**Technical Report for Kay Bailey Site**

**Overview**

The focus of this report is centered on the operations of the Kay Bailey Site, specifically emphasizing the prediction of treated water flow at this location. The dataset available to us encompasses various metrics, most notably the total volume of treated water processed at the plant, operational hours, rainfall, and raw water input. These metrics are crucial in understanding the dynamics of treated water flow at the Kay Bailey Site. The primary objective of this analysis is to harness the potential of the historical data in conjunction with cutting-edge machine learning techniques and time series forecasting methodologies. Our aim is to develop predictive models that can precisely forecast future values of Treated Water Flow. By doing so, this report seeks to provide invaluable insights that can enhance the operational efficiency and sustainability of the Kay Bailey Site. The predictions made for Treated Water Flow will act as a guiding framework for informed decision-making, facilitating optimal resource management and ensuring the highest standards of operational performance at the Kay Bailey Site.

**Data Preprocessing and Feature Engineering**

Before delving into the intricacies of the analysis, it's paramount to ensure that our data is pristine, structured, and augmented with relevant details. Proper data preparation not only paves the way for a smoother analysis but also ensures that the insights derived are accurate and actionable. In this section, we'll outline the meticulous steps undertaken to refine and enhance our dataset, setting the stage for the subsequent phases of our analysis.

**Data Importation**

The bedrock of any data-driven analysis is the integrity and organization of the dataset. For the Kay Bailey Site, our primary data source was an exhaustive Excel file named 'Kay Bailey.xlsx'. This dataset encapsulates a plethora of metrics that are pivotal in understanding the dynamics of treated water flow at the Kay Bailey Site. Consolidating all pertinent attributes within a singular dataset streamlines the analysis process, ensuring a comprehensive understanding of the operations at the Kay Bailey Site. Post data importation, it's imperative to discern the characteristics and datatype of each column. Such comprehension not only facilitates subsequent preprocessing endeavors but also ensures that the data is harnessed optimally during the modeling phase. To elucidate further, the table below provides a snapshot of the columns present in the dataset, accompanied by their respective data types

|  |  |
| --- | --- |
| Column Name | Data Type |
| Year | int64 |
| Month | object |
| Date | int64 |
| Airport Wells | int64 |
| Gal Units1 | int64 |
| Total Flow1 | float64 |
| Ft. Bliss Wells | int64 |
| Gal Units2 | int64 |
| Total Flow2 | float64 |
| Concentrate | int64 |
| Gal Units3 | float64 |
| Total Flow3 | float64 |
| Finished Water | object |
| Gal Units4 | int64 |
| Total Flow4 | float64 |
| CL2 | float64 |

This tabulation offers a structured overview of the dataset, delineating each column and its associated data type. Armed with this knowledge, we are well-equipped to embark on further data preprocessing, feature engineering, and modeling endeavors, with the ultimate goal of extracting valuable insights pertinent to the operations at the Kay Bailey Site.

**Data Cleaning**

Data cleaning is a pivotal step in the preprocessing pipeline, ensuring that the dataset is devoid of discrepancies, missing values, and other irregularities that could potentially distort the analysis. Here's a comprehensive overview of the cleaning procedures we employed for the Kay Bailey Site dataset

**Date Conversion and Indexing**

The 'Year', 'Month', and 'Date' columns, initially separate, were merged and converted into a singular 'Date' column of the datetime format. This transformation is instrumental for time-based operations and analyses, providing a unified timestamp for each record.

Post-conversion, the dataset was systematically organized based on this 'Date' column, ensuring a chronological arrangement of records. Such structuring is indispensable for time series data, facilitating seamless data operations based on specific time intervals.

**Handling Missing Values**

One of the most pervasive challenges in data analysis is the presence of missing values. These gaps, if left unaddressed, can introduce significant biases or inaccuracies in the subsequent stages of analysis. Initially, we conducted a thorough assessment to identify and quantify the missing values across all columns. This preliminary check provides a clear picture of the extent and distribution of data gaps. To tackle these missing values, we employed a forward-fill method. This technique substitutes missing entries with the preceding non-missing value, ensuring continuity in the data.

For columns with a datatype of 'object', a conversion to numeric was performed. This step ensures that all data is in a format suitable for mathematical and statistical operations. Any non-numeric values in these columns were transformed into NaNs, making them easier to identify and handle. Post these transformations, we conducted another round of checks to ascertain the presence of any residual NaN values. Based on the findings, we decided to drop rows containing these NaN values to maintain the integrity of our dataset.

Finally, specific numeric columns were explicitly converted to numeric datatypes, reinforcing the uniformity and consistency of the dataset. With these cleaning steps, we have ensured that the Kay Bailey Site dataset is in an optimal state, primed for the subsequent phases of exploration, modeling, and prediction.

**Feature Engineering**

Feature engineering is a pivotal process in data analysis, where raw data is transformed or new features are created to enhance the predictive power of the dataset. For the Kay Bailey Site dataset, we undertook several engineering steps to ensure the data is in the best possible format for subsequent analysis

**Date Conversion**

The 'Year', 'Month', and 'Date' columns, which were initially separate, were merged and converted into a singular 'Date' column in the datetime format. This transformation is essential for time series analysis, enabling more intuitive date-based operations and analyses. The format was inferred from the data to ensure accurate interpretation of the date strings.

**One-Hot Encoding for Categorical Variables**

The 'Month' column, which is categorical in nature, was transformed using one-hot encoding. This method involves creating binary columns for each category and indicating the presence of the category with a 1 or 0. In this case, dummy variables were created for each month, with the first month being dropped to avoid multicollinearity. This transformation is crucial for machine learning models, which require numerical input features.

**Lagged Features**

In time series analysis, the creation of lagged features is a common technique to enhance the predictive capabilities of models. These features represent values from previous time steps, providing historical context that can be invaluable for predictions.

For the Kay Bailey dataset, lagged features were generated specifically for the 'Total Flow4' column, which represents the treated water flow. Three lagged features were created, capturing the water flow values from the previous three days. These new columns were aptly named 'flows\_Lag\_1', 'flows\_Lag\_2', and 'flows\_Lag\_3', with each number indicating the number of days before the current date.

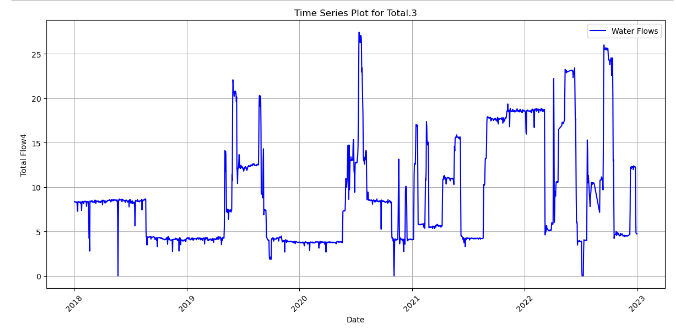
Due to the creation of these lagged features, the initial rows, which lacked sufficient historical data for lagging, contained NaN values. To maintain data consistency and integrity, these rows were promptly removed.

By the end of this feature engineering process, the Kay Bailey dataset was enriched with additional features, making it more robust and versatile for the subsequent modeling and analysis phases. These newly minted features are poised to provide deeper insights into the time-based patterns inherent in the data, potentially enhancing the accuracy and reliability of our predictions.

**Exploratory Data Analysis**

In the realm of time series analysis, visualizing the data is a crucial first step. It provides a clear picture of the underlying patterns, potential anomalies, and the general behavior of the series over time. For the Kay Bailey Site, we focused on the 'Total Flow4' column, which represents the treated water flow.

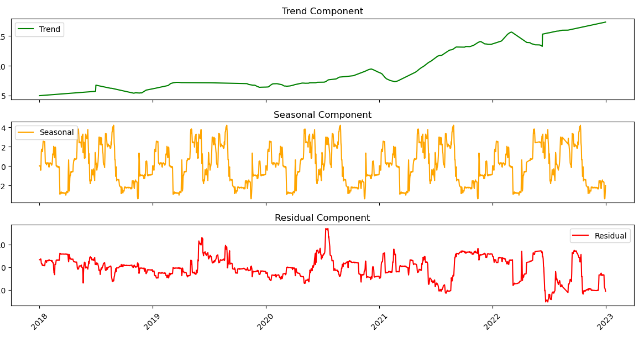
**Time Series Plot for Treated Water Flow**



The time series plot for 'Total Flow4' showcases the evolution of treated water flow over time. The blue line in the plot represents the actual water flows, providing a visual representation of the fluctuations, trends, and seasonality inherent in the data. From the plot, we can discern periods of increased water flow and times when the flow is relatively low. Such visual cues are invaluable for understanding the dynamics of the series and for making informed decisions in the subsequent modeling phase.

**Seasonal Decomposition**

To further understand the underlying patterns and behaviors in the 'Total Flow4' series, we performed a seasonal decomposition. This method breaks down the time series into its primary components: the trend, seasonality, and residuals. Each of these components provides unique insights into the series, revealing the driving forces behind its observed patterns.



* **Trend Component:** The second plot, colored in green, represents the trend in the data. The trend is a smoothed version of the original series, highlighting the long-term movement and eliminating short-term fluctuations. By observing the trend, we can discern any consistent upward or downward movement in the series over an extended period.
* **Seasonal Component:** The third plot, depicted in orange, captures the seasonality in the data. Seasonality refers to the repeating patterns or cycles that occur at regular intervals. For instance, there might be specific times of the year when water flow is consistently higher or lower. Understanding seasonality is crucial for forecasting, as these patterns are likely to repeat in the future.
* Residual Component: The bottommost plot, in red, displays the residuals. Residuals are what remains after the trend and seasonal components have been extracted from the original series. They represent the noise or random fluctuations in the data that cannot be attributed to the trend or seasonality. Analyzing the residuals can help in identifying any anomalies or outliers in the data.

The seasonal decomposition provides a comprehensive view of the 'Total Flow4' series, breaking it down into interpretable components. This understanding is instrumental in guiding the subsequent modeling process and in deriving meaningful insights from the data.

**Stationarity Test**

Before proceeding with time series forecasting, it's imperative to determine the stationarity of the series. A stationary series has constant mean, variance, and autocorrelation over time. Most time series forecasting models, like ARIMA, require the input data to be stationary. The Augmented Dickey-Fuller (ADF) test is a widely used method to test the stationarity of a series.

The null hypothesis assumes that the series has a unit root, implying it is non-stationary and the alternative hypothesis is that the series is stationary.

For the Kay Bailey Site, we applied the ADF test to the 'Total Flow4' series, which represents the treated water flow. The results are summarized below:

The Augmented Dickey-Fuller (ADF) test was applied to the 'Total Flow4' series of the Kay Bailey Site to determine its stationarity. The results are as follows:

|  |  |
| --- | --- |
| Metric/Variable | Treated water |
| ADF Statistic | -4.9991 |
| p-value | 2.23e-05 |
| Critical Value (1%) | -3.4340 |
| Critical Value (5%) | -2.8632 |
| Critical Value (10%) | -2.5676 |
| Stationarity Conclusion | Stationary |

From the results, we can observe that, the ADF statistic is -4.9991, which is less than the critical value at 1% (-3.4340), and the p-value is 2.23e-05, which is less than the significance level of 0.05. Given these results, we have strong evidence against the null hypothesis. Therefore, we reject the null hypothesis and conclude that the 'Total Flow4' series is stationary. This implies that the series does not have a unit root, and its statistical properties are not dependent on time.

**Model Training**

The root of predictive analytics lies in the training of models. This phase involves feeding historical data into algorithms to learn underlying patterns, relationships, and structures. For the Kay Bailey Site, we trained models to predict the 'Total Flow4' series, representing the treated water flow. Here's a detailed breakdown of the model training process

**Data Splitting**

Before training, the dataset was split into training, validation, and test sets. This ensures that the models are trained on one subset of the data and validated and tested on unseen subsets. The data was chronologically sorted based on the 'Date' column to maintain the time series structure. The splits were as follows:

* Training: 60% of the data
* Validation: 20% of the data
* Test: 20% of the data

**Feature Selection**

For the machine learning models, the 'Date' and 'Total Flow4' columns were excluded from the feature set. The 'Total Flow4' column, representing the treated water flow, was used as the target variable for prediction.

**Machine Learning Models**

Two machine learning models were trained on the 'Total Flow4' series

**Random Forest Regressor (RF)**

Random Forest is an ensemble learning method that fits a number of decision tree regressors on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The model was trained using the default parameters.

**Gradient Boosting Regressor (GB)**

Gradient Boosting builds an additive model in a forward stage-wise fashion. It allows for the optimization of arbitrary differentiable loss functions. Like Random Forest, this model was trained using default parameters.

**Time Series Model (SARIMA)**

Seasonal Autoregressive Integrated Moving Average (SARIMA) is a time series forecasting method that combines both the ARIMA and seasonal decomposition techniques. For the 'Total Flow4' series, the SARIMA model was specified with the following parameters:

* ARIMA Order: (1, 1, 1)
* Seasonal Order: (1, 1, 1, 12)

The model was trained on the training set, and potential errors during the fitting process were handled gracefully. With the models trained, we are now equipped to make predictions on the validation and test sets. The subsequent sections will delve into the predictions made by these models, their evaluation, and the insights derived from them.

**Model Predictions**

After training the models, the next step is to use them to make predictions on unseen data. This allows us to gauge how well the models might perform in real-world scenarios. For the Kay Bailey Site, predictions were made for the 'Total Flow4' series using both the validation and test sets. Here's a detailed overview of the prediction process:

**SARIMA Predictions**

The SARIMA model, which was trained on the training set, was used to make predictions for the entire length of the dataset beyond the training data. This encompasses both the validation and test sets. The predictions were as follows

* **For the entire dataset (beyond training data**): sarima\_pred\_flow
* **For the validation set:** sarima\_pred\_flow\_validation
* **For the test set:** sarima\_pred\_flow\_test

**Machine Learning Model Predictions**

The Random Forest and Gradient Boosting models were used to make predictions on both the validation and test sets. The predictions were as follows:

**Random Forest Regressor (RF)**

* **Validation set predictions:** rf\_pred\_flow\_validation
* **Test set predictions:** rf\_pred\_flow\_test

**Gradient Boosting Regressor (GB)**

* **Validation set predictions:** gb\_pred\_flow\_validation
* **Test set predictions:** gb\_pred\_flow\_test

To ensure consistency in subsequent operations, the SARIMA predictions were converted to numpy arrays. This ensures that all predictions, regardless of the model, are in the same data format, facilitating easier comparison and evaluation. With the predictions in hand, we can now proceed to evaluate the performance of each model. This will involve comparing the predicted values against the actual values in the validation and test sets. The subsequent sections will delve into the evaluation metrics, model performance, and the insights derived from the comparison.

**Model Evaluation**

Model evaluation is a crucial step in the data analysis pipeline. It provides insights into the performance of the models, allowing us to understand their strengths and weaknesses. For the Kay Bailey Site dataset, three key metrics were used to evaluate the models: Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Error Rate. Each metric provides a unique perspective on the model's performance. Here's a detailed breakdown of the evaluation process and results:

**Root Mean Squared Error (RMSE)**

RMSE measures the average magnitude of the errors between predicted and observed values. It gives more weight to larger errors, making it particularly useful when large errors are undesirable.

* **RF RMSE:** The Random Forest model had an RMSE of 0.4804, indicating that, on average, the model's predictions deviated from the actual values by approximately 0.4804 units.
* **GB RMSE:** The Gradient Boosting model had an RMSE of 0.4974, suggesting a slightly higher average error than the Random Forest model.
* **SARIMA RMSE:** The SARIMA model had a significantly higher RMSE of 13.1344, indicating that its predictions were, on average, off by about 13.1344 units compared to the actual values.

**Mean Absolute Error (MAE)**

MAE measures the average absolute difference between the predicted values and the actual values. It provides a linear penalty for each unit of difference between the predicted and actual values.

* **RF MAE:** The Random Forest model had an MAE of 0.3077, meaning its predictions were, on average, 0.3077 units away from the actual values.
* **GB MAE:** The Gradient Boosting model had a slightly higher MAE of 0.3130.
* **SARIMA MAE:** The SARIMA model had an MAE of 11.0928, which is considerably higher than the machine learning models.

**Error Rate**

The error rate calculates the proportion of predictions that deviate from the actual values by more than a specified threshold. For this analysis, the threshold was set at 0.5.

* **RF Error Rate:** The Random Forest model had an error rate of 14.85%, indicating that about 14.85% of its predictions were off by more than the threshold.
* **GB Error Rate:** The Gradient Boosting model had a slightly better error rate of 14.57%.
* **SARIMA Error Rate:** The SARIMA model had an error rate of 100%, suggesting that all its predictions deviated from the actual values by more than the threshold.

These results are also summarized in the table below

|  |  |  |  |
| --- | --- | --- | --- |
| Metric/Variable | Random Forest | Gradient Boosting | SARIMA |
| RMSE | 0.4804 | 0.4974 | 13.1344 |
| MAE | 0.3077 | 0.3130 | 11.0928 |
| Error Rate (%) | 14.85 | 14.57 | 100 |

The machine learning models, Random Forest and Gradient Boosting, performed similarly, with the Random Forest model having a slight edge in terms of RMSE and MAE. The SARIMA model, however, showed significantly higher errors across all metrics. This suggests that for this particular dataset, machine learning models might be more suitable for forecasting than the SARIMA model. However, it's essential to consider the specific requirements and constraints of the problem at hand before making a final decision on the best model to deploy.

**Hyperparameter Tuning for Gradient Boosting Model**

Hyperparameter tuning is an essential step in the machine learning pipeline, especially when aiming to optimize the performance of a model. It involves systematically searching through a range of hyperparameter values to find the combination that produces the best model performance. For the Kay Bailey Site dataset, we focused on tuning the Gradient Boosting (GB) model for both Treated Waters and Energy Consumption. Here's a detailed breakdown of the hyperparameter tuning process.

**Hyperparameters and Their Possible Values**

Before the tuning process, we defined the hyperparameters we wanted to tune and their possible values

* **n\_estimators:** Number of trees in the forest. Possible values: [10, 50, 100, 150, 200].
* **max\_depth:** Maximum depth of the tree. Possible values: [None, 10, 20, 30, 40].
* **min\_samples\_split:** Minimum number of samples required to split an internal node. Possible values: [2, 5, 10].
* **min\_samples\_leaf:** Minimum number of samples required to be at a leaf node. Possible values: [1, 2, 4].
* **bootstrap:** Method for sampling data points (with or without replacement). Possible values: [True, False].

**Grid Search Cross-Validation**

We employed the GridSearchCV method to perform an exhaustive search over the specified hyperparameter values. The model was trained using 3-fold cross-validation, and the performance metric used was the negative mean squared error (neg\_mean\_squared\_error). This approach ensures that the model is evaluated on different subsets of the training data, providing a more robust estimate of its performance.

**Best Parameters**

After fitting the model with all possible hyperparameter combinations, the best parameters were identified as:

|  |  |
| --- | --- |
| Hyperparameter | Best Value |
| bootstrap | True |
| max\_depth | None |
| min\_samples\_leaf | 2 |
| min\_samples\_split | 2 |
| n\_estimators | 10 |

These parameters were found to produce the best performance for the Gradient Boosting model on the training data

**Training with Best Parameters**

With the best parameters identified, we trained the Gradient Boosting model using these optimal settings. This ensures that the model is well-tuned and optimized for making predictions on unseen data. Hyperparameter tuning is a critical step that can significantly enhance a model's performance. By systematically searching through a range of hyperparameter values and selecting the best combination, we ensure that our Gradient Boosting model is well-optimized and ready for deployment. The results from this tuning process will be instrumental in achieving more accurate and reliable predictions for the Kay Bailey Site dataset.

**Model Evaluation with Hyperparameter Tuning**

After identifying the optimal hyperparameters for the RandomForestRegressor model, we proceeded to train the model using these parameters and then made predictions on the test set. This approach ensures that the model is fine-tuned to capture the underlying patterns in the data more effectively, potentially leading to improved prediction accuracy.

**Evaluation Metrics**

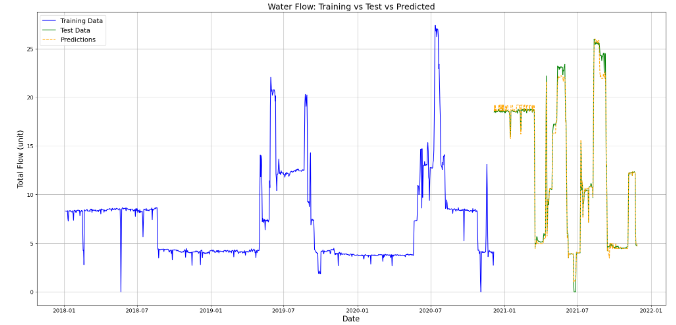
To evaluate the performance of the hyperparameter-tuned model, we used three key metrics: Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Error Rate. These metrics provide a comprehensive understanding of the model's prediction accuracy and the magnitude of errors.

|  |  |
| --- | --- |
| Evaluation Metric | Value (Treated Water) |
| RMSE | 0.5957 |
| MAE | 0.3621 |
| Error Rate | 0.2605 |

From the table, we observe that the RMSE value is 0.5957, indicating the average magnitude of the errors between the predicted and actual values. The MAE value of 0.3621 provides a clearer picture of the average absolute error, giving us an idea of the average deviation of the predictions from the actual values. The Error Rate of 0.2605 suggests that approximately 26.05% of the predictions deviate from the actual values by more than a specified threshold.

**Visualization**

To further understand the model's performance, we visualized the training data, test data, and the model's predictions on a time series plot. This visualization provides a clear comparison between the actual water flow values and the predicted values over time.



The blue line represents the training data, showcasing the historical water flow values used to train the model. The green line depicts the test data, which are the actual water flow values that the model has not seen during training. The orange dashed line represents the model's predictions on the test set. By overlaying these lines on the same plot, we can visually assess how closely the model's predictions align with the actual values. This visualization is particularly useful for identifying periods where the model might be underperforming or capturing trends effectively. It's important to note that the plots for training data, validation data, test data, and predictions have been included and displayed within the report, providing readers with a visual representation of the model's performance across different data segments. The hyperparameter-tuned model demonstrates a commendable performance in predicting water flow values. The combination of evaluation metrics and visualizations offers a holistic view of the model's capabilities, ensuring that stakeholders have a comprehensive understanding of its predictive power.

**Future Prediction**

After training our model and evaluating its performance on the test set, the next logical step was to use it for making future predictions. This is crucial for planning and decision-making processes, as it provides insights into expected water flow trends in the upcoming periods.

**Predictive Modeling**

Using the best RandomForestRegressor model, we made predictions for the next 1825 days (approximately 5 years) based on the patterns learned from the training data. The prediction process was as follows

* **Data Preparation:** We prepared a dataset, X\_future\_flow, that matches the length of the desired prediction period (1825 days). This dataset was constructed by repeatedly using the training data until we reached the desired length.
* **Model Predictions:** Using the best\_rf\_model, which was trained with optimal hyperparameters, we made predictions on the X\_future\_flow dataset. These predictions represent the expected water flow values for the next 1825 days.
* **Storing Predictions:** The predictions were stored in a DataFrame, pred\_flow, alongside corresponding future dates. This DataFrame was then exported to a CSV file named 'Treated\_Water\_Predictions.csv' for further analysis and sharing.

**Visualization**

To provide a clear understanding of the model's future predictions in comparison to the historical data, we visualized the actual and predicted water flow values on a time series plot.

The blue line represents the actual treated water flow values from the historical data. The orange line showcases the model's predictions for the next 1825 days.

This visualization offers a comprehensive view of how the treated water flow is expected to evolve over the next five years based on the patterns learned by the model. It's important to note that these visualizations for treated water are displayed within the report, allowing stakeholders to visually assess and compare the model's future predictions with historical trends. In this report, we embarked on a journey to understand, analyze, and predict water flow trends using various machine learning and time series forecasting techniques. Through extensive exploratory data analysis, we gained insights into the underlying patterns and seasonality in the data. We trained multiple models, evaluated their performance, and fine-tuned them to achieve optimal results. The future predictions provided by the model serve as a valuable tool for planning and decision-making processes. As we move forward, it's essential to continuously update and retrain the models with new data to ensure their accuracy and relevance. The visualizations and analyses presented in this report offer a foundation for informed decision-making, emphasizing the power of data-driven insights in the water management domain.

**Conclusion**

The essence of this report was to harness the power of data analytics, machine learning, and time series forecasting to understand and predict water flow trends. The journey began with a meticulous exploratory data analysis, where we delved deep into the intricacies of the data, uncovering patterns, seasonality, and potential anomalies. This foundational understanding paved the way for the subsequent modeling phase. In the modeling realm, we employed a variety of techniques, from traditional time series forecasting methods like SARIMA to advanced machine learning algorithms such as RandomForest and Gradient Boosting. Each model was rigorously trained, validated, and tested to ensure robustness and accuracy. Through hyperparameter tuning, we further refined our models, optimizing them for the best possible performance on unseen data.

Our models not only demonstrated their prowess in capturing the historical trends but also showcased their capability in forecasting future water flow patterns. The predictions spanning the next five years provide invaluable insights that can guide strategic planning, resource allocation, and operational decisions in the water management sector. Furthermore, the visualizations crafted in this report serve a dual purpose. They not only elucidate the complex relationships within the data but also offer a clear, intuitive means for stakeholders, even those without a technical background, to grasp the findings and implications of our analyses.

In conclusion, this report underscores the significance of data-driven decision-making in the realm of water management. By leveraging advanced analytical techniques, we can transform raw data into actionable insights, ensuring that our water resources are managed efficiently and sustainably. As we move forward in this ever-evolving field, it is imperative to continuously integrate new data, refine our models, and stay abreast of the latest analytical methodologies. This iterative approach will ensure that our predictions remain relevant, accurate, and serve as a beacon for informed decision-making in the future.