Appliance Energy Prediction Project

CAPSTONE PROJECT - III

Made By: SANKET CHOURIYA

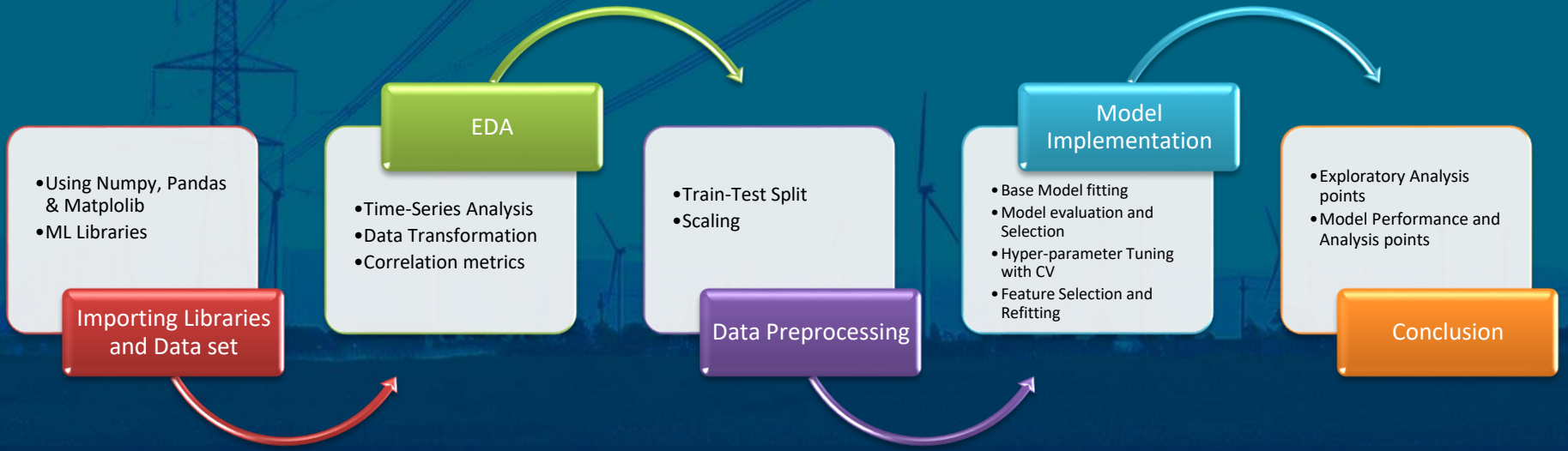
TABLE OF CONTENT

INTRODUCTION

The understanding of the appliances energy use in buildings has been the subject of numerous research studies, since appliances represent a significant portion of the electrical energy demand. By Using different data sources and environmental parameters (indoor and outdoor conditions), specifically, data from a nearby weather station, temperature and humidity in different rooms in the house from a wireless sensor network and one sub-metered electrical energy consumption (lights) have been calculated.

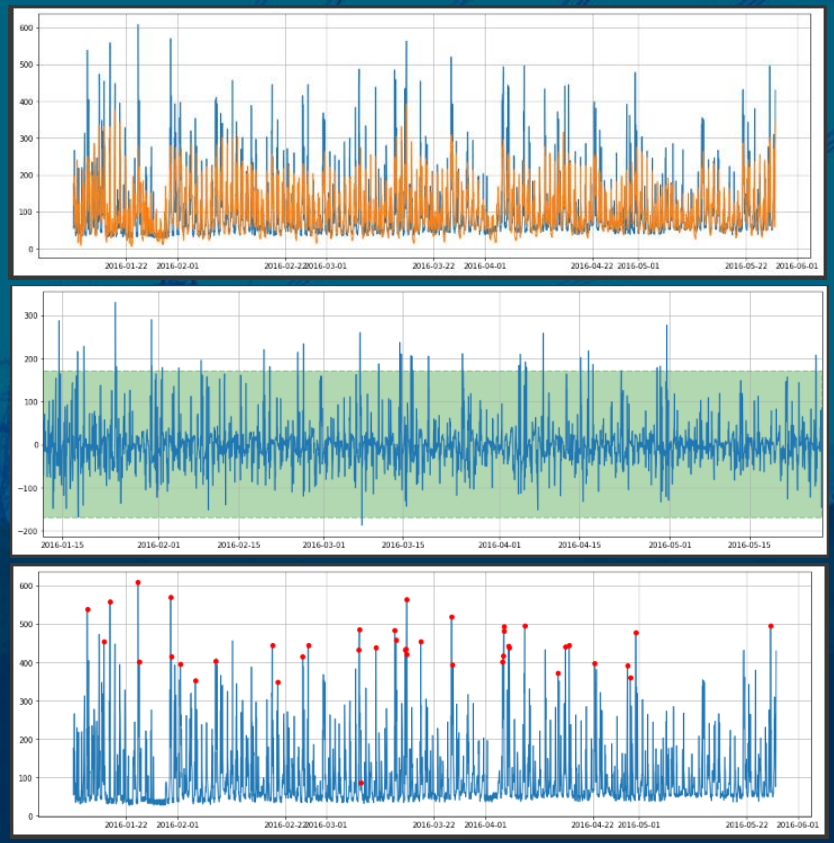
The goal of our project is to predict the energy consumption of appliances in households based on the sensor data that we have from a random apartment and corresponding weather reports with help of Machine Learning Algorithms.

PROCESS FLOW CHART



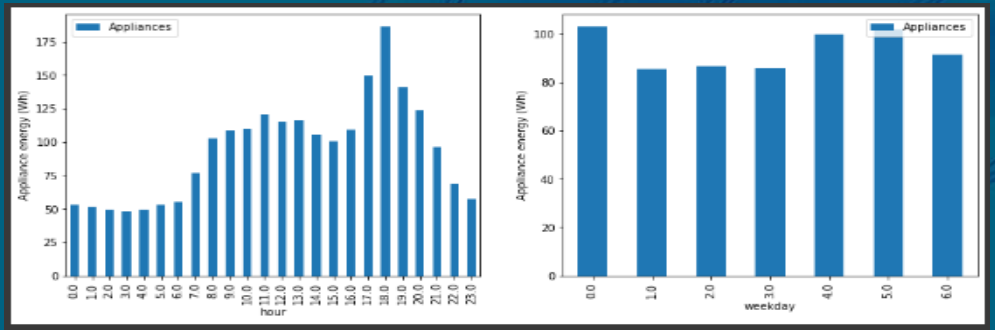
EXPLORATORY ANALYSIS

Time-Series Analysis on Target Variable

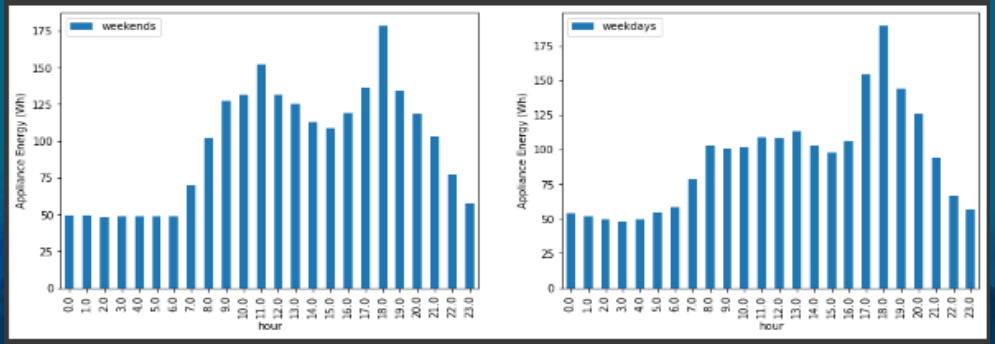


As visualized above, Outliers are considered as the less than 5% of top values of appliances' load because it is fact that recordings of power average load higher than 400Wh per hour from a house appliance are not logical. So, the main idea of using time series in anomaly detection is to calculate the mean and standard deviation of the Residuals component of the decomposition and exclude all point for which residuals differs from the average by more than 3.5 times the Std_dev. Thus, We will remove these marked outlier to reduce Noise.

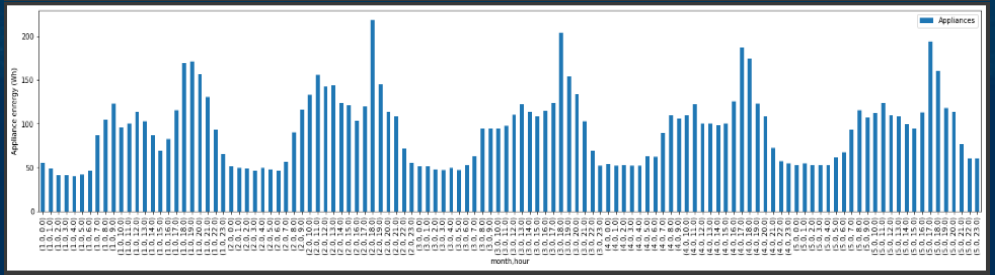
```
No of Outliers removed:  42
Portion of Outlier data removed:  1.28
No. of Non-Outliers:  3248
Portion of Good data:  98.71
```



Highest consumption Hour wise: **5PM - 7 PM**
Peak consumption at hour: **6PM**
Highest consumption day wise: **Mon & Fri**



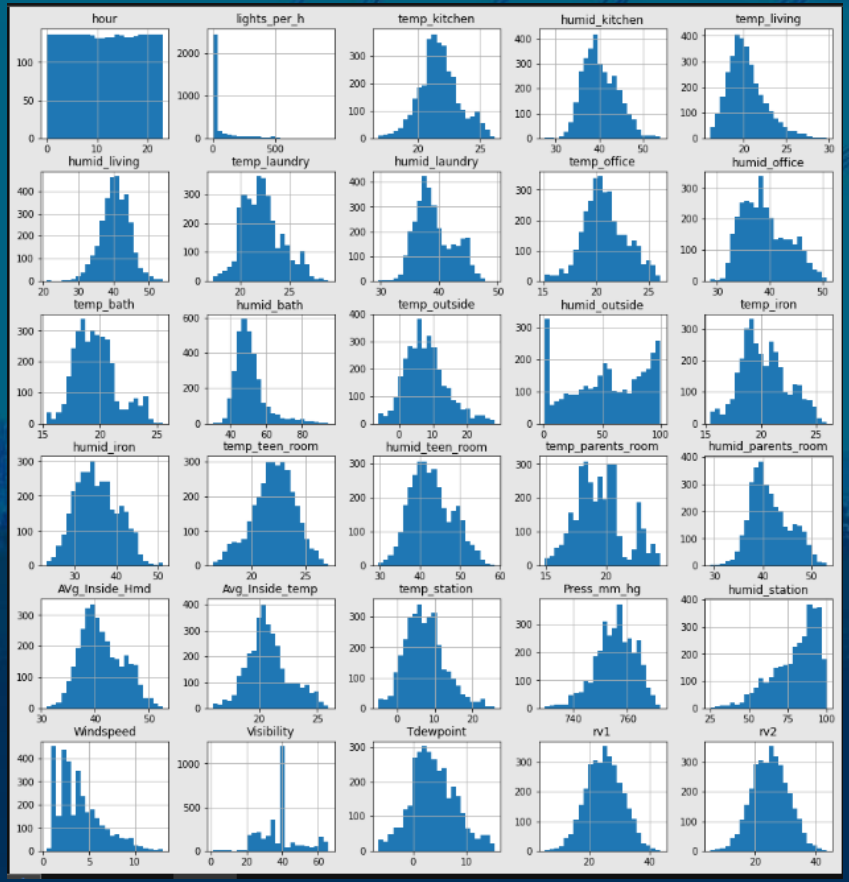
We observe that the energy consumption of appliances during the office hours **(8 AM - 4 PM)** is higher in weekends compared to the weekdays. Also, average overall consumption is higher in weekends is pretty high.



Monthly consumption graph pattern resembles similar traits with Average Hourly consumption plot.

Lowest consumption: **Lowest at 50Wh in sleeping hours (11 PM - 6 AM)**
Highest Consumption: **Above 100 Wh in Evening (5PM - 10 PM)**

DISTRIBUTION OF FEATURE VARIABLES

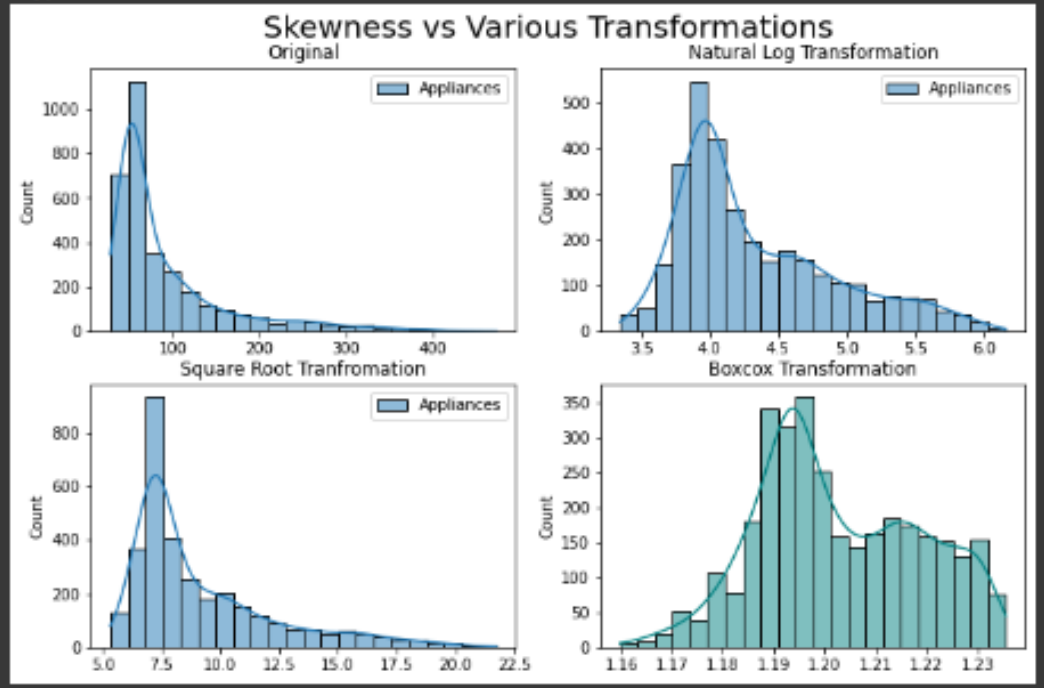


Observations based on distribution plot:

- All features values except 'lights_per_h' ,'humid_outside' and 'Visibility' follow a Normal distribution, i.e., Almost all the readings from sensors inside the home are from a Normal distribution.
- Similarly, all temperature readings follow a Normal distribution except for 'temp_parents_room' which is sort of unstable near tail end.
- Out of the remaining columns, we can see that humid_outside,temp_teen_room, humid_living, Press_mm_hg and humid_station are strong negatively skewed whereas 'lights_per_h' ,'humid_bath' and wind speed is strong positively skewed.
- The random variables rv1 and rv2 have more or less the same values for all the recordings.

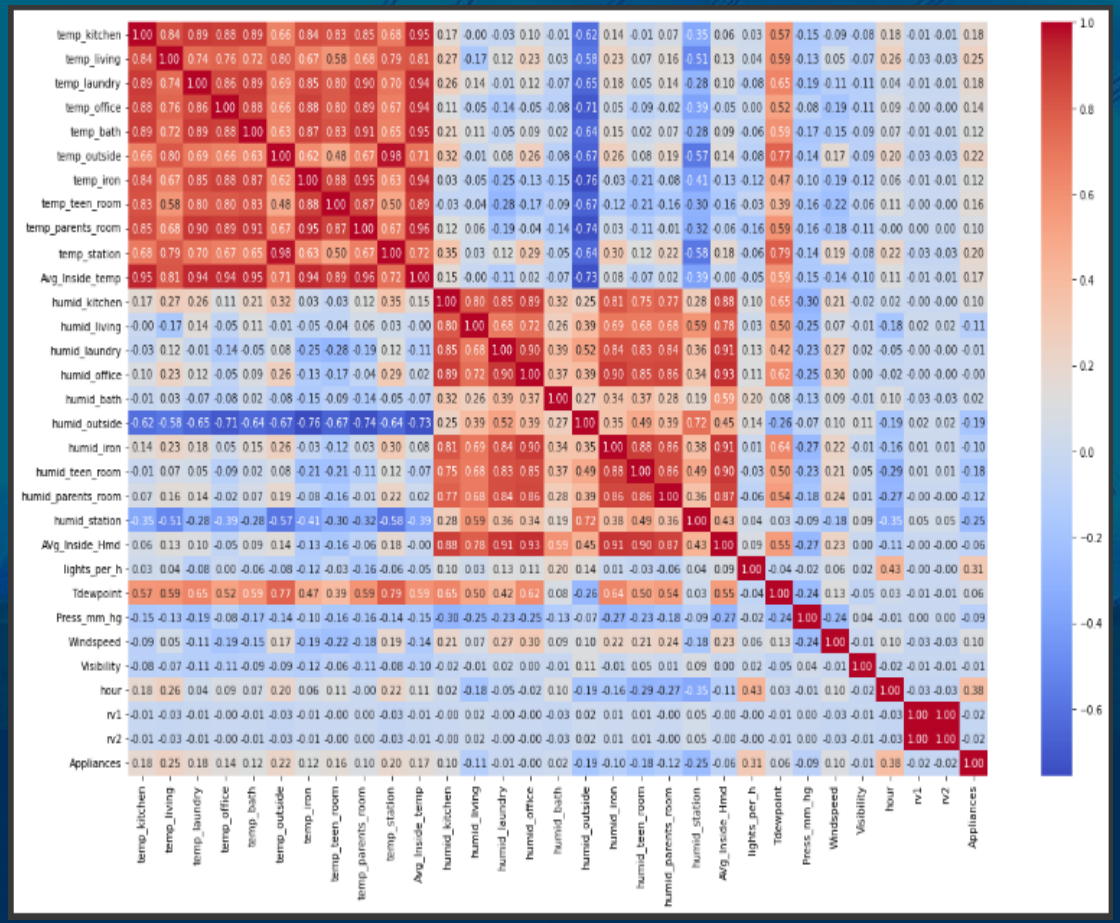
DISTRIBUTION OF TARGET VARIABLE

```
Appliances    2.126806
dtype: float64
```



The distribution is right skewed and majority of appliances uses less than 250 Wh of energy. With the maximum consumption of 1080 Wh , there will be outliers in this column and there are small number of cases where consumption is very high.

After applying Log, Square root and Reciprocal Transformation we can see that the Log transformation works the best to remove the skewness of the Target Variable.



CORRELATION MATRIX

OBSERVATIONS:

- From the correlation graph we clearly observe that the features related to temperature and features related to humidity have positive correlation within themselves whereas have a very little to no correlation with each other.
- Four columns have a high degree of correlation with temp_parent_room - temp_laundry, temp_bath, temp_iron, temp_teen_room also temp_outside & temp_station has high correlation (both temperatures from outside).
- Tdewpoint shows a high correlation with most of the inside temperature and humidity level features than any other weather parameters. Pressure, wind speed and visibility show little to no correlation.
- Features like 'rv1', 'rv2', 'month' and 'week_type' are safe to remove.

MODEL IMPLIMENTATION

DATA PREPROCESSING

TRAIN-TEST SPLIT: It is used to estimate the performance of machine learning algorithms when they are used to make predictions on data not used to train the model. Ratio of **80:20** has been kept for train and test set respectively.

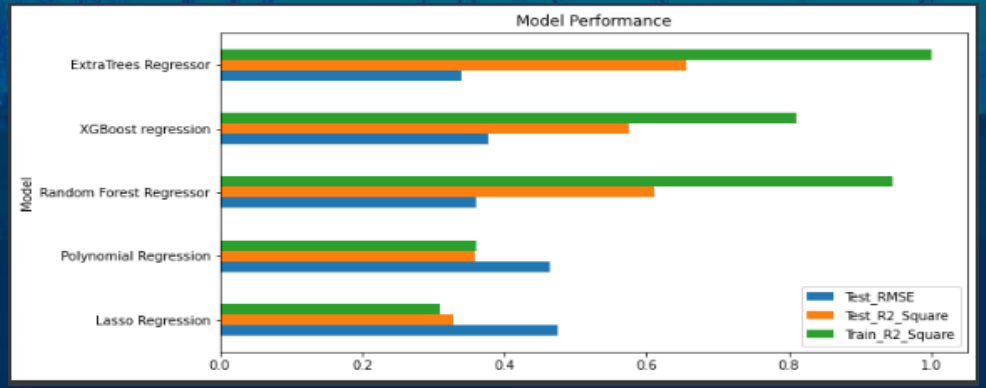
SCALING: The feature set has data in varying ranges . Temperature(-5 to 26.1) , Humidity (1–100) , Windspeed (0 to 14), Pressure (729–772) and Application Energy Usage (log Transformed, 28.3 - 563.3). Due to different ranges of features, it is possible that some features will dominate the Regression algorithm. To avoid this situation, all features need to be scaled. Thus, the data was scaled to 0 mean and unit variance using the **StandardScaler** class in sklearn.preprocessing module.

EVALUATION MATRICS: Following are the main evaluation metrics used for models,

- **Root Mean Squared Error (RMSE)** – It is a standard way to measure the error of a model in predicting quantitative data.
- **R2** - Compares our model with the baseline model. It is the proportion of the variation in the dependent variable that is predictable from the independent variable(s).

MODEL EVALUATION & SELECTION

	Model	Test_RMSE	Test_R2_Square	Train_RMSE	Train_R2_Square
0	Lasso Regression	0.474316	0.327783	4.847785e-01	0.309533
1	Polynomial Regression	0.463741	0.357423	4.667119e-01	0.360039
2	Random Forest Regressor	0.360689	0.611278	1.372019e-01	0.944693
3	XGBoost regression	0.377674	0.573804	2.546554e-01	0.809471
4	ExtraTrees Regressor	0.339331	0.655950	6.134726e-15	1.000000



Observations

- Best results over test set are given by ExtraTrees Regressor with R2 score of 0.655 while Random forest regressor holds the second place with 0.611
- For ETR, we can see that the Train R2 score is much greater than Test R2 score which can be due to OVERFITTING on the training data.
- Least Test RMSE score is also by ExtraTrees Regressor of 0.339
- Lasso regularization over Linear regression was worst performing model with R2 Score of 0.327
- For our final model, I have selected ExtraTrees Regressor for tuning.

MODEL VISUAL COMPARISON

From above scatter plot, Its visible that the Extra Tree Regressor has the most linear plot compared to others which indicate less outliers.

HYPER-PARAMETER TUNING

Instead of GridSearchCV, I have decided to go with RandomizedSearchCV for hyper parameter tuning of our Base Model due to processing time.

GridSearchCV is basically considering all the combinations of the candidates in finding the best parameters. This would in turn take a very long time when there are a greater number of parameter and their values to tune.

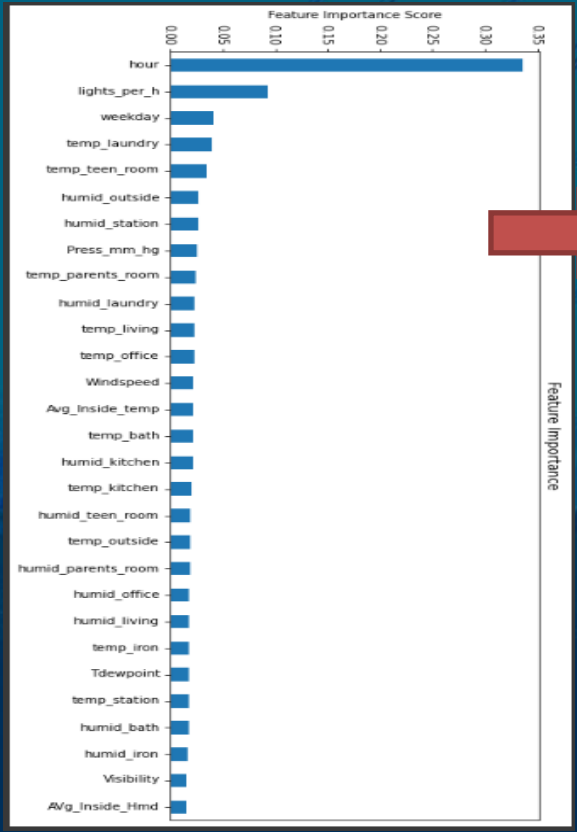
While, RandomizedSearchCV implements a “fit” and a “score” method and randomized search on hyper parameters.

```
ExtraTreesRegressor(max_depth=225, min_samples_leaf=2, n_estimators=600,
                    random_state=0)

Average Error       :  0.57181 degrees
Variance score R^2  :  65.52 %
RMSE                :  0.33971
Accuracy            :  86.93 %
_____________________________________
```

Even after parameter tuning there a very little to no improvement observed in the model.

FEATURE IMPORTANCE

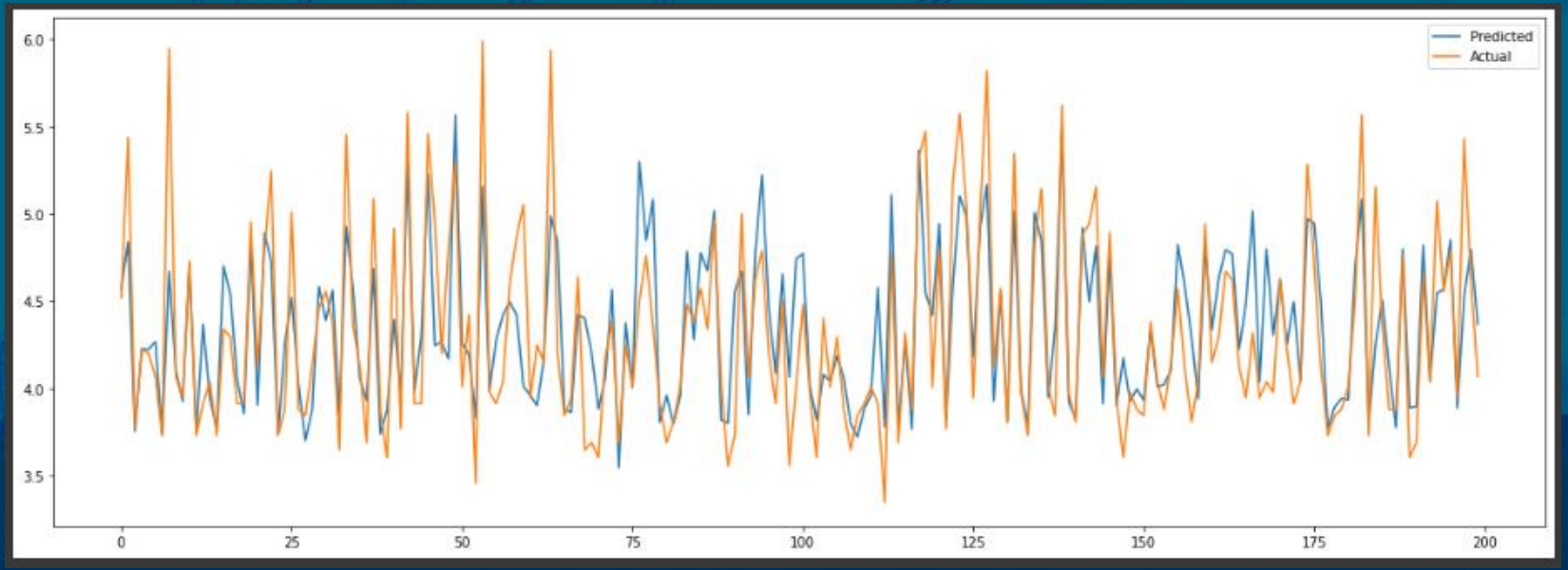



```
Average Error        :  0.57218 degrees
Variance score R^2   :  62.38 %
RMSE                 :  0.35485
Accuracy             :  86.93 %
_____________________________________
```

Observations

- Performed Feature selection on the basis of importance in the predictive relationship with the target variable. Top 10 feature where selected for the model and yet the model performance decrease by ~3% compared to Tuned model.
- As per the chart, I picked top most important features like: **'hour','lights_per_h','temp_parents_room','temp_laundry','temp_teen_room', 'Press_mm_hg','humid_laundry','humid_bath','temp_bath'** and **'temp_office'.**

PREDICTABILITY



CONCLUTION

- The best Algorithm to use for this dataset Extra Trees Regressor.
- Although, Extra trees regressor over-fits our training data with a R2 score of 1, it also gives, by far, the best performance on test set as well compared to other models, with a R2 score of 0.653
- Hyper parameter tuning of our model doesn’t have any significant impact on our test results. We were able to improve the test R2 score results by less than 0.01
- The final model had 22 features.
- Feature reduction was not able to add to better R2 score.


Thanks You!