Hydroinformatics: A Machine and Deep Learning Paradigm for Water Potability Assurance

Kamryn Robertson
MATH 5310: Deep Learning
May 9, 2024

Abstract

An essential and basic human right is having access to safe drinking water. Globally, locally, and regionally, this necessity remains important as a health issue. Studies show that in some regions, investing into the water supply and its sanitation can increase the economic flow as the health effects of non-potable water increase health effects, care and cost as opposed to not intervening. This binary classification problem is crucial because access to clean and safe drinking water is a fundamental human right, yet many regions face challenges with water contamination, which can lead to severe health issues when consumed. The problem this study aims to solve is the assurance of water potability, further ensuring that the water meets standards for consumption based on the metrics within the data utilizing machine learning and modeling.

1 Background

Water potability in essence is referring to the ability of a body of water's use for human consumption that will not cause harm. Potable water has to meet a certain level of quality and safety that is presumably set by regulators to guarantee that what we consume is rid of any hazardous contaminants. Some of these standards that need to be met can include pH level, turbidity, pollutant levels, and safeguarding various other substances that could arise as risks of human and animal health if the water were to be consumed. Ultimately, we break down the need to continuously address the issue of water potability due to following reasons:

a. Health Concerns: Non Potable water can contain hazardous chemicals, pathogens, and pollutants, which can lead to waterborne diseases such as dysentery, cholera, and typhoid.
b. Environmental Impact: Biodiversity and even ecosystem imbalance can occur due to the pollutants of water from various sources. This not only affects us as humans but also aquatic life.
c. Societal Equity: Regardless of geographical location of socio-economic background, potable water is essential for everyone. Addressing the problem of water potability further ensures equality in access to the basic necessity of clean drinking water.
d. Economic Implications: Various economic hardships in the healthcare system like medical expense increase, loss of productivity, and an increase of infrastructure cost for the treatment and sanitation of water could be onset by the waterborne diseases.

Machine Learning can effectively address the aspect of predicting water potability based on some of the previously stated water quality parameters. By analyzing the water potability dataset, which contains information on parameters such as pH value, hardness, total dissolved solids, and presence of contaminants like sulfate and chloramines, the chosen model can learn to classify the water samples in the data as potable or non-potable.To demonstrate this, we recreated an existing exploratory data analysis and performed a prediction model comparison using Python in Google Collab on the publicly available water potability dataset. Additionally, we investigated the limitations of the already existing data analysis in Kaggle, identified the impacts of trying different ensemble methods, model stacking, and feature selection in order to reduce the dimensionality of the data. In the final analysis, we found that we were able to recreate the original analysis, identified major trends in the features of the data, and visualized our findings in a comprehensible manner.

2 Data Source

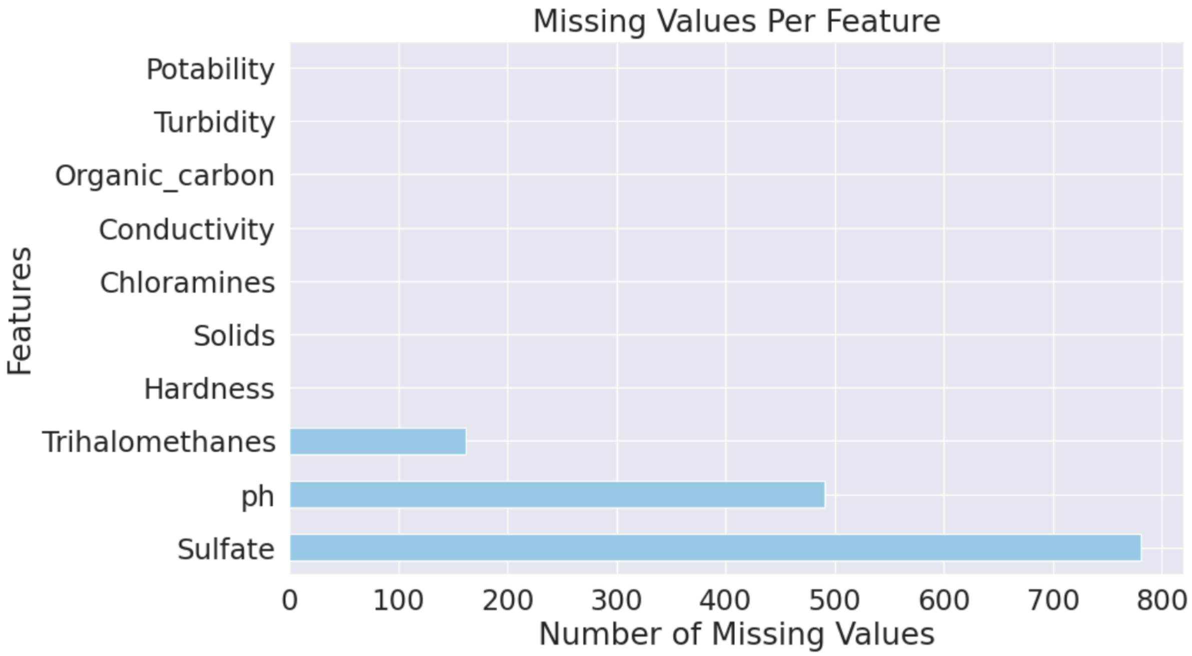
The Water Potability dataset, which is a public domain file, contains water quality metrics for 3276 different water bodies. The columns of the dataset represent the following parameters that must be met in order to deem a body of water potable:

1. pH value: The Safe Drinking Water Foundation mandates that the pH level should be between 6.5 and 8.5, per the Secondary Maximum Contaminant Level. [3]

2. Hardness: The dataset quantifies hardness based on its ability to precipitate soap, which is measured in mg/L. [3]
3. Solids (Total dissolved solids - TDS): TDS limits for drinkable water is recommended at 500 mg/l and a maximum of 1000 mg/l. [3]
4. Chloramines: Chlorine levels are considered safe up to 4 mg/L or 4 ppm. [3]
5. Sulfate: Sulfate concentration in seawater approximates to 2,700 mg/L and sulfate concentration in freshwater can range from 3 to 30 mg/L. It is also not unusual for other regions to contain up to 1000 mg/L. [3]
6. Conductivity: The electrical conductivity value can not be more than 400 μS/cm, according to WHO guidelines. [3]
7. Organic_carbon: TOC levels can not be more than 2 mg/L in treated water and below 4 mg/L in source water by US EPA standards. [3]
8. Trihalomethanes (THMs): THM concentrations that are up to 80 ppm are considered acceptable for consumption. [3]
9. Turbidity: By WHO guidelines, the average turbidity threshold is 5.00 NTU. [3]
10. Potability: Our target variable with assesses the level of consumption where 1 signifies potable and 0 signifies nonpotable. [3]

3 Treatment

We executed a series of foundational exploratory queries to determine the current state of the data and determined the trihalomethanes, ph, and sulfate columns contained missing values. However, as we aim to recreate the existing Kaggle analysis, we find that the author did not handle this problem in their early analysis. We noted that this might be why they received their prediction results. We then decided to impute the missing values for 'Sulfate', 'ph', and 'Trihalomethanes' with the mean of each respective feature. Using the fillna() method with inplace=True, we ensured that the changes were applied directly to the original dataset. After these changes were made we generate the following graph that verifies the amount of missing values for the sulfate,ph, and trihalomethanes are 781, 491, and 162 respectively:

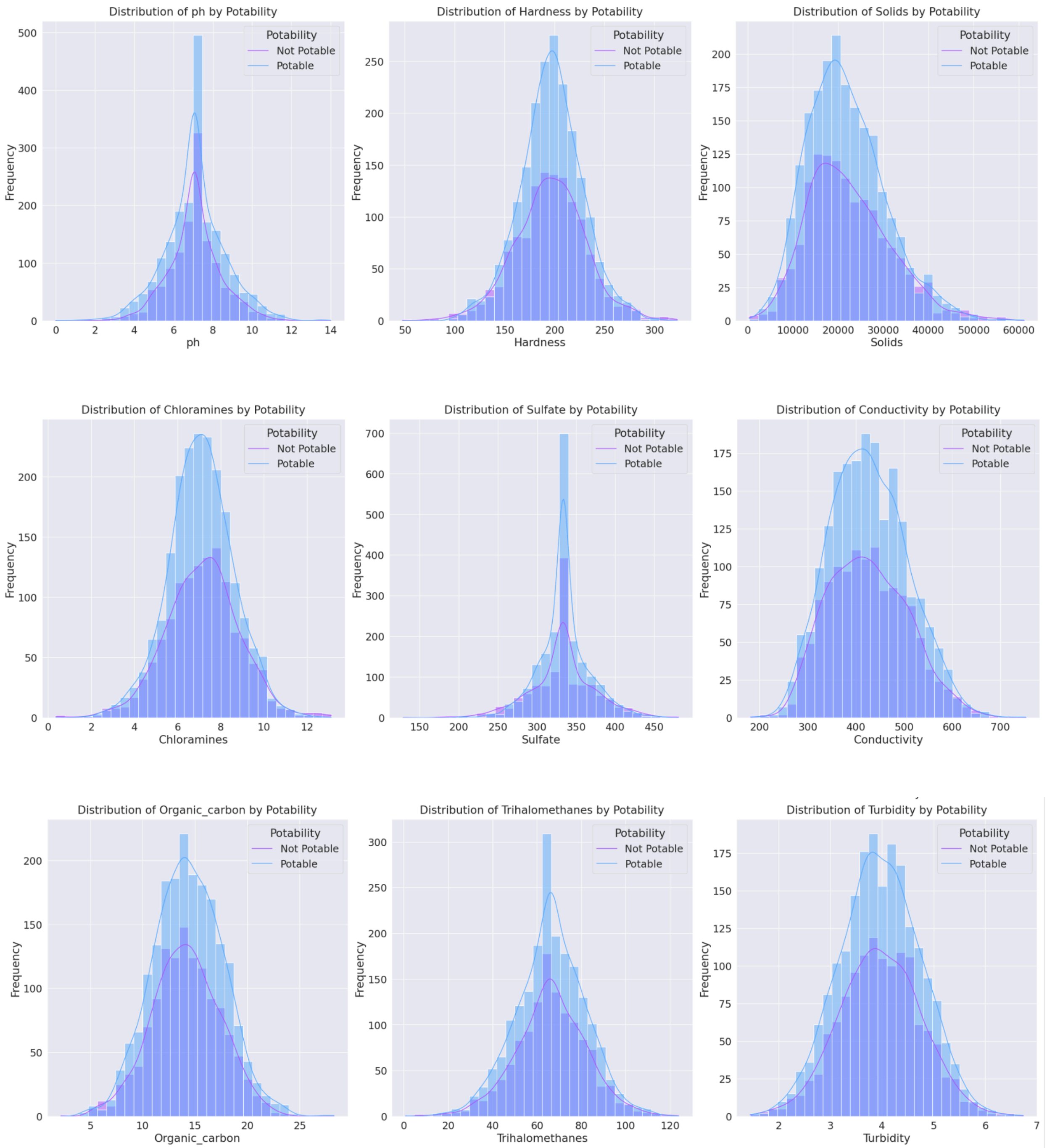


4 Exploratory Data Analysis

For this portion of our project, we analyzed the outcomes of a series of queries within the main focus of predicting the potability of the various bodies of water in the dataset. This was done utilizing feature correlation and selection, comparing Random Forest, AdaBoost, and Support Vector Machine classifiers, and analyzing the performance of model stacking. We follow up each analysis with visualization and explanation.

4.1 Distribution of Features

To begin, we analyzed the distribution of each feature in respect to potability. In order to visualize this, we generate the following histograms for each feature. We see that in each histogram the hue parameters are distinguished between potable (1) and non-potable (0) water:


Distribution of ph by Potability

Distribution of Hardness by Potability

Distribution of Solids by Potability

Distribution of Chloramines by Potability

Distribution of Sulfate by Potability

Distribution of Conductivity by Potability

Distribution of Organic_carbon by Potability

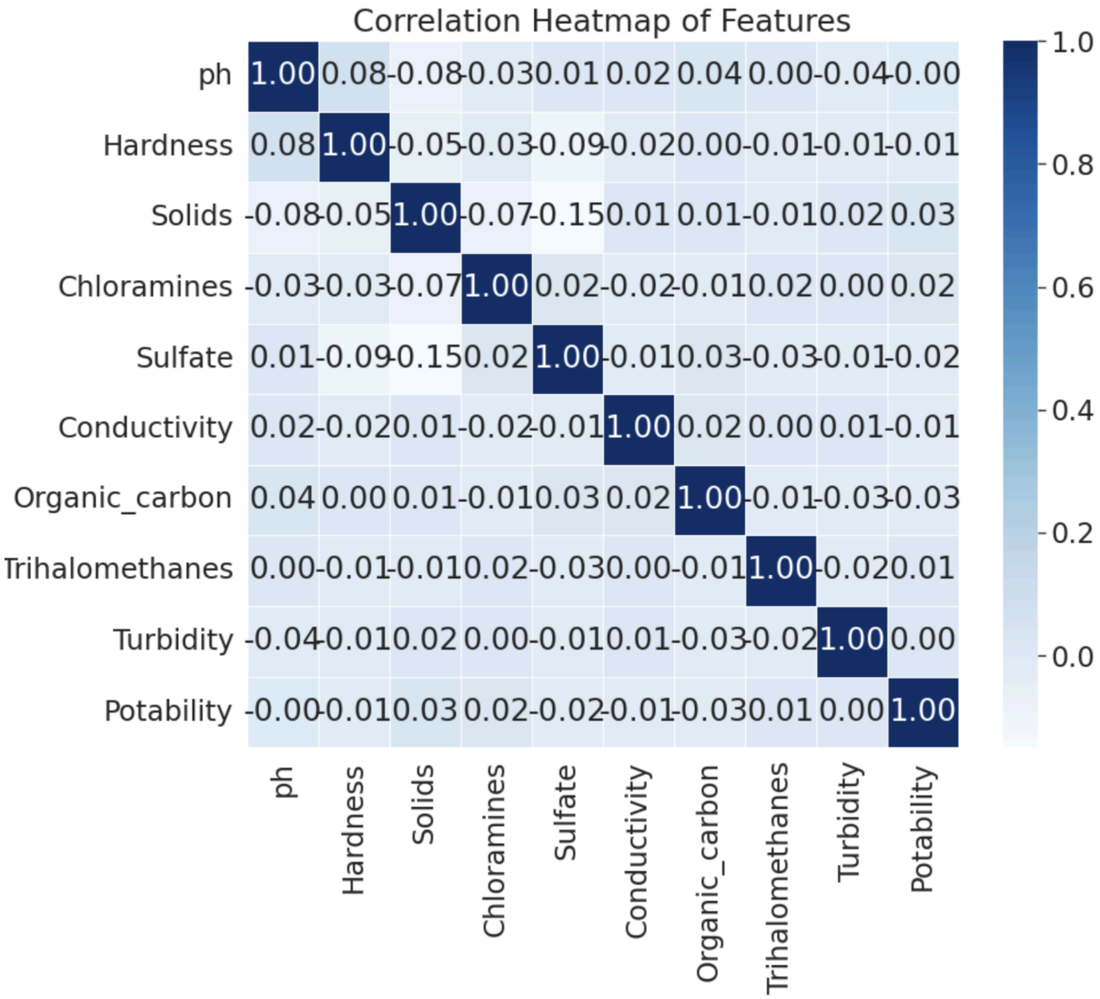
Distribution of Trihalomethanes by Potability

Distribution of Turbidity by Potability

As you can see, the organic carbon, pH, and turbidity histograms are the most symmetrically distributed around the mean. Which we later assess in our feature selection.

4.2 Correlation Coefficient Between Pair of Features

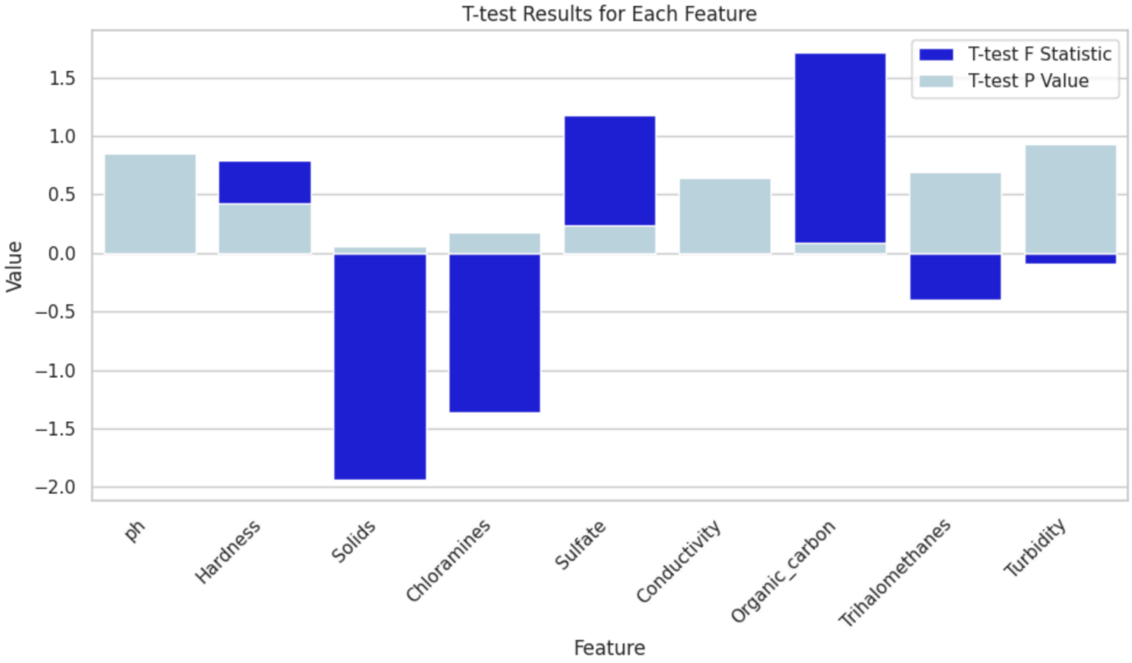
With the findings of our histograms, we then decided to check the correlation between each feature. We generated the following heatmap to show the correlation coefficients between each pair of features in the dataset:



Since we have 1.00 across the diagonal, this indicates that each feature in the dataset is perfectly correlated with itself and suggests that there is a strong positive correlation. This is due to our binary problem and scale of potability being either 0 or 1 which is normally distributed across the dataset.

4.3 T-Tests of Features

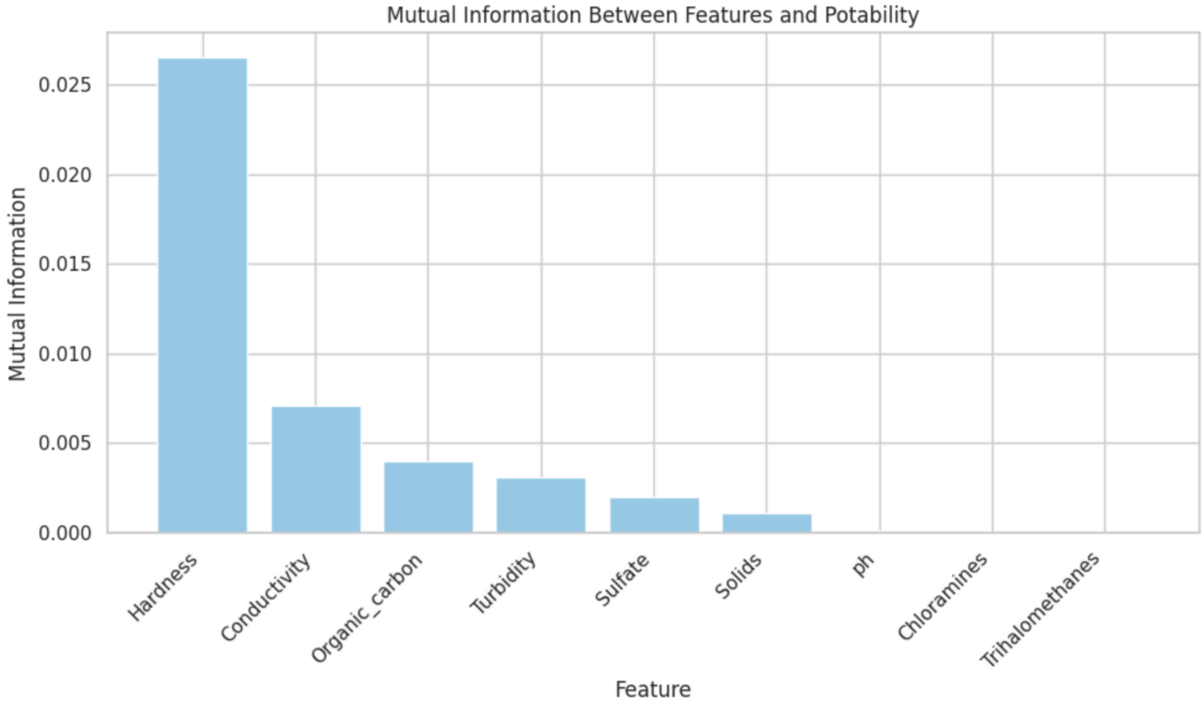
Since we also know that 1.00 across the diagonal can be a result of normalization or standardization, we conducted independent t-tests for each feature in our dataset to compare the means between two groups. For each feature, excluding our target variable potability, we calculated the t-statistic and p-value using the ttest_ind function from SciPy's stats module. We generated the following bar plot to represent these calculations:



We note that organic carbon and solids are the only features whose p-values fall below our predefined alpha.

4.4 Mutual Information Between Features Potability

As an additional metric, we also analyzed the mutual information between each feature and potability. This was done as a means to find any connection between the distribution of our features and the distribution of potability which is Bernoulli. We generated the following plot:



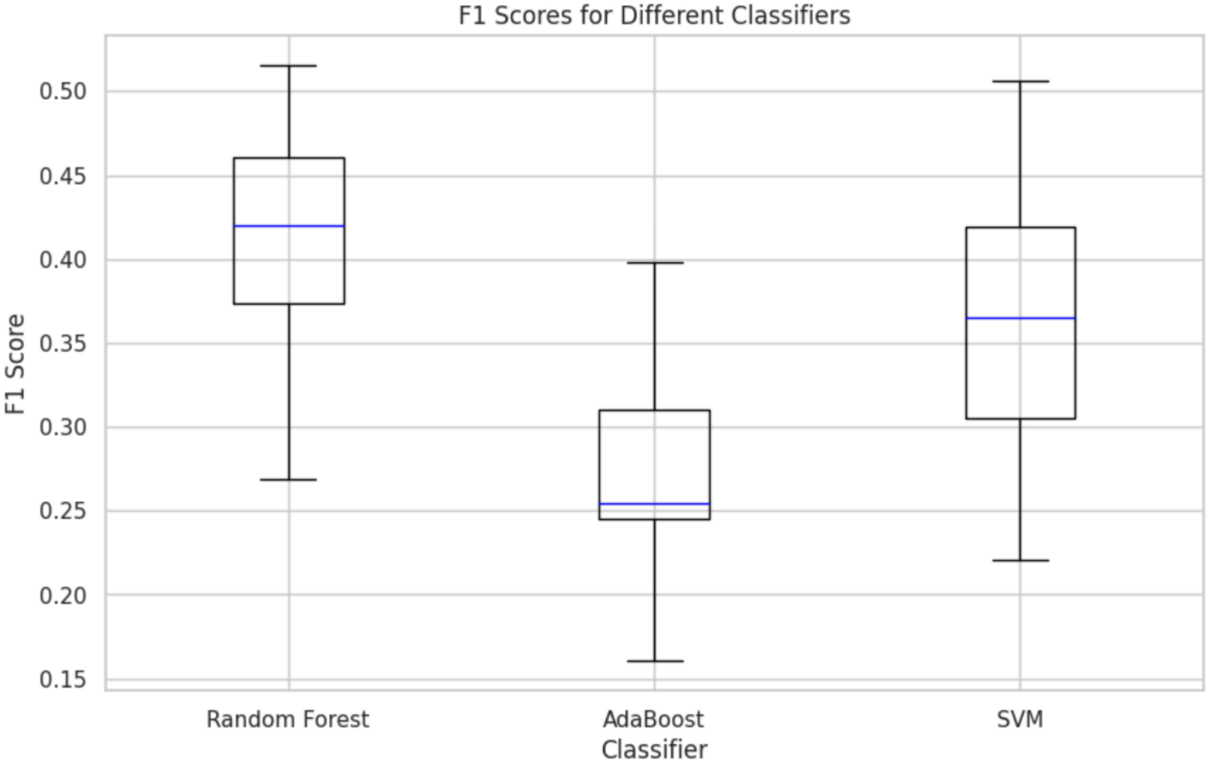
We notice that in comparison to our t-test scores, the features with low t-test scores have high mutual information with potability. We assume also pH, Chloramines, and THMs do not give us any information on their contribution to a body of water's potability. However, the remaining features indicate a strong relation with potability and include this in modeling.

5 Top Model Comparison

As stated earlier, some features we tested during exploratory data analysis showed evidence of being uniquely correlated to potability. As a result of this, we selected the top three features with the lowest p-values from the results of our t-tests and resulted in selecting turbidity as our chosen feature in our baseline modeling. We split the dataset into training and testing sets and used the other lowest features as input features for the model while preserving potability as our target.

5.1 Baseline Model Evaluation

We began our modeling by setting up pipelines for Random Forest, AdaBoost, and Support Vector Machine classifiers. Each classifier consisted of a StandardScaler for feature scaling. We then conducted a 10-fold cross-validation for each classifier to evaluate their performance based on F1 score. We visualized these results with the following box plot with each box representing the interquartile range (IQR) of F1 scores for each classifier:



From these results, we see that the median line in both the Random Forest and SVM show to be fair.

5.2 Hyperparameter Tuning

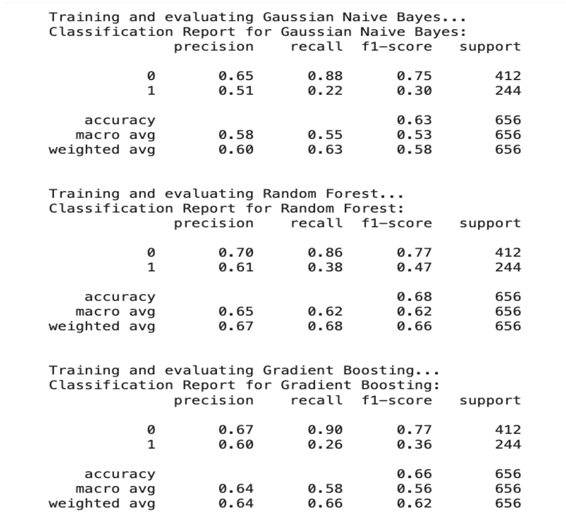
In order to get more accurate results we then performed a grid search with cross-validation to fund the best hyperparameters for the AdaBoost classifier with a RandomForest base estimator. After fitting the grid search object to the data, we found an output of the best parameters found during the search along with the corresponding cross-validated score of 0.632. With these best parameters we found from the optimization process, we defined the Random Forest base estimator with specified parameters. Finally, we fit the pipeline to the dataset, trained the AdaBoost classifier on the features to predict potability and got an overall 62% prediction rate.

We observed this to be fair results and a moderate balance between precision and recall but believe there is room for improvement in performance.

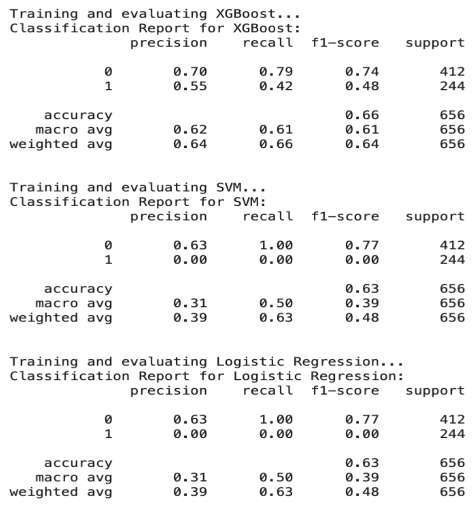
5.3 Model Stacking

Since we were only recreating a portion of the Kaggle notebook, we decided to further the author’s analysis to increase prediction performance by exploring this additional analysis and model comparison. We originally tried to perform various ensemble methods which included Gradient Boosting or XGBoost but these methods provided a performance that was lower than 62% and opted to choose another route to increase performance.

We then decided to combine the predictions from multiple models to create a more robust and accurate prediction. We first imported several classifiers from scikit-learn and XGBoost and then created a dictionary of these models which contained instances of these classifiers. We split the data into training and testing sets again and iterated over each model in the dictionary. We finally were provided with the following metrics of each models performance:

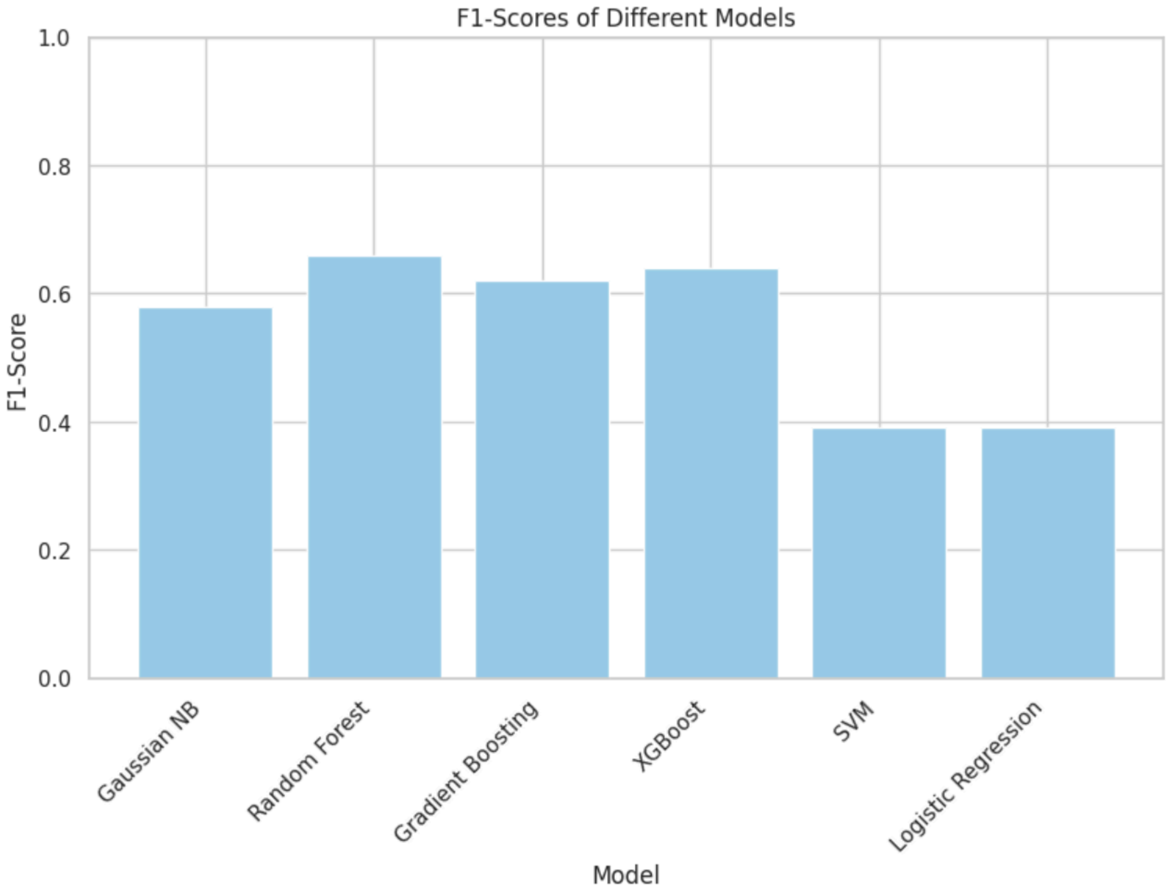


```
Training and evaluating Gaussian Naive Bayes...
Classification Report for Gaussian Naive Bayes:
              precision    recall  f1-score   support

           0       0.65      0.88      0.75       412
           1       0.51      0.22      0.30       244

    accuracy                           0.63       656
   macro avg       0.58      0.55      0.53       656
weighted avg       0.60      0.63      0.58       656

Training and evaluating Random Forest...
Classification Report for Random Forest:
              precision    recall  f1-score   support

           0       0.70      0.86      0.77       412
           1       0.61      0.38      0.47       244

    accuracy                           0.68       656
   macro avg       0.65      0.62      0.62       656
weighted avg       0.67      0.68      0.66       656

Training and evaluating Gradient Boosting...
Classification Report for Gradient Boosting:
              precision    recall  f1-score   support

           0       0.67      0.90      0.77       412
           1       0.60      0.26      0.36       244

    accuracy                           0.66       656
   macro avg       0.64      0.58      0.56       656
weighted avg       0.64      0.66      0.62       656
```



```
Training and evaluating XGBoost...
Classification Report for XGBoost:
              precision    recall  f1-score   support

           0       0.70      0.79      0.74       412
           1       0.55      0.42      0.48       244

    accuracy                           0.66       656
   macro avg       0.62      0.61      0.61       656
weighted avg       0.64      0.66      0.64       656

Training and evaluating SVM...
Classification Report for SVM:
              precision    recall  f1-score   support

           0       0.63      1.00      0.77       412
           1       0.00      0.00      0.00       244

    accuracy                           0.63       656
   macro avg       0.31      0.50      0.39       656
weighted avg       0.39      0.63      0.48       656

Training and evaluating Logistic Regression...
Classification Report for Logistic Regression:
              precision    recall  f1-score   support

           0       0.63      1.00      0.77       412
           1       0.00      0.00      0.00       244

    accuracy                           0.63       656
   macro avg       0.31      0.50      0.39       656
weighted avg       0.39      0.63      0.48       656
```

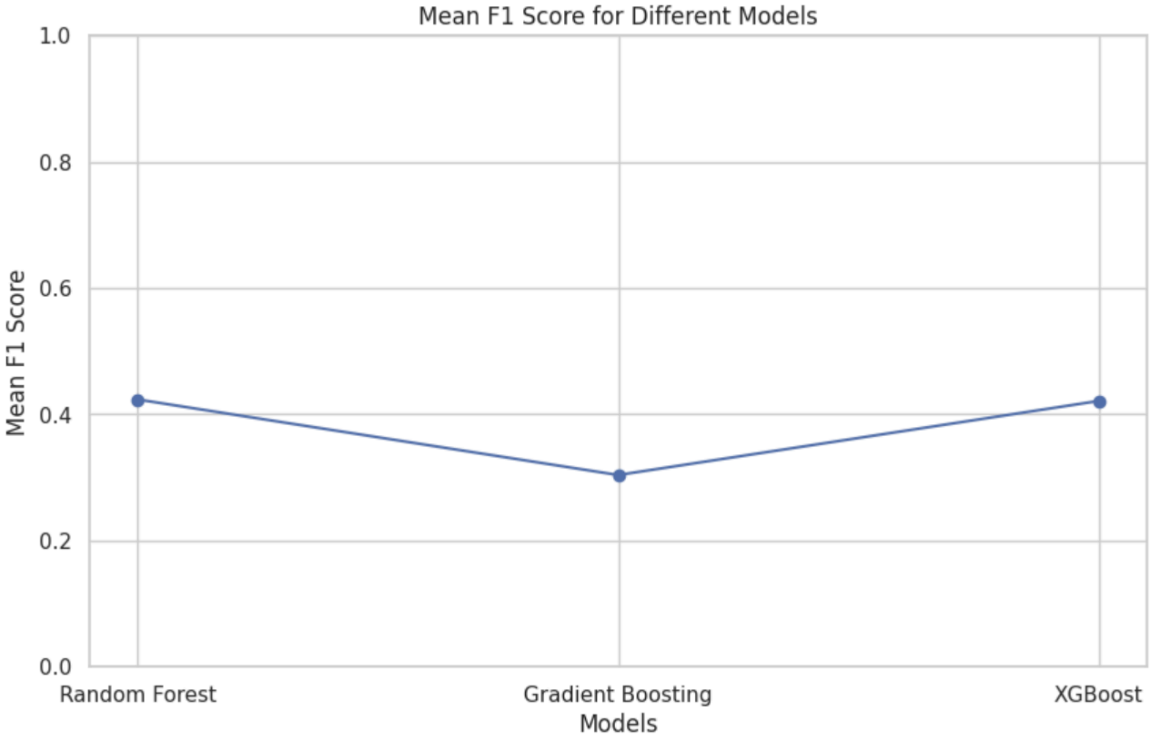
We then generated the following bar plot to visualize the F1 scores of each model:



We observe that a majority of the models were around 60% accurate in predicting potability.

5.4 Feature Selection

Since our models averaged around 60% accuracy, as a final means to increase the accuracy we performed feature selection to reduce the dimensionality of our data. We decided to fine tune the hyperparameters of the Random Forest, Gradient Boosting, and XGBoost models since their accuracies were the highest in our model stacking. We selected the top k features with the highest mutual information scores then trained and tuned the models using only those selected features. These were pH, hardness, sulfate, conductivity, and organic carbon. We evaluated the performance of each model and generated the following graph along with the metrics:



For Random Forest we got an accuracy of 0.64 with an F1 score for class 0 at 0.74 and an F1 score for class 1 at 0.44. For Gradient Boosting we got an accuracy of 0.65 with an F1 score for class 0 at 0.75 and an F1 score for class 1 at 0.43. For XGBoost we got an accuracy of 0.64 and an F1 score for class 0 at 0.73 and an F1 score for class 1 at 0.45.

6 Conclusion

Overall, each model seemed to have a similar performance and accuracy percentage which we again note as a moderate outcome. Since the F1 scores for Class 0 are more balanced between precision and recall than the F1 scores for Class 1, we can conclude that the models' performance is better at predicting water samples that are not potable compared to those that are potable. These outcomes indicate that there is potential in improving the models for the classification task and that each model's performance varies between the two classes.

Appendix

https://colab.research.google.com/drive/1EC5pmRN64RAuX2W3i6Gl6bfoFQDI68TY?usp=sharing

References

[1] Hancock, N. (2022, September 7). TDS and ph. Safe Drinking Water Foundation. https://www.safewater.org/fact-sheets-1/2017/1/23/tds-and-ph#:~:text=In%20the%20United%20States%2C%20pH,bitter%2C%20metallic%20taste%20and%20corrosion.

[2] Thomaskonstantin. (2021, September 12). Exploring and predicting drinking water potability. Kaggle. https://www.kaggle.com/code/thomaskonstantin/exploring-and-predicting-drinking-water-potability

[3]

Kadiwal, A. (2021, April 25). Water quality. Kaggle. https://www.kaggle.com/datasets/adityakadiwal/water-potability/data