Carnegie Mellon University Africa

DATA ANALYTICS

STATISTICAL ANALYSIS AND VISUALIZATION OF ENERGY DEMAND PATTERNS REPORT

By: Agnes Nakalembe

Imported libraries/Modules.

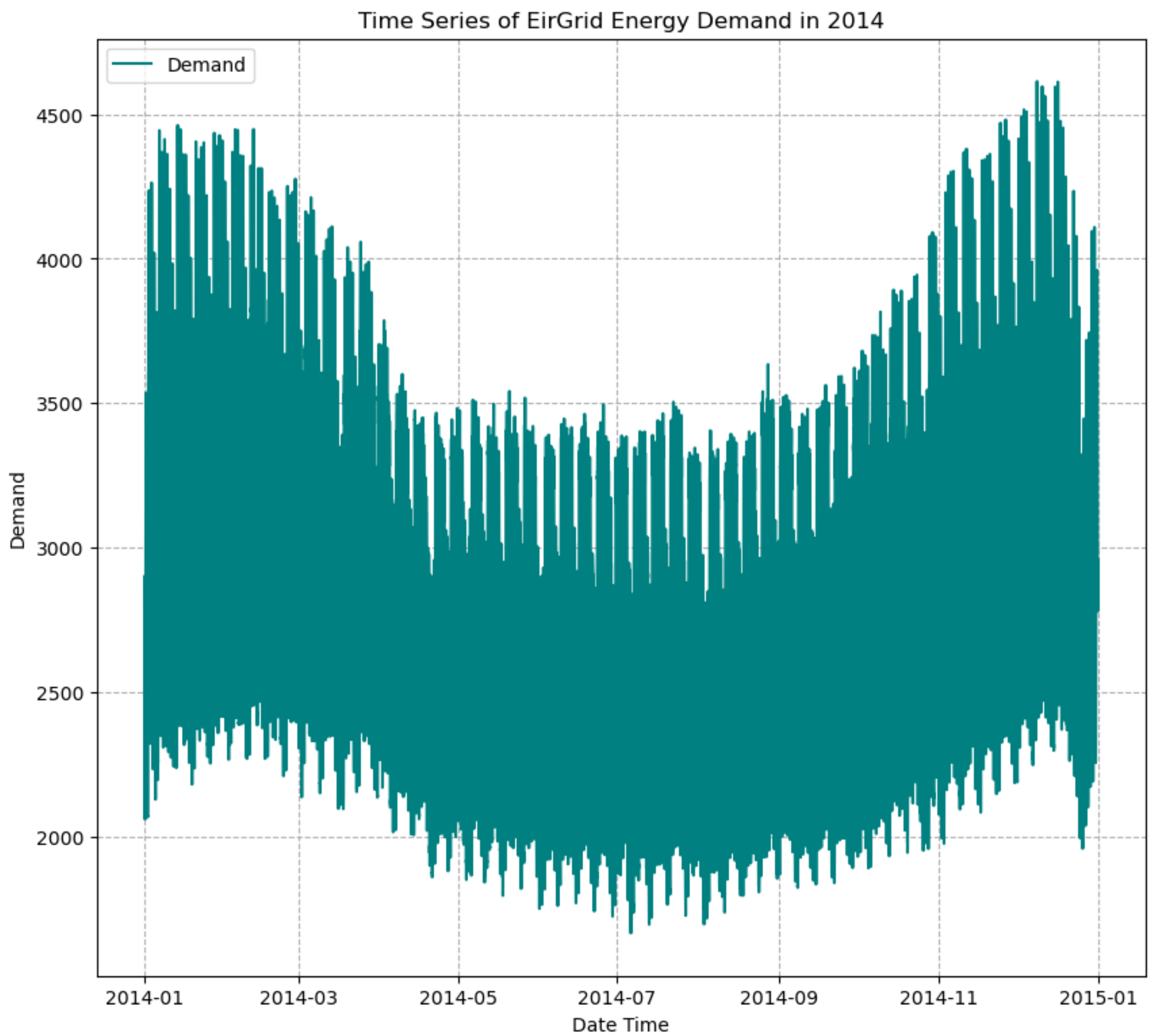
- pandas as
- matplotlib.pyplot
- stats from scipy
- statsmodels.api
- ttest_ind from scipy.stats
- mean_absolute_error, mean_absolute_percentage_error from sklearn.metrics

Analysis of Seasonal Energy Demand Variations

Procedure

- Having read EirGrid’s 2014 intraday 15-minute Energy Demand data into a pandas data frame, it was inspected for missing values. The missing values found, were then filled using linear interpolation by utilizing pandas interpolate function with the method parameter of this function set to ‘linear’.
- The ‘Date’ and ‘Time’ features of the data were merged to get a date time feature.
- A plot of Demand against Date-time was then plotted using Matplotlib’s pyplot module particularly using the plot function.

Results



Insights

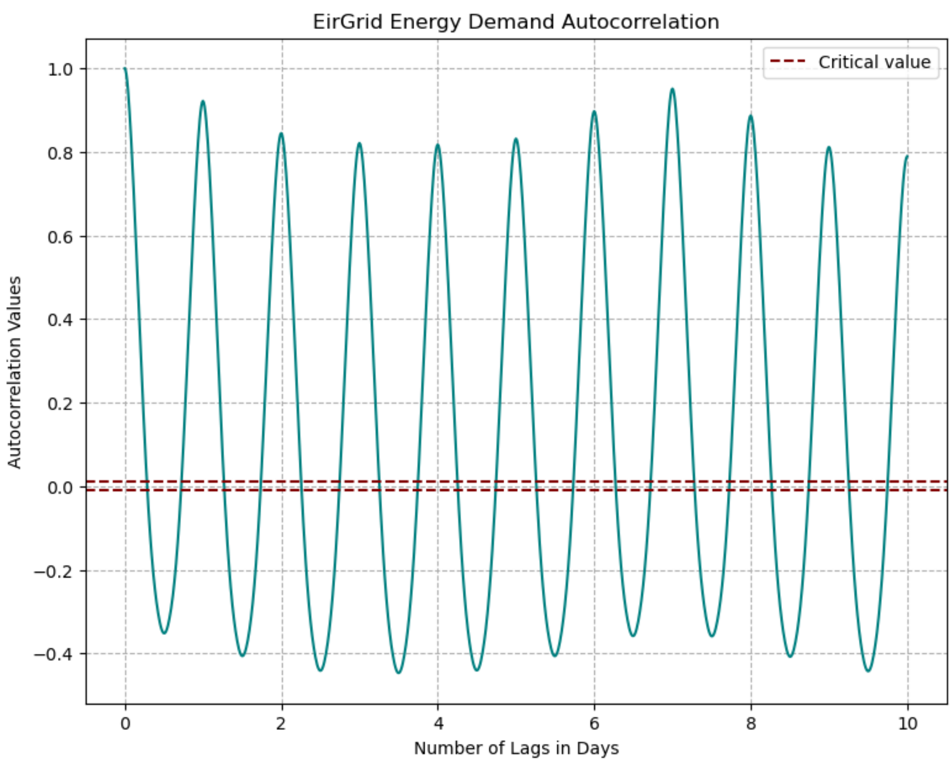
- From around September till the end of the year, there is an increase in the demand for energy. This could be attributed to the autumn and winter seasons of these months, during which there is great use of heating/warming devices.
- Between May and August, there is low demand for Energy. This could be due to the summer season, during which the temperatures are a bit warm, reducing the use of energy-intensive heating systems.
- Also, in January and February, the demand for energy is high. These are winter months, and therefore it could be suspected that during these times, people spend most of their time indoors, heavily using heating systems.
- On average, the energy demand takes a value concentrated between 2500 and 3000, through out the year.

Autocorrelation Analysis of Energy Demand

Procedure

- The c=autocorrelation co-efficients of EirGrid's energy demand were estimated by applying the acf function in statsmodels, with the number of lags specified as 960, which is equivalent to 10 days.
- These coefficients values were then plotted on a graph with the number of lags converted to days, using the plot function in matplotlib's pyplot module,

Results



Insights

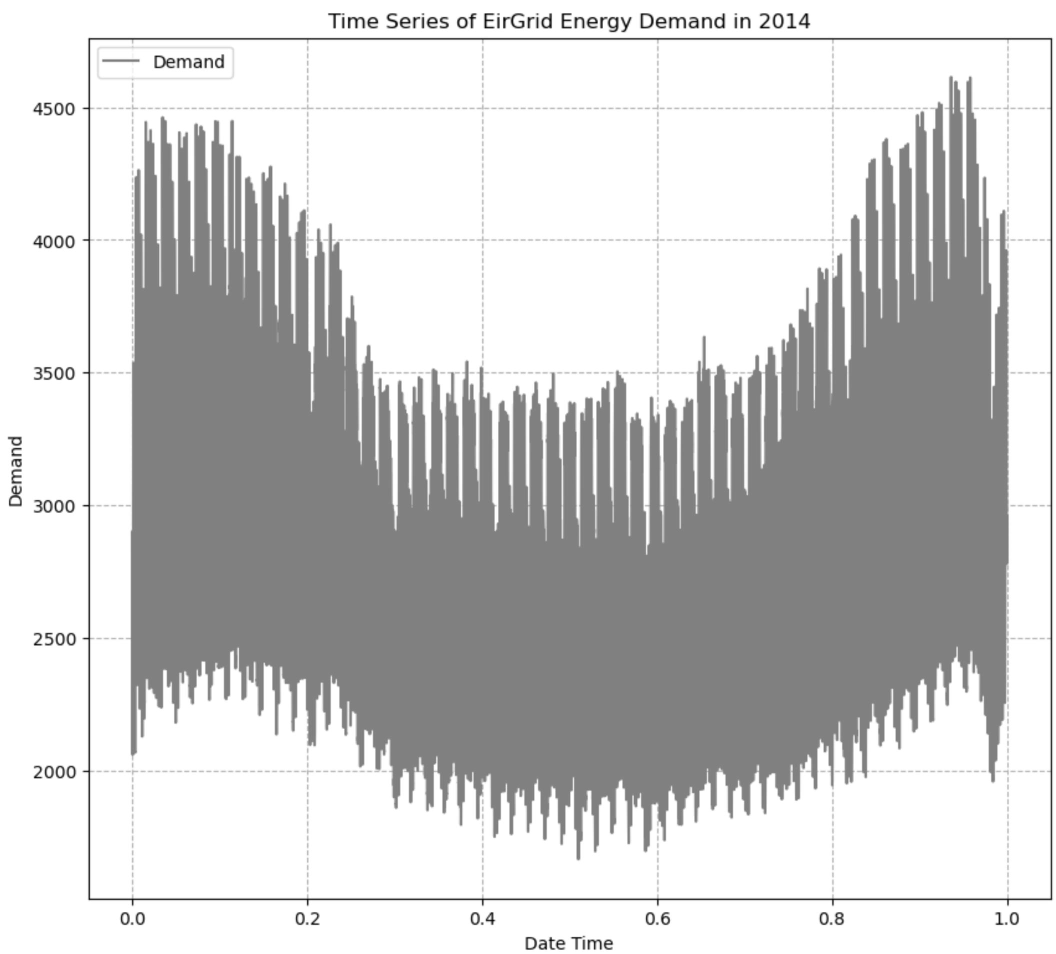
- The highest correlation values occurred for the 1 day and 7 days lag. This implies that today's energy demand can be a good predictor of tomorrow's demand. Also, the high ACF value at 7 days lag, is an indicator of a weekly pattern which in turn suggests that weekly data is a great predictor for energy demand.
- The above graph shows a consistent pattern in energy demand that has highs and lows repetitive. This pattern makes the energy consumption predictable.

Normalization and Trend Analysis of Energy Demands Throughout the Year

Procedure

- A ‘time of the year’ feature was obtained by normalizing the date time feature of the data. This normalization was done by subtracting the minimum date_time value from each date time value and then dividing the result by the difference of the minimum and maximum values in the date_time column.
- A plot of Demand against Date-time was then plotted using Matplotlib’s pyplot module particularly using the plot function.

Results



Insights

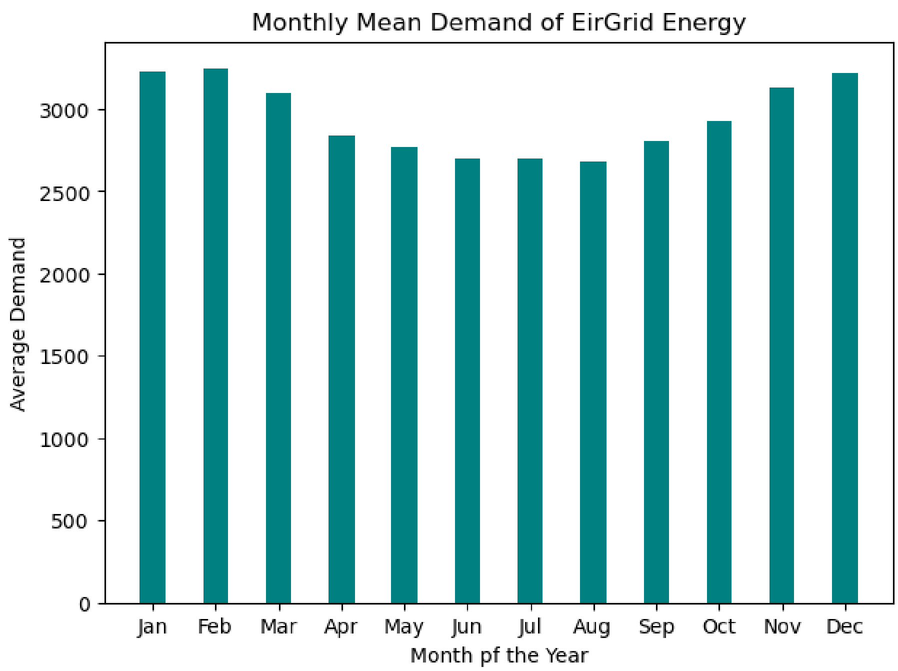
- For the initial third of the year, which includes the winter to early spring months, there is a high energy demand.
- In the two middle thirds of the year, which covers late spring, summer and early autumn, there is a low energy demand. This could be attributed to the warmer weather during this period compared to other months of the year, which reduce the use of heating systems.
- There is an increase in energy demand during the final third of the year, this could be because of the cooler weather of later autumn and winter that make people intensively use heating systems.
- On average, the energy demand takes a value concentrated between 2500 and 3000, throughout the year.

Monthly Energy Demand Distribution Analysis

Procedure

- Firstly, a 'Month' column was created in the data frame by extracting the month from each date time entry in the 'Date_Time' column.
- Then the average energy demand for each of the months was calculated by, grouping the demand values by this new 'Month' feature, and applying the mean function on the grouped monthly demand values.
- While utilizing the bar function in matplotlib's pyplot function, a bar graph of the monthly average demands was plotted.

Results



Observations and Insights

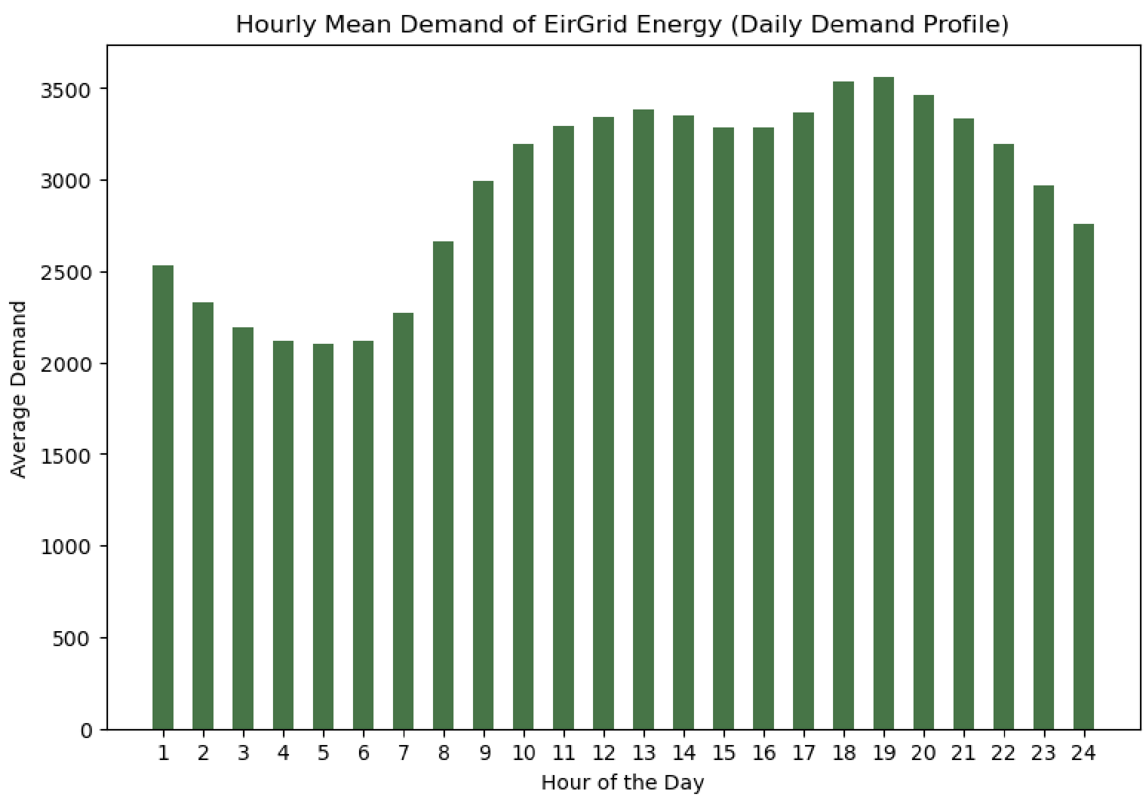
- December, January and February are the three months that had the highest average energy demand during the year 2014.
- June, July and August, are the three months that had the lowest average energy demand during the year. These are typically summer months, with warm temperatures that make people rely less on some of the high energy consumption systems to get warmth.

Daily Energy Demand Profile Analysis

Procedure

- To obtain the daily demand profile for the data, an 'Hour' column was first created in the data frame by extracting the hour from each date time entry in the 'Date_Time' feature.
- Then the average energy demand for each of the 24 hours in a day was calculated by, grouping the demand values by this new 'Hour' feature, and applying the mean function on the grouped hourly demand values.
- While utilizing the bar function in matplotlib's pyplot function, a bar graph of the hourly average demands was plotted.

Results



Insights

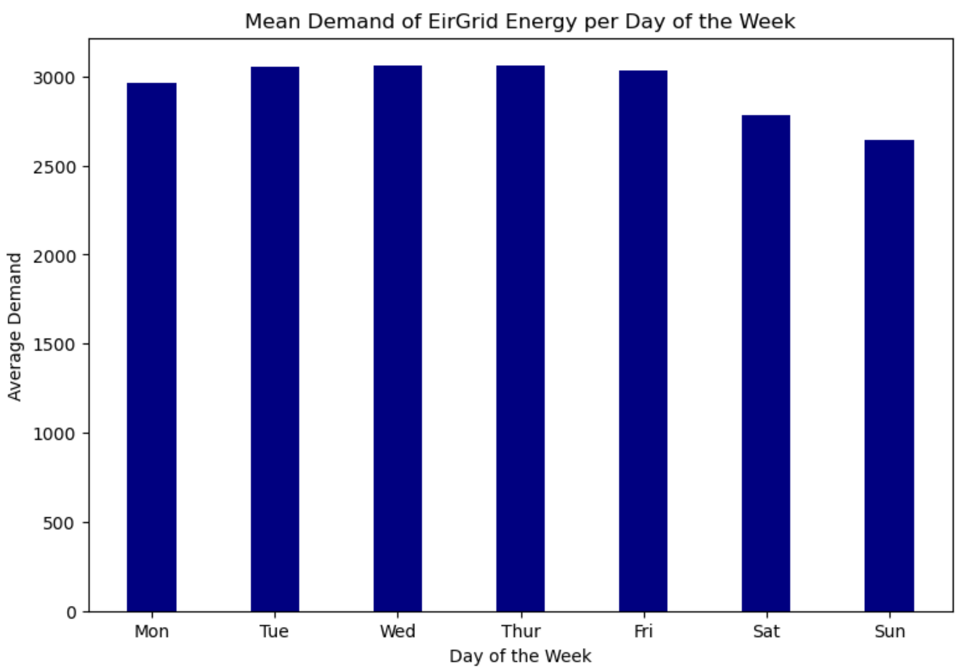
- During the first 6 hours of the day, there is a low demand for energy. This could be because during these hours, most people are asleep, most workplaces including industries are closed.
- From the 7th to around the 14th hours of the day, there is an increase in energy consumption. During these hours, people are awake, working intensively in their homes and offices, industries/businesses are operating, using several energy consuming systems, thus the high demand for energy.
- There is a slight drop in energy demand during the 15th and 16th hours of the day. I suspect that during these times, some workplaces close, and some people go for lunch during these hours.
- There is an increase in energy demand in the evening hours of the day (especially 18th and 19th hours). This could be attributed to the fact that some businesses continue operating, and at the same time, people get back from work to their homes, intensively engaging in tasks such as cooking food, using water heaters, using their home entertainment devices such as Televisions.

Weekly Patterns in Energy Demand

Procedure

- Firstly, the day of the week was extracted from the date time feature of entry in the data frame, obtaining a 'Day_of_Week' feature for the data.
- Then the average energy demand for each day of the week was calculated by, grouping the demand values by this new 'Day_of_Week' feature, and applying the mean function on the grouped daily demand values.
- While utilizing the bar function in matplotlib's pyplot function, a bar graph of the daily average demands was plotted.

Results



Insights

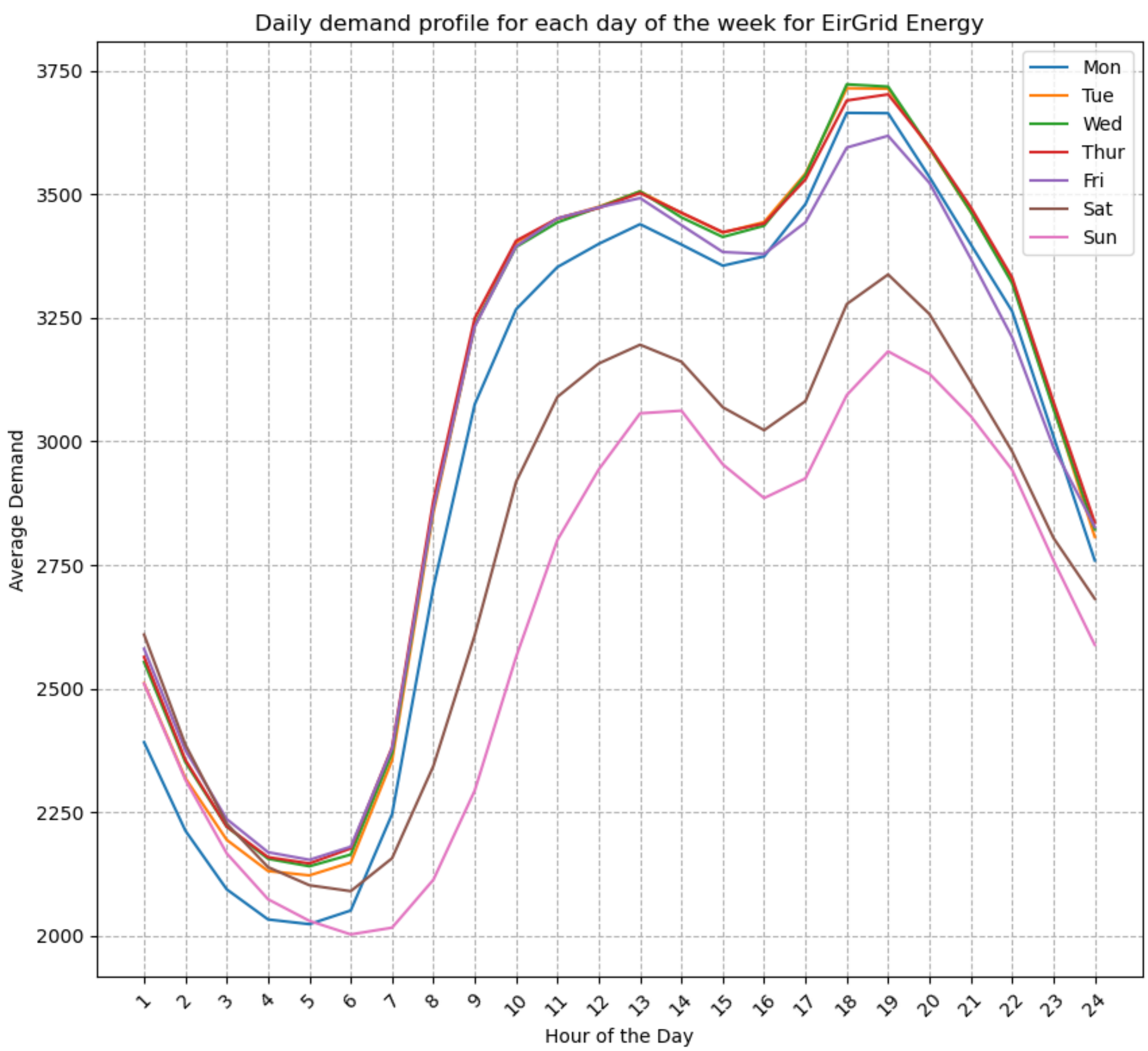
- There is a high demand for energy during the weekdays i.e. Monday, Tuesday, Wednesday, Thursday, and Friday. This can be attributed to the fact that these are the main workdays in most businesses/industries. During these days, high energy consuming systems such as electric machines in industries, both air conditioners and warmth devices, are being used. At the same time, people in homes use energy reliant devices and systems.
- Over the weekend, i.e. Saturday and Sunday, there is low average demand for energy. These days are considered rest days in most workplaces during which most of these workplaces/offices are closed hence the low demand for energy.
- This bar plot makes sense based on my intuition about energy consumption during the week-days and over the weekend.

Hourly Energy Demand by Day of the Week

Procedure

- The average hourly energy demand for each day of the week was computed. This was done by filtering the energy demand data for each day, grouping by the 'Hour' column, and then calculating the mean 'Demand'.
- A plot for each day's hourly average demand was created and labeled with the corresponding day of the week.

Results



Insights

- There is a kick-off of energy demand increase from the 6^{th} hour on all days of the week. Most people wake up around this time and start engaging in activities such as preparing breakfast/meals, taking heated baths, watching televisions, among others, which activities all require intensive electricity.
- This increase in demand for energy continues up to around the 14^{th} hour on all days of the week. This is correlated with the fact that, as some people stay in their homes continuing with activities such as cooking, using their electronic devices, most businesses and offices are open during these times.
- The highest peaks in energy demand are observed in all days of the week in the evening hours especially between the 18^{th} and 20^{th} hours. During these hours, some energy intensive businesses continue operating, as most people go back home and start using their home energy consuming systems such as heating systems, electronic devices.
- In as much as the hourly energy demand curves for all the days follow a similar trend, it is evident that the Saturday and Sunday curves take on lower values. This is consistent with the intuition that these days have most workplaces closed, implying that the energy demand during these days comes mainly from home related activities which are certainly not as energy consuming as the business activities specially manufacturing industries.

Statistical Test for Weekday versus Weekend Energy Demand

Null Hypothesis, H_0: There is no difference between the mean of the weekday energy demand and the mean of weekend energy demand.

Alternative Hypothesis, H_a: The mean of the weekday energy demand is different from the mean of weekend energy demand.

Procedure

- Two subsets of the data for weekday and weekend energy demand were created by filtering the energy demand dataframe by corresponding day of the week('Day_of_Week') feature. Records where 'Day_of_Week' ranged from 0 to 4 were classified as weekdays, while those greater than 4 were categorized as weekends.
- A t-test was performed to compare the means of the 'Demand' values between the two subsets using ttest_ind() function, resulting in the calculation of a p-value that indicates whether there is a statistically significant difference in energy demand between weekdays and weekend

Results and Insights

```
The P-value of 0.0 obtained is less than 0.05
Therefore, there is a significant difference between the weekend and
weekday energy demand
```

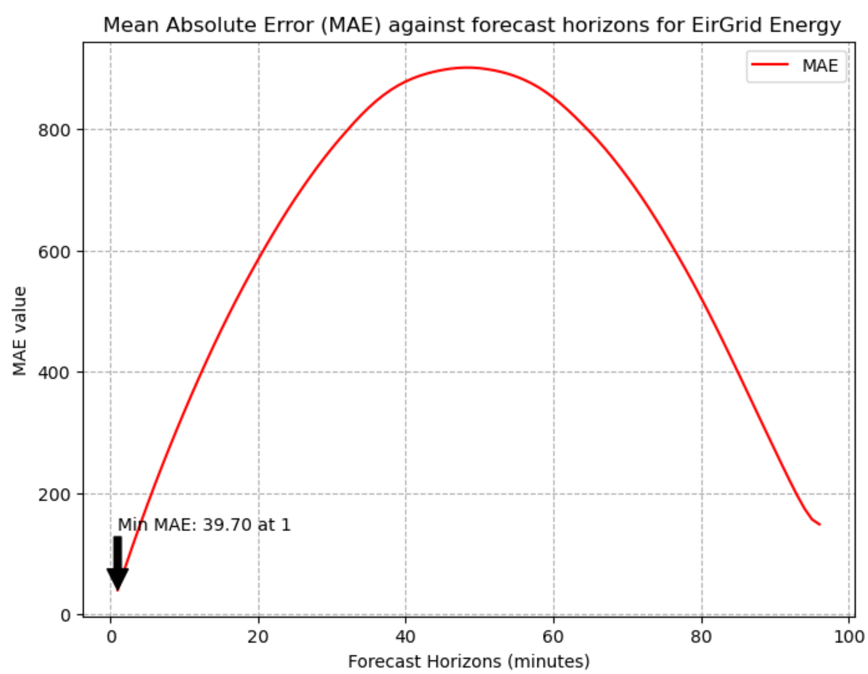
- The obtained p-value above indicated that the null hypothesis is rejected. This implies that the demand for energy during the weekdays is different from the energy demand during the weekend.
- Once again, this is evidence to the earlier obtained results that the energy demand during weekdays is higher than, therefore nor=t equal to, that of the weekend.

Forecasting Accuracy Analysis Using MAE for Energy Demand

Procedure

- The energy demand data was split into two halves to form the training and evaluation sets.
- A persistence forecast model was set up to predict 24 hours ahead, with forecasts made for every 15 minutes within a day.
- Mean Absolute Error (MAE) values were computed for each forecast horizon using the mean_absolute_error function in sklearn's metrics module and stored in a list.
- These MAE values were then plotted against the forecast horizons, creating a visual representation of forecast accuracy over time.

Results



Insights

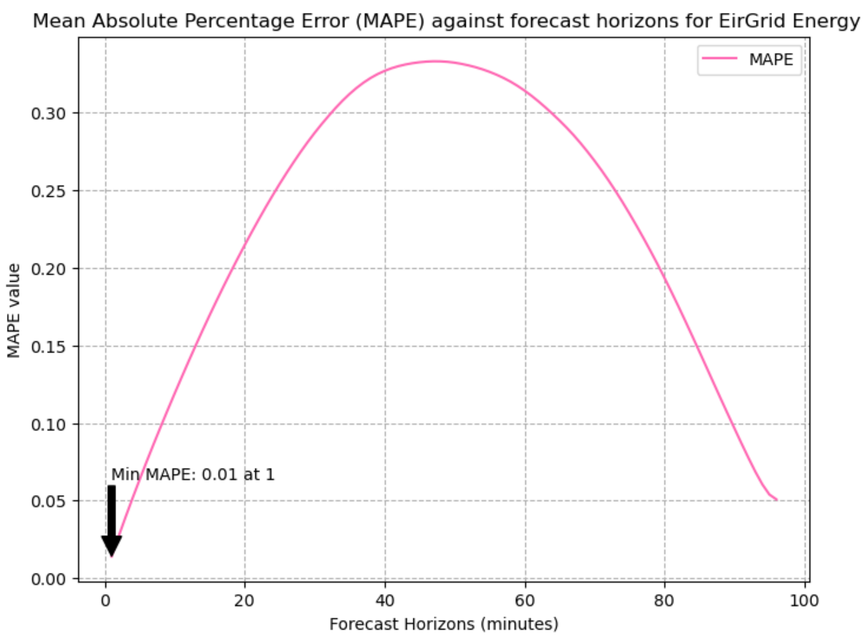
- The MAE is lowest when the forecast horizon is just one. This indicates that forecasting the energy demand is best using the most recent values.
- The MAE values increase as the forecast horizon increases until about the 50th -forecast horizon.
- Once the error hits its highest point, it starts to go down again as we look further ahead, up to horizon 100. This drop in error might mean there's a regular cycle in the energy demand which helps the model predict better.

Forecasting Accuracy Analysis Using MAPE for Energy Demand

Procedure

- The energy demand data was split into two halves to form the training and evaluation sets.
- A persistence forecast model was set up to predict 24 hours ahead, with forecasts made for every 15 minutes within a day.
- Mean Absolute Percentage Error (MAPE) values were computed for each forecast horizon using the mean_absolute_percentage_error function in sklearn's metrics module and stored in a list.
- These MAPE values were then plotted against the forecast horizons, creating a visual representation of forecast accuracy over time.

Results



Insights

- The MAPE is at its minimum value when a 1-forecast horizon is used. This shows that the most recent energy demand value gives the best idea on what could happen next.
- The MAPE values increase as the forecast horizon increases until about the 50-forecast horizon.
- After reaching its peak, the MAPE decreases for predictions going beyond the 50th step and up to the 100th. This reduction in error could suggest that the energy demand follows a pattern that makes predictions more accurate over these intervals.