

# Project 4: Measure Energy Consumption

## PHASE-3 PROJECT SUBMISSION

**Team Member:** Jeswin Bradman B

**Reg No:** 311121205029

### Objective:

The objective of this project is to create an automated system that measures energy consumption, analyzes the data, and provides visualizations for informed decision-making. This solution aims to enhance efficiency, accuracy, and ease of understanding in managing energy consumption across various sectors.

In this phase we concentrate on loading our dataset and removing duplicate values. Finally preprocessing the data generated.

### Loading data:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler

# Load the AEP Hourly Energy Consumption dataset
data = pd.read_csv('AEP_hourly.csv')
data
```

### Output:

Out[18]:

	Datetime	AEP_MW
0	2004-12-31 01:00:00	13478.0
1	2004-12-31 02:00:00	12865.0
2	2004-12-31 03:00:00	12577.0
3	2004-12-31 04:00:00	12517.0
4	2004-12-31 05:00:00	12670.0
...	...	...
121268	2018-01-01 20:00:00	21089.0
121269	2018-01-01 21:00:00	20999.0
121270	2018-01-01 22:00:00	20820.0
121271	2018-01-01 23:00:00	20415.0
121272	2018-01-02 00:00:00	19993.0

121273 rows × 2 columns

## Removing Duplicates and Description:

### # Data Cleaning

```
data.dropna(inplace=True) # Drop rows with missing values
data.drop_duplicates(inplace=True) # Remove duplicate rows, if any
print(data.describe())
```

### Output:

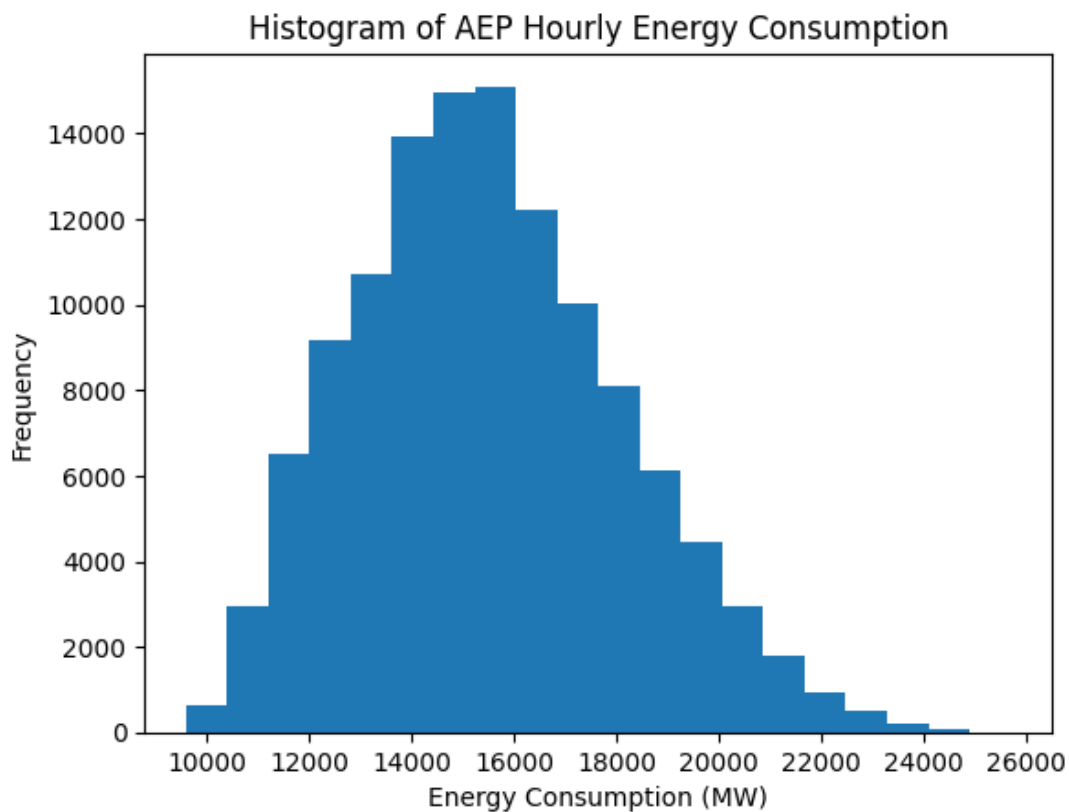
```
In [22]: # Data Cleaning
data.dropna(inplace=True) # Drop rows with missing values
data.drop_duplicates(inplace=True) # Remove duplicate rows, if any
print(data.describe())
```

	AEP_MW
count	121273.000000
mean	15499.513717
std	2591.399065
min	9581.000000
25%	13630.000000
50%	15310.000000
75%	17200.000000
max	25695.000000

## Plotting:

```
plt.hist(data['AEP_MW'], bins=20)
plt.xlabel('Energy Consumption (MW)')
plt.ylabel('Frequency')
plt.title('Histogram of AEP Hourly Energy Consumption')
plt.show()
```

## Output:



## Data Preprocessing:

```
# Data Splitting (if for machine learning)
X = data.drop('AEP_MW', axis=1) # Assuming 'AEP_MW' is the target
variable
y = data['AEP_MW']
```

```

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Drop non-numeric columns before scaling
non_numeric_columns = ['Datetime'] # Add the names of non-numeric
columns here
X_train = X_train.drop(columns=non_numeric_columns)
X_test = X_test.drop(columns=non_numeric_columns)

# Convert DataFrames to NumPy arrays
X_train = X_train.to_numpy()
X_test = X_test.to_numpy()

# Normalization/Scaling (if needed)
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Save Preprocessed Data
data.to_csv('preprocessed_AEP_hourly.csv', index=False)

```

## **Conclusion:**

Finally, We are saving all the preprocessed data into a new csv file `preprocessed_AEP_hourly.csv`. This is the file that we are going to use for the future analysis as we have removed all the duplicate values.