# Time Series:

## Hourly Energy Consumption

Rajas (MDS202131), Rishika (MDS202135)

April 13, 2023

# Introduction to Dataset

- Data of hourly power consumption of PJM Interconnection LLC from year 2002 to 2018
- PJM Interconnection LLC (PJM) is a regional transmission organization (RTO) in the United States. It is part of the Eastern Interconnection grid operating an electric transmission system serving all or parts of Delaware, Illinois, Indiana, Kentucky, Maryland, etc.

# Dataset Summary

- We have in total of 145366 data points from 1st Jan 2002 to 2nd August 2018
- There are no null values in the data.

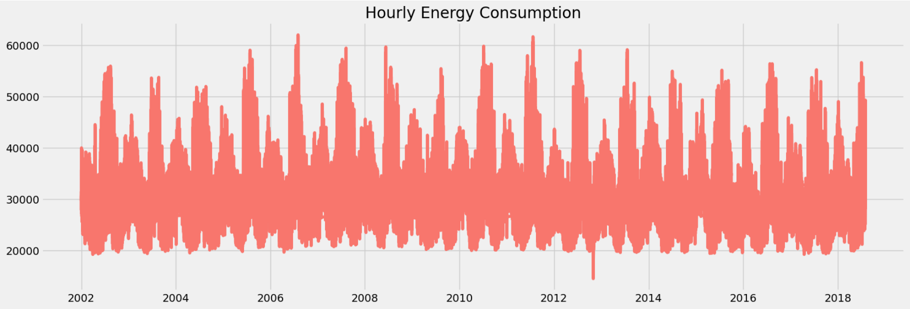
| | Energy |
|---|---|
| **count** | 145366.000000 |
| **mean** | 32080.222831 |
| **std** | 6464.012166 |
| **min** | 14544.000000 |
| **25%** | 27573.000000 |
| **50%** | 31421.000000 |
| **75%** | 35650.000000 |
| **max** | 62009.000000 |

- Note : The data points are in MegaWatts

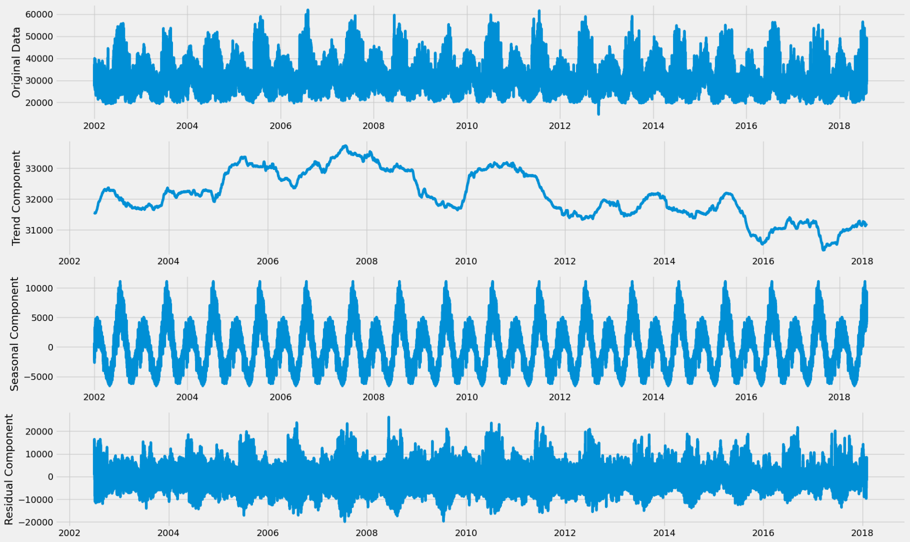
# Objective

- Explore and visualize the data
- Check for Stationarity and any underlying patterns in the data over the years
- To check whether there is any trend or seasonality in the data
- To build models on the dataset to predict energy consumption
- To test the models on the test set and compare the results

# Exploratory Data Analysis



# Classical Decomposition



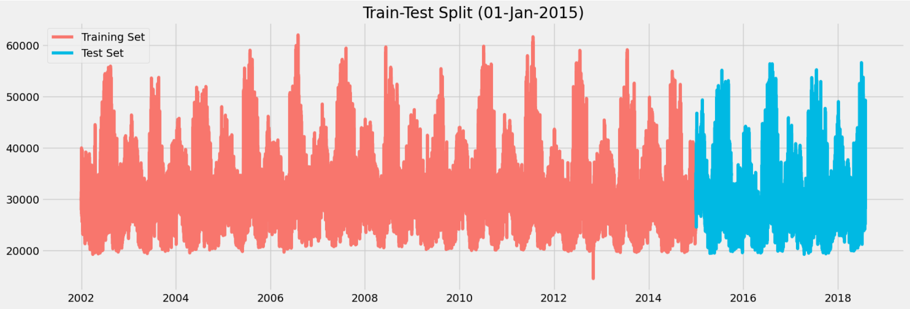
# Stationarity

- Augmented Dickey Fuller test
  - The Augmented Dickey-Fuller (ADF) test, is a statistical hypothesis test
  - Used to determine whether a time series data set is stationary or not
  - An extension to the Dickey-Fuller test, which tests for the presence of a unit root in a time series
  - Adds additional terms to the Dickey-Fuller test equation to account for the potential presence of trends, seasonality, and autocorrelation in the data.
  - The null hypothesis of the ADF test is that the time series has a unit root, indicating non-stationarity

# Results of ADF test

```
Observations of Dickey-fuller test
Test Statistic                  -1.882891e+01
p-value                          2.022125e-30
#lags used                       7.400000e+01
number of observations used      1.452910e+05
critical value (1%)             -3.430395e+00
critical value (5%)             -2.861560e+00
critical value (10%)            -2.566781e+00
dtype: float64
```

# Train-Test Split



# Prophet Model

- Developed by Facebook's Core Data Science team.
- Designed to predict for data with complex patterns like trends, seasonality, and holiday effects.
- Based on an additive regression model that decomposes the data into four components: **trend**, **seasonality**, **holidays**, and **an error term**.

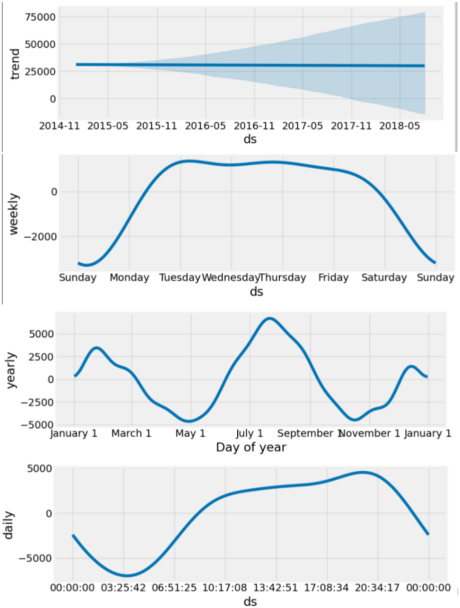
$$y(t) = g(t) + h(t) + s(t) + \epsilon(t)$$

where $g(t)$ is trend component, $s(t)$ is the seasonal component, $h(t)$ captures the holiday effects, and $\epsilon(t)$ is a white noise error term.

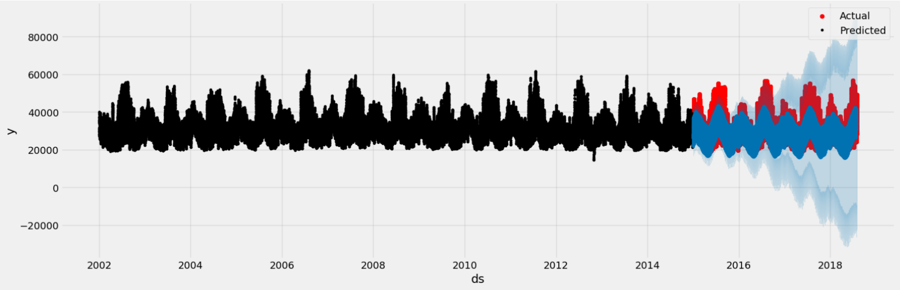
# Prophet Model

- Using time as a regressor, Prophet fits several linear and non linear functions of time as components. So, the forecasting problem is being framed as a curve-fitting exercise rather than looking explicitly at the time based dependence of each observation within a time series.
- The input to Prophet is always a dataframe with two columns: ds and y. The ds (datestamp) column should be of a format expected by Pandas, ideally YYYY-MM-DD for a date or YYYY-MM-DD HH:MM:SS for a timestamp. The y column must be numeric, and represents the measurement we wish to forecast.

# Prophet Model Components



# Prophet Model Results



# Prophet Model Evaluation Metrics

## Metrics for Prophet's performance on test set

```
[ ] y_true=test['Energy']
    y_pred=test_pred['yhat']
    print("Mean absolute error is ",mean_absolute_error(y_true,y_pred))
    print("Mean squared error is ",mean_squared_error(y_true,y_pred))
    print("Mean absolute percentage error is ",mean_absolute_percentage_error(y_true,y_pred))
    print("R square is ",r2_score(y_true,y_pred))
```

```
Mean absolute error is  3104.669376503115
Mean squared error is  16984051.40389661
Mean absolute percentage error is  9.620061802158176
R square is  0.5917286822204624
```

# XGBoost

- XGBoost stands for extreme gradient boosting machine.
- It is a tree based ensemble machine learning algorithm which is a scalable machine learning system for tree boosting.It uses more accurate approximations to find the best tree model.
- Using XGBoost for time-series analysis is an advance approach of time series analysis. this approach also helps in improving our results and speed of modelling.

# XGBoost

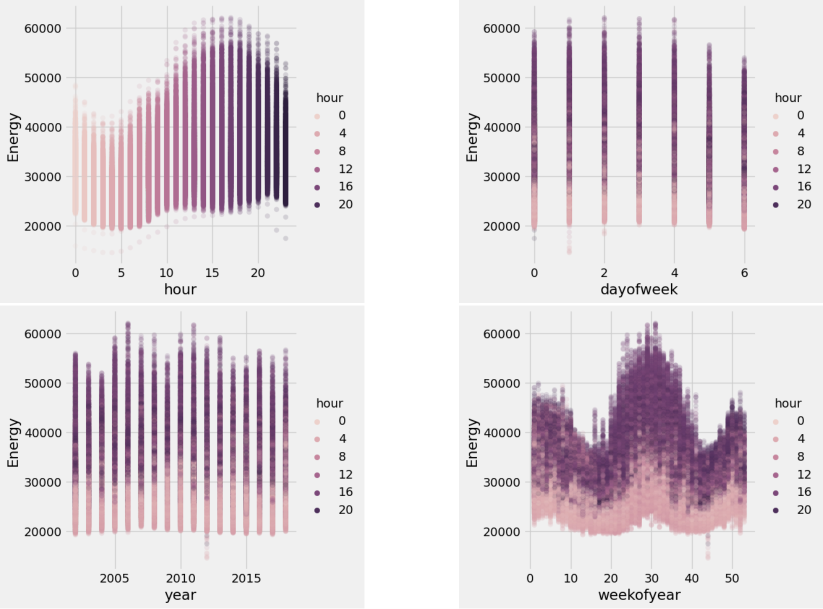
- When to Use XGBoost? Consider using XGBoost for any supervised machine learning task when satisfies the following criteria:
  - When you have large number of observations in training data.
  - Number features is less than the number of observations in training data.
  - When the model performance metrics are to be considered.

# Feature Engineering

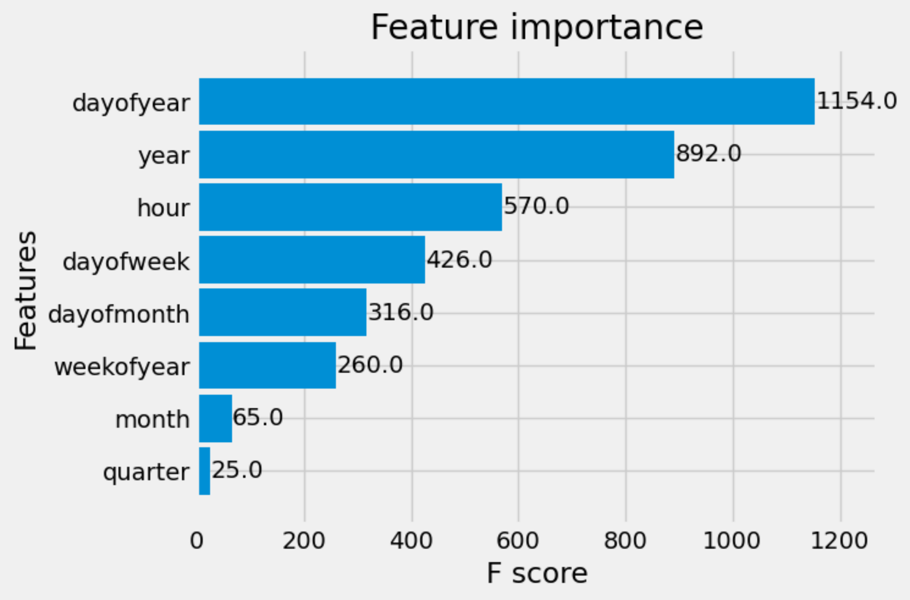
Create features such as hour, day of the week, month, week of the year, day of the year etc., to understand if there are any patterns in the power consumption depending on these factors.

| Datetime | hour | dayofweek | quarter | month | year | dayofyear | dayofmonth | weekofyear | Energy |
|---|---|---|---|---|---|---|---|---|---|
| 2009-10-30 19:00:00 | 19 | 4 | 4 | 10 | 2009 | 303 | 30 | 44 | 32392.0 |
| 2006-10-02 15:00:00 | 15 | 0 | 4 | 10 | 2006 | 275 | 2 | 40 | 32744.0 |
| 2008-01-15 22:00:00 | 22 | 1 | 1 | 1 | 2008 | 15 | 15 | 3 | 38510.0 |
| 2006-10-08 16:00:00 | 16 | 6 | 4 | 10 | 2006 | 281 | 8 | 40 | 26989.0 |
| 2002-01-26 06:00:00 | 6 | 5 | 1 | 1 | 2002 | 26 | 26 | 4 | 26029.0 |

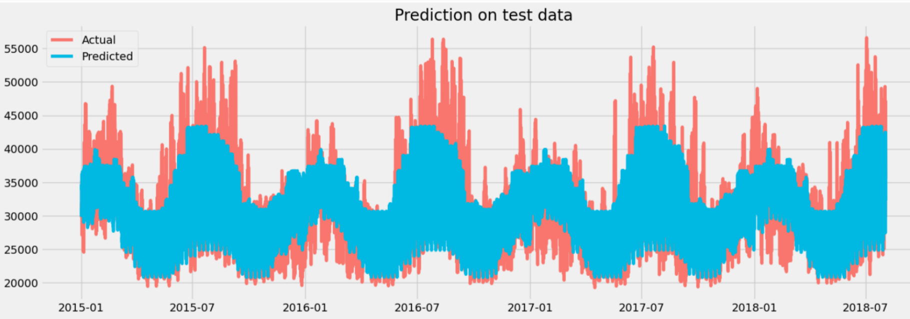
# Patterns in the data



# Feature Importance of XGBoost



# XGBoost Results

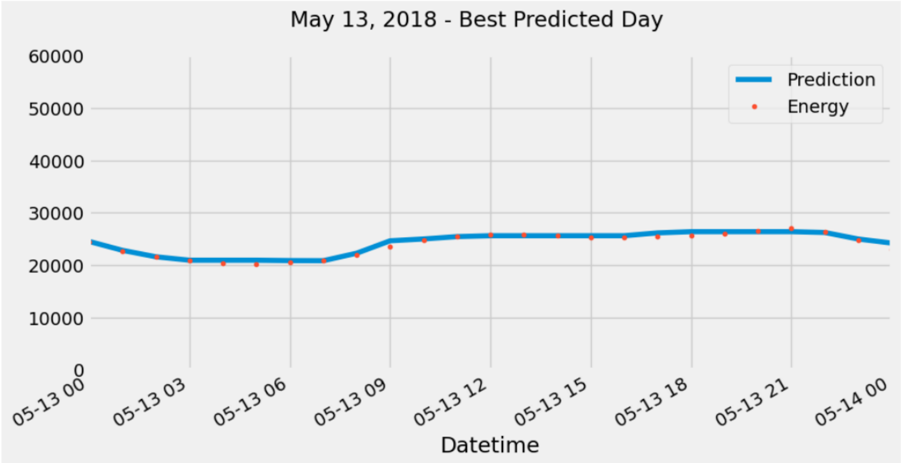


# XGBoost Evaluation Metrics

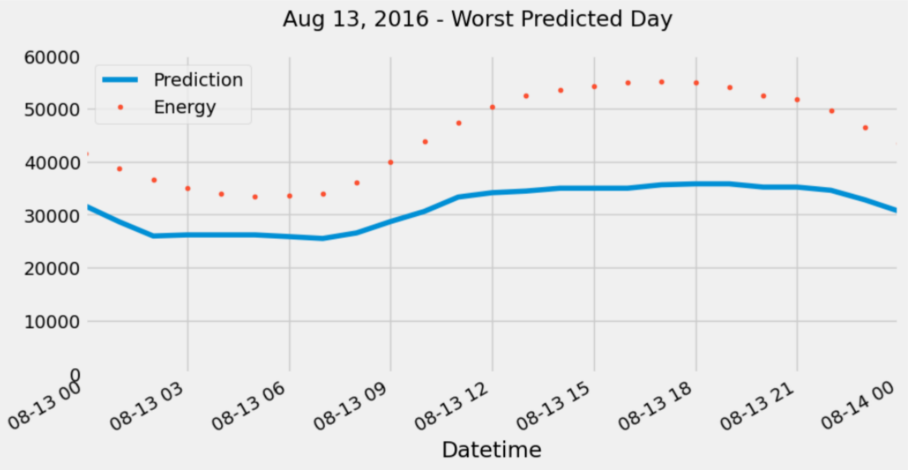
```
print("Mean absolute error is ",mean_absolute_error(y_true,y_pred))
print("Mean squared error is ",mean_squared_error(y_true,y_pred))
print("Mean absolute percentage error is ",mean_absolute_percentage_error(y_true,y_pred))
print("R square is ",r2_score(y_true,y_pred))
```

```
Mean absolute error is  2757.2169004982866
Mean squared error is  13960913.30565261
Mean absolute percentage error is  8.510847348494002
R square is  0.6644004226578725
```

# XGBoost Best Predicted Day



# XGBoost Worst Predicted Day



# Comparison of Prophet Model, with and without Holidays and XGBoost

| Performance Measure | Prophet | XGBoost |
|---|---|---|
| Mean Absolute Error | 3105 | 2757 |
| Mean Square Error | 16984051 | 13960913 |
| Mean Absolute % Error | 9.62% | 8.51% |
| R Square | 0.59 | 0.66 |