, total dissolved solids, and electrical conductivity; and chemical constituent parameters, such as: pH, sulfates, and chlorides. The characteristics employed in the aforementioned research' individual data sets vary, but they all share the goal of creating a reliable classifier for water quality. Three of the parameters from the potability data set utilized in this experiment were used in at least three of the previously examined studies. The improvement of the classification of water quality in future studies into accurate and efficient water quality predictions would benefit from further development and research into hyper-tuning algorithms.

# Data Description

A comma-separated values format is used to display the Water Quality Dataset (water\_potatbility.csv) (Kadiwal, 2021). The collection includes measurements of water quality for 3276 different bodies of water. There are nine categorical attributes and one class attribute, potability, that can be employed as predictor attributes. The binary representation of the categorization attribute uses 1 for potability and 0 for non-potability. 1998 entries in the dataset are categorized as drinkable, whereas 3276 records are labeled as non-potable. A summary of the data is shown in Table 1, and each attribute is described below. The data were imported into Python 3.7.

1. pH value:

pH is an important parameter in evaluating the acid–base balance of water. pH is measured between 0 to 14. The lower the value of pH, the more acidic and the higher the pH, the more alkaline the water condition. The WHO has recommended maximum permissible limit of pH from 6.5 to 8.5 (World Health Organization, 2022).

1. Hardness:

This capacity of water to precipitate soap in milligrams per litre (mg/L). Hardness is mainly caused by calcium and magnesium salts. These salts are dissolved from geologic deposits through which water travels. The length of time water is in contact with hardness producing material helps determine how much hardness there is in raw water.

1. Solids (Total dissolved solids - TDS):

The total dissolved solids in ppm (parts per million). Water has the ability to dissolve a wide range of inorganic and some organic minerals or salts such as potassium, calcium, sodium, bicarbonates, chlorides, magnesium, sulfates etc. These minerals produce unwanted taste and diluted color in the appearance of water. Water with high TDS values indicate that water is highly mineralized. The desirable limit for TDS is 500 mg/L with a maximum limit of 1000 mg/L which prescribed for drinking purpose.

1. Chloramines:

The measure of chloramines in ppm. Chlorine and chloramine are the major disinfectants used in public water systems. Chloramines are most commonly formed when ammonia is added to chlorine to treat drinking water. Chlorine levels up to 4 mg/L or 4 ppm are considered safe in drinking water.

1. Sulfate:

The measure of sulfates dissolved in mg/L. Sulfates are naturally occurring substances that are found in minerals, soil, and rocks. They are present in ambient air, groundwater, plants, and food. Sulfate concentration in seawater is about 2,700 mg/L. It ranges from 3 to 30 mg/L in most freshwater supplies, although much higher concentrations (1000 mg/L) can be found in some geographic locations.

1. Conductivity:

Electrical conductivity of water in micro-siemens per centimeter (μ∙S/cm). Pure water is not a good conductor of electric current but it’s a good insulator. Increase in ion concentration enhances the electrical conductivity of water. Generally, the number of dissolved solids in water determines the electrical conductivity. Electrical conductivity (EC) actually measures the ionic process of a solution that enables it to transmit current.

According to WHO standards, EC values should not exceed 400 μ∙S/cm.

1. Organic Carbon:

The amount of organic carbon is measured in ppm. Total Organic Carbon (TOC) in source waters comes from decaying natural organic matter (NOM) as well as synthetic sources. TOC is a measure of the total amount of carbon in organic compounds in pure water. According to US EPA < 2 mg/L as TOC in treated drinking water, and < 4 mg/L in source water which is use for treatment.

1. Trihalomethanes:

The amount of trihalomethanes (THMs) in micro-grams per litre (µ∙g/L). THMs are chemicals which may be found in water treated with chlorine. The concentration of THMs in drinking water varies according to the level of organic material in the water, the amount of chlorine required to treat the water, and the temperature of the water that is being treated. THM levels up to 80 µ∙g/L is considered safe in drinking water.

1. Turbidity:

The measure of light emitting property of the water in nephelometric turbidity units

(NTU). The turbidity of water depends on the quantity of solid matter present in the

suspended state. It is also the test is used to indicate the quality of waste discharge with respect to colloidal matter. The WHO recommends value of 5.00 NTU.

1. Potability:

Indicates if water is safe for human consumption where 1 equals Potable and 0 equals.

Not potable.

Attribute Summary:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Attributes | Type | Min | Max | Mean | Standard Deviation | Distinct Values | Missing Values |
| 1 | pH | quantitative | 0.00 | 14.00 | 7.08 | 1.59 | 2785 | 491 |
| 2 | Harness | quantitative | 47.43 | 323.12 | 196.37 | 32.88 | 3276 | 0 |
| 3 | Solids | quantitative | 320.94 | 61227.20 | 22014.09 | 8768.57 | 3276 | 0 |
| 4 | Chloramines | quantitative | 0.35 | 13.13 | 7.12 | 15.8 | 3276 | 0 |
| 5 | Sulfate | quantitative | 129.00 | 481.03 | 333.78 | 41.42 | 2495 | 781 |
| 6 | Conductivity | quantitative | 181.48 | 753.34 | 426.21 | 80.82 | 3276 | 0 |
| 7 | Organic Carbon | quantitative | 2.20 | 28.30 | 14.28 | 3.31 | 3276 | 0 |
| 8 | Trihalomethanes | quantitative | 0.74 | 124.00 | 66.40 | 16.18 | 3114 | 162 |
| 9 | Turbidity | quantitative | 1.45 | 6.74 | 3.97 | 0.78 | 3276 | 0 |
| 10 | Potability | nominal | - | - | - | - | 2 | 0 |

Table 1. Attribute summary of water potability dataset

# Exploratory Data Analysis

Visual Analysis:

Both nonportable and potable records exhibit a normal/Gaussian distribution pattern when the distribution of the data within each of the predictor variables is plotted. This is a bell-shaped ideal result, with the exception of Solids where it is somewhat right skewed. This enables us to make choices during the data cleaning stages and informs us that the data distribution is acceptable without requiring us to reject any of the predictor variables.

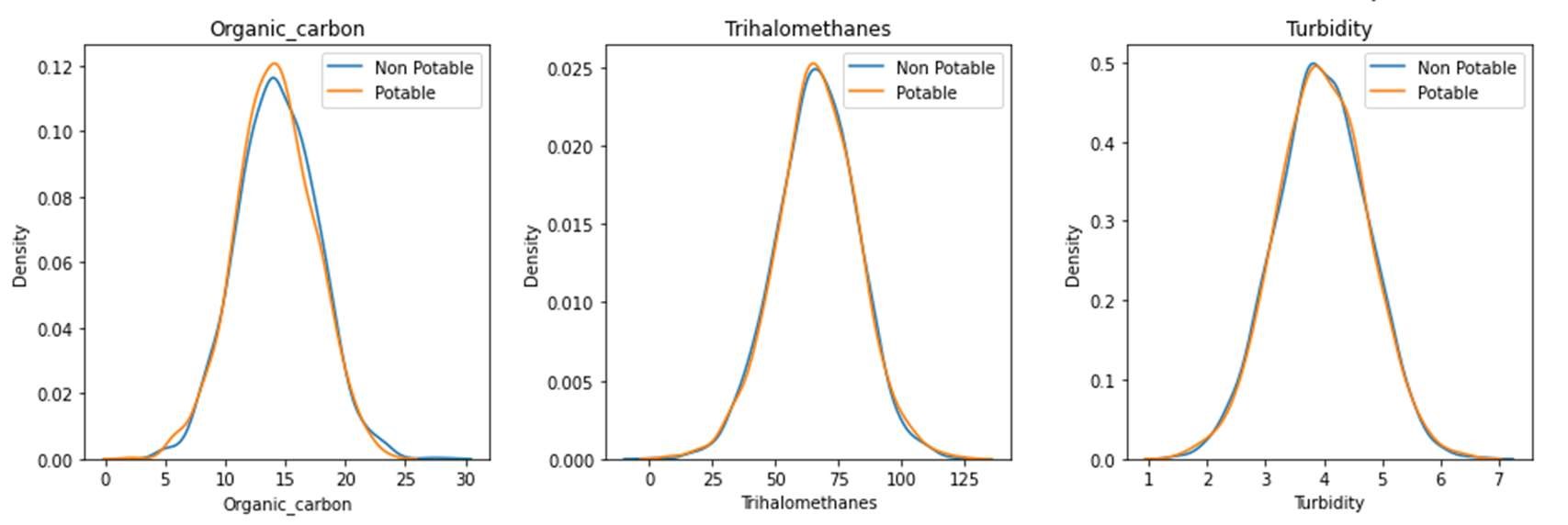


Figure 1. Distribution graph of each predictor variables differentiating between non-potable and potable records.

Three of the predictor variables—pH, sulfate, and trihalomethanes—have missing data points in the original dataset. There were 162 missing results for trihalomethanes, 491 missing values for pH, and 781 missing values for sulfate. We are unable to exclude those entries from the dataset because the fraction of missing values is too high to ignore.

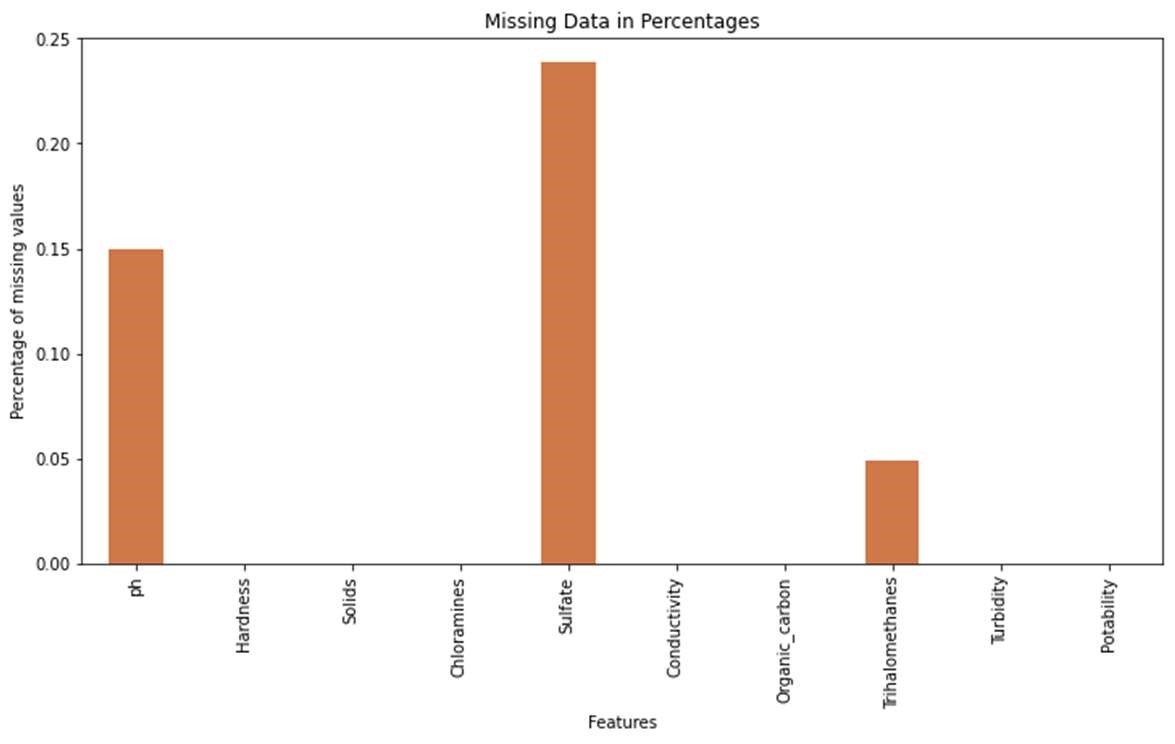


Figure 2. Percentage of missing data points in each predictor variable.

Since we have normal/Gaussian distribution, it was safe to replace all the missing values with the mean value of their corresponding variable.

Using boxplot, we can see the data distribution of the nine predictor variables and clearly identify outlier data points outside minimum and maximum whiskers within all variables.

We can also confirm the skewness in Solids as seen in the distribution plots.

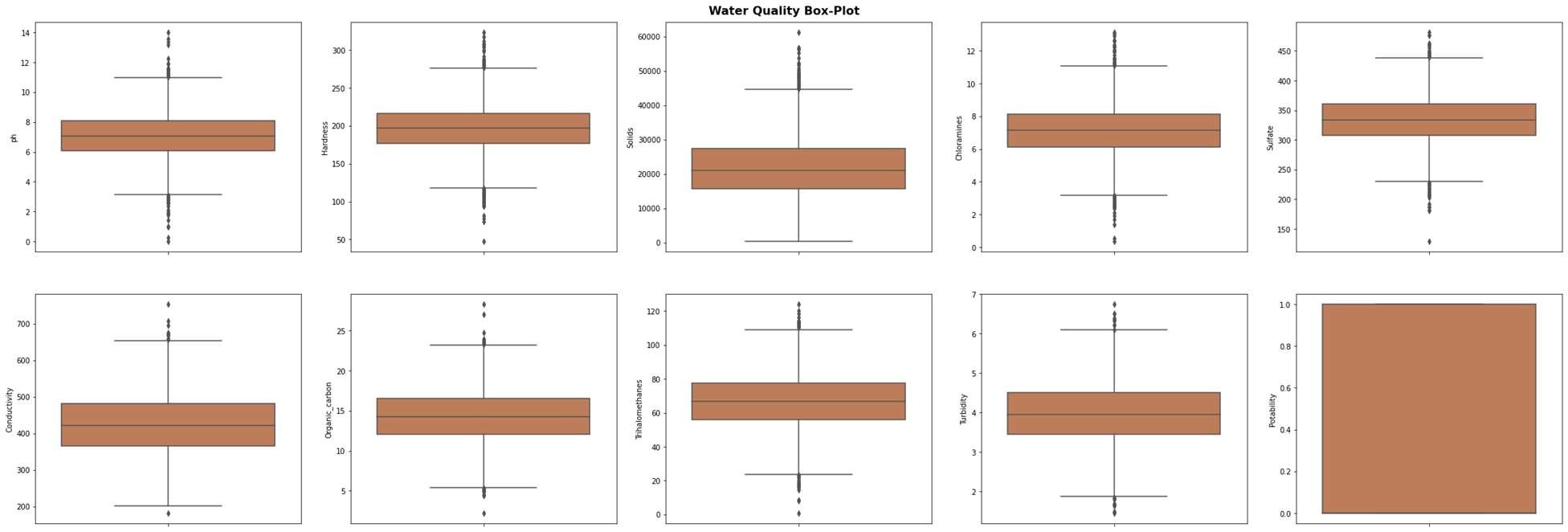


Figure 3. Boxplot of each predictor variable displaying outlier data points.

Since the dataset's distribution is primarily gaussian, as was already indicated, we can exclude outlier data outside of three standard deviations without eliminating all of the remaining outlier data (outside the maximum and minimum). This enables us to save part of the outlier data while still preserving the consistency of the dataset's variability. Again, the Solids variable showed the greatest visual change, and it also showed the greatest skewness.

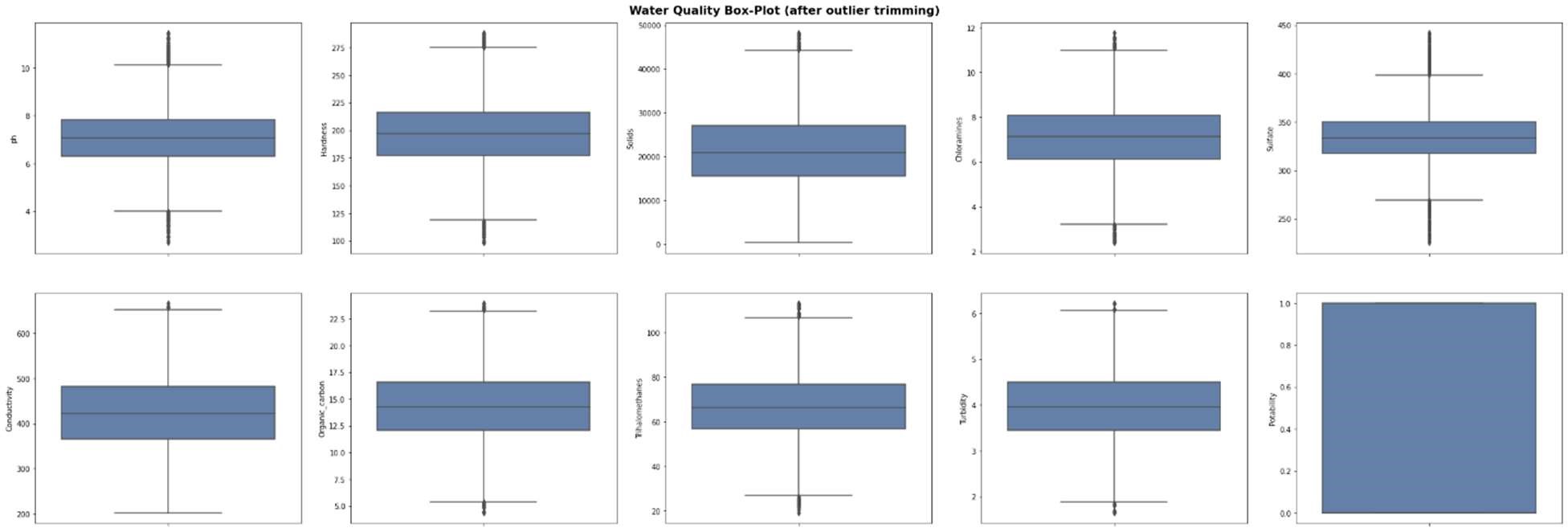
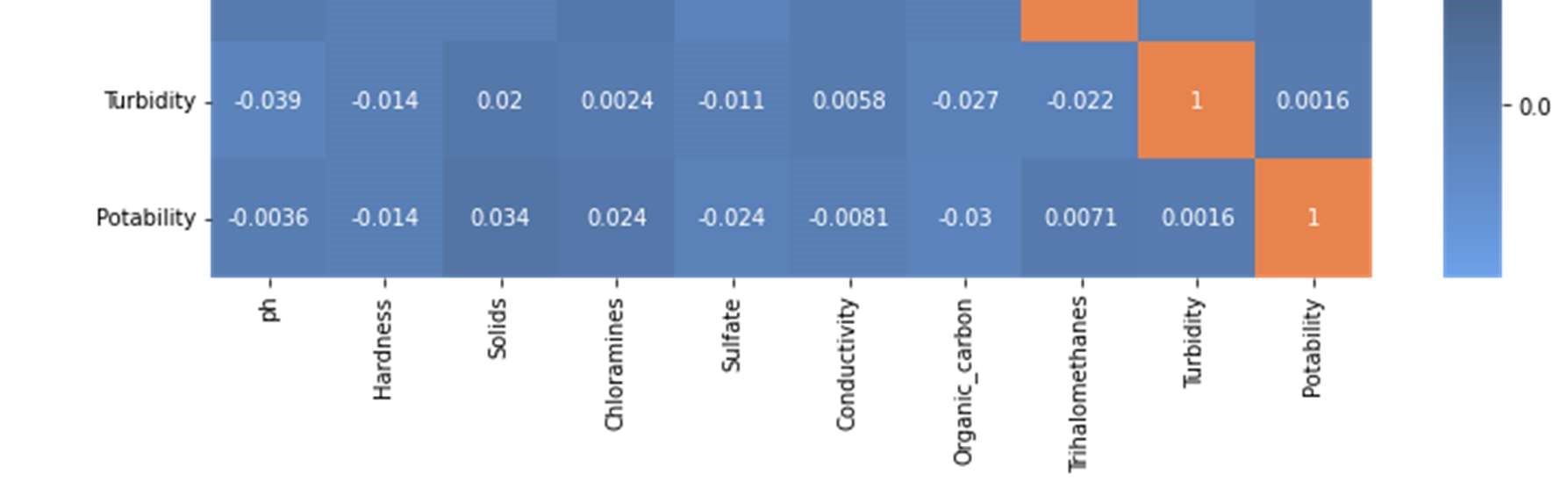
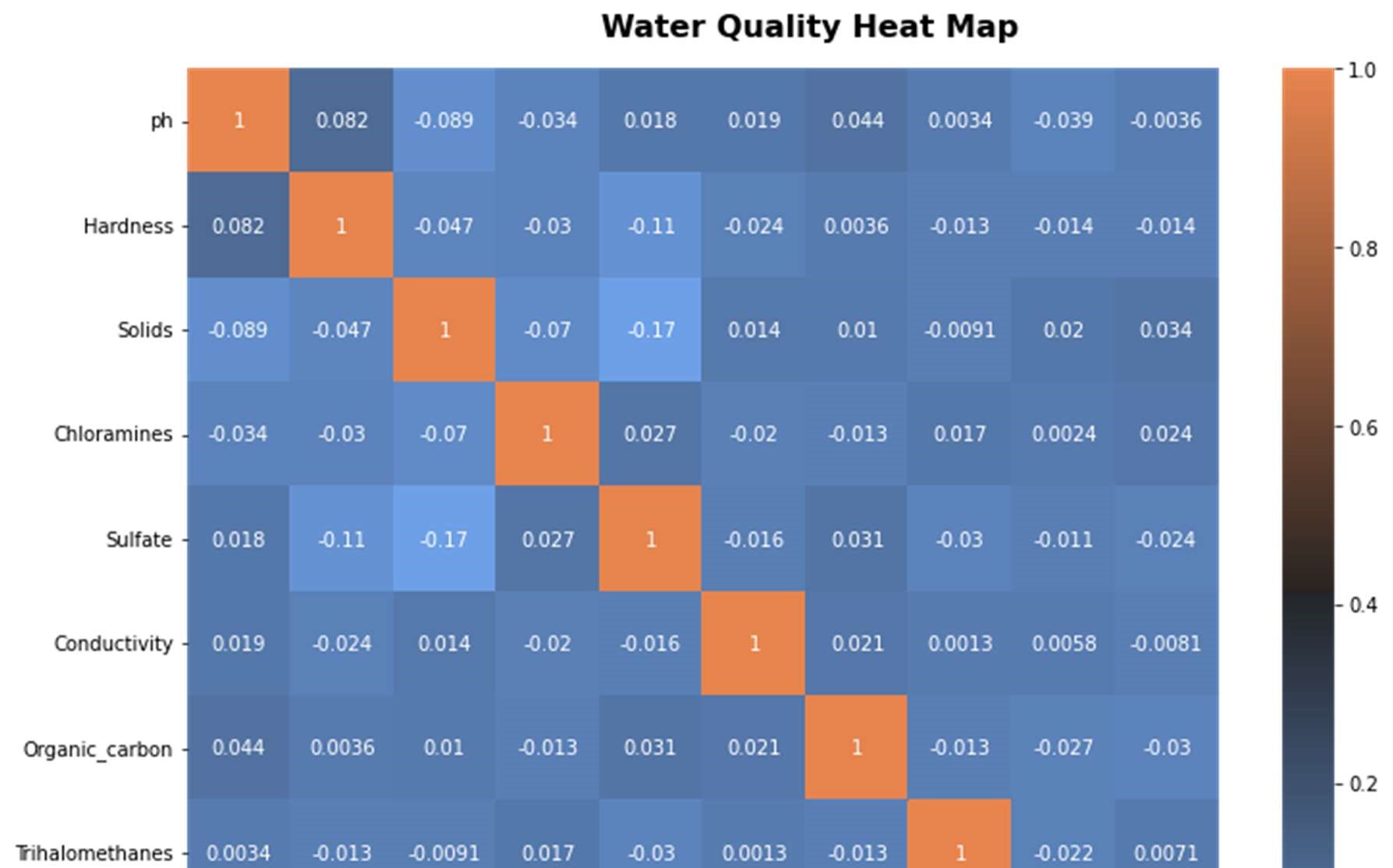


Figure 4. Box-plot of dataset after outliers greater than three standard deviations were removed.

To find out if there was any clear association between any of the predictor factors and the target variable, a correlation matrix was created and displayed using a heatmap. Unfortunately, there were no significant associations to be found because of the correlation values. Looking specifically at correlations with the target variable, Solids and Organic Carbons had the strongest correlation at +0.034 and -0.030, respectively.

Figure 5. Heatmap displaying correlation values.



Using a pair-plot to also visualize the relationships between the predictor variables, the pairwise relationships did not yield any obvious linear relationships. As such, we can rule out using a simple linear model for classification.

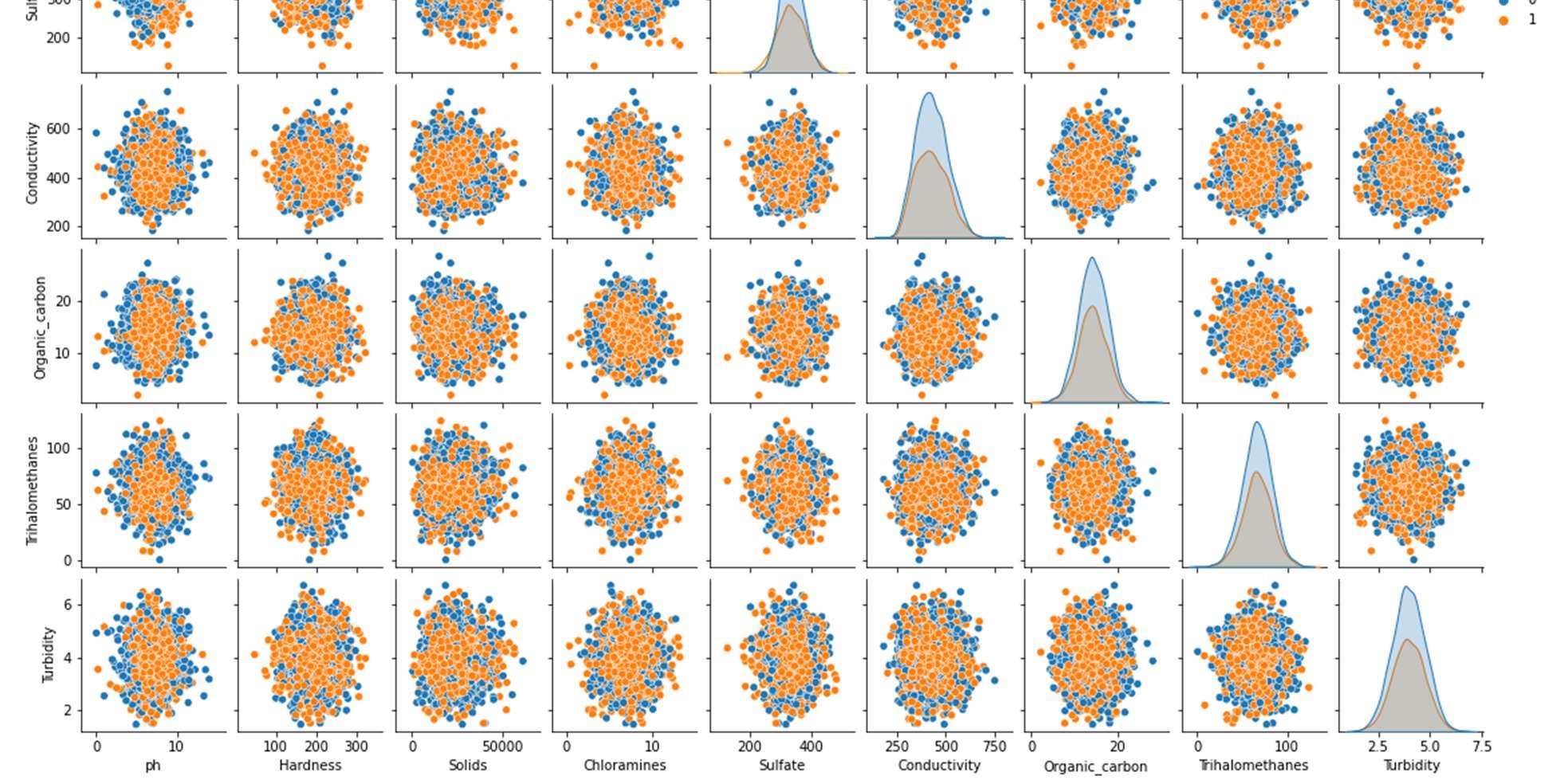
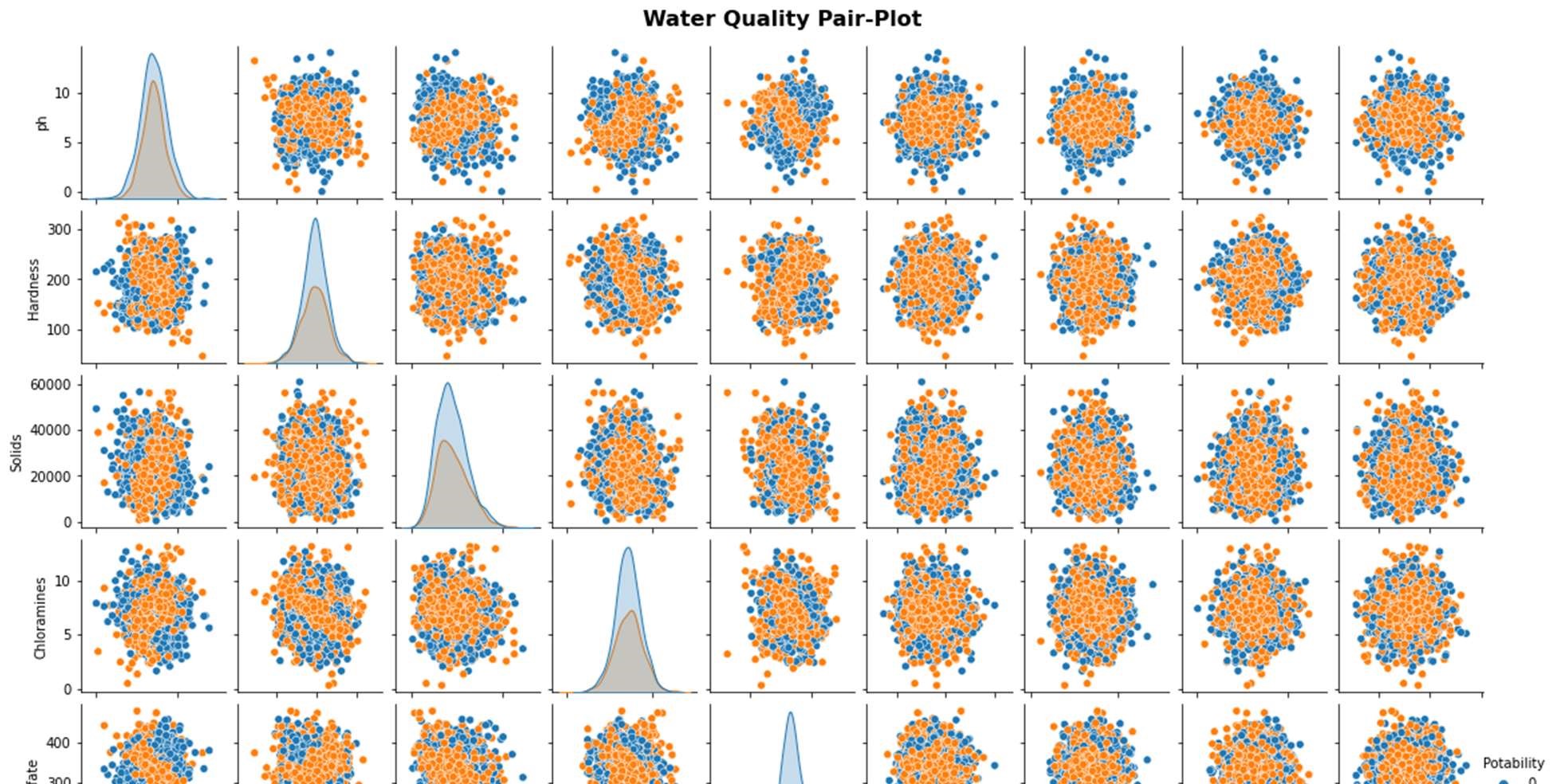
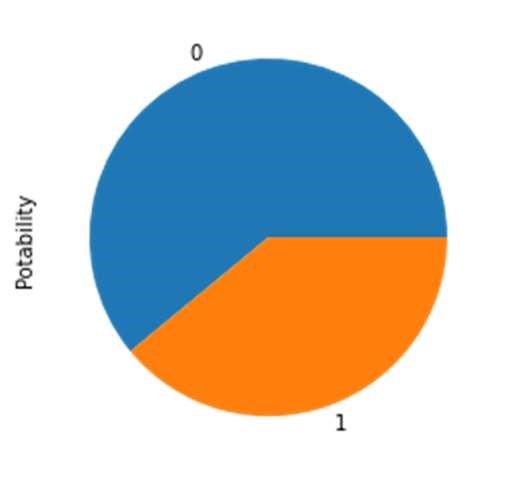


Figure 6. Pair-plot graphs of predictor variables

An imbalance of records with reference to our target variable, potability, is evident in the source data's initial analysis. 61% of the dataset's records from 1998 are non-potable. 39% of the collection, or the remaining 1278 records, are drinkable records. The categorization will be prejudiced or slanted towards the majority class in this situation, which is not drinkable, as a result of the distributional imbalance that will be troublesome during model training.



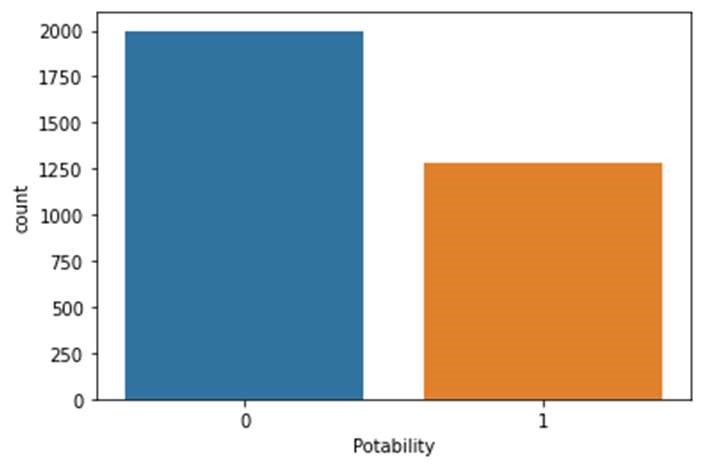
Non

-

Potable



Potable



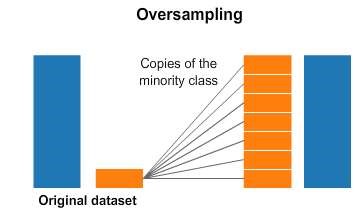
1998



1278

Figures 7 & 8. Pie chart of the record counts and bar chart of the original record counts within the dataset.

To handle this imbalance, Python’s SKlearn resample method was used to up-sample the minority class to create a balanced dataset. It essentially will over-sample the minority class by duplicating random records to create a balanced data set.



1

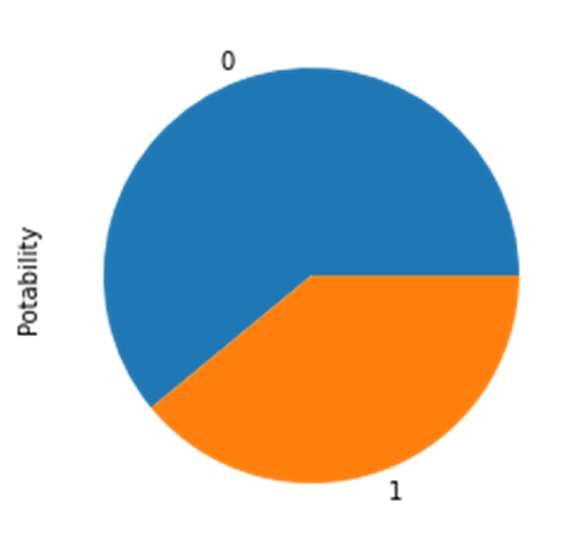
0

0

1

Figure 9. Visual representation of resampling method in SKlearn when dealing with imbalance in the dataset.

After the handling of missing data points and outlier detection through trimming, the total count of records were 1930 non-potable and 1198 potable. After implementing the resampling method, the balanced data set became 1930 non-potable and 1930 potable.



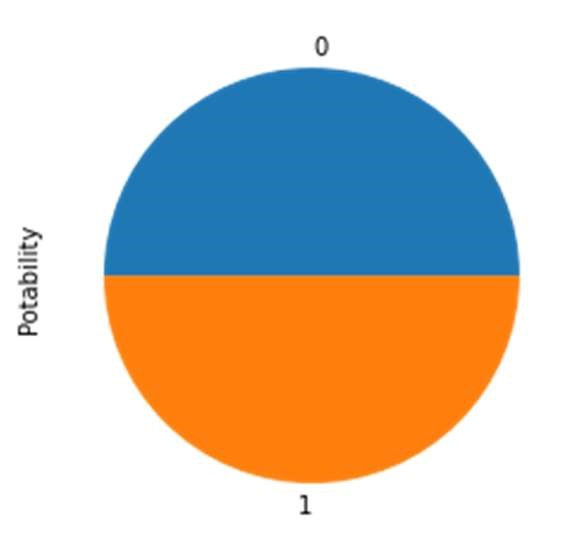
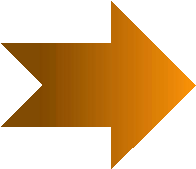
61.7

%



38.3

%



50

%



50

%

Figure 10. Pie chart of imbalanced class data to balanced class data.

Statistical Variable Analysis:

A statistical study was done to support the conclusions drawn from our visual analyses. We compared the variation between the sample means and the variation within each sample using the Analysis of Variance (ANOVA) test. A p-value was generated from the results of the ANOVA test as a comparison metric to ascertain whether there was in fact a statistical link between the nine predictor variables and the target variable. We were able to establish that none of the nine predictor variables had a significant enough correlation with the target variable using the conventional 0.05 significance as our alpha.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Predictor Variable | Correlation with Potability | P-Value |
| 0 | pH | No | 0.739300 |
| 1 | Hardness | No | 0.515201 |
| 2 | Solids | No | 0.243760 |
| 3 | Chloramines | No | 0.388958 |
| 4 | Sulfate | No | 0.812122 |
| 5 | Conductivity | No | 0.672787 |
| 6 | Organic Carbon | No | 0.295948 |
| 7 | Trihalomethanes | No | 0.680139 |
| 8 | Turbidity | No | 0.829477 |

Table 2. P-value results from ANOVA test.

In an effort to decrease the dataset's dimensions (i.e., the predictor variables), principle component analysis (PCA) was also applied as an unsupervised statistical technique. All nine predictor variables were independently determined via PCA, and at least eight components (variables) were required to account for 80% of the variation. Dimensionality reduction was therefore not advantageous for this dataset.

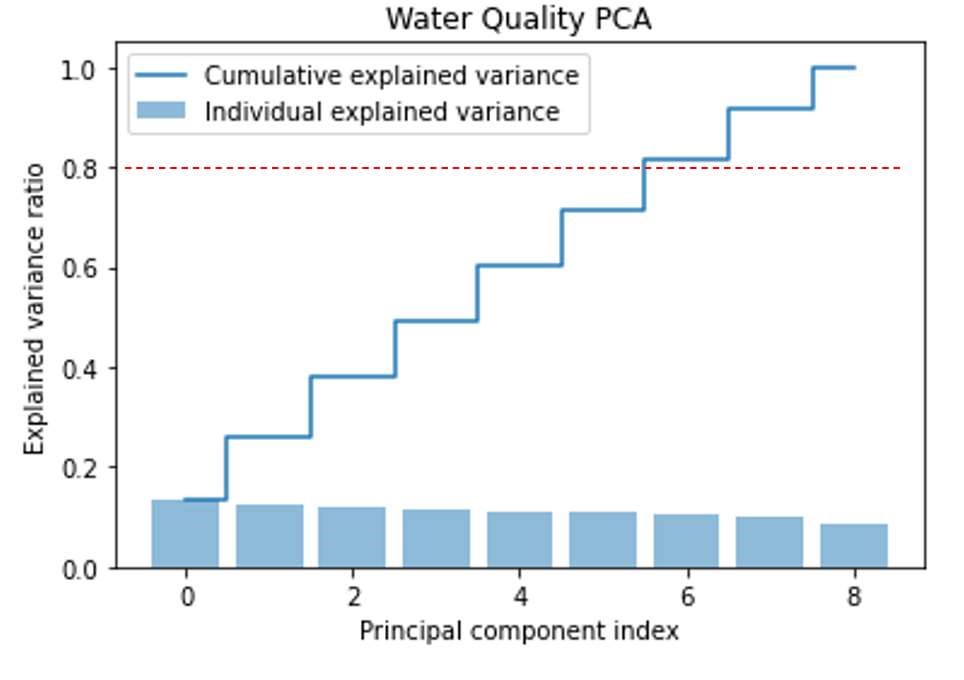


Figure 11. PCA graph depicting component variance ratio and the cumulative variance ratio.

Approach

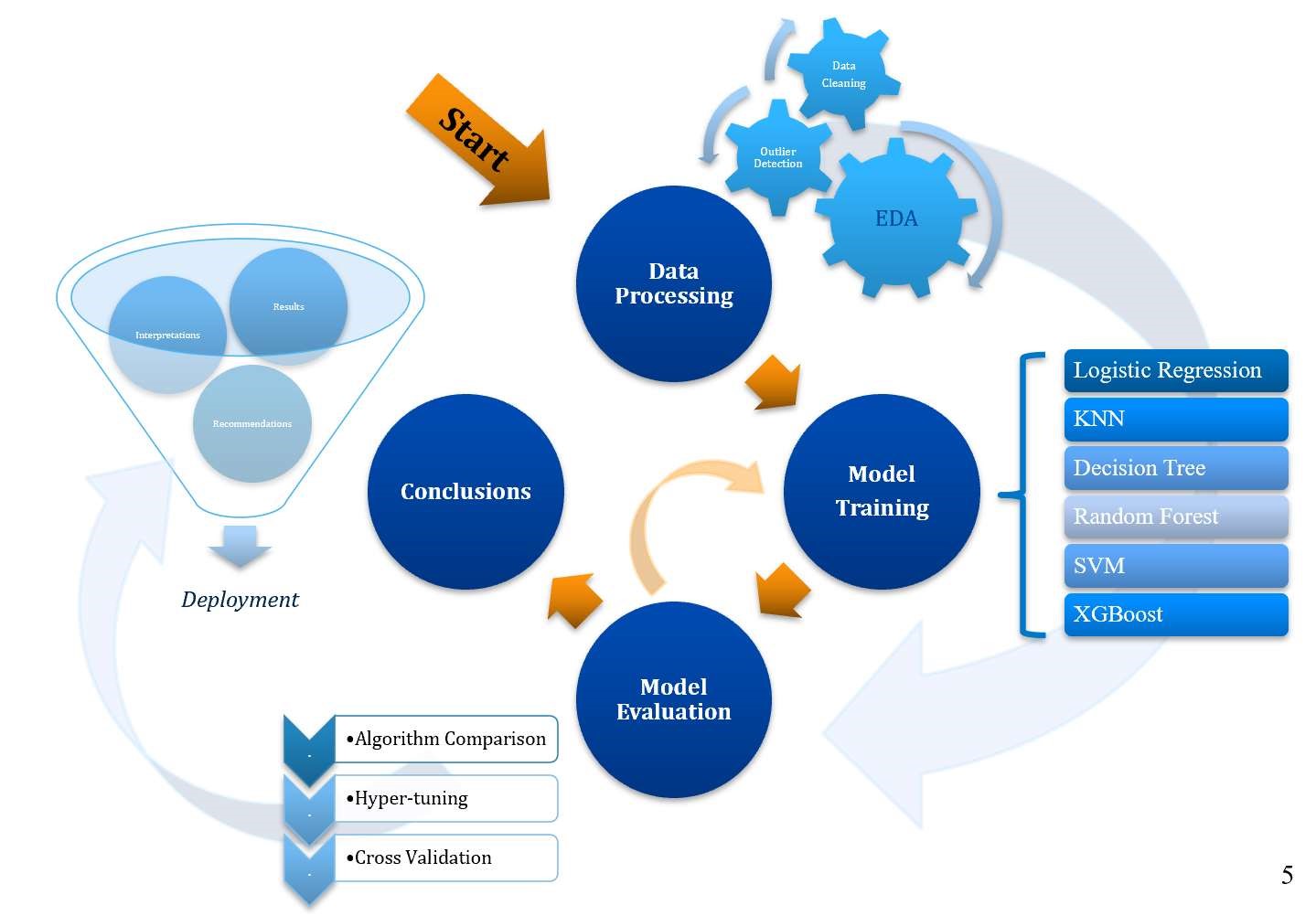


Figure 12. Approach Process

This study's methodology included four main steps. The first level of data processing involves performing the initial analyses on the raw data. Data cleansing, transformation, and source data correction are all parts of this process. and the identification of discrepancies in the records. In Python 3.7, where the data exploration and analysis are performed, the raw data is imported and maintained.

(EDA) got going. Finding usable information and producing the technically accurate data for preprocessing to become the consistent data that will be used for the remaining modeling were the objectives of EDA. Preprocessing and modeling were done using the SKlearn package after the data was determined to be consistent, with 80% of the dataset going toward training and 20% going toward testing. The dataset is then scaled to fit the data points inside the proper scale, preventing greater value ranges from dominating the calculation of data point distances. For modeling, the dataset is now prepared.

There were six methods used: support vector modeling, decision tree classifier, random forest classifier, logistic regression, and k-nearest neighbor regression. The modeling process was repeated twice, once with default parameters and once with parameters that had been hyper-tuned. We further performed cross validation on the top 3 performing models after evaluating the models and their assessment metrics.

Finally, some post-prediction analyses are discussed to finalize our results and review any recommendations with our final conclusions with the intent to deploy a robust water quality classifier.

# Modeling Algorithms

Logistic Regression

Logistic regression is a statistical method similar to linear regression that estimates the probability of a binary event occurring. Based on the given independent variables in a dataset the model predicts the probability a classification will occur using log odd ratios and an iterative maximum likelihood method (Hoffman, 2019).

K-Nearest Neighbour Regression

The k-nearest neighbor algorithm assumes that similar things exist in close proximity. It is a non-parametric classifier which uses similar data points that are close to each other to make classifications. By storing all available cases, it will classify new cases based on similarity measures by pattern recognition.

Decision Tree Classifier

A decision tree classifier is a flowchart-like structure where attributes or variables of the dataset are represented as nodes and the branches from each node represent a decision rule to another node. Eventually, a decision will be reach to a final leaf-node that will be either one outcome or the other, in this case, potable or non-potable.

Random Forest Classifier

Random Forest classifier consists of a large number of individual decision tress that operate as a group or a ‘forest’. Each individual tree will classify and return a prediction and the prediction with the most votes will become the final prediction. The fundamental concept of random forest is majority wins and is beneficial in datasets where the attributes or predictor variables have low correlation. The idea of having multiple decision trees essentially protects the forest from individual error (Yiu, 2019).

Support Vector Machine Classifier

Support Vector Machine (SVM) is a linear model for classification and creates a line or a hyperplane which separates the data into classes. The hyperplane in an n-dimensional Euclidean space is flat, n-1 dimensional subset of that space that divides it into two disconnected parts (Pupale, 2018).

XGBoost Classifier

XGBoost stands for extreme gradient boosting, and is an implementation of gradient boosted decision trees designed for speed and performance. Gradient boosting is a technique where new models are created to correct the errors made by previous models. The results are combined to make the final prediction (Brownlee, 2016).

# Results

First Iteration

In the first iteration of modeling, the algorithms were all run just the default parameters of their respective functions. The confusion matrices of the algorithms are as follows:

A picture containing text, screenshot, design

Description automatically generated A screenshot of a graph

Description automatically generated with low confidence

A picture containing text, screenshot, design

Description automatically generated A screenshot of a graph

Description automatically generated with low confidence

A screenshot of a graph

Description automatically generated with low confidence A picture containing text, screenshot, font, design

Description automatically generated

Figure 13. Confusion Matrix of each algorithm after the first iteration of modeling.

Statistically, type II errors would be considered more hazardous in particular for assessing water quality or potability. Returning a classification of a false positive would be detrimental to a community consuming unclean water. In reviewing the confusion matrices, the highest type II error occurs in Logistic Regression at 21.37% while the lowest type II error occurs in the Decision Tree algorithm at 7.12%.

After reviewing the evaluation metrics of each algorithm for first iteration of modeling, we observed that Random Forest algorithm performed the best with an accuracy of 84.33% while Logistic Regression performed the worst at 50.00%.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Precision** | **Recall** | **F1 Score** |
| Random Forest | 0.843264 | 0.838275 | 0.836022 | 0.837147 |
| Decision Tree | 0.794041 | 0.738255 | 0.887097 | 0.805861 |
| XGBoost | 0.792746 | 0.754808 | 0.844086 | 0.796954 |
| Support Vector | 0.659326 | 0.640103 | 0.669355 | 0.654402 |
| KNN Regression | 0.648964 | 0.622871 | 0.688172 | 0.653895 |
| Logistic Regression | 0.485751 | 0.470998 | 0.545699 | 0.505604 |

Table 3. Evaluation metrics from the first iteration of modeling.

A picture containing screenshot, rectangle, square, parallel

Description automatically generated Figure 14. Performance of each algorithm after the first iteration of modeling by accuracy.

Hyper-Tunning Parameters

Using hyperparameter optimization or hyper-tunning the parameters within each algorithm, the goal is to choose a most optimal set of parameters to control the algorithm to yield the best results (i.e., performance). By deploying a Grid Search approach to five of the algorithms and Random Search to just XGBoost, these functions essentially run the algorithm across all the parameters identified and report the best parameters to use. The performance of each hyper-tunning execution is also cross validated on the training set. Hyper-tunning returned the following best parameters and were then used in the second iteration of modeling:

|  |  |
| --- | --- |
| Algorithm | Hyper-tuned Parameters: |
| Logistic Regression | penalty: l1, solver: liblinear |
| KNN | algorithm: auto, n\_neighbors: 1, weights: uniform |
| Decision Tree | criterion: entropy, max\_depth: 44, min\_samples\_leaf: 1 |
| Random Forest | min\_samples\_leaf: 2, n\_estimators: 200 |
| SVM | C: 10, gamma:1, kernel: rbf |
| XGBoost | n\_estimators: 600, learning\_rate: 0.8 |

Table 4. Hyper-tunning parameters for each algorithm.

Second Iteration

In the second iteration of modeling, the algorithms were all run with the hyper-tuned parameters of their respective functions (listed in Table 4). The confusion matrices of the algorithms are as follows:

A picture containing text, screenshot, font, design

Description automatically generated A screenshot of a computer screen

Description automatically generated with low confidence

A screenshot of a computer screen

Description automatically generated with low confidence A screenshot of a computer screen

Description automatically generated with low confidence

A screenshot of a computer screen

Description automatically generated with low confidence A screenshot of a computer screen

Description automatically generated with low confidence

Figure 15. Confusion Matrix of each algorithm after the second iteration of modeling.

As mentioned in the first iteration, type II errors are significant with regards to classifying water potability. After hyper-tunning parameters we can see that Logistic Regression still has the highest false positive results at 21.76%, while the rest of the algorithms have significant decreases of type II errors after hyper-tunning.

Looking at the evaluation metrics after the second iteration we can see that hyper-tunning greatly increased the performance of nearly all the algorithms. SVM and Random Forest performed the best with the highest accuracy of 83.81% and 83.42% respectively. Logistic Regression remained the lowest performing algorithm with minute changes in overall performance.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Precision** | **Recall** | **F1 Score** |
| Random Forest | 0.848446 | 0.84 | 0.846774 | 0.843373 |
| Support Vector | 0.822539 | 0.844575 | 0.774194 | 0.807854 |
| XGBoost | 0.816062 | 0.780488 | 0.860215 | 0.818414 |
| Decision Tree | 0.787565 | 0.734234 | 0.876344 | 0.79902 |
| KNN Regression | 0.774611 | 0.722973 | 0.862903 | 0.786765 |
| Logistic Regression | 0.485751 | 0.470998 | 0.545699 | 0.505604 |

Table 5. Evaluation metrics from the second iteration of modeling.

A picture containing screenshot, rectangle, square, line

Description automatically generated

Figure 16. Performance of each algorithm after the second iteration of modeling by accuracy.

A quick comparison between the two iterations of modeling. In fact, there was a 0.26% decrease in accuracy after hyper-tunning for the Logistic Regression algorithm and 0.91% decrease in accuracy for Random Forest. SVM algorithm had the largest increase in accuracy after hyper-tunning, a difference of 17.75 %.

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **2nd Iteration** | **1st Iteration** | **Difference in Accuracy** |
| Logistic Regression | 48.58% | 48.58% | 0.00% |
| KNN Regression | 77.46% | 64.90% | 12.56% |
| Decision Tree | 78.76% | 79.40% | -0.65% |
| Random Forest | 84.84% | 84.33% | 0.52% |
| Support Vector | 82.25% | 65.93% | 16.32% |
| XGBoost | 81.61% | 79.27% | 2.33% |

Table 6. Comparison of accuracy between first and second iterations of modeling.

A screenshot of a graph

Description automatically generated with low confidence

Figure 17. Graph depicting the performance of each algorithm after the first and second iteration of modeling.

Cross Validation

After the second iteration of model training, we selected the top three algorithms to apply cross validation to. Using the K-Fold Cross-Validation method, the consistent dataset (the dataset before train-test split) was used to be split into k number of subsets, where k-1 subsets are used to train the models and the last subset is kept for validation to test the models. The scores of each fold are then averaged to evaluate the overall performance of each model. Cross-validation using 10-folds, where 9 folds were used for training and 1 used for testing, returned higher accuracy results in all three algorithms: Random Forest, SVM, and XGBoost.

|  |  |  |
| --- | --- | --- |
| Algorithm | Mean Accuracy Score | Standard Deviation |
| Random Forest | 85.65% | 1.94% |
| SVM | 88.32% | 2.22% |
| XGBoost | 81.17% | 27.70% |

Table 7. Results of K-Fold Cross Validation.

SVM Performance Results

Since SVM has the highest mean accuracy score after cross validation, we returned to the second iteration of model to produce a Receiver Operating Characteristic Curve (ROC Curve) graph to visualize the SVM model’s performance with respects to their classification threshold levels. The ROC Curve plots the True Positive Rate (recall) against the False Positive Rate (type II error). We can also calculate the area under the curve (ROC AUC) which will allow us to understand the classifier’s performance numerically as a perfect classifier is equal to 1.0. The ROC AUC for SVM after the second iteration was 0.8368 and the cross validated ROC AUC was 0.8674 which is consistent with the rest of our evaluation metrics.

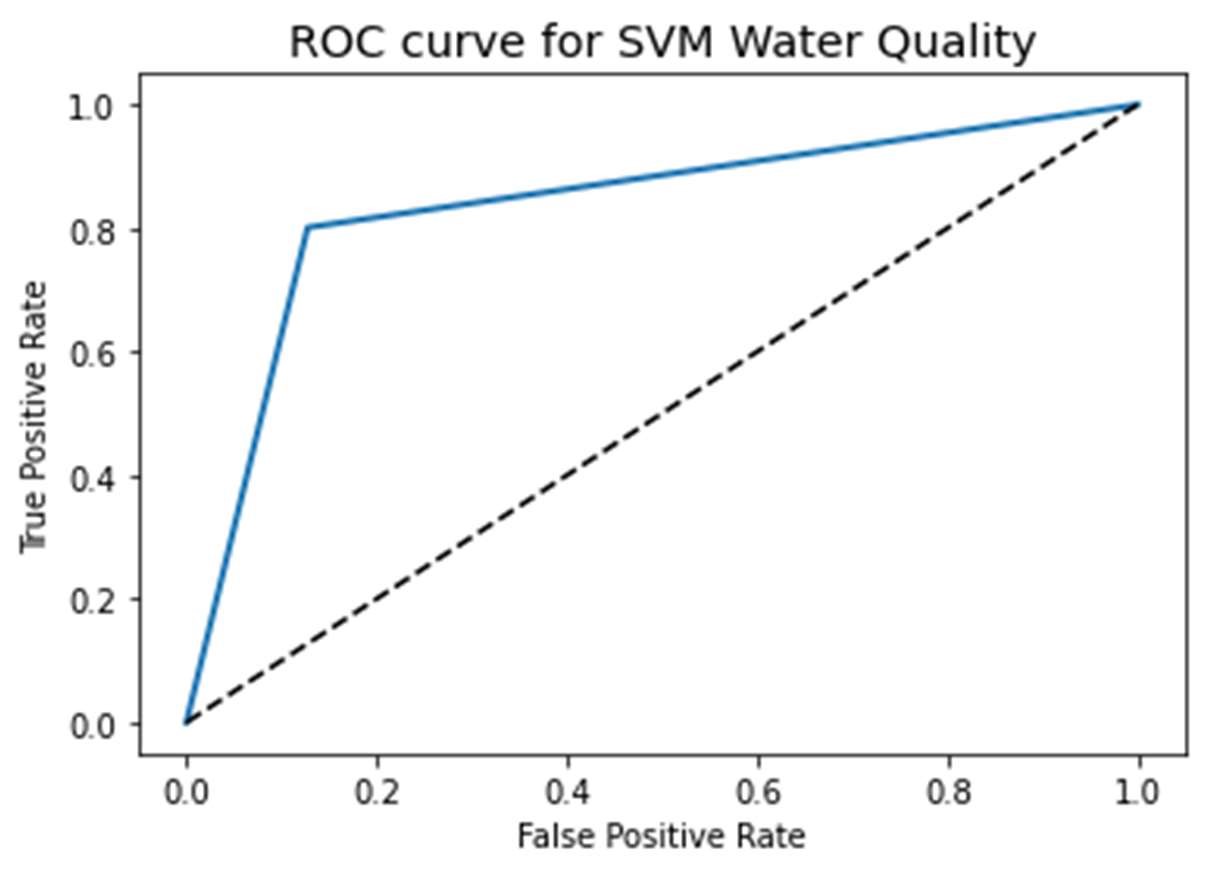


Figure 18. ROC Curve of SVM Model after second iteration.

# Conclusions

Interpretation

The results of the modeling confirm that we can classify water potability using this dataset. After the first iteration, the best performing algorithm was Random Forest with a 84.33% accuracy. SVM was the best performing model after the second iteration with hypertunned parameters of C = 10, gamma = 1, and kernel = rbf. This yielded an accuracy of 83.81%. Using cross-validation, SVM mean accuracy rose to 87.98%, which is critical in assessing safe drinking water for communities. In terms of processing time, there were no obvious delays in when executing the models as the dataset was not very large. Thus, in terms of run-time evaluations, there was not enough a difference between the all models to determine a more efficient model in terms of efficiency.

Data preparation was essential in the modeling process. Handling missing values and outliers provided a more comprehensive dataset to model with and increased the overall accuracy. Equally important was dealing with class imbalance to ensure our modeling was not skewed or biased towards the majority class of the original dataset.

Using both the train-test split for our initial modeling and then cross validating the highest performing models gave higher confidence in final selection of the algorithm best suited for water classification. Comparing our final evaluation metrics on SVM’s performance, accuracy from the first iteration was 66.06% and increased to 83.81% in the second iteration. ROC AUC values were 0.8368 and cross validated ROC AUC value was 0.8674. Final K-Fold cross validation score was 87.98%.

From literature review of previous studies conducted for water testing as well as some other general water testing research done, this particular dataset lacks, in my opinion, some critical predicting parameters like coliform or bacteria and heavy metals such as lead or copper. These particular attributes are the most noted for at-home water testing kits and for detection of water-borne diseases. These attributes do also have higher correlation with water potability which would have made initial exploratory analysis much easier.

Recommendations

Before deployment of this classifier, it would be beneficial to test against another water quality dataset with the above-mentioned additional attributes and understand what the thresholds of coliforms in our waters or heavy metals influence modeling. In the previous studies on water quality predictions, all mentioned pH, hardness, solids, sulfates, conductivity and turbidity as feature attributes and at least bacteria or coliform. As we know, fecal coliform or E. coli can have significant consequences to consumers which should be an important predictor variable in all water classification studies. Another recommendation would to have location of water samples indicated such as treatment facility or open waters. We have learned from the literature review that water samples taken where water sources are active can affect sampling as opposed to sampling from more contained systems like treatment facilities. And finally, explore machine learning with Artificial Neural Network (ANN) which had accuracy results in the 90% range in previous studies.

Conclusions

Water classification can be improved using machine learning algorithms and can be accomplished to a high accuracy. Using Support Vector Machine Classifier yielded the best performance overall at 87.98% accuracy after hyper-tunning the algorithm parameters. While we were able to effectively classify water with the predictor variables in the dataset, a few other critical features should be included in the future before deployment such as coliform values and heavy metals as well as explore more advanced deep machine learning algorithms.

# References

1. Kadiwal, A. (2021). Water Quality Dataset Version 3. Retrieved from https://www.kaggle.com/datasets/adityakadiwal/water-potability/
2. Government of Canada. Ending long-term drinking water advisories. Retrieved from https://www.sac-isc.gc.ca/eng/1506514143353/1533317130660
3. Public Health Ontario. Drinking Water Testing – Private Citizen. Retrieved from https://www.publichealthontario.ca/en/laboratory-services/test-information-index/drinking-watertesting-private-citizen
4. Government of Canada. (2015) Maintaining water quality and availability. Retrieved from https://www.canada.ca/en/environment-climate-change/services/archive/sustainabledevelopment/2015-progress-report/water-quality-availability.html
5. World Health Organization. (2022). Drinking-water quality guidelines. Retrieved from https://www.who.int/publications/i/item/9789240045064
6. Ahmed, U., Mumtaz, R., Anwar, H., Shah, A. A., Ifran, R., & Garcia-Nieto, J. (2019). Efficient Water Quality Prediction Using Supervised Machine Learning. www.mdpi.com/journal/water. Retrieved May 10, 2022.
7. Yogalakshmi, S. & Mahalakshmi, A. (2021). Efficient Water Quality Prediction for Indian Reivers

Using Machine Learning. www.ajast.net. Retrieved May 31, 2022.

1. Sakizadeh, M. Artificial intelligence for the prediction of water quality index in groundwater systems.

(2016). https://link.springer.com. Retrieved May 10, 2022.

1. Fernandez del Castillo, A., Yebra-Montes, C., Garibay, M. V., de Anda, J., Garcia-Gonzalez, A. & Gradilla-Hernandez, M. S. (2022). Simple Prediction of an Ecosystem-Specific Water Quality Index and Water Quality Classification of a Highly Polluted River through Supervised Machine Learning. www.mdpi.com/journal/water. Retrieved May 22, 2022.
2. Ubah, J. I., Orakwe, L. C., Ogbu, K. N., Awu, J. I., Ahaneku, I. E. & Chukwuma, E. C. (2021).

Forecasting water quality parameters using artificial neural network for irrigation purposes.

www.nature.com/scientificreports. Retrieved May 31, 2022.

1. Tan, G., Yan, J., Gao, C. & Yang, S. (2012). Prediction of water quality time series data based on least squares support vector machine. www.elsevier.com/locate/procedia. Retrieved May 31, 2022.
2. Meride, Y. & Ayenew, B. (2016). Drinking water quality assessment and its effects on residents’ health in Wondo genet campus, Ethiopia. https://environmentalsystemsresearch.springeropen.com.

Retrieved May 31, 2022.

1. Hoffman, J. (2019). Basic Biostatistics for Medical and Biomedical Practitioners (2nd Edition), https://www.sciencedirect.com/topics/medicine-and-dentistry/logistic-regression-analysis. Retrieved July 7, 2022.
2. Yiu, T. (2019). Understanding Random Forest: How the Algorithm Works and Why it is so Effective.

https://towardsdatascience.com/understanding-random-forest-58381e0602d2. Retrieved July 7, 2022.

1. Pupale, R. (2018). Support Vector Machines (SVM) – An Overview.

https://towardsdatascience.com/https-medium-com-pupalerushikesh-svm-f4b42800e989. Retrieved Jul 10, 2022.

1. Brownlee, J. (2016). A Gentle Introduction to XGBoost for Applied Machine Learning.

https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/.

Retrieved July 10, 2022.