WATER QUALITY ANALYSIS USING MACHINE LEARNING

REPORT

ITSOLERA PVT LTD

MACHINE LEARNING (INTERNS)
TEAM ETA

Zarnab Zafar (Team Lead)
Muhammad Mustafa Shah
Ayesha Majeed
Wajahat Hussain
Syeda Fatima
Moeed Ahmed
Ayesha Kamran

Introduction

1. Problem Statement

Objective

The objective of this project is to develop a machine learning model to classify water samples as safe or unsafe for consumption based on key water quality indicators. This model aims to help stakeholders, such as water quality management authorities, by providing accurate predictions that can aid in ensuring safe drinking water.

Background

Access to safe drinking water is a fundamental human right and a critical need for all individuals. Ensuring water quality involves understanding various factors that affect potability and identifying contaminants that may render water unsafe for consumption. This project leverages machine learning techniques to analyze data on water quality and predict water potability.

2. Description

Overview

The Water Quality Analysis Using Machine Learning project aims to develop a predictive model using various machine learning algorithms. The project includes data collection, preprocessing, feature engineering, model development, evaluation, optimization, and deployment.

Scope

The scope of the project includes:

- Collecting and preprocessing a comprehensive dataset of water quality indicators.
- Developing and training machine learning models.
- Evaluating and optimizing the models for high accuracy and reliability.
- Implementing the predictive model in a user-friendly application.

Dataset Details

1. Data Source

The dataset used for this project is sourced from Kaggle, and contains data on major factors affecting water potability.

2. Features and Target

The dataset includes various features such as pH, hardness, dissolved oxygen, turbidity, and presence of contaminants. The target variable is a binary indicator of water potability (safe or unsafe).

3. Data Preprocessing

- Handling missing values
- Normalizing numerical features
- Encoding categorical variables
- Splitting the data into training, validation, and test sets

Methodology

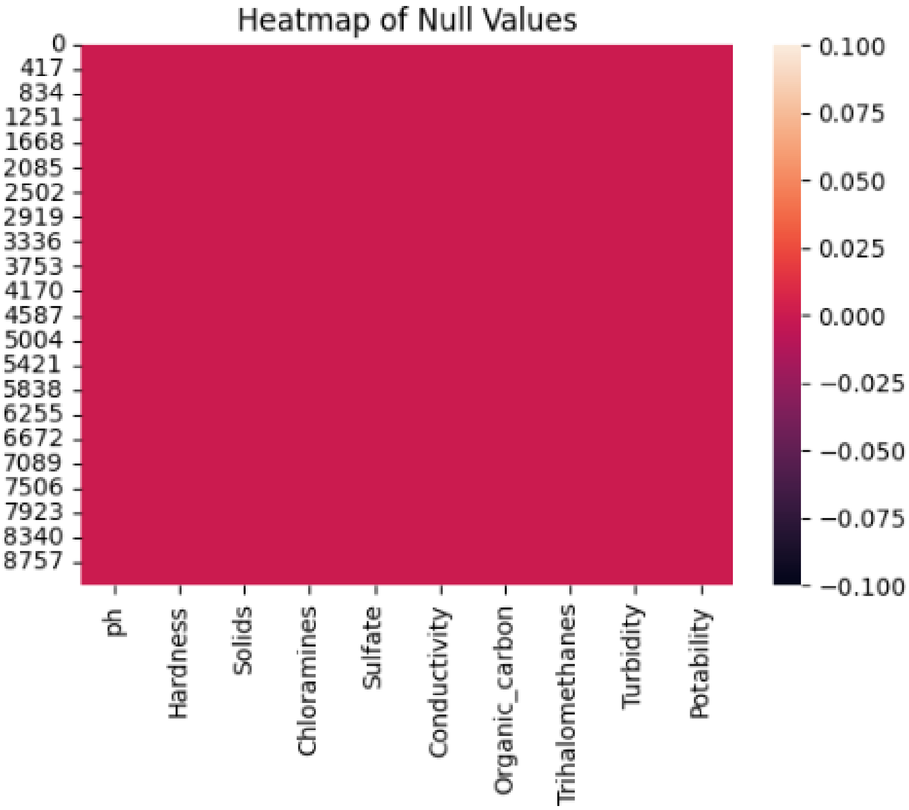
1. Understanding Water Quality Indicators

- Studied factors affecting water quality, such as pH, hardness, dissolved oxygen, and turbidity.
- Explored relationships between these indicators and water potability.

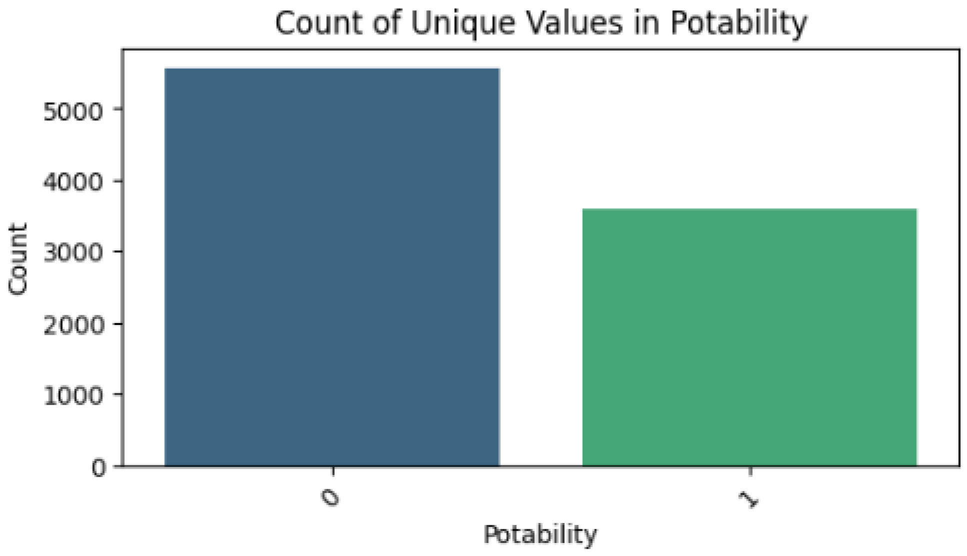
2. Data Collection and Preprocessing

Utilized a Kaggle dataset containing data on water quality indicators.

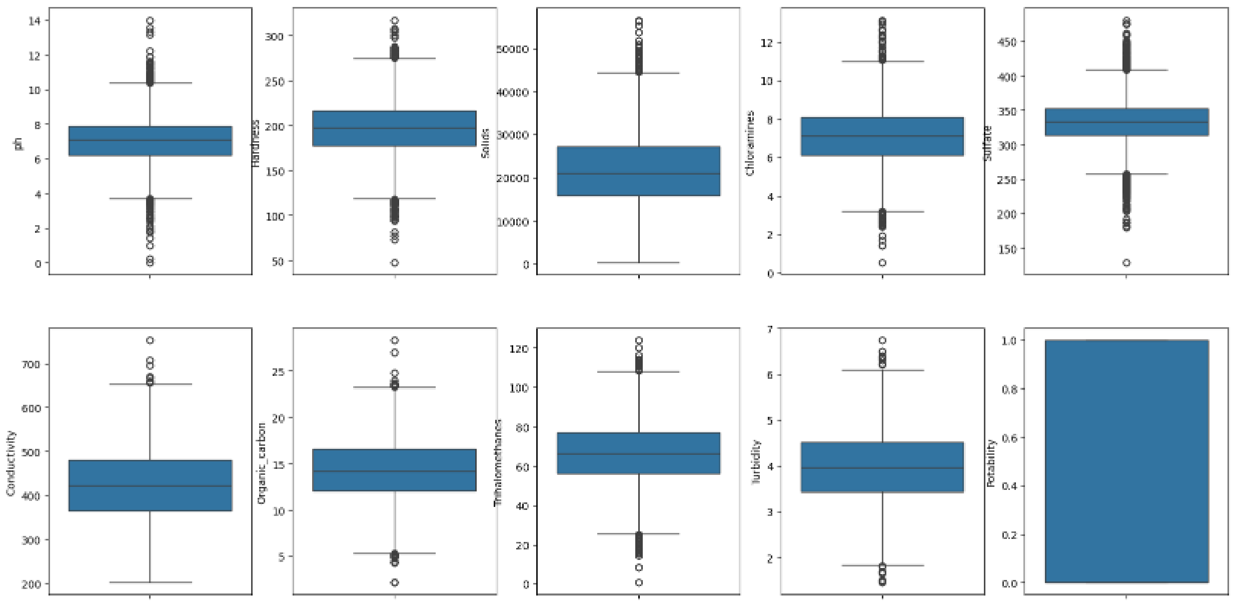
- **Pre-processed the dataset by handling missing values, normalizing features, and encoding categorical variables.**



The heatmap titled "Heatmap of Null Values" visualizes the presence of null values in a Data Frame. Each cell in the heatmap corresponds to an element in the Data Frame, with the x-axis representing different columns such as 'ph', 'Hardness', 'Solids', etc., and the y-axis representing the row indices. The color of the cells indicates the presence or absence of null values, with this particular heatmap showing a uniform color across all cells, suggesting there are no null values in the Data Frame.



The bar graph titled "Count of Unique Values in Potability" shows the distribution of two unique values within the 'Potability' column of a dataset. The x-axis represents the potability values, where '0' indicates non-potable water and '1' indicates potable water. The y-axis represents the count of occurrences for each value. The graph reveals that there are more instances of non-potable water (value '0') compared to potable water (value '1'), with counts of approximately 5000 and 2500 respectively.

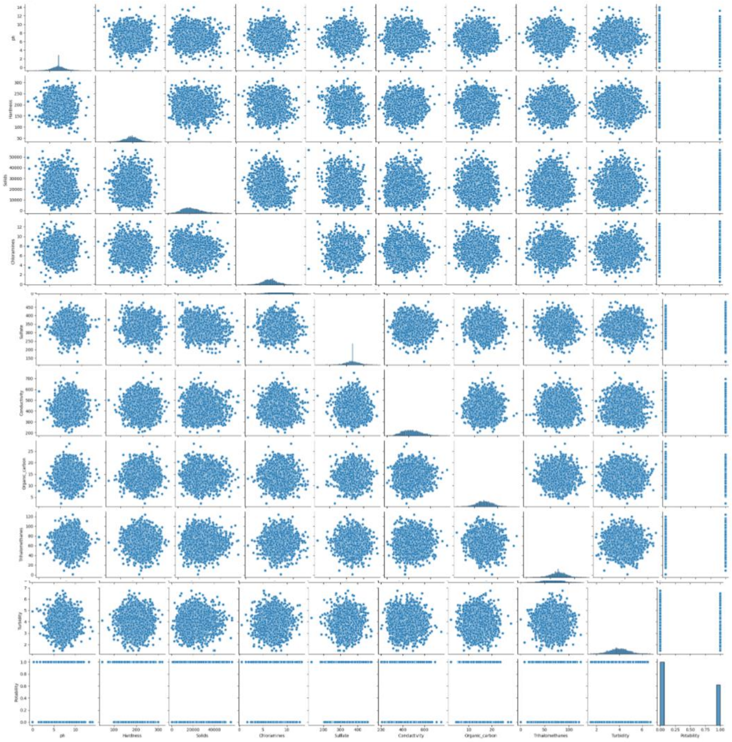


The graph consists of a series of box plots and a bar chart visualizing different features of a dataset. Each box plot represents the distribution of values for a particular feature: 'ph', 'Hardness', 'Solids', 'Chloramines', 'Sulfate', 'Conductivity', 'Organic_carbon', 'Trihalomethanes', and 'Turbidity'.

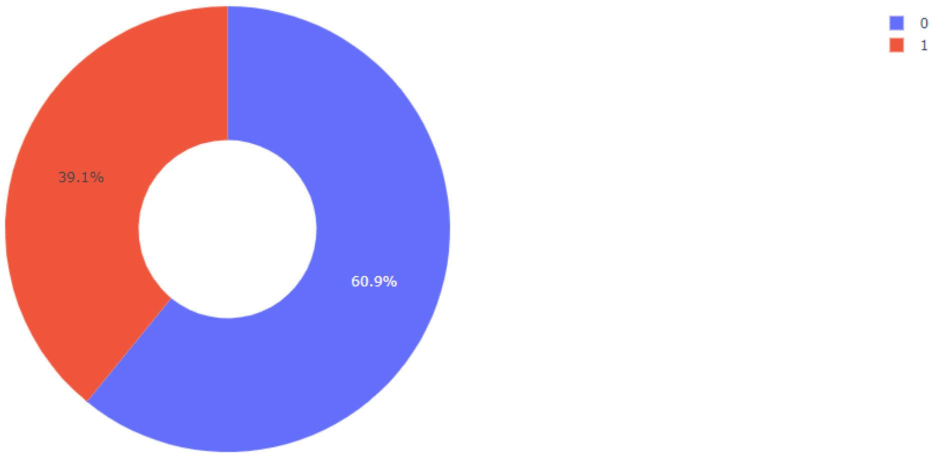
In the box plots:

- The central box shows the interquartile range (IQR) where 50% of the data points lie.
- The line inside the box represents the median.
- The whiskers extend to show the range of the data, excluding outliers.
- Circles outside the whiskers represent outliers.

The final chart on the bottom right is a bar chart representing 'Potability', showing the distribution of potable (1) and non-potable (0) water samples. The bar height indicates the frequency of each category. The uniform bar at 1 indicates that the feature is categorical, showing the count of unique values in 'Potability'.



The pair plot consists of a square matrix of subplots, with each variable in the dataset represented on both the x-axis and y-axis. The diagonal subplots show the distribution of each variable, likely as histograms or density plots. The off-diagonal subplots display scatter plots of each pair of variables, with the x-axis variable on the bottom and the y-axis variable on the left. These scatter plots reveal the relationships between each pair of variables, such as correlations, patterns, or outliers. The graph provides a comprehensive overview of the relationships between all variables in the dataset, allowing for quick identification of correlations, patterns, and potential issues with the data.



This chart, created using Plotly Express, features a hole in the center and uses the "plotly" template, giving it a clean and modern look. Each segment of the donut represents a different category of "Potability" and its proportion relative to the entire dataset. The chart is interactive, allowing users to hover over segments to see detailed information such as the category name and its percentage. This visualization effectively highlights the relative distribution of the different categories, making it easy to compare their proportions at a glance

3. Feature Engineering

- Analysed the dataset to identify important features for water quality analysis.
- Created new features based on domain knowledge and data analysis to enhance model performance.

4. Model Development

- Explored various machine learning algorithms: Logistic Regression, Decision Tree, Random Forest, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), AdaBoost, and XGBoost.
- Trained multiple models using the preprocessed dataset, optimizing hyperparameters through cross-validation.
- Implemented techniques such as feature selection and regularization to prevent overfitting and improve generalization.

5. Model Evaluation and Optimization

- Evaluated trained models using metrics like accuracy, precision, recall, F1-score, and AUC-ROC.
- Compared the performance of different models and select the best-performing one for further optimization.
- Fine-tuned the selected model by adjusting hyperparameters and incorporating ensemble methods.

6. Implementation and Testing

- Developed a user-friendly application interface using Flask.
- Integrated the predictive model into the application for user inputs and potability predictions.
- Tested the application with real-world data to ensure reliability and effectiveness.

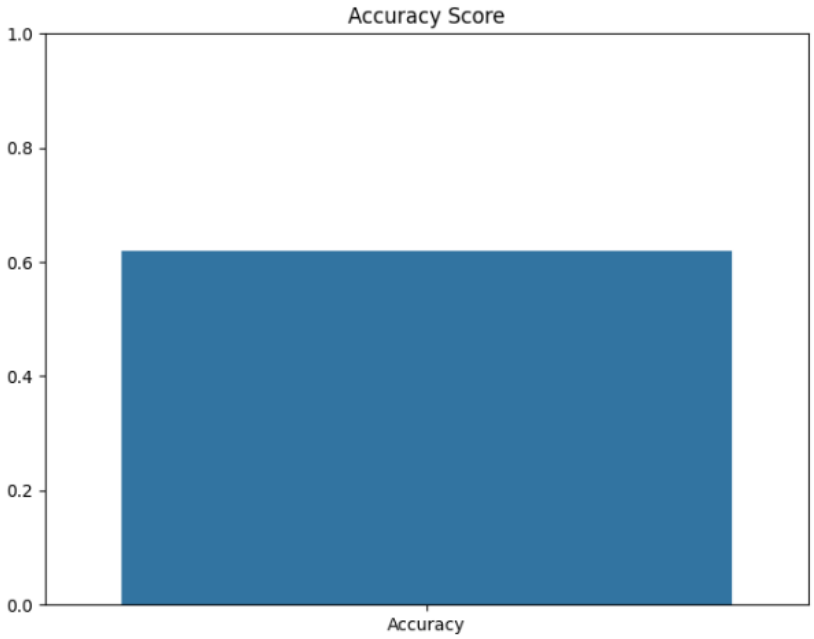
7. Deployment and Maintenance

- Deployed the application in a real-world environment, providing support and training for users.
- Monitored system performance and update the model periodically with new data to maintain accuracy.
- Implemented data security measures to protect user data and ensure compliance with privacy regulations.

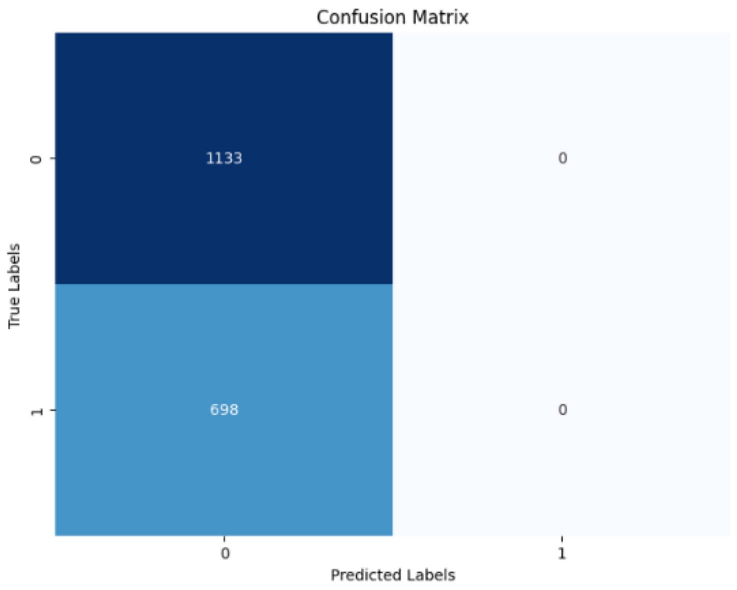
Results

1. Logistic Regression

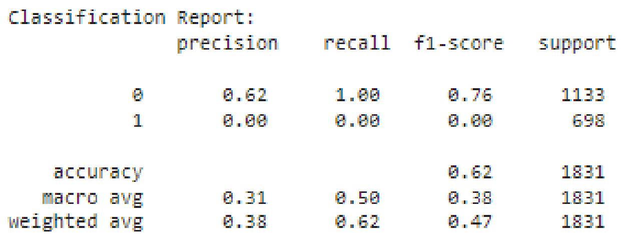
The bar chart titled "Accuracy Score" that displays the accuracy of a model. The chart consists of a single bar, labeled "Accuracy", with a height corresponding to the accuracy score of 0.62. The vertical axis ranges from 0 to 1, with 0 indicating complete inaccuracy and 1 representing perfect accuracy. The graph provides a simple and clear visualization of the model's performance. In this case, the bar's height is at approximately 0.62, indicating an accuracy score of 62%.



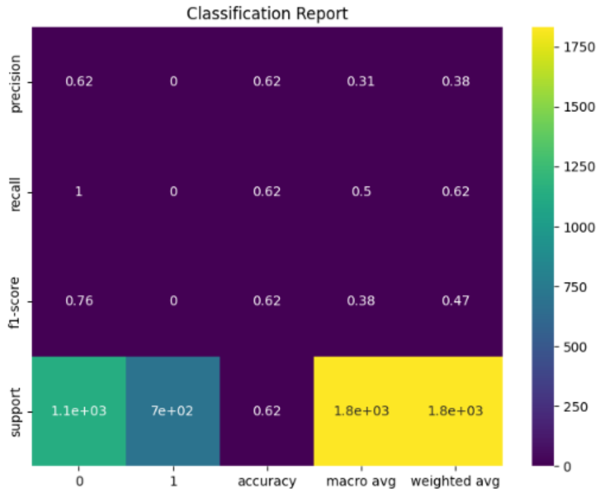
The below graph is a heatmap titled "Confusion Matrix" that displays the performance of a classification model. The heatmap is divided into four quadrants, with the x-axis representing the Predicted Labels and the y-axis representing the True Labels. The color scheme is a range of blues, with darker shades indicating higher values. The numbers within each quadrant are annotated, showing the exact count of data points. The top-left quadrant shows 1133 true negatives (correctly predicted as 0), and the top-right quadrant shows 0 false positives (incorrectly predicted as 1 when true label is 0). The bottom-left quadrant shows 0 false negatives (incorrectly predicted as 0 when true label is 1), and the bottom-right quadrant shows 698 true positives (correctly predicted as 1).



The report shows that the model has a precision of 0.62, a recall of 1.00, and an F1-score of 0.76 for class 0. For class 1, the model has a precision of 0.00, a recall of 0.00, and an F1-score of 0.00. The overall accuracy of the model is 0.62. The report also shows the support for each class, which is the number of samples in each class. The support for class 0 is 1133 and the support for class 1 is 698.The graph is a heatmap that shows the values of the precision, recall, and F1-score for each class. The heatmap shows that the model is better at predicting class 0 than class 1. This is because the values of the precision, recall, and F1-score are higher for class 0 than for class 1. The heatmap also shows that the model is more likely to predict class 0 than class 1.

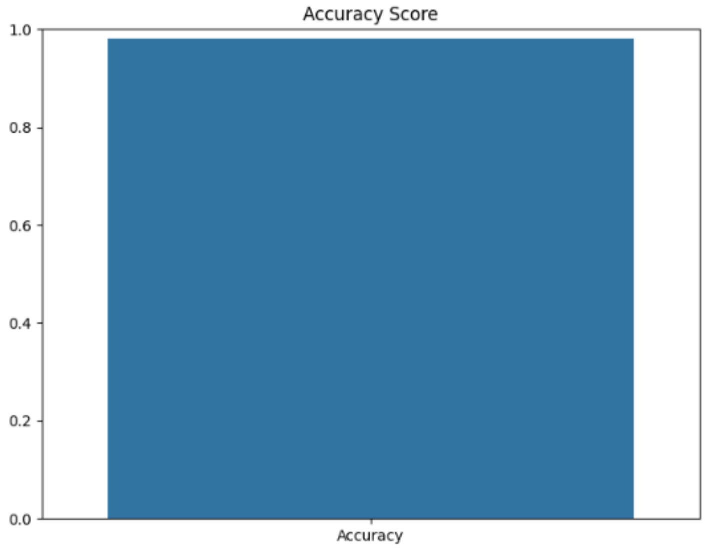


```
Classification Report:
              precision    recall  f1-score   support

           0       0.62      1.00      0.76      1133
           1       0.00      0.00      0.00       698

    accuracy                           0.62      1831
   macro avg       0.31      0.50      0.38      1831
weighted avg       0.38      0.62      0.47      1831
```

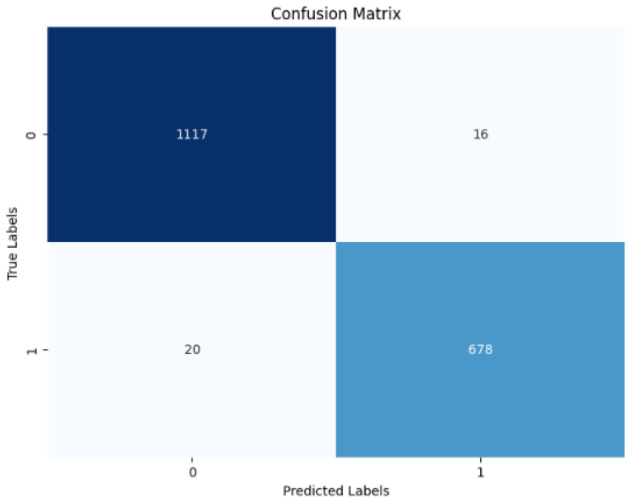


2. Decision Tree

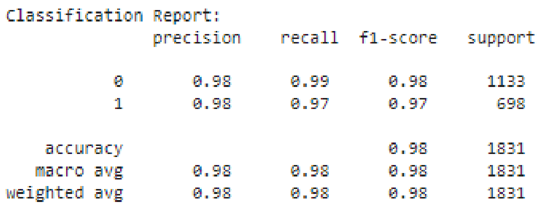
The title of the graph is 'Accuracy Score'. The x-axis is labeled as 'Accuracy' and the y-axis represents the accuracy score, ranging from 0 to 1. The height of the bar corresponds to the accuracy score, which is 0.9 in this case. The accuracy score is 0.98, which means that the model is able to correctly classify 98% of the data points. This is a very high accuracy score, indicating that the model is performing well.



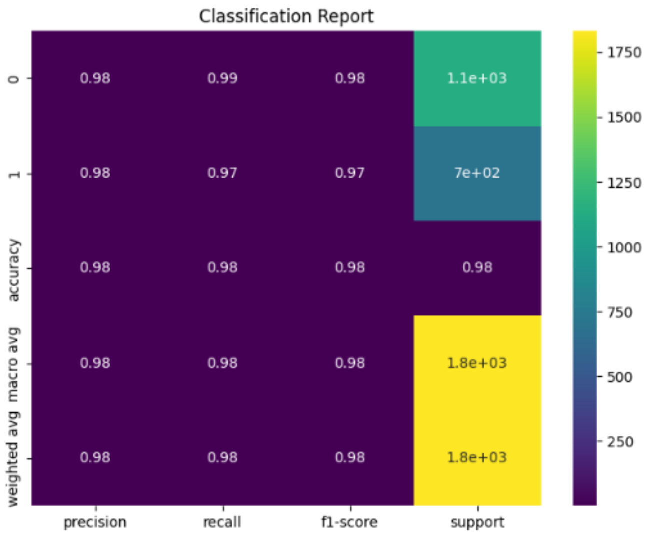
The below graph is of 'Confusion Matrix'. The x-axis is labeled as 'Predicted Labels' and the y-axis is labeled as 'True Labels'. The heatmap is colored according to the 'Blues' color map, with darker colors indicating higher values. The numbers within each cell of the heatmap are annotated, showing the exact count of samples for each combination of true and predicted labels. The graph shows that the model correctly classified 1117 instances of class 0 and 678 instances of class 1, while misclassifying 16 instances of class 0 and 20 instances of class 1. The color intensity and annotated values help to quickly understand the performance of the model.



The classification report shows the performance of a machine learning model on a dataset. The report displays the precision, recall, f1-score, and support for each class, as well as the overall accuracy, macro average, and weighted average. The model has an accuracy of 0.98, indicating that it is performing well. The precision, recall, and f1-score are all high for both classes, indicating that the model is able to accurately identify both classes. The support is also high for both classes, indicating that the model has been trained on a large amount of data. The graph would provide a visual representation of the model's accuracy, precision, recall, and other performance metrics, allowing for a quick and easy understanding of the model's strengths and weaknesses.

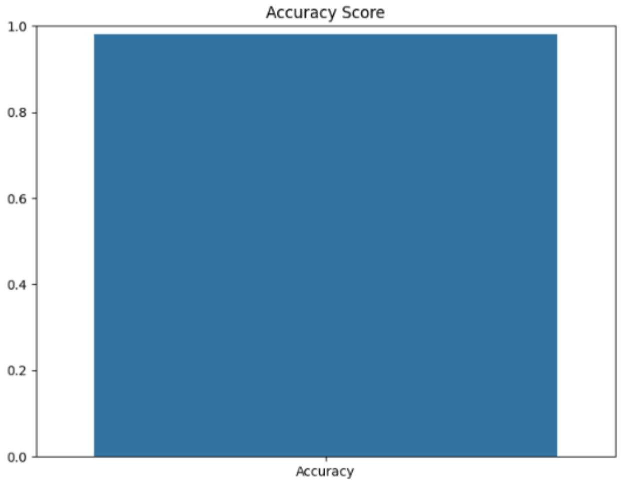


```
Classification Report:
              precision    recall  f1-score   support

           0       0.98      0.99      0.98      1133
           1       0.98      0.97      0.97       698

    accuracy                           0.98      1831
   macro avg       0.98      0.98      0.98      1831
weighted avg       0.98      0.98      0.98      1831
```

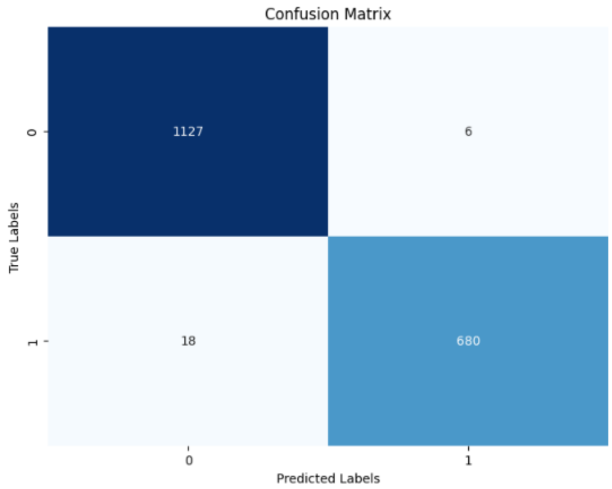


3. Random Forest

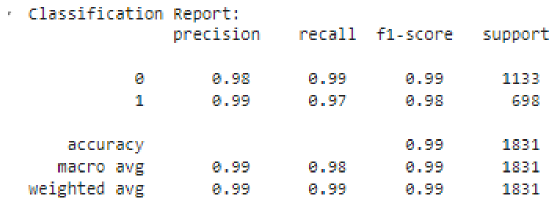
The graph has a single bar labeled "Accuracy" on the x-axis, and the corresponding accuracy score on the y-axis. In this case, the bar's height is at approximately 0.99, indicating an accuracy score of 99%. The bar's height corresponds to the accuracy score, which is a value between 0 and 1, indicating the proportion of correctly classified instances by the Random Forest model. The title of the graph is "Accuracy Score", providing a clear and concise summary of the model's performance. The graph provides a quick and easy way to visualize the accuracy of the Random Forest model, allowing for a rapid assessment of its effectiveness in making predictions.



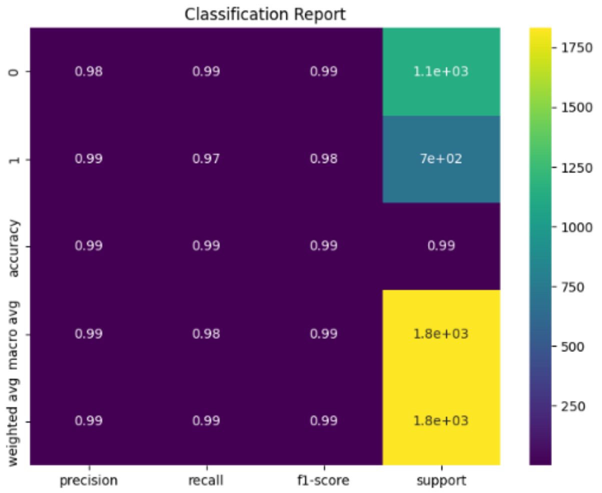
The heatmap is titled "Confusion Matrix" and has "True Labels" on the y-axis and "Predicted Labels" on the x-axis. The matrix is divided into four quadrants, with the top-left quadrant representing true positives (correctly predicted instances), the top-right quadrant representing false positives (incorrectly predicted instances), the bottom-left quadrant representing false negatives (missed instances), and the bottom-right quadrant representing true negatives (correctly rejected instances). The color scheme used is a range of blues, with darker colors indicating higher values. The numbers within each quadrant are annotated, providing a clear and detailed view of the model's performance. The heatmap provides a visual representation of the model's accuracy, precision, and recall, allowing for a quick and intuitive understanding of its strengths and weaknesses



The heatmap displays the performance of a binary classification model, with two classes: 0 and 1. The top section of the heatmap shows the performance metrics for class 0, while the bottom section shows the performance metrics for class 1. The columns of the heatmap represent precision, recall, F1-score, and support, with values ranging from 0.97 to 0.99, indicating that the model is highly precise, effective in detecting true positives, and achieves a good balance between precision and recall for both classes. The support values reveal that the model was trained on a slightly imbalanced dataset, with more instances in class 0 (1133) than class 1 (698). The color scheme of the heatmap is a range of viridis colors, with darker colors indicating higher values, and the overall heatmap suggests that the model is highly accurate and effective in classifying instances into both classes 0 and 1.

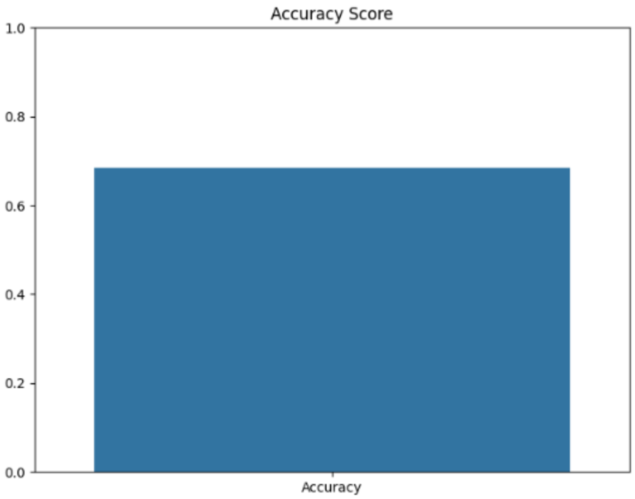


```
Classification Report:
              precision    recall  f1-score   support

           0       0.98      0.99      0.99      1133
           1       0.99      0.97      0.98       698

    accuracy                           0.99      1831
   macro avg       0.99      0.98      0.99      1831
weighted avg       0.99      0.99      0.99      1831
```

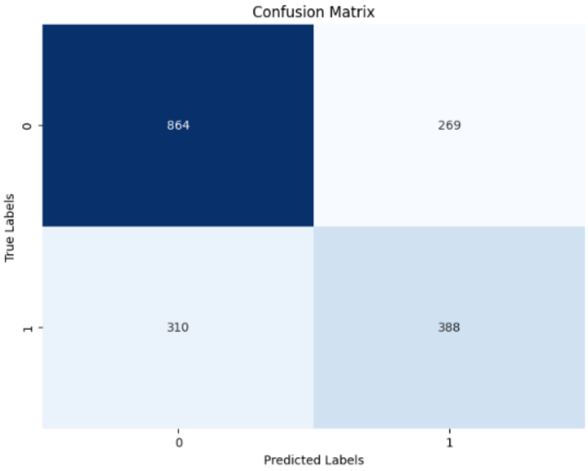


4. KNN

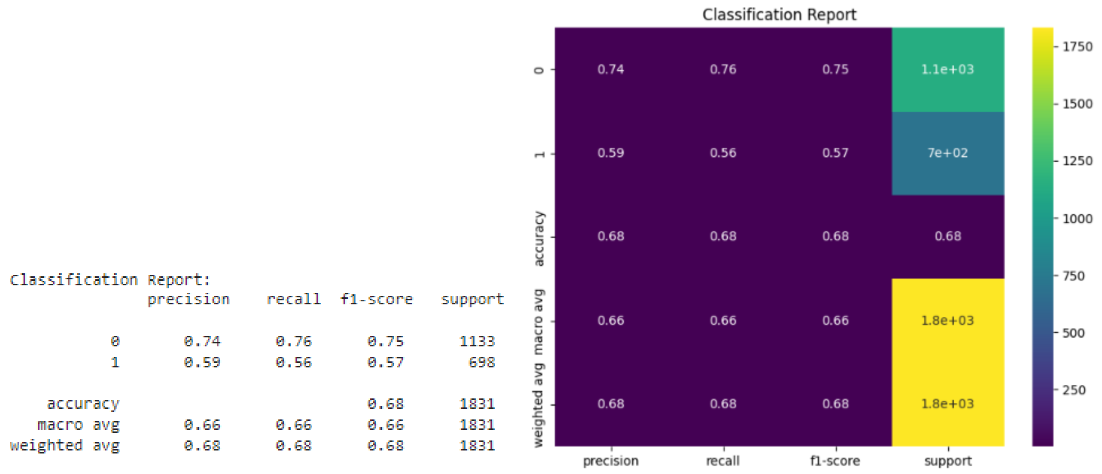
A K-Nearest Neighbors (KNN) model, which is a supervised learning algorithm that classifies new instances based on the majority vote of its neighbors. The chart has a single bar labeled "Accuracy" on the x-axis, and the corresponding y-axis value represents the accuracy score, which ranges from 0 to 1. The bar's height indicates the accuracy score, which is a value between 0 and 1, with 1. In this case, the bar's height is at approximately 0. 68, indicating an accuracy score of 68%. The chart has a clean and minimalistic design, with a title "Accuracy Score" at the top, and the y-axis limits set to 0 and 1 for easy interpretation.



The graph is a confusion matrix, a table showing the performance of a classification model. The rows represent the true labels, while the columns represent the predicted labels. The numbers in the cells show the number of instances that were correctly or incorrectly classified. The matrix reveals that 864 instances were correctly classified as belonging to class 0, while 310 instances were incorrectly classified as belonging to class 0 when they actually belonged to class 1. On the other hand, 269 instances were incorrectly classified as belonging to class 1 when they actually belonged to class 0, and 388 instances were correctly classified as belonging to class 1. This confusion matrix suggests that the model is performing well at classifying instances belonging to class 1, but it is less accurate at classifying instances belonging to class 0, as reflected in the higher number of false positives compared to false negatives.



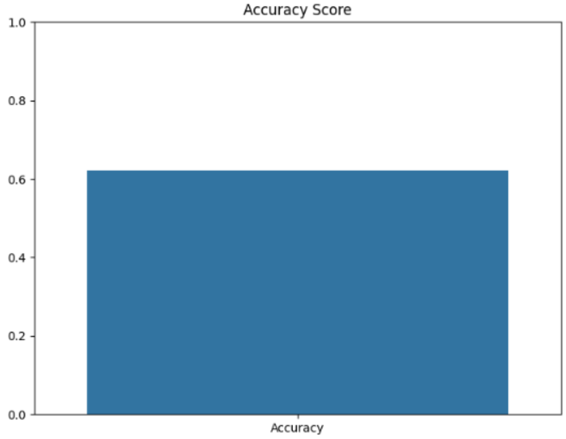
The graph shows the classification report. The report includes the precision, recall, f1-score, and support for each class, as well as the overall accuracy, macro average, and weighted average. The precision for class 0 is 0.74, and the recall is 0.76. The f1-score is 0.75, and the support is 1133. The precision for class 1 is 0.59, and the recall is 0.56. The f1-score is 0.57, and the support is 698.The overall accuracy is 0.68. The macro average precision, recall, and f1-score are all 0.66. The weighted average precision, recall, and f1-score are all 0.68. This indicates that the model is performing better on class 0 than class 1. The model has a higher precision for class 0, meaning that it is more likely to correctly identify instances of class 0. The model also has a higher recall for class 0, meaning that it is more likely to correctly identify all instances of class 0.



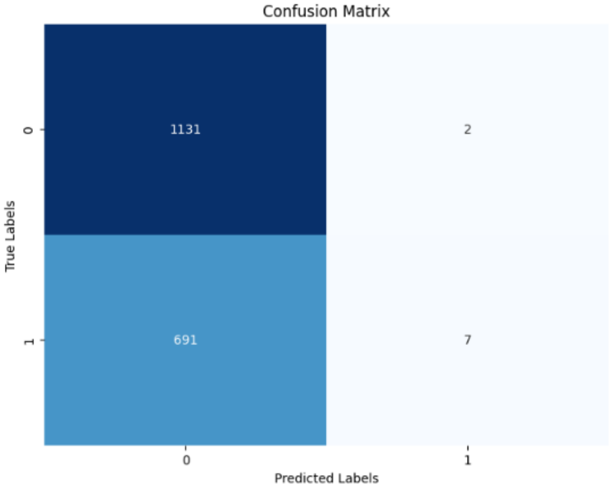
```
Classification Report:
              precision    recall  f1-score   support

           0       0.74      0.76      0.75      1133
           1       0.59      0.56      0.57       698

    accuracy                           0.68      1831
   macro avg       0.66      0.66      0.66      1831
weighted avg       0.68      0.68      0.68      1831
```

5. SVM

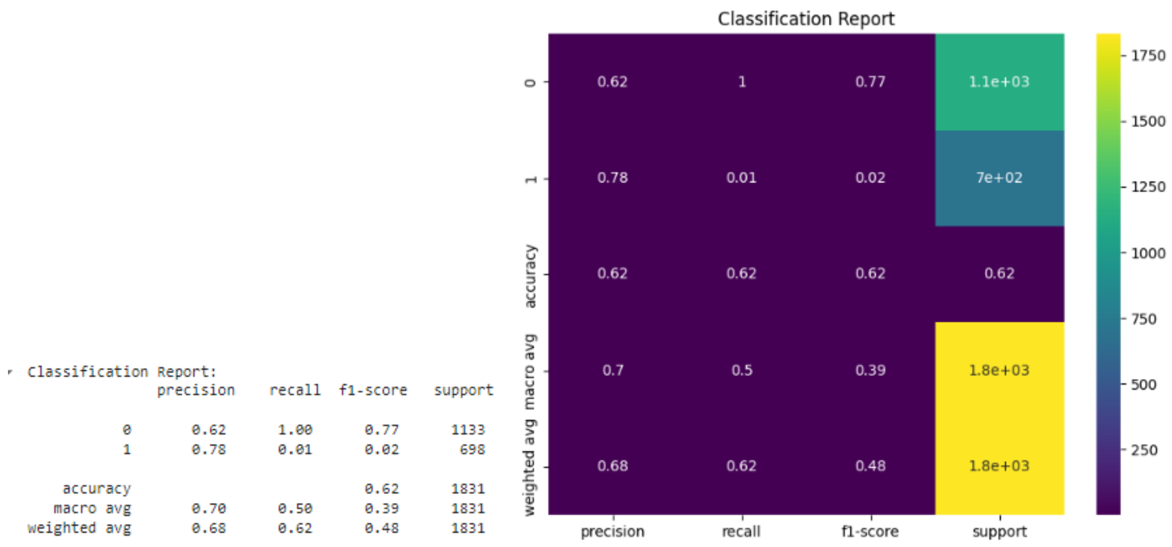
The graph displays the accuracy score of a Support Vector Machine (SVM) model. The accuracy score is 0.62, indicating that the model correctly classified approximately 62% of the instances. The graph is a horizontal bar chart with a single bar representing the accuracy score. The x-axis is labeled "Accuracy" and the y-axis is a numerical scale ranging from 0.0 to 1.0, with increments of 0.2 (0.0, 0.2, 0.4, 0.6, 0.8, 1.0). The bar is colored blue and extends from the y-axis to the point corresponding to the accuracy score of 0.62. The graph provides a visual representation of the SVM model's performance, allowing for easy interpretation of its accuracy.



This is a confusion matrix, a table used to evaluate the performance of a classification model. The matrix is divided into four quadrants, each representing a different outcome of the classification process. The rows of the matrix represent the actual classes, also known as the true labels, while the columns represent the predicted classes, also known as the predicted labels. The top-left quadrant shows the number of true negatives (TN), which is 1131, indicating that the model correctly classified 1131 instances as belonging to class 0. The top-right quadrant shows the number of false positives (FP), which is 2, indicating that the model misclassified 2 instances as belonging to class 1 when they actually belonged to class 0. The bottom-left quadrant shows the number of false negatives (FN), which is 7, indicating that the model misclassified 7 instances as belonging to class 0 when they actually belonged to class 1. The bottom-right quadrant shows the number of true positives (TP), which is 691, indicating that the model correctly classified 691 instances as belonging to class 1. Overall, the confusion matrix provides a detailed breakdown of the model's performance, allowing for the calculation of various metrics such as accuracy, precision, recall, and F1-score. In this case, the model appears to have a high accuracy, with a low number of misclassifications.



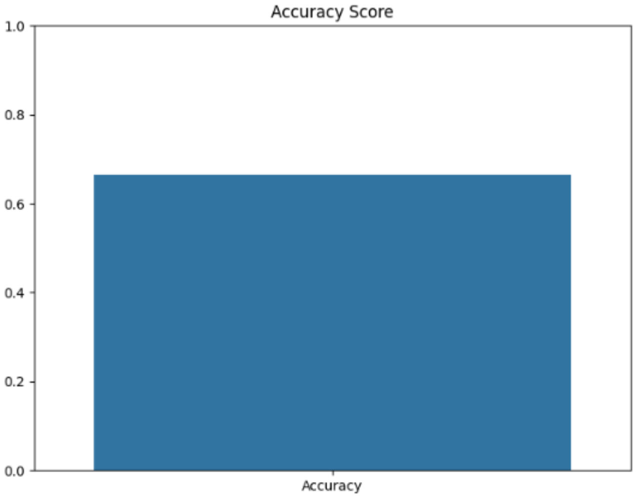
The graph is a classification report that shows the performance of a machine learning model. The report includes metrics such as precision, recall, F1-score, and support, which provide a comprehensive view of the model's performance. The model performs well on the first class, with a precision of 0.62, recall of 1.00, and F1-score of 0.77, indicating that it correctly classifies most instances of this class. However, it performs poorly on the second class, with a precision of 0.78, recall of 0.01, and F1-score of 0.02, indicating a high false negative rate for this class. The weighted average metrics show an overall accuracy of 0.62 and a weighted average F1-score of 0.48, indicating that the model has an acceptable overall performance, but it could be improved by addressing the issue of high false negative rate for the second class. The confusion matrix on the right side of the graph further visualizes the performance of the model, with the color intensity indicating the number of samples in each cell. As can be seen, there are many more samples correctly classified as the first class than the second class.



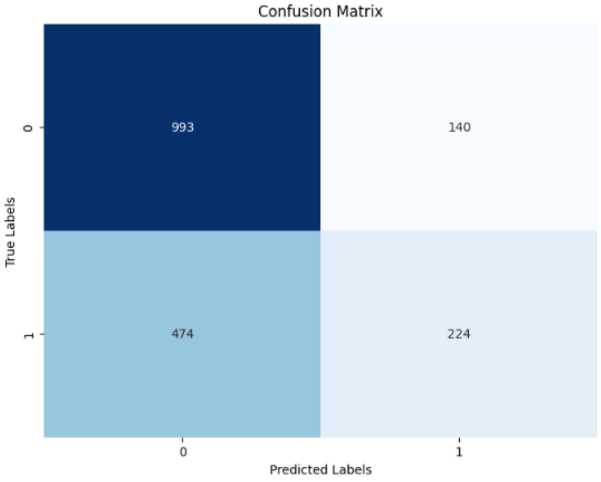
```
Classification Report:
              precision    recall  f1-score   support

           0       0.62      1.00      0.77      1133
           1       0.78      0.01      0.02       698

    accuracy                           0.62      1831
   macro avg       0.70      0.50      0.39      1831
weighted avg       0.68      0.62      0.48      1831
```

6. ADA Boost

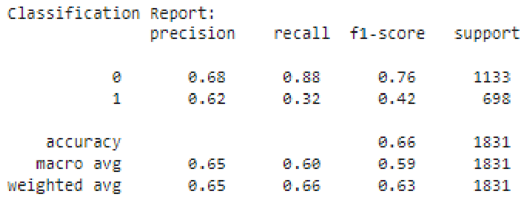
The graph is a bar chart showing the accuracy score of an ADA Boosting model. The x-axis displays the different iterations of the ADA Boosting algorithm, with values ranging from 0.0 to 1.0, incrementing by 0.2. In this case, the bar's height is at approximately 0.66, indicating an accuracy score of 66%.



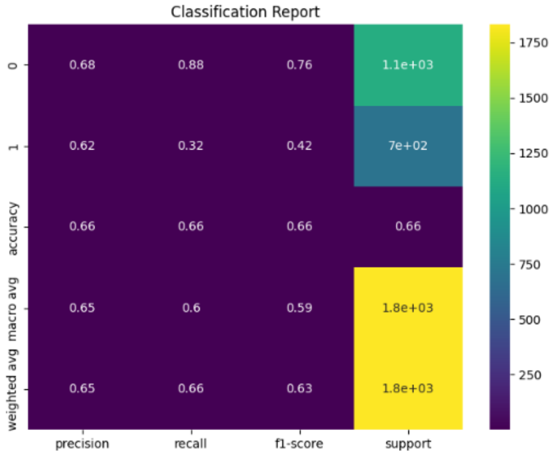
The matrix is divided into four quadrants, each representing a different outcome. The top-left quadrant shows the number of true negatives (TN), which is 993, indicating that the model correctly predicted 993 instances as belonging to class 0. The top-right quadrant shows the number of false positives (FP), which is 140, indicating that the model misclassified 140 instances of class 0 as class 1. The bottom-left quadrant shows the number of false negatives (FN), which is 474, indicating that the model misclassified 474 instances of class 1 as class 0. The bottom-right quadrant shows the number of true positives (TP), which is 224, indicating that the model correctly predicted 224 instances as belonging to class 1.



The graph shows a classification report, evaluating its performance on a dataset of 1831 samples with two classes. The report includes metrics such as precision, recall, F1-score, and support, which indicate the model's ability to correctly classify samples. The model performs well on class 0, with a precision of 0.68 and a recall of 0.88, but struggles with class 1, with a precision of 0.62 and a recall of 0.32. The overall accuracy of the model is 0.66, and the macro average and weighted average metrics suggest reasonable performance. However, the significant difference in performance between the two classes is notable, with the model performing much better on class 0 than class 1. This may indicate bias towards class 0, which could be due to an imbalance in the training data or the presence of features that are more predictive of class 0.

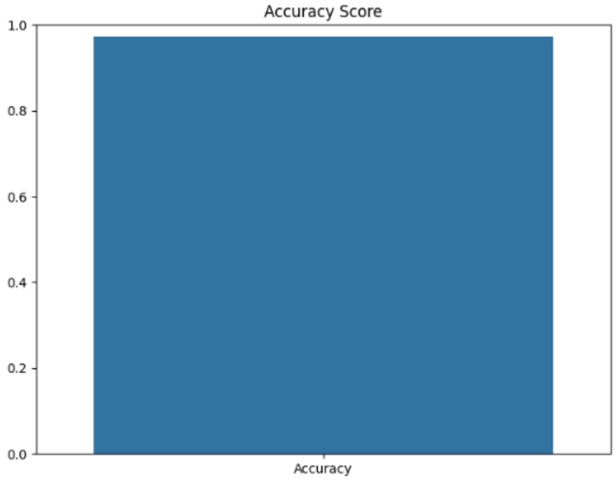


```
Classification Report:
              precision    recall  f1-score   support

           0       0.68      0.88      0.76      1133
           1       0.62      0.32      0.42       698

    accuracy                           0.66      1831
   macro avg       0.65      0.60      0.59      1831
weighted avg       0.65      0.66      0.63      1831
```

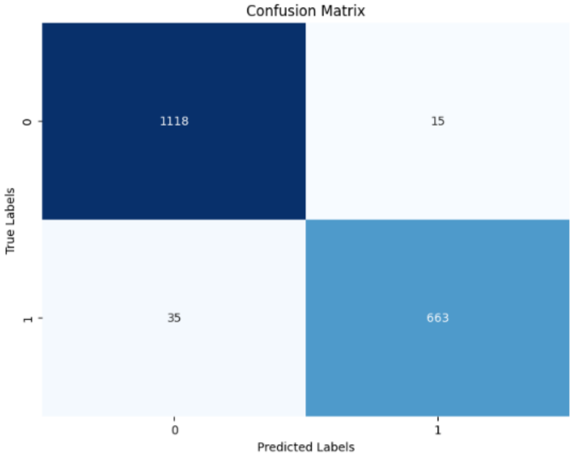


7. XG Boost

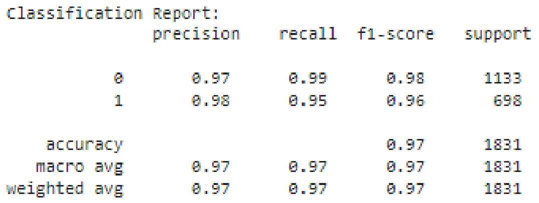
The graph displays the accuracy score of an XG Boost model, with a range of possible scores from 0.0 to 1.0. The model's accuracy score is plotted on the graph, with six distinct points marked: 1.0, 0.8, 0.6, 0.4, 0.2, and 0.0. The actual accuracy score of the XG Boost model in this case, the bar's height is at approximately 0.97, indicating an accuracy score of 97%.



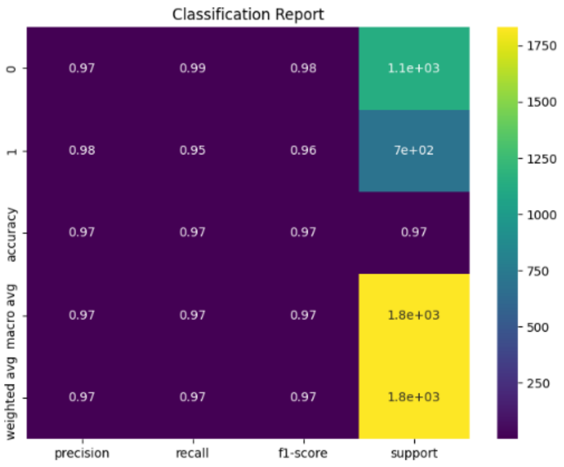
The graph shows four distinct regions, with a large, dark blue square in the top-left corner indicating a high value of 1118, which represents the True Negatives (TN). In contrast, the top-right and bottom-left regions are small, light blue squares, indicating low values of 15 and 35, respectively, which represent the False Positives (FP) and False Negatives (FN). The bottom-right region is another large, dark blue square, indicating a high value of 663, which represents the True Positives (TP). Overall, the heatmap provides a clear visual representation of the XG Boost model's performance, highlighting its accuracy in predicting the True Negatives and True Positives, while also revealing the errors in predicting False Positives and False Negatives.



The heatmap shows that the model has a very high accuracy, with an overall accuracy of 0.97. This means that the model is able to correctly classify 97% of the examples. The model also has high precision, recall, and f1-score for both classes, which means that it is able to correctly identify both classes with a high degree of accuracy. The support column shows the number of examples in each class, which is 1133 for class 0 and 698 for class 1. Overall, the heatmap indicates that the model is performing very well.



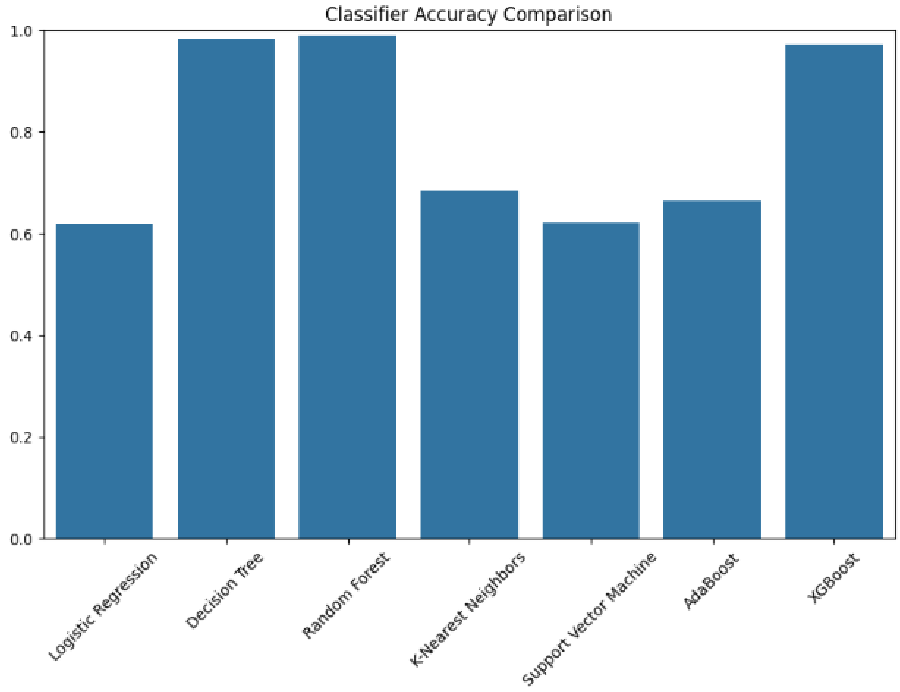
```
Classification Report:
              precision    recall  f1-score   support

           0       0.97      0.99      0.98      1133
           1       0.98      0.95      0.96       698

    accuracy                           0.97      1831
   macro avg       0.97      0.97      0.97      1831
weighted avg       0.97      0.97      0.97      1831
```



COMPARISON & FINDING BEST ONE

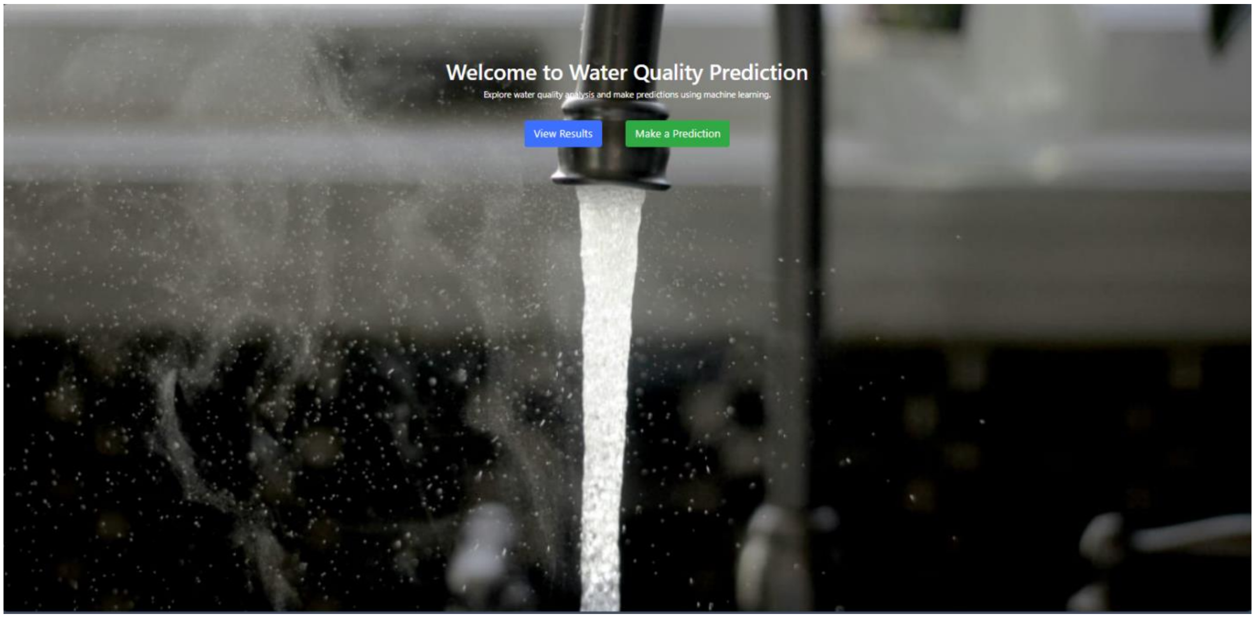
The graph is a bar chart comparing the accuracy of different machine learning algorithms. The x-axis lists the different algorithms: Logistic Regression, Decision Tree, Random Forest, K-Nearest Neighbors, Support Vector Machine, AdaBoost, and XG Boost. The y-axis represents the accuracy of each algorithm.

The algorithm with the highest accuracy is Random Forest, with an accuracy of 1.0, followed closely by XG Boost and AdaBoost. Logistic Regression, Decision Tree, K-Nearest Neighbors, and Support Vector Machine have significantly lower accuracies. This suggests that Random Forest is the best performing algorithm for this particular dataset, making it the top choice for the program.



Deployment

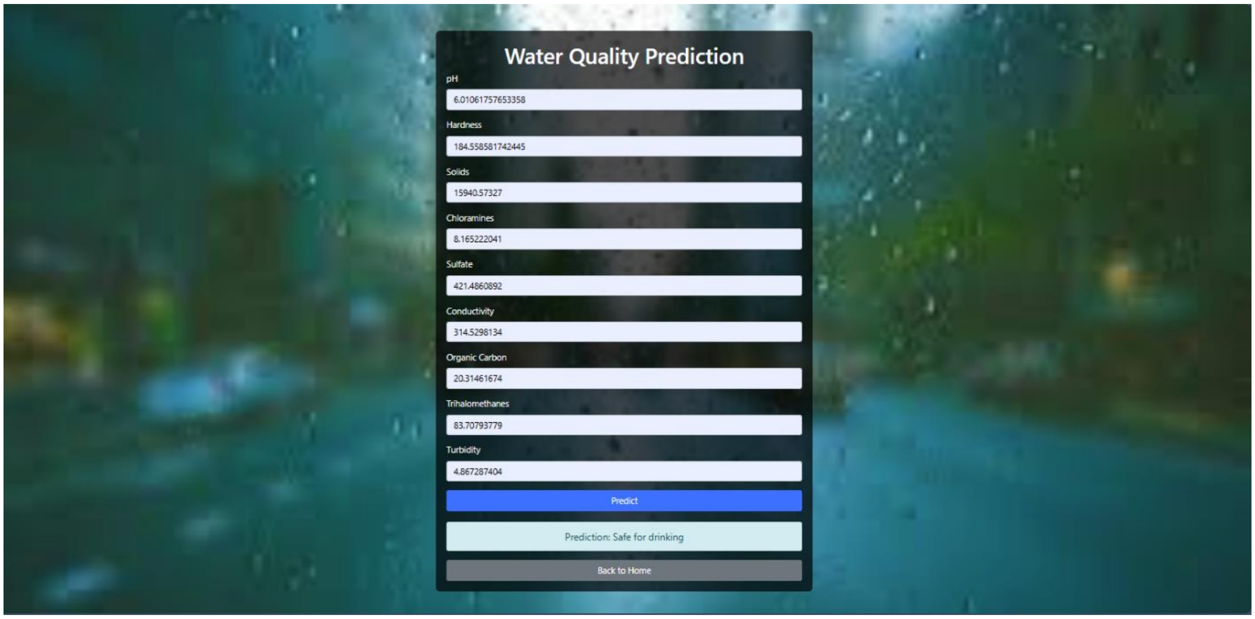
1. Welcome Page:



Welcome to Water Quality Prediction

Explore water quality analysis and make predictions using machine learning.

2. Prediction:



Water Quality Prediction

pH

6.01061757653358

Hardness

184.558581742445

Solids

15940.57327

Chloramines

8.165222041

Sulfate

421.4860892

Conductivity

314.5298134

Organic Carbon

20.31461674

Trihalomethanes

83.70793779

Turbidity

4.867287404

Prediction: Safe for drinking