Capstone Project - 2

Appliances Energy Prediction

Gulzar

Table of content

- Importing the Relevant Libraries
- Data Inspection
- Exploratory Data Analysis
- Feature selection
- Feature engineering
- Train and Test split
- Model training
- Cross validation
- Model Explainability
- Conclusion

Problem Statement

The data set is at 10 min for about 4.5 months. The house temperature and humidity conditions were monitored with a ZigBee wireless sensor network. Each wireless node transmitted the temperature and humidity conditions around 3.3 min. Then, the wireless data was averaged for 10 minutes periods. The energy data was logged every 10 minutes with mbus energy meters. Weather from the nearest airport weather station (Chievres Airport, Belgium) was downloaded from a public data set from Reliable Prognosis (rp5.ru) and merged together with the experimental data sets using the date and time column. Two random variables have been included in the data set for testing the regression models and to filter out non-predictive attributes (parameters).

Objective

Energy usage is rapidly increasing in today's world. We are experiencing a lack of energy due to the increased energy consumption in some regions of the world, which is causing environmental damage. Our main purpose in this project is to analyse what factors are affecting the increase energy consumption of home appliances, how we may reduce energy consumption of home appliances, and predict energy consumption of appliances using regression models.

Data set information

Columns Used:

1. Date time year-month-day hour:minute:second
2. Appliances, energy use in Wh
3. Lights, energy use of light fixtures in the house in Wh
4. T1, Temperature in kitchen area, in Celsius
5. RH_1, Humidity in kitchen area, in %
6. T2, Temperature in living room area, in Celsius
7. RH_2, Humidity in living room area, in %
8. T3, Temperature in laundry room area
9. RH_3, Humidity in laundry room area, in %
10. T4, Temperature in office room, in Celsius
11. RH_4, Humidity in office room, in %
12. T5, Temperature in bathroom, in Celsius
13. RH_5, Humidity in bathroom, in %
14. T6, Temperature outside the building (north side), in Celsius
15. RH_6, Humidity outside the building (north side), in %

16. T7, Temperature in ironing room , in Celsius
17. RH_7, Humidity in ironing room, in %
18. T8, Temperature in teenager room 2, in Celsius
19. RH_8, Humidity in teenager room 2, in %
20. T9, Temperature in parents room, in Celsius
21. RH_9, Humidity in parents room, in %
22. To, Temperature outside (from Chievres weather station), in Celsius
23. Pressure (from Chievres weather station), in mm Hg
24. RH_out, Humidity outside (from Chievres weather station), in %
25. Wind speed (from Chievres weather station), in m/s
26. Visibility (from Chievres weather station), in km
27. Tdewpoint (from Chievres weather station), Â°C
28. rv1, Random variable 1, nondimensional
29. rv2, Random variable 2, nondimensional

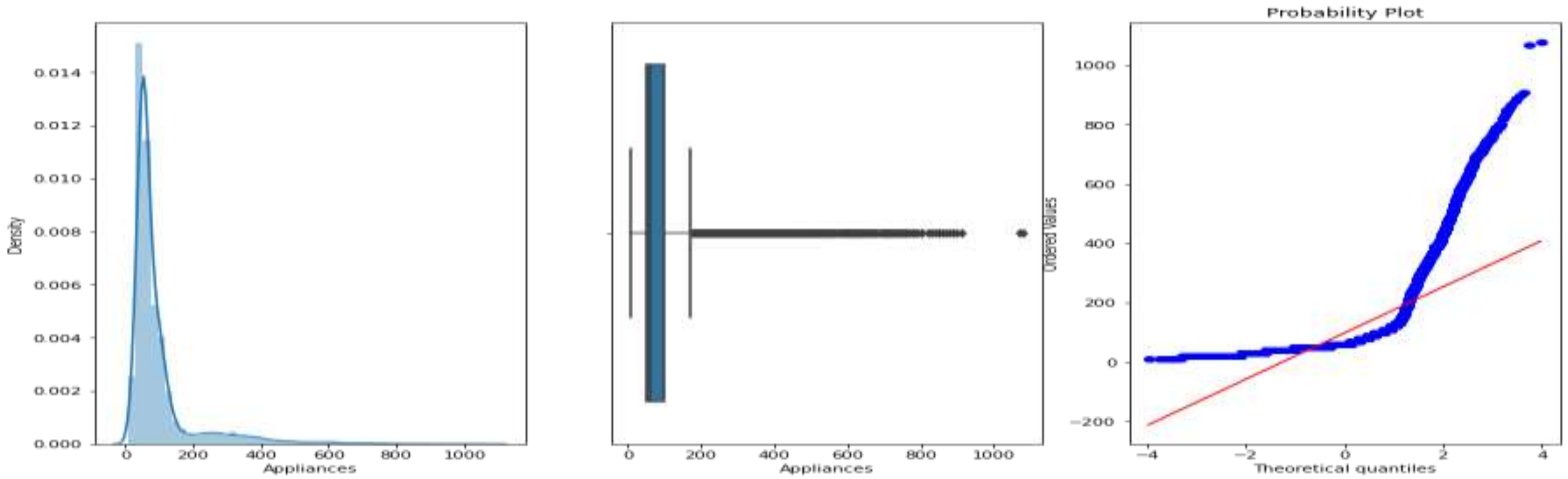
Data Inspection

From the statistics part of our data we can observe:

- There are 29 columns and 19735 rows in our dataset.
- The maximum energy consumption of the appliance is 1080 watts, while the minimum is 10 watts.
- The majority of the data in the light column are 0 values.
- The maximum pressure outside the home is 772.3 mm hg.
- There are no categorical columns in the dataset other than the date column.
- Average temperature outside is about 7.5 degrees. While it ranges from -6 to 28 degrees.
- There are no null or missing values.
- Outside humidity is higher than inside humidity.
- The maximum wind speed is 14 m/s.

Exploratory Data Analysis

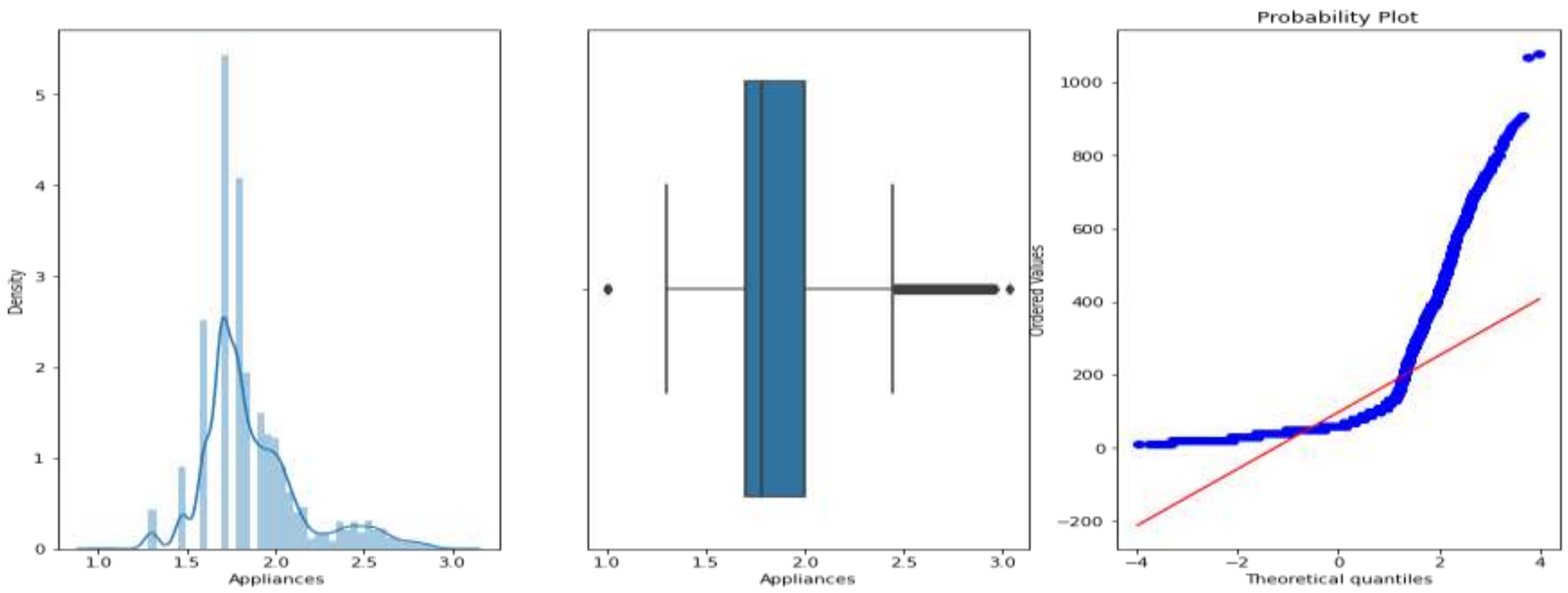
Checking distribution of target variable



<u>Observation:</u>

Since our graph is positively skewed, it is moving towards the y axis, and we couldn't get a better visualisation with this type of graph. As a result, it is better to apply a Log, Square Root, or Exponential transformation and check the dependent variable's distribution.

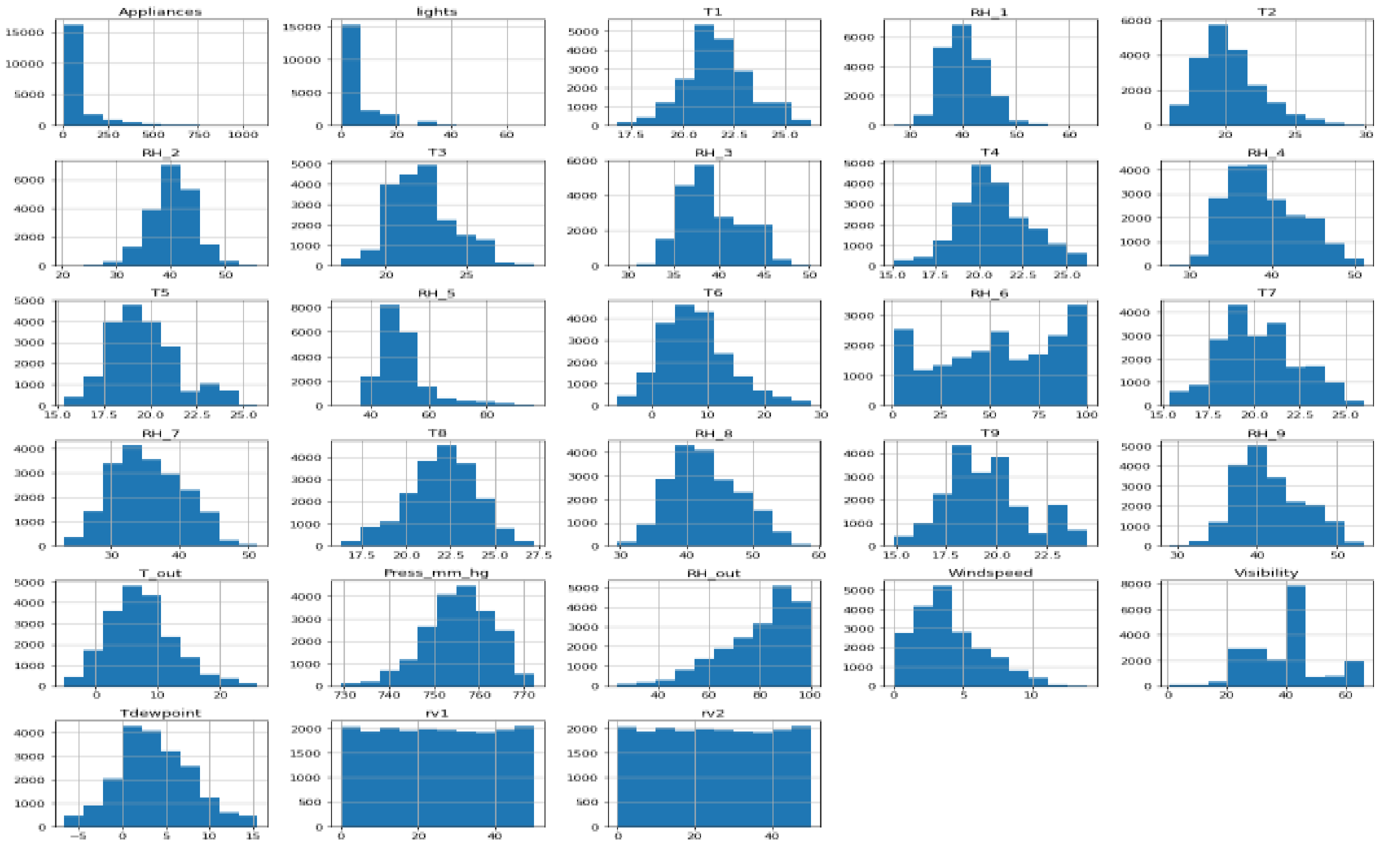
To check the distribution using Log transformation method on dependent variable



<u>Observation:</u>

It almost has a normal distribution after the log10 transformation.

Checking distribution of all the features

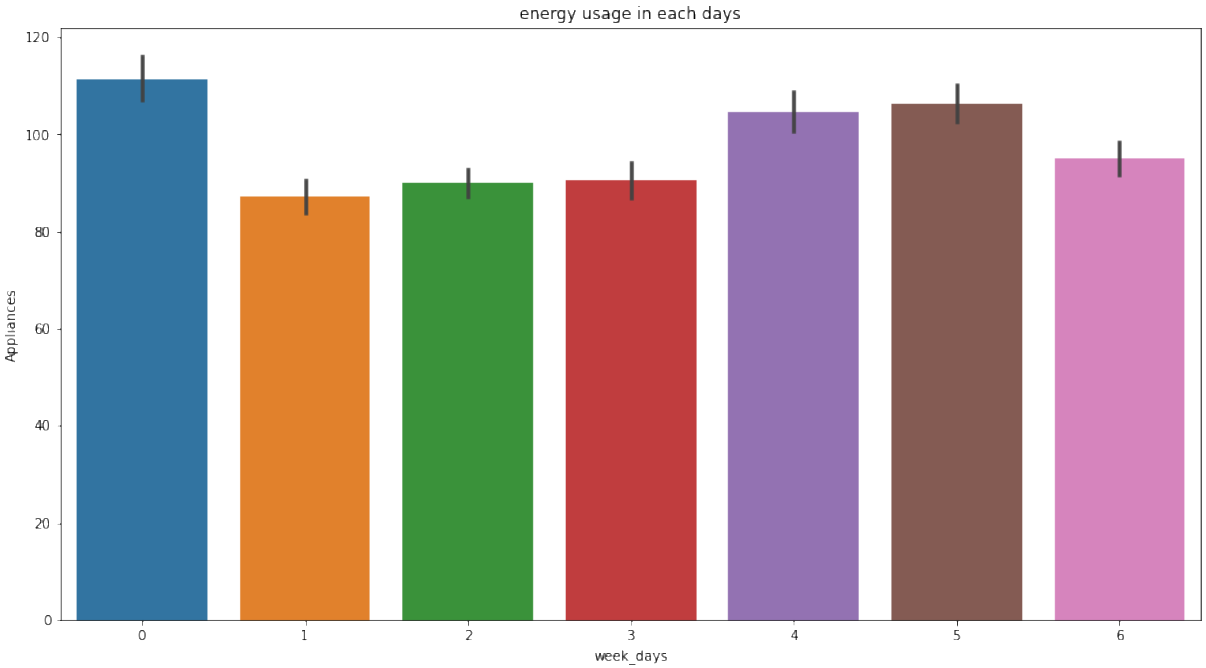


Except for lights, T2, RH6, RH out, windspeed, rv1 and rv2, the rest columns are normally distributed.

Observation

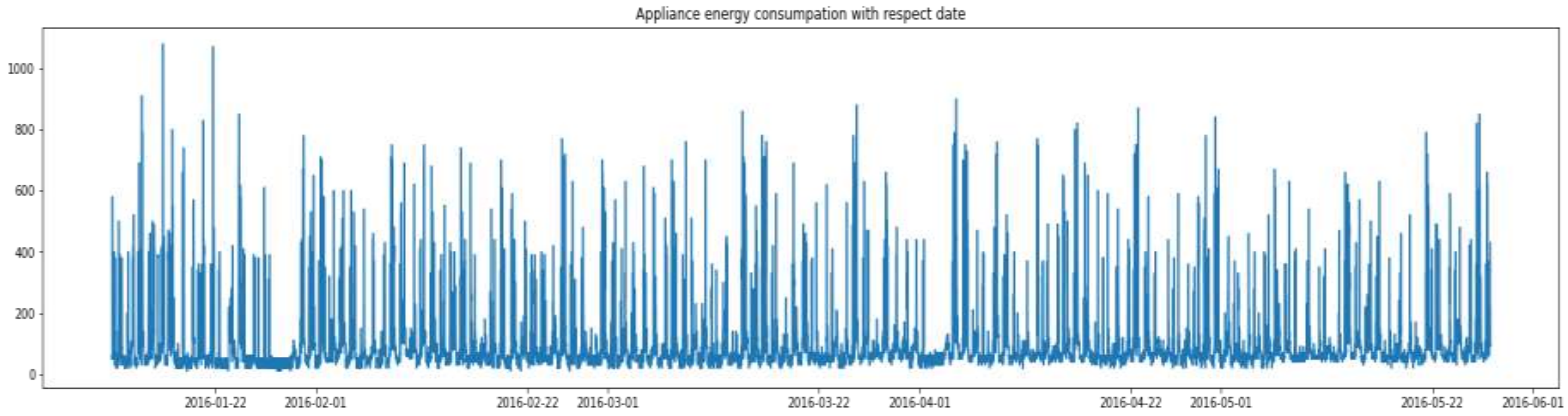
- Positively skewed(>1):- Appliances, RH_5.
- Moderately Positively skewed(0.5 to 1):- T2, T5, T6, T_out, RH_out, Windspeed.
- Normal Distributed(-0.5 to +0.5):- T1, T3, T4, T7, T8, T9, RH_1, RH_2, RH_3, RH_4, RH_6, RH_7, RH_8, RH_9, Press_mm_hg, Visibility, Tdewpoint, rv1, rv2,
- Negative skewed(-0.5 to -1):- No features.
- Moderately Negatively skewed(>-1):- RH_out.

Checking which day of week has more energy consumption?



0 - Sunday has a higher energy consumption rate, which indicates that more individuals are at home on Sunday.

Energy consumption vs Date



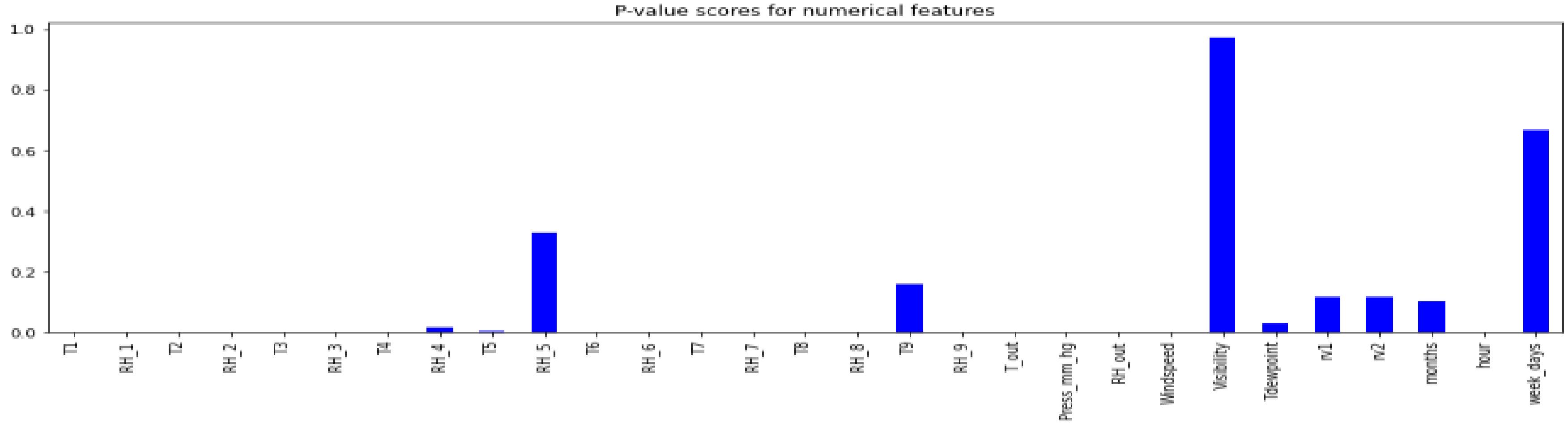
In the month of March, we can clearly see that appliances consume more energy, whereas in the month of January, appliances consume less energy.

Feature selection

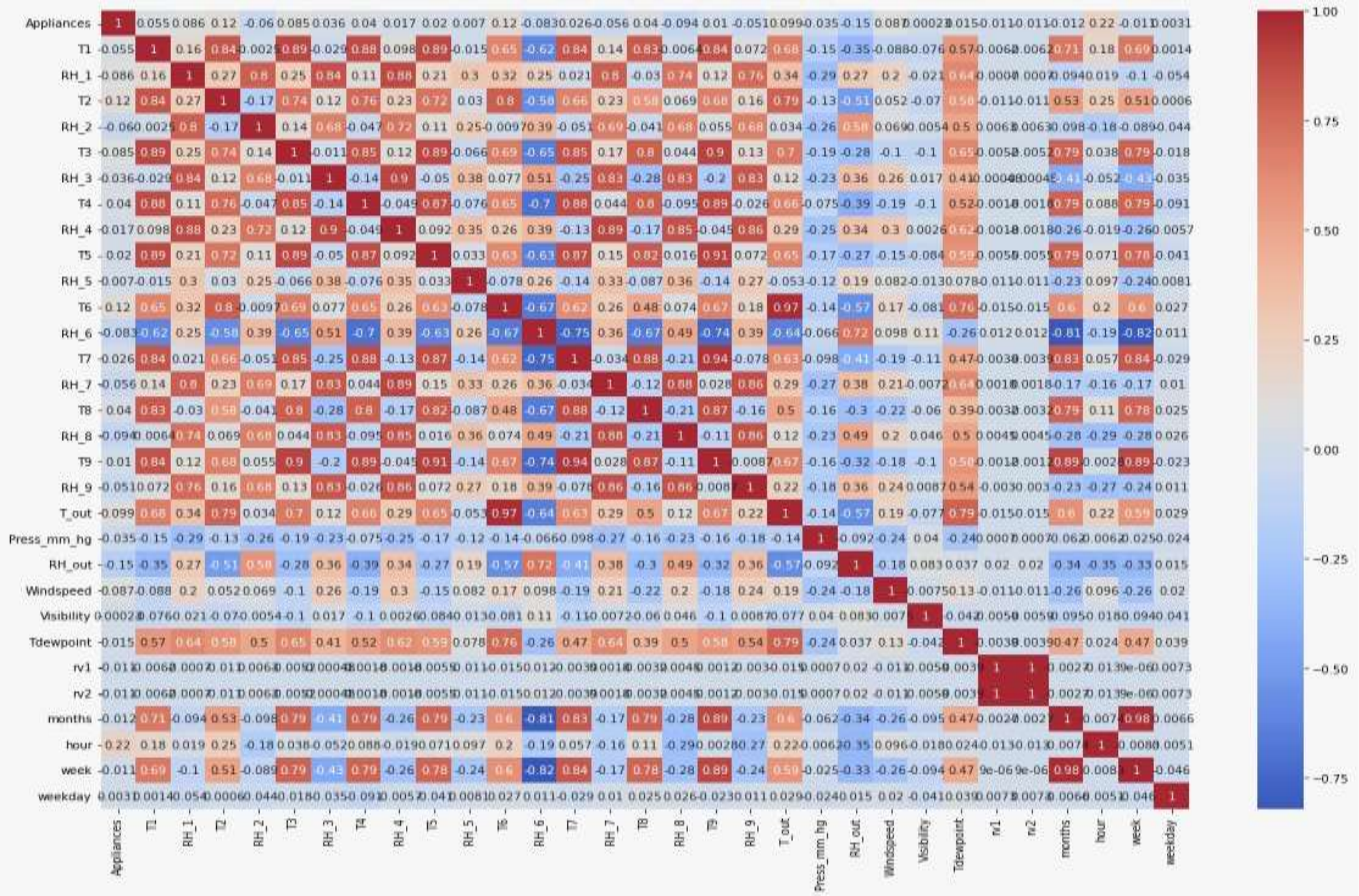
Removing Date and Light column

Date dropping reason: Since we're not trying to analyse the problem as a Time Series, we shall regress on the "Appliance" column.

Feature selection for numerical features using f_regression.



Correlation feature selection



Observations based on correlation plot

- Temperature - All the temperature variables from T1-T9 and T_out have positive correlation with the target Appliances.
- For the indoor temperatures, the correlations are high as expected, since the ventilation is driven by the HRV unit and minimizes air temperature differences between rooms.
- Five columns have a high degree of correlation with T9 - T3,T4,T5,T7,T8 also T6 & T_Out has high correlation(both temperatures from outside) . Hence T6 and T9 can be removed from training set as information provided by them can be provided by other fields.
- Weather attributes - Visibility, Tdewpoint, Press_mm_hg have low correlation values
- Humidity -There are no significantly high correlation cases (> 0.9) for humidity sensors.
- Random variables have no role to play

Feature Engineering

Checking outliers

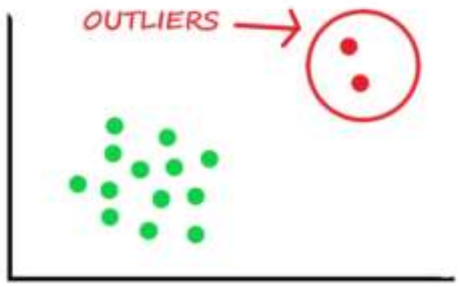
```
#checking the outliers
Q1 = df.quantile(0.25)
Q3 = df.quantile(0.75)
IQR = Q3 - Q1

((df < (Q1 - 1.5 * IQR)) | (df > (Q3 + 1.5 * IQR))).sum()
```

```
Appliances     857
T1             437
RH_1           127
T2             473
RH_2           199
T3             117
RH_3            11
T4             204
T5             249
RH_6             0
T7               0
RH_7            37
T8              93
RH_8            18
RH_9            23
T_out          332
Press_mm_hg    189
RH_out         279
Windspeed      224
hour             0
dtype: int64
```

majority of outliers are removed

```
[ ] df.shape
```

```
(17597, 20)
```



Test and Train split

```
[ ] Y=df['Appliances']
```

```
X=df.iloc[:,1:]
```

+ Code + Text

```
[ ] #spliting train and test
    from sklearn.model_selection import train_test_split
    X_train1, X_test1, y_train, y_test = train_test_split( X,Y, test_size = 0.2, random_state = 10)
    print(X_train1.shape)
    print(X_test1.shape)

(14077, 19)
(3520, 19)
```

using minmax scaler for scaling down data

```
[ ] # Transforming data
    scaler = StandardScaler()
    X_train = scaler.fit_transform(X_train1)
    X_test = scaler.transform(X_test1)
```

Fitting the multiple models

```
metrics_df1 = pd.DataFrame(model_data)
metrics_df1
```

	Name	MAE_train	MAE_test	R2_Score_train	R2_Score_test	RMSE_Score_train	RMSE_Score_test
0	LinearRegression:	17.303593	17.588016	0.312427	0.317623	23.532652	23.851801
1	Lasso:	18.176908	18.551637	0.257384	0.250863	24.456470	24.991352
2	Ridge:	17.303776	17.588532	0.312427	0.317590	23.532656	23.852377
3	PolynomialRegression:	11.389275	12.500642	0.690731	0.618888	15.782632	17.825252
4	DecisionTreeRegressor:	0.000000	12.906250	1.000000	0.486929	0.000000	20.682255
5	RandomForestRegressor:	3.832003	10.387699	0.959188	0.710397	5.733282	15.538564
6	GradientBoostingRegressor:	13.593958	14.254239	0.553749	0.519613	18.958373	20.012651
7	XGBRegressor:	13.635931	14.314385	0.549291	0.515243	19.052843	20.103483
8	LGBMRegressor:	10.314516	11.687117	0.736848	0.654073	14.558434	16.982481

Random forest is performing good. Now let's perform hyperparameter tuning on the all models

Cross validation and hyperparameter tuning

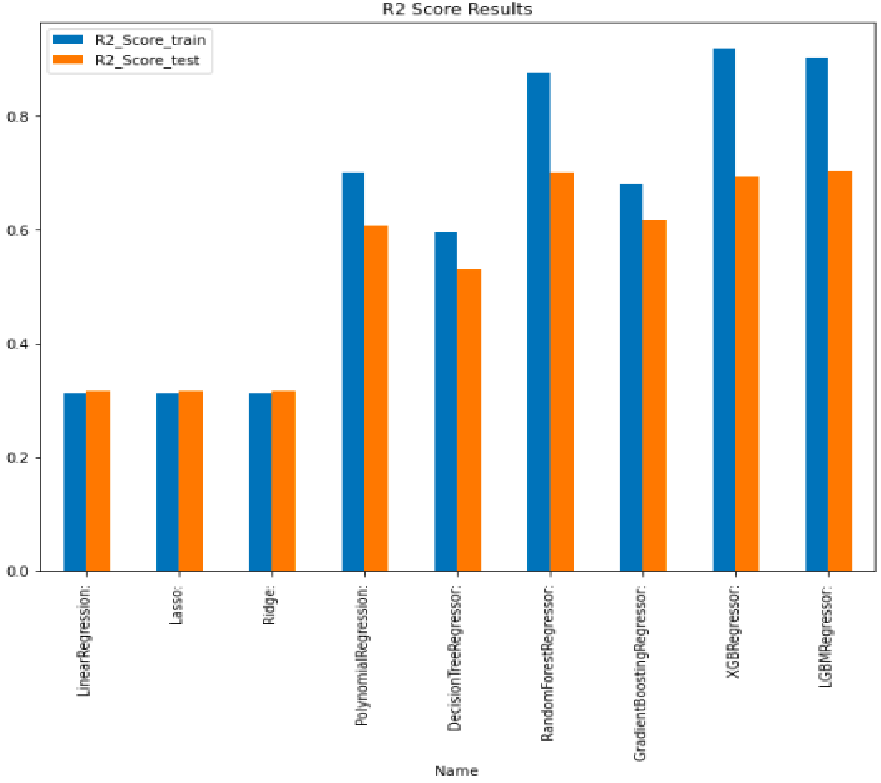
```
metrics_df = pd.DataFrame(model_data)
metrics_df
```

	Name	MAE_train	MAE_test	R2_Score_train	R2_Score_test	RMSE_Score_train	RMSE_Score_test
0	LinearRegression:	17.303593	17.588016	0.312427	0.317623	23.532652	23.851801
1	Lasso:	17.303677	17.588252	0.312427	0.317616	23.532652	23.851932
2	Ridge:	17.303776	17.588532	0.312427	0.317590	23.532656	23.852377
3	PolynomialRegression:	11.296273	12.705824	0.699887	0.607834	15.547259	18.081901
4	DecisionTreeRegressor:	12.818136	13.958515	0.596552	0.530430	18.026248	19.786049
5	RandomForestRegressor:	6.693986	10.650576	0.875777	0.701961	10.002581	15.763239
6	GradientBoostingRegressor:	11.283794	12.416449	0.681364	0.615583	16.019865	17.902374
7	XGBRegressor:	5.878341	10.740346	0.919322	0.694782	8.060985	15.951970
8	LGBMRegressor:	6.375924	10.631450	0.902580	0.702723	8.858004	15.743072

Representing r2 score through bar plot

```
#representing r2 score through bar plot
metrics_df.plot(x="Name", y=['R2_Score_train' , 'R2_Score_test'], kind="bar" , title = 'R2 Score Results' , figsize= (10,8))
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f32571e43d0>
```



Observation

- From above DataFrame we can see LinearRegression is not performing good at all.
- XGBRegression is giving r2 value of 0.91 for train data and 0.69 for test data.
- LGBMRegression is giving r2 value 0.90 for train data and 0.70 for test data.
- RandomForest Regression is giving r2 value of 0.87 train data and 0.70 for test data.
- **By comparing these models, RandomForest regressor is performing well with a high r2 score and low MSE and RMSE values.**

Model Explainability

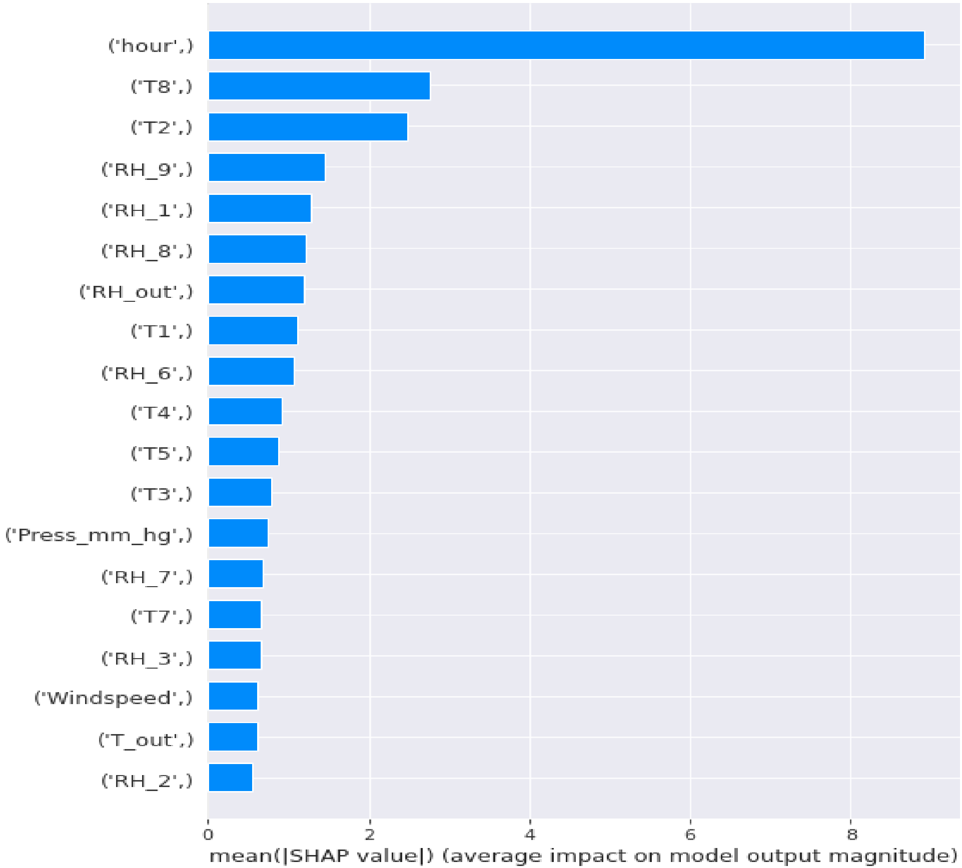
```python
# Initialize JavaScript visualizations in notebook environment
shap.initjs()
explainer_train = shap.TreeExplainer(rf_model)
# obtain shap values for the first row of the test data
shap.force_plot(explainer.expected_value[0], shap_values_train[0], X_train1.iloc[0])
```

This plot gives us the explainability of a single model prediction. Force plot can be used for error analysis, finding the explanation to specific instance prediction.

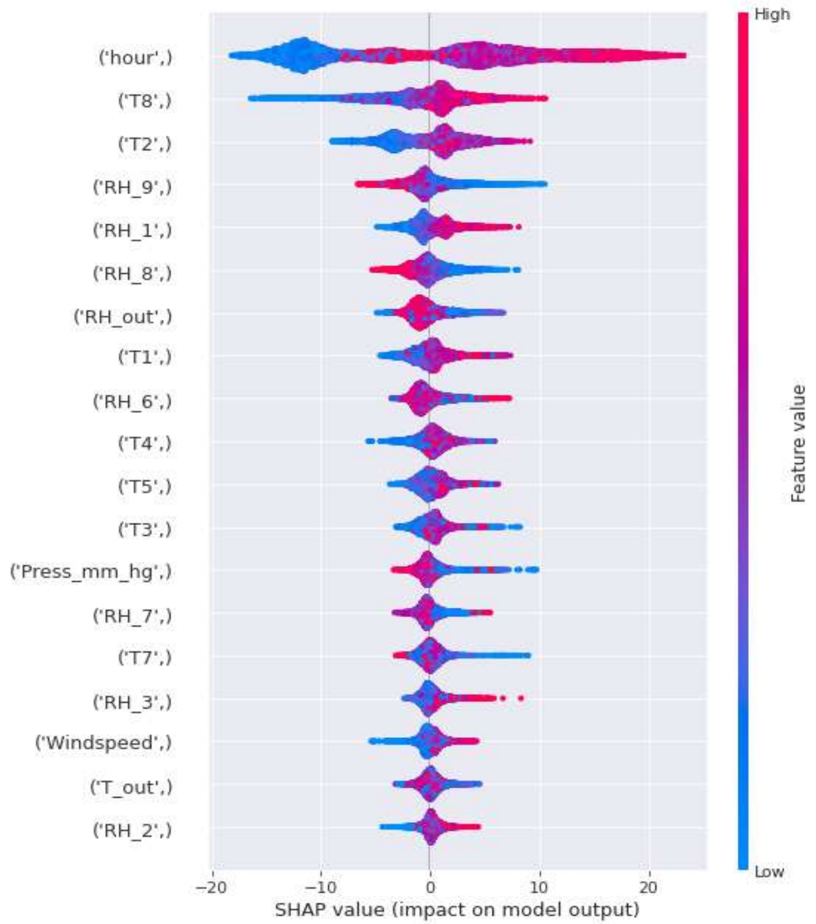
From the plot we can see:

- The model output value: 42.14
- The base value: this is the value that would be predicted if we didn’t know any features for the current instance. The base value is the average of the model output over the training dataset
- The numbers on the plot arrows are the value of the feature for this instance.
- Red represents features that pushed the model score higher, and blue representing features that pushed the score lower.
- The bigger the arrow, the bigger the impact of the feature on the output.
- The amount of decrease or increase in the impact can be seen on the x-axis.

Obtain a Bar Summary Plot



Obtain a dot Summary Plot



Summary:

The summary plot combines the relevance of features with the effects of features. Each point on the summary plot represents a Shapley value for a single instance of a feature. The feature determines the position on the y-axis, and the Shapley value of each instance determines the position on the x-axis. You can see that the most essential feature, the hour, has a high Shapley value range. The colour denotes the feature's value, which ranges from low to high. Overlapping points are jittered in the y-axis direction to give us a sense of the Shapley value distribution per feature. The features are arranged in descending order of importance.

Conclusion

The project's main goal is to predict appliance energy usage. First, we analyse the data. The data set is collected at regular intervals of time, so it is time series data, but we are not implementing time series techniques on the model due to a lack of awareness on time series. (Yet to be taught)

- Many columns in the dataset are not normally distributed, and the target column is also right skewed.
- The dataset has many outliers and no null values.
- We have a high correlation with the dependent variable in the hours column, and many features have less than a 0.1 correlation with the dependent variable in the non linear dataset.
- Energy consumption is high in March and low in January, and a rise in temperature results in higher energy consumption.
- Decreased humidity leads to an increase in electricity consumption. Humidity is proportional to the dependent variable.
- The most important determining factor for energy consumption is the hour of day.
- During the evening hours of 16:00 to 20:00, there is a high usage of electricity of more than 140Wh. Electricity use is highest on weekends (Saturdays and Sundays). (more than 25% higher than on weekdays)
- As a feature, lights are extremely undervalued.

We excluded features that were not important for predicting the dependent variable using a variance threshold, f regression, and the Pearson correlation matrix. We removed outliers from our model using feature engineering.

- Implementing the XGBM and LGBM regression algorithms was done along with cross validation and hyperparameter adjustment on all models. Decision tree, Random forest, Gradient Boosting, and LinearRegression were also used. **In a comparison of all models, the RandomForest regressor is the best, having a high r2 score, a low MSE, and a low RMSE value.** **Due to the time series nature of the dataset and the lack of time series concept implementation, some overfitting is occurring** The model explainability Shap approach is used to determine which attributes are crucial for predicting output and understanding the model. The most significant feature is the hour feature.

Improvement points:

- Definitely, we have a scope of improvement here, specially in the feature engineering.
- We may apply the time series concept to data that we obtain at regular intervals of time and analyse how the accuracy varies.
- Since there is just data for one house, analysing several houses will yield vital information.
- Additional information may be gained from aspects like the house's geometry and its occupant population over time.
- For better data gathering, positioning and sensor quality can be analyzed.

Future Work

Due to the availability of time features, we can do dynamic regression time series modelling. We can use topic modelling to address views in each topic separately.


THANK YOU