Let’s walk through each step used in the file based on the project to give a more detailed understanding of how it was structured, from loading the data to final model evaluation and conclusion:

**1. Introduction and Problem Statement:**

* The project focuses on **forecasting electricity demand and price** using **time series data**. The dataset consists of two parts: **energy consumption data** and **weather-related data**. These variables are highly interconnected, as electricity demand and prices are influenced by various external factors like temperature, humidity, and other weather conditions.
* The project goal is to use machine learning models to forecast electricity prices and demand accurately, which has wide applications in energy management, cost reduction, and ensuring grid stability.

**2. Loading and Preparing the Data:**

* The first operational step in the file involves loading two datasets:
  + **Energy dataset**: This contains information on electricity prices, demand, and generation.
  + **Weather dataset**: This dataset captures weather conditions like temperature, humidity, wind speed, etc.
* **Data Reading**: The data is loaded into the environment using pandas. CSV files are read for both energy and weather data, and a preliminary analysis of their structure is performed using methods like .head() to inspect the first few rows of the data.

**3. Data Cleaning and Preprocessing:**

* **Handling Missing Values**: Often, datasets contain missing or incomplete information. Filling in missing values or removing rows/columns with too much missing data is critical for making the data suitable for model training.
  + Techniques like **interpolation** or **forward fill** may have been used to handle time series gaps.
* **Merging Datasets**: Since two different datasets (energy and weather) are being used, they need to be **merged** into a single dataset using a common key, likely the date and time. This ensures that for each energy consumption record, the corresponding weather data is available for that same time period.
* **Feature Selection**: This step involves selecting the relevant features from both datasets for the model. Features might include:
  + From the energy dataset: electricity prices, energy demand, energy generation.
  + From the weather dataset: temperature, humidity, wind speed, etc.

**4. Feature Engineering and Normalization:**

* **Feature Engineering**: This step involves creating new variables from existing data that might help the model perform better. For example:
  + **Lag features**: Creating lag features for energy demand (e.g., energy demand from the previous day or hour) to help the model understand past patterns and how they affect the future.
  + **Rolling averages**: Taking moving averages of energy consumption or weather variables over a window of time to smooth out short-term fluctuations.
  + **Time features**: Extracting day, month, hour, or season information from the datetime variable to help the model understand periodic patterns.
* **Normalization/Scaling**: Models such as neural networks and tree-based models perform better when features are scaled similarly. For instance, energy demand may range from hundreds to thousands, while temperature ranges may be much smaller. **Min-Max scaling** or **Z-score standardization** is often applied to ensure all features have similar ranges.

**5. Exploratory Data Analysis (EDA):**

* **Visualizations**: The project likely employs visual tools such as:
  + **Line plots** to observe trends in electricity prices and demand over time.
  + **Scatter plots or heatmaps** to analyze the relationships between different variables (e.g., temperature vs. demand, humidity vs. price).
  + **Distribution plots** to understand the spread of the data and check for skewness.
* **Trend Analysis**: Identifying long-term trends, seasonality, or cyclic behaviors in the energy demand data, which is important for effective time series forecasting.
* **Correlation Analysis**: Understanding which weather factors (e.g., temperature, humidity) are most strongly correlated with changes in electricity demand and price.

**6. Data Splitting:**

* **Train-Test Split**: The dataset is divided into training and testing sets. The training set is used to train the models, while the testing set is reserved for model evaluation. Typically, in time series forecasting, the most recent data is used for testing (e.g., the last few months of data).
* The file likely uses a method like train\_test\_split() from sklearn or creates a manual split ensuring that the time order is preserved (since shuffling is not appropriate in time series data).

**7. Model Selection and Training:**

* The project explores multiple models for predicting electricity prices and demand. These models are evaluated based on their ability to capture the underlying patterns in the data:

1. **Linear Regression**:
   * This model attempts to establish a linear relationship between the dependent variable (e.g., electricity price) and independent variables (e.g., weather conditions).
   * In this case, the model performed poorly because electricity demand and price are influenced by complex, non-linear factors which the linear model fails to capture.
2. **CATBoost**:
   * **CATBoost** is a machine learning algorithm based on gradient boosting and is particularly good at handling categorical data and complex relationships. In this project, CATBoost performed significantly better than Linear Regression, as evidenced by the higher R² value and lower error rates.
   * CATBoost automatically handles categorical variables and has robust performance on tabular data, making it well-suited for this task.
3. **LightGBM**:
   * **LightGBM** is another gradient boosting framework that uses tree-based learning algorithms. It’s known for its speed and efficiency, especially with large datasets. LightGBM outperformed both Linear Regression and CATBoost, with the lowest MSE and RMSE, indicating its strong capability to model the non-linear and complex interactions in the data.
   * Its superior performance can be attributed to its ability to handle large datasets and complex interactions between energy demand, prices, and weather conditions.

**8. Model Evaluation:**

* **Mean Squared Error (MSE)**: Measures the average squared difference between predicted and actual values. Lower MSE indicates better model performance. Here, LightGBM had the lowest MSE, followed closely by CATBoost.
* **Root Mean Squared Error (RMSE)**: RMSE provides an interpretation of the prediction error in the original scale of the data (e.g., in terms of currency for price or energy units). LightGBM had the lowest RMSE, meaning it had the smallest prediction error on average.
* **R² (Coefficient of Determination)**: R² indicates how well the model explains the variability in the target variable (electricity prices and demand). LightGBM had the highest R² value (0.954), indicating that it explains 95.4% of the variance in the data, making it the best-performing model.
* **Adjusted R²**: Adjusted R² accounts for the number of predictors in the model, penalizing the use of too many variables. LightGBM also had the highest Adjusted R², confirming that its good performance wasn’t due to overfitting.

**9. Final Model Comparison and Selection:**

* Based on the evaluation metrics:
  + **Linear Regression** performed poorly, as shown by its extremely high MSE and RMSE values and negative R².
  + **CATBoost** provided a much better fit with a high R² (0.948) and reasonable MSE, but it was still outperformed by LightGBM.
  + **LightGBM** proved to be the most accurate model, achieving the best performance across all metrics, with an R² of 0.954, an MSE of 9.17, and the lowest RMSE of 3.03. This suggests that LightGBM was able to capture the complex relationships between electricity demand, prices, and weather factors most effectively.

**10. Conclusion and Insights:**

* **LightGBM** emerged as the best model for forecasting electricity demand and prices, outperforming both CATBoost and Linear Regression by a considerable margin.
* The project highlighted the importance of using advanced models like **gradient boosting** to handle the complex and non-linear relationships present in energy consumption and price forecasting.
* Accurately predicting electricity demand and prices using models like LightGBM has practical applications in energy management, such as optimizing energy production, reducing costs, and planning for periods of peak demand.

4o