Business Analytics and Data Mining Championship 2019

Analytics Report (Case 1)

Presented by team **BADM_1002**:
Palak Sood
Varan Singh Rohila
Aryan Khandal

Introduction

> “Where there is data smoke, there is business fire.”
>
> — **Thomas Redman**

Data is the fuel of today’s success engine. Changes and disruption are all fueled by data. In the textbook, data science is a multidisciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from structured and unstructured data. But on the practical side, the definition goes way beyond those few words.

Today, data science is about telling a story; opening new doors to unexplored areas where lies the truth to be told. Here in this brief report, we highlight this storytelling capabilities of data using techniques like exploratory data analysis and generating insights using statistical models. This report elucidates a few of the many capabilities of applied data science in present industries.

Contents:

General guidelines for reading this report:

- Bullet points are used most extensively to bring out clarity and clearness.
- Supporting graphs, images and code snippets are pinned wherever necessary.

Objective

For this addressing the case objective, an imaginary scenario/case study has been considered so as to ease the explainability of our approach and depth of the mentioned implementation. Here goes the scenario.

A renowned energy corporation approached our firm with the following requirements:

- Since it is crucial to match supply with demand, what pattern or schedule they should adopt for their energy/power purchases?
- What are the estimated energy consumption rates of their consumers for the upcoming years (long-term forecast)?

Analyzing their requirements and domain of operation, the case objective formed was medium-term/long-term energy demand forecasting adhering to their requirement of the schedule (medium-term forecasts are used to schedule the energy/power processes).

Hence, our case objective defined is **"Energy Demand Forecasting for Power Industry"**.

What is energy load forecasting?

Electric load forecasting refers to forecasting electricity demand and energy a few minutes to a few decades ahead. As a fundamental business problem in the utility industry, load forecasting has extensive applications, such as power systems operations and planning, customer service, revenue management, energy trading, and so forth. Organizations in many sectors of the utility industry need load forecasts, such as the utilities themselves, regulatory commissions, retailers, and trading firms.

Evaluation metrics/scoring logic:

The metrics used to evaluate the forecasted values is Mean Absolute Percentage Error (MAPE) as mentioned in the case notes.

Data Preparation/Preprocessing:

For achieving better results from the applied models, the format of the data has to be in a proper manner. So, the following preprocessing techniques were applied to the following column in the given dataset:

1. DATE: this column was converted to pandas data type, datetime for easier handling and resampling of data.
2. Daylight: a new column daylight was created by subtracting Sunrise and Sunset, and then it was converted to total daylight minutes.
3. Different scalers were used to scale the TOTAL Load and the best one was chosen based on their performance.

Handling missing values:

1. It was found after analysing the data set that average temperature values on the date 31-08-2006 were missing.
2. First the average temperature values of the year 2004 and 2005 were observed and found to be coherent.
3. So to fill the missing values, the gaps were replaced by the values observed in the other two years.

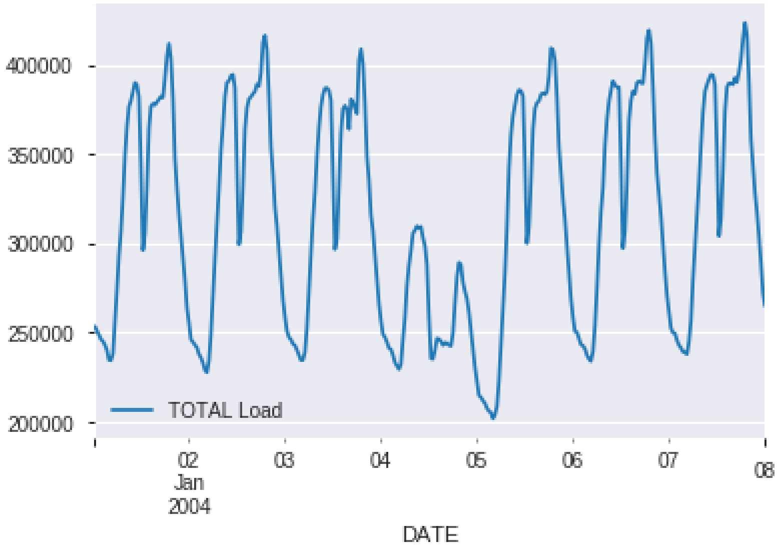
Feature Engineering:

By studying the dataset we can see a yearly/monthly/weekly seasonality so by using this property of the dataset we can incorporate:

1. Day of Week: To understand this, we can think about the weekends, there will be data seasonality based on the day of the week and thus will be a factor affecting the power consumption.
2. Week of Year: Most of the people go on trips and vacations depending upon the week holidays, so if there are two to three public holidays then people of that area would consume less energy thus creating a data trend making this as an import atn factor.

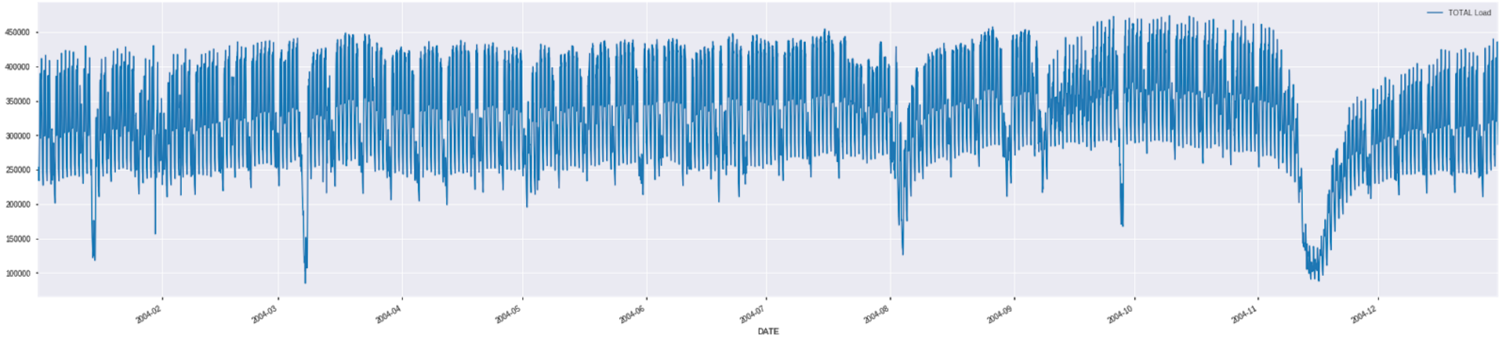
Data Analysis

First of all, we conducted exploratory analysis in the data presented to us. Graphs were plotted inorder to visualize the trends and seasonality of the time series.



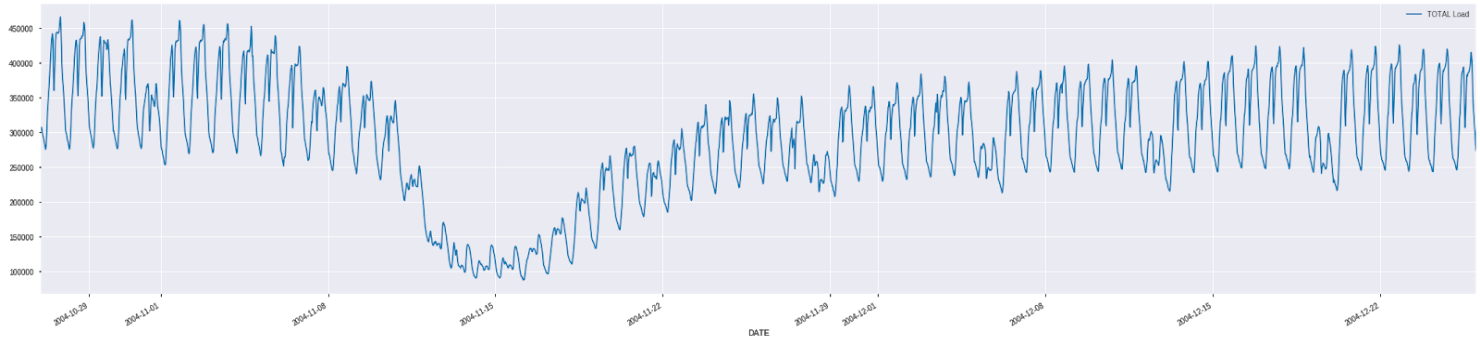
1st Week's power consumption

Simple viewing of the graph shows clear seasonality with slight decrease in the middle.



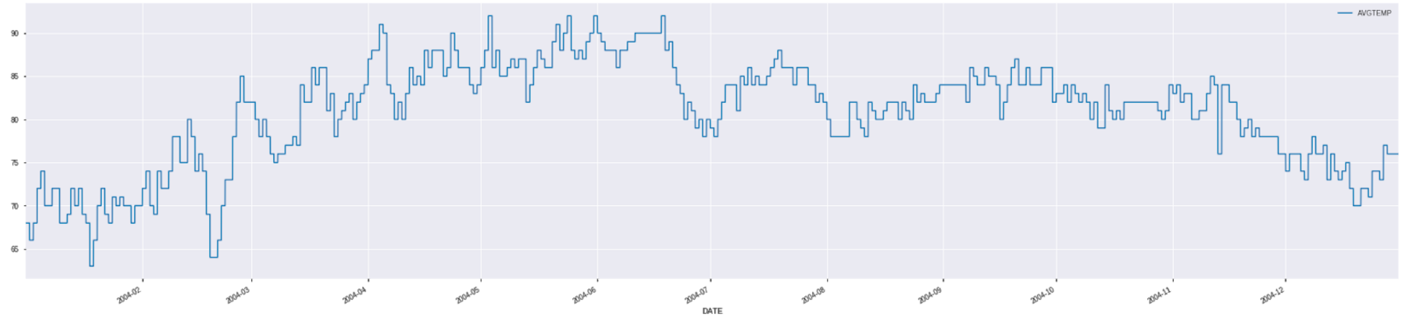
1st year's power consumption

Seasonality is prevalent all over the year with some drops and a decreasing curve at the end of the graph (between 11th and 12th month). Expanding on the curve we found out an interesting trend.

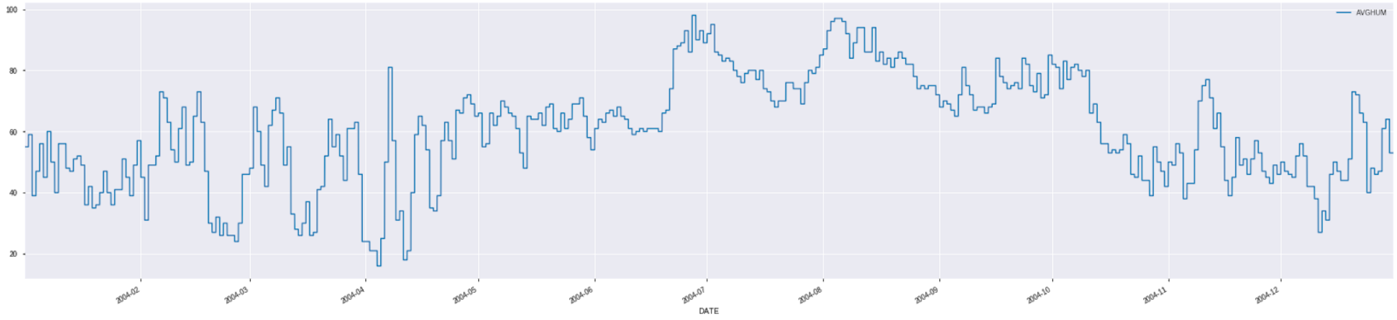


Expanded view of the decreasing curve

The power consumption first decreases and then increases at a rapid pace with regular drops in between. The same trend is observed in the year 2005 and 2006.



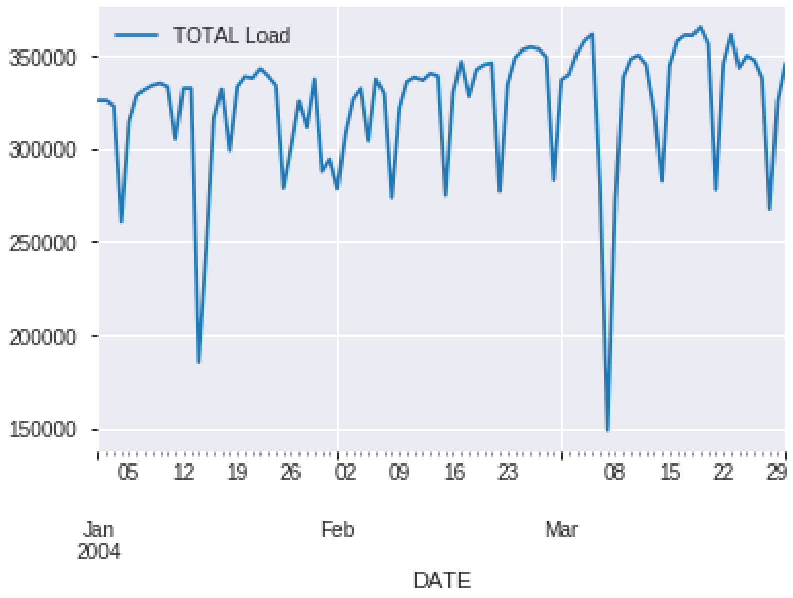
1 year’s avg temperature variation



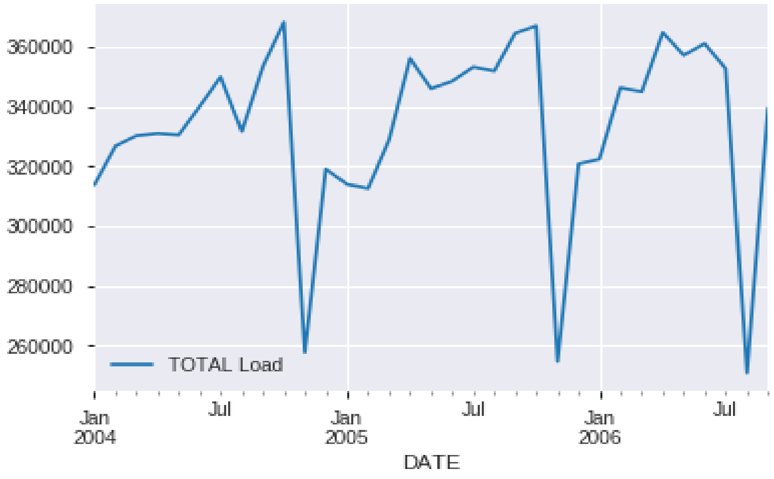
1 year’s avg. humidity variation

Possible reasons for drops:

1. The timestamp of the drops were recorded and then looked up in the dataset provided. It was observed that the drops occurred mostly when there was a holiday.
2. For example, on 6th and 7th March, 2004 there is a clear drop in the power consumption and they are holidays.
3. The reason for the decreasing curve near the end of the year could be a season change since power consumption varies greatly in the event of a season change.



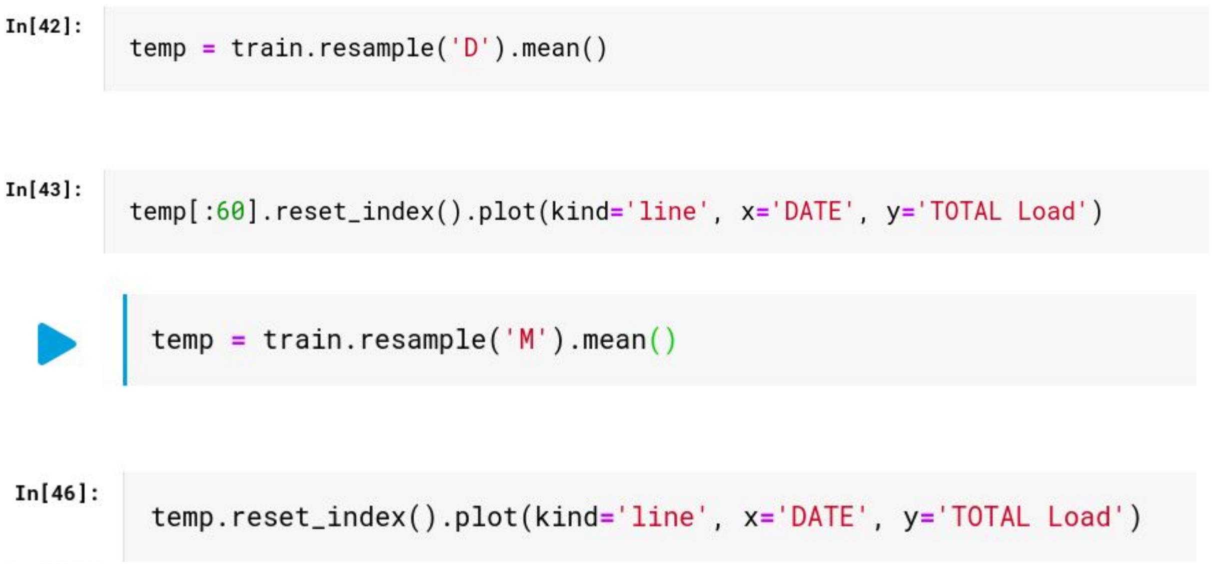
3 month power consumption resampled daily by mean



Full data resampled monthly by mean

Resampling was done monthly and daily to view the trends in the data. There are no constant trends but it can be seen that it is seasonal and quite stationary.

Code snippets:



```
In[42]:  temp = train.resample('D').mean()
```

```
In[43]:  temp[:60].reset_index().plot(kind='line', x='DATE', y='TOTAL Load')
```

```
temp = train.resample('M').mean()
```

```
In[46]:  temp.reset_index().plot(kind='line', x='DATE', y='TOTAL Load')
```

Resampling daily and monthly

Insights from EDA:

1. Drops in power consumption are observed usually when there is a holiday.
2. There is a decreasing and increasing trend at the end of every year which probably indicates a season change.

Approach and model building:

1. SARIMA

Autoregressive Integrated Moving Average, or ARIMA, is one of the most widely used forecasting methods for univariate time series data forecasting.

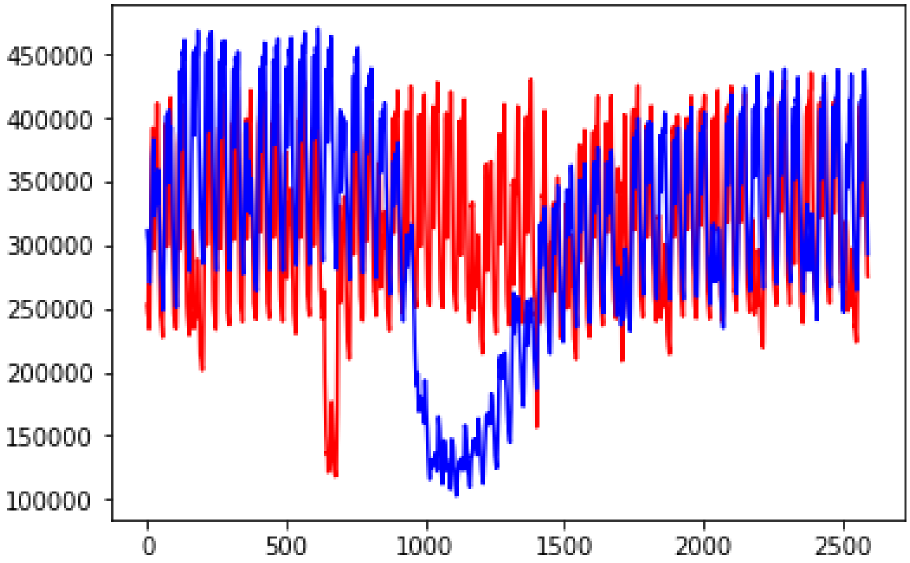
Although the method can handle data with a trend, it does not support time series with a seasonal component.

An extension to ARIMA that supports the direct modeling of the seasonal component of the series is called SARIMA.

Seasonal Autoregressive Integrated Moving Average, SARIMA or Seasonal ARIMA, is an extension of ARIMA that explicitly supports univariate time series data with a seasonal component.

It adds three new hyperparameters to specify the autoregression (AR), differencing (I) and moving average (MA) for the seasonal component of the series, as well as an additional parameter for the period of the seasonality.

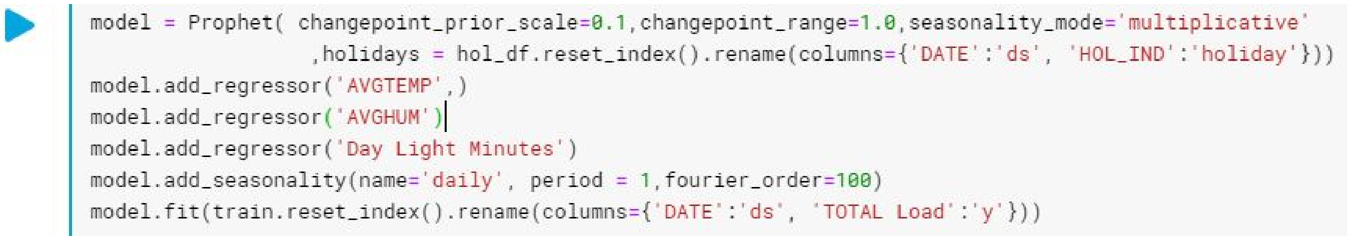
The **MAPE** obtained for the **validation dataset** is 28.



2. Prophet

Prophet is a procedure for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. It works best with time series that have strong seasonal effects and several seasons of historical data.

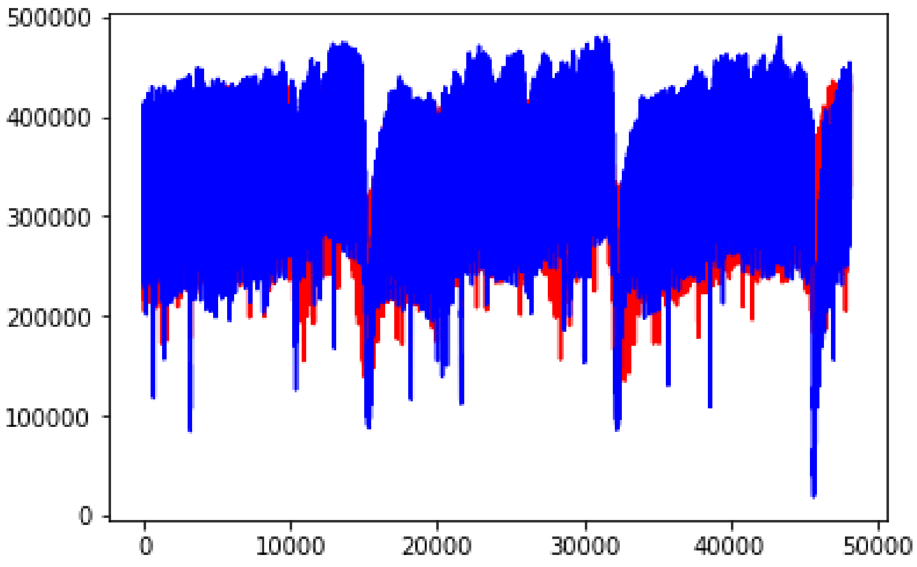
The Prophet took as input the **Datetime** column and **Total Load** as major inputs. The Holiday Indicator column was used as a parameter to the Prophet function. Additional regressors are added to the linear part of the model i.e. Average Temperature, Average Humidity and Day



```
model = Prophet( changepoint_prior_scale=0.1,changepoint_range=1.0,seasonality_mode='multiplicative'
                 ,holidays = hol_df.reset_index().rename(columns={'DATE':'ds', 'HOL_IND':'holiday'}))
model.add_regressor('AVGTEMP',)
model.add_regressor('AVGHUM')
model.add_regressor('Day Light Minutes')
model.add_seasonality(name='daily', period = 1,fourier_order=100)
model.fit(train.reset_index().rename(columns={'DATE':'ds', 'TOTAL Load':'y'}))
```

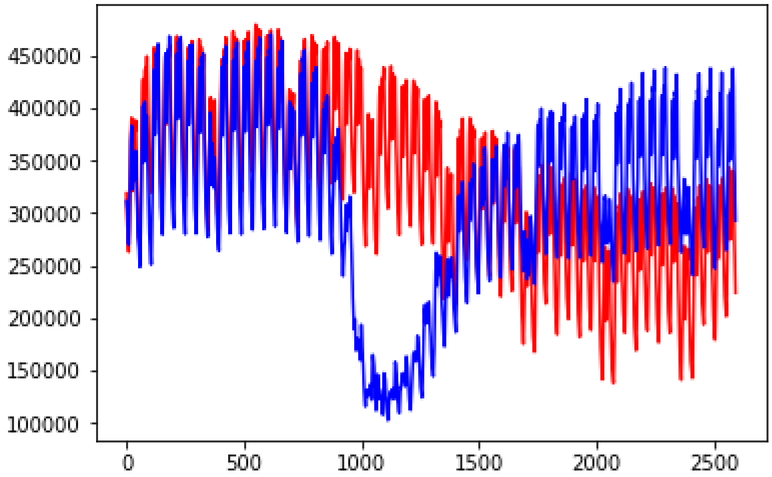
Light Minutes were input using add_regressor function.

Training Curve:



Actual
Predicted

Testing Curve:



The **training MAPE** for the dataset is **10** while the **validation MAPE** is **32**. The dip in the graph is not getting detected by the model as the dataset is for only 2 years, so the drop in the energy consumption occurs only two times in the respective two years which is very less compared to the granularity of data i.e. for 30 minute intervals.

3. XGBoost

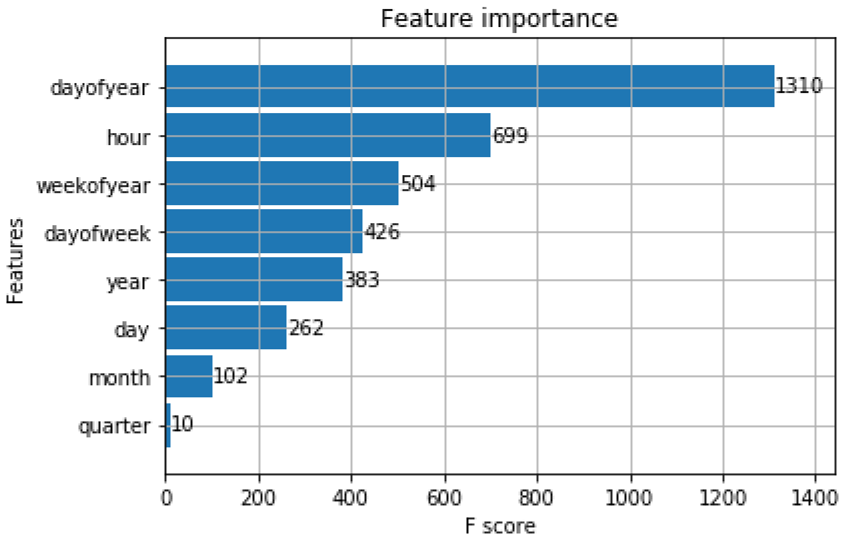
XGBoost, a Gradient Boosted tree algorithm can be used for the forecasting problems.

Feature Importances

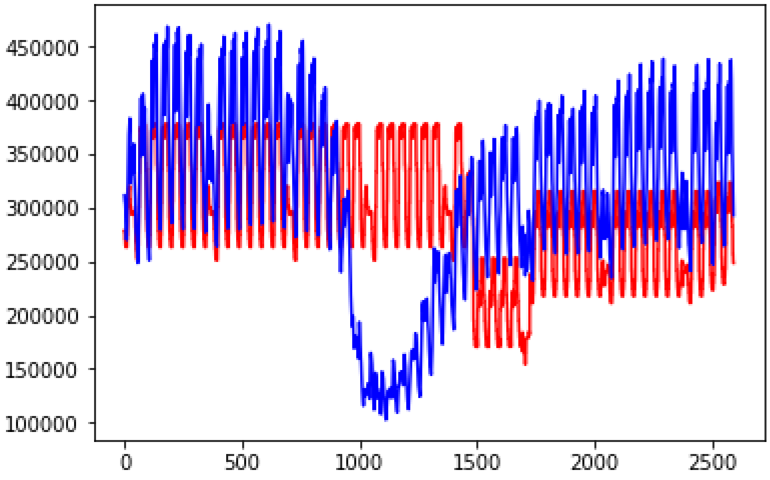
Feature importance is a great way to get a general idea about which features the model is relying on most to make the prediction. This is a metric that simply sums up how many times each feature is split on.

We can see that the day of the year was most commonly used to split trees, while hour and year came in next. Quarter has low importance due to the fact that it could be created by different dayofyear splits.

This signifies that the seasonality of the data is yearly which is true in terms of electric power consumption as all the major factors affecting consumption repeat over the years.



The **testing MAPE** observed is **32**.



Actual
Predicted

Conclusions/Inferences:

1. Since the energy consumption is low on holidays, the energy corporation should generate less power for those days.

2. The dip in the graph is not getting detected by the model as the dataset is for only 3 years, so the drop in the energy consumption occurs only two times in the respective three years which is very less compared to the granularity of data i.e. for 30 minute intervals.

3. The dip in the graph probably represents a season change since there is a sharp decrease in the power consumption. The corporation can limit the generation of power during that period.

4. The training MAPE for the dataset is 10 while the validation MAPE is 32 for prophet, 28 for SARIMA model.

5. Day of the year was most commonly used to split trees, while hour and year came in next thus stating their respective importance.

6. The forecasted energy demand values can be used to optimize the power generation process thus decreasing wastage and increasing the profits of the corporation.

7. The SARIMA model has given the closest forecasted values and will also prove to be effective in the production environment.