Predict Parameters for Efficient Performance Buildings (EPB)

Sanjive Kumar & Musa T. Ganiyu

City University New York

DATA 621: Final Project

Abstract

Based on the Energy efficiency Data Set from UCI Machine Learning Repository, we tried developing a statistical machine learning model to study the effect of eight input variables (relative compactness, surface area, wall area, roof area, overall height, orientation, glazing area, glazing area distribution) on two output variables, namely heating load (HL) and cooling load (CL), of residential buildings to make it a smart building with optimized heating and cooling load which eventually saves energy and overall cost for any company or individual.

We performed systematic analysis to find the association strength of each input variable with each of the output variables using Generalized Linear Regression. Then we refined the model using forward and backward model selection which has optimized the Generalized Linear Regression and provided the strongly related input variables for response variables heating load (HL) and cooling load (CL), of residential buildings. Regression on 768 diverse residential buildings show that we can predict HL and CL for best model having lowest AIC, Null and Residual deviance and significant p value (<0.05). With this study we can establish the fact that Machine Learning can be leveraged to estimate the building parameters easily, accurately and consistently across globe and apply it for new regions too where getting the benchmark data is not feasible.

Our best models for heating load (HL – Y1) and cooling load (CL- Y2), based on Linear Regression are:

$Y1=19.151577-17.950712*RelativeCompactness-0.029736*SurfaceArea+0.030449*WallArea+1.072304*RoofArea+7.100736*GlazingArea+GlazingAreaDistributionx1+GlazingAreaDistributionx5+0.709064*CoolingLoad+e$

$Y2=10.995725-6.463506*RelativeCompactness-0.011112*WallArea+0.864825*RoofArea+7.100736*GlazingArea+GlazingAreaDistributionx1+GlazingAreaDistributionx5+0.858350*HeatingLoad+e$

Keywords: Smart Buildings, Energy efficient Buildings, Save Energy, Statistical Modeling, Machine Learning,

Introduction

Energy is pivotal for all the things in this world and as we are growing in population the demand is exponentially increasing, on the other side supply is diminishing and becoming more and more expensive. Energy we are consuming comes in various form such as residual energy such as coals and petroleum and natural energy sources such wind and solar. All of them converge to cater our demand for energy. Lot of emphasis can also be seen in saving energy and building a self-reliant society. Our major consumption of energy is on maintaining livable condition inside ay building which is using heating, ventilation, air conditioning (HVAC) and optimizing these parameters is for a building also known as energy performance buildings(EPB).

When we consider HVAC as core to the energy performance building, we should consider heating load (HL)and cooling load (CL) as 2 key aspects of EPB. There are many companies who has introduced energy simulation tools in the market to profile the energy need based on certain key parameters. For this study we have considered UCI Machine Learning Repository[5], Energy efficiency data set whose key input and output variables are as shown below:

Variable Names	Input & Output variables
X1	Relative Compactness

X2	Surface Area
X3	Wall Area
X4	Roof Area
X5	Overall Height
X6	Orientation
X7	Glazing area
X8	Glazing area distribution
Y1	Heating Load
Y2	Cooling Load

Significant shift has been observed in past decade due to high acceptance of machine learning in various areas, energy simulation is also one among that where we have considered building features and dimensions as key to predict the energy saving forecast.

Literature review

In past, many companies have introduced the energy simulation tools which are well known among consumers and used to determine the energy consumption of any building. They are quiet often shows inaccuracy [1] due to old heuristic data and set demography or condition under which the tool was designed. Martin They are generally outdated due to many environmental conditions such as Lighting [2] consumption by various sections of the society, envelop talks about the weather/ environmental condition as energy will be consumed more if people are in cold region than hot regions or moderate climate. Machine learning gives power to predict to common man without any sophisticated toolsets. And the parameter which is now being considered are also not based out of stale conditions or old heuristic data, they are key variables related with building feature and dimensions which will change and provide a more realistic prediction based as and when needed, we don't have to rely on heuristic data to predict the energy consumption need.

We understand the overall energy performance building guidelines [3] in US to design an energy efficient building is provided by United States Department of Energy by Arlan Burdick which advocates the Heating and cooling load calculations are dependent on the building location, indoor design conditions, orientation, and building construction. This provides similar guideline as our current study which will help in comparing the results. Fumo, Mago, and Luck et al. (2010) [6] used the DOE's EnergyPlus Benchmark models (U.S. DOE EERE 2011) to developed a series of EnergyPlus normalized consumption coefficients (ENECC) that can be used to estimate hourly building energy consumption from utility bill information. Karaguel and Lam et al. (2011) provided a simplified approach[4] using hourly energy profiles which are not typically available for existing buildings in practice. The authors argued that having pre-determined coefficients (derived from actual data and energy simulation models of typical U.S. buildings) could relieve the user from the burden of performing a detailed dynamic simulation. Our study on energy performance buildings will be in line with the US Department of Energy's guideline considering building related variables as most strong inputs to predict Heating Load and cooling.

Methodology

To develop a statistical framework for determining the high energy performance smart buildings we used the given dataset and initiated analysis with our 4 step process as:

Data Exploration:

In this we just perform descriptive analysis of the data and identify all gaps or transformation opportunity to prepare the data for building models. So while going through this phase using Energy efficiency data set, we found that :

- there are 768 observations and 10 variables. Out of 10 variables, there are 2 response variables (Y1 : Heating Load & Y2: Cooling Load) and 8 dependent variables or predictors(X1,X2…X8), more description for each of the predictor variable is as shown in table above
- There are no missing values across all 10 variables
- The correlation coefficient of all predictor variables with Y1 is X2(Surface Area),X4 (Roof Area) and X6(Orientation) are negatively correlated and X1(Relative Compactness),X3(Wall Area),X5(Overall Height),X7(Glazing Area),X8(Glazing area distribution),Y2(Cooling Load) have positive correlation which means if X1,X3,X5,X7,X8 and Y2 increased it will increase the Heating Load also and if X2,X4 and X6 increased it will reduce the Heating Load
- The dataset is approximately symmetric as the skewness lies between -0.5 to 0.5
- After analyzing the data closely it seems like X6 and X8 are variable which needs to be converted to dummy variables in next phase due to their categorical type data, although it is showing as int in the data description. Conversion to dummy variable will help more in understanding their relationship with response variables (Y1 and Y2)

Data Preparation:

Based on the Data Exploration recommendation, we will use 2 format first one will be long where we will not do any data transformation and use the dataset as is for Generalized Linear Method and then perform step wise method to select the best fit model and significant variables for Y1(Heating Load) and Y2 (Cooling Load) and for the Model 2 we will transform the data to Wide format for input variables X6 and X8. Wide format will help to understand the orientation and glazing area distribution impact on Heating and Cooling load to given categorical value which will be more precise compare to long format

Build Models:

This phase we will be building the model from the clean dataset which we got from Data Preparation phase. Key steps in building models are:

- Split the clean data set into train and test as 75:25 ratio
- Apply Generalized Linear Method regression for Y1 and Y2 and get the model, variables which has significance value <0.05 are selected to perform the step wise method with backward regression and get the best fit model
- Repeat the same process for Wide format also, the only difference would be we need to transform the dataset to wide format for X6 and X8
- Once we have best fit equation and variables we will test it using ANOVA and checking their confidence interval. ANOVA provides the confirmation from $p<0.05$ for all the significant variables and confidence intervals for all the variables will not have zero in between.
- Run the test dataset against best fit model and get the predicted values

Select Models:

Selecting models is mostly comparing their key statistics which can qualify them to be best are deviance (Null and Residual) is a measure of goodness of fit of a generalized linear model lower the deviance better will be the model , AIC - **Akaike information Criterion** is a measure of the relative quality of statistical models for a given set of data, lower the AIC better the model would be. Based on these 2 criteria, the best model is selected

Experimentation and Results

We adopted the 4 steps method to perform complete data analysis and recommendation as shown below:

1. DATA EXPLORATION:

The dataset link here : http://archive.ics.uci.edu/ml/datasets/Energy+efficiency

Attribute Information:

The dataset contains eight attributes (or features, denoted by X1...X8) and two responses (or outcomes, denoted by y1 and y2). The aim is to use the eight features to predict each of the two responses.

```
## 'data.frame':    768 obs. of  10 variables:
##  $ X1: num  0.98 0.98 0.98 0.98 0.9 0.9 0.9 0.9 0.86 0.86 ...
##  $ X2: num  514 514 514 514 564 ...
##  $ X3: num  294 294 294 294 318 ...
##  $ X4: num  110 110 110 110 122 ...
##  $ X5: num  7 7 7 7 7 7 7 7 7 7 ...
##  $ X6: int  2 3 4 5 2 3 4 5 2 3 ...
##  $ X7: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ X8: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Y1: num  15.6 15.6 15.6 15.6 20.8 ...
##  $ Y2: num  21.3 21.3 21.3 21.3 28.3 ...
```

Data Description for Energy Efficiency data set:

	Vars	N	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se	NAs	corsY1	corsY2
X1	1	768	0.764	0.1057775	0.75	0.7564935	0.118608	0.62	0.98	0.36	0.4935786	-0.7157384	0.0038169	0	0.6222722	0.6343391
X2	2	768	671.708	88.0861161	673.75	673.7500000	108.971100	514.50	808.50	294.00	-0.1246425	-1.0654194	3.1785339	0	-0.6581202	-0.6729989
X3	3	768	318.500	43.6264814	318.50	315.9545455	36.323700	245.00	416.50	171.50	0.5313356	0.0999447	1.5742351	0	0.4556712	0.4271170
X4	4	768	176.604	45.1659502	183.75	179.1363636	54.485550	110.25	220.50	110.25	-0.1621288	-1.7763946	1.6297858	0	-0.8618283	-0.8625466
X5	5	768	5.2500	1.7511404	5.25	5.2500000	2.594550	3.50	7.00	3.50	0.0000000	-2.0026025	0.0631888	0	0.8894307	0.8957852
X6	6	768	3.500	1.1187626	3.50	3.5000000	1.482600	2.00	5.00	3.00	0.0000000	-1.3642681	0.0403699	0	-0.0025865	0.0142896
X7	7	768	0.234	0.1332206	0.25	0.2383117	0.222390	0.00	0.40	0.40	-0.0600191	-1.3311582	0.0048072	0	0.2698410	0.2075050

X8	8	768	2.812	1.5509597	3.00	2.8441558	1.482600	0.00	5.00	5.00	-0.0883430	-1.1538633	0.0559654	0	0.0873676	0.0505251
Y1	9	768	22.307	10.0901957	18.95	21.7143994	11.141739	6.01	43.10	37.09	0.3590421	-1.2498464	0.3640986	0	1.0000000	0.9758618
Y2	10	768	24.58	9.5133056	22.08	23.9455032	11.178804	10.90	48.03	37.13	0.3944470	-1.1523586	0.3432818	0	0.9758618	1.000000

With descriptive analysis of the data, we will be able to identify all gaps or transformation opportunity to prepare the data for building models. So while going through this phase using Energy efficiency data set, we found that:

- There are 768 observations and 10 variables. Out of 10 variables, there are 2 response variables (Y1 : Heating Load & Y2: Cooling Load) and 8 dependent variables or predictors(X1,X2…X8), more description for each of the predictor variable is as shown in table above
- There are no missing values across all 10 variables
- The correlation coefficient of all predictor variables with Y1 is X2(Surface Area),X4 (Roof Area) and X6(Orientation) are negatively correlated and X1(Relative Compactness),X3(Wall Area),X5(Overall Height),X7(Glazing Area),X8(Glazing area distribution),Y2(Cooling Load) have positive correlation which means if X1,X3,X5,X7,X8 and Y2 increased it will increase the Heating Load also and if X2,X4 and X6 increased it will reduce the Heating Load
- The dataset is approximately symmetric as the skewness lies between -0.5 to 0.5
- After analyzing the data closely it seems like X6 and X8 are variable which needs to be converted to dummy variables in next phase due to their categorical type data, although it is showing as int in the data description. Conversion to dummy variable will help more in understanding their relationship with response variables (Y1 and Y2) `

2. DATA PREPARATION

Based on the Data Exploration recommendation, we will transform X6 and X8 to wide format as dummy variables and remove X6 and X8 from the dataset.

```
dat <- mutate(dat, id = rownames(dat))
dat$id = as.numeric(dat$id)
dat$X6 <- factor(dat$X6);
dat$X8 <- factor(dat$X8);
newdfx6<-dcast(dat, id  ~ X6,length)

## Using 'id' as value column. Use 'value.var' to override

newdfx6<-newdfx6[order(newdfx6$id),]
total <- merge(dat,newdfx6,by="id")
newdfx8<-dcast(dat, id  ~ X8,length)

## Using 'id' as value column. Use 'value.var' to override

newdfx8<-newdfx8[order(newdfx8$id),]
dataset1<-merge(total,newdfx8, by="id")
dat <- dat[,-11 ]
dataset1=dataset1[(c(-1,-7,-9))]
head(dataset1)

colnames(dataset1) <- c("X1", "X2", "X3", "X4" , "X5", "X7", "Y1", "Y2", "6X_2", "6X_3", "6x_4","6X_5",
"8X_0", "8X_1", "8X_2", "8X_3", "8X_4", "8X_5")
head(dataset1)
```

3. BUILD MODELS

This phase we will be building the model from the clean dataset which we got from Data Preparation phase. First step in building models is to split the clean data set into train and test as 75:25 ratio

Now we are ready to build our first model for Y1(Heating Load) using Generalized Linear Model

MODEL 1 [Y1 HEATING LOAD]: Generalized Linear Model

We would first obtain the glm of Y1 (response variable) against all other explanatory variables excluding Y2 (Cooling Load)

```
## glm(formula = Y1 ~ . - Y2, data = trainsdat)
```

We can see that only X1, X2, X3,X5,X7 and X8 are statistically significant as they have p-values less than 0.05, and null deviance is quiet high compared to residual deviance which suggest that we need to further optimize the model for lesser deviance and low AIC value. So we proceed further for selection of best fit model using backward model selection.

```
## glm(formula = Y1 ~ X1 + X2 + X3 + X5 + X7 + X8, data = trainsdat)
```

So after performing the backward selection on the significant variables, we observed that the best fit model is confirmed for Y1 target variable and they have X1,X2,X3,X5,X7 and X8 with AIC as 2806. So lets run the glm model once again using the significant input variables and get the best fit model for Y1 (Heating Load)

```
## glm(formula = Y1 ~ X1 + X2 + X3 + X5 + X7 + X8, data = trainsdat)
```

It shows that we have a significant model with P-value less than 0.05, let’s test it with Analysis of Variance

Analysis of Variance (ANOVA)

anova(besty2model, test="F")

The ANOVA revealed and confirmed that the variable selected was indeed the best model for prediction as all variables are significant $p<0.05$ and null deviance is same, however residual deviance has reduced drastically because of the variance considered between variables.

Let's confirm the best fit model with confidence intervals too as:

```
confint(besty2model)
```

Above confident interval also confirm the model to be significance for prediction and none of the interval differences is equal to zero. So the best fit equation will be:

MODEL 1: Best Fit Equation

$HeatingLoad = 86.661793 - 66.762900 * RelativeCompactness - 0.092251 * SurfaceArea + 0.062780 * WallArea + 4.026343 * OverallHeight + 16.309679 * GlazingArea + GlazingAreaDistribution + GlazingAreaDistributionx5 + e$

Now using the above equation we can very well predict the Y1(Heating Load) for any building and make it more smarter, so we will run the testdat against the best fit model equation and it will give the results as shown below:

X1	X2	X3	X4	X5	X6	X7	X8	pred
0.98	514.5	294.0	110.25	7.0	3	0	0	20.41294
0.98	514.5	294.0	110.25	7.0	5	0	0	20.41294

MODEL 1 [Y2 COOLING LOAD]: Generalized Linear Model

```
## glm(formula = Y2 ~ . - Y1, data = trainsdat)
```

From the above, we can see that only X1,X2,X3,X5,X7 and X8 are statistically significant as they have p-values less than 0.05, and null deviance is quiet high compared to residual deviance which suggest that we need to further optimize the model for lesser deviance and low AIC value. So we proceed further for selection of best fit model using backward model selection.

```
## glm(formula = Y2 ~ X1 + X2 + X3 + X5 + X7 + X8, data = trainsdat)
```

So after performing the backward selection on the significant variables, we observed that the best fit model is confirmed for Y1 target variable and they have X1,X2,X3,X5,X7 and X8 with AIC as 2934. So lets run the glm model once again using the significant input variables and get the best fit model for Y2 (Cooling Load)

```
## glm(formula = Y2 ~ X1 + X2 + X3 + X5 + X7 + X8, data = trainsdat)
```

It shows that we have a significant model with p-value less than 0.05, let’s test it with Analysis of Variance

Analysis of Variance (ANOVA)

anova(besty2model, test="F")

The ANOVA revealed and confirmed that the variable selected was indeed the best model for prediction as all variables are significant $p<0.05$ and null deviance is same, however residual deviance has reduced drastically because of the variance considered between variables.

Let's confirm the best fit model with confidence intervals too as:

confint(besty2model)

Above confident interval also confirm the model to be significance for prediction and none of the interval diffenrences is eqaul to zero. so the best fit equation will be:

Best Fit Model Equation for Y2 Cooling Load:

The best model includes: X1, X2, X3, X5, X7 and X8

$CoolingLoad=95.210349-68.840330*RelativeCompactness-0.088165*SurfaceArea+0.045596*WallArea+4.166112*OverallHeight+12.987467*GlazingArea+GlazingAreaDistribution+GlazingAreaDistributionx5+e$

Now using the above equation we can very well predict the Y2(Cooling Load) for any building and make it more smarter, so we will run the testdat against the best fit model equation and it will give the results as shown below:

X1	X2	X3	X4	X5	X6	X7	X8	pred
0.98	514.5	294.0	110.25	7.0	3	0	0	24.95414
0.98	514.5	294.0	110.25	7.0	5	0	0	24.95414

The above is the predicted value (pred columns) for Y1 which can be interpreted as if X1=0.98, X2=514.5,X3=294, X4=110.25, X5=7, X6=3 then using the best fit equation for cooling load will be 24.95

So we observed in Data Exploration phase that X6 and X8 are although numeric datatype they have categorical value and can be transformed to Wide format as shown below:

MODEL 2 : Y1 (Heating Load) WIDE FORMAT (X6, X8)

Now running the GLM regression for Y1 and Y2 and performing stepwise method to select the best model for Y1 and Y2

```
## glm(formula = Y1 ~ . - Y2, data = trainsdat_2)
## glm(formula = Y1 ~ X1+X2+X3+X5+X7+`8X_0`+`8X_4,data = trainsdat_2)
```

MODEL 2 : Y2 (Cooling Load) WIDE FORMAT (X6, X8)

```
coolingload_2 <- glm(Y1 ~.-Y2, data=trainsdat_2);
coolingload_2 <- step(glm(Y2 ~.-Y1,data=trainsdat_2),direction = "backward");

MOD2Y1<-c('Model_WY1',58768.5,4225.6,2812.5,'X1,X2,X3,X5,X7,8X_0')
MOD2Y2<-c('Model_WY2',51843.1,5284.6,2929.3,'X1,X2,X3,X5,X7,8X_0')
```

we can deduce that the best fit model for Y1 (Heating Load) has X1, X2, X3, X5,X7 and 8X_0 are statistically significant having $p<0.05$, Null deviance: 58768.5, Residual deviance: 4225.6 and AIC: 2812.5

And, for Y2 (Cooling Load) has X1, X2, X3, X5,X7 and 8X_0 are really statistically significant having $p<0.05$, Null deviance: 51843.1, Residual deviance: 5284.6 and AIC: 2929.3

4. SELECT MODEL

Since we have both the models in place we can compare side wise their prediction capabilities and also their deviance and AIC to select the best model which has low deviance and AIC and higher prediction rate

Side-by-side comparison of selection criteria

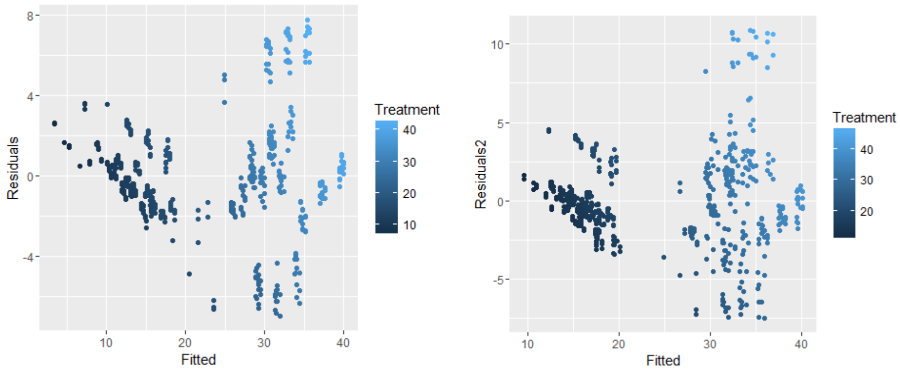
```
MOD_STAT<- cbind(MOD1Y1,MOD2Y1,MOD1Y2,MOD2Y2)
names(MOD_STAT)=c("Model_Name","Null_Deviance","Residual_Deviance","AIC","Significant Vars")
kable(MOD_STAT)
```

	MOD1Y1	MOD2Y1	MOD1Y2	MOD2Y2
Model	Model_1LY1	Model_WY1	Model_LY2	Model_WY2
Null Dev	51843.1	58768.5	51843.1	51843.1
Res Dev	5266.4	4225.6	5276.8	5284.6
AIC	2939.3	2812.5	2934.4	2929.3
Sig. Vars	X1,X2,X3,X5,X7,X8	X1,X2,X3,X5,X7,8X_0	X1,X2,X3,X5,X7,X8	X1,X2,X3,X5,X7,8X_0

Side-by-side predictions for long and wide format

Pred_HL_W	Pred_HL_L	Pred_CL_W	Pred_CL_L
20.417600	20.412937	24.95662	24.95414
20.417600	20.412937	24.95662	24.95414

A side-by-side plot between Fitted values and Residual for Heating Load & Cooling Load



A side-by-side plot between Fitted values and Residual for CoolingLoad confirms that both are similar distribution.

Discussion and Conclusions:

From above analysis, we saw that both response variables (CoolingLoad and HeatingLoad) have same best fit models that includes (X1, X2, X3, X5, X7 and X8), their p-values are statistically significant and have non-zero confidence interval. So we arrive to the conclusion that both the models are very much similar in nature, however going statistically where the AIC and Deviance should be low we will select Model 1 Long format compared to Model 2 Wide Format et al.

- Cooling Load: The rate at which a cooling system or process must remove heat from a conditioned zone to maintain it at a constant dry bulb temperature and humidity would increase by 2-5 units over period of time.

- Heating load: The quantity of heat per unit time that must be supplied to maintain the temperature in a building or portion of a building at a given level would increase by -3 to +3 over period of time

Limitations

This study is limited to building dimensions and features and not the external environmental factors like climate condition for specific regions. For example, if we are assessing a building in Bayarea CA, will have a different heating and cooling requirements then building at Redmond WA. If we consider that it will definitely have an impact on Heating Load and Cooling Load.

Future Work

This study needs more in depth analysis using region wise data and also considering specific building regulations into consideration for developing a complete predictive model applicable to any state or city of US. So we would like to extend this project and seek all the relevant information required for constructing building and then applying data analytics and modelling technique would be helpful and make it more realistic model.

References

1. http://www.greenbuildingadvisor.com/blogs/dept/musings/energy-modeling-isn-t-very-accurate
2. http://www.nrel.gov/docs/fy08osti/41956.pdf
3. http://www.nrel.gov/docs/fy11osti/51603.pdf
4. https://www.academia.edu/19771878/Development_of_whole-building_energy_performance_models_as_benchmarks_for_retrofit_projects
5. http://archive.ics.uci.edu/ml/datasets/Energy+efficiency#
6. https://uttyler.influuent.utsystem.edu/en/publications/methodology-to-estimate-building-energy-consumption-using-energyp

Appendices

R Code and detailed results:

```
The dataset link here : http://archive.ics.uci.edu/ml/datasets/Energy+efficie
ncy

## 'data.frame':    768 obs. of  10 variables:
##  $ X1: num  0.98 0.98 0.98 0.98 0.9 0.9 0.9 0.9 0.86 0.86 ...
##  $ X2: num  514 514 514 514 564 ...
##  $ X3: num  294 294 294 294 318 ...
##  $ X4: num  110 110 110 110 122 ...
##  $ X5: num  7 7 7 7 7 7 7 7 7 7 ...
##  $ X6: int  2 3 4 5 2 3 4 5 2 3 ...
##  $ X7: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ X8: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Y1: num  15.6 15.6 15.6 15.6 20.8 ...
##  $ Y2: num  21.3 21.3 21.3 21.3 28.3 ...
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se	NAs	cors Y1	cors Y2
X1	1	768	0.7641667	0.1057775	0.75	0.7564935	0.1186088	0.62	0.98	0.36	0.4935786	-0.7157384	0.0038169	0	0.6222722	0.6343391
X2	2	768	671.7083333	88.0861161	673.75	673.7500000	108.9711000	514.50	808.50	294.00	-0.1246425	-1.0654194	3.1785339	0	-0.6581202	-0.6729989
X3	3	768	318.5000000	43.6264814	318.50	315.9545455	36.323700	245.00	416.50	171.50	0.5313356	0.0999447	1.5742351	0	0.4556712	0.4271170
X4	4	768	176.6041667	45.1659502	183.75	179.1363636	54.485550	110.25	220.50	110.25	-0.1621288	-1.7763946	1.6297858	0	-0.8618283	-0.8625466
X5	5	768	5.2500000	1.7511404	5.25	5.2500000	2.5945500	3.50	7.00	3.50	0.0000000	-2.0026025	0.0631888	0	0.8894307	0.8957852

X6	6	768	3.5000000	1.1187626	3.50	3.5000000	1.482600	2.00	5.00	3.00	0.0000000	-1.3642681	0.0403699	0	-0.0025865	0.0142896
X7	7	768	0.2343750	0.1332206	0.25	0.2383117	0.222390	0.00	0.40	0.40	-0.0600191	-1.3311582	0.0048072	0	0.2698410	0.2075050
X8	8	768	2.8125000	1.5509597	3.00	2.8441558	1.482600	0.00	5.00	5.00	-0.0883430	-1.1538633	0.0559654	0	0.0873676	0.0505251
Y1	9	768	22.3072005	10.0901957	18.95	21.7143994	11.141739	6.01	43.10	37.09	0.3590421	-1.2498464	0.3640986	0	1.0000000	0.9758618
Y2	10	768	24.5877604	9.5133056	22.08	23.9455032	11.178804	10.90	48.03	37.13	0.3944470	-1.1523586	0.3432818	0	0.9758618	1.0000000

DATA PREPARATION ####Based on the Data Exploration recommendation, we will transform X6 and X8 to wide format as dummy variables and remove X6 and X8 from the dataset.

```
dat <- mutate(dat, id = rownames(dat))
dat$id = as.numeric(dat$id)

dat$X6 <- factor(dat$X6);
dat$X8 <- factor(dat$X8);


newdfx6<-dcast(dat, id  ~ X6,length)

## Using 'id' as value column. Use 'value.var' to override

newdfx6<-newdfx6[order(newdfx6$id),]

total <- merge(dat,newdfx6,by="id")
newdfx8<-dcast(dat, id  ~ X8,length)

## Using 'id' as value column. Use 'value.var' to override

newdfx8<-newdfx8[order(newdfx8$id),]


dataset1<-merge(total,newdfx8, by="id")

dat <- dat[,-11 ]
```

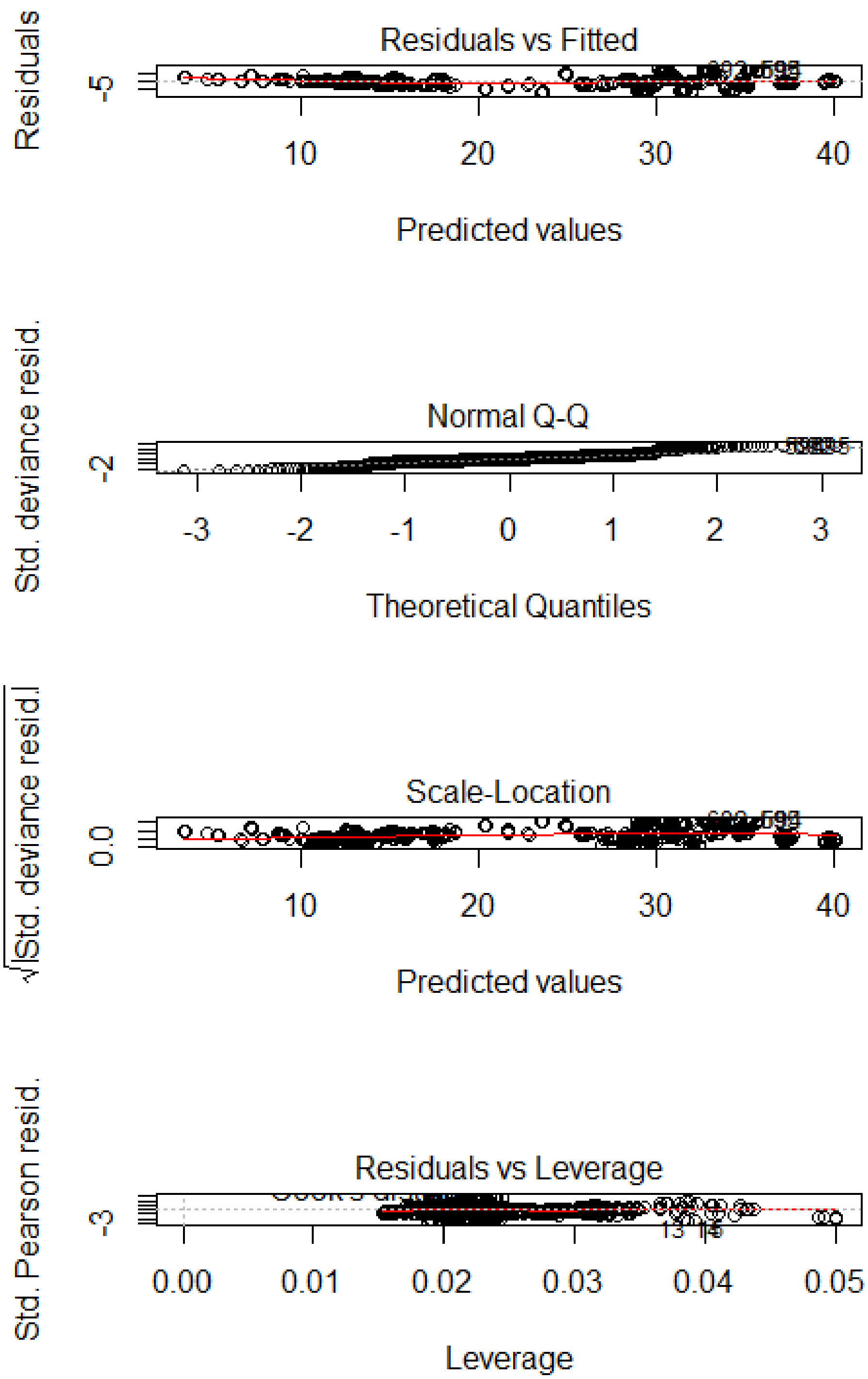
```
dataset1=dataset1[(c(-1,-7,-9))]
head(dataset1)

##      X1    X2    X3     X4 X5 X7    Y1    Y2 2.x 3.x 4.x 5.x 0 1 2.y 3.y
## 1 0.98 514.5 294.0 110.25  7  0 15.55 21.33   1   0   0   0 1 0   0   0
## 2 0.98 514.5 294.0 110.25  7  0 15.55 21.33   0   1   0   0 1 0   0   0
## 3 0.98 514.5 294.0 110.25  7  0 15.55 21.33   0   0   1   0 1 0   0   0
## 4 0.98 514.5 294.0 110.25  7  0 15.55 21.33   0   0   0   1 1 0   0   0
## 5 0.90 563.5 318.5 122.50  7  0 20.84 28.28   1   0   0   0 1 0   0   0
## 6 0.90 563.5 318.5 122.50  7  0 21.46 25.38   0   1   0   0 1 0   0   0
##   4.y 5.y
## 1   0   0
## 2   0   0
## 3   0   0
## 4   0   0
## 5   0   0
## 6   0   0

colnames(dataset1) <- c("X1", "X2", "X3", "X4" , "X5", "X7", "Y1", "Y2", "6X_
2", "6X_3", "6x_4","6X_5", "8X_0", "8X_1", "8X_2", "8X_3", "8X_4", "8X_5")

head(dataset1)

##      X1    X2    X3     X4 X5 X7    Y1    Y2 6X_2 6X_3 6x_4 6X_5 8X_0 8X_1
## 1 0.98 514.5 294.0 110.25  7  0 15.55 21.33    1    0    0    0    1    0
## 2 0.98 514.5 294.0 110.25  7  0 15.55 21.33    0    1    0    0    1    0
## 3 0.98 514.5 294.0 110.25  7  0 15.55 21.33    0    0    1    0    1    0
## 4 0.98 514.5 294.0 110.25  7  0 15.55 21.33    0    0    0    1    1    0
## 5 0.90 563.5 318.5 122.50  7  0 20.84 28.28    1    0    0    0    1    0
## 6 0.90 563.5 318.5 122.50  7  0 21.46 25.38    0    1    0    0    1    0
##   8X_2 8X_3 8X_4 8X_5
## 1    0    0    0    0
## 2    0    0    0    0
## 3    0    0    0    0
## 4    0    0    0    0
## 5    0    0    0    0
## 6    0    0    0    0

head(dat)

##      X1    X2    X3     X4 X5 X6 X7 X8    Y1    Y2
## 1 0.98 514.5 294.0 110.25  7  2  0  0 15.55 21.33
## 2 0.98 514.5 294.0 110.25  7  3  0  0 15.55 21.33
## 3 0.98 514.5 294.0 110.25  7  4  0  0 15.55 21.33
## 4 0.98 514.5 294.0 110.25  7  5  0  0 15.55 21.33
## 5 0.90 563.5 318.5 122.50  7  2  0  0 20.84 28.28
## 6 0.90 563.5 318.5 122.50  7  3  0  0 21.46 25.38

#contrasts(dat$X6) = contr.treatment(4)
#contrasts(dat$X8) = contr.treatment(6)
```

BUILD MODELS

```
## split_ins
## FALSE  TRUE
##   192   576

## [1] 0.75
```

MODEL 1 [Y1 HEATING LOAD]: Generalized Linear Model

```
##
## Call:
## glm(formula = Y1 ~ . - Y2, data = trainsdat)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -7.0109  -1.3914  -0.1186   1.2443   7.7082
##
## Coefficients: (1 not defined because of singularities)
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  86.67058   20.29951   4.270 2.30e-05 ***
## X1          -66.74819   10.95588  -6.092 2.06e-09 ***
## X2           -0.09226    0.01822  -5.063 5.60e-07 ***
## X3            0.06282    0.00717   8.761  < 2e-16 ***
## X4                 NA         NA      NA       NA
## X5            4.02461    0.36632  10.987  < 2e-16 ***
## X63           0.02885    0.32588   0.089    0.929
## X64          -0.08316    0.32294  -0.258    0.797
## X65          -0.01457    0.32637  -0.045    0.964
## X7           16.30929    0.96668  16.871  < 2e-16 ***
## X81           3.96182    0.58875   6.729 4.21e-11 ***
## X82           3.94913    0.58756   6.721 4.43e-11 ***
## X83           3.69451    0.58085   6.361 4.17e-10 ***
## X84           4.25586    0.58290   7.301 9.80e-13 ***
## X85           3.75414    0.58442   6.424 2.84e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 7.518788)
##
##     Null deviance: 58768.5  on 575  degrees of freedom
## Residual deviance:  4225.6  on 562  degrees of freedom
## AIC: 2812.5
##
## Number of Fisher Scoring iterations: 2
```




```
## Start:  AIC=2812.47
## Y1 ~ (X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8 + Y2) - Y2
##
##
```

```
## Step:  AIC=2812.47
## Y1 ~ X1 + X2 + X3 + X5 + X6 + X7 + X8
##
##         Df Deviance    AIC
## - X6    3   4226.6 2806.6
## <none>      4225.6 2812.5
## - X2    1   4418.3 2836.2
## - X1    1   4504.6 2847.3
## - X8    5   4650.5 2857.7
## - X3    1   4802.7 2884.2
## - X5    1   5133.1 2922.5
## - X7    1   6365.7 3046.5
##
## Step:  AIC=2806.6
## Y1 ~ X1 + X2 + X3 + X5 + X7 + X8
##
##         Df Deviance    AIC
## <none>      4226.6 2806.6
## - X2    1   4419.8 2830.4
## - X1    1   4506.6 2841.6
## - X8    5   4652.0 2851.8
## - X3    1   4803.6 2878.3
## - X5    1   5135.9 2916.8
## - X7    1   6366.9 3040.6

##
## Call:
## glm(formula = Y1 ~ X1 + X2 + X3 + X5 + X7 + X8, data = trainsdat)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -6.9650  -1.3787  -0.1214   1.2644   7.7546
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  86.661793  20.211742   4.288 2.12e-05 ***
## X1          -66.762900  10.910744  -6.119 1.76e-09 ***
## X2           -0.092251   0.018150  -5.083 5.07e-07 ***
## X3            0.062780   0.007148   8.783  < 2e-16 ***
## X5            4.026343   0.365186  11.025  < 2e-16 ***
## X7           16.309679   0.964219  16.915  < 2e-16 ***
## X81           3.962435   0.586781   6.753 3.61e-11 ***
## X82           3.948121   0.585723   6.741 3.90e-11 ***
## X83           3.696926   0.578788   6.387 3.53e-10 ***
## X84           4.255075   0.580970   7.324 8.34e-13 ***
## X85           3.752939   0.582607   6.442 2.53e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 7.480625)
```

```
##
##     Null deviance: 58768.5  on 575  degrees of freedom
## Residual deviance:  4226.6  on 565  degrees of freedom
## AIC: 2806.6
##
## Number of Fisher Scoring iterations: 2
```

```
##
## Call:
## glm(formula = Y1 ~ X1 + X2 + X3 + X5 + X7 + X8, data = trainsdat)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -6.9650  -1.3787  -0.1214   1.2644   7.7546
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  86.661793  20.211742   4.288 2.12e-05 ***
## X1          -66.762900  10.910744  -6.119 1.76e-09 ***
## X2           -0.092251   0.018150  -5.083 5.07e-07 ***
## X3            0.062780   0.007148   8.783  < 2e-16 ***
## X5            4.026343   0.365186  11.025  < 2e-16 ***
## X7           16.309679   0.964219  16.915  < 2e-16 ***
## X81           3.962435   0.586781   6.753 3.61e-11 ***
## X82           3.948121   0.585723   6.741 3.90e-11 ***
## X83           3.696926   0.578788   6.387 3.53e-10 ***
## X84           4.255075   0.580970   7.324 8.34e-13 ***
## X85           3.752939   0.582607   6.442 2.53e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 7.480625)
##
##     Null deviance: 58768.5  on 575  degrees of freedom
## Residual deviance:  4226.6  on 565  degrees of freedom
## AIC: 2806.6
##
## Number of Fisher Scoring iterations: 2
```

```
## Analysis of Deviance Table
##
## Model: gaussian, link: identity
##
## Response: Y1
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev        F    Pr(>F)
## NULL                   575      58769
```

```
## X1    1  22722.6       574      36046 3037.532 < 2.2e-16 ***
## X2    1   5938.6       573      30107  793.865 < 2.2e-16 ***
## X3    1  20892.4       572       9215 2792.867 < 2.2e-16 ***
## X5    1    748.8       571       8466  100.100 < 2.2e-16 ***
## X7    1   3814.1       570       4652  509.858 < 2.2e-16 ***
## X8    5    425.5       565       4227   11.376 1.788e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Waiting for profiling to be done...

##                    2.5 %       97.5 %
## (Intercept)  47.04750628 126.27608044
## X1          -88.14756518 -45.37823581
## X2           -0.12782387  -0.05667744
## X3            0.04877021   0.07678996
## X5            3.31059109   4.74209580
## X7           14.41984552  18.19951331
## X81           2.81236495   5.11250414
## X82           2.80012551   5.09611676
## X83           2.56252213   4.83132976
## X84           3.11639377   5.39375583
## X85           2.61105062   4.89482823
```

MODEL 1: Best Fit Equation

$Y_1 \quad = \quad B_0 \quad + \quad B_1 x_1 \quad + \quad B_2 x_2 \quad + \quad$ ....+ $B_n x_n$ +$\quad { e}\\$

Where,

Y_1 = Reponse or Dependent Variable,

x_1 .....x_n = Explantory or Independent Variables

B_0 = Intercept,

B_1 ,...., B_n = Slope of Independent variables or Model Parameter.

$\quad { e}\\$ = Residual or Error term (the difference between an actual and a predicted value of y)

The best fit model includes: X1, X2, X3, X5, X7 and X8

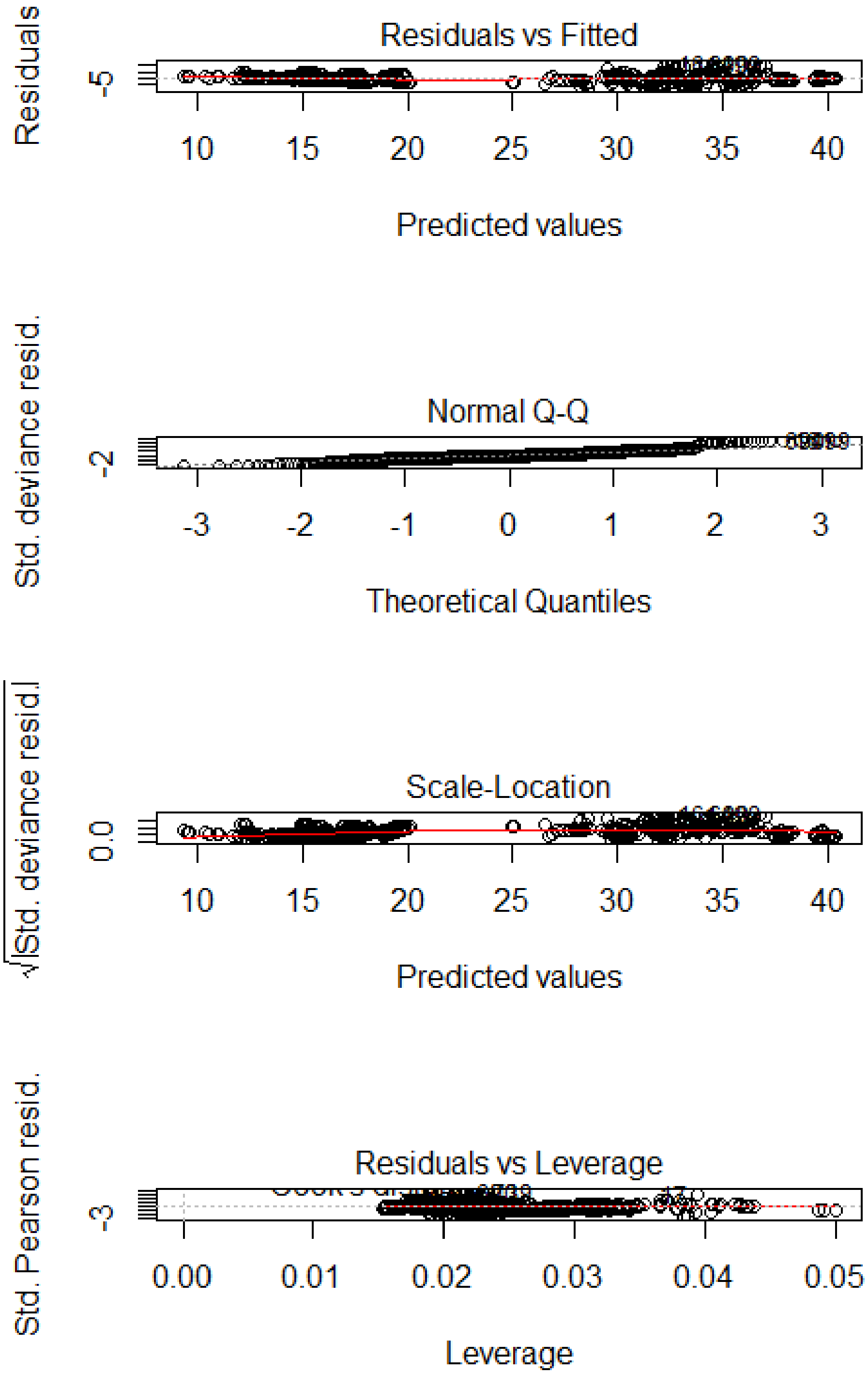
${ HeatingLoad }\quad =\quad { 86.661793 }\quad - \quad { 66.762900 }{Relative Compactness}\quad - \quad { 0.092251 }{ Surface Area }\quad + \quad { 0.062780 }{ Wall Area }\quad+\quad { 4.026343 }{ Overall Height }\quad+\quad { 16.309679 }{ Glazing Area }\quad+\quad { Glazing Area Distribution }{ x }_{ 1 }\quad ....+\quad { Glazing Area Distribution }{ x }_{ 5 }\quad +\quad { e}\\$

X1	X2	X3	X4	X5	X6	X7	X8	pred
0.98	514.5	294.0	110.25	7.0	3	0	0	20.41294

0.98	514.5	294.0	110.25	7.0	5	0	0	20.41294
0.90	563.5	318.5	122.50	7.0	2	0	0	22.77180
0.90	563.5	318.5	122.50	7.0	5	0	0	22.77180
0.86	588.0	294.0	147.00	7.0	4	0	0	21.64406
0.82	612.5	318.5	147.00	7.0	5	0	0	21.64406
0.79	637.0	343.0	147.00	7.0	5	0	0	21.64406
0.76	661.5	416.5	122.50	7.0	2	0	0	23.59255
0.76	661.5	416.5	122.50	7.0	5	0	0	23.59255
0.71	710.5	269.5	220.50	3.5	4	0	0	23.59255

MODEL 1 [Y2 COOLING LOAD]: Generalized Linear Model

```
##
## Call:
## glm(formula = Y2 ~ . - Y1, data = trainsdat)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -7.6314  -1.5590  -0.3477   1.3428  10.8587
##
## Coefficients: (1 not defined because of singularities)
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  96.214045  22.662008   4.246 2.55e-05 ***
## X1          -69.401161  12.230948  -5.674 2.23e-08 ***
## X2           -0.088979   0.020343  -4.374 1.45e-05 ***
## X3            0.045682   0.008005   5.707 1.86e-08 ***
## X4                  NA         NA      NA       NA
## X5            4.160294   0.408951  10.173  < 2e-16 ***
## X63          -0.229450   0.363808  -0.631  0.52850
## X64           0.072335   0.360529   0.201  0.84106
## X65           0.121959   0.364354   0.335  0.73796
## X7           12.985773   1.079189  12.033  < 2e-16 ***
## X81           1.963947   0.657265   2.988  0.00293 **
## X82           1.742474   0.655937   2.656  0.00812 **
## X83           1.603331   0.648452   2.473  0.01371 *
## X84           2.268047   0.650733   3.485  0.00053 ***
## X85           1.662300   0.652437   2.548  0.01110 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 9.370734)
##
##     Null deviance: 51843.1  on 575  degrees of freedom
## Residual deviance:  5266.4  on 562  degrees of freedom
## AIC: 2939.3
##
## Number of Fisher Scoring iterations: 2
```




```
## Start:  AIC=2939.3
## Y2 ~ (X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8 + Y1) - Y1
##
##
```

```
## Step:  AIC=2939.3
## Y2 ~ X1 + X2 + X3 + X5 + X6 + X7 + X8
##
##         Df Deviance    AIC
## - X6    3   5276.8 2934.4
## <none>      5266.4 2939.3
## - X8    5   5388.8 2942.5
## - X2    1   5445.6 2956.6
## - X1    1   5568.1 2969.4
## - X3    1   5571.5 2969.8
## - X5    1   6236.1 3034.7
## - X7    1   6623.1 3069.3
##
## Step:  AIC=2934.43
## Y2 ~ X1 + X2 + X3 + X5 + X7 + X8
##
##         Df Deviance    AIC
## <none>      5276.8 2934.4
## - X8    5   5401.5 2937.9
## - X2    1   5453.3 2951.4
## - X1    1   5574.6 2964.1
## - X3    1   5581.1 2964.7
## - X5    1   6250.3 3030.0
## - X7    1   6633.9 3064.3

##
## Call:
## glm(formula = Y2 ~ X1 + X2 + X3 + X5 + X7 + X8, data = trainsdat)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -7.4986  -1.5370  -0.3307   1.3461  10.8858
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  95.210349  22.583652   4.216 2.90e-05 ***
## X1          -68.840330  12.191153  -5.647 2.59e-08 ***
## X2           -0.088165   0.020280  -4.347 1.63e-05 ***
## X3            0.045596   0.007987   5.709 1.84e-08 ***
## X5            4.166112   0.408042  10.210  < 2e-16 ***
## X7           12.987467   1.077373  12.055  < 2e-16 ***
## X81           1.980576   0.655642   3.021 0.002635 **
## X82           1.761407   0.654459   2.691 0.007326 **
## X83           1.616077   0.646711   2.499 0.012740 *
## X84           2.288043   0.649149   3.525 0.000458 ***
## X85           1.679475   0.650978   2.580 0.010134 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 9.339394)
```

```
##
##     Null deviance: 51843.1  on 575  degrees of freedom
## Residual deviance:  5276.8  on 565  degrees of freedom
## AIC: 2934.4
##
## Number of Fisher Scoring iterations: 2
```

```
##
## Call:
## glm(formula = Y2 ~ X1 + X2 + X3 + X5 + X7 + X8, data = trainsdat)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -7.4986  -1.5370  -0.3307   1.3461  10.8858
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  95.210349  22.583652   4.216 2.90e-05 ***
## X1          -68.840330  12.191153  -5.647 2.59e-08 ***
## X2           -0.088165   0.020280  -4.347 1.63e-05 ***
## X3            0.045596   0.007987   5.709 1.84e-08 ***
## X5            4.166112   0.408042  10.210  < 2e-16 ***
## X7           12.987467   1.077373  12.055  < 2e-16 ***
## X81           1.980576   0.655642   3.021 0.002635 **
## X82           1.761407   0.654459   2.691 0.007326 **
## X83           1.616077   0.646711   2.499 0.012740 *
## X84           2.288043   0.649149   3.525 0.000458 ***
## X85           1.679475   0.650978   2.580 0.010134 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 9.339394)
##
##     Null deviance: 51843.1  on 575  degrees of freedom
## Residual deviance:  5276.8  on 565  degrees of freedom
## AIC: 2934.4
##
## Number of Fisher Scoring iterations: 2
```

- Analysis of Variance (ANOVA)

```
## Analysis of Deviance Table
##
## Model: gaussian, link: identity
##
## Response: Y2
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev         F  Pr(>F)
```

```
## NULL                    575       51843
## X1    1  20977.7        574       30865 2246.1573 < 2e-16 ***
## X2    1   5868.6        573       24997  628.3759 < 2e-16 ***
## X3    1  16616.5        572        8380 1779.1840 < 2e-16 ***
## X5    1    857.6        571        7523   91.8211 < 2e-16 ***
## X7    1   2121.1        570        5402  227.1143 < 2e-16 ***
## X8    5    124.8        565        5277    2.6715 0.02126 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## Waiting for profiling to be done...
```

```
##                     2.5 %       97.5 %
## (Intercept)  50.94720523 139.47349348
## X1          -92.73455025 -44.94611003
## X2           -0.12791262  -0.04841694
## X3            0.02994232   0.06125028
## X5            3.36636364   4.96585979
## X7           10.87585583  15.09907913
## X81           0.69554247   3.26560998
## X82           0.47869032   3.04412313
## X83           0.34854786   2.88360696
## X84           1.01573412   3.56035154
## X85           0.40358213   2.95536799
```

Best Fit Model Equation for Y2 Cooling Load:

$Y_2 \quad = \quad B_0 \quad + \quad B_1 x_1 \quad + \quad B_2 x_2 \quad +$ ....+ $B_n x_n$ +$\quad { e}\\$

Where,

Y_2 = Reponse or Dependent Variable,

x_1 .....x_n = Explantory or Independent Variables

B_0 = Intercept,

B_1 ,...., B_n = Slope of Independent variables or Model Parameter.

$\quad { e}\\$ = Residual or Error term (the difference between an actual and a predicted value of y)

The best model includes: X1, X2, X3, X5, X7 and X8

${ CoolingLoad }\quad =\quad { 95.210349 }\quad - \quad { 68.840330}{Relative Compactness}\quad - \quad {0.088165}{ Surface Area }\quad+\quad { 0.045596 }{ Wall Area }\quad+\quad { 4.166112 }{ Overall Height }\quad+\quad { 12.987467}{Glazing Area}\quad+\quad { Glazing Area Distribution }{ x }_{ 1 }\quad ....+\quad { Glazing Area Distribution }{ x }_{ 5 }\quad+\quad { e}\\$

Now using the above equation we can very well predict the Y2(Cooling Load) for any building and make it more smarter, so we will run the testdat against the best fit model equation and it will give the results as shown below:

X1	X2	X3	X4	X5	X6	X7	X8	pred
0.98	514.5	294.0	110.25	7.0	3	0	0	24.95414
0.98	514.5	294.0	110.25	7.0	5	0	0	24.95414
0.90	563.5	318.5	122.50	7.0	2	0	0	27.25840
0.90	563.5	318.5	122.50	7.0	5	0	0	27.25840
0.86	588.0	294.0	147.00	7.0	4	0	0	26.73487
0.82	612.5	318.5	147.00	7.0	5	0	0	26.73487
0.79	637.0	343.0	147.00	7.0	5	0	0	26.73487
0.76	661.5	416.5	122.50	7.0	2	0	0	28.44555
0.76	661.5	416.5	122.50	7.0	5	0	0	28.44555
0.71	710.5	269.5	220.50	3.5	4	0	0	28.44555

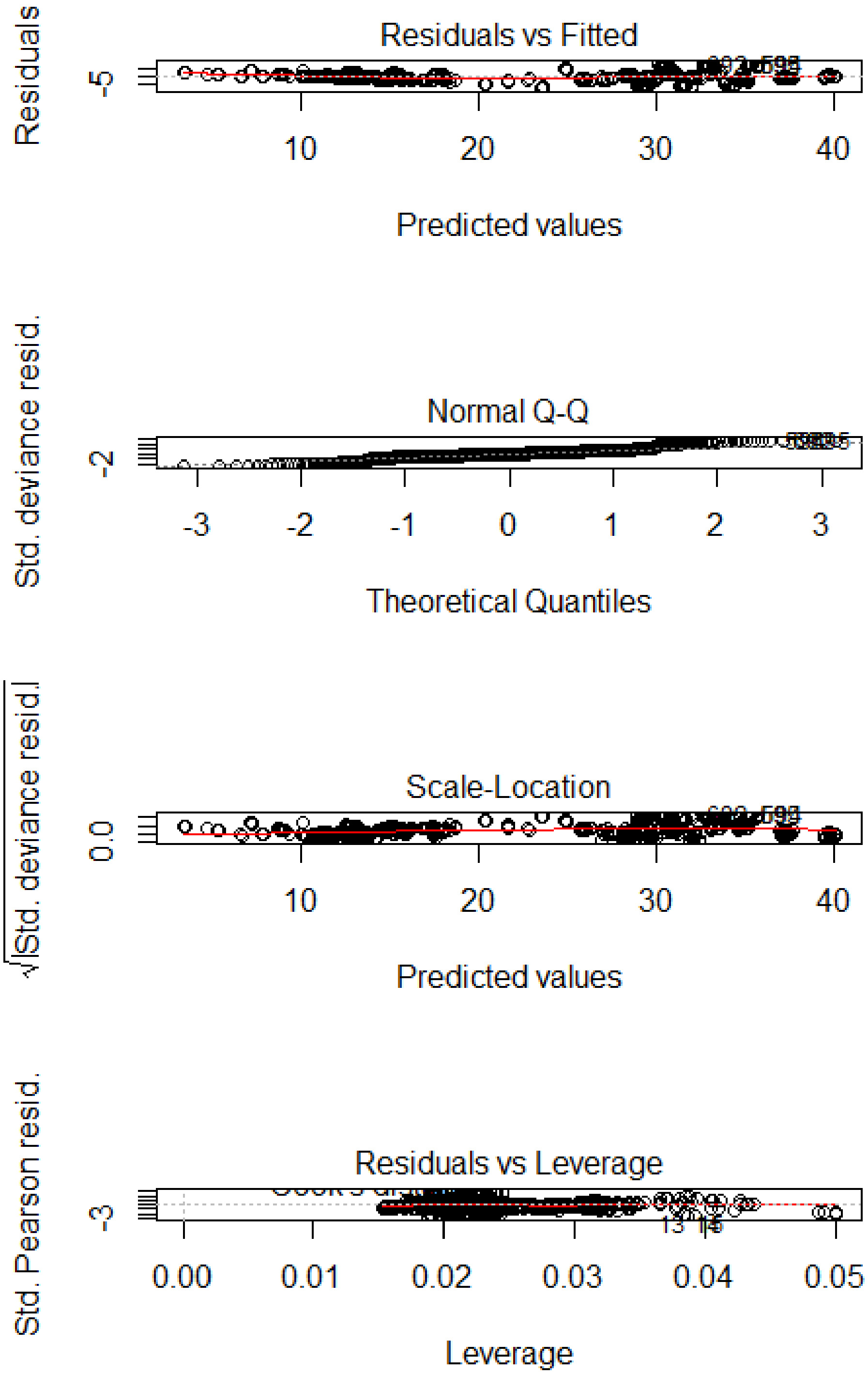
The above is the predicted value (pred columns) for Y1

```
##    pred_heat$pred pred_cool$pred
## 1        20.41294       24.95414
## 2        20.41294       24.95414
## 3        22.77180       27.25840
## 4        22.77180       27.25840
## 5        21.64406       26.73487
## 6        21.64406       26.73487
## 7        21.64406       26.73487
## 8        23.59255       28.44555
## 9        23.59255       28.44555
## 10       23.59255       28.44555
```

MODEL 2 : Y1 (Heating Load) WIDE FORMAT (X6, X8)

```
## split_ins
## FALSE  TRUE
##   192   576

## [1] 0.75

##
## Call:
## glm(formula = Y1 ~ . - Y2, data = trainsdat_2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -7.0109  -1.3914  -0.1186   1.2443   7.7082
##
## Coefficients: (3 not defined because of singularities)
```

```
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  90.41015   20.29475   4.455 1.01e-05 ***
## X1          -66.74819   10.95588  -6.092 2.06e-09 ***
## X2           -0.09226    0.01822  -5.063 5.60e-07 ***
## X3            0.06282    0.00717   8.761  < 2e-16 ***
## X4                 NA         NA      NA       NA
## X5            4.02461    0.36632  10.987  < 2e-16 ***
## X7           16.30929    0.96668  16.871  < 2e-16 ***
## `6X_2`        0.01457    0.32637   0.045    0.964
## `6X_3`        0.04342    0.32456   0.134    0.894
## `6x_4`       -0.06859    0.32164  -0.213    0.831
## `6X_5`             NA         NA      NA       NA
## `8X_0`       -3.75414    0.58442  -6.424 2.84e-10 ***
## `8X_1`        0.20768    0.37903   0.548    0.584
## `8X_2`        0.19498    0.37867   0.515    0.607
## `8X_3`       -0.05963    0.36874  -0.162    0.872
## `8X_4`        0.50172    0.36381   1.379    0.168
## `8X_5`             NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 7.518788)
##
##     Null deviance: 58768.5  on 575  degrees of freedom
## Residual deviance:  4225.6  on 562  degrees of freedom
## AIC: 2812.5
##
## Number of Fisher Scoring iterations: 2
```




```
## Start:  AIC=2812.47
## Y1 ~ (X1 + X2 + X3 + X4 + X5 + X7 + Y2 + `6X_2` + `6X_3` + `6x_4` +
##     `6X_5` + `8X_0` + `8X_1` + `8X_2` + `8X_3` + `8X_4` + `8X_5`) -
##     Y2
```

```
##
##
## Step:  AIC=2812.47
## Y1 ~ X1 + X2 + X3 + X4 + X5 + X7 + `6X_2` + `6X_3` + `6x_4` +
##     `6X_5` + `8X_0` + `8X_1` + `8X_2` + `8X_3` + `8X_4`
##
##
## Step:  AIC=2812.47
## Y1 ~ X1 + X2 + X3 + X4 + X5 + X7 + `6X_2` + `6X_3` + `6x_4` +
##     `8X_0` + `8X_1` + `8X_2` + `8X_3` + `8X_4`
##
##
## Step:  AIC=2812.47
## Y1 ~ X1 + X2 + X3 + X5 + X7 + `6X_2` + `6X_3` + `6x_4` + `8X_0` +
##     `8X_1` + `8X_2` + `8X_3` + `8X_4`
##
##          Df Deviance    AIC
## - `6X_2`  1   4225.6 2810.5
## - `6X_3`  1   4225.7 2810.5
## - `8X_3`  1   4225.8 2810.5
## - `6x_4`  1   4225.9 2810.5
## - `8X_2`  1   4227.6 2810.7
## - `8X_1`  1   4227.8 2810.8
## - `8X_4`  1   4239.9 2812.4
## <none>        4225.6 2812.5
## - X2      1   4418.3 2836.2
## - X1      1   4504.6 2847.3
## - `8X_0`  1   4535.8 2851.3
## - X3      1   4802.7 2884.2
## - X5      1   5133.1 2922.5
## - X7      1   6365.7 3046.5
##
## Step:  AIC=2810.47
## Y1 ~ X1 + X2 + X3 + X5 + X7 + `6X_3` + `6x_4` + `8X_0` + `8X_1` +
##     `8X_2` + `8X_3` + `8X_4`
##
##          Df Deviance    AIC
## - `6X_3`  1   4225.7 2808.5
## - `8X_3`  1   4225.8 2808.5
## - `6x_4`  1   4226.1 2808.6
## - `8X_2`  1   4227.6 2808.7
## - `8X_1`  1   4227.8 2808.8
## - `8X_4`  1   4239.9 2810.4
## <none>        4225.6 2810.5
## - X2      1   4418.4 2834.2
## - X1      1   4504.8 2845.3
## - `8X_0`  1   4535.9 2849.3
## - X3      1   4802.7 2882.2
## - X5      1   5133.1 2920.5
## - X7      1   6365.7 3044.5
```

```
##
## Step:  AIC=2808.49
## Y1 ~ X1 + X2 + X3 + X5 + X7 + `6x_4` + `8X_0` + `8X_1` + `8X_2` +
##     `8X_3` + `8X_4`
##
##          Df Deviance    AIC
## - `8X_3`  1   4225.9 2806.5
## - `6x_4`  1   4226.6 2806.6
## - `8X_2`  1   4227.7 2806.8
## - `8X_1`  1   4228.0 2806.8
## - `8X_4`  1   4240.0 2808.4
## <none>        4225.7 2808.5
## - X2      1   4419.4 2832.3
## - X1      1   4506.2 2843.5
## - `8X_0`  1   4535.9 2847.3
## - X3      1   4803.5 2880.3
## - X5      1   5133.2 2918.6
## - X7      1   6365.8 3042.5
##
## Step:  AIC=2806.52
## Y1 ~ X1 + X2 + X3 + X5 + X7 + `6x_4` + `8X_0` + `8X_1` + `8X_2` +
##     `8X_4`
##
##          Df Deviance    AIC
## - `6x_4`  1   4226.7 2804.6
## - `8X_2`  1   4229.4 2805.0
## - `8X_1`  1   4229.8 2805.0
## <none>        4225.9 2806.5
## - `8X_4`  1   4247.6 2807.5
## - X2      1   4419.4 2830.3
## - X1      1   4506.2 2841.5
## - `8X_0`  1   4567.2 2849.3
## - X3      1   4803.5 2878.3
## - X5      1   5133.6 2916.6
## - X7      1   6367.8 3040.7
##
## Step:  AIC=2804.63
## Y1 ~ X1 + X2 + X3 + X5 + X7 + `8X_0` + `8X_1` + `8X_2` + `8X_4`
##
##          Df Deviance    AIC
## - `8X_2`  1   4230.2 2803.1
## - `8X_1`  1   4230.6 2803.2
## <none>        4226.7 2804.6
## - `8X_4`  1   4248.4 2805.6
## - X2      1   4419.8 2828.4
## - X1      1   4506.6 2839.6
## - `8X_0`  1   4568.7 2847.4
## - X3      1   4803.6 2876.3
## - X5      1   5136.2 2914.9
## - X7      1   6368.8 3038.8
```

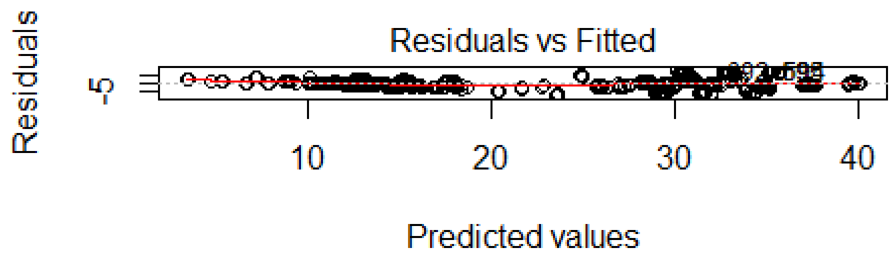
```
##
## Step:  AIC=2803.1
## Y1 ~ X1 + X2 + X3 + X5 + X7 + `8X_0` + `8X_1` + `8X_4`
##
##          Df Deviance    AIC
## - `8X_1`  1   4232.3 2801.4
## <none>        4230.2 2803.1
## - `8X_4`  1   4248.5 2803.6
## - X2      1   4424.8 2827.0
## - X1      1   4511.8 2838.2
## - `8X_0`  1   4598.4 2849.2
## - X3      1   4807.6 2874.8
## - X5      1   5138.1 2913.1
## - X7      1   6370.7 3036.9
##
## Step:  AIC=2801.39
## Y1 ~ X1 + X2 + X3 + X5 + X7 + `8X_0` + `8X_4`
##
##          Df Deviance    AIC
## <none>        4232.3 2801.4
## - `8X_4`  1   4248.7 2801.6
## - X2      1   4427.0 2825.3
## - X1      1   4513.9 2836.5
## - `8X_0`  1   4615.4 2849.3
## - X3      1   4810.7 2873.2
## - X5      1   5139.2 2911.2
## - X7      1   6373.0 3035.2

##
## Call:
## glm(formula = Y1 ~ X1 + X2 + X3 + X5 + X7 + `8X_0` + `8X_4`,
##     data = trainsdat_2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -6.9636  -1.3797  -0.1595   1.2459   7.8722
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  90.789610  20.140868   4.508 7.96e-06 ***
## X1          -66.885797  10.881499  -6.147 1.49e-09 ***
## X2           -0.092527   0.018103  -5.111 4.38e-07 ***
## X3            0.062841   0.007133   8.810  < 2e-16 ***
## X5            4.020031   0.364390  11.032  < 2e-16 ***
## X7           16.306784   0.962072  16.950  < 2e-16 ***
## `8X_0`       -3.834442   0.534761  -7.170 2.34e-12 ***
## `8X_4`        0.421136   0.284546   1.480    0.139
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

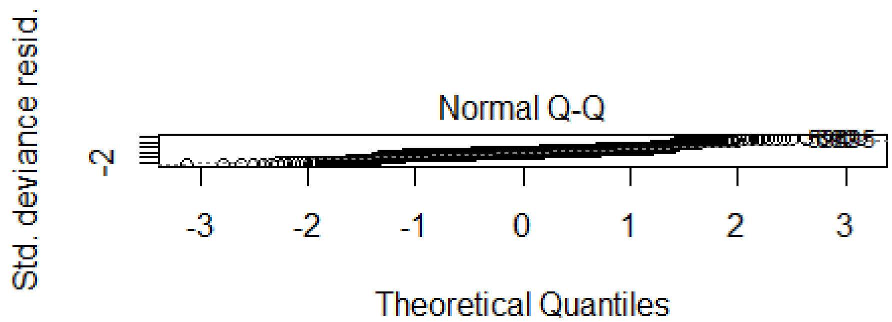
```
## (Dispersion parameter for gaussian family taken to be 7.4513)
##
##     Null deviance: 58768.5  on 575  degrees of freedom
## Residual deviance:  4232.3  on 568  degrees of freedom
## AIC: 2801.4
##
## Number of Fisher Scoring iterations: 2
```

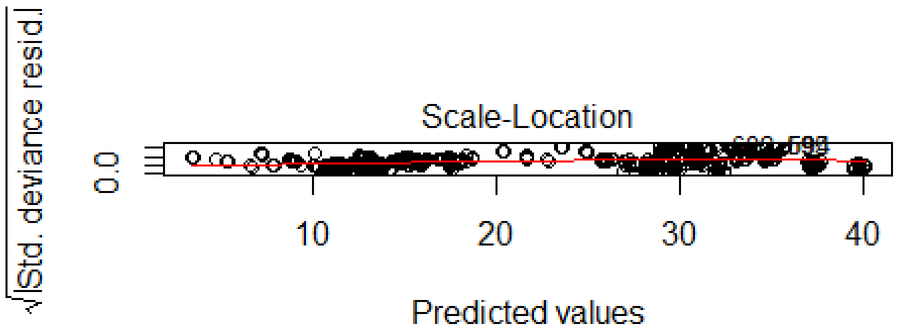
MODEL 2 : Y2 (Cooling Load) WIDE FORMAT (X6, X8)

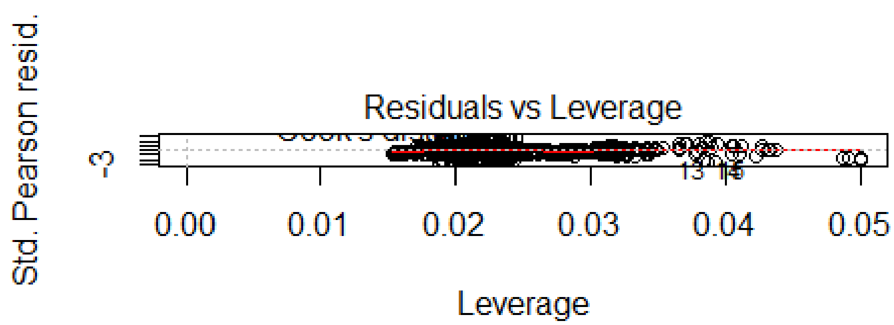
```
coolingload_2 <- glm(Y1 ~.-Y2, data=trainsdat_2);

summary(coolingload_2)

##
## Call:
## glm(formula = Y1 ~ . - Y2, data = trainsdat_2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -7.0109  -1.3914  -0.1186   1.2443   7.7082
##
## Coefficients: (3 not defined because of singularities)
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  90.41015   20.29475   4.455 1.01e-05 ***
## X1          -66.74819   10.95588  -6.092 2.06e-09 ***
## X2           -0.09226    0.01822  -5.063 5.60e-07 ***
## X3            0.06282    0.00717   8.761  < 2e-16 ***
## X4                 NA         NA      NA       NA
## X5            4.02461    0.36632  10.987  < 2e-16 ***
## X7           16.30929    0.96668  16.871  < 2e-16 ***
## `6X_2`        0.01457    0.32637   0.045    0.964
## `6X_3`        0.04342    0.32456   0.134    0.894
## `6x_4`       -0.06859    0.32164  -0.213    0.831
## `6X_5`             NA         NA      NA       NA
## `8X_0`       -3.75414    0.58442  -6.424 2.84e-10 ***
## `8X_1`        0.20768    0.37903   0.548    0.584
## `8X_2`        0.19498    0.37867   0.515    0.607
## `8X_3`       -0.05963    0.36874  -0.162    0.872
## `8X_4`        0.50172    0.36381   1.379    0.168
## `8X_5`             NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 7.518788)
##
##     Null deviance: 58768.5  on 575  degrees of freedom
## Residual deviance:  4225.6  on 562  degrees of freedom
## AIC: 2812.5
##
## Number of Fisher Scoring iterations: 2
```

```
par(mfrow=c(2,1))
plot(coolingload_2)
```









```
coolingload_2 <- step(glm(Y2 ~.-Y1, data=trainsdat_2), direction = "backward"
);

## Start:  AIC=2939.3
## Y2 ~ (X1 + X2 + X3 + X4 + X5 + X7 + Y1 + `6X_2` + `6X_3` + `6x_4` +
##     `6X_5` + `8X_0` + `8X_1` + `8X_2` + `8X_3` + `8X_4` + `8X_5`) -
##     Y1
##
##
## Step:  AIC=2939.3
## Y2 ~ X1 + X2 + X3 + X4 + X5 + X7 + `6X_2` + `6X_3` + `6x_4` +
##     `6X_5` + `8X_0` + `8X_1` + `8X_2` + `8X_3` + `8X_4`
##
##
## Step:  AIC=2939.3
## Y2 ~ X1 + X2 + X3 + X4 + X5 + X7 + `6X_2` + `6X_3` + `6x_4` +
##     `8X_0` + `8X_1` + `8X_2` + `8X_3` + `8X_4`
##
##
## Step:  AIC=2939.3
## Y2 ~ X1 + X2 + X3 + X5 + X7 + `6X_2` + `6X_3` + `6x_4` + `8X_0` +
##     `8X_1` + `8X_2` + `8X_3` + `8X_4`
##
##          Df Deviance    AIC
## - `6x_4`  1   5266.5 2937.3
## - `8X_3`  1   5266.5 2937.3
## - `8X_2`  1   5266.7 2937.3
## - `6X_2`  1   5267.4 2937.4
## - `8X_1`  1   5271.1 2937.8
## - `6X_3`  1   5275.2 2938.3
## <none>        5266.4 2939.3
## - `8X_4`  1   5287.2 2939.6
## - `8X_0`  1   5327.2 2943.9
## - X2      1   5445.6 2956.6
## - X1      1   5568.1 2969.4
## - X3      1   5571.5 2969.8
## - X5      1   6236.1 3034.7
## - X7      1   6623.1 3069.3
##
## Step:  AIC=2937.32
## Y2 ~ X1 + X2 + X3 + X5 + X7 + `6X_2` + `6X_3` + `8X_0` + `8X_1` +
##     `8X_2` + `8X_3` + `8X_4`
##
##          Df Deviance    AIC
## - `8X_3`  1   5266.7 2935.3
## - `8X_2`  1   5266.9 2935.3
## - `6X_2`  1   5267.4 2935.4
## - `8X_1`  1   5271.3 2935.8
## - `6X_3`  1   5276.7 2936.4
## <none>        5266.5 2937.3
```

```
## - `8X_4`  1   5287.4 2937.6
## - `8X_0`  1   5327.5 2941.9
## - X2      1   5445.6 2954.6
## - X1      1   5568.1 2967.4
## - X3      1   5571.5 2967.8
## - X5      1   6237.9 3032.8
## - X7      1   6623.5 3067.4
##
## Step:  AIC=2935.34
## Y2 ~ X1 + X2 + X3 + X5 + X7 + `6X_2` + `6X_3` + `8X_0` + `8X_1` +
##     `8X_2` + `8X_4`
##
##          Df Deviance    AIC
## - `8X_2`  1   5267.5 2933.4
## - `6X_2`  1   5267.6 2933.4
## - `8X_1`  1   5274.3 2934.2
## - `6X_3`  1   5277.0 2934.5
## <none>        5266.7 2935.3
## - `8X_4`  1   5297.7 2936.7
## - `8X_0`  1   5332.6 2940.5
## - X2      1   5445.7 2952.6
## - X1      1   5568.1 2965.4
## - X3      1   5571.6 2965.8
## - X5      1   6238.2 3030.8
## - X7      1   6624.9 3065.5
##
## Step:  AIC=2933.43
## Y2 ~ X1 + X2 + X3 + X5 + X7 + `6X_2` + `6X_3` + `8X_0` + `8X_1` +
##     `8X_4`
##
##          Df Deviance    AIC
## - `6X_2`  1   5268.4 2931.5
## - `8X_1`  1   5274.3 2932.2
## - `6X_3`  1   5277.9 2932.6
## <none>        5267.5 2933.4
## - `8X_4`  1   5298.7 2934.8
## - `8X_0`  1   5338.8 2939.2
## - X2      1   5447.3 2950.8
## - X1      1   5569.9 2963.6
## - X3      1   5572.7 2963.9
## - X5      1   6238.4 3028.9
## - X7      1   6625.2 3063.5
##
## Step:  AIC=2931.52
## Y2 ~ X1 + X2 + X3 + X5 + X7 + `6X_3` + `8X_0` + `8X_1` + `8X_4`
##
##          Df Deviance    AIC
## - `8X_1`  1   5275.2 2930.3
## - `6X_3`  1   5277.9 2930.6
## <none>        5268.4 2931.5
```

```
## - `8X_4`  1   5299.6 2932.9
## - `8X_0`  1   5339.8 2937.3
## - X2      1   5448.3 2948.9
## - X1      1   5571.0 2961.7
## - X3      1   5574.0 2962.0
## - X5      1   6238.8 3026.9
## - X7      1   6626.2 3061.6
##
## Step:  AIC=2930.26
## Y2 ~ X1 + X2 + X3 + X5 + X7 + `6X_3` + `8X_0` + `8X_4`
##
##          Df Deviance    AIC
## - `6X_3`  1   5284.6 2929.3
## <none>        5275.2 2930.3
## - `8X_4`  1   5301.1 2931.1
## - `8X_0`  1   5354.2 2936.8
## - X2      1   5455.2 2947.6
## - X1      1   5577.5 2960.4
## - X3      1   5581.9 2960.8
## - X5      1   6243.7 3025.3
## - X7      1   6633.1 3060.2
##
## Step:  AIC=2929.29
## Y2 ~ X1 + X2 + X3 + X5 + X7 + `8X_0` + `8X_4`
##
##          Df Deviance    AIC
## <none>        5284.6 2929.3
## - `8X_4`  1   5311.0 2930.2
## - `8X_0`  1   5364.7 2936.0
## - X2      1   5461.8 2946.3
## - X1      1   5583.0 2958.9
## - X3      1   5590.1 2959.7
## - X5      1   6255.7 3024.5
## - X7      1   6642.7 3059.0
```

```
summary(coolingload_2)
```

```
##
## Call:
## glm(formula = Y2 ~ X1 + X2 + X3 + X5 + X7 + `8X_0` + `8X_4`,
##     data = trainsdat_2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -7.4941  -1.5451  -0.2943   1.3274  11.0573
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  97.060325  22.505848   4.313 1.90e-05 ***
## X1          -68.854379  12.159226  -5.663 2.37e-08 ***
```

```
## X2           -0.088282   0.020229  -4.364 1.52e-05 ***
## X3            0.045674   0.007971   5.730 1.63e-08 ***
## X5            4.159902   0.407178  10.216  < 2e-16 ***
## X7           12.988480   1.075041  12.082  < 2e-16 ***
## `8X_0`       -1.753091   0.597553  -2.934  0.00348 **
## `8X_4`        0.534827   0.317958   1.682  0.09310 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 9.30393)
##
##     Null deviance: 51843.1  on 575  degrees of freedom
## Residual deviance:  5284.6  on 568  degrees of freedom
## AIC: 2929.3
##
## Number of Fisher Scoring iterations: 2

MOD2Y1<-c('Model_WY1',58768.5,4225.6,2812.5,'X1,X2,X3,X5,X7,8X_0')
MOD2Y2<-c('Model_WY2',51843.1,5284.6,2929.3,'X1,X2,X3,X5,X7,8X_0')
```

SELECT MODEL

Side-by-side comparison of Null and Residual Deviance, AIC and significant variables for best fit model selection

```
MOD_STAT<- cbind(MOD1Y1,MOD2Y1,MOD1Y2,MOD2Y2)
names(MOD_STAT)=c("Model_Name","Null_Deviance","Residual_Deviance","AIC","Sig
nificant Vars")
kable(MOD_STAT)
```

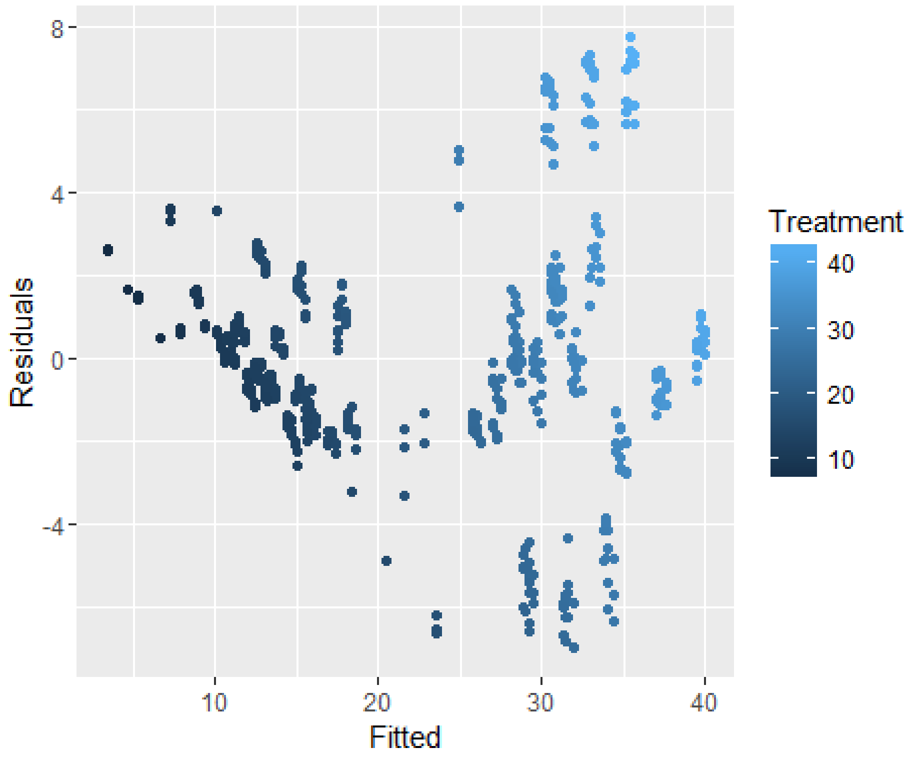
MOD1Y1	MOD2Y1	MOD1Y2	MOD2Y2
Model_1LY1	Model_WY1	Model_LY2	Model_WY2
51843.1	58768.5	51843.1	51843.1
5266.4	4225.6	5276.8	5284.6
2939.3	2812.5	2934.4	2929.3
X1,X2,X3,X5,X7,X8	X1,X2,X3,X5,X7,8X_0	X1,X2,X3,X5,X7,X8	X1,X2,X3,X5,X7,8X_0

Side-by-side predictions for long and wide format

Pred_HL_W	Pred_HL_L	Pred_CL_W	Pred_CL_L
20.417600	20.412937	24.95662	24.95414
20.417600	20.412937	24.95662	24.95414

```
## [1]  0.004662986  0.004662986  0.002464134  0.002464134 -0.000867172
## [6] -0.000867172

## [1]  0.002480081  0.002480081 -0.000208347 -0.000208347 -0.004414388
## [6] -0.004414388
```

A side-by-side plot between Fitted values and Residual for HeatingLoad.



A side-by-side plot between Fitted values and Residual for CoolingLoad.