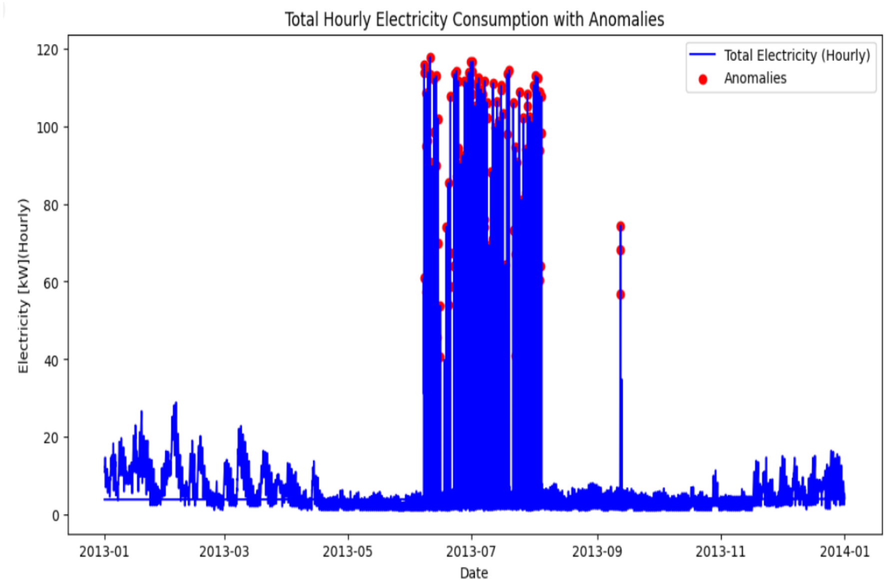
Assignment 1:

1. **Define NERC Holidays**:
    - Create a Helper Function:
        - Create a helper function `get_nerc_holidays(year)` to determine NERC holidays for a given year.
    - Fixed Dates:
        - Use fixed dates for holidays like New Year's Day, Independence Day, and Christmas.
    - Calculated Holidays:
        - Calculate holidays like Memorial Day, Labor Day, and Thanksgiving based on their rules (e.g., Memorial Day is the last Monday in May).
2. **Check NERC Holidays**:
    - Implement Holiday Check:
        - Implement `is_nerc_holiday(date)` to check if a given date is a NERC holiday.
3. **Handle Daylight Saving Time**:
    - DST Start and End:
        - Implement `is_dst_start(date)` and `is_dst_end(date)` to check if a given date is the start or end of DST.
    - Adjust Hours:
        - Adjust the number of hours accordingly when DST starts or ends, except for MISO which does not observe DST.
4. **Parse Period**:
    - Determine Start and End Dates:
        - Parse the period parameter to determine the start and end dates. The period can be annual, quarterly, monthly, or daily.
    - Date Manipulations:
        - Use the `datetime` and `calendar` modules to handle date manipulations.
5. **Calculate Hours**:
    - Initialize Counter:
        - Initialize `num_hours` to 0.
    - Iterate Over Each Day:
        - Iterate over each day in the period and add hours based on the peak type:
            - For onpeak, add hours only for weekdays and non-holidays from HE7 to HE22.
            - For offpeak, add hours for non-peak times or weekends/holidays.
            - For flat, add 24 hours for each day.
            - For 2x16H, add 16 hours for weekends and holidays.
            - For 7x8, add 8 hours for off-peak times during weekdays.
    - Adjust for DST:
        - Adjust for DST if necessary, based on the ISO and peak type.
6. **Return Result**:
    - Create Result Dictionary:

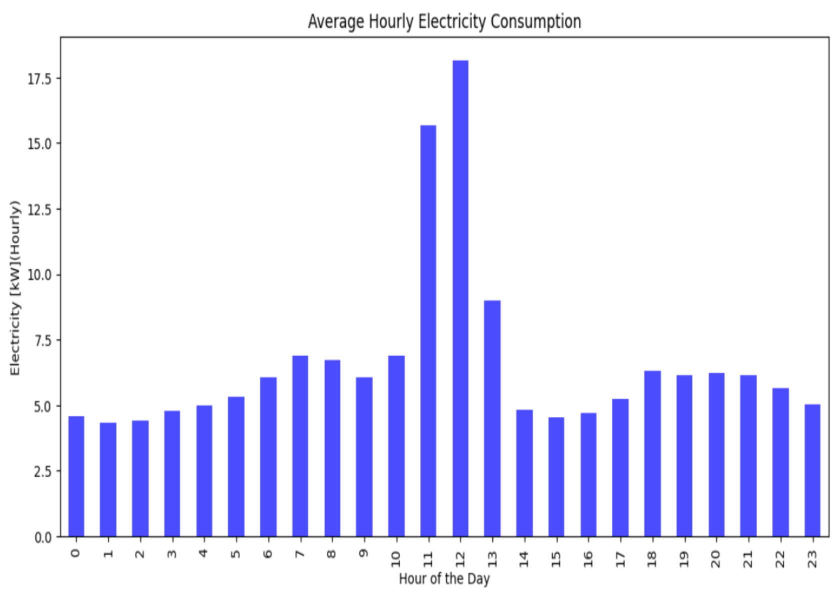
    - Create a result dictionary containing the ISO, peak type, start date, end date, and total number of hours.
- Return Dictionary:
    - Return this dictionary.

Assignment 2:

1. **Data Processing Steps**:
   - Loading Data:
     - Loaded two datasets: one containing hourly electricity consumption and another with minute-by-minute appliance consumption.
   - Data Cleaning:
     - Handled issues such as removing an unnamed column from the second dataset and fixing the '24:00:00' format in the first dataset.
   - DateTime Handling:
     - Converted the time columns to datetime format and adjusted the years to match between the datasets.
   - Resampling Data:
     - Resampled the minute-by-minute data to hourly data, converting units from watts to kilowatts.
   - Merging Data:
     - Merged the datasets on their respective datetime indices.
   - Calculating Total Consumption:
     - Created a new column for the total hourly consumption by summing all relevant columns.
   - Anomaly Detection:
     - Used the Z-score method to identify anomalies in the total hourly consumption and visualized them using a plot.
   - Pattern Visualization:
     - Visualized the average hourly, weekly, and monthly electricity consumption patterns to identify any trends.
   - Documentation and Output:
     - Wrote clear code documentation and saved the merged dataset to a CSV file.
2. **Data Visualization and Analysis**:
   - Total Hourly Electricity Consumption with Anomalies:
     - **Description**: This plot shows the total hourly electricity consumption over time, with anomalies highlighted in red.
     - **Analysis**: There is a significant increase in electricity consumption during the middle part of the year, particularly around July. The anomalies, marked in red, correspond to periods of extremely high consumption, indicating potential issues or unusual usage patterns. Outside this peak period, the electricity consumption remains relatively stable and low.

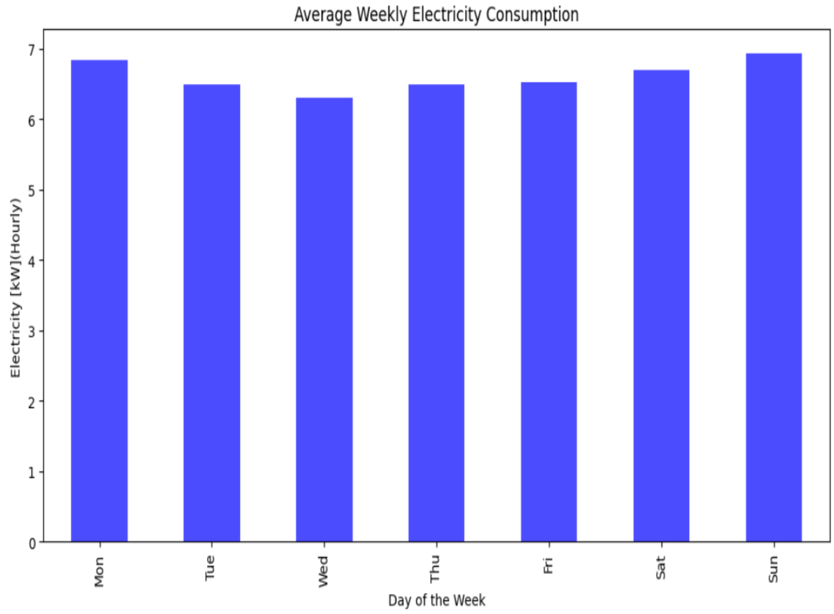


- Average Hourly Electricity Consumption:
  - **Description**: This bar chart illustrates the average electricity consumption for each hour of the day.
  - **Analysis**: Electricity consumption peaks around noon (12 PM), with another noticeable increase starting from 6 AM and a smaller peak at 6 PM. Consumption is lowest during the early morning hours (12 AM to 5 AM), which is typical as people are generally asleep or not at home.

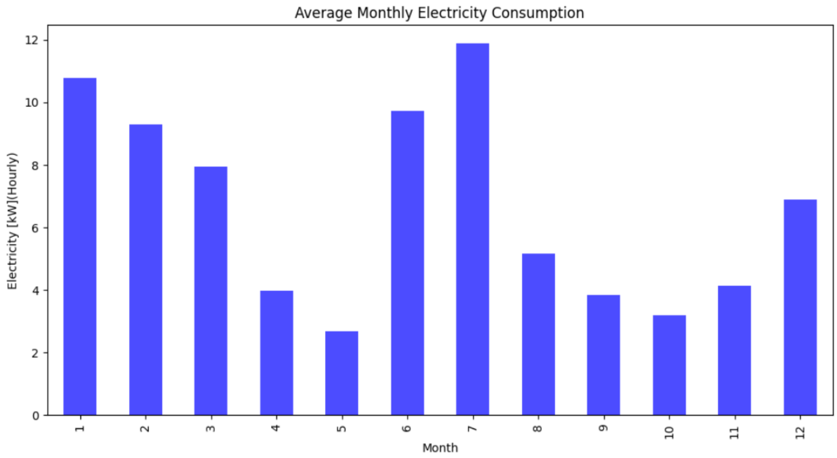


- Average Weekly Electricity Consumption:
  - **Description**: This bar chart displays the average electricity consumption for each day of the week.
  - **Analysis**: The consumption is fairly consistent throughout the week, with only slight variations. There are no significant differences between weekdays and weekends, indicating a

stable usage pattern without significant changes based on the day of the week.

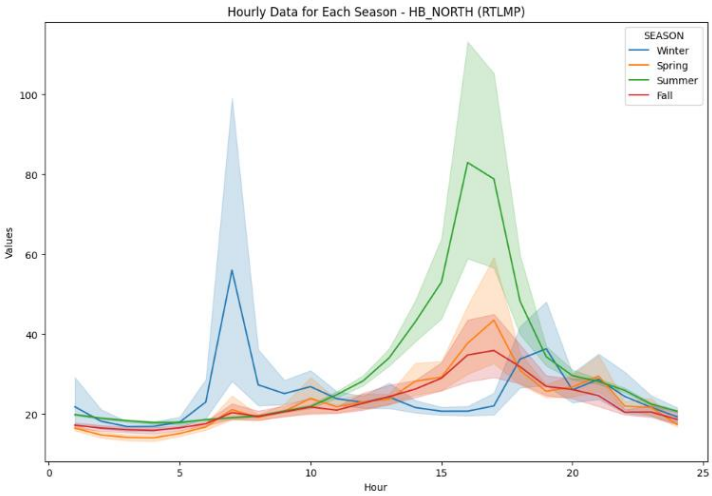


- Average Monthly Electricity Consumption:
  - **Description**: This bar chart shows the average electricity consumption for each month.
  - **Analysis**: There is a clear seasonal pattern, with higher electricity consumption in the summer months (June and July), likely due to increased use of air conditioning. Another peak is observed in January, which could be attributed to heating. The consumption is lower during the spring and fall months (April, May, September, October), which are typically milder in temperature.

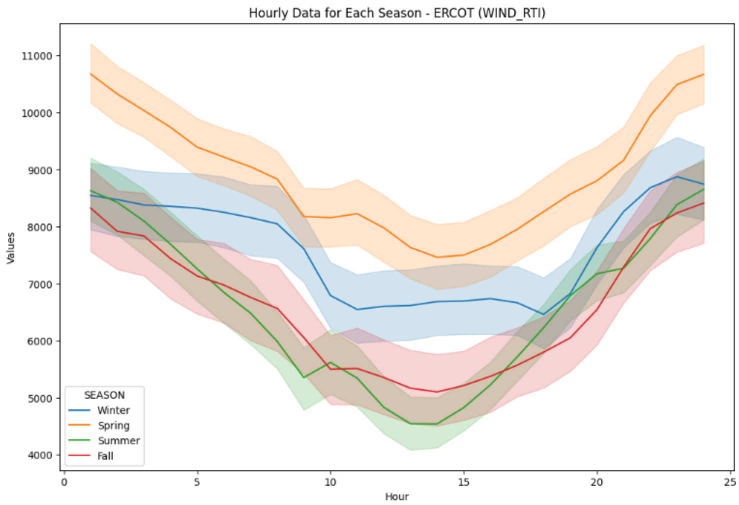


Assignment 3:

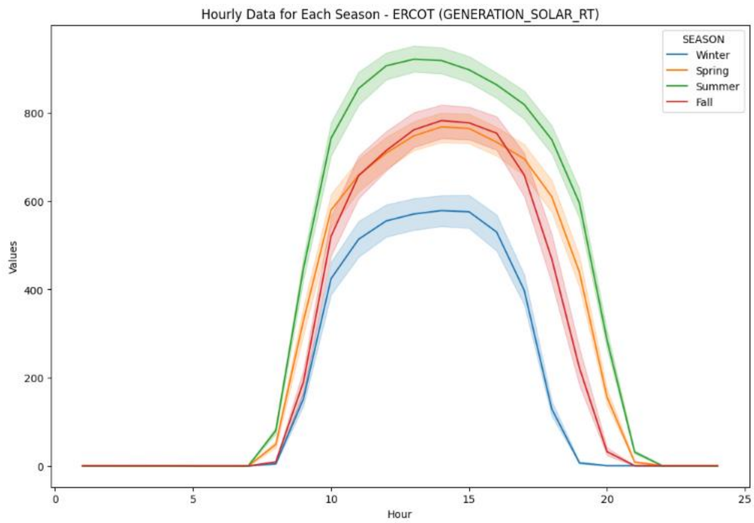
1. **Exploratory Data Analysis**:
    - Hourly data for each season for each column is shown below. Shaded areas around the lines represent the confidence interval, indicating the uncertainty around the mean (or other estimator) and is computed using bootstrapping. This provides a visual representation of the variability in the data.
    - **HB_NORTH (RTLMP)**:
        - **Hourly Data for Each Season**: The price reaches its highest point between 15:00 to 18:00, with significant peaks in winter around 6:00-7:00 and in summer from 15:00-18:00. Fall generally shows lower prices compared to other seasons.
        - **Insight**: This indicates higher prices during the afternoon and evening peak hours, with winter mornings also experiencing high prices, likely due to increased heating demand.



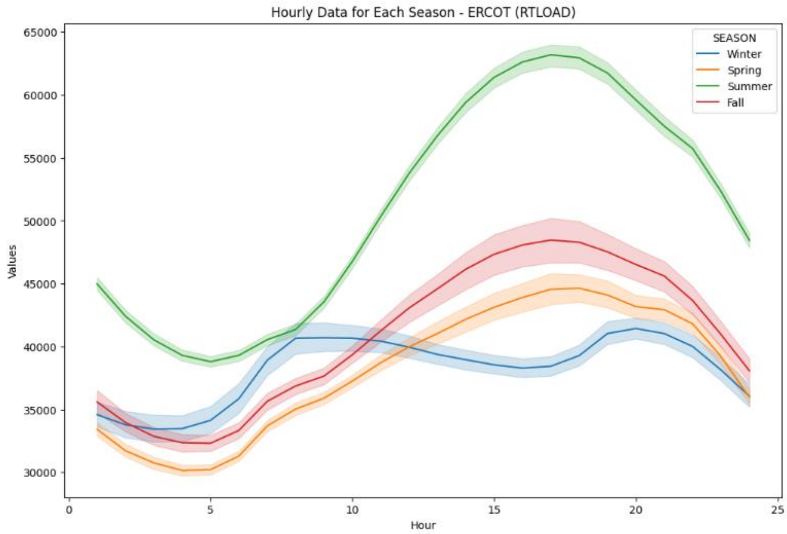
    - **ERCOT (WIND_RTI)**:
        - **Hourly Data for Each Season**: Wind energy production shows a decrease during the day and an increase in the evening and early morning, with the highest values in spring and the lowest in winter.
        - **Insight**: Wind patterns suggest a dip during midday hours, potentially due to diurnal wind patterns.



- **ERCOT (GENERATION_SOLAR_RT)**:
  - **Hourly Data for Each Season**: Solar energy production follows a typical daily pattern, peaking around midday and dropping to zero during night hours, with summer showing the highest production.
  - **Insight**: Solar generation is highest during sunny summer months, aligning with daylight hours.



- **ERCOT (RTLOAD)**:
  - **Hourly Data for Each Season**: Energy demand peaks from 15:00 to 20:00, with winter showing an additional peak around 7:00-9:00. Summer has the highest overall demand.
  - **Insight**: Energy demand peaks during the late afternoon to evening hours, and winter mornings also see high demand, which corresponds to increased heating needs.



2. **Correlation Analysis**:
   - Negative Impact of Wind Generation: Higher wind generation tends to lower electricity prices, possibly due to the increased supply of electricity from wind.
   - Positive Impact of Solar Generation: Higher solar generation is associated with higher electricity prices, which might be influenced by factors such as time of day and demand patterns.
   - Seasonal Variations: Significant seasonal effects on electricity prices and real-time load are observed, with summer showing higher demand and prices.
   - Peak Type Influences: Prices are lower during off-peak hours and higher during weekday peak hours, reflecting typical demand patterns.
3. **Data Preprocessing**:
   - Handle Missing Values:
     - Checked for missing values.
     - Replaced missing values with 0.
   - Handle Negative Values:
     - Counted the number of negative values in numerical columns. There were 75 rows containing negative values.
     - Deleted rows containing negative values.
   - Outlier Detection and Replacement:
     - Defined a function `replace_outliers_iqr` to replace outliers based on the Interquartile Range (IQR).
     - Applied this function to specific columns to replace outliers with calculated bounds.
   - Log Transformation:
     - Because the range of original data is large (from -17.860000 to 2809.357500) and the standard deviation is two times higher than the mean, indicating a skewed distribution which can affect the assumptions of many statistical and machine learning models that assume normally distributed data, I applied a log transformation to specified columns to normalize the data.
   - Datetime Processing:
     - Ensured the 'DATETIME' column is in datetime format.

        - Extracted hour, month, and season information from the 'DATETIME' column.
        - Created a 'SEASON' column based on the extracted month.
    - One-Hot Encoding:
        - Applied one-hot encoding to categorical columns ('SEASON', 'PEAKTYPE', 'MONTH', and 'HOURENDING') to convert them into numerical format.
    - Define Features and Target:
        - Defined the feature matrix X by dropping irrelevant columns ('DATETIME', 'HB_NORTH (RTLMP)', 'MARKETDAY', 'YEAR').
        - Defined the target variable y as 'HB_NORTH (RTLMP)'.
    - Split Train Set and Test Set:
        - 80% of the feature matrix X and target variable y are set as training data and 20% of the feature matrix X and target variable y are set as test data.
4. **Models**:

Each model was trained with the same input and using hyperparameter tuning.

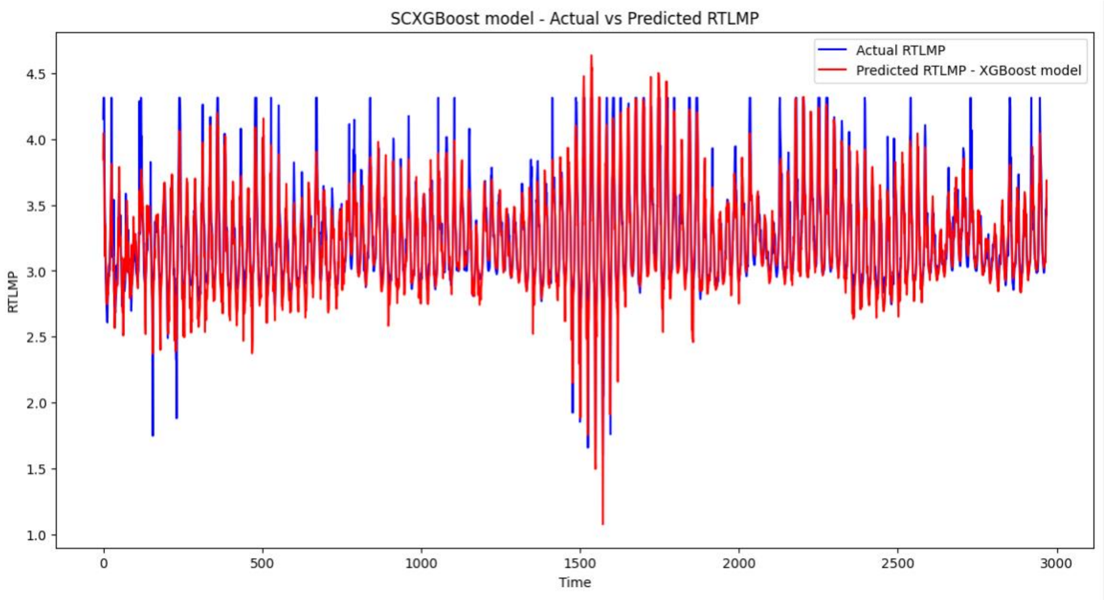
- Linear regression model (LR)
- Random forest model (RF)
- Separated Random forest model (SRF):
    - Developed three separate RandomForestRegressor models to predict 'HB_NORTH (RTLMP)' values. The first model will be trained for the hours between 6 and 8, the second for the hours between 16 and 18, and the third for the remaining hours of the day. After training these models, three predictions were combined to evaluate the overall performance using appropriate metrics.
- XGBoost
- Gradient Boosting Regressor (GBR)
- Long-Short Term Memory model (LSTM):
    - Using past 24 hours to predict the next hour.
- LASSO principal component averaging (LPCA):
    - Combines PCA and LASSO, where PCA extracts principal components from a large panel of forecasts, and LASSO is used to select the most relevant components by shrinking less important parameters to zero.
- Separated Enhanced LASSO principal component averaging (SELPCA):
    - Developed three separate LPCA models to predict 'HB_NORTH (RTLMP)' values. The first model will be trained for the hours between 6 and 8, the second for the hours between 16 and 18, and the third for the remaining hours of the day. After training these models, three predictions were combined to evaluate the overall performance using appropriate metrics.
- Seasonal Component with XGBoost (SCXGBoost):

- Use STL (Seasonal and Trend decomposition using Loess) to decompose the time series into trend, seasonal, and residual components and use XGBoost to model the residual.

Matrics of different models

	LR	RF	SRF	XGBoost	GBR	LSTM	LPCA	SELPCA	SCXGBoost
MAPE (%)	4.449	3.650	4.483	21.584	22.345	5.500	5.340	3.080	2.90
R2	0.614	0.677	0.616	0.62	0.593	0.507	0.545	0.821	0.865
MAE	0.142	0.11	0.140	0.138	0.142	0.175	0.162	0.104	0.095
MSE	0.046	0.039	0.062	0.072	0.077	0.130	0.068	0.032	0.019

The best model is Seasonal Component with XGBoost (SCXGBoost), I used STL (Seasonal and Trend decomposition using Loess) to decompose the time series into trend, seasonal, and residual components and used XGBoost to model the residual. This indicates that seasonal trend exists and can be used to predict the future. And XGBoost captures the indescribable and non-linear relationship well.



Assignment 4:

Assumption

1.You hold 100 MW of physical power and the corresponding position in Product 1 for December 2016. And Strip of these 12/1/2016 The settlement price for Product 1 (Power Futures - ERN) is based on the average daily electricity prices at the ERCOT North 345KV hub.

2.The value change of 100 MW physical power is the same as the value change of 100 product 1 contracts.

3.Trading cost can be ignored.

4.Total Peak hours from 01/26/2016 to 11/28/2016 is 4160 hours.

5.From the given Excel file, ERN settlement price=H settlement price*XPR settlement price, but it may not always be true in the real market.

Position

Physical Power: Long (Guzman Energy holds a long position in physical power due to ownership or PPAs.)

Product 1 (Power Futures - ERN): None, because Product 1 has no liquidation in the market and we are holding the physical power.

Product 2 (Natural Gas Futures - H): Short (We hold physical power, so we short natural gas futures to hedge.)

Product 3 (Heat Rate Futures - XPR): Short (We hold physical power, so we short natural gas futures to hedge.)

From the given Excel file, ERN settlement price=H settlement price*XPR settlement price, but it may not always be true in the real market.

Number of Contracts

1. Number of Contracts for Product 2:

Heat Rate at 2016/1/26: 9.07 MMBtu/MWh

Total Power: 100 MW * 4160 hours = 416000 MWh

Total Natural Gas Needed: 416000 MWh * 9.07 MMBtu/MWh = 3773120MMBtu

Number of Natural Gas Futures Contracts: 3773120MMBtu / 2500 MMBtu/contract ≈ 1509 contracts

I recalculate the number of contracts for product 2 every week.

2. Number of Contracts for Product 3 (Heat Rate Futures - XPR):

Contract Size:

50 MW per hour.

Calculation:

Since your position is 100 MW, you need 2 contracts of Product 3 to hedge this position.

Number of Product 3 Contracts=100MW/50MW per contract=2contracts

Change of Value

Physical Power: 100MW*4160 hours*(ERE settlement price at T2- ERE settlement price at T1)

Product 2 (Natural Gas Futures - H): -number of contracts *2500*(H settlement price at T2- H settlement price at T1)

Product 3 (Heat Rate Futures - XPR): -100MW*4160 hours*(H settlement price* XPR settlement price at T2- H settlement price* XPR settlement price at T1)

I use Excel to calculate the required number of contracts of Product 2 every week (since it is weekly rebalanced), value changes of each product.

Reference

[1] Maciejowska, K., & Weron, R. (2015). Forecasting of daily electricity prices with factor models: Utilizing intra-day and inter-zone relationships. *Computational Statistics*, 30(3), 805–819.

[2] Uniejewski, B., & Maciejowska, K. (2023). LASSO principal component averaging: A fully automated approach for point forecast pooling. *International Journal of Forecasting*, 39, 1839–1852.

[3] Uniejewski, B., Nowotarski, J., & Weron, R. (2019). Modeling and forecasting electricity loads and prices: A review of methods and results. *International Journal of Forecasting*, 35(1), 1520–1532.

[4] Mayer, K., & Trück, S. (2018). Electricity markets around the world. *Journal of Commodity Markets*, 9, 77–100.

[5] Maciejowska, K., & Weron, R. (2018). Econometric modeling and forecasting of intraday electricity prices. *Journal of Commodity Markets*, 19, 100107.

[6] Olivares, K. G., Challu, C., Marcjasz, G., et al. (2023). N-BEATS: Neural basis expansion analysis for interpretable time series forecasting. *International Journal of Forecasting*, 39, 884–900.