Assignment Report

GitHub Repo: https://github.com/Jimmmnny/YuxianWang_summer-2024.git
Yuxian Wang

Section 1: Power Calendar function

In question 1, a function get_hours(iso, peak.type, period) need to be built, which can show the number of hours if the iso, peak type, and period can be provided. Since there are lots of condition need to be considered, we need to prepare some setting and functions in advance.

- Generate NERC holidays

The first one is we need to generate the NERC date for specific year, since NERC holidays would make a difference to the hours counting. Besides, since there are some holidays are not on fixed date, like Memorial, Labor, and Thanksgiving Day. For this function, there are two other functions are needed, the first one is getting the nth weekday we want, which is for Labor Day and Thanksgiving Day. The other is getting the last weekday of the month, which is for Memorial since Memorial is on the last Monday in May. With this NERC function, we can get the NERC dates only if we input the year.

- ISO config

The second part is ISO config. Eastern power market has a set of setting for hours counting in different peak types. Thus, I defined the calculation logic here for all ISO and peak types. In eastern power market, they share the same calculation logic, but only different in the daylight-saving setting. MISO doesn't need to take the daylight-saving setting into account.

- Daylight-saving setting

As I mentioned, daylight-saving setting is needed to take care. Since the beginning and end date of daylight-saving setting is also unfixed in each year, I can easily use the function to get the nth weekday, which I wrote for NERC holiday function. I set the year as an input; thus, I can get the start date on the second Sunday in March of the year, and the first Sunday in November of the year as the end date.

- Leap year

Whether the year is a leap year also important, since it will determine the if there is 29 days in February, which will impact the hour counts.

- Get day number in month

To simplify the get_hours function, I wrote a function to get the number of days in month directedly, the input is year and month, the output is the number of days in such year and month.

- Parse period

This part is to parse the date information. There are four types of date input, daily, monthly, quarterly, and annually. Thus, I wrote this function to recognize the input type and output the right start date and end date.

Depend on all the previous settings and functions, it is more easily to write the get_hours function now.

1. Parse the period first, get the start date and end date
2. Generate NERC holiday
3. Get the start and end time of daylight-saving setting
4. Loop from current date to the end date, update the hours counting within the holiday, weekday, weekends logic. In this loop, when the date is equal to the start date of daylight-saving setting, total

hours would minus one, and for the end date of daylight-saving setting, the total hours would plus one.

5. Finally, we can get all the information we want

Section 2: Meter Data formatting

The objective of question 2 is to merge different data sources into single dataset and evaluate the dataset for anomaly.

1.Merge

There are two datasets, new.app4.csv is one appliance's electricity consumption minute by minute, and the unit is in watt, we call it dataset_1; USA_AL_Auburn-Opelika.AP.722284_TMY3_BASE.csv provides hourly electricity consumptions, and the unit is in kw, we call it dataset_2.

- Dataset _1 processing

I summed the watt within each hour and changed the time format to match the format of dataset_2, then set the formatted timestamp as index, which can be used for dataset merging. Finally, I transformed the unit from watt to kw.

- Dataset_2 processing

The hour setting in dataset_1 is 00:00:00 to 23:00:00, but the hour setting in dataset_2 is 01:00:00 to 24:00:00. To solve this problem, I marked all the 24:00:00 columns, replaced it with 00:00:00 and add 1 day to the date. Then, same as dataset_1, I formatted the timestamp, and set it as index, making it prepared to be merged.

- Merge, add total hourly consumption and output

The final step is to merge two datasets by the formatted timestamp. Right merging is used because dataset_2 rows are more than the dataset_1. Then I sum all columns for total hourly consumption as new column. Finally, I dropped out two useless columns, and output it as csv file.

2.Anomaly detection

It is not a good choice to conduct anomaly detection on the whole merged dataset directly, since dataset_1 only has 2461 rows after processing, and dateset_2 has 8760 rows. Obviously, there are some time slots have zero appliance's electricity consumption. Thus, anomaly detection will be divided into two parts.

1. Resident electricity consumption

Since these 13 features are highly related, multivariate gaussian distribution is a powerful tool for anomaly detection. Multivariate gaussian distribution can capture the relationships between features, especially for such high dimensional data. Even if the features distributions are not gaussian distribution, this anomaly detection tool is still powerful.

Firstly, all features are standardized to prevent data of different magnitudes from affecting the model. Secondly, mean and covariance metric are computed, which is key parameter of the model development. Then multivariate gaussian distribution was built for P-value computation. I set the threshold at 1% here, which means 1% data points will be marked as anomaly, then I got 88 anomaly data points.

To analysis the anomaly data points further, I conducted cluster analyze on the anomaly data points. The first step is utilizing Principal Component Analysis (PCA) to reduce the data dimensions, I chose 3 components based on the cumulative explained variance ratio, which is over 90%.

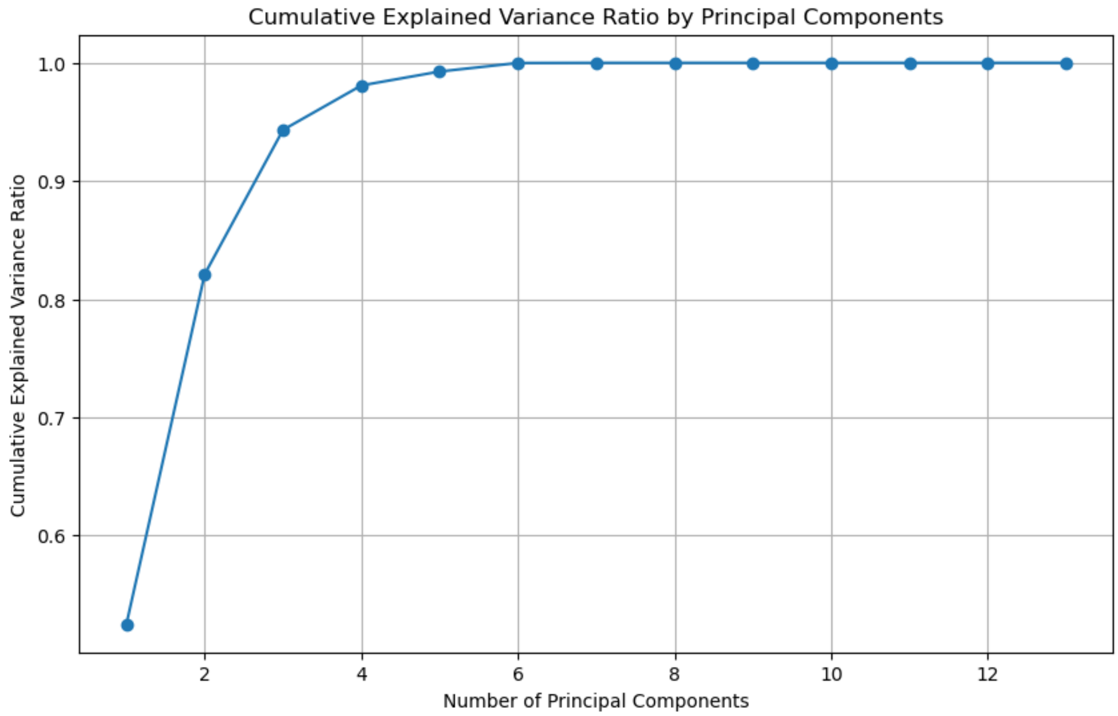


Figure 1. Cumulative explained variance ratio

K-means algorithm was conducted on the 3 components. To determine the optimal cluster number, I used Elbow Method to find the optimal K, which we can see from the figure which is 3 too. With optimal k equals to 3, I can get 3 groups of anomaly data points.

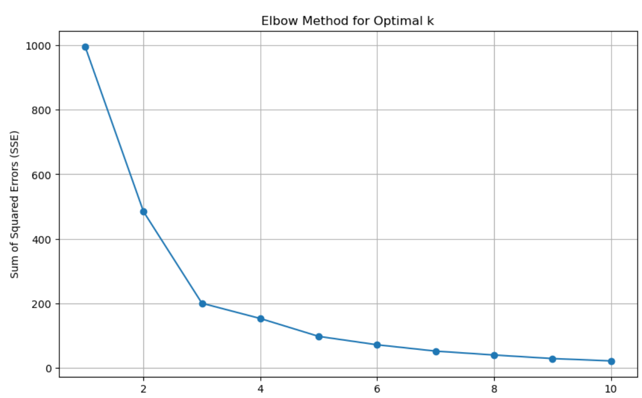


Figure 2. Elbow method

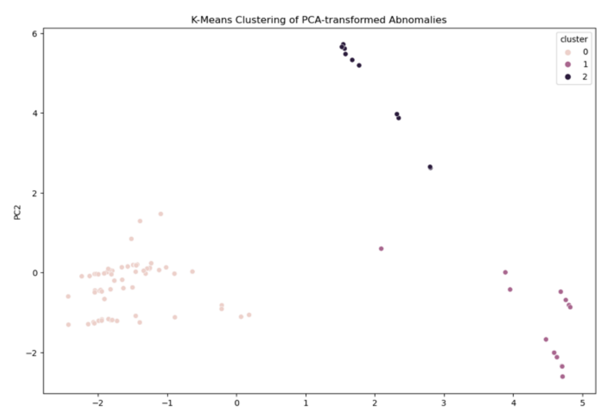


Figure 3. Clusters

Given these 3 groups of anomaly data points, we can find some anomaly from the describe. All clusters and normal data points mean are calculated in the following table.

Table 1. Mean value of clusters and normal data points

Features[kW](Hourly)	Cluster 0	Cluster 1	Cluster 2	Normal
Electricity:Facility	2.4955	2.2600	1.4013	1.4273
Gas:Facility	0.0244	1.7249	12.0558	1.0784
Heating:Electricity	0.0000	0.0000	0.0000	0.0000
Heating:Gas	0.0000	1.6861	12.0277	1.0515
Cooling:Electricity	1.1485	0.0155	0.0000	0.1325
HVACFan:Fans:Electricity	0.3120	0.0473	0.2950	0.0663
Electricity:HVAC	1.4606	0.0628	0.2950	0.1988
Fans:Electricity	0.3120	0.0473	0.2950	0.0663
General:InteriorLights:Electricity	0.0659	0.6949	0.1183	0.1957
General:ExteriorLights:Electricity	0.0142	0.1497	0.0255	0.0422
Appl:InteriorEquipment:Electricity	0.3372	0.3571	0.1141	0.2429
Misc:InteriorEquipment:Electricity	0.3545	0.5642	0.4086	0.3908
Water Heater:WaterSystems:Electricity	0.2435	0.3847	0.4202	0.3384

Compare with normal data points, the time slots of cluster 0 obviously consume more electricity, including cooling, HVAC fans, HVAC, and interior equipment. And the electricity consumption of water heater is more concentrated with small variance. Combine with the situation that they almost never use gas for heating, it seems like a hot summer.

In the time slot of cluster 1, residents pretend to use more gas for heating, more electricity for interior and exterior lighting, and interior equipment. It should be assumed at dark winter night.

As for the time slot of cluster 2, residents consume far more gas than ever for heating, and more electricity for HVAC fans, and fans. While other consumptions are similar to normal data points, the distribution of water heater consumption is wider than usually. It seems like some people prefer to take more shower in the cold day, while others not. All details are shown in appendix.

To get more information from the anomaly detection, I will analysis the time distribution of anomaly data points hourly, monthly and weekday.

- Cluster 0

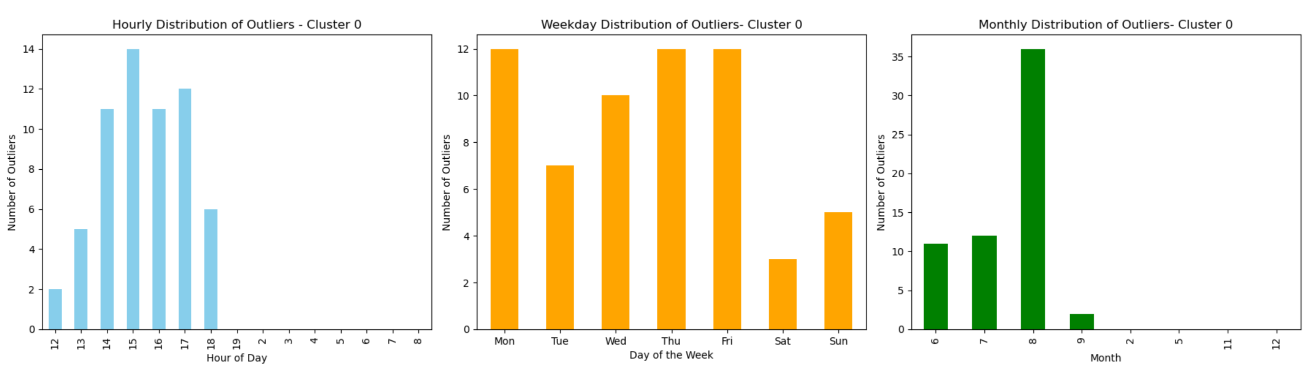


Figure 4. Cluster 0 distribution

It is easy to understand the pattern of energy consumption of cluster 0 time slots with this figure. The time slots of cluster 0 is in summer days' afternoon, from 12 PM to 6 PM. That is the reason of high electricity consumption of cooling and AC. From the perspective of day of the week, it is more concentrated on weekdays. Residents may hang out in weekends afternoon, thus less time slots on weekends.

- Cluster 1

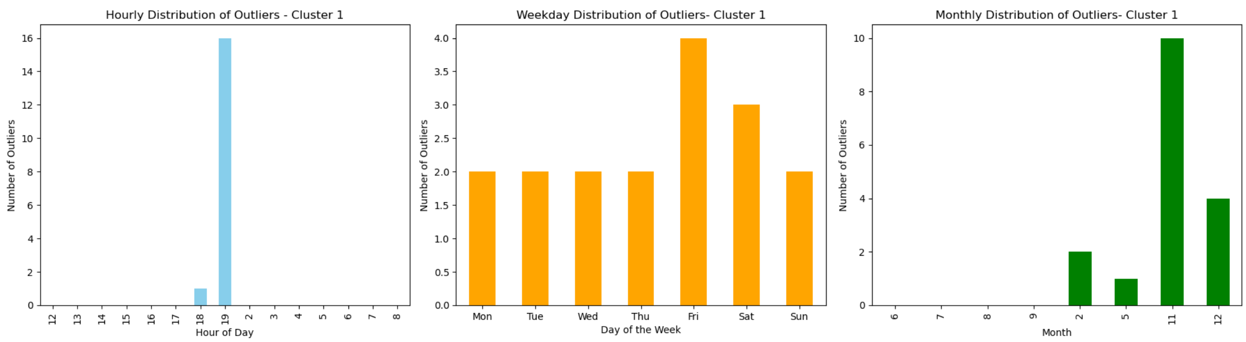


Figure 5. Cluster 1 distribution before drop

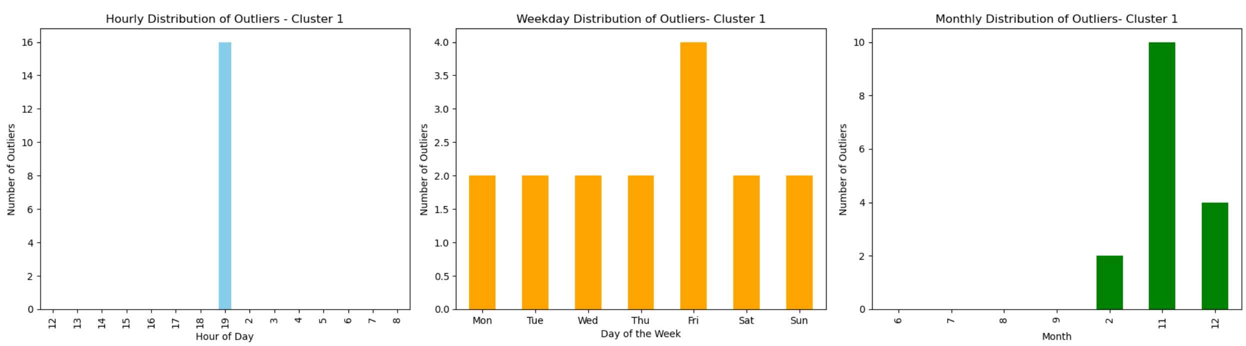


Figure 6. Cluster 1 distribution after drop

After checking, the May data point is an outlier among this group of outliers. Thus figure 6 is the right distribution. The time slots of cluster 1 is on winter night, November, December, and February at 7 PM. The distribution of day of the week is normal, just a few more days on Friday. That is the reason the gas consumption for heating is much more than usual, and the same reason for more electricity for interior, exterior lighting, and interior equipment.

- Cluster 2

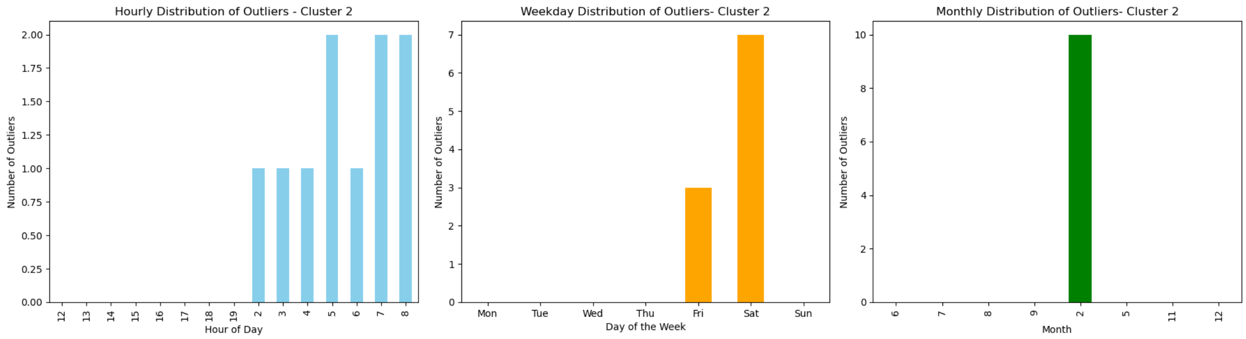


Figure 7. Cluster 2 distribution after drop

The time slots of cluster 2 concentrated in two days, which is on Feb 4 and 5 from 2 AM to 8 AM, with rarely high consumption of gas for heating and electricity for HVAC. It seems people have a large party at that night. I computed the average energy consumption of normal data points in same month and in same period, which is the mean value of consumption everyday 2 AM to 8 AM in February.

Table 2. Mean value of Clusters & Same period normal data points

Features[kW](Hourly)	Cluster 2	Same Period
Electricity:Facility	1.4013	0.9128
Gas:Facility	12.0558	4.0710
Heating:Electricity	0.0000	0.0000
Heating:Gas	12.0277	4.0444
Cooling:Electricity	0.0000	0.0000
HVACFan:Fans:Electricity	0.2950	0.1022
Electricity:HVAC	0.2950	0.1022
Fans:Electricity	0.2950	0.1022
General:InteriorLights:Electricity	0.1183	0.0787
General:ExteriorLights:Electricity	0.0255	0.0170
Appl:InteriorEquipment:Electricity	0.1141	0.0763
Misc:InteriorEquipment:Electricity	0.4086	0.3853
Water Heater:WaterSystems:Electricity	0.4202	0.2367

Obviously, the energy consumption of other normal days in February early morning is totally different than the data of cluster 2. Thus, I guessed either there was a big party, the residents' heating malfunctioned, or it was extremely cold weather during those two days.

2. Appliance’s electricity consumption

Since the electricity consumption of appliance doesn’t follow the gaussian distribution, and there is cyclical fluctuation in the time series data. Gaussian distribution is not an appropriate way for anomaly detection. Given the cyclical fluctuation impact, seasonal and trend decomposition using Loess (STL) was conducted for anomaly detection. This tool can decomposition time series into three main components, trend, seasonal, and residual. The first step is to estimate the seasonal component, and minus it from the time series. Then LOESS would be used for trend component regression on the processed time series.

Finally residual can be get from the original time series data minus seasonal and trend components. The residual we get has no longer cyclical fluctuation, which means it can be used for anomaly detection.

The STL has two parameters, the first one is seasonal, which controls the smoothness of the seasonal components in the STL. The other one is period, which defines the periodicity of the time series. I used grid research to find the optimal parameters to minimize the residual variance, which can improve the model performance of capturing seasonal and trend components. The threshold of anomaly detection is set as two times residual standard deviation, if the absolute value of residual is larger than two times residual standard deviation, this data point will be defined as anomaly data.

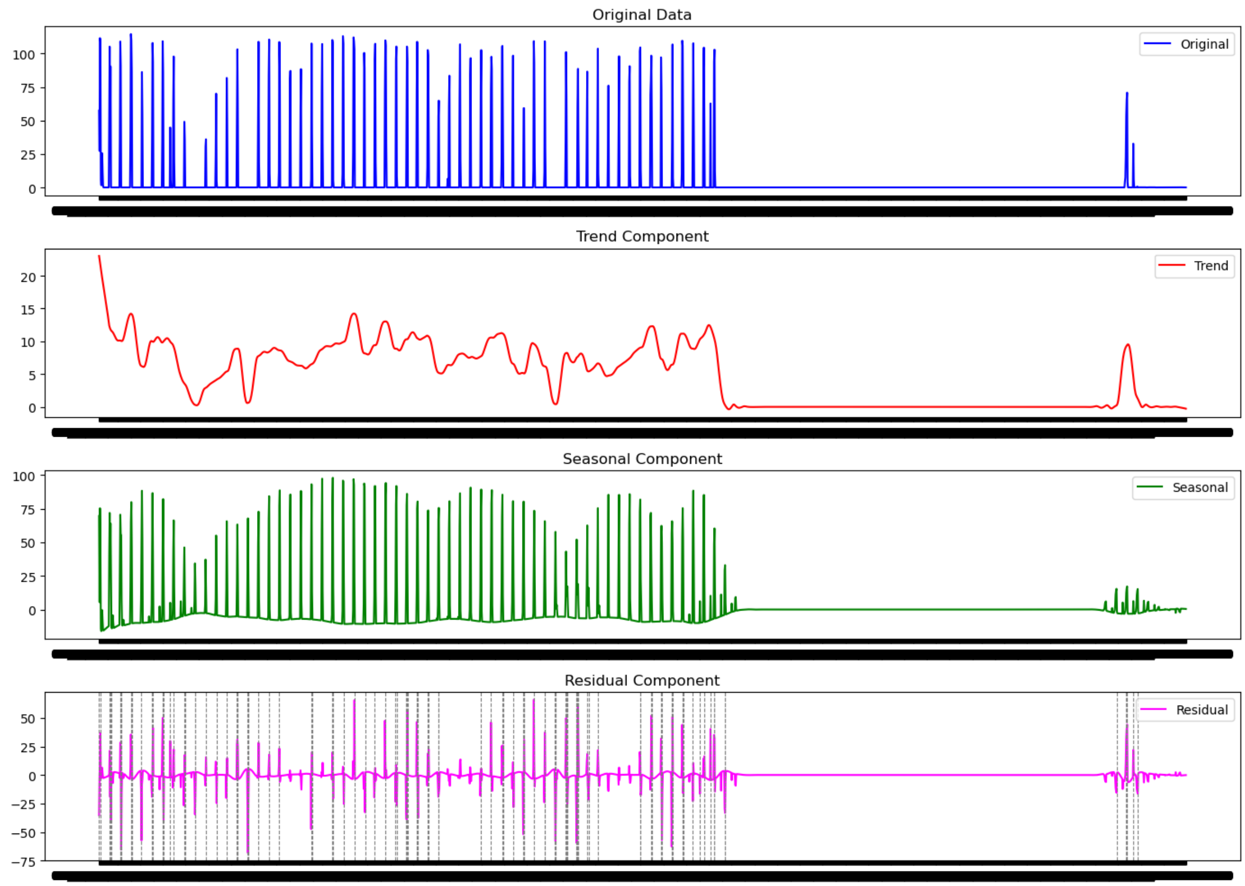


Figure 8. Result of STL

Figure 8 can effectively visualize the whole process of STL, residual is computed from the original data minus trend and seasonal components. In the subfigure of the residual component, the data, which is accompanying the dotted line, is the anomalies. There are 112 anomalies are detected by STL in total, which means these 112 data points' residuals are larger than two times the residual standard deviation.

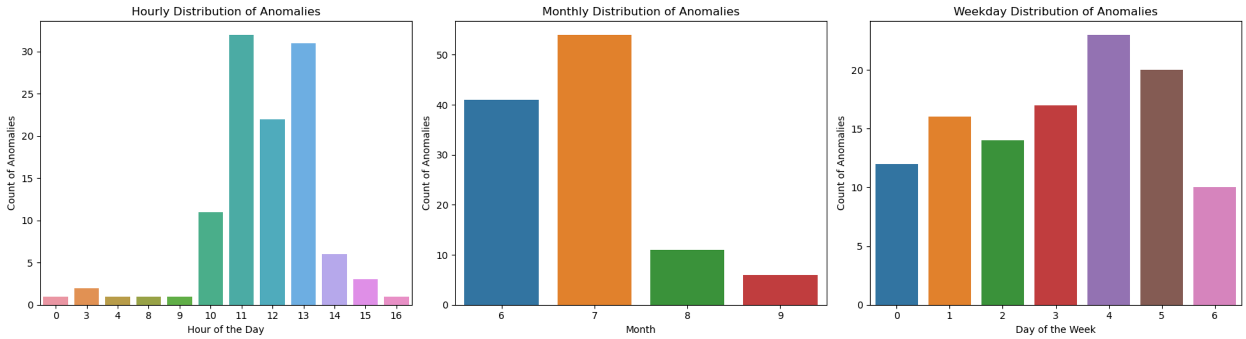


Figure 9. Anomalies happened times distribution

Anomalies is still quite concentrated at midday in June and July, from 10 AM to 1 PM. From the perspective of the week anomalies are more likely to occur on Fridays and Saturdays, with the lowest frequency occurring on Sundays. To capture the details of anomalies, anomalies need to be compared with normal data points, to determine whether it has higher consumption or lower consumption.

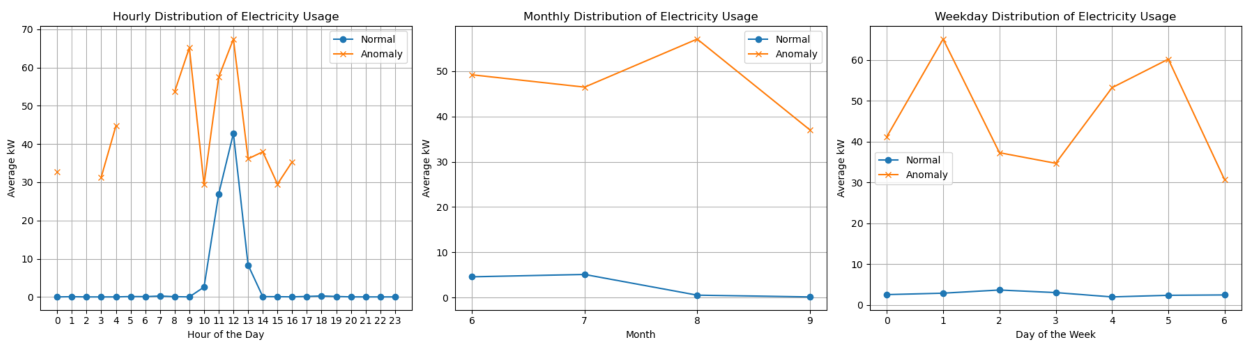


Figure 10. Anomalies & Normal data points

The hourly distribution of electricity usage is comparing the mean value of anomalies consumption with mean value of normal consumption in same hour but at different day. It is same logic for monthly comparison and weekday comparison. All anomalies have far more electricity consumption than the normal time. The anomalies at 9 AM is the most significant difference we should pay attention, there is few electricity usages in normal distribution, but the anomaly consumption is about 65 kW, which is almost close to the peak consumption at 12 PM. Combined with the anomaly frequency, August is a severe problem too, it has highest mean electricity consumption, but with very low frequency.

Section 3: EDA and Forecast Model

The objective of question 3 is to create EDA and forecast model to predict RTLMP.

1. EDA

Before EDA, an overall understanding of data is necessary. I noticed the mean value of RTLMP is 25.7664, but the maximum is 2809.3575, and the minimum is even less than 0, which is -17.86. Thus, the first step is to check the data authority. However, I noticed that all data are correct after I searched the market information, these outliers would be a big problem for later forecast model. Besides, some null values were found in data processing. I noticed those null values is because the daylight-saving set, since the datetimes are at the start and end date of DST. There are two other rows with null values, which are the last two columns, I just deleted them directly.

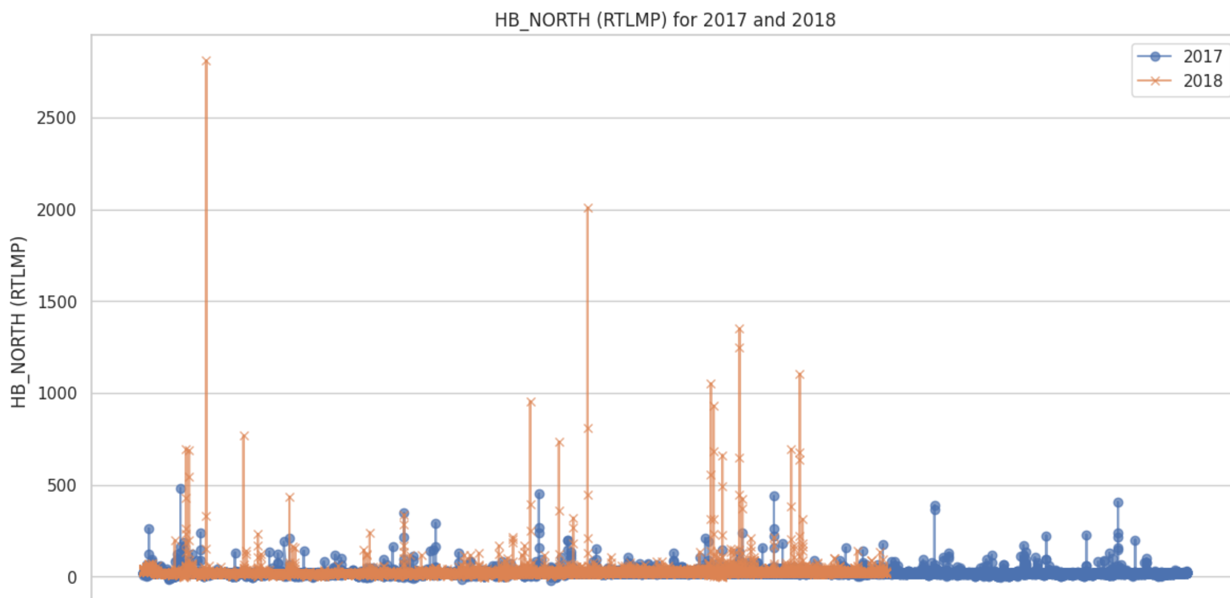
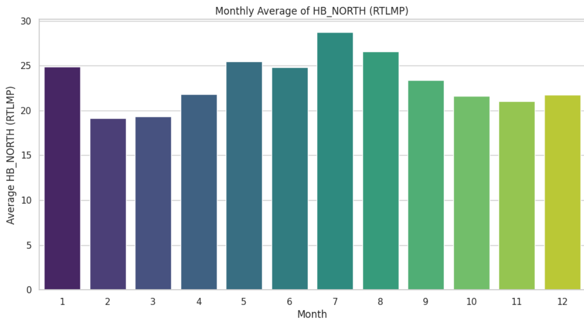


Figure 10. RTLMP for 2017 and 2018

This figure is difficult to read the details, but it shown a significant information that there are more severe outliers in 2018. The more intense the market fluctuations, the more risks, and opportunities there are.



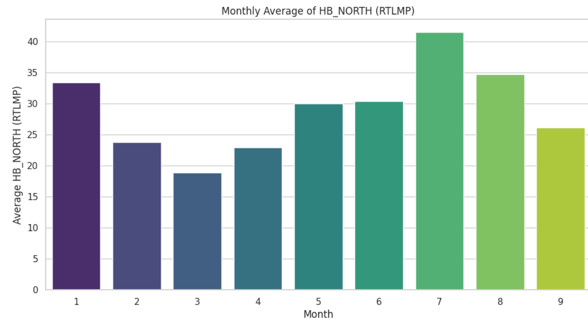
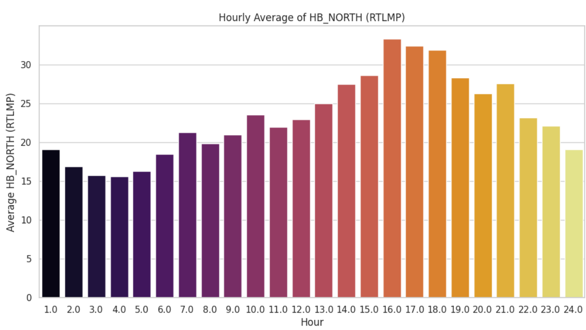


Figure 11. Monthly Average RTLMP for 2017 Figure 12. Monthly Average RTLMP for 2018

RTLMP were divided into two separate years monthly, they share some trends. They all have a decreasing trend till the march, then increase to the peak in July, and decrease again. This price trend should be impacted by the electricity demand in summer and winter. However, the average electricity price in 2018 is higher than the price in 2017. From the year perspective, electricity continues to be in short supply, thus the price is keep increasing from 2017 to 2018.



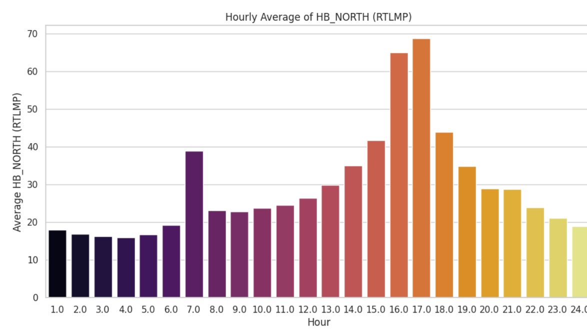
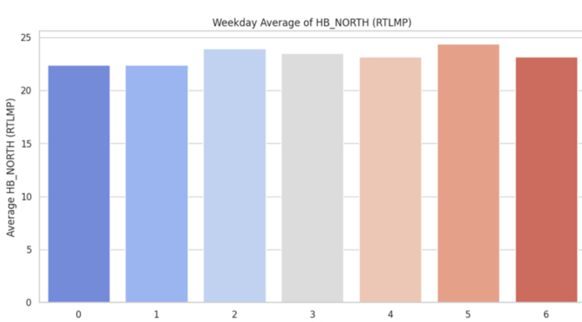


Figure 13. Hourly Average RTLMP for 2017 Figure 14. Hourly Average RTLMP for 2018

From the hourly perspective, RTLMP in two years still share same trends. RTLMP kept increasing till 4 PM and reached the peak. But the RTLMP variance in 2018 is much larger than the variance in 2017. When the price reached the peak at 4 PM, RTLMP in 2018 is around 70, which is larger than two times RTLMP in 2017, the curve is very sharp.



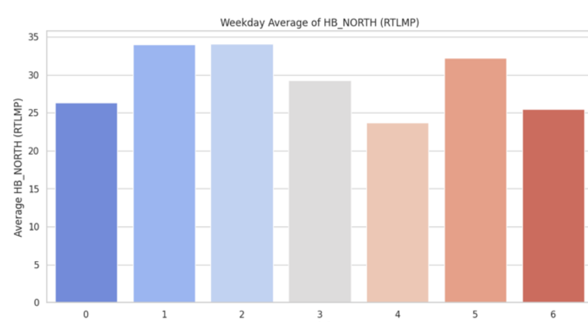


Figure 14. Weekday Average RTLMP for 2017 Figure 15. Weekday Average RTLMP for 2018

The distribution of RTLMP in 2017 is even, but the distribution in 2018 shows the variance difference. The RTLMP has the highest average price on Tuesday, Wednesday, and Saturday, while the lowest average price is on Friday. Even though the price on Friday is lowest in 2018, it is almost close to the average highest price in 2017 on Saturday.

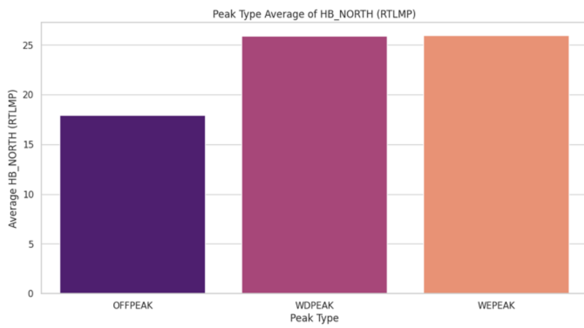


Figure 16. Peak type Average RTLMP for 2017

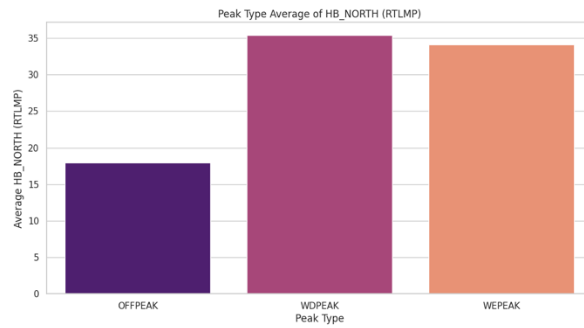


Figure 17. Peak type Average RTLMP for 2018

The prices of different types in 2017 are also more even than 2018. The price difference between on peak and off-peak periods is more obvious in 2018, which brings more arbitrage space for traders.

2. Real time forecasting with LSTM

From the EDA, it can be noticed that there are several problems we need to be concerned. RTLMP has cyclical fluctuation which is because of the demand fluctuation from spring to winter. Besides, there are a few outliers, which is far away from the mean value. To solve these two problems simultaneously, I combined SARIMA model and LSTM together for real time price forecasting. SARIMA was conducted for the seasonal trend, then the residual of SARIMA would be the input of LSTM with other numerical features. LSTM was conducted to predict more precise residual. Finally, the sum of outputs of SARIMA and LSTM would be the predict RTLMP. However, the integration of these two models didn't improve the model performance, R-square is even less than 0.1. I noticed that even LSTM could capture the precise relationship between the residual and time series, SARIMA cannot estimate the appropriate seasonal trend. I decided to use simple LSTM directly.

The first step of LSTM is to standardize all the features to avoid the impact of excessive differences in data magnitude, which can also improve the model performance. Considering the timestamps between target value and features, lag was introduced into LSTM model. Lag means how long will the target value take to react if the features changed. Besides, peak type as a string feature should also be consider, thus I used one-hot encoding for the peak type.

The LSTM framework was developed using Python and PyTorch libraries and was run on Google Colab's T4 GPU with 16 Gigabyte memory, which provided enough computational resources to manage the data preprocessing and run model training.

The training process was divided into two sequential stages. The first stage involved finding optimal lag through cross validation, the loss function here is Mean Square Error (MSE), early stop was also introduced into the training process, which can improve the model performance and efficiency. The optimal lag from cross validation is 3, with minimum MSE 0.7543. With the optimal lag value 3, the second stage is to train the best model and evaluation. The LSTM was trained for 46 epochs, using the Adam optimizer with learning rates ranging from 0.0001 to 0.001. This phase focused on accurately predicting real time price.

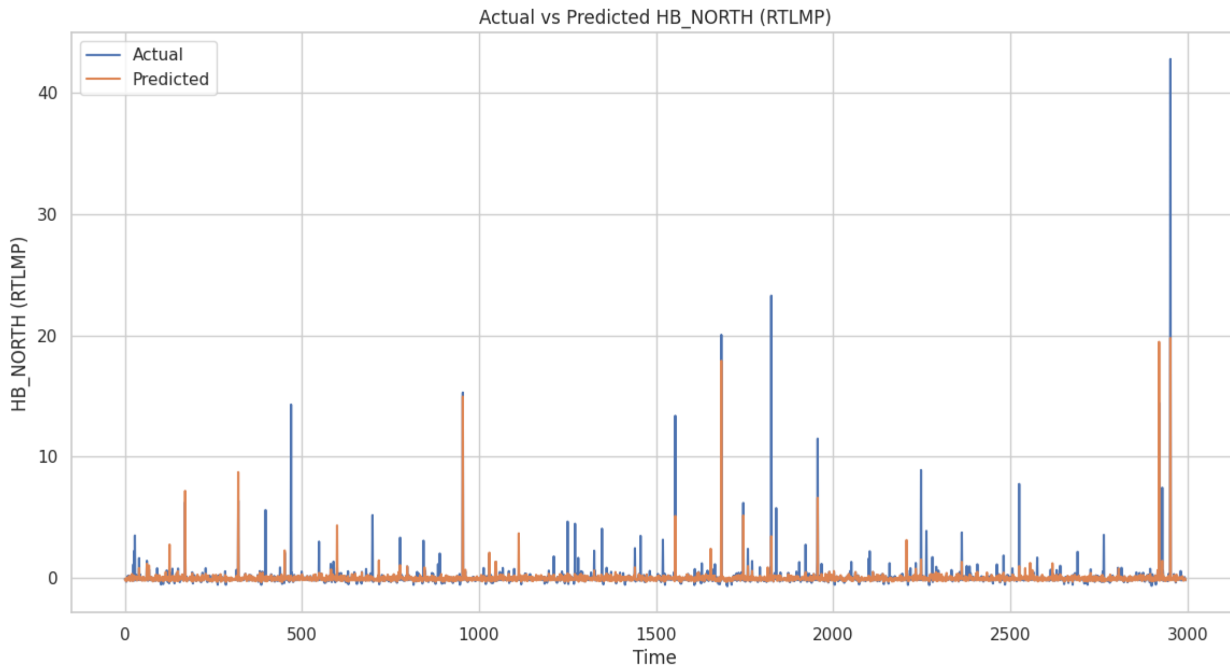


Figure 18. Actual RTLMP and predict RTLMP

The results from this rigorous training process demonstrated the robustness and accuracy of the framework. The LSTM achieved a total loss of **0.63**, indicating its capability to effectively forecast the real time price. **R-square** of the regression is **0.6079**, which means LSTM can explain 60.79% of the variability in the data. This R-square suggest a moderate quality of fit, while it is not perfect. It can be noticed from the figure. LSTM didn't perform well at fitting the outliers at some points. To improve the model performance, it is important to smoothen the target to reduce the outlier's impact. The smoothing method I tried is to use log transformation.

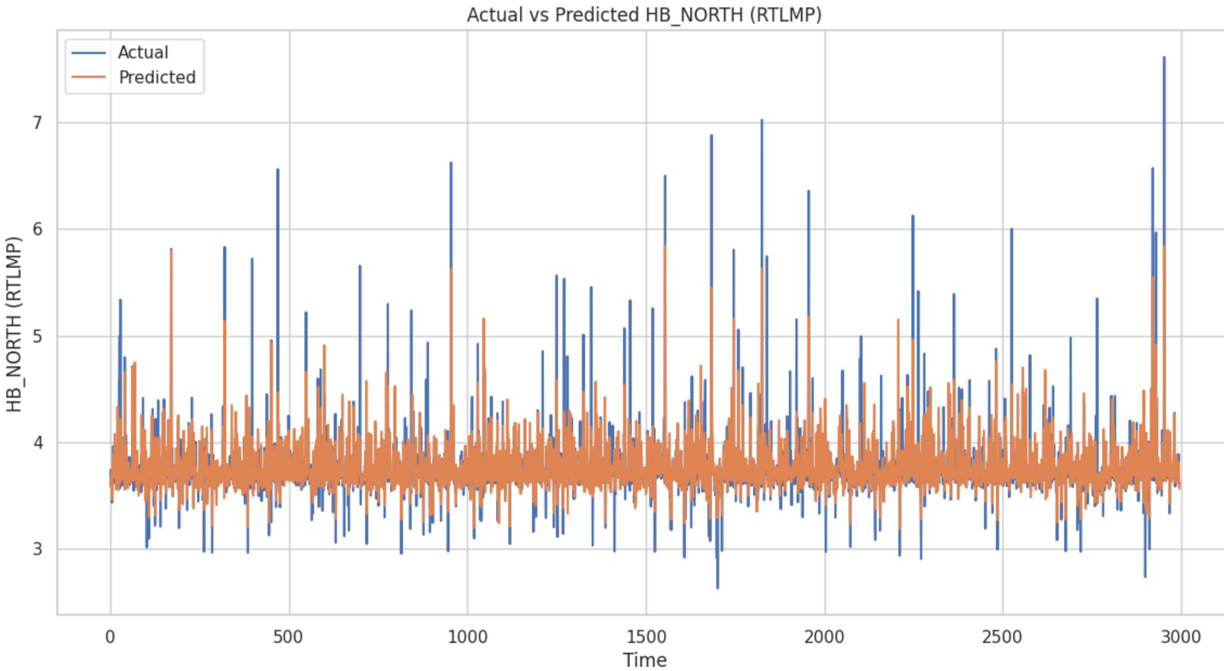


Figure 19. Log transformation of Actual RTLMP and predict RTLMP

With the log transformation, R-square was raised to **0.6754**, which is a little better than the normal LSTM. However, LSTM still cannot predict the outliers precisely and effectively. But it is acceptable, since with such limited features, LSTM has no way of making excellent predications that are reasonable. Those occurring of outliers should be influenced by other features, like market information, extreme weather, or accidents, while no such features are provided to LSTM.

Section 4: Learn products of Futures

1. Future

A type of derivative contract agreement to buy or sell a specific commodity asset or security at a set future date for a set price.

2. Electricity futures

Electricity futures is a vital component of the modern electricity market, offer unique trading methods and risk management mechanisms, providing effective market tools for various stage of electricity industry chain. In the electricity industry chain, the stages of generation, transmission, distribution, and retail are closely interconnected, collectively forming the foundation of the electricity market.

Generation Stage: The generation stage is the starting point of the electricity industry chain and includes various forms of power generation such as thermal power, hydroelectric power, nuclear power, and wind power. Due to the complex and dynamic supply-demand relationships in the electricity market, electricity prices are often highly volatile. The electricity futures market provides power generators with an effective risk management tool. By buying and selling electricity futures contracts, power generators can lock in future electricity prices, thereby **reducing the risks** associated with market volatility.

Transmission Stage: The transmission stage is primarily responsible for transporting the electricity produced by power plants to different regions. Significant investment in equipment and technology is crucial at this stage for the stable operation of the electricity market. The existence of the electricity futures market helps transmission companies **better predict future electricity demand**, allowing them to plan the construction and operation of transmission infrastructure more rationally.

Distribution Stage: In the distribution stage, electricity transmitted through the transmission stage is further distributed to various customers. Distribution companies need to adjust the electricity supply flexibly according to the diverse needs of different customers. The electricity futures market offers a platform for **price discovery** for distribution companies, enabling them to grasp market dynamics and devise reasonable electricity distribution plans more accurately.

Retail Stage: The retail stage is the final stage of the electricity industry chain, delivering electricity to end users. The electricity futures market provides retail companies with opportunities to **hedge market risks**. By purchasing electricity futures contracts, retail companies can lock in future electricity procurement costs, ensuring the stable operation of their businesses.

3. Product 1: Power Futures – ERN

A **monthly cash settled** Exchange Futures Contract based upon the mathematical average of daily prices calculated by averaging the peak hourly electricity prices published by ERCOT for the location specified in Reference Price A.

Trading Screen Product Name: Peak Futures (1 MW)

- Used for traders identify and choose specific products

Trading Screen Hub Name: ERCOT North 345KV Hub RT

- In electricity market, hub refers to a geographical location or a central point in an electric power transmission network. The electricity trades here are standardized and published.

- Different hubs have different prices, which reflects factors such as supply and demand in the region, transmission constraints, and more.

Settlement Method: Cash settlement

- The other method: physical delivery, for different purpose.

Contract Size: 1 MW

- Contract Size refers to the quantity of the underlying asset represented by each future contract.

Minimum Price Fluctuation: One cent per MWh

- Minimum increasement of the price

Listing Cycle: Up to 50 consecutive monthly Contract Periods

- Listing Cycle means the contract cycle that is listed and available for trading on the exchange.
- Consecutive monthly Contracts means that the exchange offers contracts for trading that cover up to 50 months in a row. Example: Now is May 2024, contracts can be listed for trading from now up to June 2028. It is beneficial for liquidity, market depth, and price discovery.

Last Trading Day: The last Business Day of the Contract Period

- No more trading after the last trading day. Settlement begins.

Reference Price A: ELECTRICITY-ERCOT-NORTH 345KV HUB-REAL TIME

- The price for a Pricing Date will be that day's Specified Price per MWh of electricity for delivery on the Delivery Date.
- Pricing date: Monday to Friday, excluding NERC holidays.
- Specified price: Average of SPPs for all hours ending 7 AM to 10 PM CPT from ERCOT. SSPs (System Price Points)
- Deliver date: Every day in contract period, and the deliver price is the specified price of the pricing day.

Final Payment Date: The sixth Clearing Organization business day following the Last Trading Day

- 6 days for clearing.

MIC Code: IFED

- Market identifier code is a unique identifier code for identifying exchange.
- IFED (Intercontinental Exchange Futures U.S.)

Clearing Venues: ICEU

- Clearing Venues is the institution for clearing transactions
- ICEU means ICE Clear Europe, the clearing arm of ICE that clears futures and options trades made on its platform.

4. Product 2: Natural Gas Futures -H

A **monthly cash settled** Exchange Futures Contract based upon the monthly price published by NYMEX for the location specified in Reference Price A.

Listing Cycle: Up to 156 consecutive monthly Contract Periods

- It is different from the Power future -ERN, since natural gas markets generally have longer investment and hedging needs. Natural gas supply chain and infrastructure projects (such as pipeline construction, LNG facilities, etc.) involve long-term planning and significant capital investment, so market participants may require longer-term hedging instruments.

- Electricity markets typically have short investment cycles. Power trading is mainly concentrated in the short term and is greatly affected by seasonal demand, weather conditions and changes in market supply and demand, so market participants may focus more on short-term hedging and trading.

Reference Price A: NATURAL GAS-NYMEX

- The price for a pricing date will be the specified price for the delivery date. Same as power future.
- Pricing date is the last scheduled trading day. Different from power future.

Final Payment Date: The first Clearing Organization business day following the Last Trading Day

- Only one day

Markers: TAS

- Markers means specific price or time, which is used for marking the trade happening.
- TAS is a special trading instruction, allows people to buy or sell at the settlement price

5. **Product 3: Heat Rate Futures – XPR**

Firm Energy with Liquidated Damages. Physical ERCOT Monthly Heat Rate Power delivered at a specified ERCOT location.

- Firm Energy is a guaranteed supply electricity contract, meaning the seller promises to deliver a fixed amount of electricity for a specific period. Such contracts are often used to ensure power supply to key customers or important facilities.
- Liquidated Damages, if the seller fails to fulfill its obligation to provide fixed electricity, liquidated damages will be paid in pre-agreed amount.
- Physical means actual delivery of electricity rather than through cash settlement.
- Heat Rate is a measure of the efficiency of electricity generation, usually expressed as the fuel energy required per MWh (megawatt hour) of electricity in unites of MMBTU (Millions of British thermal units).
- Specified ERCOT location indicates specific power delivery location.

Contract Size: 50 MW per hour within 0700-2200
Currency: US$ and cents per MWH
Last Trading Day: The second to last business day of the month prior to the delivery month
Daily Settlement: Daily settlement will be determined by NGX (Natural Gas Exchange) using price data from spot, forward, and derivative markets for both physical and financial products. This settlement is like an estimation, help traders to calculate every day PnL.
Final Settlement: Actual heat rate times the NYMEX last day settlement price.
Payment Dates: Payment due on the 20th of the month following the Delivery Period by wire transfer of Federal Funds, unless otherwise stated by a master agreement between the Buyer and Seller. Example: Deliver date is in May 2024, payment date is on June 20th, 2024.

This future is a little bit abstract. This future is to combine the power and gas together for hedge.
Heat Rate = required fuel (MMBtu) / electricity (MWh)
Currency is ($/MWh): price of per MWh
The Contract Size is 50 MW
The roadmap should be:

a. Determine heat rate. Assume the heat rate is 8 MMBtu/MWh

b. Get the fuel price. Assume the price in delivery date is $3/MMBtu
c. Calculate the electricity price = Heat Rate * Fuel Price = $24 /MWH
d. Calculate the total cost per hour = Price * Contract Size= $24/MWh * 50MW = $1200/h

Question 2: Assume Product 1 has no liquidation in the market and we are holding the physical power (same settlement as Product 1), how to use Product 2 & 3 to hedge our exposure to physical power (again, same settlement as Product 1)?

Assumptions:

1. We are holding 1000 MWH of physical power.
2. Heat Rate is 10 MMBtus/ MWH
3. There are 20 trading dates in a month, which means there are 20*15=300 hours on peak

Action:

1. Long 0.0667 contracts of Product 3.
2. Short 4 contracts of Product 2.

Since we are holding 1000 MWH of physical power, which means my market exposure is long 1000 MWH. Thus, I need to short 1000 MWH to hedge the exposure. Given the assumptions, there are 20 trading dates in a month, the contract size of XPR (Product 3) is

$$50\ MW \times 300h = 15000\ MWH$$

The contracts of XPR that I need to short is

$$\frac{1000\ MWH}{15000MWH} \approx 0.0667$$

Since the price of XPR is composed of Heat Rate and Gas price (Product 2), we also need to long project 2 to hedge the exposure of gas price which is from XPR. The volume of gas we need to long is

$$1000\ MWH \times 10\ MMBtus/MWH = 10000\ MMBtus$$

Given the contract size of product 2, we need to long is

$$\frac{10000\ MMBtus}{2500\ MMBtus} = 4$$

By these two steps, we can totally hedge the 1000 MWH of physical power.

Question 3: Create Excel file model with weekly rebalance of your positions (only rebalance Product 2) to try to achieve hedging. within the Excel file, use parameter to decide your rebalance and summarize the efficiency of hedging.

In this question, I set two parameters, Rebalance Ratio as 0.1, and initial position as 100. Assume we are holding 100 contracts H, and it need to be rebalanced with the price fluctuation.

1. **Weekly price change** was calculated by the prices of Mondays. The price difference between weeks equals to this Monday price minus last Monday price.
2. Depended on the rebalance ratio we set at 0.1, the **target position** is

$$Original\ Position - (Rebalance\ Ratio * Weekly\ Price\ Change)$$

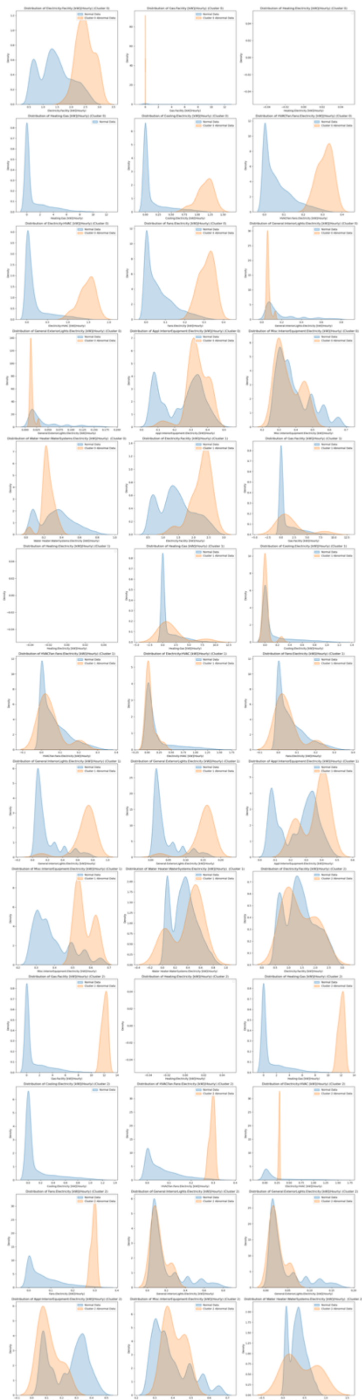
If the price raise, thus we need less contracts; While if the price decrease, we need more position for hedging.

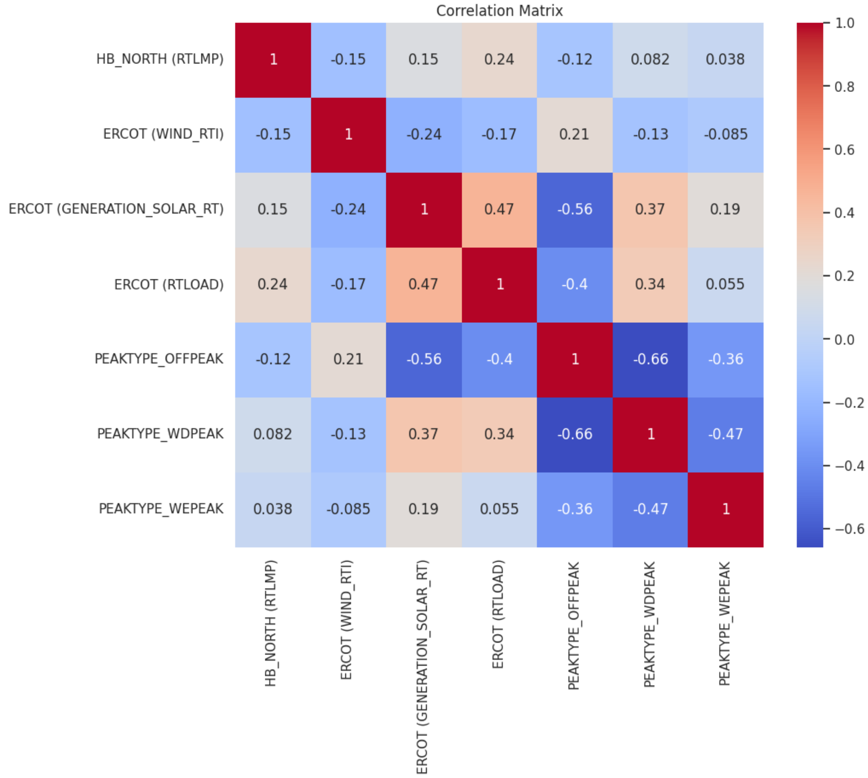
3. **Position Value** is calculated by multiplying the real time price and the target position.
4. Calculate **Hedging efficiency**, the formula is

$$1 - \left|\frac{Actual\ Value - Target\ Value}{Target\ Value}\right|$$

The more this metric is closer to 1, the better hedging is.

Appendix




Correlation Matrix
HB_NORTH (RTLMP)
ERCOT (WIND_RTI)
ERCOT (GENERATION_SOLAR_RT)
ERCOT (RTLOAD)
PEAKTYPE_OFFPEAK
PEAKTYPE_WDPEAK
PEAKTYPE_WEPEAK
1 -0.15 0.15 0.24 -0.12 0.082 0.038
-0.15 1 -0.24 -0.17 0.21 -0.13 -0.085
0.15 -0.24 1 0.47 -0.56 0.37 0.19
0.24 -0.17 0.47 1 -0.4 0.34 0.055
-0.12 0.21 -0.56 -0.4 1 -0.66 -0.36
0.082 -0.13 0.37 0.34 -0.66 1 -0.47
0.038 -0.085 0.19 0.055 -0.36 -0.47 1
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6