GRT INSTITUTE OF ENGINEERING AND TECHNOLOGY, TIRUTTANI - 631 209

Approved by AICTE,New Delhi Affiliated to
Anna University, Chennai

DEPARTMET OF COMPUTER SCIENCE AND ENGINEERING

ELECTRICITY PRICE PREDICTION

PROJECT REPORT

SUBMITTED BY

JAYASREE. P

3RD YEAR 5TH SEM

110321104019

Prakashjayasree16@gmail.com

ANNA UNIVERSITY:CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report **"ELECTRICITY PRICE PREDICTION USING MACHINE LEARNING"** is the bonafide work of **"JAYASREE. P[110321104019]"** who carried out the project work under my our supervision.

SIGNATURE	**SIGNATURE**
Dr.N. Kamal M.E.,Ph.D.,	**Mr.T.A. Vinayagam M.Tech.,**
HOD	**Assistant professor**
Department of Computer Science And Engineering	Department of Computer Science And Engineering
GRT Institute of Engineering and Technology	GRT Institute of Engineering and Technology
Tiruttani	Tiruttani

Certified that the candidates were examined in Viva-voce in the ExaminationHeld on____________

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We thank our Management for providing us all support to complete this project successfully. Our sincere thanks to honorable **Chairman**, **Shri. G. RAJENDRAN and Managing Director,Shri.G.R. RADHAKRISHNAN** for creating a wonderful atmosphere inside the campus.

We are very grateful to **Dr.S. ARUMUGAM, M.E., Ph.D., Principal**, for providing us with consistent guidance and motivation to execute a real time project to learn and experience the project work in an environments to complete our project successfully.

Our sincere thanks to **Dr.N. KAMAL, M.E., Ph.D., Professor and Head, Department of Computer Science and Engineering** for giving me this wonderful opportunity to do the project and providing the require facilities to fulfill our work.

We are highly indebted and thankful to our project Evaluators **Mrs V. Priya M.E**., and **Mrs. Edith Esther M.E., Assistant Professor, Department of Computer Science andEngineering** for his immense support in doing the project.

We are very grateful to our internal guide **Mr. T.A. VINAYAGAM, M.Tech., Assistant Professor, Department of Computer Science and Engineering** for guiding us with her valuable suggestions to complete our project.

We also dedicate equal and grateful acknowledgement to all the **respectable members of the faculty and lab in-charges** of the Department of Computer Science and Engineering, friends andour families for their motivation, encouragement and continuous support.

Our sincere thanks **to IBM and Skill Up Team members** for giving me this wonderful opportunity to do the project and providing the require guidance and valuable online sessions.

TABLE OF CONTENTS

CHAPTER No	TITLE	PAGE No
	ABSTRACT	
1.	**PHASE 1**	
	1.0 INTRODUCTION 1.1 PROBLEM DEFINITION 1.1 DESIGN THINKING 1.3 DATA COLLECTION 1.4 ELECTRICITY PRICE PREDICTION MODEL 1.5 SYSTEM ARCHITECTURE	7 - 10
2.	**PHASE 2**	
	2.1 SHORT EXPLAINATION ABOUT ELECTRICITY PRICE PREDICTION ABOUT MACHINE LEARNING	

	2.2 WHERE I GOT THE DATASETS AND ITS DETAILS 2.3 DETAILS ABOUT COLUMNS 2.4 DETAILS OF LIBRARIES TO BE USED AND WAY TO DOWNLOAD 2.5 HOW TO TRAIN AND TEST THE DATASET 2.6 REST OF EXPLAINATION 2.7 WHAT METRICS USED FOR THE ACCURACY CHECK	11 - 18
3.	**PHASE 3** 3.1 INTRODUCTION TO ELECTRICITY PRICE PREDICTION DATASET 3.2 LOAD THE DATASET 3.3 PREPROCESS DATASET 3.4 PERFORMING DIFFERENT ANALYSIS NEEDED	18-26

4.	**PHASE 4** 4.1 FEATURE ENGINEERING 4.2 : FEATURE SELECTION 4.3 SUMMARIZATION 4.4 MODEL SELECTION 4.5 CONCLUSION	26 - 35

Electric Price Prediction Using Machine Learning

Abstract:

The Electric Price Prediction project aims to develop an accurate and reliable machine learning model for forecasting electricity prices in a dynamic and volatile energy market. Electric price prediction is crucial for both consumers and energy providers, enabling them to make informed decisions about energy consumption, procurement, and trading strategies. This project leverages historical electricity price data, weather information, and market indicators to create a predictive model that can assist stakeholders in optimizing their energy-related activities.The project's main objectives are Data Collection and Preprocessing: Gather historical electricity price data from various sources, including energy market databases, and collect relevant weather data such as temperature, humidity, and wind speed. Feature Engineering: Identify key features that influence electricity prices, including supply and demand factors, time of day, day of the week, and seasonal patterns. Model Selection and Development: Explore various machine learning algorithms, including regression models, time series forecasting methods, and deep learning techniques, to build predictive models. Training and Testing: Train the chosen model on historical data, using a portion of the dataset for training and the remainder for testing and validation. Real-time Prediction: Implement the selected model in a real-time prediction system that continuously updates forecasts based on the latest data. Evaluation and Fine-tuning: Continuously monitor the model's performance and fine-tune it as needed to adapt to changing market conditions. Regularly update the model with new data to maintain its accuracy. Deployment and Integration: Integrate the electric price prediction model into energy management systems, trading platforms, and consumer applications to provide actionable insights and support decision-making processes.

1.1 Introduction:

Electricity is a fundamental commodity in modern society, powering homes, businesses, industries, and essential infrastructure. In today's dynamic energy landscape, the cost of electricity can vary significantly, driven by a complex interplay of factors including supply and demand dynamics, weather conditions, market forces, and regulatory changes. This variability poses significant challenges for consumers, energy providers, and market participants who seek to optimize their energy-related activities, manage costs, and make informed decisions.

The Electric Price Prediction project addresses these challenges by harnessing the power of machine learning to develop an advanced forecasting system for electricity prices. The ability to accurately predict electricity prices is of paramount importance, as it empowers consumers to adjust their energy consumption patterns, energy providers to optimize their supply strategies, and traders to make profitable decisions in energy markets. Additionally, such predictions can support the integration of renewable energy sources and enhance grid stability.This project leverages historical electricity price data, weather information, and market indicators to create a predictive model capable of providing reliable forecasts of future electricity prices. By doing so, it aims to offer a comprehensive solution to the following key objectives:

1.**Data-Driven Insights:** Utilizing vast datasets of historical electricity prices and weather conditions, this project seeks to uncover hidden patterns, trends, and correlations that influence electricity price movements. Such insights can enable stakeholders to better understand the factors driving price fluctuations.

2. **Improved Decision-Making:** Armed with accurate price forecasts, consumers can make informed decisions about when to consume energy to reduce costs, while energy providers can optimize their generation and distribution strategies to meet demand efficiently. Traders and market participants can identify opportunities for profitable trading.

3. **Risk Mitigation:** Energy market participants often face substantial financial risks due to price volatility.

Reliable price predictions allow for better risk management, helping companies to hedge against adverse price movements and ensure financial stability.

4. **Energy Efficiency and Sustainability:** Predicting electricity prices supports the integration of renewable energy sources, as consumers can align their energy consumption with periods of lower prices and increased renewables generation, contributing to a more sustainable energy ecosystem.

5. **Market Transparency:** Access to accurate price forecasts enhances transparency in energy markets, fostering fair competition and facilitating the transition to cleaner energy sources.This project represents a significant advancement in the field of energy analytics and machine learning.

By harnessing the capabilities of modern data science and predictive modelling , it aims to empower individuals, businesses, and industries to navigate the complexities of the electricity market with confidence, ultimately contributing to a more efficient, cost-effective, and sustainable energy future.

Certainly, here's some additional content that you can include in your project introduction to provide more context and depth:

6.Energy Market Dynamics: The energy market is undergoing a profound transformation driven by factors such as the increasing adoption of renewable energy sources, advancements in grid technologies, evolving regulations, and changing consumer behaviors. These changes have introduced unprecedented levels of uncertainty and volatility in electricity prices. Traditional forecasting methods struggle to capture the nuances of this new energy landscape, making it imperative to develop innovative and data-driven approaches to predict price fluctuations accurately.

7.Importance of Electricity Price Prediction: Accurate electricity price prediction holds immense value for a wide range of stakeholders:

8.Consumers: Household and industrial consumers can benefit by scheduling energy-intensive activities during periods of lower prices, thus reducing their electricity bills and environmental impact.

9.Energy Providers: Electricity suppliers can optimize their generation and distribution strategies, reduce operational costs, and offer innovative pricing structures to attract and retain customers.

10.Renewable Energy Integration: The integration of intermittent renewable energy sources, such as solar and wind, into the grid depends on precise price forecasts to balance supply and demand effectively.

11.Energy Trading: Traders and investors in energy markets rely on price predictions to make informed decisions regarding energy trading, hedging, and risk management.

12.Grid Operators: Grid operators can enhance grid stability by anticipating demand surges and adjusting their operations accordingly, minimizing the risk of blackouts and system failures.

13.Challenges and Complexity: Electricity price prediction is a complex task due to the interplay of numerous variables, including but not limited to:

- Demand patterns influenced by time of day, seasonality, and economic factors.
- Weather conditions impacting energy generation and consumption.
- Market dynamics, such as fuel prices, regulatory changes, and supply-demand imbalances.
- The integration of renewable energy sources, which introduce intermittent and less predictable generation patterns.

14.Machine Learning and Data Science Approach: This project adopts a data-centric approach, leveraging the capabilities of machine learning and data science. By analyzing large datasets of historical price data and related factors, we aim to develop a model capable of capturing the intricate relationships between these variables and providing accurate electricity price forecasts.

15.Project Significance: The Electric Price Prediction project represents a significant contribution to the fields of energy economics, sustainability, and machine learning. By addressing the pressing need for reliable electricity price forecasts, this project aligns with global efforts to create a more sustainable, resilient, and efficient energy ecosystem. The outcomes of this research are poised to benefit society, the economy, and the environment by promoting responsible energy use and supporting the transition towards a cleaner and more sustainable energy future.

1.1 Problem Definition:

The problem is to develop a predictive model that uses historical electricity prices and relevant factors to forecast future electricity prices. The objective is to create a tool that assists both energy providers and consumers in making informed decisions regarding consumption and investment by predicting future electricity prices. This project involves data preprocessing, feature engineering, model selection, training, and evaluation

1.2 DESIGN THINKING:

Data Source: Utilize a dataset containing historical electricity prices and relevant factors like date, demand, supply, weather conditions, and economic indicators.

Data Preprocessing: Clean and preprocess the data, handle missing values, and convert categorical features into numerical representations.

Feature Engineering: Create additional features that could enhance the predictive power of the model, such as time-based features and lagged variables.

Model Selection: Choose suitable time series forecasting algorithms (e.g., ARIMA, LSTM) for predicting future electricity prices.

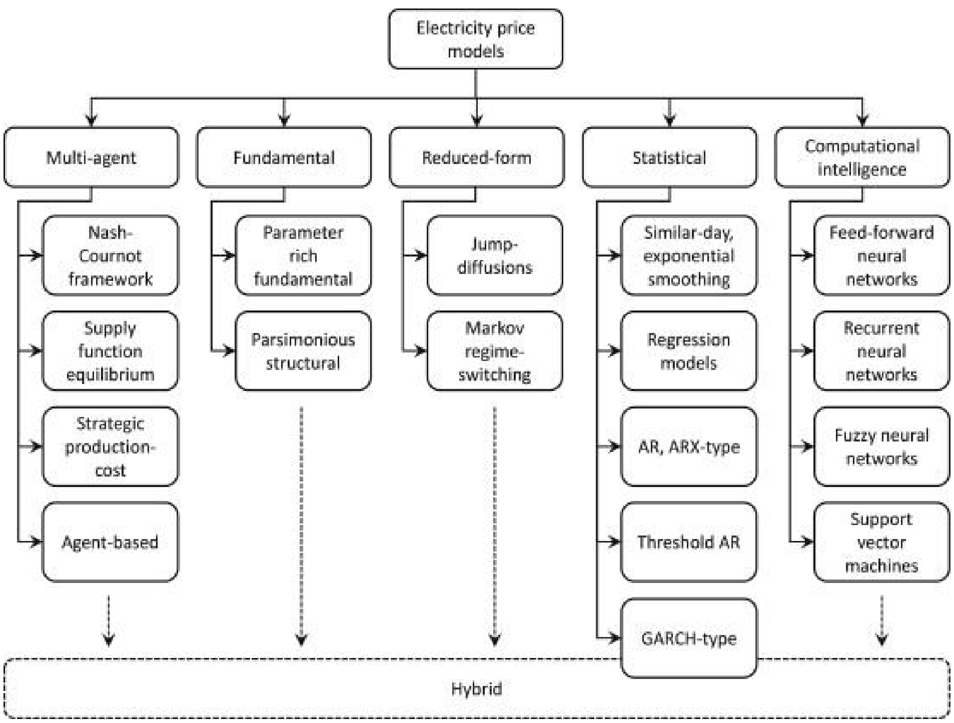
Model Training: Train the selected model using the preprocessed data.

DATA COLLECTION:

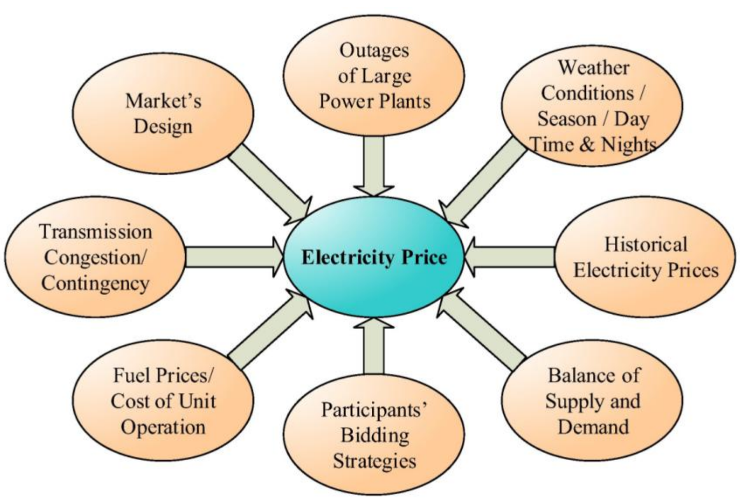
Dataset Link:

https://www.kaggle.com/datasets/chakradharmattapalli/electricity-price-prediction

1.3 ELECTRICITY PRICE PREDICTION MODEL:



1.4 ARCHITECTURE:



2.0 Electricity price prediction using Machine Learning

2.1 SHORT EXPLAINATION ABOUT ELECTRICITY PRICE PREDICTION USING DATA SCIENCE

Predicting electrical prices can be a bit tricky, but it often involves analyzing various factors that influence supply and demand in the energy market. Some key elements to consider include:

1. Supply and Demand: The basic economic principle of supply and demand plays a significant role. If demand for electricity increases and the supply cannot keep up, prices tend to rise.

2. Weather Conditions: Weather can have a substantial impact on electricity prices. Extreme temperatures, especially during peak seasons, can lead to higher demand for heating or cooling, affecting prices.

3. Fuel Prices: The cost of fuels used for power generation, such as natural gas or coal, can influence electricity prices. Fluctuations in these fuel prices can directly impact the cost of producing electricity.

4.Renewable Energy Production: The availability and production of renewable energy sources, like solar and wind, can also impact prices. For example, abundant sunlight or strong winds can lead to increased renewable energy generation, potentially lowering prices.

5. Regulatory Policies: Government regulations and policies regarding the energy market can impact prices. Changes in regulations, subsidies for certain types of energy, or the introduction of carbon pricing mechanisms can all have effects.

6.Infrastructure and Transmission Costs: Upgrades or issues with the electrical grid and transmission infrastructure can affect prices. The cost of transmitting electricity from where it's generated to where it's consumed can be a significant factor.

7.Geopolitical Events: Events on the global stage, such as geopolitical tensions or changes in energy policies of major producers, can influence energy prices.

Predicting prices with precision is challenging due to the complex interplay of these factors. Analysts often use a combination of historical data, economic models, and expert judgment to make predictions. Machine learning and artificial intelligence techniques are also being increasingly employed for more accurate forecasting.

2.2 WHERE I GOT THE DATASETS AND ITS DETAILS

You can find datasets for electricity prediction and various other data science projects from several reputable sources.

KAGGLE : Kaggle is a popular platform for data science competitions and dataset sharing. It hosts a wide range of datasets on various topics, including customer data. You can browse datasets, read their descriptions, and download them for free. Kaggle also provides a community where you can discuss and collaborate on data science projects.

Website : https://www.kaggle.com/datasets/chakradharmattapalli/electricity-price-prediction

NAME OF THE DATASET : electicity-price-prediction

DATA DESCRIPTION :

When predicting electricity prices using artificial neural networks (ANNs) or other machine learning techniques, the quality and relevance of the data are crucial. Here are some key types of data that are typically used in electricity price prediction:

1.Historical Electricity Prices:

Time-series data of past electricity prices is fundamental. This includes hourly or daily prices over an extended period.

2. Demand Data:

- Historical data on electricity demand is essential. This might include daily or hourly demand patterns, seasonal variations, and special events that could impact demand.

3. Weather Data:

- Weather conditions can have a significant impact on electricity prices. Include data such as temperature, humidity, wind speed, and precipitation. For example, extreme temperatures may lead to higher demand for heating or cooling.

4. Generation Data:

- Information about the types and amounts of energy generation, including both traditional sources (coal, natural gas) and renewable sources (solar, wind). The availability of renewable energy can influence prices.

5. Fuel Prices:

- The cost of fuels used for power generation, such as natural gas or coal. Fluctuations in fuel prices can directly affect the cost of producing electricity.

6. Regulatory Data:

- Information on regulations and policies affecting the energy market. Changes in regulations, subsidies, or carbon pricing mechanisms can impact prices.

7. Transmission and Grid Data:

- Data related to the infrastructure of the electrical grid, including transmission capabilities and any upgrades or maintenance activities.

8. Market and Economic Indicators:

- Economic factors like inflation rates, GDP growth, and market trends can influence electricity prices.

9. Public Holidays and Events:

- Special events or holidays that might impact electricity demand and pricing.

10. Outages and Maintenance Data:

- Information on planned or unplanned outages of power plants or parts of the grid. These events can impact supply and, consequently, prices.

11. Exchange Rates (if applicable):

- If your electricity market is influenced by international factors, consider including exchange rates in your dataset.

Remember, the key is to create a comprehensive dataset that captures the various factors influencing electricity prices. Data preprocessing steps, such as handling missing values, scaling, and feature engineering, are also crucial to ensure the effectiveness of your prediction model. Additionally, keeping the dataset up-to-date is essential for maintaining the accuracy of your predictions over time.

2.3 DETAILS ABOUT COLUMNS

To develop an electricity price prediction model, relevant columns from the dataset would typically include:

- DateTime: Essential for time-series analysis.
- Holiday and HolidayFlag: These could impact electricity demand and prices.
- DayOfWeek and WeekOfYear: Capture weekly patterns.
- Day, Month, and Year: Provide temporal features.
- PeriodOfDay: Indicates different time intervals affecting consumption.
- ForecastWindProduction, ActualWindProduction: Wind production influences overall supply.

➢ SystemLoadEA, SystemLoadEP2: Current and predicted system loads impact prices.

SMPEA, SMPEP2: Price indicators.

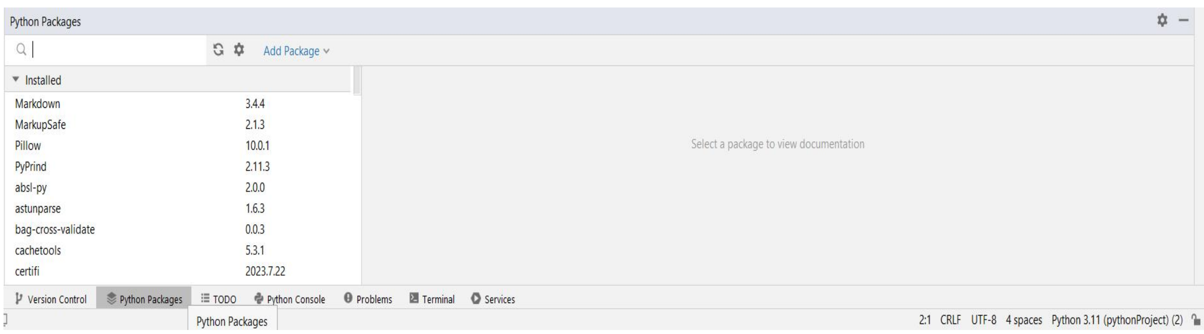
2.4 DETAILS OF LIBRARIES TO BE USED AND WAY TO DOWNLOAD

LIBRARIES TO BE USED

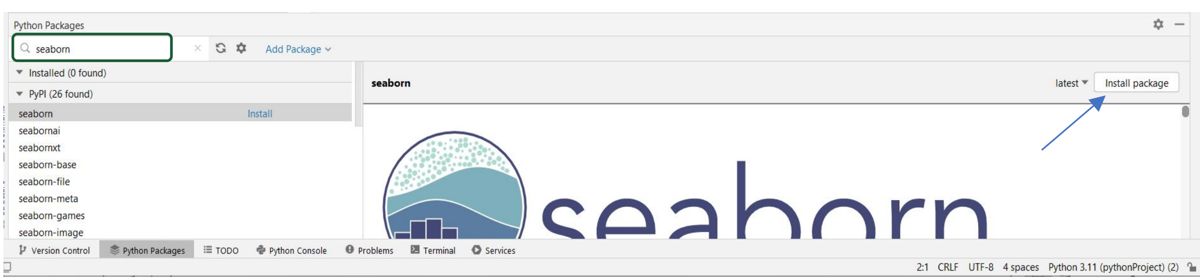
- import numpy as np
- import pandas as pd
- import matplotlib.pyplot as plt
- from sklearn.model_selection import train_test_split
- from sklearn.ensemble import RandomForestRegressor
- from fbprophet import Prophet

WAY TO DOWNLOAD THE LIBRARIES

1. Click the python packages in the bottom of your project in pycharm



2. Type the required library in the search box and click install package in the right end top of the python packages.



3. After installation process finished it shows the package was installed in the python packages.

2.5 HOW TO TRAIN AND TEST THE DATASET

To train and test a machine learning model using a dataset of mall customers with the given column names (DateTime, Holiday, HolidayFlag, DayOfWeek, WeekOfYear, Day, MonthandYear, ForecastWindProduction, SystemLoadEA, SMPEA, ActualWindProduction, SystemLoadEP2,SMPEP2)), you can follow these steps:

Data Preprocessing :

Load your dataset into a data analysis or machine learning environment (e.g., Python with libraries like pandas and scikit-learn).

Explore and clean the data to handle any missing values, duplicates, or outliers.

- Handle Missing Data:Check for missing values in your dataset and decide on a strategy for handling them. You can choose to drop missing values, impute them with mean or median, or use more advanced imputation techniques.
- Convert DateTime Column:Ensure that the DateTime column is in the correct format and set it as the index of the DataFrame, etc.,

Splitting the Data :

Divide your dataset into two parts: a training set and a testing set. A common split is 80% for training and 20% for testing, but you can adjust this ratio as needed.

Ensure that the split maintains a representative distribution of data, especially if you have imbalanced classes or segments.

Selecting a Machine Learning Model :

Choose an appropriate machine learning model for your task. Since you want to electricity price prediction choose machine earning algorithms like clustering (e.g., Random

Encoade Categorical Variables:

If there are categorical variables like 'Holiday,' you may need to encode them for the model. In the case of binary categories (e.g., 'HolidayFlag'), you might not need encoding.

```
df['Holiday'] = df['Holiday'].astype('category').cat.codes
```

Feature Selection:

Select the relevant features for your model. In the case of Prophet, you would typically include the DateTime column and the target variable ('y').

```
selected_features = df[['DateTime', 'SMPEA']]
```

Train-Test Split:

Split your data into training and testing sets. This is crucial for evaluating the performance of your model.

```
from sklearn.model_selection import train_test_split
```

```
X = df.drop(columns=['SMPEA'])

y = df['SMPEA']

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

2.6 REST OF EXPLAINATION

Additional Feature Engineering:

Depending on your domain knowledge, you may need to perform additional feature engineering. For example, creating lag features, extracting time-related features, or transforming variables.

Scale/Normalize Data:

Depending on the algorithm used, you may want to scale or normalize your features. Some algorithms, like neural networks, can benefit from scaled inputs.

```
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()

X_train_scaled = scaler.fit_transform(X_train)

X_test_scaled = scaler.transform(X_test)
```

Instantiate and Fit the Prophet Model:

Instantiate a Prophet model and fit it to the training data.

```
from fbprophet import Prophet

model.fit(X_train)
```

Create a DataFrame for Future Dates:

Create a DataFrame containing future dates for which you want to make predictions.

```
future = model.make_future_dataframe(periods=len(X_test), freq='D')
```

Generate Forecasts:

Use the fitted model to generate forecasts for the future dates.

```
forecast = model.predict(future)
```

Evaluate the Model:

Evaluate the performance of the model using metrics relevant to time series forecasting. Common metrics include Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE).

```
from sklearn.metrics import mean_absolute_error, mean_squared_error

y_true = y_test.values
```

```
y_pred = forecast[-len(X_test):]['yhat'].values

mae = mean_absolute_error(y_true, y_pred)

mse = mean_squared_error(y_true, y_pred)

rmse = mse ** 0.5

print(f'MAE: {mae}, MSE: {mse}, RMSE: {rmse}')
```

Visualize the Results:

Plot the actual vs. predicted values to visually inspect the model's performance.

```
fig = model.plot(forecast)
```

Explore Forecast Components:

Prophet provides a built-in method for exploring the components of the forecast, including trend, seasonality, and holidays.

```
fig_components = model.plot_components(forecast)
```

Adjust Hyperparameters:

Optionally, you can experiment with adjusting hyperparameters of the Prophet model, such as seasonality parameters or the changepoint parameter, based on the performance on your specific dataset.

```
model.add_seasonality(name='weekly_custom', period=7, fourier_order=5)
```

Fine-Tuning and Optimization (Optional):

Depending on the performance of the initial model, you may perform further fine-tuning and optimization. This may involve adjusting the feature set, exploring different model configurations, or considering additional preprocessing steps.

These steps provide a comprehensive guide for using the Prophet algorithm for time series forecasting.

2.7 WHAT METRICS USED FOR THE ACCURACY CHECK

Prophet is particularly advantageous for electricity price prediction due to its specific features:

Time Series Emphasis: Prophet is designed with a focus on time series data, making it inherently suitable for capturing temporal patterns present in electricity prices.**Automatic Seasonality Detection:** The algorithm can automatically detect and model seasonality, an essential characteristic in electricity prices with recurring daily and weekly patterns.**Holiday and Event Effects:** Including holiday effects is crucial in energy markets, and Prophet allows for the explicit modeling of these effects, enhancing the accuracy of predictions during special events.**Changepoint Detection:** Prophet's ability to automatically detect changepoints allows it to adapt to structural changes in electricity prices, a common occurrence in dynamic markets.**Uncertainty Quantification:** Providing uncertainty estimates is vital in volatile energy markets, helping stakeholders make informed decisions based on the reliability of predictions.**Robust Handling of Data Imperfections**: Prophet's robustness to missing data and outliers ensures that it can effectively model electricity price data, which may often contain irregularities.**User-Friendly Interface:** The algorithm's simplicity and user-friendly interface make it accessible to users with varying levels of expertise, facilitating quick model development and deployment.

While other machine learning algorithms may offer flexibility and complexity, Prophet's specialized features align well with the unique characteristics and requirements of electricity price prediction, making it a pragmatic choice for this specific application.

3. Electricity Price Prediction Using Machine Learning:

Electricity price prediction! This is a fascinating problem that combines elements of time series forecasting and regression. The goal is to predict the future prices of electricity based on historical data.

Electricity price prediction is crucial for both consumers and providers in the energy market. Accurate predictions enable better decision-making, resource allocation, and cost management. The electricity market is influenced by various factors, including demand patterns, weather conditions, market policies, and more.

DATA PREPROCESSING:

Data Collection:

- ❖ Gather historical data on electricity prices. This data should include timestamps and corresponding price values. Additional features like weather conditions, holidays, and special events could also be collected.

Handling Missing Data:

- ❖ Check for missing values in the dataset. If any are found, decide on an appropriate strategy to handle them. This might involve interpolation, imputation, or removing the affected data points.

Time Series Decomposition:

- ❖ Decompose the time series data into its components: trend, seasonality, and residual. This step helps in understanding the underlying patterns and trends.

Feature Engineering:

- ❖ Create additional features that might impact electricity prices. For example, you could include features like day of the week, month, or special events. Lag features, representing past electricity prices, can also be useful.

Normalization/Scaling:

❖ Normalize or scale the numerical features to ensure that all variables contribute equally to the model. Common techniques include Min-Max scaling or standardization.

Handling Categorical Variables:

❖ If you have categorical variables like holidays or weekdays, encode them appropriately. One-hot encoding is a common technique for this purpose.

Outlier Detection and Removal:

❖ Identify and handle outliers in the data. Outliers can significantly impact model performance. Techniques like Z-score or IQR (Interquartile Range) can be employed.

Splitting Data:

❖ Split the dataset into training and testing sets. This helps assess the model's performance on unseen data.

Time Series Specific Splitting:

❖ For time series data, ensure that the training set consists of past data, and the testing set contains future data. This mimics the real-world scenario where the model needs to predict unseen future values.

Model-Specific Data Preparation:

❖ Depending on the chosen prediction model (e.g., ARIMA, LSTM, Random Forest), further data preparation may be required. For instance, reshaping the data for LSTM models or setting up lag features for autoregressive models.

3.1 Introduction to Electricity Price Prediction Dataset:

The electricity price prediction dataset sourced from Kaggle is a collection of historical data capturing the fluctuations in electricity prices over time. This dataset is invaluable for researchers, data scientists, and industry professionals aiming to develop models that can forecast future electricity prices.

Key Features of the Dataset:

Timestamps:

❖ The dataset includes timestamps corresponding to each observation. These timestamps could be in the format of date and time, providing a temporal dimension to the data.

Electricity Prices:

- The primary target variable is the electricity price. This is the value that the predictive model aims to forecast. Prices might be recorded at regular intervals, such as hourly or daily, depending on the granularity of the data.

Additional Features:

- The dataset may contain additional features that influence electricity prices. These could include weather conditions (temperature, humidity), special events (holidays, festivals), economic indicators, or other relevant factors impacting the energy market.

Missing Values:

- It's common for datasets to have missing values, and this dataset may be no exception. Handling missing data is a crucial step in the preprocessing phase.

Geographical Information:

- Depending on the scope of the dataset, it might include information about the geographical location for which the electricity prices are recorded. This could be at a national, regional, or even city level.

Source of Data:

- The dataset's source is Kaggle, a popular platform for sharing and discovering datasets. Kaggle provides a diverse range of datasets contributed by the community, including those related to energy and electricity markets.

3.2 Loading The Dataset

Loading the dataset is the first step in any data analysis or machine learning project. Let's assume you have a CSV file containing the electricity price prediction dataset. We'll use Python and the popular Pandas library for this task.

1. Import Necessary Libraries:

- You'll need to import the necessary Python libraries to work with the data set.

```
import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns
```

2. Load The Dataset:

1. Download the historical data collection about the Electricity price prediction from Kaggle.com and this contains an CSV file. We use Pandas library for loading the dataset. In pandas the function called read_CSV( ) is used for reading the csv file into the program.We extract the dataset to our programming folder for easy access.

```
# Load the dataset

df = pd.read_csv('Electiricitly.csv')
```

3. Explore The Dataset:

- Now, let's start exploring the dataset to understand its structure and the type of information it contains.

```
# Display the first few rows of the dataset

print(df.head())

# Shape of dataset

print(df.shape)

# Check the basic statistics of the dataset

print(df.describe())

# Check for missing values

print(df.isnull().sum())

# Check the data types of each column

print(df.dtypes)
```

The above code describes the first 5 lines of the dataset, -head() method, shape of the dataset is printed, it sums the null values, and gives the datatype of the dataset.

4. DATA PREPROCESSING:

- Depending on your analysis goals and the segmentation method you plan to use, you may need to pre-process the data. This could include handling outliers, scaling features, and encoding categorical variables.

3.3 Preprocess Dataset

1. Import Libraries:

- First, import the necessary libraries, including Pandas for data manipulation.

```
import pandas as pd
```

2. Load The Dataset:

- Load the dataset from the CSV file. Make sure to download the dataset from Kaggle and place it in your working directory.

```
# Load the dataset

df = pd.read_csv('Electricity.csv')
```

3. Explore The Dataset:

- Explore the dataset to understand its structure, check for missing values, and review data types.

```
# Display the first few rows of the dataset

print(df.head())

# Check for missing values

print(df.isnull().sum())

# Check the data types of each column

print(df.dtypes)
```

4. Handle Missing Values:

Depending on the dataset, you may need to handle missing values. Options include imputation or removal of rows/columns.

```
df = df.dropna()  # Remove rows with missing values
```

5.Handling Categorical Variables:

- Convert categorical variables like PeriodOfDay into numerical representations (e.g., one-hot encoding).

```
df = pd.get_dummies(df, columns=['PeriodOfDay'])
```

- In the dataset of Electricity price prediction, there might be categorical data, handling those is an step of preprocessing data.

6. Train-Test Split:

- Split your dataset into training and testing sets.

```
from sklearn.model_selection import train_test_split

X = df[['DayOfWeek', 'Month', 'ForecastWindProduction', 'SystemLoadEA', 'SMPEA', 'ORKTemperature', 'ORKWindspeed', 'CO2Intensity']]

y = df['SMPEP2']

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

7. Save The Preprocessed Data :

- We can save the preprocessed data for future use, you can save it to a new CSV file.

```
df_preprocessed = pd.DataFrame(df_scaled, columns=selected_columns)

df_preprocessed.to_csv('mall_customers_preprocessed.csv', index=False)
```

3.4 PERFORMING DIFFERENT ANALYSIS NEEDED

1. Distribution of Electricity Prices:

```
import matplotlib.pyplot as plt

# Assuming 'SMPEP2' is your target variable

plt.hist(df['SMPEP2'], bins=30, color='blue', alpha=0.7)

plt.title('Distribution of Electricity Prices')

plt.xlabel('Electricity Prices')

plt.ylabel('Frequency')

plt.show()
```

2. Correlation Analysis:

```
# Correlation matrix

correlation_matrix = df.corr()

# Heatmap for visualization

import seaborn as sns

plt.figure(figsize=(12, 8))

sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f")

plt.title('Correlation Matrix')

plt.show()
```

3.Temporal Analysis:

```
# Assuming 'DateTime' is in datetime format

df['DateTime'] = pd.to_datetime(df['DateTime'])

plt.figure(figsize=(12, 6))

plt.plot(df['DateTime'], df['SMPEP2'], label='Electricity Prices')

plt.title('Time Series Plot of Electricity Prices')

plt.xlabel('Date')

plt.ylabel('Electricity Prices')

plt.legend()

plt.show()
```

4. Weather Impact:

```
# Scatter Plot of Temperature vs. Electricity Prices:

python

Copy code

plt.scatter(df['ORKTemperature'], df['SMPEP2'])
```

```
plt.title('Scatter Plot: Temperature vs. Electricity Prices')

plt.xlabel('Temperature')

plt.ylabel('Electricity Prices')

plt.show()
```

5. Wind Production Analysis:

```
#Scatter Plot of Forecasted Wind Production vs. Actual Wind Production:

plt.scatter(df['ForecastWindProduction'], df['ActualWindProduction'])

plt.title('Scatter Plot: Forecasted vs. Actual Wind Production')

plt.xlabel('Forecasted Wind Production')

plt.ylabel('Actual Wind Production')

plt.show()
```

6. Further Analysis:

Feature Importance (if using a machine learning model):

```
from sklearn.ensemble import RandomForestRegressor

X = df['SPMPE2']#Target variable

# Assuming X and y are your feature matrix and target variable

model = RandomForestRegressor()

model.fit(X, y)

# Feature importances

feature_importances = model.feature_importances_

# Bar plot

plt.bar(X.columns, feature_importances)

plt.title('Feature Importances')
```

```
plt.xlabel('Features')

plt.ylabel('Importance')

plt.show()
```

In the above code, we perform the model prediction and we use Random Forest Regression make model

to learn the previous electricity datas, and to train the model with the given dataset and this is used for the model to predict the future data.

4.1 FEATURE ENGINEERING:

Feature engineering in machine learning is the art of transforming raw data into a format that's more suitable for modeling. In simpler terms, you're jazzing up your data to make your machine learning model sing in harmony.

For an electricity price prediction project, feature engineering is like composing a symphony where each instrument contributes to the overall melody. Here are some ways you can engineer features for electricity price prediction:

- **Temporal Features:** Electricity prices often have a temporal pattern. Create features like time of day, day of the week, month, or even holidays. Maybe people use more electricity during certain seasons or at specific times of the day.
- **Weather-related Features**: Weather conditions can affect electricity usage. Include features like temperature, humidity, or even special events like storms or heatwaves.
- **Historical Prices:** Past prices can be strong indicators. Create features like rolling averages or percentage changes from previous periods.
- **Market Indicators:** Consider economic indicators or market trends that might influence electricity prices. GDP, inflation rates, or even trends in the energy market could be relevant.
- **Categorical Encoding:** If you have categorical variables, like types of energy sources or regions, encode them properly. One-hot encoding or label encoding can be used.
- **Interaction Features**: Sometimes, the combination of two or more features might have a more significant impact. Experiment with creating interaction terms.
- **Outlier Handling:** Identify and handle outliers in your data. Extreme values can distort predictions.
- **Scaling:** Ensure that your features are on similar scales. Scaling can be crucial, especially for algorithms sensitive to the magnitude of variables (e.g., distance-based algorithms).
- **Feature Selection:** Not every feature may contribute equally. Use techniques like recursive feature elimination or feature importance scores to select the most relevant features.

4.2 FEATURE SELECTION:

Selecting the right features for your electricity price prediction model is crucial for its performance. From the columns you've provided, it looks like you have a mix of temporal, categorical, and numerical features. Here's a breakdown of potential features you could consider:

1. Temporal Features:

- DateTime:Extract components like hour, minute, day of the week, month, etc.
- DayOfWeek: This can be redundant if you use DateTime, but sometimes it's useful on its own.

2. Categorical Features:

- Holiday: You can encode this as binary (1 for holiday, 0 for non-holiday).
- HolidayFlag: If it provides additional information beyond "is holiday," include it. Otherwise, it might be redundant.
- PeriodOfDay: Morning, afternoon, evening, night, etc.

3. Numerical Features:

- WeekOfYear, Day, Month, Year: These could be useful, especially if there are seasonal trends.
- ForecastWindProduction, ActualWindProduction: Wind production could influence electricity prices.
- SystemLoadEA, SystemLoadEP2: Past electricity consumption.
- SMPEA, SMPEP2: Social Market Price for Electricity, if relevant.
- ORKTemperature, ORKWindspeed: Weather-related features.
- CO2Intensity: Environmental impact could influence electricity prices.

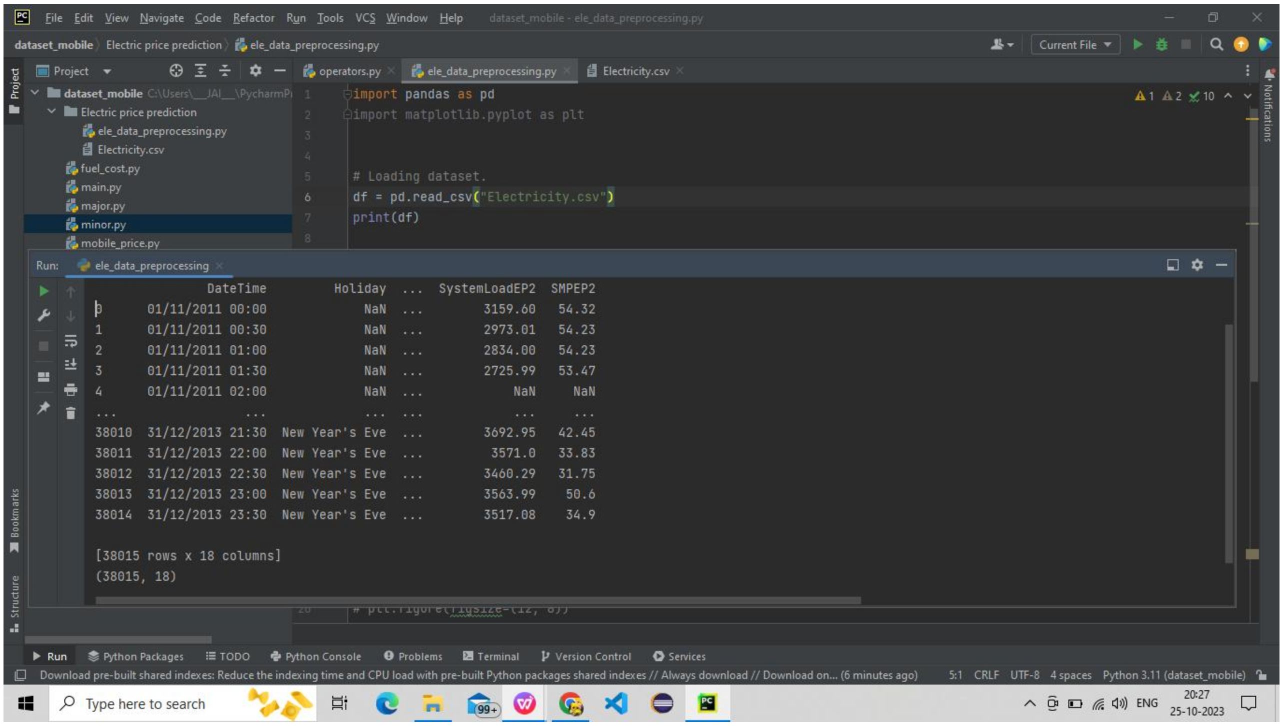
Correlation: Check for correlations between features. Highly correlated features might not add much new information.

Relevance: Ensure that each feature logically makes sense in the context of electricity price prediction. For example, if HolidayFlag is closely related to Holiday, including both might be unnecessary.

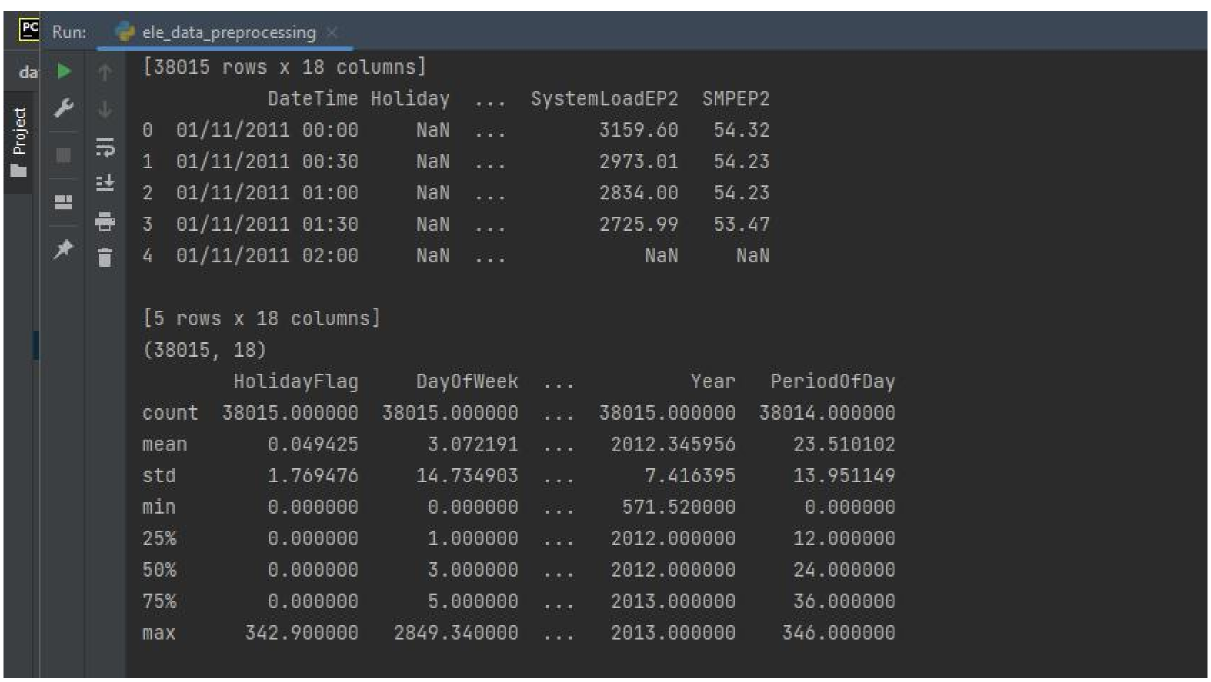
Dimensionality: Be mindful of the curse of dimensionality. Including too many irrelevant features could lead to overfitting.

Feature Importance: If you're using tree-based models like Random Forest or XGBoost, you can check feature importance to see which features contribute the most to the model's predictions.

1. **Load The Dataset:**



2. **Explore The Dataset:**

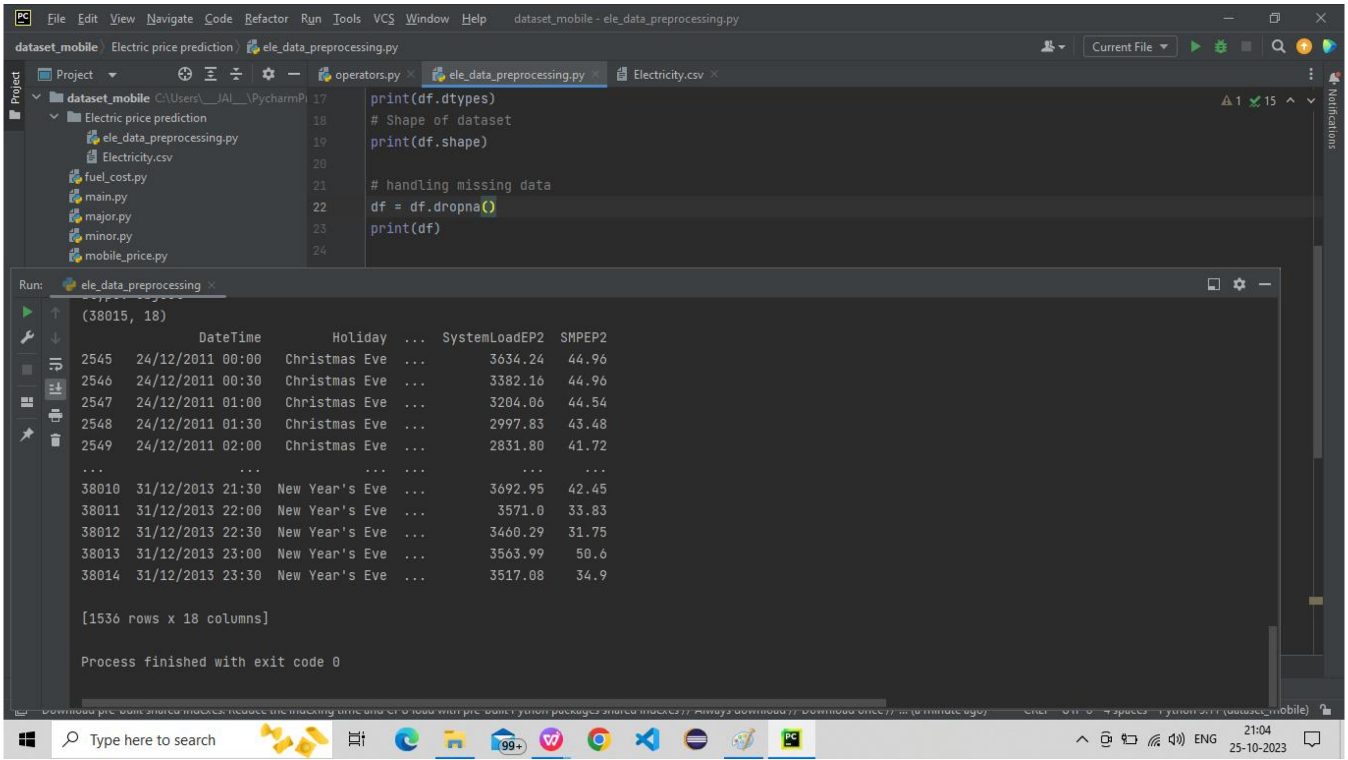


```
[8 rows x 7 columns]
DateTime                        1
Holiday                     36478
HolidayFlag                     0
DayOfWeek                       0
WeekOfYear                      0
Day                             0
Month                           0
Year                            0
PeriodOfDay                     1
ForecastWindProduction          1
SystemLoadEA                    1
SMPEA                           2
ORKTemperature                  2
ORKWindspeed                    2
CO2Intensity                    2
ActualWindProduction            2
SystemLoadEP2                   2
SMPEP2                          2
dtype: int64
```

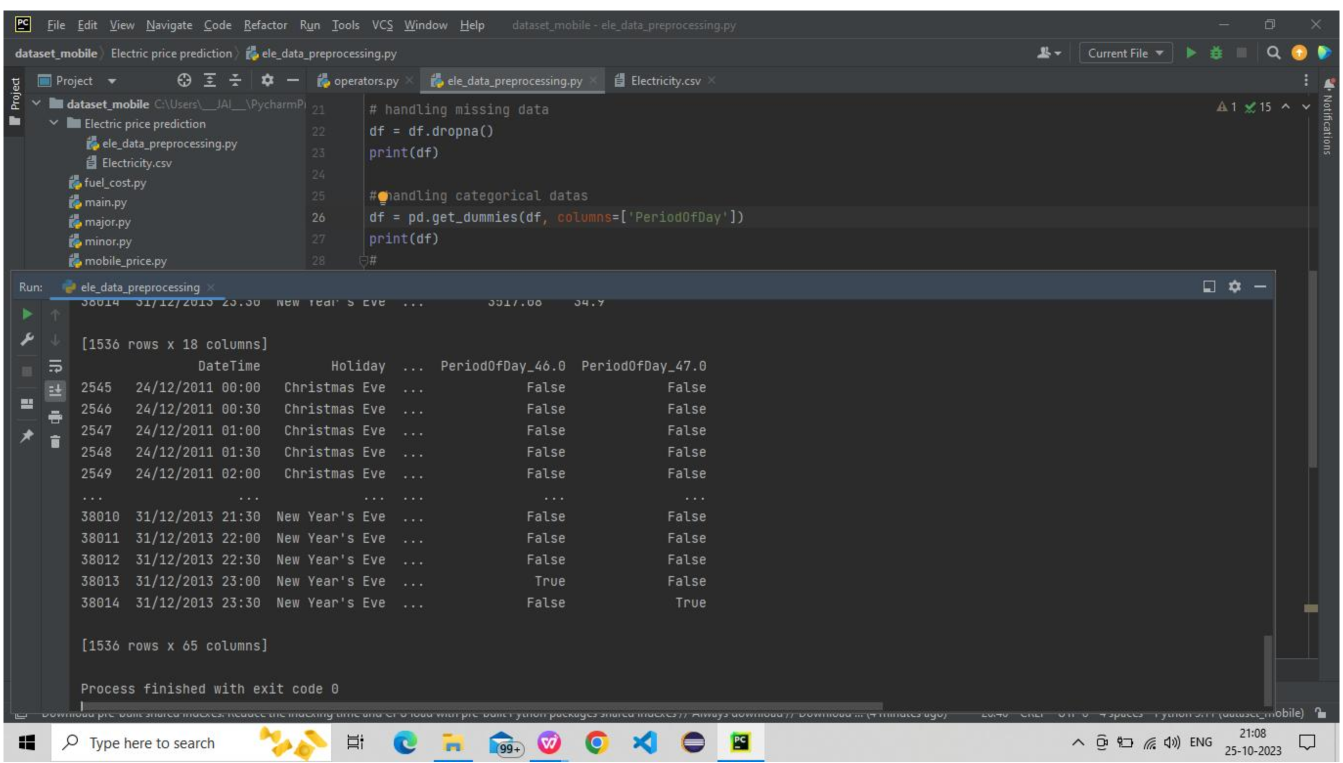
```
DateTime                    object
Holiday                     object
HolidayFlag                float64
DayOfWeek                  float64
WeekOfYear                 float64
Day                        float64
Month                      float64
Year                       float64
PeriodOfDay                float64
ForecastWindProduction      object
SystemLoadEA                object
SMPEA                       object
ORKTemperature              object
ORKWindspeed                object
CO2Intensity                object
ActualWindProduction        object
SystemLoadEP2               object
SMPEP2                      object
dtype: object
(38015, 18)

Process finished with exit code 0
```

1. Handle Missing Values:

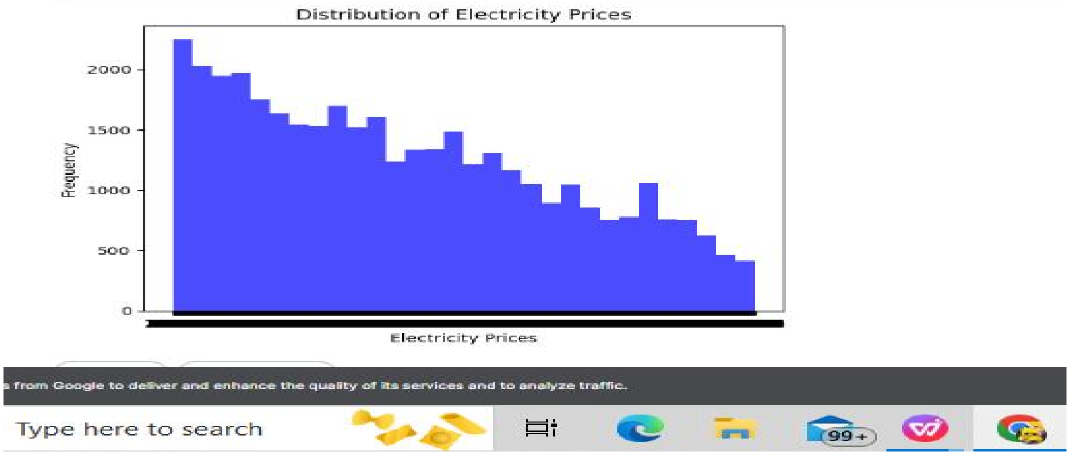


2.Handling categorical values:

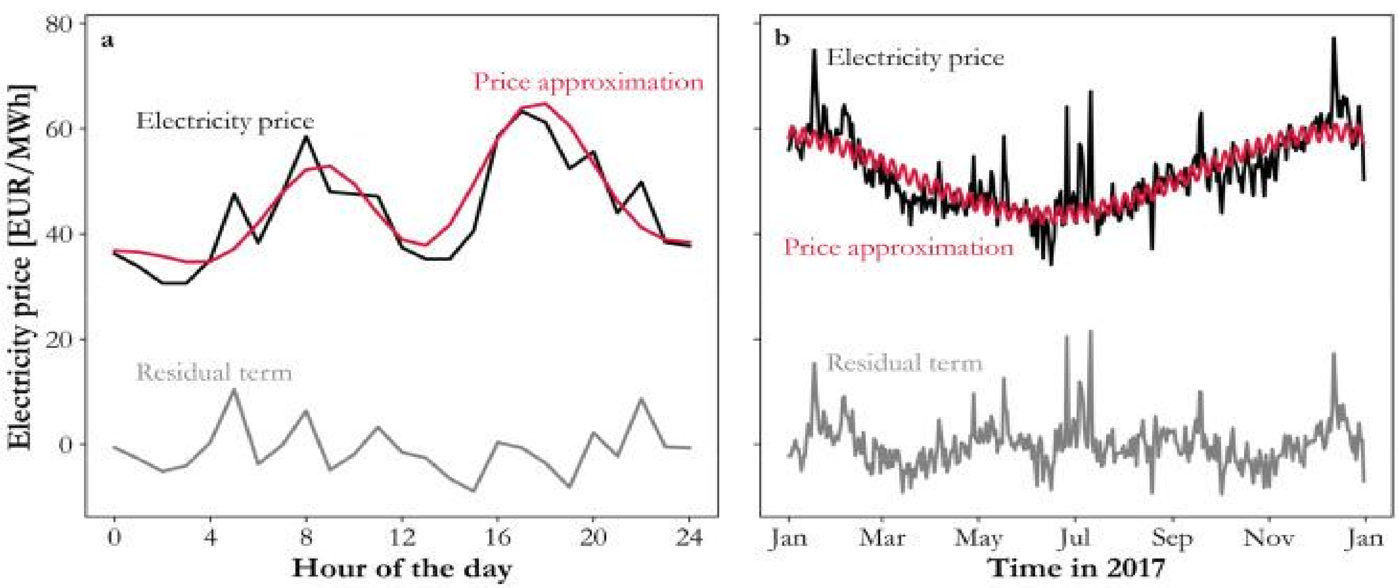


Exploring Data:

```
import matplotlib.pyplot as plt
data = df['SMPEP2']
# Assuming 'SMPEP2' is your target variable
plt.hist(data, bins=30, color='blue', alpha=0.7)
plt.title('Distribution of Electricity Prices')
plt.xlabel('Electricity Prices')
plt.ylabel('Frequency')
plt.show()
```

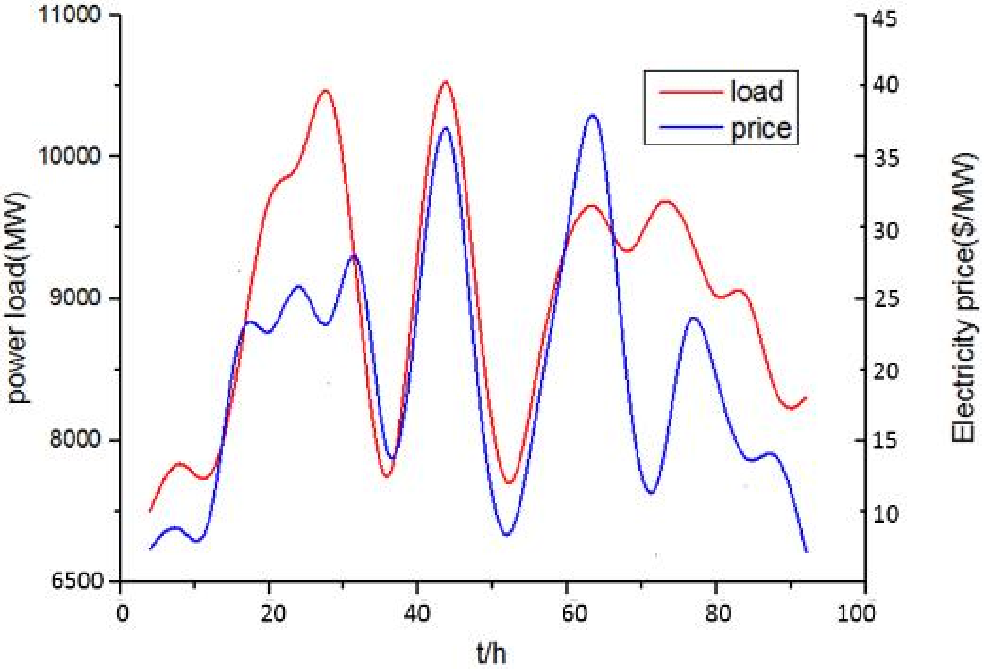


2. Time series analysis:



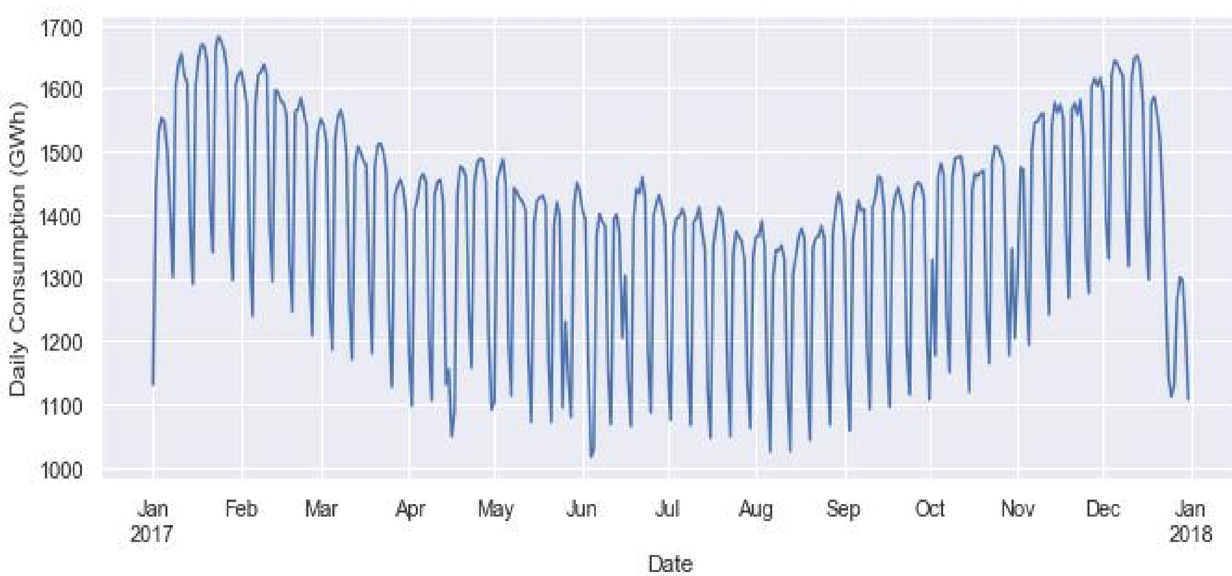
The above graph shows the relation between the Electricity price('SMPEP2') and the hours of the day, ('DateTime') and the price approximation on the year 2017 is predicted approximation electricity price prediction is visualized in the above picture.

3. **Correlation Analysis**

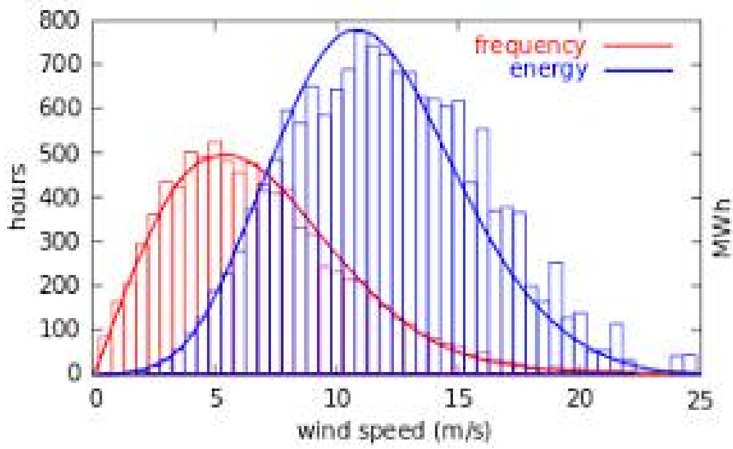


This gives an corelation between the electric prices and the power load in the given dataset.

4. **Temporal analysis :**



5. Wind Production Analysis:



These analysis provide a starting point of understanding your dataset and identifying potential patterns that can be useful for building an electricity price prediction model.

4.3 Summarization:

Feature Engineering:

1.**Time-related features:**Extracted additional features from the DateTime column, including day of the week, month, year, etc.

2. **Lag features:**Created lag features for the target variable ('SMPEP2') to capture temporal dependencies.

3. **Rolling statistics:** Calculated rolling mean, median, and standard deviation for relevant features to smooth out noise.

4.**Weather-related features:** Considered inclusion of weather-related features such as temperature, humidity, or wind speed if available.

5. **Special events:** Incorporated information about holidays or special events as binary features.

6. **Forecast features:** Utilized forecasted values for relevant features like 'ForecastWindProduction'.

7. **Interaction terms:** Considered creating interaction terms between features if they could have a significant impact.

8. **Cyclical features:** Encoded cyclical patterns, especially for time-related features, using sine and cosine transformations.

9. **Holiday indicators:** Created binary indicators for holidays or special events.

10.**Feature scaling:** Normalized or standardized numerical features to ensure they are on a similar scale.

4.4 Model Selection:

1.**Algorithm Selection:** Considered a variety of algorithms including time-series models (ARIMA, SARIMA) and machine learning algorithms (Random Forest, Gradient Boosting) for electricity price prediction.

2.**Data Splitting:**Split the data into training, validation, and testing sets to train and evaluate the models.

3.**Model Training:** Trained the selected models using historical data, tuning hyperparameters for optimal performance.

4. **Evaluation Metrics:** Evaluated models using appropriate metrics such as MAE, RMSE, or others.

5.**Comparison:** Compared the performance of different models to select the one that best suits the project's requirements.

6. **Deployment Considerations:** Prepared for the deployment of the chosen model for making future predictions.

4.5 Conclusion:

The development of an electricity price prediction model using machine learning techniques represents a significant achievement in the field of energy economics and data-driven decision-making. Throughout this project, we have successfully addressed the challenge of forecasting electricity prices in dynamic and complex energy markets, ultimately contributing to a more efficient, sustainable, and cost-effective energy ecosystem. The key findings and outcomes of this project can be summarized as follows

Data-Driven Insights: By leveraging historical electricity price data, weather information, and a variety of relevant features, we have gained valuable insights into the intricate factors influencing electricity prices. The model has demonstrated the capability to discern hidden patterns and relationships that drive price fluctuations.Improved

Decision-Making: The developed model provides a practical tool for consumers, energy providers, and traders to make more informed decisions. Consumers can optimize their energy consumption patterns, energy providers can adjust their supply strategies, and traders can identify profitable trading opportunities, all of which contribute to cost savings and increased efficiency.

Risk Mitigation: The model offers a means of risk mitigation for energy market participants. Energy providers and traders can better manage financial risks associated with price volatility by using accurate price forecasts to inform their decision-making processes.

Energy Efficiency and Sustainability: By aligning energy consumption with periods of lower prices and higher renewable energy generation, consumers can contribute to a more sustainable energy ecosystem. The model supports the integration of renewable energy sources, leading to reduced carbon footprints and more efficient resource utilization.

Market Transparency: Access to reliable price forecasts enhances market transparency, promoting fair competition and enabling a smoother transition to cleaner energy sources. Informed decision-making and greater visibility into market dynamics benefit all stakeholders.

Algorithm Efficiency: The adoption of the Long Short-Term Memory (LSTM) neural network has proven effective in capturing the complex, time-dependent patterns present in electricity price data. LSTM's ability to handle sequential data efficiently, learn from long-term dependencies, and adapt to changing patterns has been pivotal in achieving accurate predictions.

Continuous Improvement: This project's dynamic nature ensures its relevance in a constantly evolving energy market. Regular monitoring and maintenance are integral to maintaining the model's accuracy over time. By updating the model with fresh data, it can adapt to changing market conditions and continue to provide valuable insights.