KINGS ENGINEERING COLLEGE

ELECTRICITY PRICE PREDICTION

Department: B.Tech. Information Technology

Batch No: 07

Domain: Applied Data Science

Topic: Electricity Prices - Prediction

Team Members:

Akash.M (210821205007)

Kumar.P (210821205048)

Lingeshwaran.R (210821205049)

keerthivasan.V (210821205060)

Team Mentor: Ms. D. Rejees Jenifa

PHASE 3

Building an electricity price prediction model involves several steps, including loading and preprocessing the dataset. Here's a step-by-step guide

Import Necessary Libraries:

First, you'll need to import the necessary Python libraries for data manipulation, visualization, and modeling. Common libraries include pandas, numpy, matplotlib, and scikit-learn.

PYTHON CODE:

```
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
```

1. Load the Dataset:

Assuming that your dataset is stored in a CSV file (let's call it your_dataset.csv'), you can load it using the `read_csv` function in Pandas:

```
#data loading
df=pd.read_csv("/content/sample_data/electricity.gui (1).zip")
df.head()
```

	DateTime	Holiday	HolidayFlag	DayOfWeek	WeekOfYear	Day	Month	Year	PeriodOfDay	ForecastWindProduction	SystemLoadEA	SMPEA	ORKTemperature	O
0	01/11/2011 00:00	None	0	1	44	1	11	2011	0	315.31	3388.77	49.26	6.00	
1	01/11/2011 00:30	None	0	1	44	1	11	2011	1	321.80	3196.66	49.26	6.00	
2	01/11/2011 01:00	None	0	1	44	1	11	2011	2	328.57	3060.71	49.10	5.00	
3	01/11/2011 01:30	None	0	1	44	1	11	2011	3	335.60	2945.56	48.04	6.00	
4	01/11/2011 02:00	None	0	1	44	1	11	2011	4	342.90	2849.34	33.75	6.00	

2.Explore the Dataset:

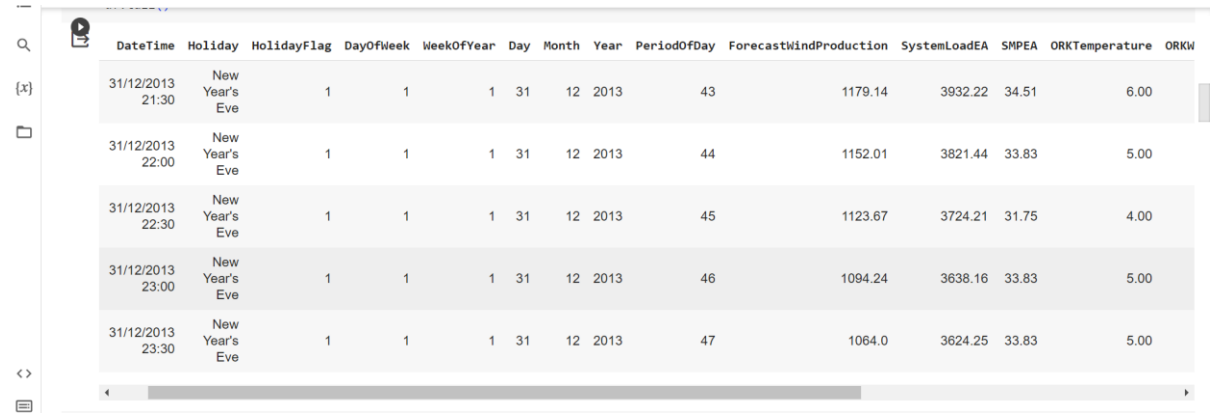
Now that you've loaded the dataset, you should explore it to understand its structure and contents. You can start by looking at the first few rows of the dataset:

Python Code:

```
df . shape
```

```
(38014, 18)
```

```
df . tail()
```



DateTime	Holiday	HolidayFlag	DayOfWeek	WeekOfYear	Day	Month	Year	PeriodOfDay	ForecastWindProduction	SystemLoadEA	SMPEA	ORKTemperature	ORKW
31/12/2013 21:30	New Year's Eve	1	1	1	31	12	2013	43	1179.14	3932.22	34.51	6.00	
31/12/2013 22:00	New Year's Eve	1	1	1	31	12	2013	44	1152.01	3821.44	33.83	5.00	
31/12/2013 22:30	New Year's Eve	1	1	1	31	12	2013	45	1123.67	3724.21	31.75	4.00	
31/12/2013 23:00	New Year's Eve	1	1	1	31	12	2013	46	1094.24	3638.16	33.83	5.00	
31/12/2013 23:30	New Year's Eve	1	1	1	31	12	2013	47	1064.0	3624.25	33.83	5.00	

3.Preprocess the Data:

Preprocessing involves tasks like handling missing values, encoding categorical variables, and scaling numerical features.

Handle Missing Values:

missing value query

```
df.isna().sum()
```

```
WeekOfYear                0
Day                       0
Month                     0
Year                      0
PeriodOfDay               0
ForecastWindProduction    0
SystemLoadEA              0
SMPEA                     0
ORKTemperature            0
ORKWindspeed              0
```

```
CO2Intensity            0
ActualWindProduction    0
SystemLoadEP2           0
SMPEP2                  0
dtype: int64
```

Encode Categorical Variables:

Encoding categorical variables is an essential step in preparing data for machine learning models, as most algorithms require numerical inputs.

```
#create a list for numeric and categorical values
cat_list=[]
num_list=[]
for i in df.columns:
    unique_val=len(df[i].unique())
    if unique_val<40:
        cat_list.append(i)
    else:
        num_list.append(i)
cat_list.append("WeekOfYear")
cat_list
```

OUTPUT:

```
['Holiday',
 'HolidayFlag',
 'DayOfWeek',
 'Day',
 'Month',
 'Year',
```

```
 'ORKTemperature',
 'WeekOfYear']
# distributions of numeric attributes
# distributions of numeric attributes
num_list.remove("DateTime")
num_list
```

OUTPUT:

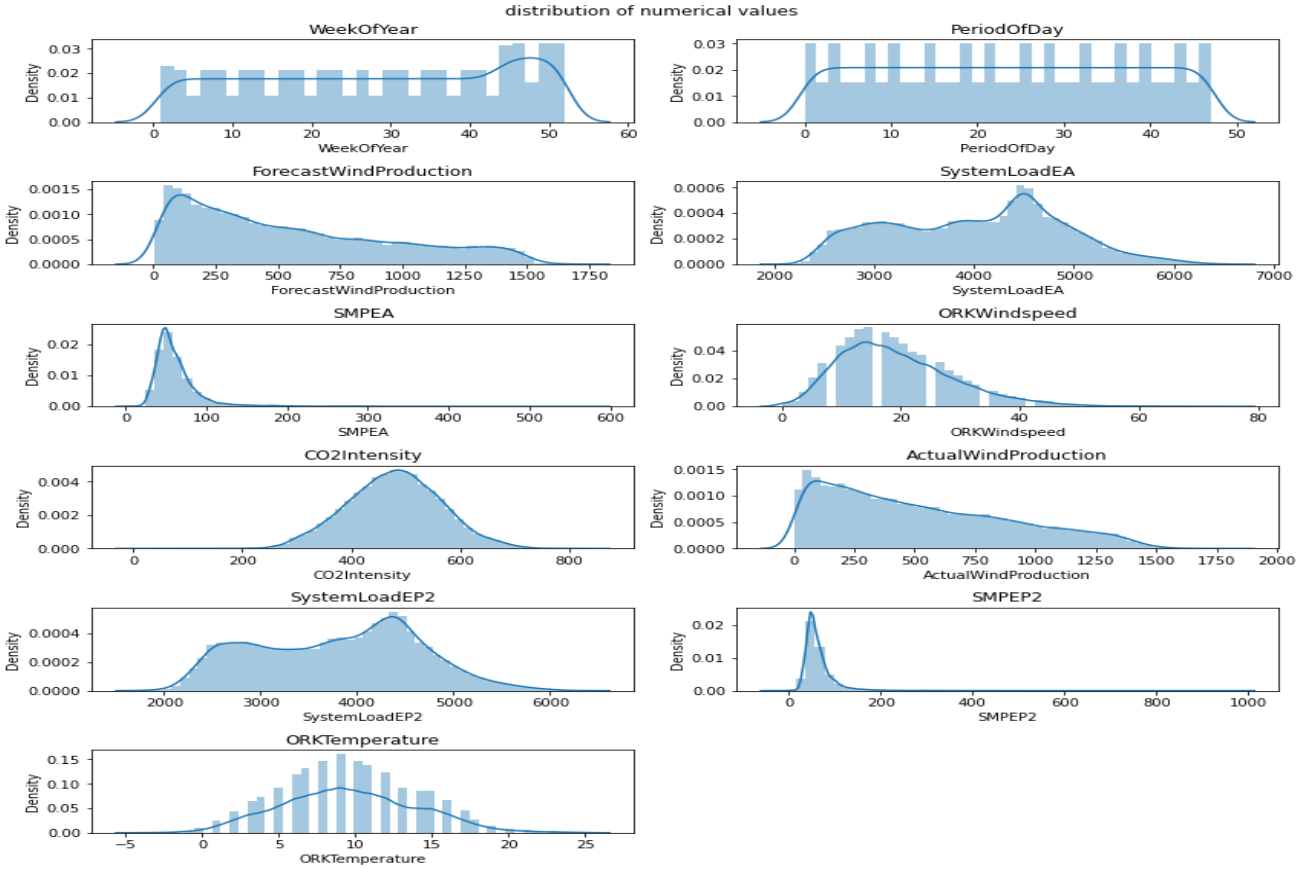
```
['WeekOfYear',
 'PeriodOfDay',
 'ForecastWindProduction',
 'SystemLoadEA',
 'SMPEA',
 'ORKWindspeed',
 'CO2Intensity',
 'ActualWindProduction',
 'SystemLoadEP2',
 'SMPEP2']
```

Scale Numerical Features:

Scaling numerical features is an important preprocessing step in many machine learning algorithms. It helps ensure that all features contribute equally to the model's training process, preventing features with larger scales from dominating the learning process.

```
num_list.append("ORKTemperature")
k=1
plt.figure(figsize=(12,12))
```

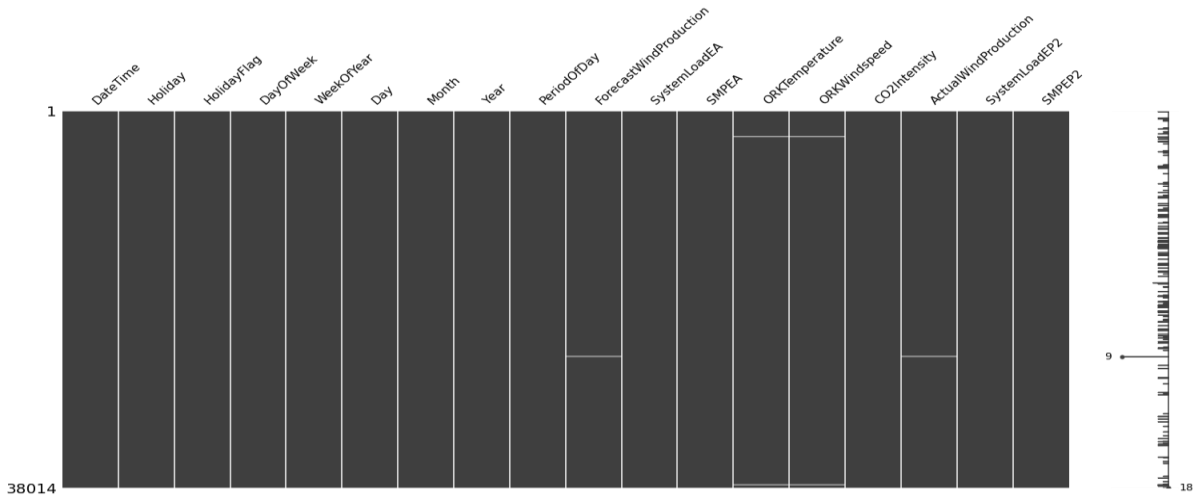
```
plt.suptitle("distribution of numerical values")
for i in df.loc[:,num_list]:
    plt.subplot(6,2,k)
    sns.distplot(df[i])
    plt.title(i)
 k+=1
    plt.tight_layout()
```



Visualization of missing values:

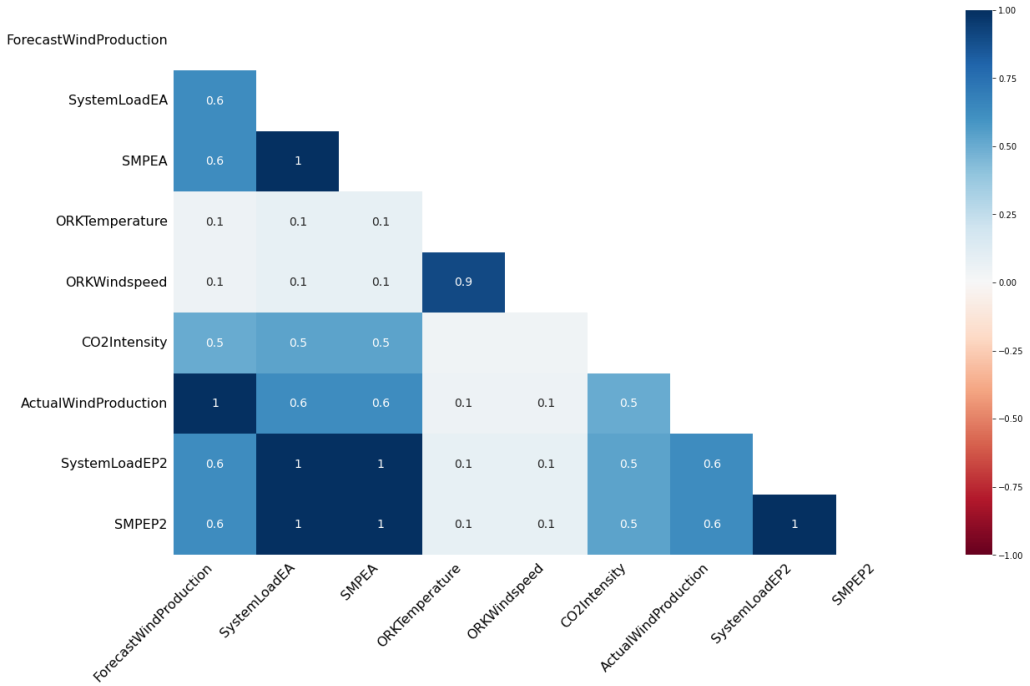
Visualizing missing data is crucial for understanding the extent and patterns of missingness in a dataset. It helps in making informed decisions about how to handle missing values.

```
import missingno as msno
msno.matrix(df);
```



let's visualize whether there is a relationship between the missing values

msno.heatmap(df);

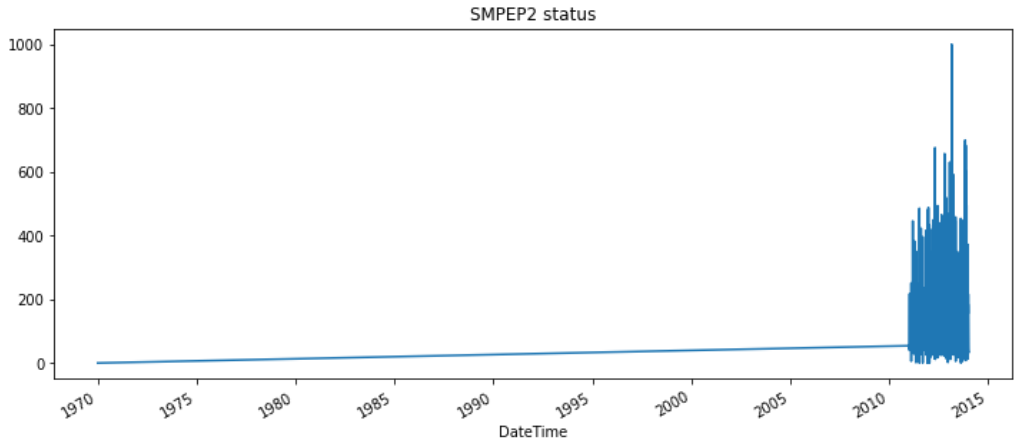


Different Analysis:

Time Series Analysis:

Time series analysis is a statistical technique used to analyze and extract meaningful information from time-ordered data points.

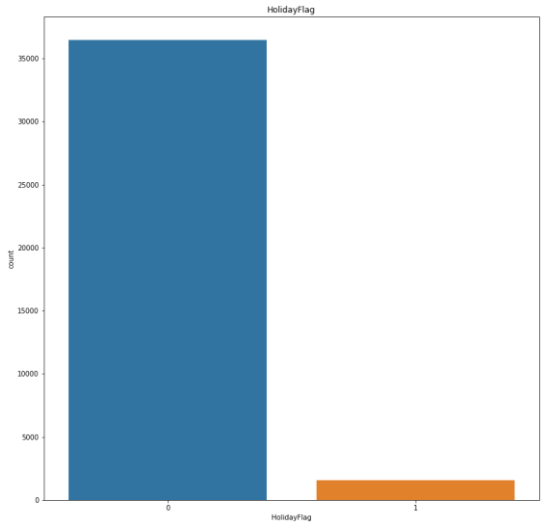
```
from datetime import datetime
df["DateTime"] = pd.to_datetime(df.DateTime)
df['year'] = df['DateTime'].dt.year
df['month'] = df['DateTime'].dt.month
df["day"]=df["DateTime"].dt.day

# We have created 3 new columns
# we can start our time series analysis
# change of real price of consumed electricity with time

custgroup=df.groupby('DateTime').mean()
plt.figure(figsize=(12,5))
custgroup['SMPEP2'].plot(x=df.DateTime)
plt.figure(figsize=(12,5))
custgroup['SMPEP2'].plot(x=df.DateTime)
plt.title("SMPEP2 status")
plt.show()
```

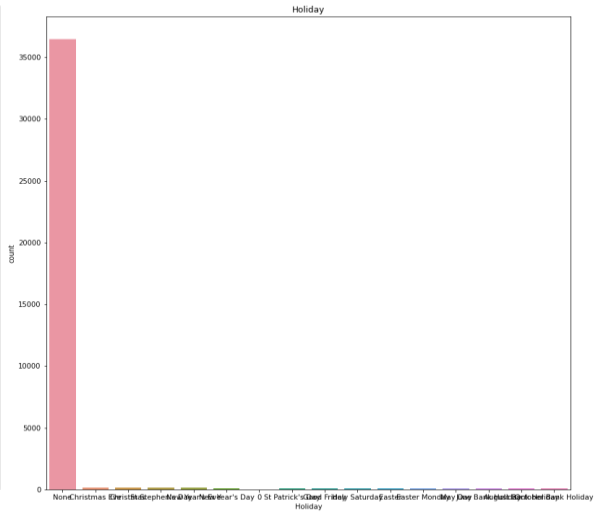


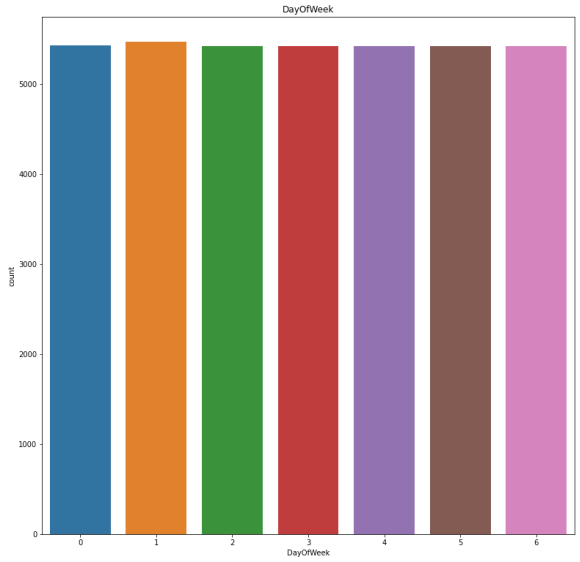
Data Visualize:

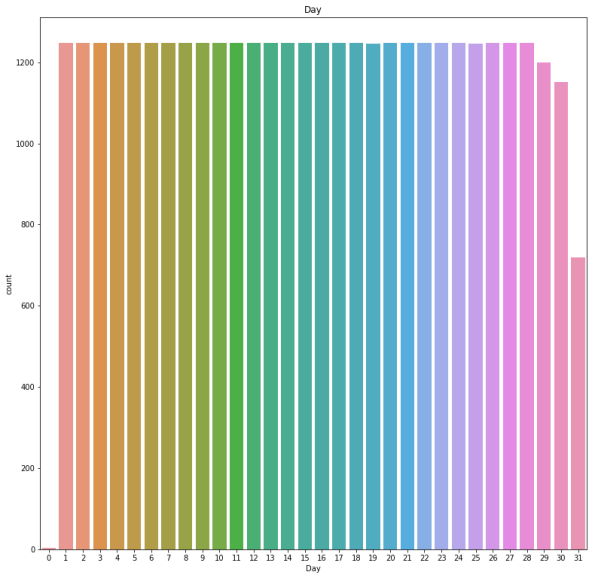
Data visualization is a critical part of the data analysis process. It helps in understanding the underlying patterns, trends, and relationships in the data.

```
# Categorical Analysis
for i in cat_list:
   plt.figure(figsize=(13,13))
    sns.countplot(x=i,data=df.loc[:,cat_list])
    plt.title(i)
```





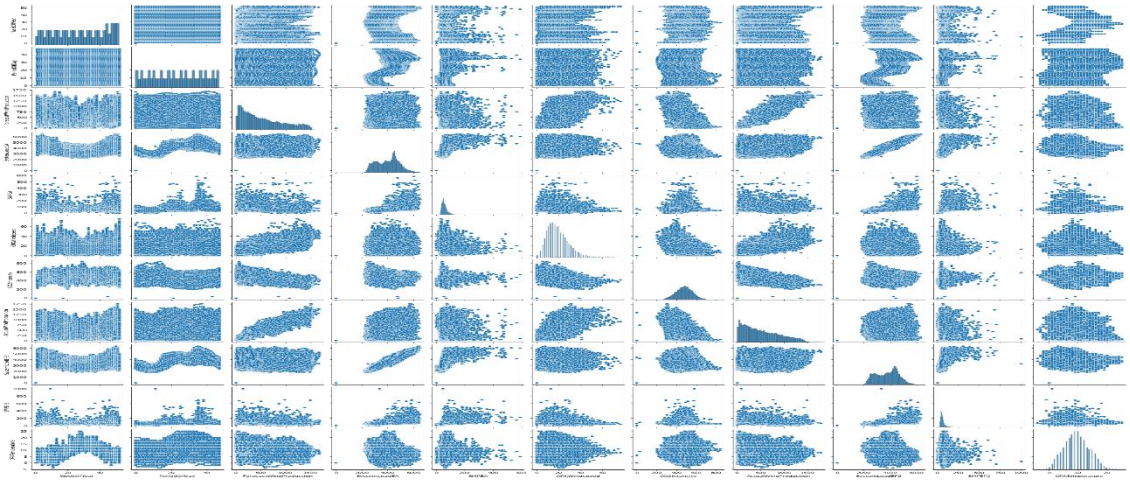




numerical analysis

Numerical analysis is a branch of mathematics and computer science that deals with the development and application of computational algorithms to solve mathematical problems. It involves techniques for approximating solutions to mathematical problems that may be too complex to solve analytically.

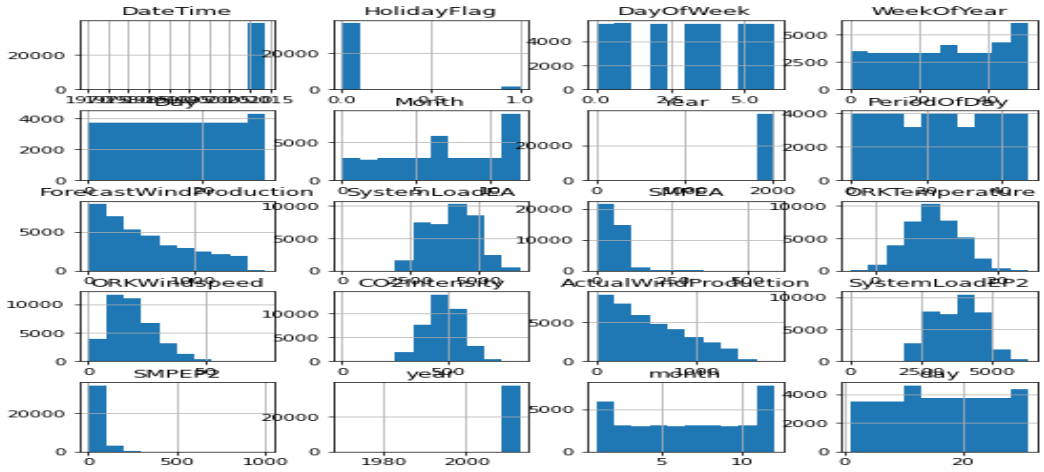
sns.pairplot(df.loc[:,num_list]);



histogram

A histogram is a graphical representation of the distribution of a dataset. It provides a visual summary of the underlying frequency distribution of a set of continuous or discrete data.

df.hist(figsize=(9,9));



Conclusion:

A predictive model utilizing historical data and influencing factors assists in forecasting electricity prices. Improved financial planning, reduced reliance on fossil fuels, and a more sustainable energy future.