Measure energy consumption

Team member Hariharan.G (au210221104009)

Phase 3: Development Part 1

Topic: Start building the model to measure energy consumption by loading and pre-processing the dataset.

Introduction

- In the quest for a sustainable and efficient energy future, machine learning has emerged as a powerful tool for measuring and optimizing energy consumption. The journey towards harnessing the potential of machine learning begins with dataset loading and preprocessing, which forms the critical groundwork for accurate predictions and actionable insights.
- This introductory phase involves gathering diverse data sources, refining their quality, and shaping them into a format that machine learning models can comprehend. By effectively mastering the art of dataset loading and preprocessing, we unlock the door to enhanced energy efficiency, demand forecasting, and informed decision-making in our increasingly energy-conscious world.

Given dataset

(AEP_hourly)

121273x2(121273 rows and 2 column)

```
==================================================
First twenty Rows

             Datetime  AEP_MW
0   31-12-2004 01:00   13478
1   31-12-2004 02:00   12865
2   31-12-2004 03:00   12577
3   31-12-2004 04:00   12517
4   31-12-2004 05:00   12670
5   31-12-2004 06:00   13038
6   31-12-2004 07:00   13692
7   31-12-2004 08:00   14297
8   31-12-2004 09:00   14719
9   31-12-2004 10:00   14941
10  31-12-2004 11:00   15184
11  31-12-2004 12:00   15009
12  31-12-2004 13:00   14808
13  31-12-2004 14:00   14522
14  31-12-2004 15:00   14349
15  31-12-2004 16:00   14107
16  31-12-2004 17:00   14410
17  31-12-2004 18:00   15174
18  31-12-2004 19:00   15261
19  31-12-2004 20:00   14774

==================================================
```

Importance of loading and processing dataset:

Data Quality Assurance: Loading and preprocessing a dataset for measuring energy consumption using machine learning is crucial for ensuring data quality. This involves identifying and addressing missing values, handling outliers, and transforming data into a consistent format. For example, Python code using Pandas can be employed to fill missing values with 'SimpleImputer' and identify outliers using statistical methods like Z-scores or IQR.

Model Performance: Proper dataset loading and preprocessing significantly impact the performance of machine learning models. Techniques such as feature scaling, encoding categorical variables, and feature engineering help models make accurate predictions.

Python code using libraries like Scikit-Learn can standardize features using 'StandardScaler' and one-hot encode categorical data with 'get_dummies'

Efficient Analysis: Loading and preprocessing the dataset prepares it for efficient analysis. By converting data into a format that machine learning algorithms can understand, it becomes possible to extract valuable insights, make predictions, and optimize energy consumption effectively. This efficient analysis contributes to better decision-making and more sustainable energy management.

Challenges involved while loading and preprocessing dataset:

- Loading and preprocessing datasets for measuring energy consumption using machine learning can be a complex task, and it comes with several challenges. Some common challenges include:
- **Data Quality**: Ensuring the data's quality is a significant challenge. This includes dealing with missing values, errors, and inconsistencies in the dataset, which can lead to incorrect modeling results.

- **Outlier Detection**: Identifying and handling outliers is crucial in energy consumption data. Incorrect handling of outliers can lead to models with poor generalization.
- **Feature Engineering**: Creating meaningful features from raw data requires domain knowledge and can be time-consuming. Deciding which features to engineer and how to engineer them effectively is a challenge.

How to overcome the challenges while loading and preprocessing dataset:

- Overcoming challenges in loading and preprocessing datasets for measuring energy consumption using machine learning requires a combination of best practices, domain knowledge, and the use of appropriate tools. Here are some strategies to tackle these challenges:

- **Data Quality**:
  - Imputation: Use techniques like mean, median, or mode imputation to address missing data.
  - Data Validation: Implement data validation checks to detect and correct errors and inconsistencies in the dataset.
- **Outlier Detection**:

- Use statistical methods or machine learning algorithms to identify and handle outliers appropriately. Techniques like z-scores, isolation forests, or local outlier factor (LOF) can be useful.

► **Feature Engineering**:

- Collaborate with domain experts to create meaningful features that capture important relationships and patterns.
- Utilize automated feature selection and dimensionality reduction techniques to streamline feature engineering.

Loading dataset

❖ Loading the dataset using machine learning is the process of bringingthe data into the machine learning environment so that it can be usedto train and evaluate a model.

❖ The specific steps involved in loading the dataset will vary dependingon the machine learning library or framework that is being used.However, there are some general steps that are common to mostmachine learning frameworks:

❖ **Identify the dataset:**

- The first step is to identify the dataset that you want to load. This dataset may be stored in a local file, in a database, or in cloud storageservice.

❖ **Load the dataset:**

- Once you have identified the dataset, you need to load it into the machine learning environment. This may involve using a built-infunction in the machine learning library, or it may involve writing yourown code.

❖ **Preprocess the dataset:**

- Once the dataset is loaded into the machine learning environment,you may need to preprocess it before you can start training and evaluating your model. This may involve cleaning the data, transforming the data into a suitable format, and splitting the data into training and test sets.

Here, how to load a dataset using machine learning in Python

Program:

Import packages

```
import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns
```

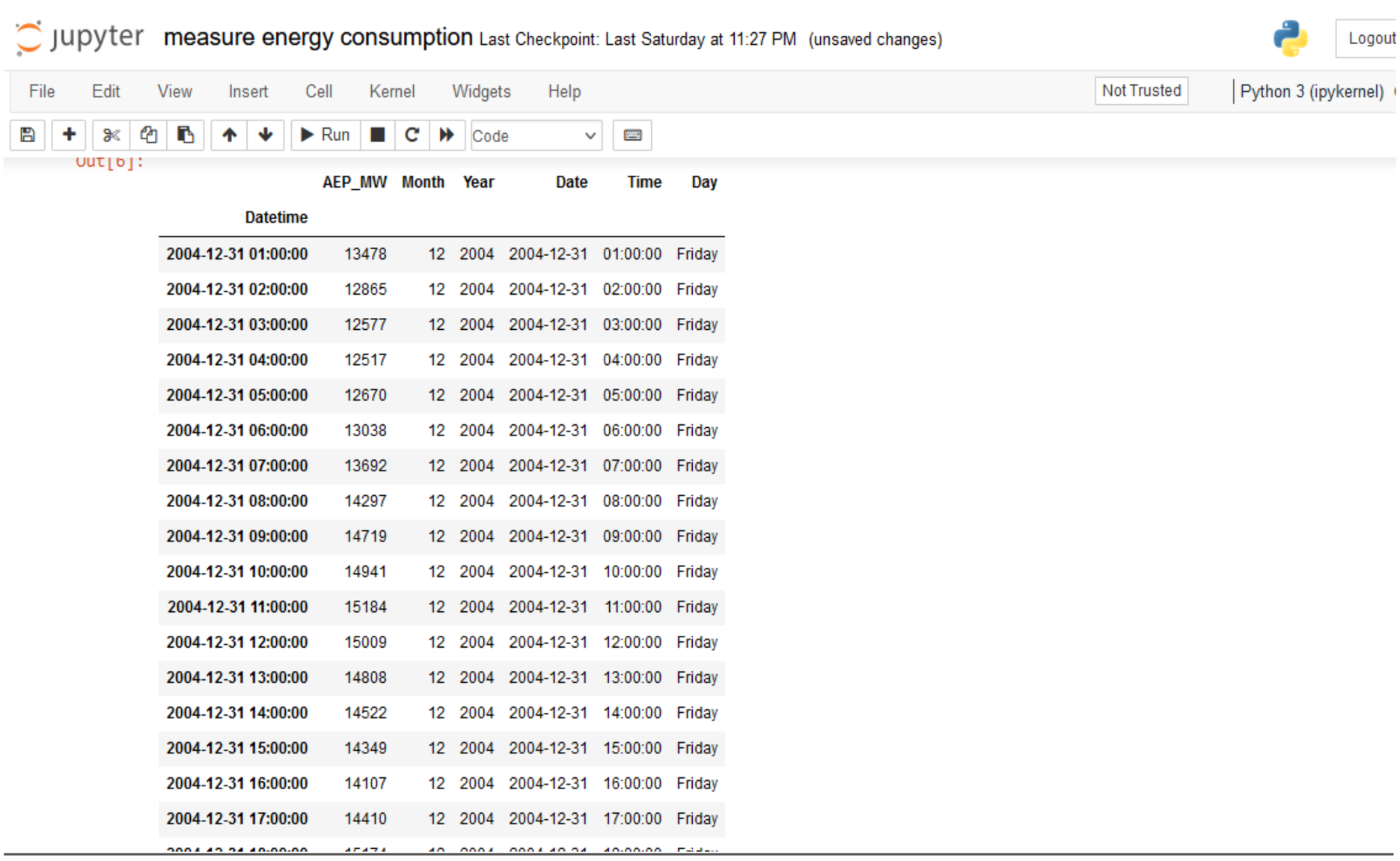
```
import pprint
%matplotlib inline
```

Loading Dataset:

```
df=pd.read_csv(r"C:\Users\WELCOME\Documents\datasets\AEP
_hourly.csv")
```

Reformat the Date Time Columns:

Dataset:



jupyter measure energy consumption Last Checkpoint: Last Saturday at 11:27 PM (unsaved changes)

Logout

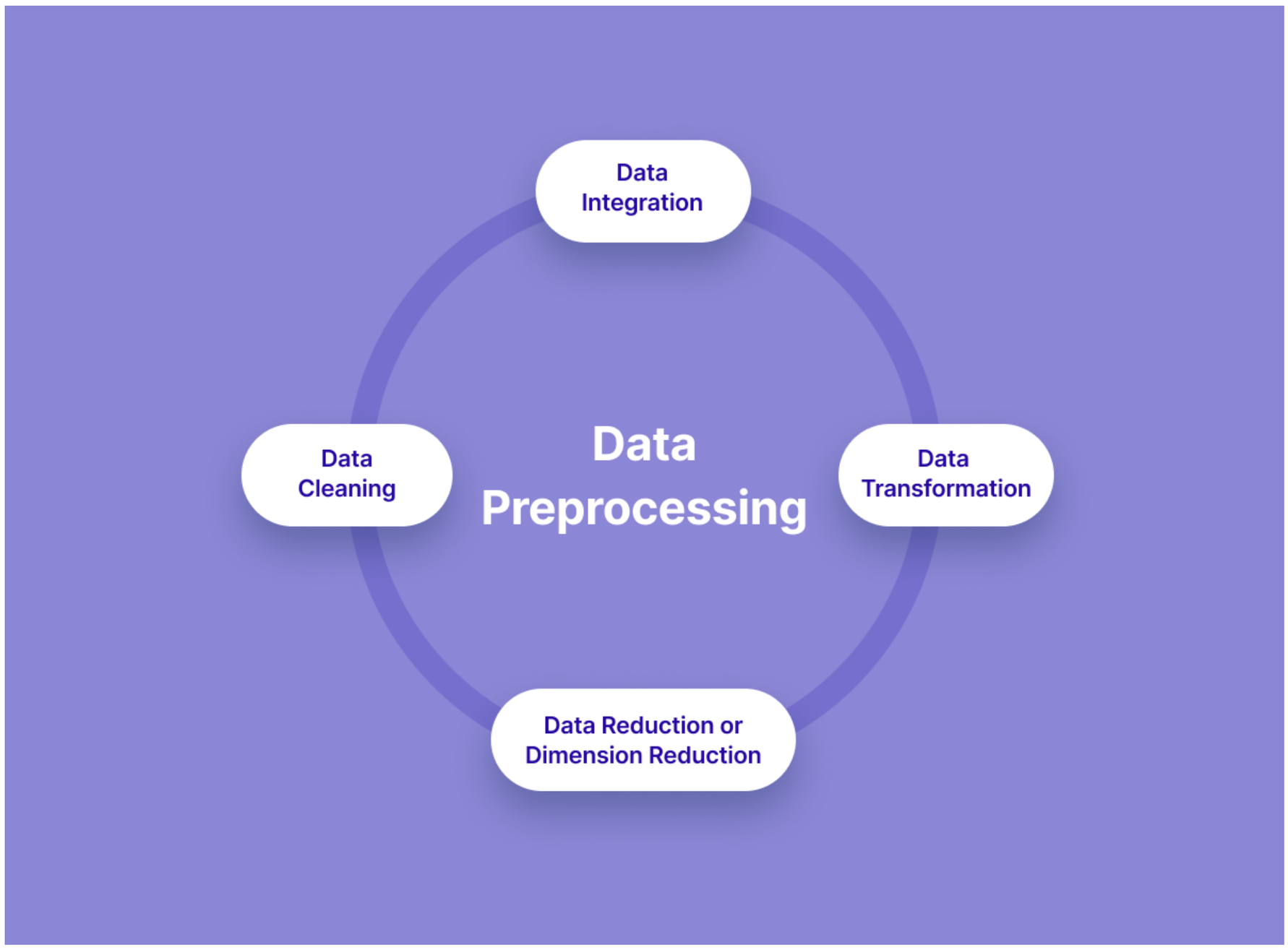
File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

Run Code

Out[6]:

Datetime	AEP_MW	Month	Year	Date	Time	Day
2004-12-31 01:00:00	13478	12	2004	2004-12-31	01:00:00	Friday
2004-12-31 02:00:00	12865	12	2004	2004-12-31	02:00:00	Friday
2004-12-31 03:00:00	12577	12	2004	2004-12-31	03:00:00	Friday
2004-12-31 04:00:00	12517	12	2004	2004-12-31	04:00:00	Friday
2004-12-31 05:00:00	12670	12	2004	2004-12-31	05:00:00	Friday
2004-12-31 06:00:00	13038	12	2004	2004-12-31	06:00:00	Friday
2004-12-31 07:00:00	13692	12	2004	2004-12-31	07:00:00	Friday
2004-12-31 08:00:00	14297	12	2004	2004-12-31	08:00:00	Friday
2004-12-31 09:00:00	14719	12	2004	2004-12-31	09:00:00	Friday
2004-12-31 10:00:00	14941	12	2004	2004-12-31	10:00:00	Friday
2004-12-31 11:00:00	15184	12	2004	2004-12-31	11:00:00	Friday
2004-12-31 12:00:00	15009	12	2004	2004-12-31	12:00:00	Friday
2004-12-31 13:00:00	14808	12	2004	2004-12-31	13:00:00	Friday
2004-12-31 14:00:00	14522	12	2004	2004-12-31	14:00:00	Friday
2004-12-31 15:00:00	14349	12	2004	2004-12-31	15:00:00	Friday
2004-12-31 16:00:00	14107	12	2004	2004-12-31	16:00:00	Friday
2004-12-31 17:00:00	14410	12	2004	2004-12-31	17:00:00	Friday

Preprocessing the dataset



- Data preprocessing is the process of cleaning, transforming, and integrating data in order to make it ready for analysis.
- This may involve removing errors and inconsistencies, handlingmissing values, transforming the data into a consistent format, and scaling the data to a suitable range.
- **Data cleaning**: This involves identifying and correcting errors and inconsistencies in the data. For example, this may involve removing duplicate records, correcting typos, and filling in missing values.

- **Data transformation**: This involves converting the data into a format that is suitable for the analysis task. For example, this may involve converting categorical data to numerical data, or scaling the data to a suitable range.
- **Feature engineering**: This involves creating new features from the existing data. For example, this may involve creating features that represent interactions between variables, or features that represent summary statistics of the data.
- **Data integration**: This involves combining data from multiple sources into a single dataset. This may involve resolving inconsistencies in the data, such as different data formats ordifferent variable names.
- Data preprocessing is an essential step in many data science projects. By carefully preprocessing the data, data scientists can improve the accuracy and reliability of their results

Program

Import packages

```
import pandas as pd
```

```
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import pprint
%matplotlib inline
```

Loading Dataset:

```
df=pd.read_csv(r"C:\Users\WELCOME\Documents\datasets\AEP_hourly.csv")
```

Exploratory Data Analysis:

```
print("="*50)

print("Information About Dataset","\n")

print(df.info(),"\n")

print("="*50)

print("Describe the Dataset ","\n")

print(df.describe(),"\n")

print("="*50)
```

```
print("Null Values t ","\n")

print(df.isnull().sum(),"\n")
```

Resampleing Data

```
NewDataSet = dataset.resample('D').mean()

print("Old Dataset ",dataset.shape )

print("New  Dataset ",NewDataSet.shape )

TestData = NewDataSet.tail(100)

Training_Set = NewDataSet.iloc[:,0:1]

Training_Set = Training_Set[:-60]

print("Training Set Shape ", Training_Set.shape)

print("Test Set Shape ", TestData.shape)

Training_Set = Training_Set.values

sc = MinMaxScaler(feature_range=(0, 1))

Train = sc.fit_transform(Training_Set)

X_Train = []
```

```
Y_Train = []

# Range should be fromm 60 Values to END

for i in range(60, Train.shape[0]):

    # X_Train 0-59

    X_Train.append(Train[i-60:i])

    # Y Would be 60 th Value based on past 60 Values

    Y_Train.append(Train[i])

# Convert into Numpy Array

X_Train = np.array(X_Train)

Y_Train = np.array(Y_Train)

print(X_Train.shape)

print(Y_Train.shape)

# Shape should be Number of [Datapoints , Steps , 1 )

# we convert into 3-d Vector or #rd Dimesnsion
```

```
X_Train = np.reshape(X_Train, newshape=(X_Train.shape[0],
X_Train.shape[1], 1))

X_Train.shape
```

Output Exploratory Data Analysis:

```
==================================================
Information About Dataset

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 121273 entries, 0 to 121272
Data columns (total 2 columns):
Datetime    121273 non-null object
AEP_MW      121273 non-null float64
dtypes: float64(1), object(1)
memory usage: 1.9+ MB
None

==================================================
Describe the Dataset

               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

==================================================
Null Values t

Datetime    0
AEP_MW      0
dtype: int64
```

Conclusion

In conclusion, dataset loading and preprocessing are the indispensable cornerstones of any machine learning endeavor aimed at measuring energy consumption. These initial steps, from data

collection to data cleaning, feature engineering, and scaling, lay the foundation upon which accurate and insightful predictions are built. Careful data curation and meticulous preparation of the dataset are essential for ensuring the reliability and efficacy of machine learning models in the realm of energy consumption. By adhering to best practices in these processes, practitioners can harness the full potential of machine learning to optimize energy usage, predict demand, and make more informed decisions in the ever-evolving energy landscape.