Exploratory Data Analysis

Factors considered for dropping the features

Features date, rv1, rv2 are just random and do no have any relation to the target variable – energy usage.

```
Features with high correlation >0.88

Feature1        Feature2        Correlation
T3               T1              0.892
RH_4             RH_1            0.88
RH_4             RH_3            0.899
T5               T1              0.885
T5               T3              0.888
RH_7             RH_4            0.894
T8               T7              0.882
RH_8             RH_7            0.884
T9               T3              0.901
T9               T4              0.889
T9               T5              0.911
T9               T7              0.945
T_out            T6              0.975
rv2              rv1             1.0
```

Count of Values for Feature = lights

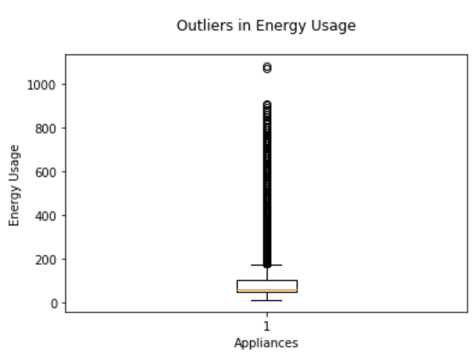
	0	10	20	30	40	50	60	70
lights	15252	2212	1624	559	77	9	1	1

Lights has almost 80% (15252) of its value to be 0. Lights feature can be removed.

From the above interpetation, columns

'date','lights','T3','T_out' ,'RH_4','T4','T7','RH_7','T5', 'rv1','rv2' can be dropped from the dataset.

Outliers Detection and Removal




```
count    19735.000000
mean        97.694958
std        102.524891
min         10.000000
25%         50.000000
50%         60.000000
75%        100.000000
max       1080.000000
Name: Appliances, dtype: float64
```

Thresold for Outlier Point

Upper Bound = (1.5*IQR) + 75th percentile, Lower Bound = (1.5*IQR) - 25th percentile

Upper Bound = 1.5*(100-50) + 100 = 175, Lower Bound = 1.5*(100-50) - 50 = -25

Number of outliers = (count of values above the upper bound) + (count of values below the lower bound)

Number of outliers = 2138 which is 10.8% of the total observations

We may not consider deleting all the outlier observations, since 10.8% of the total observations is considerably high. But the model accuracy and metrics drastically increased after the removal of these outliers. So, we are deleting the outlier observations in this situation.

Experimentation 1: Linear Regression

Accuracy/Error variation with different Learning Rates

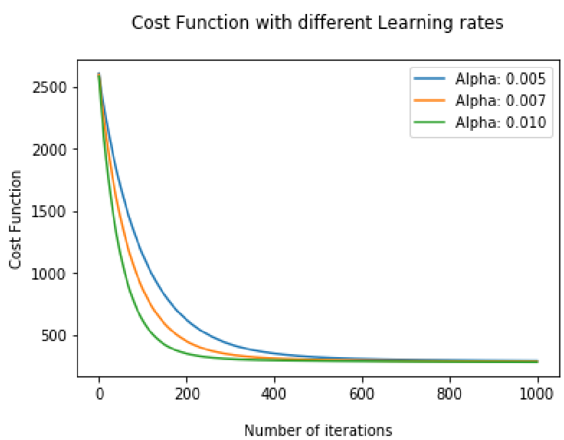
Learning Rates used: 0.005, 0.007, 0.01, Initial betas: 0.5, Number of iterations: 1000

Metrics considered: Cost Fuction, Mean Squared Error, Mean Absolute Error, R-Squared

Below are the Gradient Descent Results

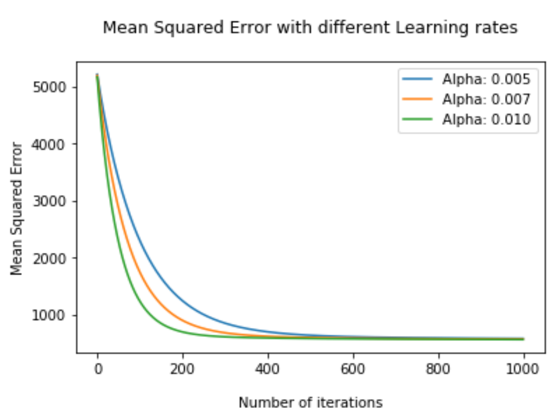
Cost Function Convergence (When difference in Cost Fuction for each iteration was <=0.1)

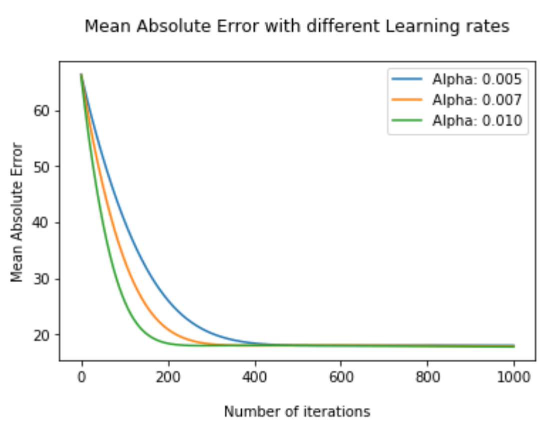
```
With learning rate of 0.005, cost fuction converged at iteration 585
With learning rate of 0.007, cost fuction converged at iteration 453
With learning rate of 0.010, cost fuction converged at iteration 347
```



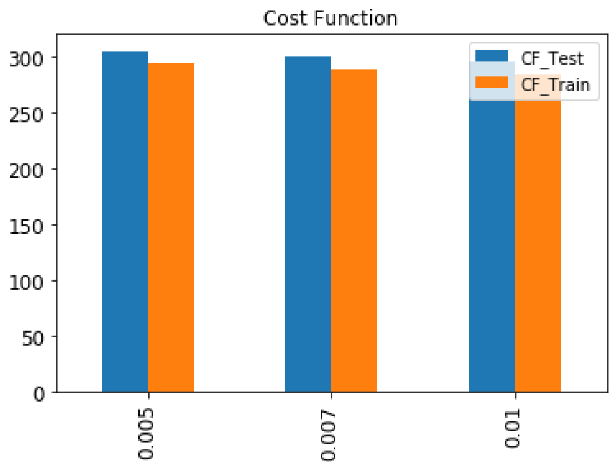
Cost Function converges quickly with increasing learning rates or increasing step sizes

Model with Alpha 0.01 performs the best as it converges quickly.





	CF_Test	CF_Train	MAE_Test	MAE_Train	MSE_Test	MSE_Train	R2_Test	R2_Train
0.005	304.5010	293.2792	18.1977	18.0251	609.0019	586.5584	0.2564	0.2738
0.007	299.5531	288.1972	18.0840	17.9445	599.1063	576.3945	0.2685	0.2864
0.010	295.7512	284.1783	17.9523	17.8141	591.5025	568.3566	0.2778	0.2963



From the results, Cost Fuction and the other errors are the lowest when learning rate = 0.01 and hence the best model.

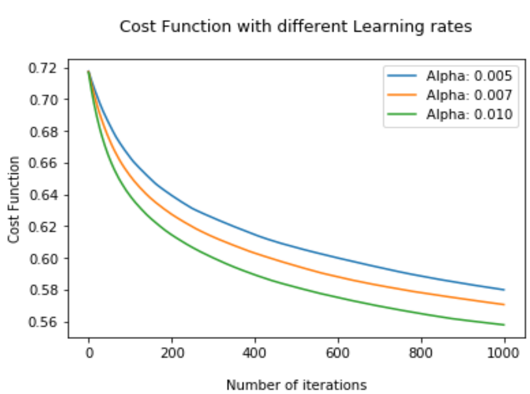
Energy Usage = 67.1349 + β1*T1 + β2*RH_1 + β3*T2+ β4*RH_2 + β5*RH_3 + β6*RH_5 + β7*T6 + β8*RH_6 + β9*T8 + β10*RH_8 + β11*T9 + β12*RH_9 + β13*Press_mm_hg + β14*RH_out + β15*Windspeed + β16*Windspeed + β17*Tdewpoint

β0=67.1349 β1=2.9932 β2=11.2539 β3=3.4322 β4=-0.553 β5=1.1047, β6=2.7928, β7=2.7835,β8=1.9768,β9=8.9667,β10=-9.7271,β11=-10.2668,β12=-8.9756,β13=-0.7374,β14=-0.5899, β15=2.2409,β16=0.1185,β17=0.9451

Logistic Regression

Transforming the energy usage into classes based on its median. Median is 60, so if energy usage <= 60 then class 0, else class =1

	Appliances_class
0	10744
1	6853



Plot shows the Cost Fuction for different learning rates after 1000 iterations. Cost Function converges quickly with increasing learning rates.

Model with Alpha = 0.01 performs the best.

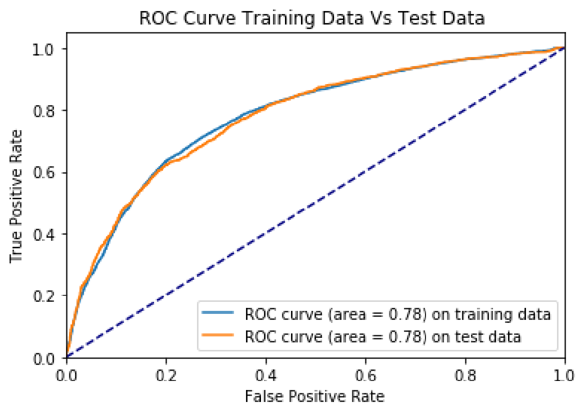
Confusion Matrix

Training Data

	Predicted 0	Predicted 1
Actual 0	6414	1108
Actual 1	2223	2572

Test Data

	Predicted 0	Predicted 1
Actual 0	2762	460
Actual 1	981	1077



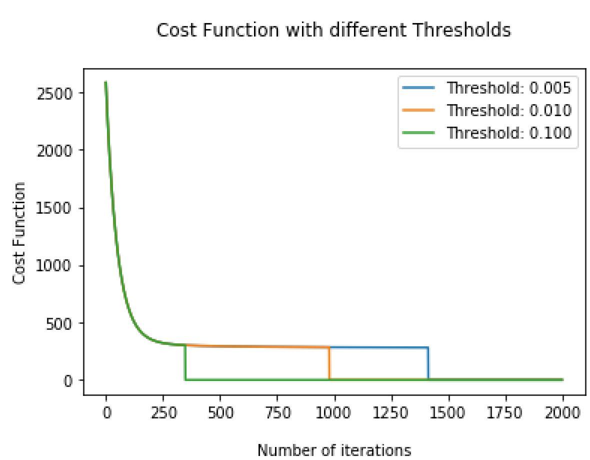
Accuracy: 0.729 **Accuracy: 0.727**

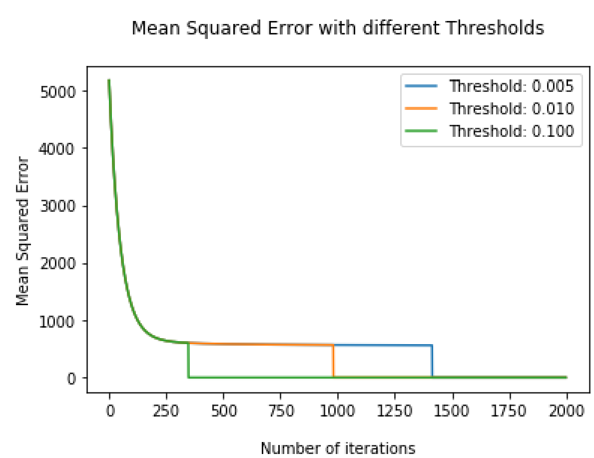
β0=-0.437 β1=0.0798 β2=0.3029 β3=0.2156 β4=-0.0554 β5=0.0445,β6=0.2512,β7=0.1891,β8=0.0643,β9=0.2423,β10=-0.3875,β11=-0.2166,β12=-0.3899,β13=-0.106,β14=-0.1542, β15=0.1746,β16=0.0261,β17=-0.0095

Experimentation 2: Varying thresholds for Cost Function

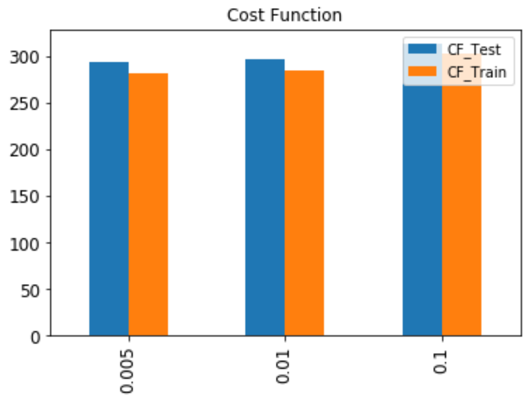
Thresholds used: 0.005, 0.01, 0.1, Initial betas: 0.5, Alpha: 0.01 Number of iterations: 2000

```
Convergence with threshold 0.005000 reached at iteration 1413
Convergence with threshold 0.010000 reached at iteration 980
Convergence with threshold 0.100000 reached at iteration 347
```





	CF_Test	CF_Train	MAE_Test	MAE_Train	MSE_Test	MSE_Train	R2_Test	R2_Train
0.005	293.0501	281.2739	17.8528	17.7112	586.1001	562.5477	0.2844	0.3035
0.010	295.9267	284.3652	17.9587	17.8206	591.8534	568.7304	0.2773	0.2959
0.100	313.1375	301.9032	18.1993	18.0263	626.2751	603.8064	0.2353	0.2524



As the threshold decreases, the number of iterations required to achieve it increases.

Model with the lowest threshold 0.005 performs the best as it takes the highest number of iterations.

Experimentation 3: Choosing 10 Random Variables

Linear Regression:

Initial betas: 0.5, Alpha: 0.01 Number of iterations: 1000

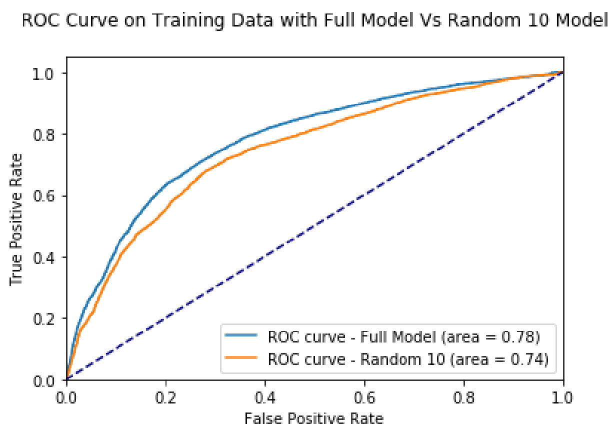
The ten random features selected are 'RH_2', 'RH_9', 'Tdewpoint', 'RH_1', 'RH_4', 'T2', 'T8', 'RH_7', 'Press_mm_hg', 'RH_6'. These features are selected using the Random Fuction.

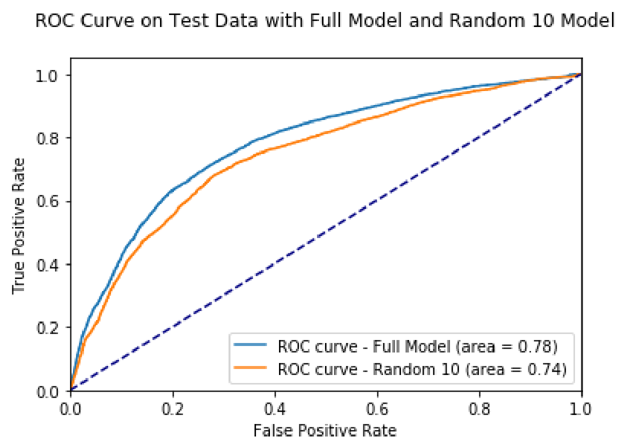
Comparison of Train and Test results for Exp: 1 Model and Exp: 3 Model

	CF_Test	CF_Train	MSE_Test	MSE_Train	R2_Test	R2_Train
Random 10	338.3284	328.2843	676.6569	656.5687	0.1738	0.1871
Full Model	295.7512	284.1783	591.5025	568.3566	0.2778	0.2963

Logistic Regression

Model	Training Accuracy	Test Accuracy
Experiment 1 Model	0.729	0.727
Experiment 3 Model	0.706	0.708





For both Linear and Logistic Regressions, Experiment 1 Model (>15 features) performs better than the Experiment 3 Model (10 Features randomly selected)

Experimentation 4

Best 10 features were selected based on each feature correlation with the target variable, based on increasing order of collinearity.

The best selected features are 'T1', 'T2', 'RH_2', 'T6', 'RH_6', 'T8', 'RH_8', 'T9', 'RH_9', 'RH_out'

Full Model here represents the experiment 1 model, with greater than 15 features.

Linear Regression Metrics

	CF_Test	CF_Train	MSE_Test	MSE_Train	R2_Test	R2_Train
Random 10	338.3284	328.2843	676.6569	656.5687	0.1738	0.1871
Best 10	315.4028	304.4411	630.8057	608.8821	0.2298	0.2462
Full Model	295.7512	284.1783	591.5025	568.3566	0.2778	0.2963

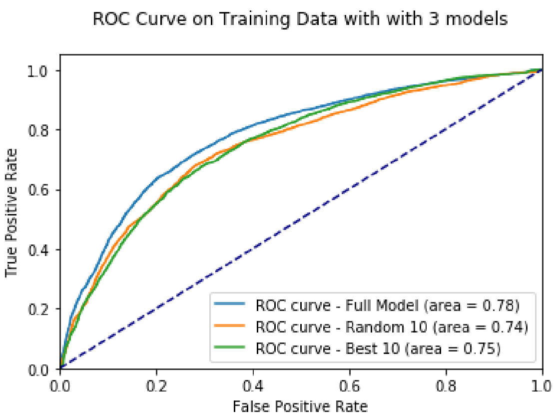
Logistic Regression Metrics

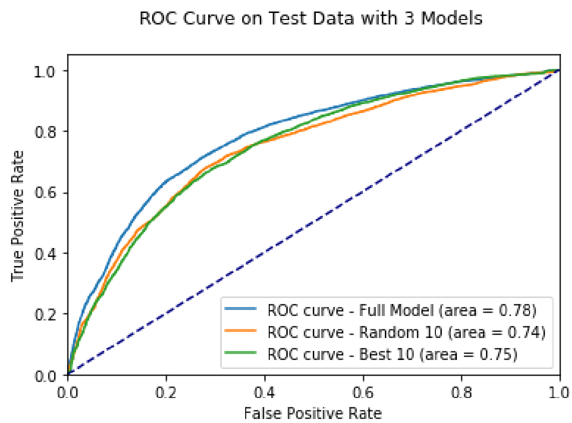
Model	Training Accuracy	Test Accuracy
Experiment 1 Model	0.729	0.727
Experiment 3 Model	0.706	0.708
Experiment 4 Model	0.6985	0.695

For both Linear and Logistic Regression, Experiment 1 model (>15 features) performs better than the Random 10 model and the Best 10 model.

This is because the Experiment 1 model has more features that can contribute towards the variation in Energy Usage. Or in other words we can say that the dataset has more than 10 features that are correlated with energy usage.

And with the best 10 model, the number of highly correlated features with target variable has been restricted to 10. So, it is always a good practice to include as many correlated features in the model to better the model metrics.





Discussion

Linear Regression:

When the learning rate = 0.01, the model gave us the best results on the testing data with Cost Function: 295.75 and R-Squared: 0.28.

With R-squared to be only 0.28, only 28% of the variation in Energy Usage is casued by the features in the dataset. This also means, there could be other factors that are contributing towards the energy usage but not present in the model. These unobserved features are captured in the error term leading the cost funtion contribution.

To improvise the model metrics and accuracy, transformation on some of the feature variables can be performed like including the logarithm, squaring, cubing of the features. In conclusion, this dataset does not perform the best on Linear Regressions. Other regression algorithms like Lasso, Ridge RandomForestRegressor can be implemented to improvise the accuracy and the metrics.

Some of the major contributors of energy usage were Temperature in teenager room (T8), Temperature in Living Room area (T2), Humidity Outside (RH_out), Temperature in Kitchen area (T1), Humidity outside the building (RH_6)

Logistic Regression

With respect to Logistic Regression, model with learning rate of 0.01 gave us the best results with accuracy close to 73%. The model has two classes: 0, when Energy Usage was less than or equal 60 and class 1, when Energy Usage was greater than 60. Although Logistic Regression gave us a good accuracy, other classification models like RandomForestClassifier, Support Vector Machines, Nearest Neighbours models could have given even more better results.