ASSIGNMENT 1 REPORT – SANDEEP PRADEEP KUMAR(SPR170000)

DATASET:

The data is from UCI Machine Learning Repository. It is about the energy of the appliances with the temperature and humidity monitored in all the rooms of a low energy building. The dataset has 29 variables and 19736 observations monitored over a period of 4.5 months. Below are the variables in the dataset:

date time year-month-day hour:minute:second
Appliances, energy use in Wh
lights, energy use of light fixtures in the house in Wh
T1, Temperature in kitchen area, in Celsius
RH_1, Humidity in kitchen area, in %
T2, Temperature in living room area, in Celsius
RH_2, Humidity in living room area, in %
T3, Temperature in laundry room area
RH_3, Humidity in laundry room area, in %
T4, Temperature in office room, in Celsius
RH_4, Humidity in office room, in %
T5, Temperature in bathroom, in Celsius
RH_5, Humidity in bathroom, in %
T6, Temperature outside the building (north side), in Celsius
RH_6, Humidity outside the building (north side), in %
T7, Temperature in ironing room , in Celsius
RH_7, Humidity in ironing room, in %
T8, Temperature in teenager room 2, in Celsius
RH_8, Humidity in teenager room 2, in %
T9, Temperature in parents room, in Celsius
RH_9, Humidity in parents room, in %
To, Temperature outside (from Chievres weather station), in Celsius
Pressure (from Chievres weather station), in mm Hg
RH_out, Humidity outside (from Chievres weather station), in %
Wind speed (from Chievres weather station), in m/s
Visibility (from Chievres weather station), in km
Tdewpoint (from Chievres weather station), Â°C
rv1, Random variable 1, nondimensional
rv2, Random variable 2, nondimensional

OBJECTIVE OF THE ASSIGNEMNT:

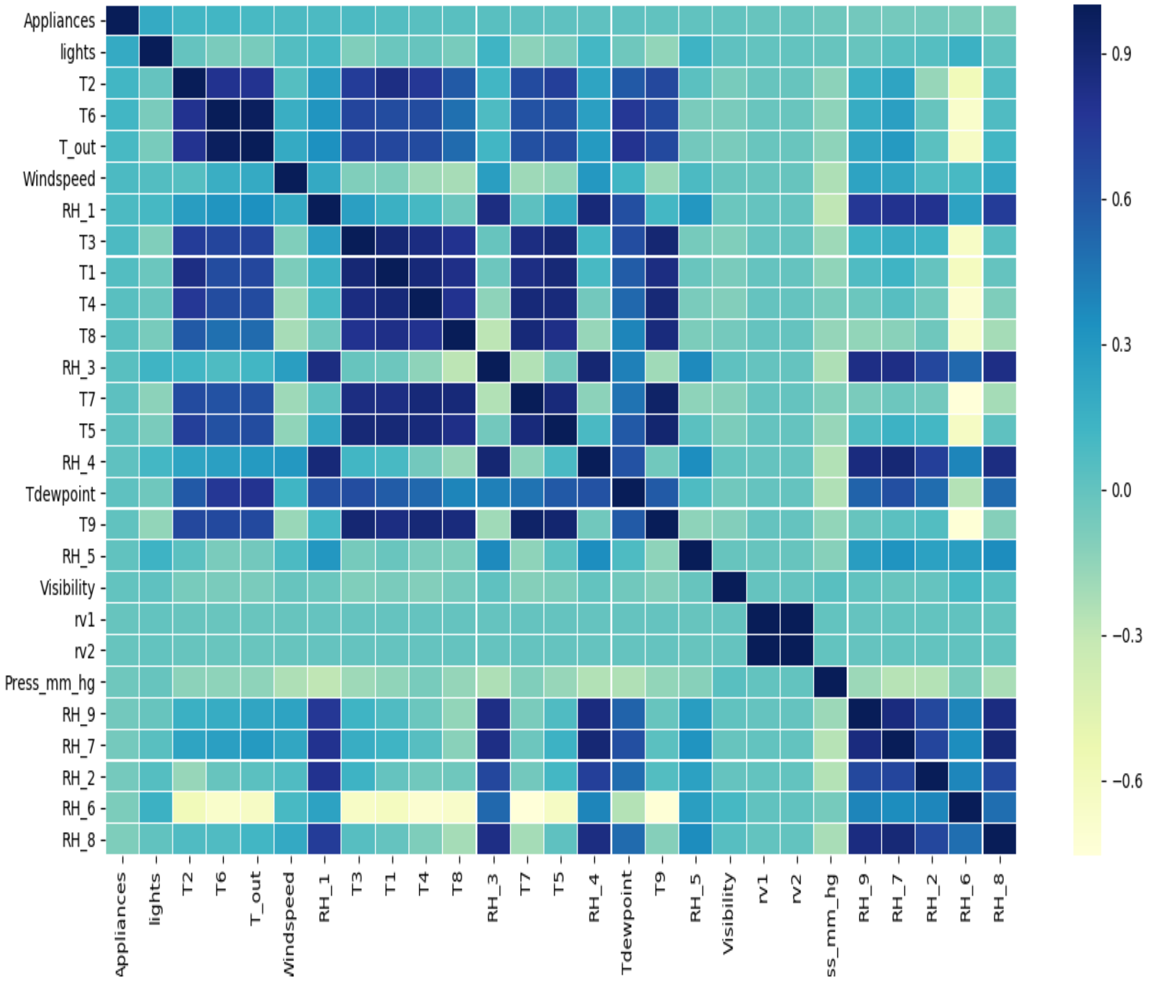
To predict the energy of the appliances by creating linear regression models given all the other factors. Gradient descent algorithm was chosen for this assignment

TARGET VARIABLE: Appliances

INITIAL DATA ANALYSIS:

The dataset does not contain any missing values and the date column has been removed from the regression model

Correlation plot:



We can see the above plot and the correlation between the variables. We can see that none of the variables are strongly correlated with Appliances. But we can clearly see that the variables are correlated among themselves.

TASKS PERFORMED:

PART 1:

The dataset has been partitioned into train and test sets in the ratio 70:30 which is the ideal ratio to divide the dataset.

PART 2:

For the regression model, I have chosen 17 variables after analyzing the correlation and choosing the ones that are at least weakly correlated with the target variable. Below is the regression model which I have used in the model:

Appliances = β_0+ β_1 *lights +β_2*T1+ β_3*RH_1+ β_4*T2+ β_5*RH_2+ β_6*T3+ β_7*RH_3+ β_8*T4+ β_9*T6+ β_{10}*RH_6+ β_{11}*RH_7+β_{12}*T8+ β_{13}*RH_8+ β_{14}*RH_9+ β_{15}T_out+β_{16}*RH_out+ β_{17}*Windspeed

The initial parameter values from the model with a learning rate of 0.1 is as follows:

B_0	97.33313966
B_1	19.89706141
B_2	-5.99278839
B_3	51.74643759
B_4	-28.58865711
B_5	-49.59298297
B_6	40.48436245
B_7	20.08105871
B_8	15.01914617
B_9	-28.15605486
B_{10}	11.52342603
B_{11}	-7.01955839
B_{12}	8.81713708
B_{13}	-21.85383021
B_{14}	-7.27336736
B_{15}	-20.52922437
B_{16}	-4.46864856
B_{17}	4.04319098

Train set error: 4387.137082220709

Test error: 4541.486440908859

Part 3:

The linear regression was implemented with batch update rule as follows:

$j=0,1,\ldots\ldots,n$

Simultaneous update

$$\beta_j := \beta_j - \alpha \frac{\partial}{\partial \beta_j} J(\beta)$$

$$\beta_j := \beta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(\hat{y}^{(i)} - y^{(i)}\right) x_j^{(i)}$$

The Gradient Descent was used as cost function:

$$J(B) = \frac{1}{2m} \sum_{i=1}^{m} \left(\hat{y}^{(i)} - y^{(i)}\right)^2$$

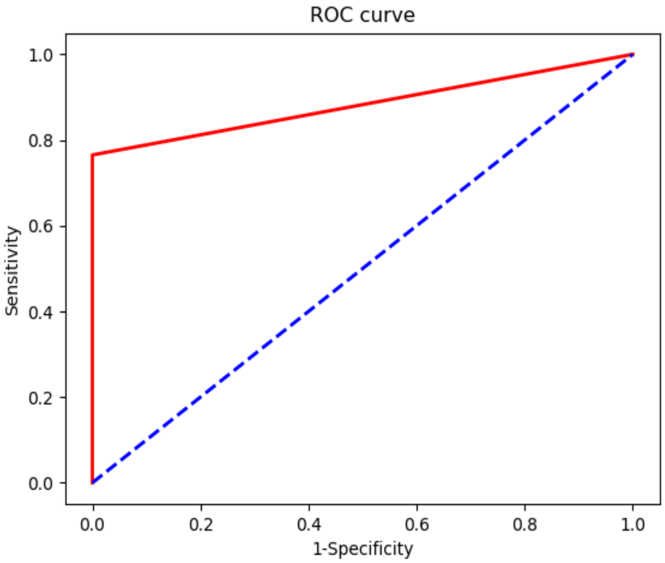
PART 4:

The dataset was converted into binary classification model with a new column created with the corresponding classification. Initially, I summarized the Appliances column and I found out that the median value was at 60. So, the values equal to and above 60 were classified as **above average** and those below 60 was **below average**. Upon trying out different threshold values, I found out that the model does well on very low values of threshold such as 0.05,0.1. On increasing the threshold, the model reaches 100% accuracy which in not ideal. For threshold value such as 0.05, the accuracy was measured to be **80.76%.** Below is the confusion matrix and ROC curve for the model. WE can see from the ROC curve that the curve is more inclined towards the top left corner indicating that model is performing well.

Confusion Matrix

1072	1139
0	3710

ROC CURVE

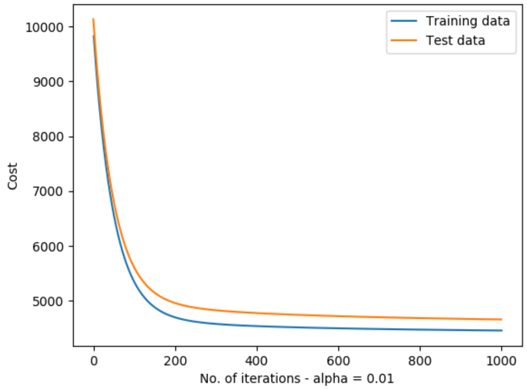


EXPERIMENTATION:

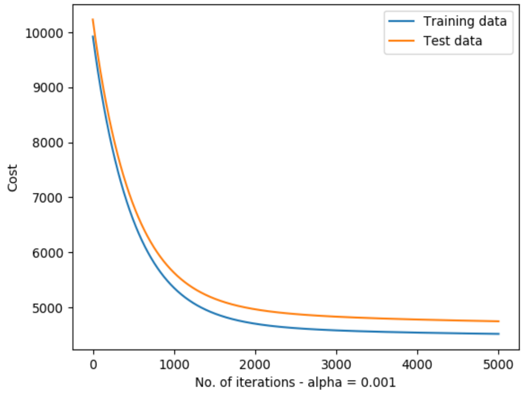
EXPERIMENTATION 1:

For this experiment, different learning rates(alpha) were chosen -0.1,0.01,0.001,0.0001 – and the respective training and test error was plotted as a function of iterations.

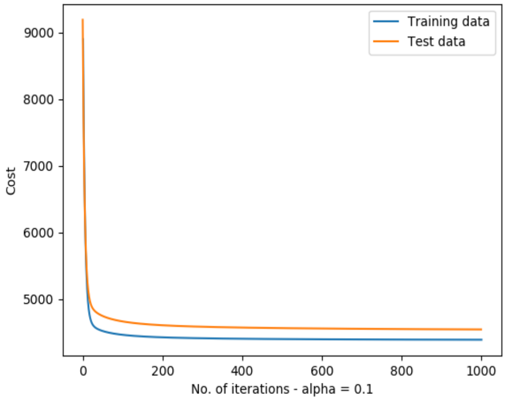
α=0.1



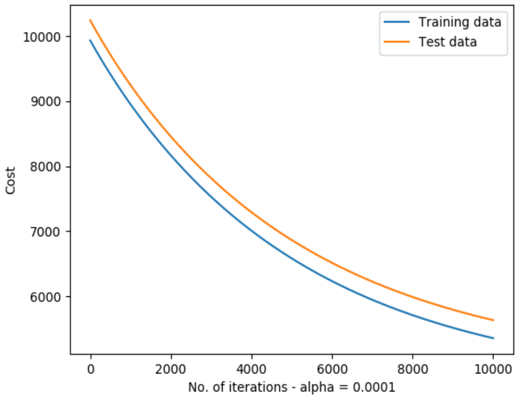
α = 0.01



α = 0.001



α = 0.0001



We can see that the number of iterations is less for higher learning rates (0.1,0.01) and it keeps increasing as the learning rate is decreased to 0.001 and 0.0001. For α= 0.0001, we can see that even for 10000 iterations, the curve does not converge and still tries to reach a global minimum. We can see that for all learning rates, the test error is always higher than the training error.

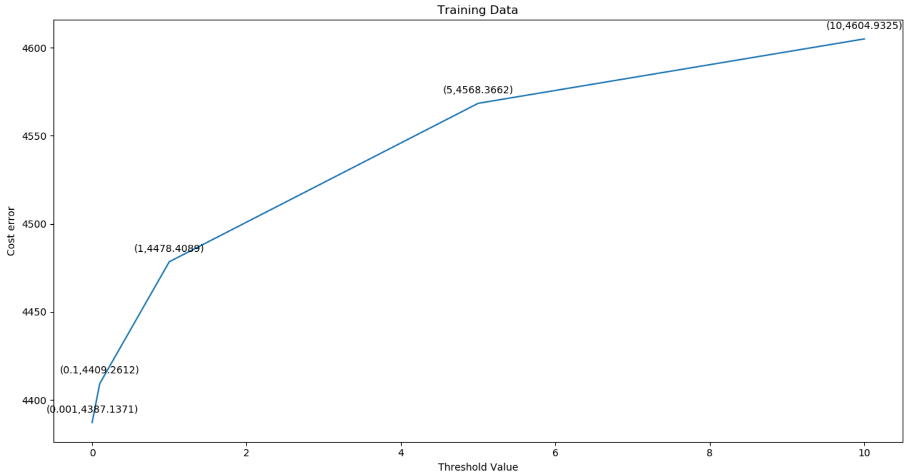
The training and test errors at which at curve reaches a minimum is given below for each learning rate.

Alpha	0.1	0.01	0.001	0.0001
Iterations	1000	1000	5000	10000
Train error	4387.14	4461.34	4518.84	5357.04
Test error	4541.49	4663.12	4747.63	5633.71

From the table, we can see that the training error is the lowest at 4387.14 for α=0.1 and test error is 5341.49. Therefore, we fix α=0.1 as our best learning rate for this model. Also, the number of iterations for convergence seem to be very low of all.

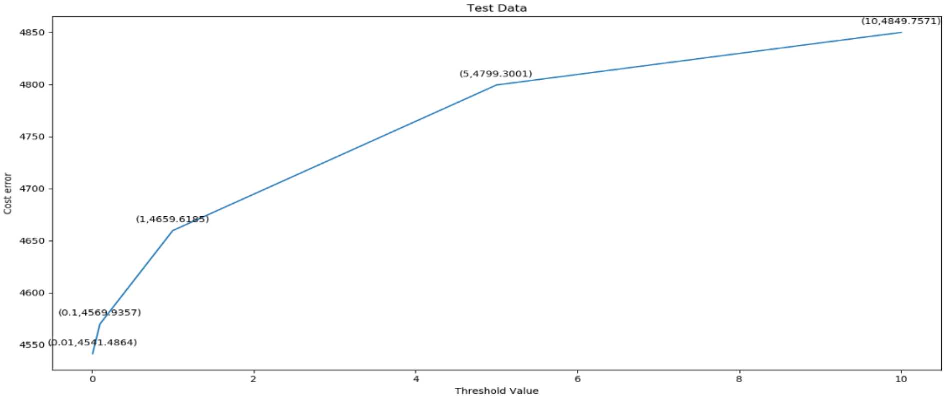
EXPERIMENT 2:

We have used the same model that we used in experiment 1. The alpha is kept at 0.1 for all the threshold values. I have run the gradient descent algorithm for various threshold values like 10,5,1,0.1,0.01,0.001,0.0001.



The above plot shows cost error as a function and we can see that the error gradually decreases and reaches a minimum point at threshold 0.01 after which it remains more or less constant.

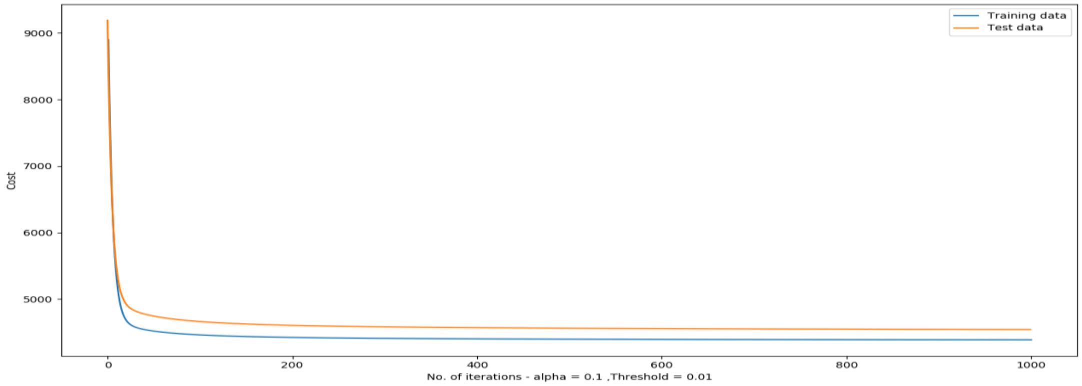
Similarly, I have plotted the same graph for the test data.



Threshold value	Training Cost error	Test Cost error
10	4604.93	4849.75
5	4568.37	4799.30
1	4478.41	4659.61
0.1	4409.26	4569.94
0.01	4387.74	4541.49
0.001	4387.14	4541.49
0.0001	4387.14	4541.49

Therefore, I choose 0.01 as the best threshold value.

Below, I have run the same model in experiment 1 with the learning rate as alpha=0.1 and threshold value = 0.01.



Hence the optimum threshold value helps us to get the best possible convergence point which in turn helps us to get the best possible beta values.

EXPERIMENT 3:

In this experiment, I have chosen 10 randomly selected variables by using random function in Python.

They are as follows:

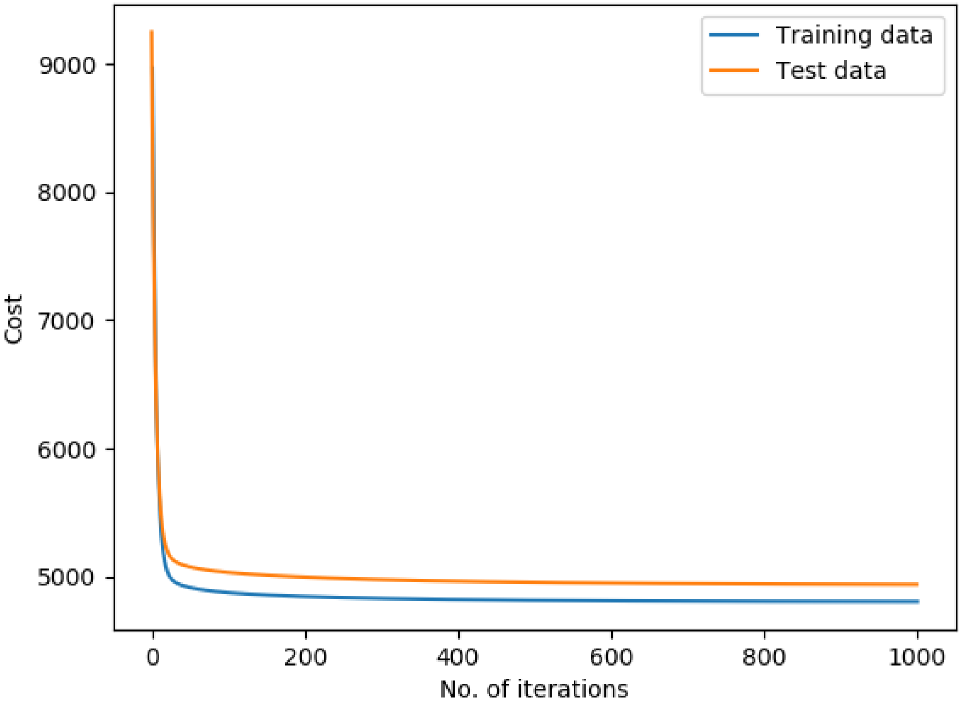
- T1
- RH1
- T2
- T3
- RH3
- T4
- T5
- T7
- RH7
- Tout

By running the model with these randomly selected features, we get the following training and test error:

Training error - 4807.998882518154

Test error - 4941.105824792052

We get the coefficients as below which are in the order given above.
[97.33313966, 26.75433502, -49.71275857, -12.14115734, -9.70644895, -19.08302949, -30.29233106, 53.18163163, 6.15913976, 16.94178676, ,35.61611149]



We can see that the training and the test error are certainly higher than the original model with 17 features.

Experiment 4:

In this experiment, I have chosen 10 variables on the basis of correlation and on common sense knowledge.

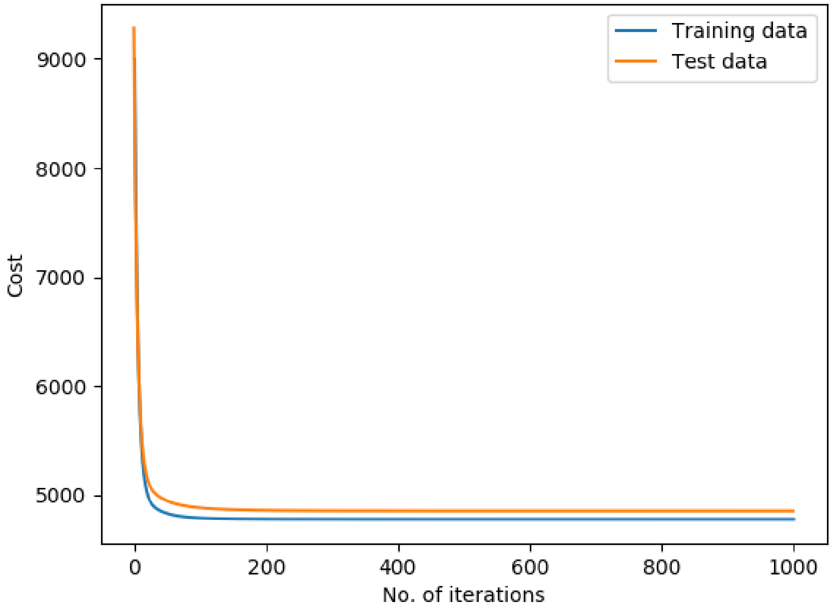
- T1
- RH1
- T2
- T3
- RH3
- T4
- T5
- T7
- RH7
- Tout

On seeing the data, obviously kitchen, living room, laundry room, office, bathroom, iron room and outside temperature seem to be important as these areas have lot of electrical appliances. Also, kitchen, laundry and iron room having high energy electrical devices like microwaves, washer, dryer, iron box etc. will consume lot of energy and hence I have included the humidity of these rooms in the list. On running the model with these features, we get the following train and test error:

Training error - 4780.08

Test error - 4856.73

We get the coefficient estimates which is in the same order as above:
[97.33313966, -10.09259221, 24.48042182, 23.70348754, 25.65696629, 9.69327706, -8.23546923, -19.25047814, -3.49560905, -39.33213263, 1.64625238]



Now the training and test error are certainly lower than that of the randomly selected features in experiment 3 but still is not as good as the original model with 17 features. It performs better than the random model because in this model, the variables are carefully selected with common sense knowledge and hence the error is a bit lower. But this model has a higher error rate than the original model because the original model has a lot of features (17) and hence can explain better. Also, the correlation plays a factor in the first model since it had a lot of correlated variables. I didn't account for correlation in this experiment and hence might have resulted in a higher error. Also, a nonlinear model with polynomial features might have given a lower error rate.

INTERPRETATIONS AND SUMMARY:

We can clearly see that learning rate is very critical in this model as a lower learning rate can result in the number of iterations to increase to a large extent. Hence finding the correct learning rate and threshold value can work a long way in reducing the training and test error rate as much as possible.

Also as seen from the experiment 4, choosing the most relevant variables is the most important aspect of a model. Feature selection must be done so that it reduces the bias – variance tradeoff and does not cause underfitting or overfitting. The model also must not be made complex by adding too many features.

In conclusion, I would say that choosing the correct learning rate and feature selection are the most important aspects of prediction.

OTHER STEPS WHICH CAN BE TAKEN TO IMPROVE THE MODEL:

Cross validation can be performed to further fine tune the model before testing the model with the test data so that the mean squared error can be decreased further.

For feature selection, various techniques such as forward, backward and hybrid methods can be used to select the most important parameters for the model. Also, the outliers have to be removed so that the variables are not skewed and biased.

If we feel that a linear model is not sufficient, we can move to a nonlinear model such as a polynomial model or a log model to increase the performance of the model. Sometimes these models can help reduce overfitting and underfitting. We can also use interaction terms if we feel that the explanatory variables have any synergy effect on the output variables. This can also improve the model to a great extent.

THE BEST MODEL WITH α=0.1 AND THRESHOLD = 0.01:

Appliances = 97.333+ 19.897 *lights -5.993*T1+ 51.746*RH_1-28.589*T2-49.593*RH_2+ 40.484*T3+ 20.081*RH_3+15.019*T4+28.156*T6+11.523*RH_6-7.020*RH_7+8.817*T8-21.854*RH_8-7.273*RH_9 - 20.529*T_out-4.469*RH_out+ 4.043*Windspeed