Forecasting Electricity Demand and Prices in PJM and MISO Markets

Machine Learning

A Data-Driven Approach for Efficient Utility Management

Group 5

Team members :

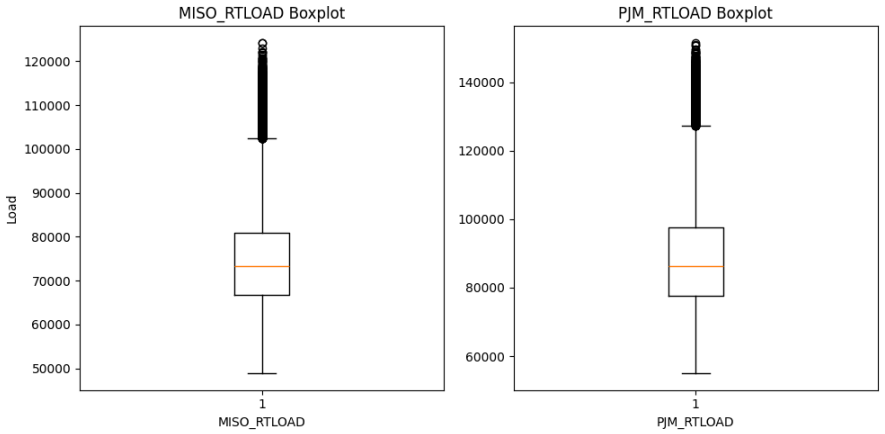
Renee Chang - 1009640051
Ke Han Bei - 1009338236
Zhanglin Liu - 1009315974
Miaoyan Qi - 1008786388
Ba Minh Dang, Le - 1006136272
Yian Yan - 1009103239

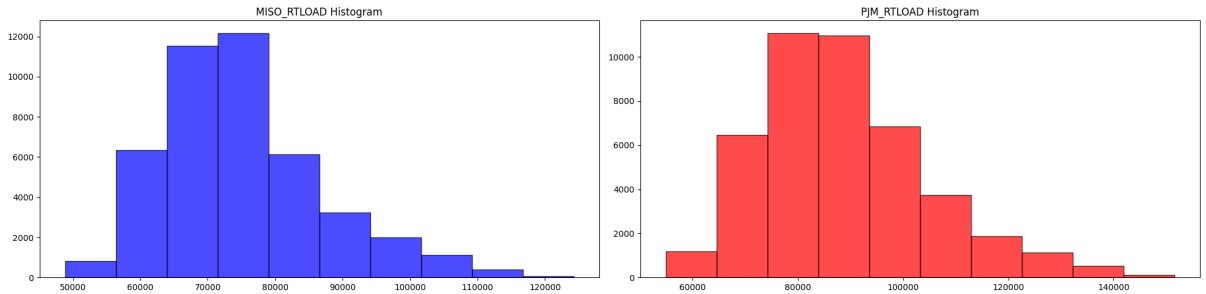
Classification - Predict Real Time Load (Low, Medium and High Demand loads)
Define the Classification Problem
As mentioned during the proposal, one objective of this project is to help forecast electricity demand in the MISO and PJM grid. To tackle this problem, we decided to implement a Decision tree classification model to solve this problem.

The intuition is that we will split each grid electricity Real Time load into three different demand categories/levels: Low, Medium and High Demand loads. Then two Decision Tree models will be fitted with a collection of features that can help the model to best predict what is the level of electricty demand of each grid.

The target variables are: MISO_RTLOAD (later will be encoded as MISO_Load_Category_encoded) and PJM_RTLOAD (later will be encoded as PJM_Load_Category encoded)

Data Summarization
Before preparing the data for the machine learning algorithm, it's helpful to visualize the target variables





Looking at the boxplots and histograms, we can see that both grids show a wide range of loads, indicating variability in electricity consumption. Furthermore, PJM experienced a higher median electricity load suggesting larger overall demand.

The presence of outliers in both box plots indicates sporadic spikes in electricity load and its clear that the electricity load in both markets share similar distribution

Data Preparation
There are 5 steps in total for the data preparation process: Binning the Target Variables, Data Encoding, Feature Engineering, Feature Selection Exploring and Test/Train Split

Feature Engineering
To improve the model performance and make sure that we are considering all possible subset of features, we transformed the original data into various variables as potential categorical features. The additional features added are: hour_category, season, ON_OFF peaks and days of the week

Binning The Target Variables
Due to the lack of available information on low, medium and high electricity demand threshold for PJM and MISO markets. We decide to categorize the target variables by 3 quartiles (33.3% quartile). If real time electricity load falls in 1st quartile then its low demand and so on. In summary, the demand levels are classified as low (bottom 33.3%), medium (middle 33.3%) and high (top 33.3%).

Data Encoding
The Data Encoding process will quantify categorical variables. Both ordinary and hot encoding are used during this process. The electricity load has a hierarchy relationship hence is most fitted with ordinary econding where Low Demand is encoded as 1 and then increases incrementally to High Demand as 3. Season on the other hand has no hierarchy relationship therefore we used hot-encoding to quantify this categorical variable.

Test/Train Splitting
The next step in data preparation is to train test split our data, where we split the dataset into a set to train and a set to test and evaluate. We have used the common ratio of a 80/20 split, where 80% of the data will be used for training while 20% are used for testing. Since only numerical values can be used, the columns with float types are used in our array of factors, filtering out the rest. The x_arrays contains all possible features and y_arrays contains the encoded electricity demand (MISO_Load_Category_ecnoced and PJM_Load_Catergory_encoded

Decision Tree Model
As mentioned above, Decision Tree is the chosen machine learning to classify real time electricity demand. The Decision tree learning uses a decision tree as its predictive model to go from observation about a particular features (represented as branches) to conclusion about the target value (represented as leaves)

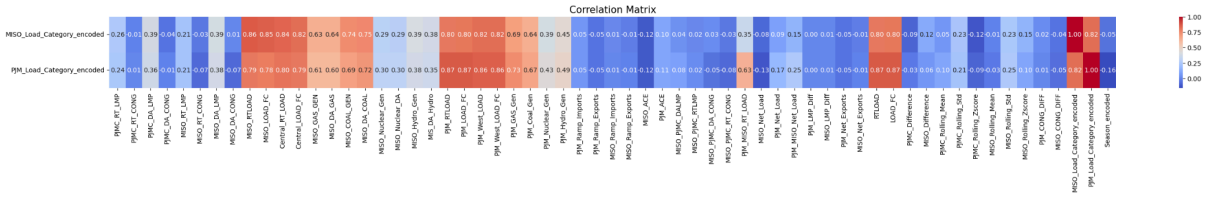
The feature selection process contain the consideration of two type of features: continuous and categorical features

For categorical features, we look at the frequency tables of our_category, season, ON_OFF peaks and days of the week. We decided to select **season** as **categorical features** since its reflect the season **fluctuation demands** the **most**

Season and Demand Frequency Table

Season	# Low Demand	# Medium Demand	#High Demand
Fall	4938	4108	1878
Spring	6163	3927	970
Summer	2244	2510	6286
Winter	1279	4078	5490

To select continuous features, the intuitive is that highly correlated features with the target variables are more likely to improve the model performance and the potential subset of these can be generated from the correlation matrix



Looking at the correlation, the best performed models should include at least one of highly correlated variables (orange to red color coded)

However, this approach won't consider the effects of potential interaction of features, this might lead to a case where a combination of 'insignificant features' ended up to be significant to our.

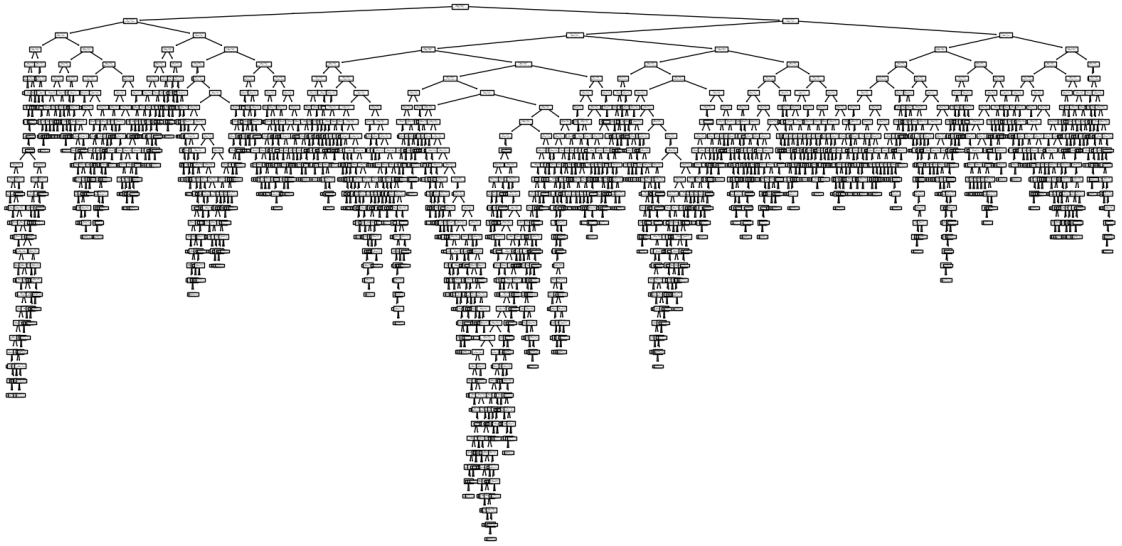
To ensure that all subset of features are considered. We started with fitting them with all possible features (see correlation matrix vertical rows) along with the hot encoded season categorical features. Despite the fact that this preliminary model will be overfitted with too many features a two step parameter tuning process will be implement to solve this issuing include:

Step 1: Finding optimal parameters setup for the decision tree (Branches and Leafs) using **Gridsearch techniques** (Improving over fitting)

Step 2: Enhancing Feature Selections using **Recursive Feature Elimination Technique** (This will improve model accuracy by selecting the best subset of features that contributes the most to the model performance)

MISO Pre and Post-tuning Models

The pre tuned model has a **0.9231** or **92.31 accuracy score** with training set which is good given the dataset. However as mentioned above this model considers all possible features subset (there will be many insignificant features) which causes it to be overfitted and poorly performed with unseen data.



The **pre-tuned** tree is exceptionally deep and complex, with a large number of **splits (nodes)**, which suggests that the model has likely learned not just the **underlying patterns** in the data but also the **noise**. This means that the model will perform poorly on unseen data due to high variances.

After the first step, the model is now a lot less complicated, the GridsearchCV methods concluded the optimal parameter for tree is:

max_dept = 20	The tree is allowed to make up to 20 decisions (or splits) from root to leaf. This controls the complexity of the model by limiting how detailed the model can get.
min_samples_leaf = 10	This parameter sets the minimum number of samples that must be present in a leaf node.
min_sample split = 2	This parameter specifies the minimum number of samples required to split an internal node.

The tree, while still complex enough to ensure performance, is now more structured with less prone to overfitting than the first overly complex tree. With optimal parameters, the branches are allowed to extend to make detailed decisions while constraining them to learn unnecessary noises. The **training set accuracy score** for the optimal tree is **0.96561 or 96.51%** indicating the improve in performance

With the optimal parameter setup, the model will then undergo optimal feature selection. The RFE method is executed on the training dataset where it fits the model multiple times, each time removing the least important feature as determined in the previous round. This process will find the most optimal subset of features for the model. The optimal set of features including: **Central_RT_LOAD, MISO_DA_GAS, PJM_RT_LOAD and PJM_LOAD_FC**

From the optimal features we can observe the significant relationship between **MISO electricity demand** with the **central region's total real time demand**,**amount of gas supplied electricity** and **PJM electrical loads.**

MISO Final Decision Model Result and Accuracy Evaluation
We now fit the final with optimal features, subset and parameters set up. The score of this model is as follow:

Performance Scores

Model Accuracy	0.94677 or 94.677%
Precision Score	0.94663 or 94.633%
Recall Score	0.94677 or 94.677%

The performance score table represents the overall accuracy of the model, indicating that about 94.67% of the predictions made by the model are correct. This is a high accuracy rate suggesting that the model performs well in classifying the data correctly across all **classes**.

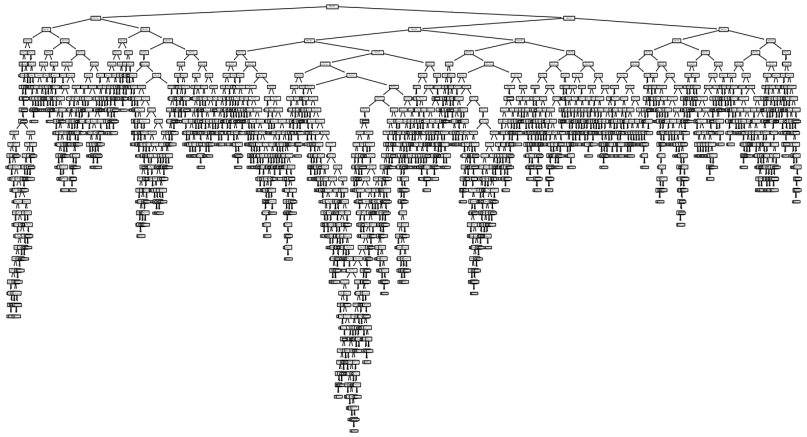
Confusion Matrix

Actual \ Predicted	Low Demand	Medium Demand	High Demand
Low Demand	**2823**	82	0
Medium Demand	141	**2702**	119
High Demand	0	124	**2764**

The model performs very well with high true predictions (blue highlighted) and low false predictions (very high true positive and low false positive rates). There are no instances where the model misclassifies a low demand as high demand loads which further ensure its reliability.

PJM Pre and Post-tuning Model

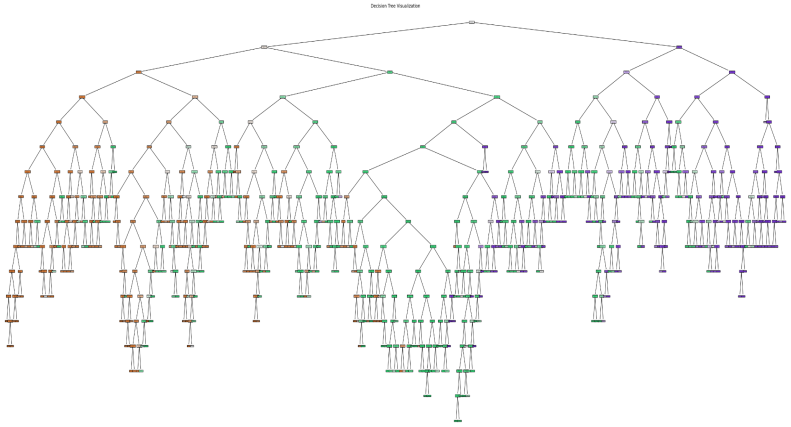
The pre tuned model has a **0.96790** or **96.790% accuracy score** with training set which is good given the dataset. Similar to the MISO case this model considers all possible features subset which caused it to be overfitted and poorly performed with unseen data.



We encounter the same issue as the MISO model where the **pre-tuned** tree is exceptionally deep and complex suggesting that the model has likely learned not just the **underlying patterns** in the data but also the **noise**. This means that the model will perform poorly on unseen data due to high variances.

After the first step, the model is a lot less complicated, the GridsearchCV methods concluded the optimal parameter setup for the optimal tree is:

max_depth: 20	The tree is allowed to make up to 20 decisions (or splits) from root to leaf. This controls the complexity of the model by limiting how detailed the model can get.
min_sample_leaf: 10	This parameter sets the minimum number of samples that must be present in a leaf node.
min_samples_split: 2	This parameter specifies the minimum number of samples required to split an internal node.



The optimal tree is complex enough and structured with having less prone to overfitting than the first overly complex tree. The **training set accuracy score** for the optimal tree is **0.98583 or 98.583%** indicating the improve in performance

Following the same framework as MISO. The RFE method is executed on the training dataset to find the most optimal subset of features for the model. The optimal set of features include: MISO_RTLOAD, PJM_West_LOAD, PJM_GAS_Gen, PJM_Coal_Gen and PJM_MISO_RT_LOAD. From the optimal features we can observe the significant relationship between **PJM electricity demand** with the **West PJM region's real time demand**,**amount of fossil fuelled electricity** and **PJM/MISO difference in real time demand.**

PJM Final Model Result and Accuracy Evaluation

We now fit the final with an optimal subset of features and parameters set up. The score of this model is as follow:

Performance Scores

Model Accuracy	0.97715 or 97.715%
Precision Score	0.97715 or 97.715%
Recall Score	0.97715 or 97.715%

The model score is almost identical across the boards, indicating that the model can predict and classify demand load at a very high accuracy across all demand load

Confusion Matrix

Actual \ Predicted	Low Demand	Medium Demand	High Demand
Low Demand	**2846**	50	0
Medium Demand	60	**2835**	41
High Demand	0	49	**2874**

The model performs very well with high true predictions (blue highlighted) and low false predictions (very high true positive and low false positive rates). There are no instances of extreme false prediction, which further ensures its reliability.

Insights and Summarizing Results

Model Performances:
Both models have high precision and recall scores, suggesting that the models are not only good at identifying all relevant cases (high recall) but also at minimizing the number of **irrelevant cases** that are **incorrectly identified** (high precision). With high accuracy scores, both models are highly effective in predicting the correct demand categories across almost all cases.

Model Reliability:
The confusion matrix for each model provided clear insights into the classification performance across different demand levels. High values on the diagonal confirm that the majority of predictions match the actual values, which is crucial for the reliability of the model in operational settings.

With **Accurate** and **Reliable** predictions, both models can significantly aid utility companies in better grid management, planning, and reducing the risk of blackouts which is the project core objective

Regression - Predict Real-Time Price in PJM and MISO Market

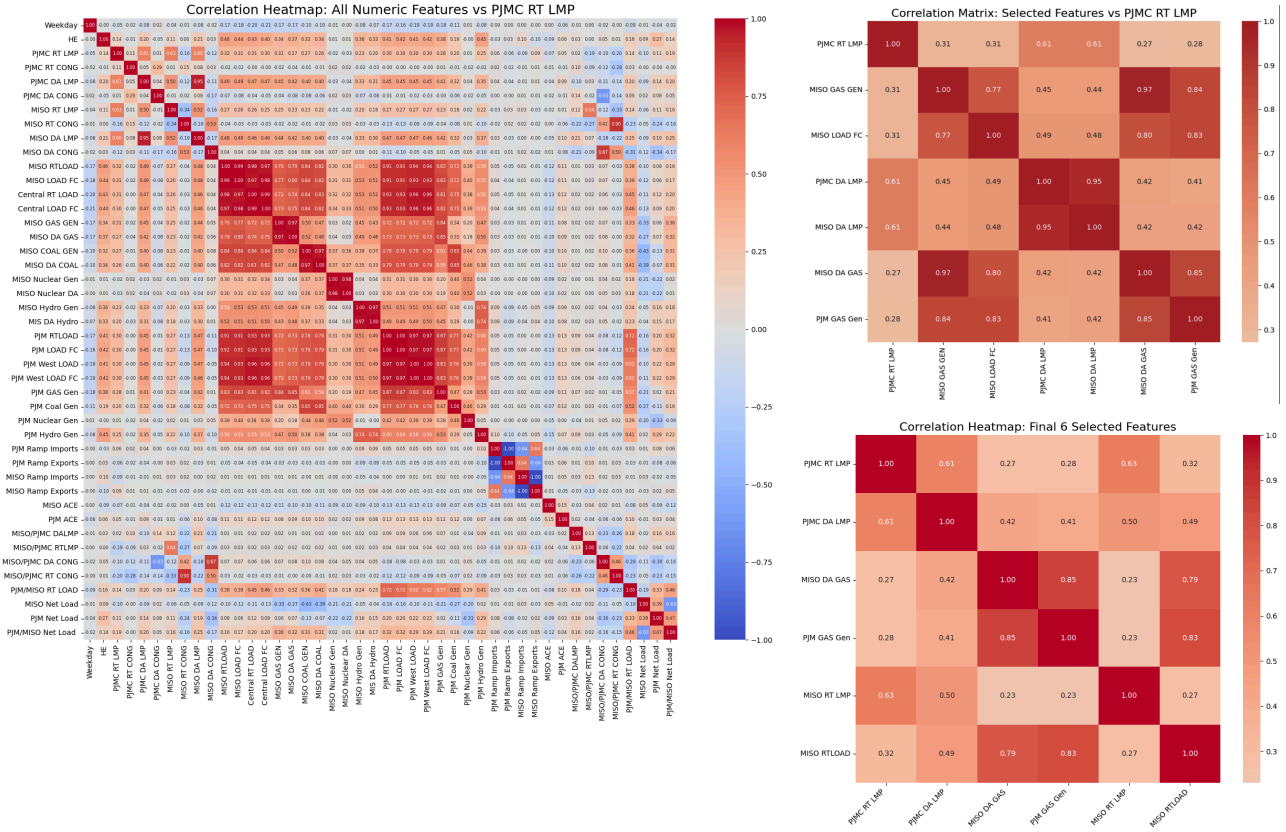
Define Problem:
In recent years, extreme weather events have occurred frequently, and many regions in the United States are facing the risk of power supply shortages or even interruptions due to factors such as electricity peaks and aging infrastructure. In this context, how to accurately predict electricity prices and demand has become an urgent problem to be solved in the power industry. Our goal is to use machine learning algorithms to model and predict electricity prices and demand in the two major electricity markets in the United States - PJM and MISO - based on historical data. Our focus is on the variable of PJM real-time electricity price (PJMC_SRT_LPM). We construct a regression model that can be used for practical prediction by analyzing the relationship between variables such as operating conditions, electricity load, inter-market flows, and hourly time periods. Our main problem is: Is it possible to accurately predict the electricity demand of MISO and PJM markets in the coming period based on data from the past few years? What other factors play a decisive role in the fluctuation of electricity demand?

Data Summarization:
We selected and used multiple variables closely related to the operation status of the electricity market from the original dataset as input features for our regression model. These variables mainly focus on electricity prices, congestion costs, load forecasting, power generation, system stability, and inter-regional power flow.

Data Preparation:
To predict Real-time electricity price in the PJM and MISO markets, we use a combination of scatter plot matrices and correlation heatmaps to select features. By screening the features that fit both moderately correlated with the target variable and minimally correlated with each other to avoid multicollinearity.

Feature Selection for PJM Market:
In the process of feature selection, we did not select all variables at once, but adopted a step-by-step filtering approach. We started with the overall correlation heatmap and observed which variables have a high correlation with the target value PJMC_SRT_LPM as candidates for the first round. Next, we further compare the relationships between these candidate variables and try to avoid selecting features that are highly correlated with each other to reduce the interference of multicollinearity on the model.

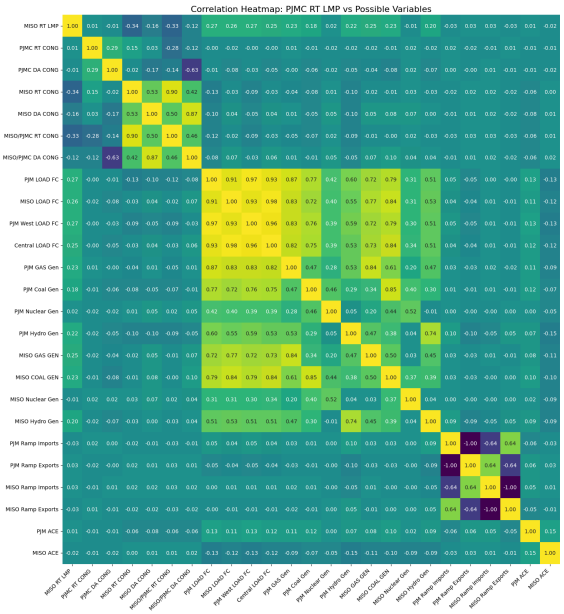
Through this layered process of selection and comparison, we ultimately identified six key feature variables: PJMC_DA_LMP、MISO_DA_GAS、PJM_GAS_Gen、MISO_RT_LMP、MISO_RTLOAD, and the target variable PJMC_RT_LMP. These variables provide useful information from different dimensions and have low repeatability with each other, making them a relatively reasonable and concise combination of features.

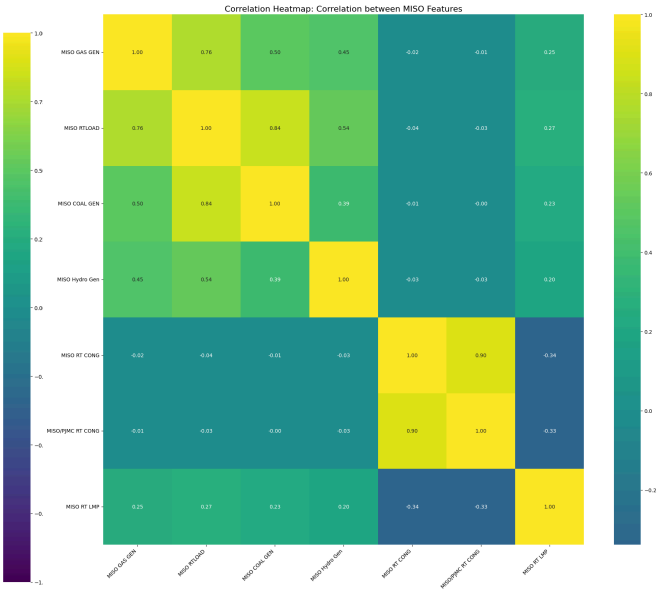


- PJMC_CA_LMP: The daily electricity price in the PJM market, used to reflect the market's expectations for future prices;
- MISO-DA_GAS: The daily natural gas power generation forecast in the MISO region represents the main dispatchable energy supply situation in the region;
- PJM-GAS-Gen: The real-time gas-fired power generation of PJM reflects the current power generation response capability of the system;
- MISO-RT_LMP: Real time electricity prices in the ISO market, which can be used to observe the price linkage between adjacent markets;
- MISO-RTLOAD: MISO's real-time power load is used to capture real-time changes in regional power demand.

Feature Selection for MISO Market:

We finally selected the following features by targeting the PJM RT LMP(Real-time electricity price in the PJM market) column:





We finally selected the following features by targeting the MISO RT LMP(Real-time electricity price in the MISO market) column :

- MISO-GAS-GEN: The natural gas power generation in the MISO region reflects the output level of the main regulated power source.
- MISO-RTLOAD: The real-time load of MISO, representing the current electricity demand intensity in the area.
- MISO-COAL_GEN: The coal-fired power generation of MISO represents the actual power generation situation of the basic power source.
- MISOHydro_Gen: The hydroelectric power generation of MISO has a certain regulatory effect on electricity prices during periods of low load.
- MISO-RTCONG: The cost of transmission congestion within MISO, used to measure the current transmission pressure of the system.
- MISO/PJMC_CONG: The difference in congestion costs between MISO and PJMC reflects whether cross regional power flow is restricted.

Data Cleaning & Preprocessing:

We cleaned all selected columns by removing commas and converting strings to numeric values, and rows with missing values were dropped to ensure a clean dataset. In addition, Feature values were standardized using StandardScaler() to give equal weight to each input.

Train-Test Split:

By using train_test_split, we split the dataset into 80% training and 20% testing. This ensures that the model is evaluated on unseen data to prevent overfitting and evaluation generalization. Overfitting is when the model learns too well about the training data (including noise and random patterns) and therefore performs well on the training data but poorly on new unseen data.

Evaluate Algorithms:
In order to find the most suitable regression model for our task, we tried various common algorithms and compared their performance, including linear regression, lasso regression, ridge regression, elastic net, KNN, SVR, decision tree regression. We also used several ensemble models such as random forest, gradient boosting, and extra trees.

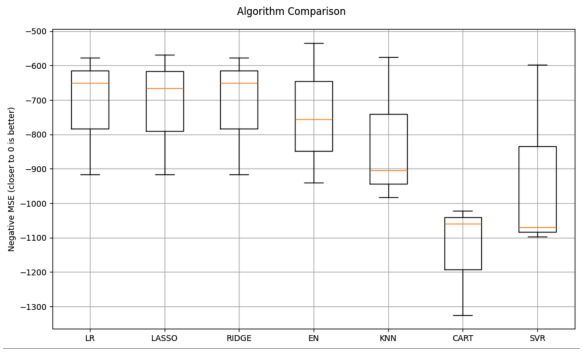
To ensure that the evaluation results are as accurate and representative as possible, we conducted 10-fold cross validation on each model and used MSE as the evaluation metric.

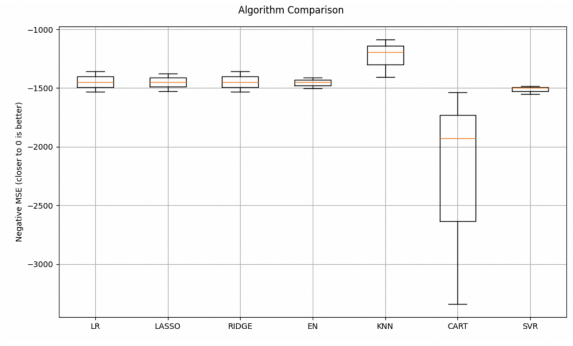
From the results of PJM, the KNN regression model performs the best. Its average error is the smallest (about 821), and it also shows strong stability in different folds, with a standard deviation of only about 10. This indicates that KNN can effectively capture the local patterns between historical load and electricity prices on this dataset, and make relatively accurate predictions for future electricity prices.

KNN regression model also performs in a good way in the MISO market's data, but its overall MSE is larger than the PJM market. Its Mean MSE is the smallest(around 1228), but with a standard deviation of 132. The SVR, instead, has the smallest standard deviation of 27.97.

In addition, SVR has certain advantages in capturing nonlinear relationships. The average error of SVR is around 760 in the PJM market, which performs significantly better than the traditional linear models. The performance of several linear models (Linear Regression, Ridge, Lasso, and ElasticNet) is relatively similar, with errors mostly ranging between 787 and 809. This indicates that although there is a certain linear trend between electricity prices and input characteristics, relying solely on linear relationships is still insufficient to accurately model complex market fluctuations.

In contrast, CART performs the worst in both PJM and MISO markets. Its error is not only the highest (over 1150.50 in PJM and over 2267 in MISO), but also fluctuates greatly, meaning that the model is prone to overfitting training data and has weak generalization ability.





We visualized the errors of the model in the form of a box plot and visually compared their performance differences at different validation splits(PJM at left, MISO at right). In addition, we also listed the average error and standard deviation of each model to help us see more clearly which models perform better in balancing accuracy and stability.

Overall, KNN is the model with the best overall performance in our current testing. However, considering its sensitivity to feature scales, further attempts can be made to normalize the features in more detail or introduce more feature engineering techniques in the future. And although integrated models such as random forests and gradient boosting have not been extensively tuned in the current version, they usually have good generalization ability in electricity demand forecasting tasks and are worth exploring in future optimization.

Improving Results:
To optimize the KNN model, we use grid search to perform over different values of K. After tuning the model, we get the result that when the KNN model with K=12(12 neighbors), it provided the most accurate predictions during validation for the PJM market and had the lowest cross-validated MSE of 736.76. Applying the same method to the MISO market, the KNN model with K = 16 (16 neighbors) gets the most accurate prediction with the Lowest MSE of 1152.5.

Present/Summarize Results:
For the PJM market, after determining the optimal K value, we applied the model to an unseen test set. The mean square error of the test set is 515.12, which is lower than the average error in cross validation, indicating that the model performs stably on new data without overfitting.

Most of the predicted values are very close to the actual values, and the overall prediction effect is good. But we also found a clear outlier: The model's predicted value was 40.21, while the actual value was 246.69, with a difference of over 200 between the two. This error does not mean that the model itself has failed, but is more likely due to external factors that some models failed to cover, such as sudden power grid congestion, extreme weather, or system failures.

This indicates that our model can handle conventional situations well, but there are still limitations when facing some extreme events. If we hope to further improve the robustness of the model and its ability to handle special situations in the future, we can consider adding more real-time or situational features, such as meteorological data, system operation status, or market warning information.

For the final model performance of the MISO market, the test MSE is 2770.76, which is higher than the cross-validated MSE. This may be due to overfitting the training data, where our model may have learned patterns specific to the training data that do not generalize well to unseen data. Cross-validation, on the other hand, gives a more optimistic estimate of performance because it still uses part of the training data for validation.

This indicates that the model doesn’t generate the data very well due to overfitting. We hope to avoid high multicollinearity and improve the interpretability and stability of the model. We will consider removing highly correlated features, such as removing a highly correlated variable based on domain knowledge or feature importance. Or we can use only the most predictive features, but it might be difficult to do because the variables in the dataset have relatively low correlation with our target variable.