# Water Consumption Prediction

## 1. Introduction

## This project focuses on predicting water consumption using machine learning. The goal is to develop a model that can analyze various factors influencing water usage, such as temperature, humidity, apartment type, income level, and appliance usage. By leveraging predictive analytics, the model helps in optimizing water distribution and improving resource management.

## 2. Approach

## The solution was developed through the following steps:

## Data Preprocessing: Cleaned the data by handling missing values, encoding categorical variables, and normalizing numerical values for consistency.

## Exploratory Data Analysis (EDA): Analyzed the data to discover trends, patterns, and relationships between variables.

## Feature Engineering: Created new features from existing data to improve model accuracy and performance.

## Model Selection & Training: Tested different machine learning models like Linear Regression, Random Forest, and XGBoost to find the best fit for the data.

## Evaluation & Metrics: Measured model performance using metrics like Mean Squared Error (MSE) and R² score to assess its predictive accuracy.

## Prediction & Submission: Generated final predictions and saved them for future use or deployment in the system.

**3. Technologies and Tools used**

* **Python:** Programming language used for data analysis and model implementation.
* **Pandas:** Data manipulation and preprocessing.
* **NumPy:** Numerical computations and array handling.
* **Scikit-learn:** Machine learning model training and evaluation.
* **XGBoost:** Final model for prediction, known for efficiency and performance.
* **Matplotlib & Seaborn:** Data visualization and EDA.
* **Jupyter Notebook:** Interactive development and testing environment.

4. **Data Preprocessing**

**Handling Missing Values**

* Numerical Features: Missing values were identified using isnull().sum() and filled with the median to maintain consistency in the data. Data with too many missing values was removed.

### Categorical Features: Missing values in categorical data were filled using the most frequent value (mode) to preserve data integrity.

### Timestamp Feature: The timestamp was converted to a proper datetime format, and useful time-based features like hour, day, and month were extracted for analysis.

### Feature Scaling & Encoding

## Categorical Features: Converted to numerical values using Label Encoding, making them compatible with machine learning models.

## Numerical Features: Scaled using MinMax Scaling to ensure all features have equal importance and are on the same scale.

## 5. Exploratory Data Analysis (EDA)

* **Dataset Overview:** Explored the dataset by checking its structure, column types, and summary statistics using .info() and .describe().
* **Missing Values Analysis:** Identified missing values and visualized them to understand their distribution.
* **Correlation Analysis:** Analyzed relationships between variables using correlation matrices to identify strong correlations and patterns.
* **Visualization:**
  + **Heatmap**: Displayed correlation between features and water consumption.
  + **Pairplots**: Explored feature relationships through scatter plots
  + **Distribution of Water Consumption (Histograms)**: Showed the frequency distribution of water usage.

## 6. Model Selection

### Models Evaluated:

* **Linear Regression**: Baseline model, but underperformed due to non-linearity in the data.
* **Random Forest**: Provided better performance and reduced overfitting.
* **XGBoost (Final Model)**: Outperformed other models due to its efficiency in handling complex data.

### Performance Metrics:

* **Mean Squared Error (MSE)**: Evaluated the average squared difference between actual and predicted values.
* **R² Score**: Measured how well the model explains variance in water consumption.

The best-performing XGBoost model achieved an R² score of **90.87%** and Mean Squared Error (MSE) as **77.4%**, indicating strong predictive accuracy.

## 7. Final Model & Submission

* The final model was trained using XGBoost, and predictions were made on the test dataset.
* A submission file (submission.csv) containing the predicted water consumption values alongside their corresponding timestamps.

```python

Import pandas as pd

submission = test\_df[['Timestamp']]

submission['Water\_Consumption'] = xgb\_model.predict(test\_df.drop(columns=['Timestamp'], errors='ignore'))

submission.to\_csv("submission.csv", index=False)

```

## 8. How to Run the Code

### Setup Environment

Ensure you have the necessary dependencies installed:

pip install pandas numpy scikit-learn xgboost matplotlib seaborn

Execution Steps

1. Load and Preprocess Data

```python

Import pandas as pd

train\_df = pd.read\_csv('train.csv')

test\_df = pd.read\_csv('test.csv')

```

2. Train the Model

```python

xgb\_model.fit(X\_train, y\_train)

```

3. Generate Predictions

```python

predictions = xgb\_model.predict(X\_test)

```

4. Save Submission

```python

submission.to\_csv("submission.csv", index=False)

```

## 9. Conclusion

This project successfully predicts water consumption using machine learning. By utilizing advanced models like XGBoost and conducting thorough data analysis, we achieved significant predictive accuracy. With further refinements in feature engineering, model tuning, and deep learning integration, this approach can be extended to real-world applications in water resource management.