ELECTRICAL PRICE PREDICTION

DS5110 PROJECT

Weipeng Zhang, Soham Shinde, Ziyue Wang, Mingni Luo

Background

- Forecasting in energy problem

- Dataset

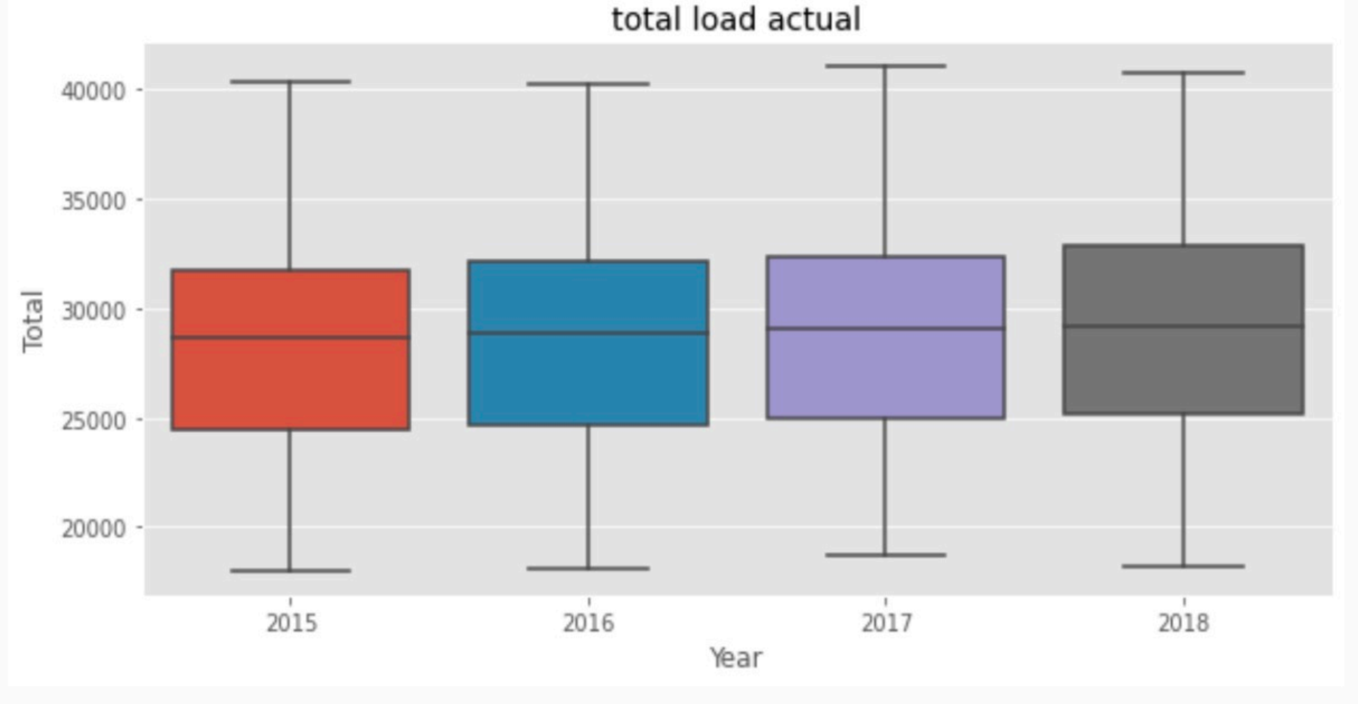
? Why...

Feature Type	**Columns**
Date	*“time”*
Energy information	“generation biomass”, “generation fossil gas”, “generation fossil oil”, “generation solar”, “total load actual” etc.
Weather information	*“temp”*, “pressure”, “humidity”, “wind speed” etc.
Price information	*“price day ahead”, “price actual” etc.*

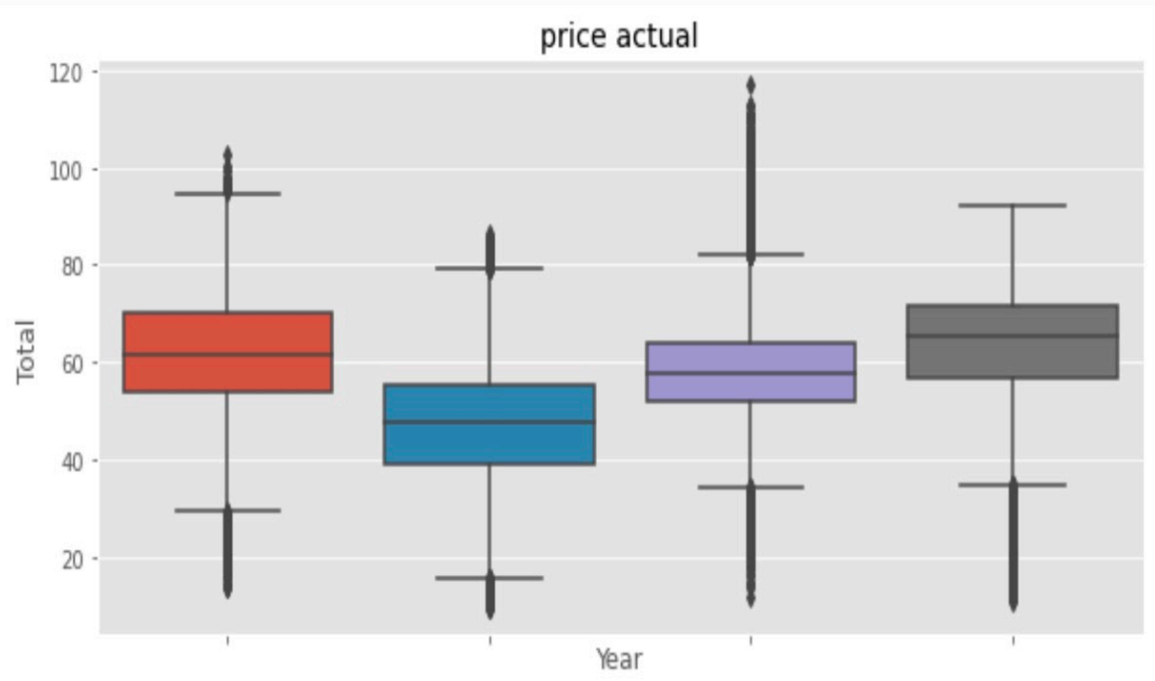
EDA

- Goal
  - *Find a pattern*
- Distribution

•generate distribution of "Total load actual" by year



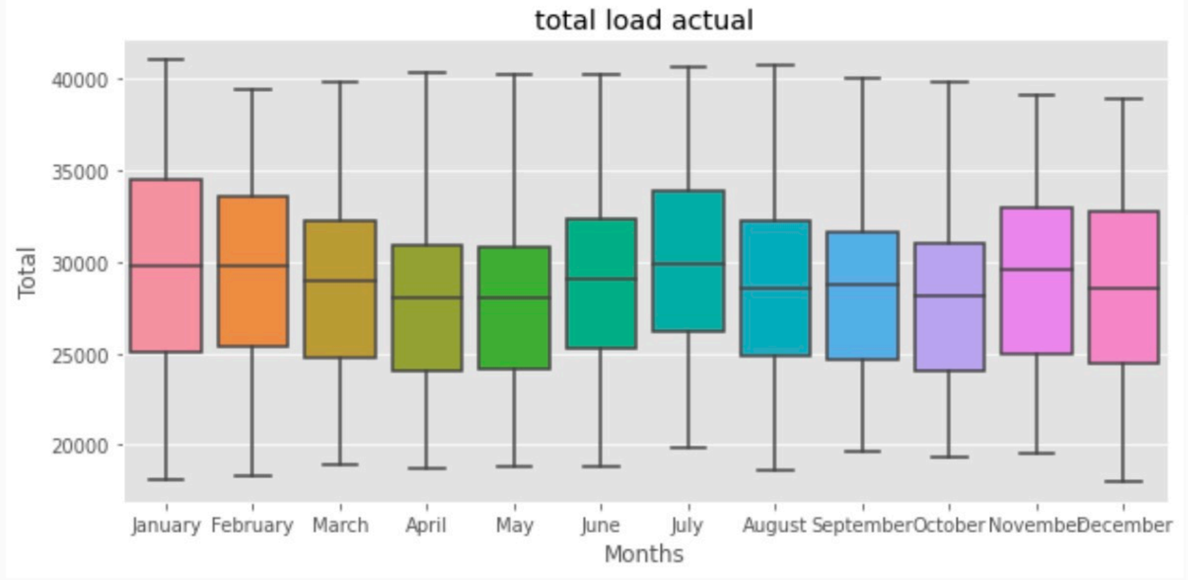
•generate distribution of "price" by year



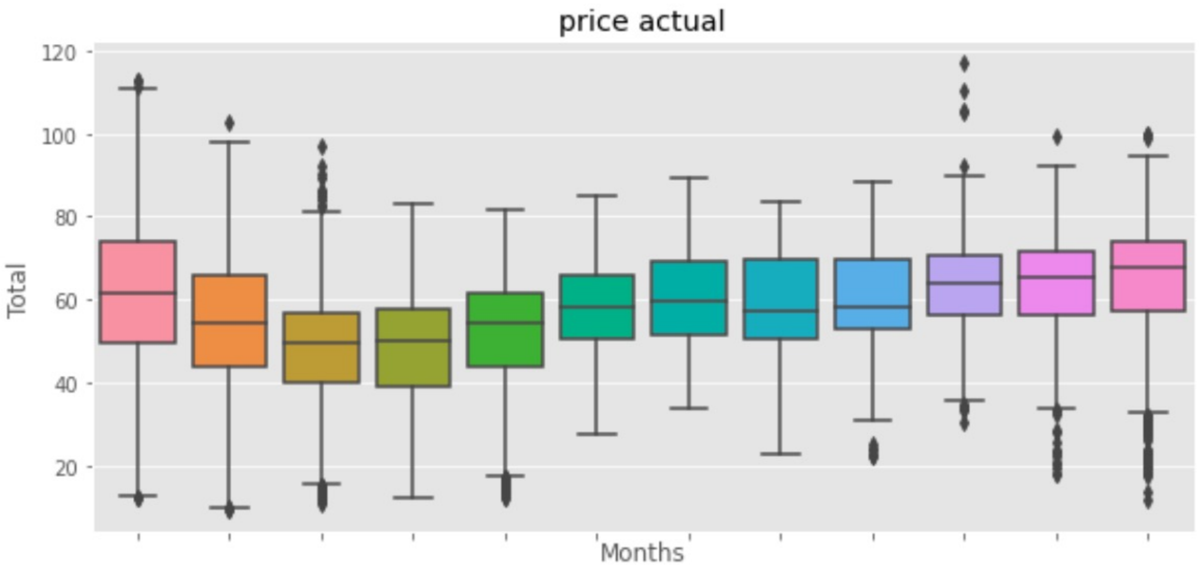
EDA

- Distribution

•generate distribution of “Total load actual” by month



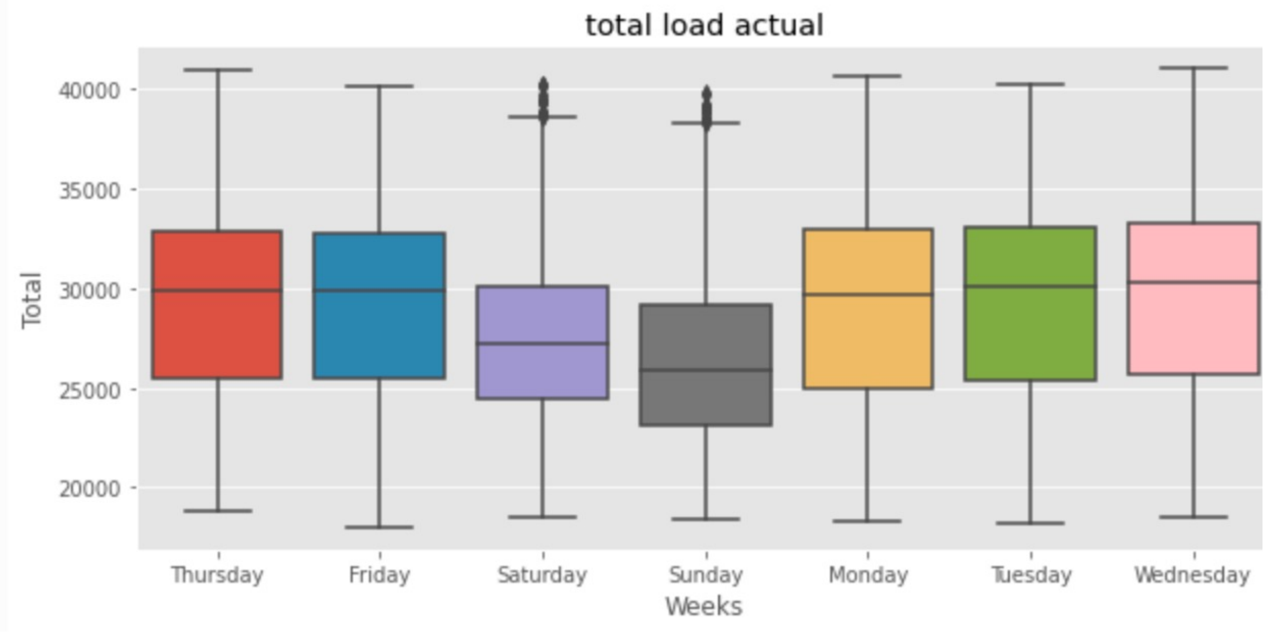
•generate distribution of “price” by month



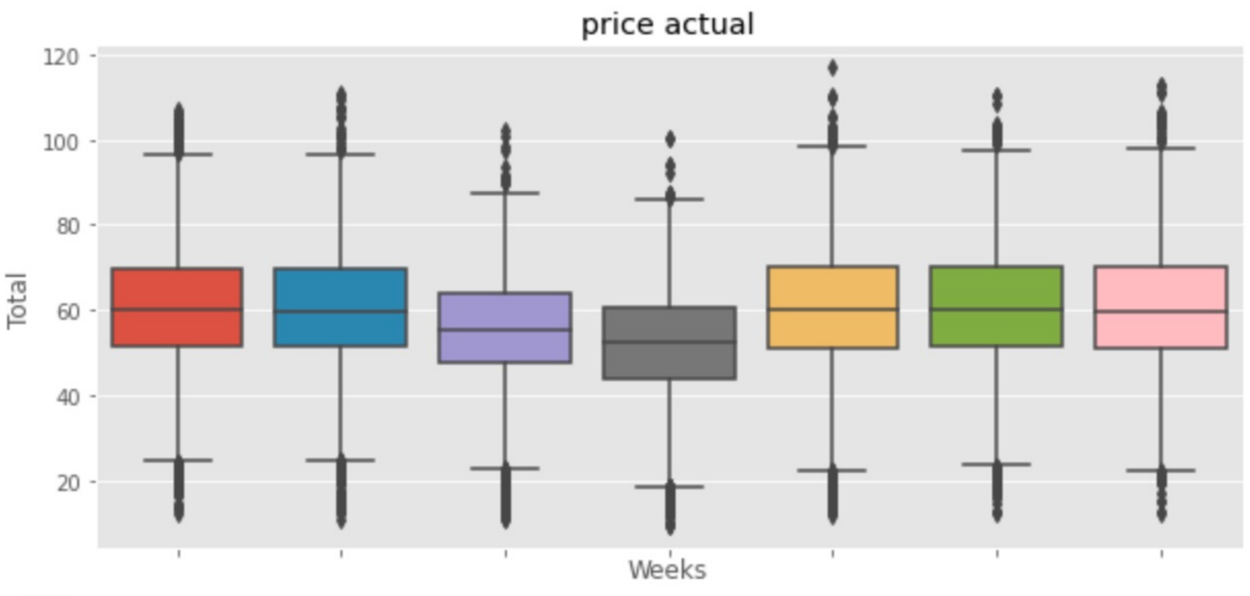
EDA

- Distribution

•*generate distribution of “Total load actual” by week*



•*generate distribution of “price” by week*



? Assumption:

- Seasonality

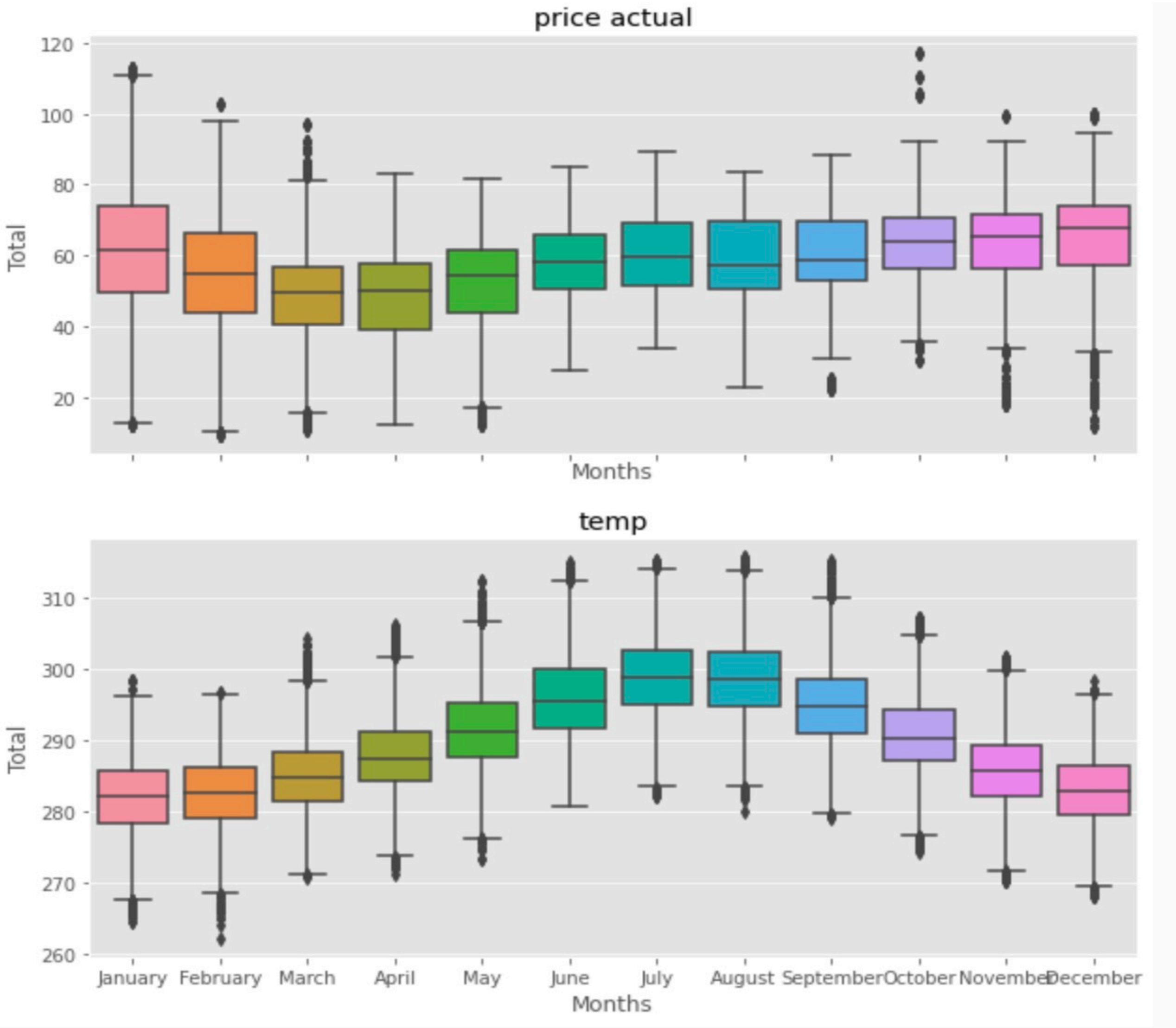
- temperature

- time window

EDA

- Assumption

  distribution of temperature and price



💥 *interesting!!!*

1.when temperature is low…

2.when temperature is high…

EDA

- Correlation matrix

    Find some correlation between

    features

Data Preprocessing

NA check
- Na values
- Zeros

Outliers check
- Pressure
- Wind speed

Impute NA values
- Last row imputation
- Linear imputation

Column drop
- Invalid columns
- Forecast columns (avoid data leakage)

Data merge
- Mean weather data
- City weather data (pivot wider)

Data shift
- Exclusively for traditional Regression model

Feature Engineering

Historical Values

price of an hour ago
price of the day before
price of a week ago
price of a month ago
…

Groupby Features

mean price of this week
mean price of this month
minimum price of this week
maximum price of this week
…

Slide Window

mean price of last 6 hours
mean price of last 7 days
mean temperature of last 6 hours
mean temperature of last 7 days
…

Trend

growth rate of price of the last hour
growth rate of price of this month
…

Date Features

day of month
day of week
quarter
hour
…

27 Original Features
+ 58 Generated Features

---

85 Features

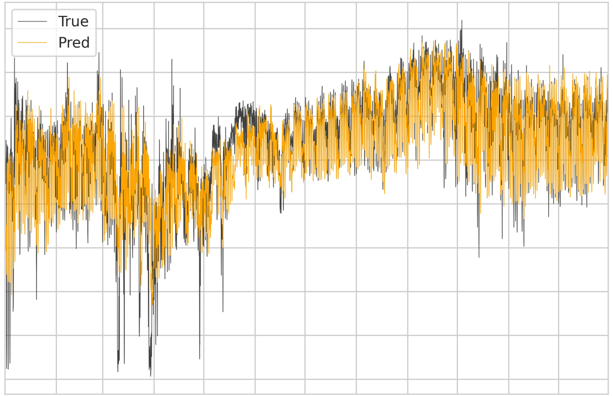
Machine Learning Models

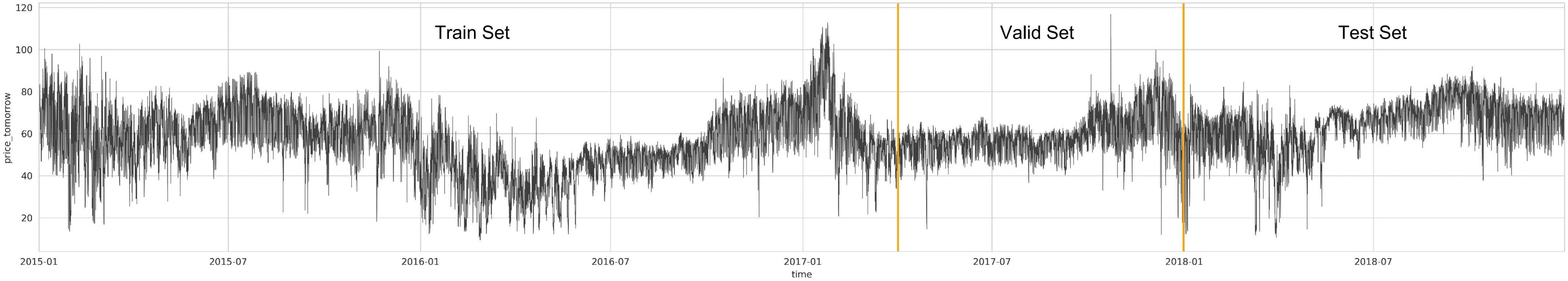
Model	RMSE
Linear Regressor	6.9827
Ridge Regressor	6.9807
Lasso Regressor	6.8987
Random Forest Regressor	6.4707
LightGBM	6.2770
XGBoost	6.3939
CatBoost	6.2158

avg

Final Prediction | RMSE **6.0847**

Prediction VS. Label





Machine Learning Models

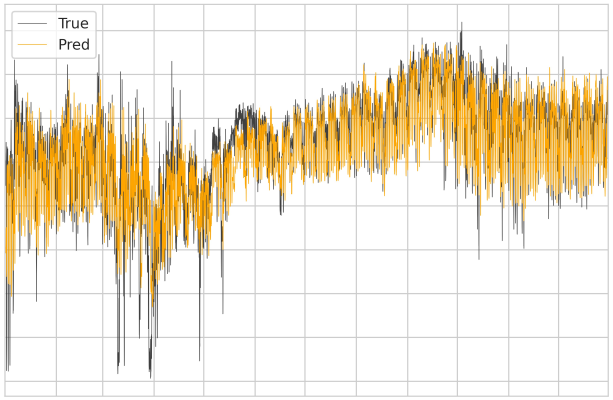
Model	Weight
Linear Regressor	0.125
Ridge Regressor	0.125
Lasso Regressor	0.125
Random Forest Regressor	0.125
LightGBM	0.15
XGBoost	0.15
CatBoost	0.2

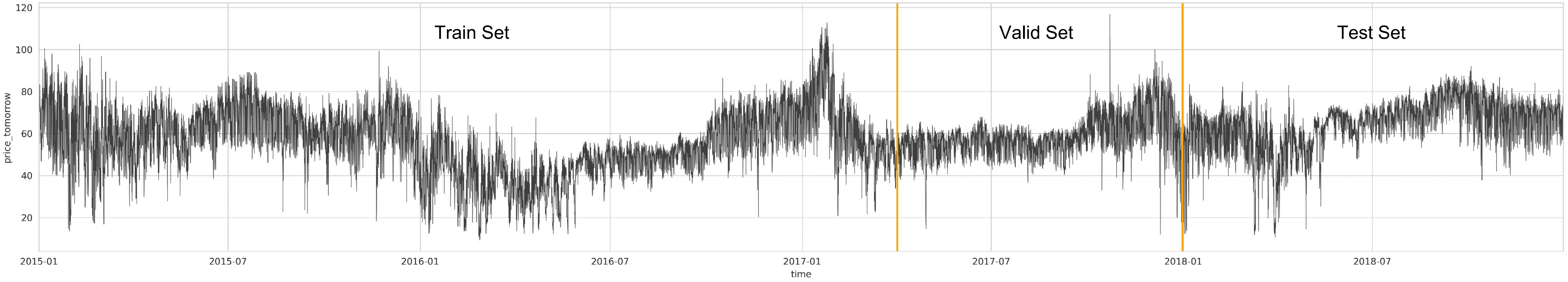
RMSE **6.0847**
RMSE **6.0187**

85 Features
\- **10** Low Importance Features

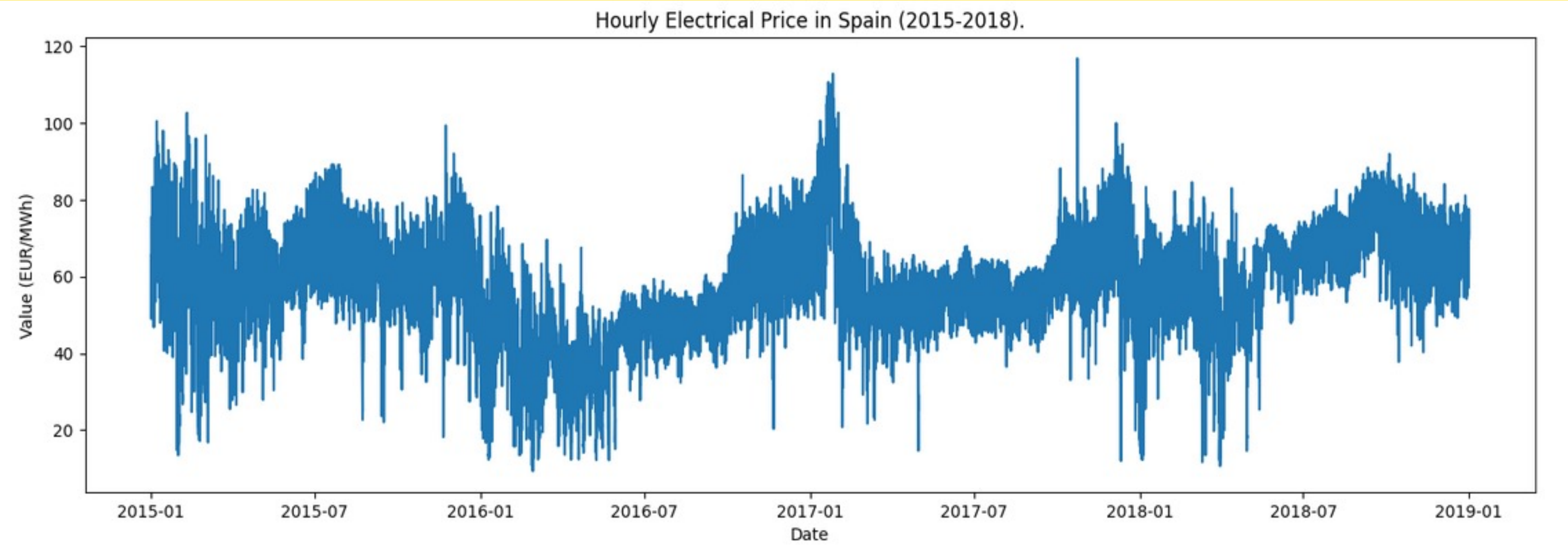
75 Features

Prediction VS. Label





Time Series Analysis



Distribution of Electrical Price over Time

Time Series Models

Time-Series data :

- is a collection of observations obtained through **repeated measurements** over **time**.
- Frequency can be Annually, Monthly, Weekly, Hourly or Second-wise.

Univariate Analysis:

•The series with a **single** time-dependent variable.

Ex.: AR model, ARIMA

Multivariate Analysis:

•The series has **more than one** time-dependent variable.

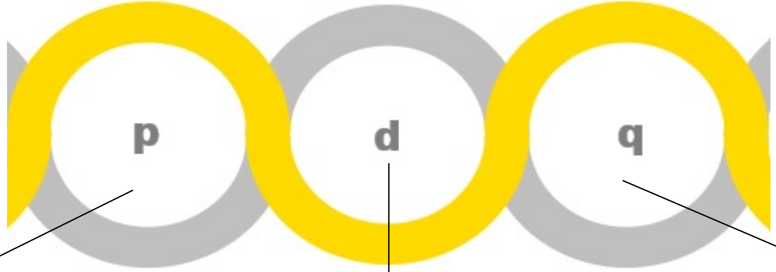
Ex.: VAR model, Prophet Multivariate model

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \varepsilon_t$$

$$\begin{aligned} Y_{1,t} &= \alpha_1 + \beta_{11,1} Y_{1,t-1} + \beta_{12,1} Y_{2,t-1} + \epsilon_{1,t} \\ Y_{2,t} &= \alpha_2 + \beta_{21,1} Y_{1,t-1} + \beta_{22,1} Y_{2,t-1} + \epsilon_{2,t} \end{aligned}$$

ARIMA Model

Auto Regressive – Integrated – Moving Average



- is the **order** of the **AR** term.
- **number of lags of Y** to be used as predictors.

- **minimum number of differencing** required to make the time series stationary

- is the order of the **MA** term.
- **number of lagged forecast errors** that should go into the model

Differencing to make Time Series Stationary.

Determine the (p,q,d) values. → Check for Stationarity of Time Series. → Determine the (p,d,q) values. → Fit the ARIMA Model.

ARIMA Model :

- Use **KPSS tests** to determine **stationarity of data**.
    - > **null hypothesis** : the series is stationary.
- determine **p** and **q** using **PACF** and **ACF** plots resp.
- **RMSE :14.15 for (1,0,0)**

- *Auto-ARIMA* eliminates the steps of making the series stationary and determine the values of **p** and **q** using the plots.

- **RMSE : 16.42**

```
Column generation hydro water reservoir: P_Values [0.0, 0.0, 0.0, 0.0]
Column generation nuclear: P_Values [0.0129, 0.0025, 0.0084, 0.0219]
Column generation other: P_Values [0.0059, 0.0001, 0.0001, 0.0]
Column generation other renewable: P_Values [0.3808, 0.0, 0.0, 0.0]
Column generation solar: P_Values [0.2346, 0.0, 0.0, 0.0]
Column generation waste: P_Values [0.0621, 0.0, 0.0, 0.0]
Column forecast solar day ahead: P_Values [0.2095, 0.0, 0.0, 0.0]
Column total load forecast: P_Values [0.0, 0.0, 0.0, 0.0]
Column total load actual: P_Values [0.0, 0.0, 0.0, 0.0]
Column price day ahead: P_Values [0.025, 0.0, 0.0, 0.0]
Column temp: P_Values [0.0021, 0.0, 0.0, 0.0]
Column temp_min: P_Values [0.0023, 0.0, 0.0, 0.0]
Column temp_max: P_Values [0.0021, 0.0, 0.0, 0.0]
Column humidity: P_Values [0.2146, 0.0, 0.0, 0.0]
Column wind_speed: P_Values [0.4165, 0.0, 0.0, 0.0]
```

```
KPSS Statistic: 0.015223089642741437
p-value: 0.1
num lags: 380
Critial Values:
   10% : 0.347
   5% : 0.463
   2.5% : 0.574
   1% : 0.739
Result: The series is stationary
```

```
stepwise_fit = auto_arima(df_train['price actual'], trace=True)
```

VAR :

- **Vectorized Auto-Regression**
- include other time series data along with target series.
- Use **Granger Causality Test**
    - **h0:** xt does not granger-causes yt
    - **h1:** xt granger-causes yt

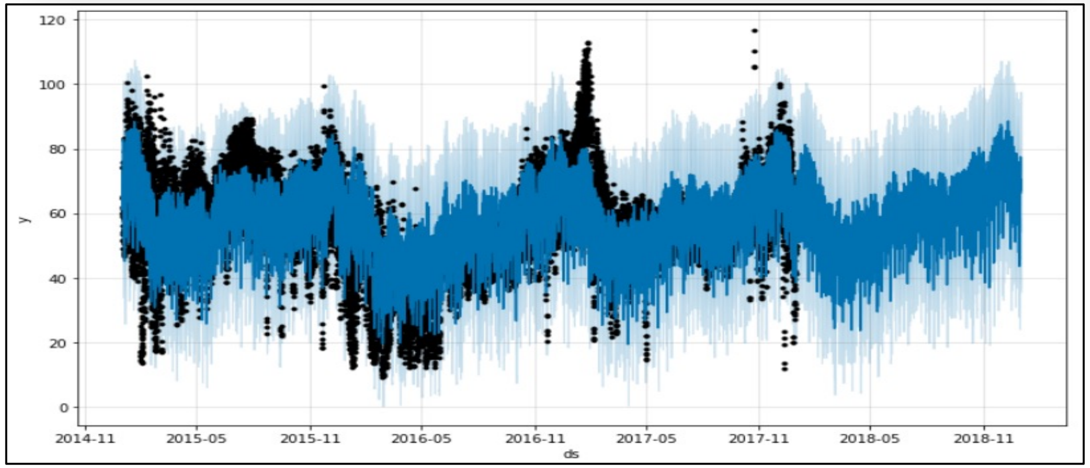
- **RMSE :** 14.18

Prophet

- is a procedure for forecasting time series data based on an additive model released by Facebook.

Univariate :

- provide the response variable to y parameter as well as to ds parameter.
- declare the future (period to predict) fit the model to find the prediction values in yhat.
- RMSE: **13.21**



Multivariate :

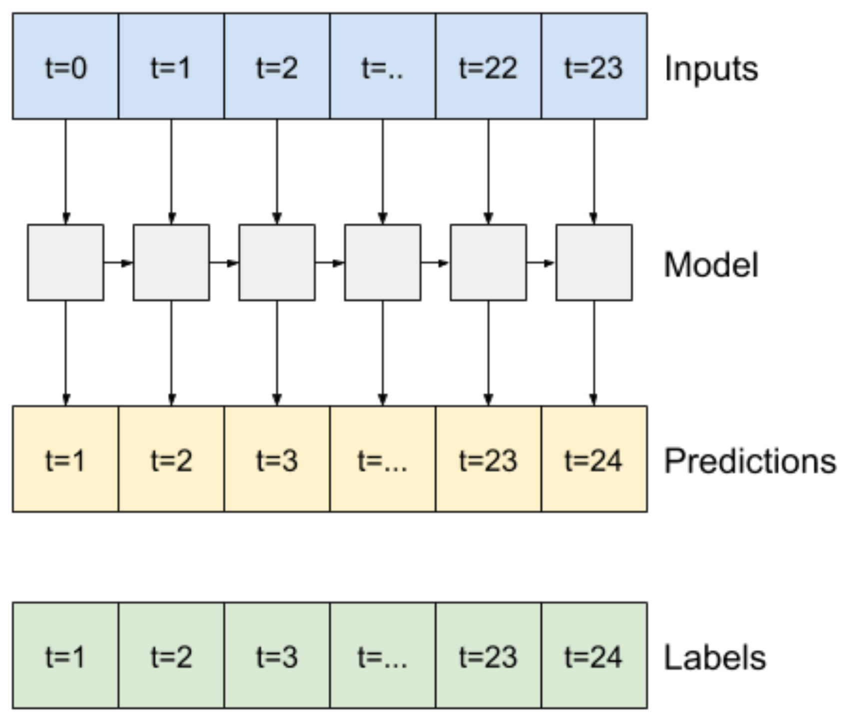
- the response variable to y
- provide other time series to series of ds parameters.
- add ds parameters as regressors to the model
- fit the model to find the prediction values in yhat.

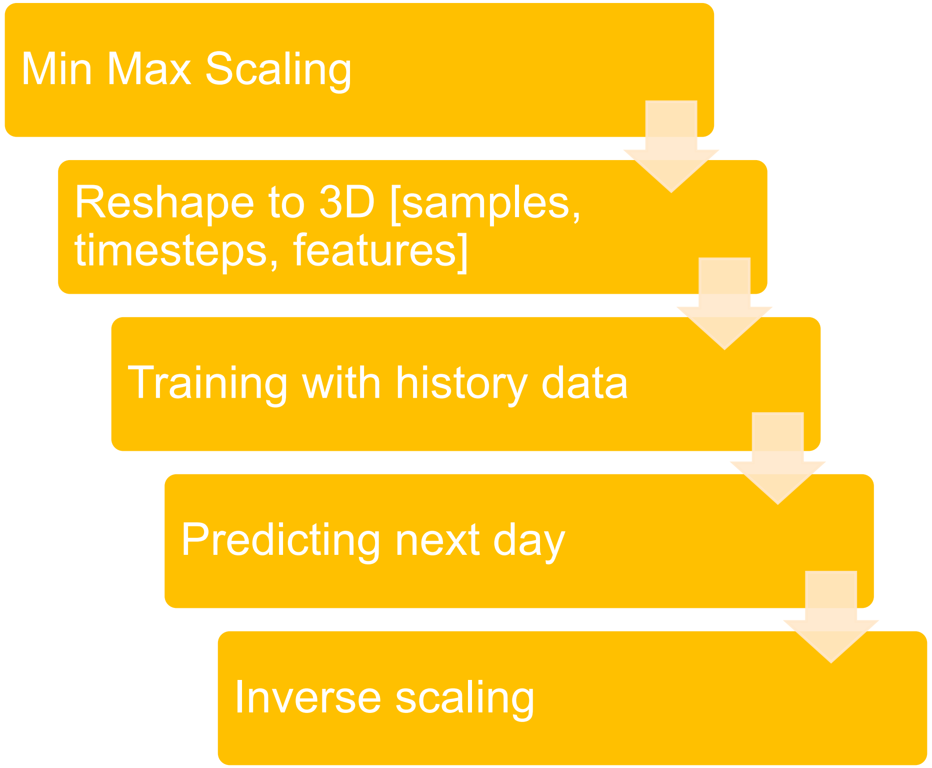
RMSE : **9.90**

```
m = Prophet(interval_width=0.95,yearly_seasonality=True)
```

Prediction of Price for year 2018

LSTM (Long Short-Term Memory) Model



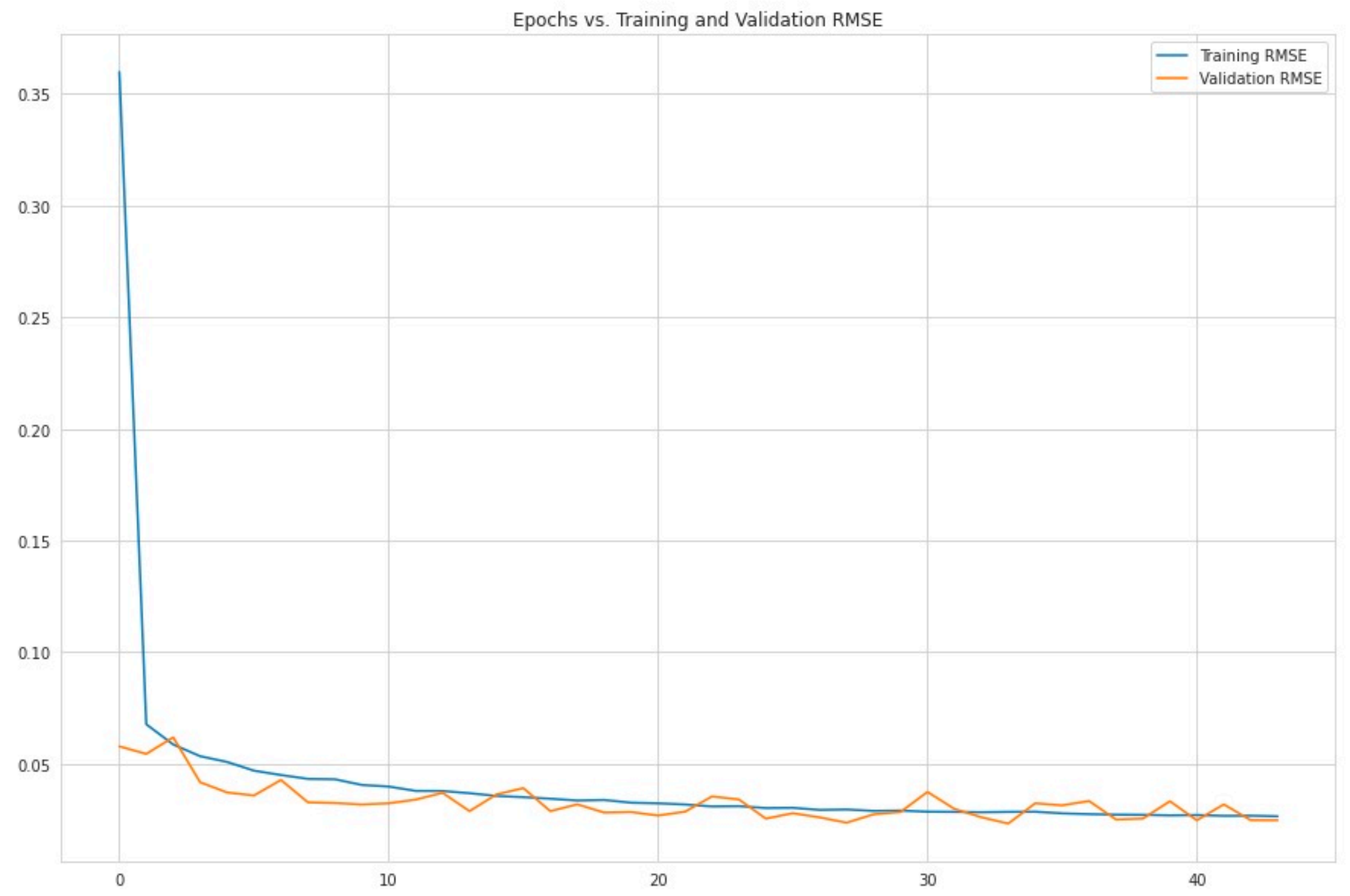


Resource: https://www.tensorflow.org/tutorials/structured_data/time_series#recurrent_neural_network

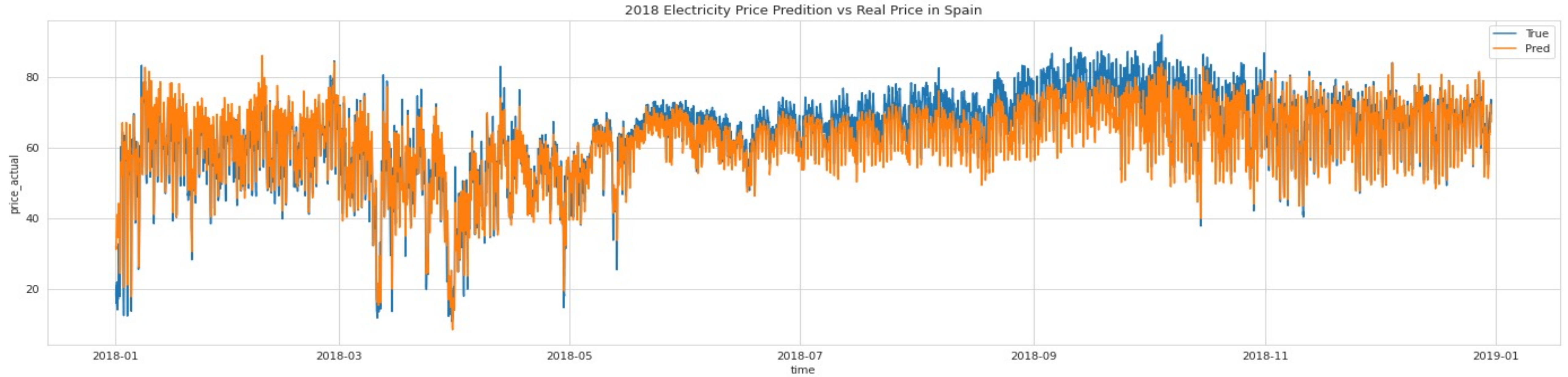
LSTM Model

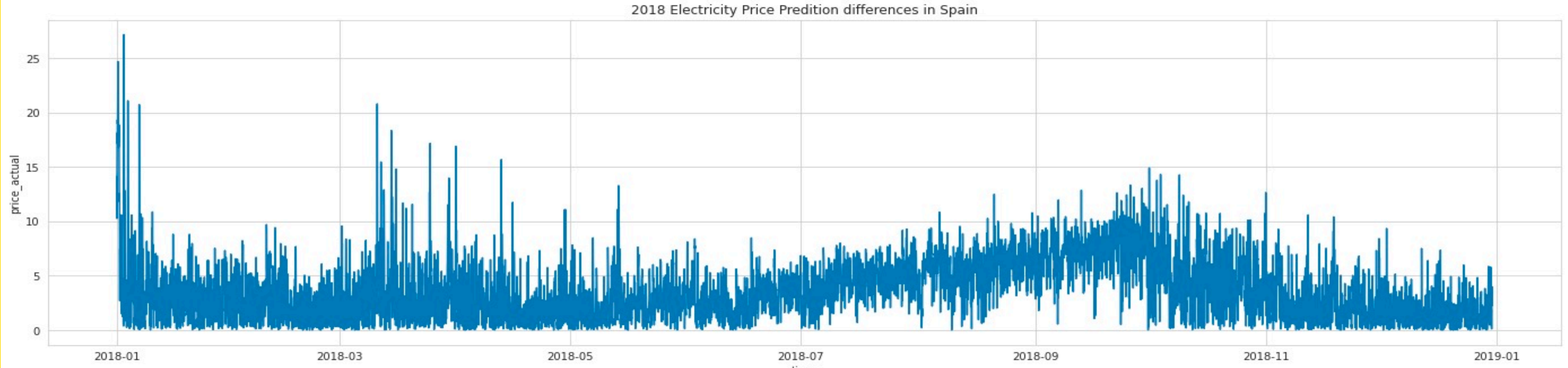
Parameters
Epoch: 50
layers: 5
Learning rate: 0.005
Activation: ‘relu’
Optimizer: ‘adam’
Train/valid: 3/1

Test Data RMSE:
4.4456



Prediction Analysis



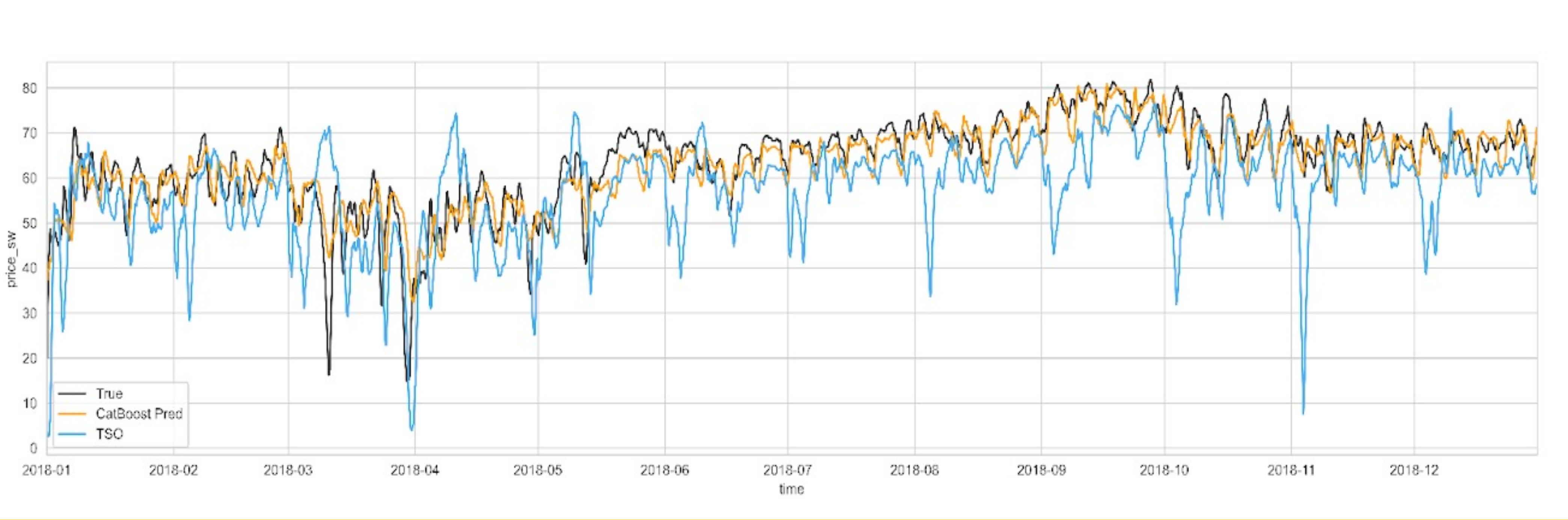


Model Results

Model	RMSE
Linear Regressor	6.982
Ridge Regressor	6.980
Lasso Regressor	6.89
Random Forest Regressor	6.47
LightGBM	6.27
XGBoost	6.39
CatBoost	6.21

Model	RMSE
AR	14.27
ARIMA	14.15
VAR	14.08
Prophet Uni.	13.21
Prophet Multi.	9.90
LSTM	4.45

Comparison with TSO

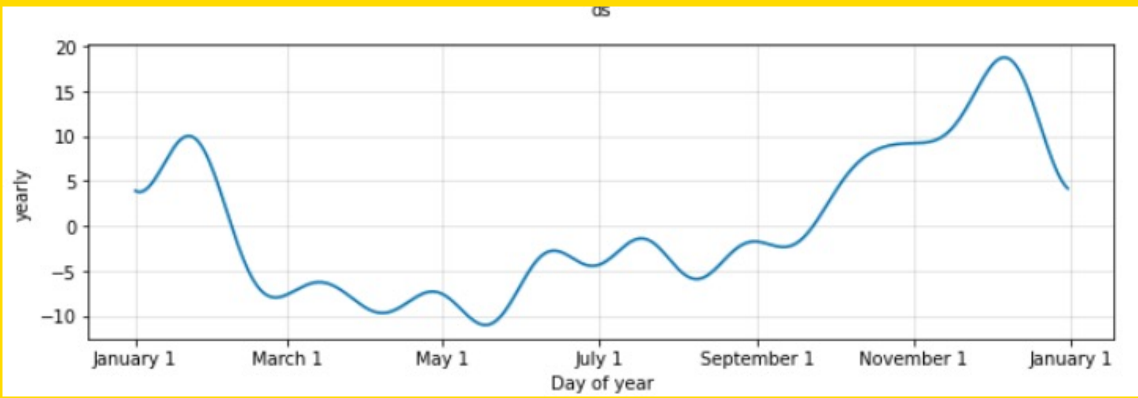


TSO Predictions & Obtained Predictions

When compared to Actual Price -

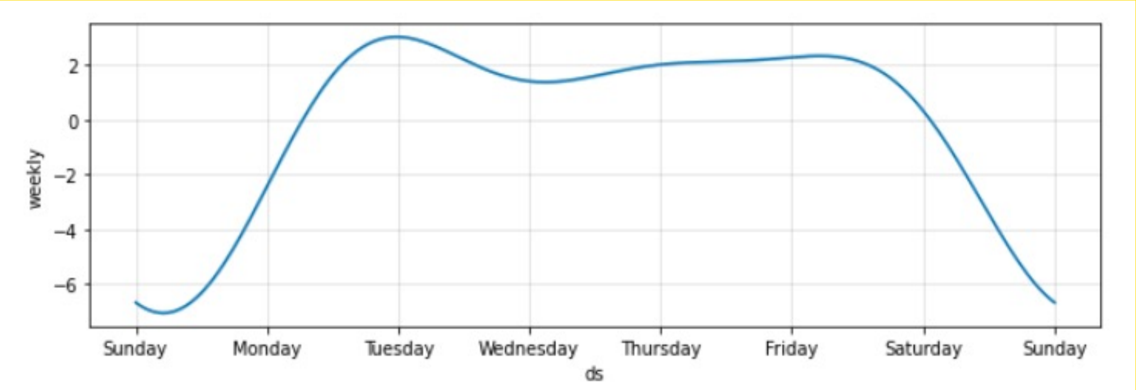
RMSE of TSO: **13.54**

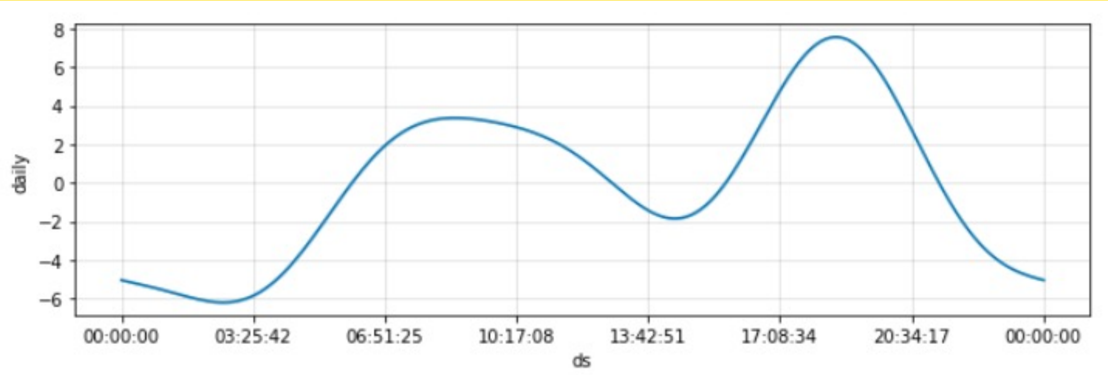
RMSE of our Predictions: **6.09**



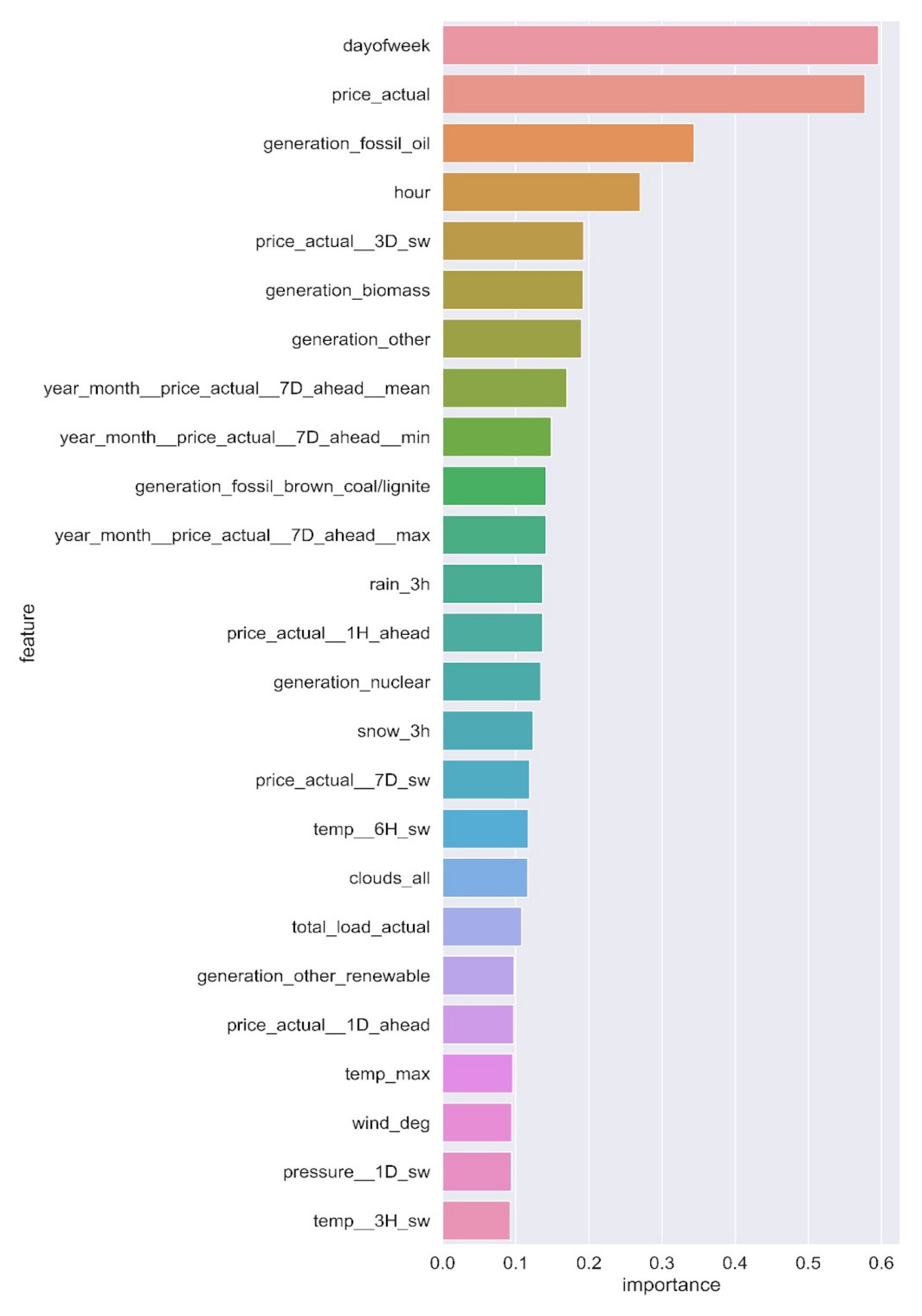
Yearly Component

Weekly Component





Daily Component



Importance of Features

THANK YOU