Capstone Project -2
Appliance Energy Prediction

by - Karan Malhotra

Prediction model of household appliance energy consumption based on machine learning

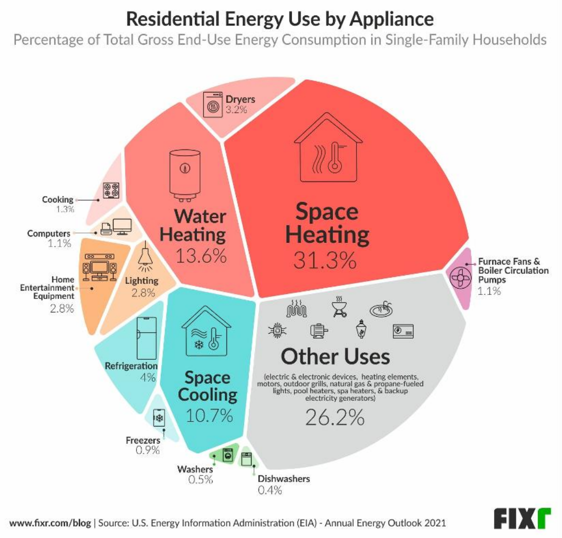
1. Defining Problem Statement
2. EDA and Feature Engineering
3. Getting Correlations
4. Feature Selection
5. Preparing Dataset for Modelling
6. Applying Model
7. Model Validation And Selection

Points for Discussion

1. Defining Problem Statement
2. Data Summary
3. Analysing Data
4. Feature Engineering And Checking for Outliers
5. Correlation Plots
6. Applying Models and Model Selection
7. Challenges while analysis
8. Conclusion

The Dilemma

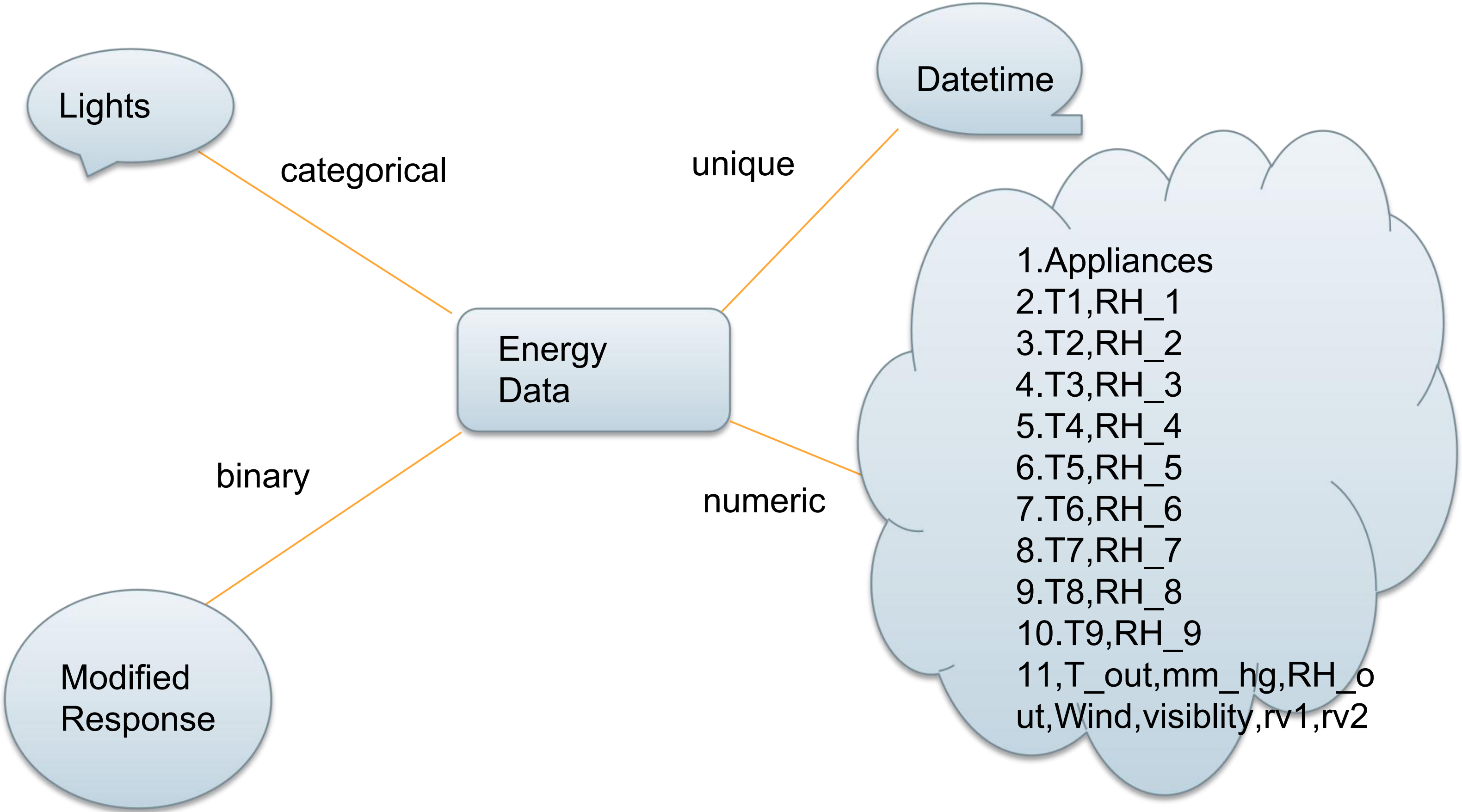
- The data set is at 10 min for about 4.5 months. The house temperature and humidity conditions were monitored with a ZigBee wireless sensor network. Data used include measurements of temperature and humidity sensors from a wireless network, weather from a nearby airport station and recorded energy use of lighting fixtures. Data filtering to remove non-predictive parameters and feature ranking plays an important role with this data. Different statistical models could be developed over this dataset. The idea of this project is to create regression models of appliances energy use in a low energy building.



Data Pipeline

- Data processing-1: In this first part we've learnt about columns and removed unnecessary features and found missing values if any.
- Data processing-2: In this part, we manually go through each features and define dependent and independent variables selected from part 1, encode the categorical features and also defining the target variable(Appliances).
- EDA: In in this part, we do some exploratory data analysis (EDA) on the features selected in part-1 and 2 and also visualize the data to see the trend and understand more about the features.
- Create a model: Finally, In this last but not the last part, we create models. Creating a model is also not an easy task. It's also an iterative process. we show how to start with a with a simple model, then slowly add complexity for better performance.

Data Summary



Data Summary

1. Independent Variables - 28(11 temperature, 10 humidity, 1 pressure, 2 randoms)

2. Dependent variable : 1 (Appliances)

3. Categorical Variables - Nearly 1 coulmn (Lights)

4. Two random variables have been included in the data set for testing the regression models and to filter out non-predictive attributes (parameters).

Data Summary

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. RH1 - Humidity in kitchen area, in %
6. T2 - Temperature in living room area, in Celsius

Data Summary

7. RH2 - Humidity in living room area, in %
8. T3 - Temperature in laundry room area
9. RH3 - Humidity in laundry room area, in %
10. T4 - Temperature in office room, in Celsius
11. RH4 - Humidity in office room, in %
12. T5 - Temperature in bathroom, in Celsius

Data Summary

13. RH5 - Humidity in bathroom, in %
14. T6 - Temperature outside the building (north side), in Celsius
15. RH6 - Humidity outside the building (north side), in %
16. T7 - Temperature in ironing room , in Celsius
17. RH7 - Humidity in ironing room, in %
18. T8 - Temperature in teenager room 2, in Celsius

Data Summary

19. RH8 - Humidity in teenager room 2, in %
20. T9 - Temperature in parents room, in Celsius
21. RH9 - Humidity in parents room, in %
22. Tout - Temperature outside (from Chievres weather station), in Celsius
23. Pressure (from Chievres weather station) - in mm Hg
24. RHout - Humidity outside (from Chievres weather station), in %

Data Summary

25. RHout - Humidity outside (from Chievres weather station), in %

26. Wind speed (from Chievres weather station) - in m/s

27. Visibility (from Chievres weather station) - in km

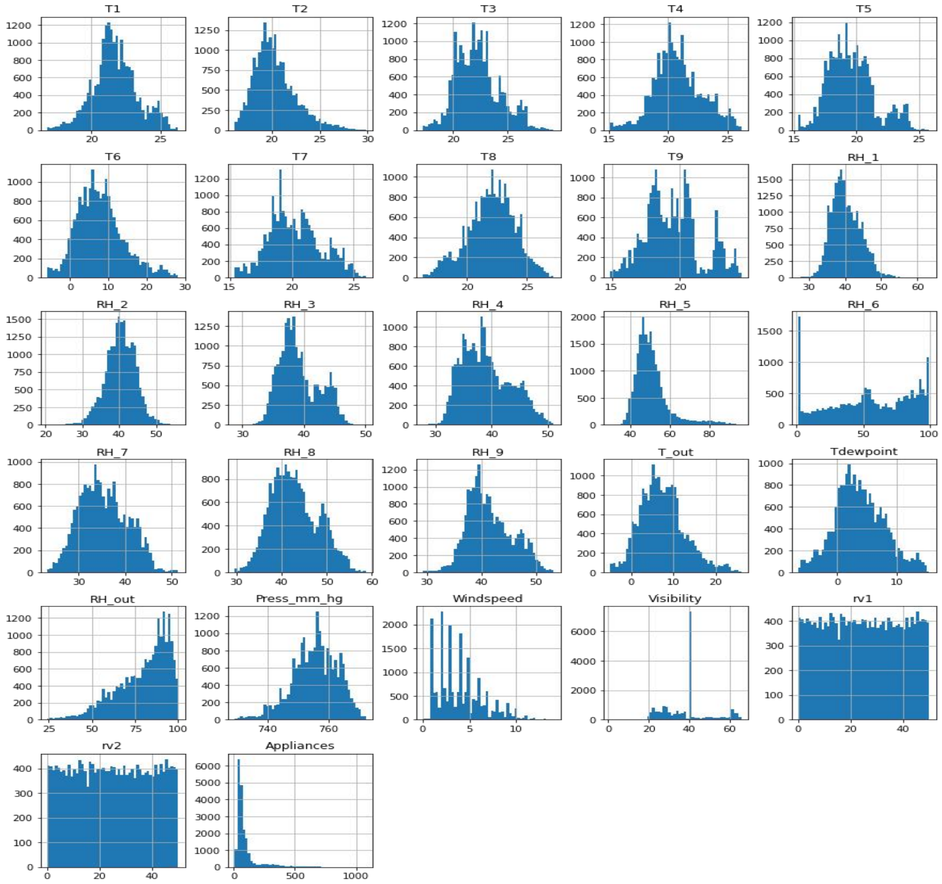
28. Tdewpoint (from Chievres weather station) - Â°C

29. rv1 - Random variable 1, nondimensional
    rv2 - Random variable 2, nondimensional

Extra features that were added

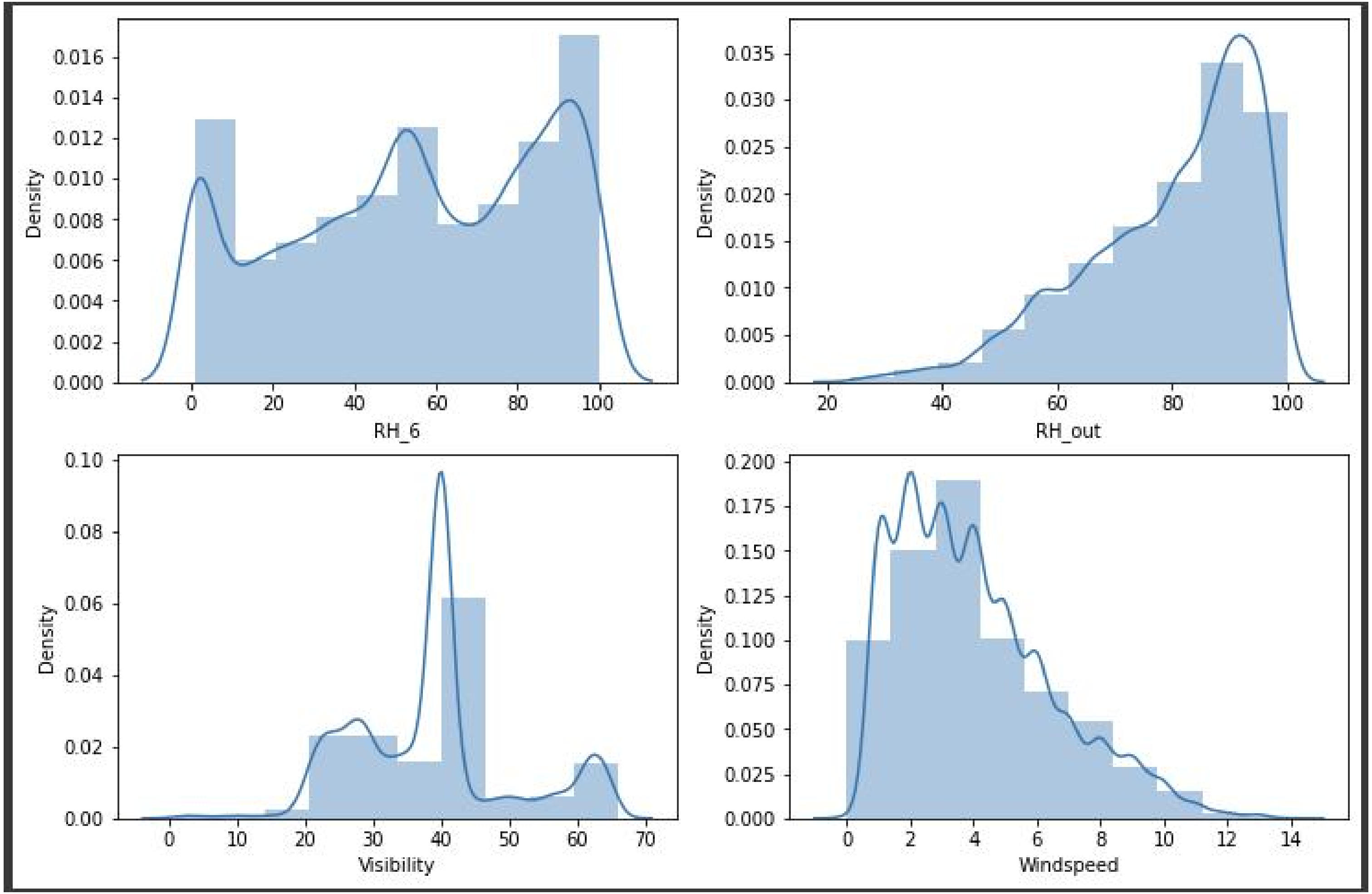
30. Modified_Response - for lights column(binary)
31. exact_date - Exact Data
32. hours - Hours (in 24hrs format)
33. seconds - Seconds(in 10min interval)
34. week - Exact day (same as days)
35. weekday - numerical column
36. log_appliances - log of appliances (normalized values)
37. days - same as week
38. days_num - same as weekday

EDA



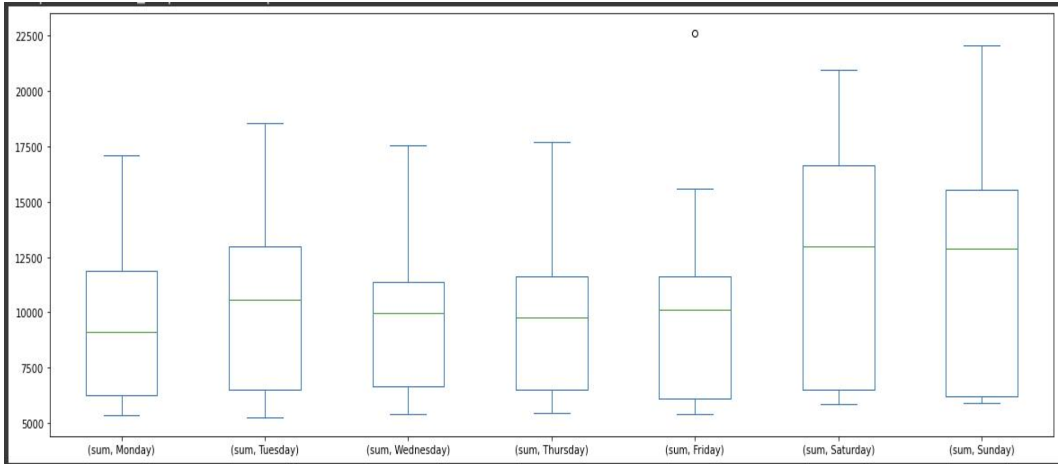
T1
T2
T3
T4
T5
T6
T7
T8
T9
RH_1
RH_2
RH_3
RH_4
RH_5
RH_6
RH_7
RH_8
RH_9
T_out
Tdewpoint
RH_out
Press_mm_hg
Windspeed
Visibility
rv1
rv2
Appliances

EDA

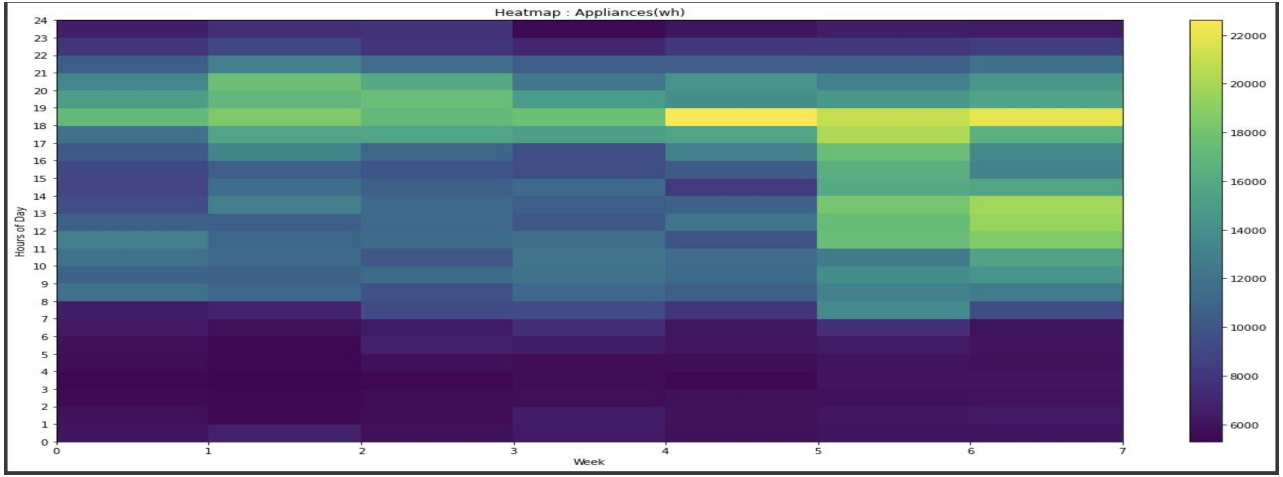


Density
RH_6
0.016
0.014
0.012
0.010
0.008
0.006
0.004
0.002
0.000
0 20 40 60 80 100
Density
RH_out
0.035
0.030
0.025
0.020
0.015
0.010
0.005
0.000
20 40 60 80 100
Density
Visibility
0.10
0.08
0.06
0.04
0.02
0.00
0 10 20 30 40 50 60 70
Density
Windspeed
0.200
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
0 2 4 6 8 10 12 14

EDA

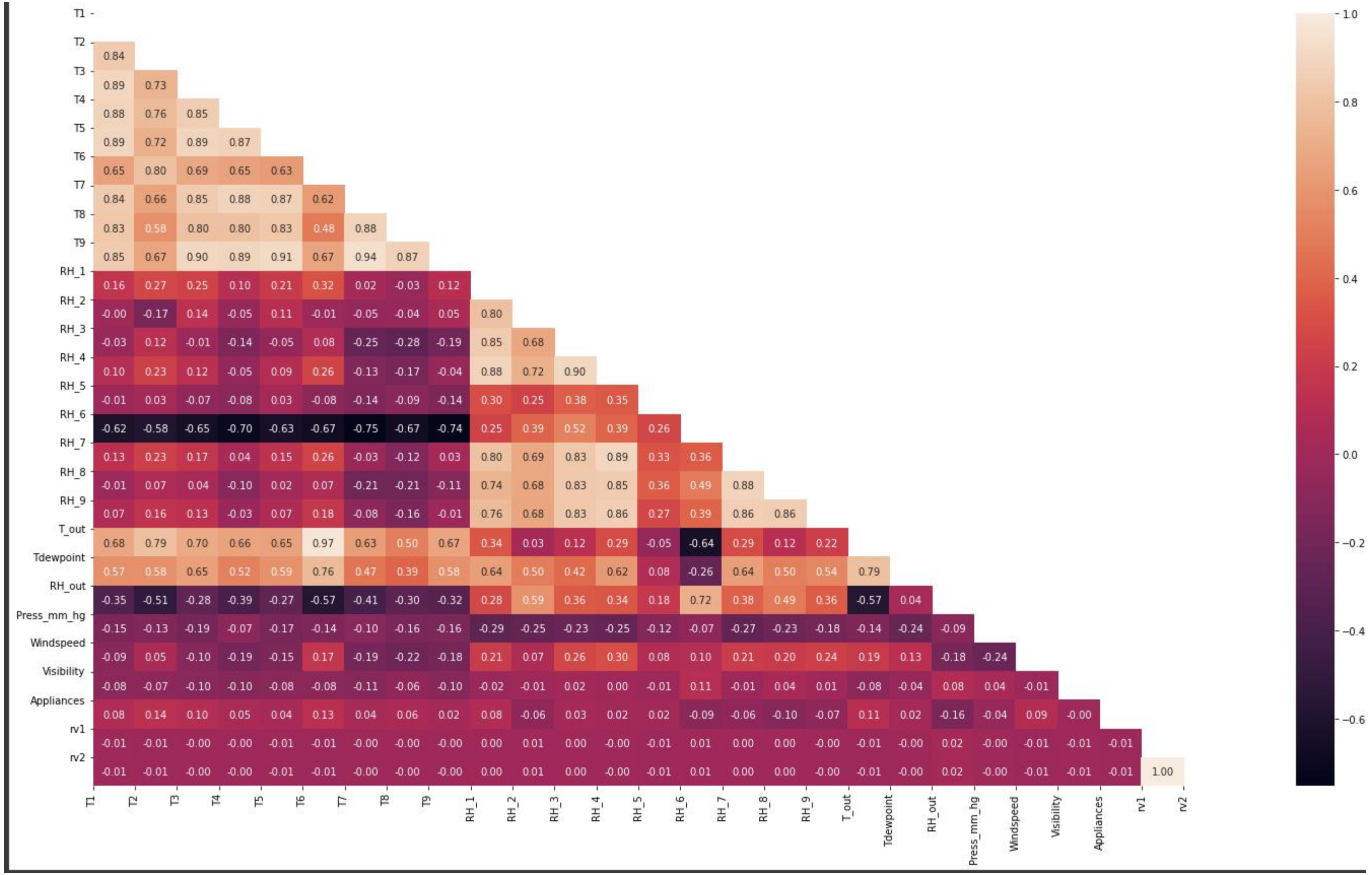


22500
20000
17500
15000
12500
10000
7500
5000
(sum, Monday)
(sum, Tuesday)
(sum, Wednesday)
(sum, Thursday)
(sum, Friday)
(sum, Saturday)
(sum, Sunday)



Heatmap : Appliances(wh)
Hours of Day
Week
0 1 2 3 4 5 6 7
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
22000
20000
18000
16000
14000
12000
10000
8000
6000

EDA continue



EDA continue

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. Four columns have a high degree of correlation with T9 - T3,T5,T7,T8 also T6 & T_Out has high correlation (both temperatures from outside) . Hence T6 & 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

Preparing Dataset for Modelling

```
sc_train.shape
```

```
(14652, 22)
```

```
sc_test.shape
```

```
(4885, 22)
```

```
#first five rows
sc_train.head()
```

date	T1	T2	T3	T4	T5	T7	T8	RH_1	RH_2	RH_3	RH_4	RH_5	RH_6	RH_7	RH_8	RH_9	T_out	Tdewpoint	RH_out	Press_mm_hg	Windspeed	Appliances
2016-03-30 13:20:00	-0.000667	0.387560	0.162027	0.190556	0.267012	0.287807	1.002244	0.354300	-0.024513	-0.034993	0.510852	-0.388441	-1.229881	-0.312569	-0.313833	-0.440669	0.514124	-0.091294	-1.096458	-0.293188	0.796933	0.105014
2016-01-17 02:50:00	-0.000667	0.199788	-0.459557	-0.283532	-0.660098	-0.323741	-0.532178	-0.107625	-0.423140	1.026454	0.632489	0.316808	1.338586	1.032912	1.050958	0.711248	-1.309564	-0.883363	1.167544	1.218009	-0.085356	-0.735455
2016-01-29 10:10:00	-2.307519	-1.490159	-1.587741	-1.945286	-1.576365	-1.508912	-2.475778	0.807840	0.924675	0.845630	1.091494	-0.052608	0.992354	0.719421	0.467768	0.414082	-0.262515	-0.162938	0.130341	1.136689	1.950696	-0.615388
2016-02-04 09:50:00	-0.526463	-0.227660	-0.442893	-0.115728	-0.546243	-0.970844	-0.625380	1.368688	0.508078	0.990358	0.581233	0.387223	1.337517	0.409824	0.464493	1.146212	-0.570656	0.004232	1.078322	0.852069	1.475617	-0.135120
2016-04-21 07:10:00	-0.125362	-0.716173	0.162027	0.530727	0.212795	0.477434	-0.111064	-0.826918	-0.082510	-1.013689	-1.188387	-0.436242	-0.666653	-0.374600	-0.522387	-0.751815	-0.564367	-0.572903	0.063425	1.028263	-0.764040	-0.735455

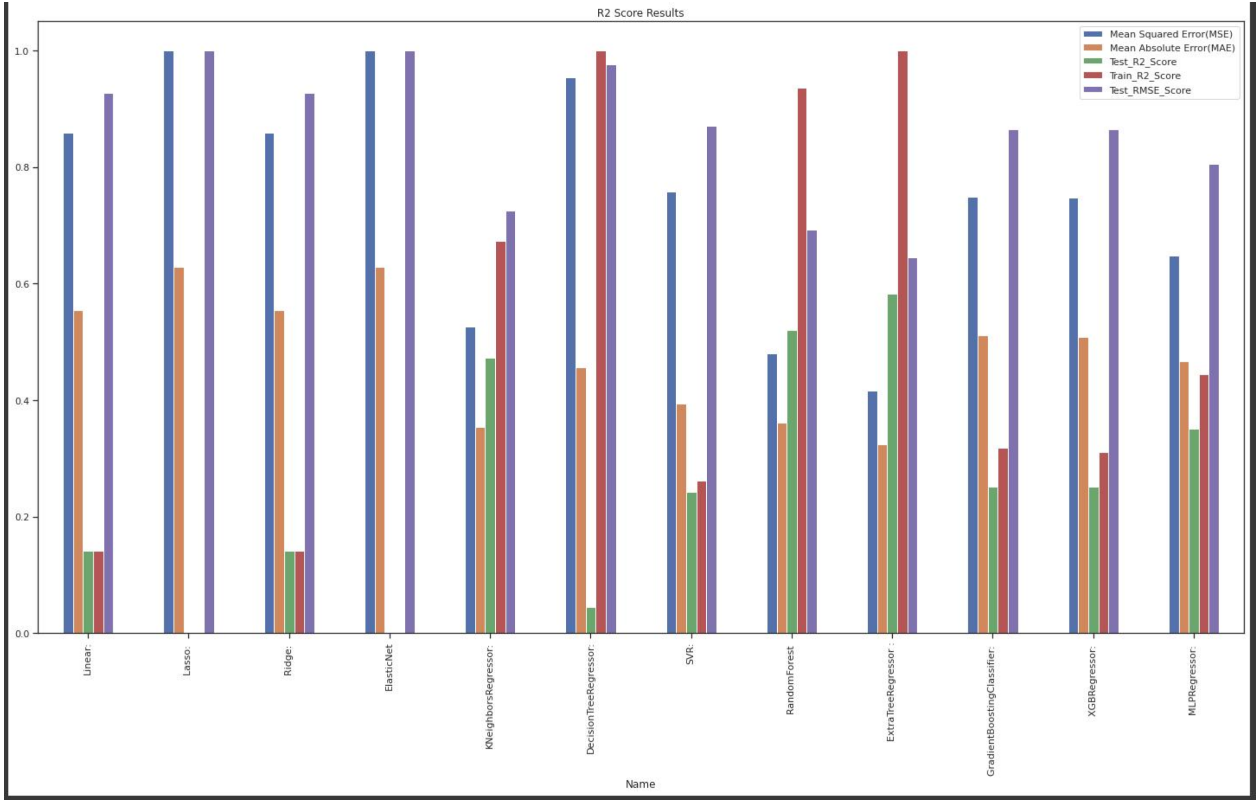
```
#first five rows
sc_test.head()
```

date	T1	T2	T3	T4	T5	T7	T8	RH_1	RH_2	RH_3	RH_4	RH_5	RH_6	RH_7	RH_8	RH_9	T_out	Tdewpoint	RH_out	Press_mm_hg	Windspeed	Appliances
2016-01-12 01:10:00	-0.403099	0.046500	-1.061017	0.220525	-0.303800	-1.157311	-1.383895	1.308524	0.933689	1.980913	1.978933	-0.098763	1.164481	2.020862	1.870751	1.143481	-0.404130	0.051145	0.808580	-2.383571	0.750173	3.163487
2016-02-29 12:10:00	-1.313305	-0.581705	-0.915520	-0.748451	-1.336236	-0.221336	-0.904713	-1.758009	-2.168314	-1.753124	-1.602668	-0.157480	-0.692411	-1.535116	-1.606538	-0.403345	-0.497358	-1.709153	-1.621593	0.998984	1.228523	0.188157
2016-04-04 22:10:00	0.815336	0.227278	1.177395	0.449675	1.266589	0.700385	1.119066	1.230218	1.610190	0.365268	1.176536	3.951622	-0.358906	0.961818	0.537202	0.750468	0.478429	1.117632	0.740761	-0.834258	-0.889884	-0.155150
2016-02-04 12:40:00	-0.533424	-0.162902	-0.065062	-0.137930	-0.536249	-0.994188	-0.837310	1.635216	0.620189	1.800359	0.833319	0.204175	1.399514	0.724929	0.400319	0.850552	-0.009465	0.665664	1.034643	0.795538	0.681837	1.675822
2016-02-24 12:50:00	-0.713396	-0.310537	-0.762562	-0.257416	-0.792849	-0.444640	-1.004967	-0.191063	-0.296387	0.331026	-0.397127	-0.086421	1.057569	-0.335660	-0.697033	-0.464697	-0.205243	-1.122387	-1.305106	0.428890	-0.821548	-0.498457

Applying Model

	Name	Train_Time	Mean Squared Error(MSE)	Mean Absolute Error(MAE)	Mean Absolute Percentage Error(MAPE)	Train_R2_Score	Test_R2_Score	Adjusted R2 score	Test_RMSE_Score	Root Mean Square Percentage Error(RMSPE)
0	Linear:	0.028795	0.858758	0.553898	1.418893	0.142006	0.141242	-0.012418	0.926692	2.958579
1	Lasso:	0.011029	1.000000	0.628939	1.000000	0.000000	0.000000	-0.012463	1.000000	1.000000
2	Ridge:	0.010204	0.858742	0.553883	1.418374	0.142006	0.141258	-0.012418	0.926683	2.956369
3	ElasticNet	0.010392	1.000000	0.628939	1.000000	0.000000	0.000000	-0.012463	1.000000	1.000000
4	KNeighborsRegressor:	1.713497	0.526855	0.353412	1.234967	0.673236	0.473145	-0.005396	0.725848	3.825796
5	DecisionTreeRegressor:	0.922083	0.954520	0.455938	1.545943	1.000000	0.045480	-4.036266	0.976995	5.636203
6	SVR:	17.621907	0.757906	0.394194	0.849942	0.262208	0.242094	-0.012372	0.870578	1.951544
7	RandomForest	31.747437	0.479912	0.361101	1.297027	0.936494	0.520088	-0.006307	0.692757	3.695460
8	ExtraTreeRegressor :	7.628698	0.416699	0.324671	1.152776	1.000000	0.583301	0.002759	0.645522	3.353880
9	GradientBoostingClassifier:	8.241798	0.749368	0.510780	1.286275	0.318070	0.250632	-0.010206	0.865660	2.696716
10	XGBRegressor:	1.586393	0.748256	0.508927	1.262491	0.311197	0.251744	-0.011044	0.865018	2.631421
11	MLPRegressor:	7.949086	0.648558	0.466437	1.576126	0.444153	0.351442	-0.009881	0.805331	3.745838

Model Validation And Selection



Model Validation And Selection

Observation 1: Best results over test set are given by Extra Tree Regressor with R2 score of 0.58, Least RMSE score is also by Extra Tree Regressor 0.6.

Observation 2: Lasso regularization over Linear regression was worst performing model.

Model Validation And Selection

Grid Search Cross Validation

```
[ ] #Hyperparameter tuning

    from sklearn.model_selection import GridSearchCV
    param_grid = [{
                  'max_depth': [80, 150, 200,250],
                  'n_estimators' : [100,150,200,250],
                  'max_features': ["auto", "sqrt", "log2"]
                }]
    reg = ExtraTreesRegressor(random_state=40)
    # Instantiate the grid search model
    grid_search = GridSearchCV(estimator = reg, param_grid = param_grid, cv = 5, n_jobs = -1 , scoring='r2' , verbose=2)
    grid_search.fit(train_X, train_y)
```

```
Fitting 5 folds for each of 48 candidates, totalling 240 fits
GridSearchCV(cv=5, estimator=ExtraTreesRegressor(random_state=40), n_jobs=-1,
             param_grid=[{'max_depth': [80, 150, 200, 250],
                          'max_features': ['auto', 'sqrt', 'log2'],
                          'n_estimators': [100, 150, 200, 250]}],
             scoring='r2', verbose=2)
```

```
grid_search.best_params_
```

```
{'max_depth': 80, 'max_features': 'sqrt', 'n_estimators': 200}
```

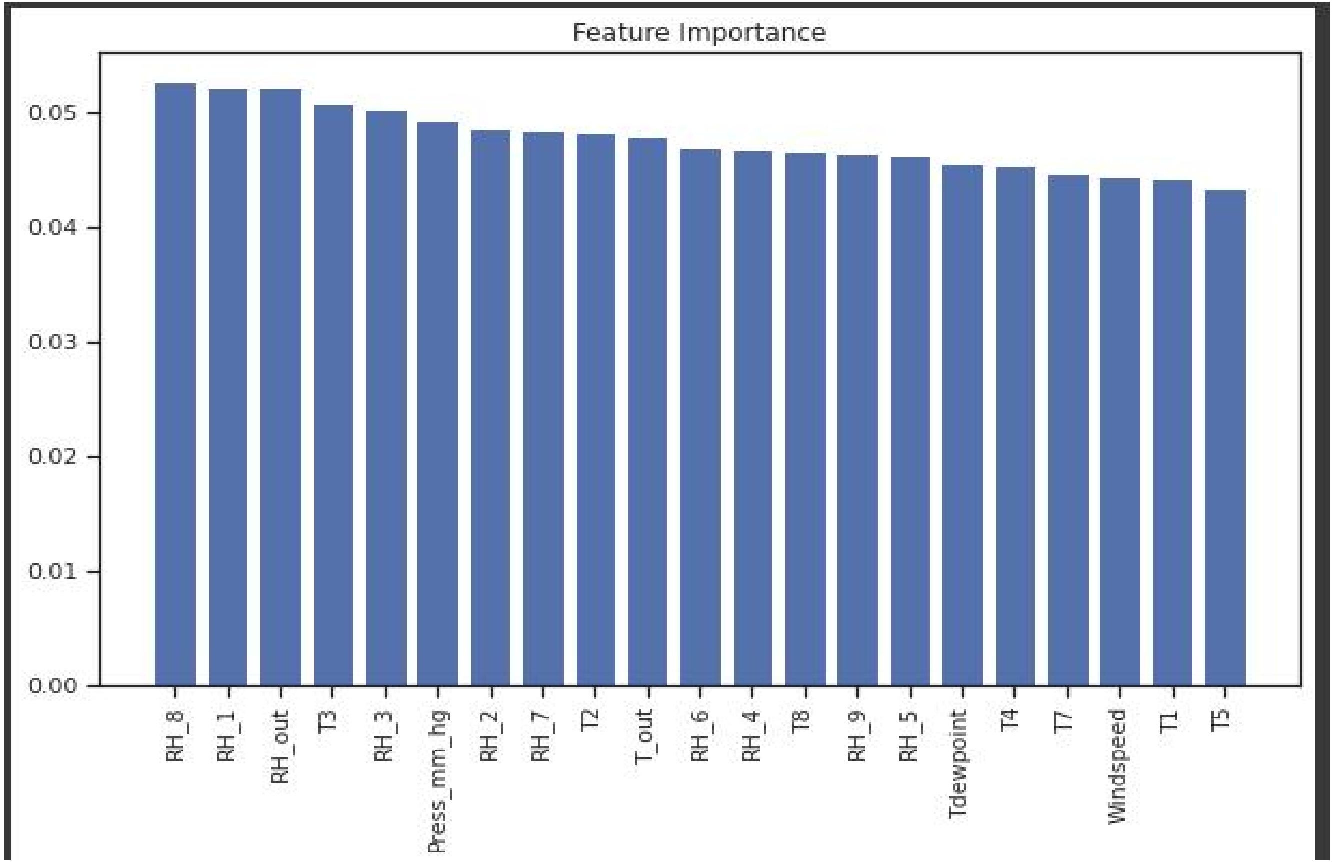
+ Code

```
[ ] # R2 score on training set with tuned parameters

    print("MAE score with tuned parameters is :",mean_absolute_error((test_y), grid_search.best_estimator_.predict(test_X)))
    print("MSE score with tuned parameters is :",mean_squared_error((test_y), grid_search.best_estimator_.predict(test_X)))
    print("RMSE score on test set with tuned parameters is :",np.sqrt(mean_squared_error(test_y, grid_search.best_estimator_.predict(test_X))))
    print("RMSPE score on test set with tuned parameters is :",np.sqrt(np.mean(np.square(((test_y - y_pred) / test_y)), axis=0)))
    print("R2 score on training set with tuned parameters is :",grid_search.best_estimator_.score(train_X,train_y))
    print("R2 score on test set with tuned parameters is :",grid_search.best_estimator_.score(test_X,test_y))
```

```
MAE score with tuned parameters is : 0.32119815263608265
MSE score with tuned parameters is : 0.408520452410966
RMSE score on test set with tuned parameters is : 0.6391560469955408
RMSPE score on test set with tuned parameters is : 3.7458379772862918
R2 score on training set with tuned parameters is : 1.0
R2 score on test set with tuned parameters is : 0.591479547589034
```

Feature Importance



Feature Importance Observation

5 most important features are - 'RH_out', 'RH_8', 'RH_1', 'T3', 'RH_3'

5 least important features are - 'T7','Tdewpoint','Windspeed','T1','T5'

Conclusion

1. Hour of the Day is the most important influencing parameter for Energy consumption

2. Overall Random Forest appears to closely fit with the test data

3. The best Algorithm to use for this dataset Extra Trees Regressor as it performed the best with default parameters.

4. The untuned model was able to explain 57% of variance on test set .

Conclusions Continue

5. The tuned model was able to explain 63% of variance on test set which is improvement of 10%.

6. The final model had 22 features.

7. Feature reduction was not able to add to better R2 score.

8. Though light consumption appeared as highly correlated with Appliance electricity consumption, lights are having very low importance as a feature.

9. Weekends (Saturdays and Sundays) are observed to have high consumption of Electricity. (> 25% than Weekdays)

Conclusions

10. High Electricity consumption of >140Wh is observed during evening hours 16:00 to 20:00

11. According to best fit model , the 5 most and least important features

The top 3 important features are humidity attributes, which leads to the conclusion that humidity affects power consumption more than temperature. Windspeed is least important as the speed of wind doesn’t affect power consumption inside the house. So controlling humidity inside the house may lead to energy savings.

Challenges

1. Feature scaling is very important for regressions models , I initially tried without it and the results were not good . On Kaggle this is suggested by all users.

2. It is very important to check the intercorrelation between all the variables in order to remove the redundant features with high correlation values.

3. While scaling data , it is useful to maintain separate copies of dataframe which can be created using index and column names of original dataframe.

4. For performing Exhaustive search or Random search in the hyperparameter space for tuning the model, always parallelize the process since there are a lot of models with different configurations to be fitted.

Q & A