

**Faculty of Engineering**

**Department of Industrial Engineering**

**Fall 2021**

**IE 48B** – **Project Report**

**GROUP 8**

**2017402177**

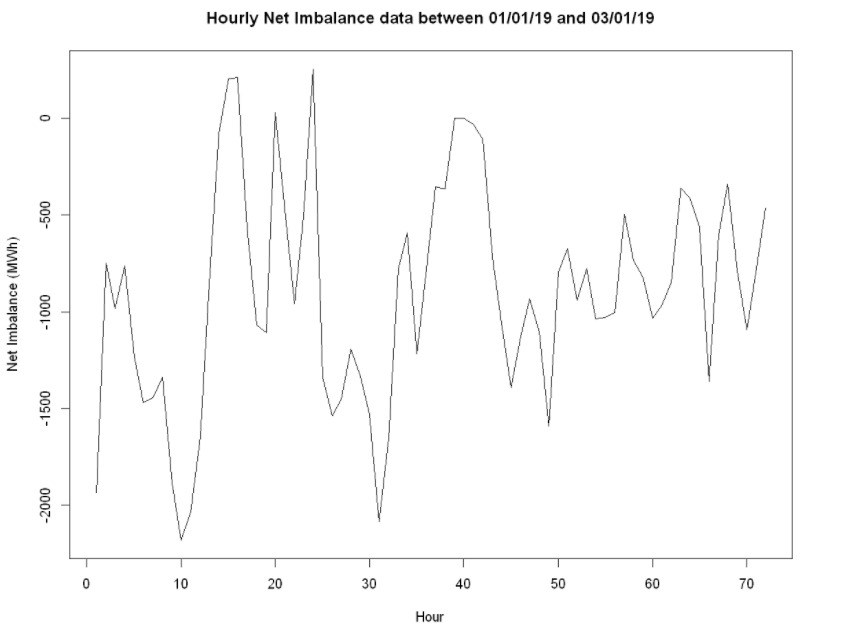
**2017402198**

**2017402264**

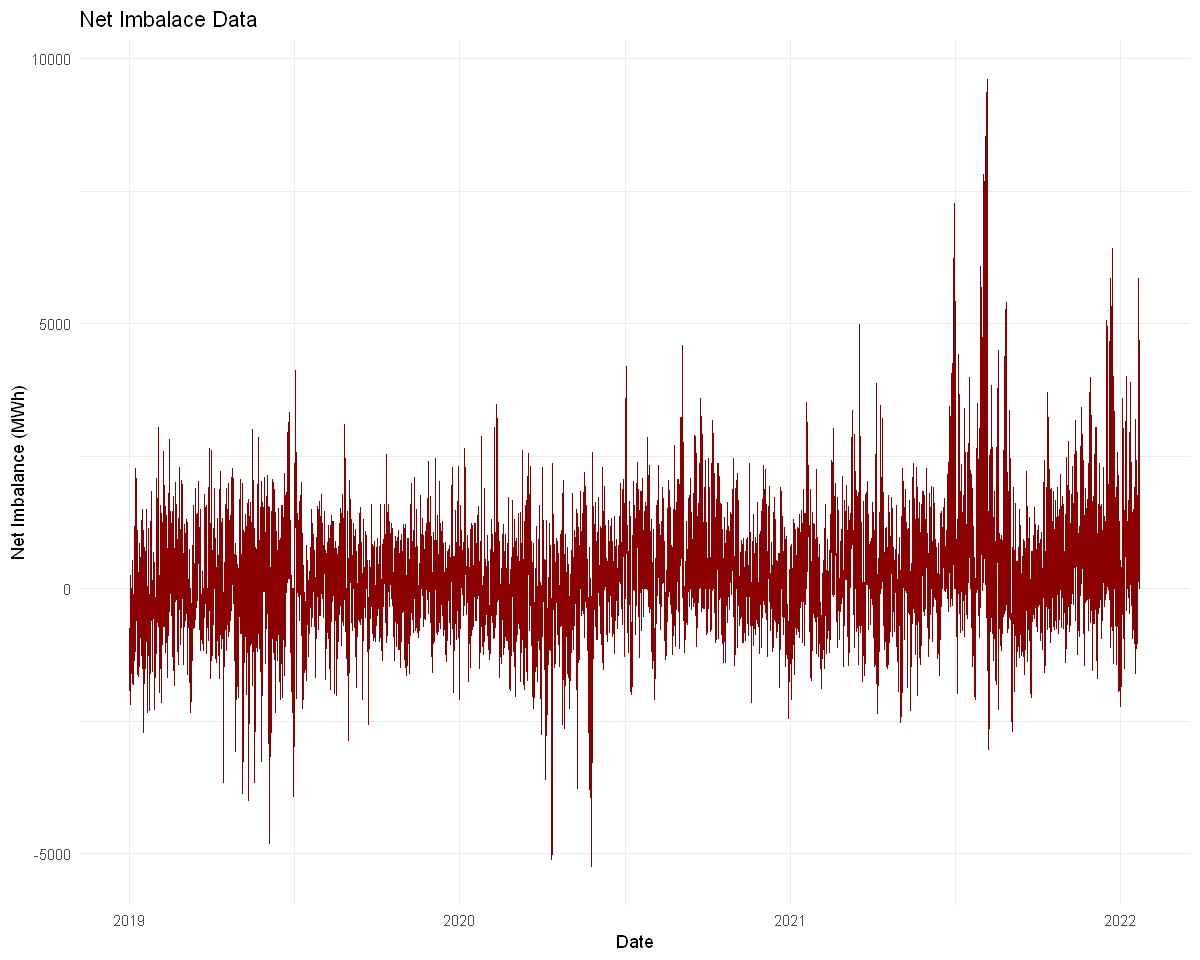
**1 Introduction**

In this project, the main task is to develop an approach for predicting the sign of imbalance for Turkish electricity market (MWh) from 9 January 2022 to 21 January 2022. Every day, the predictions consist of the intraday sign of imbalance from 12th hour to 23rd hour. Up to previous day’s data will be used for predicting the present day’s sign of imbalance.

In addition to the past net imbalance data, the hourly weather information data belonging to seven different locations close to the big cities in Turkey are also provided. These cities are Antalya, Adana, Konya, İzmir, Eskişehir, Ankara and İstanbul respectively. Weather information contains downward shortwave radiation at the surface, total cloud cover at the low cloud layer, temperature at 2 meters altitude, wind speed at 10 meters altitude, and wind direction in 10 meters altitude, relative humidity at 2 meters altitude. The best point to start is analysing the data visually. Here is the plot containing three consecutive days:



From the plot, we can see that there is a weak seasonality in the data. Some hours have a similar behavior but mostly data seems to have high variance. Also, there can be seen that there is a trend component.



As mentioned above, seasonality is weak that can be seen from the plot of net imbalance data. Opposite to the 3-day plot above, there is no trend in net imbalance data. There are some outliers in summer 2021. It will be analyzed later in the report. To use weather information effectively, mean of the seven big cities’ weather information data is used.

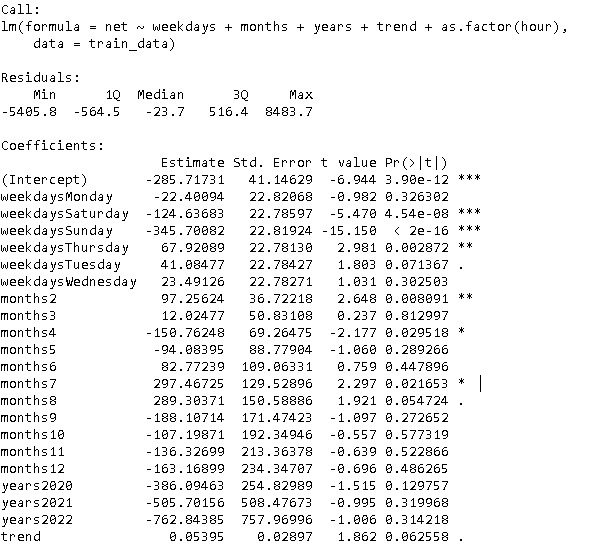
# 2 Related Literature

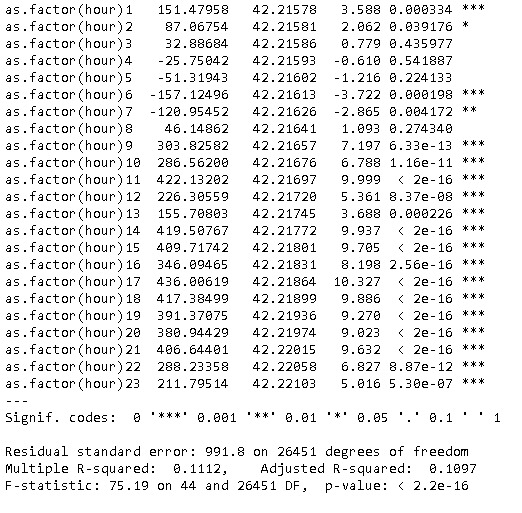
# For the explainable boosted linear regression method. Lecture material is used.

# EBLR generates nonlinear features that are also interpretable. Each generated feature corresponds to a set of decision rules. Propose a method called EBLR for time series forecasting. Iteratively updates the forecasts by generating simple binary. features through a two-step process based on regression trees. Extended to probabilistic forecasting by utilizing the empirical distribution of the residuals. Competitive forecasting performance without sacrificing “simplicity and interpretability.

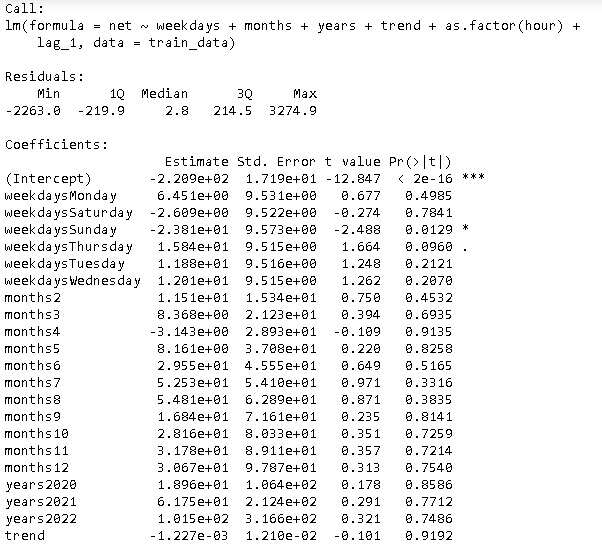
# 3 Approach

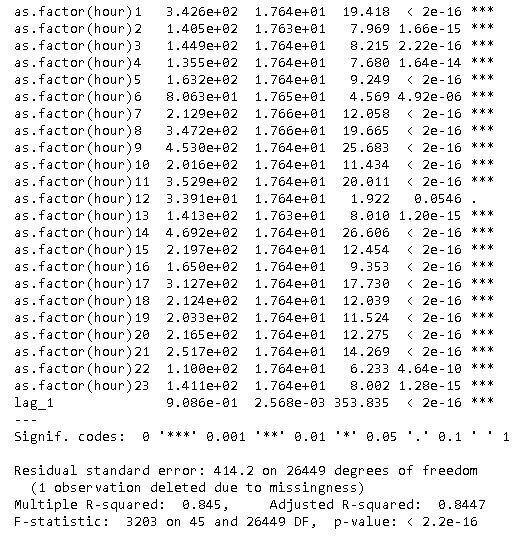
In this section, one base model will be established with only day type, month regressors, trend factor and hour regressors.





Hourly regressors seems to be very effective. Trend and month regressors have a weak performance. Overall, this model is not sufficient for predicting the net imbalance signs. Lag values will be added to this model.





With the addition of the lag value. Performance of our base model is increased significantly in terms of R squared and residual standard error. We will use explainable boosted linear regression with this base model.

Here is a demonstration of one explainable boosted linear regression iteration.

Diagram, engineering drawing, schematic

Description automatically generated

A decision tree is fitted on the residuals of the base linear regression model with the all the existed features. The path which leads to highest average value is chosen and dummy variables corresponding to that path is created. Those dummy variables are added to the model at every iteration until the improvement decreased to a significantly lower level.

After 5 iterations of explainable boosted linear regression, final model is decided.

Text

Description automatically generated

**Outlier Analysis**

As mentioned in the previous section, there was some outliers in the in the hourly imbalance data. We decided on a threshold value and created some dummy variables to exclude outlier points.

Here is our final model with the outlier analysis. This model will be used for the prediction purpose.

Text

Description automatically generated

# 4 Results

# Here are the daily results of our model compared to previous day and previous week naïve forecast.

# Table Description automatically generated

# As can be seen from the results our model has a much better performance than two others. Overall result is here:

# Table Description automatically generated with medium confidence

**5 Conclusion and Future Work**

Even though the final model has a decent accuracy, further improvements and extensions can be made in the model to have a better approach.

Firstly, we used lag 1 values in our model but since we don’t have the lag 1 values, predictions are used as lag 1 values. Usage of lag 1 may not the best approach.

We had some external variables for 7 big cities. For simplicity the mean of the 7 cities is used but there could be a better approach for the usage of those external variables.

Day, year, trend, and month regressors were not much effective in the final model. It could be possible to get some useful information from those regressors with some representations or extensive analysis.

There were some important outliers in 2021 summer, In final model this outlier period is excluded from model but there could be better analysis for this outlier period.

**6 Appendices**

* The R Markdown of this report is [here](https://bu-ie-360.github.io/fall20-araldortogul/files/R-scripts/IE360_group12_project_report.Rmd). (.Rmd file)
* The R script used for retrieving data from API and making submissions is [here](https://bu-ie-360.github.io/fall20-araldortogul/files/R-scripts/IE360_group12_project.R). (.R file)
* The data (.csv and .xlsx files) used in these files are [here](https://bu-ie-360.github.io/fall20-araldortogul/files/IE360_group12_project_data.zip). (.zip file)

**7 References**

[Ilic, I., Görgülü, B., Cevik, M., & Baydoğan, M. G. (2021). Explainable boosted linear regression for time series forecasting. Pattern Recognition, 120, 108144.](https://www.sciencedirect.com/science/article/pii/S0031320321003319?casa_token=vTwhYWV2nh4AAAAA:ct7Qjv8AAS0Pq20S9VRtfIBMXdvaioB0SHb-9YFqx6iZR38T65L0g0V9f2cbbAy9dmRBFYfSgg)