Energy Efficiency Regression Analysis

Bradley Assaly-Nesrallah

12/13/2020

Introduction:

The goal of this regression analysis project is to perform a multiple linear regression analysis on the Energy Efficiency Data Set for the responce variable y2, cooling load. We want to use regression analysis to predict the cooling load of a building when given the building parameters. In other words we seek to obtain a multiple linear regression model that can predict a dependent variable in terms of other variables in the model. In our case we have a data set with 8 attributes X1-X8 and two response variables y1, heating load and y2, cooling load. In our analysis we will drop the y1 variable and simply focus on y2 as our response variable. Our goal is to find a model that best predicts y2 using some of all of the 8 predictors, x1-x8, in other words we want to use the building shape features to predict the cooling load of the building. An in depth treatment of these varibles will take place in the data set description section of this report. A literature review of other papers concerning using regression analysis on the Energy Efficiency Data Set showed that a variety of machine learning methods such as decision trees and random forests could efficiently predict the response variables, however, we will attempt to use multiple linear regression in this paper. We will begin by cleaning the data to get it ready for manipulation and analysis. Then, we will describe the methods we used to find the best set of predictors to use for multiple regression to analyze the data set, in particular we will discuss how the steps were carried out based on analysis discussed in class. Then we will provide a summary of the output and the results, along with a discussion of the finding and its inferences. Finally we will consider the limitations and further questions raised by the study.

Description of the Data Set:

The Data Set we are analyzing is the Energy efficiency Data Set which is a study looking into the heating load and cooling load requirements of buildings as a function of building paramenters. The Data Set contains 768 observations with eight attributes denoted X1-X8 and two responses denoted Y1 and Y2. The attributes in the data set are X1-Relative Compactness ,X2-Surface Area, X3-Wall Area, X4-Roof Area, X5-Overall Height, X6-Orientation, X7-Glazing Area, X8-Glazing Area Distribution and the response variables are y1-Heating Load, y2-Cooling Load, but we are only considering the response variable y2. The attributes are real numbers for X1-X5,X7 and integers for X6 and X8. We are looking to predict the response variable y2 using attributes X1-X8 in our multiple regression analysis.

Analysis of the Data:

Here we will describe the specific steps we took using multiple regression to analyze the data set. In particular we will discuss the how we carried out the steps to analyze the data set. In our analysis we will show how we defined the model, estimated the results of the model, determined the interactions within the model, verified the model assumptions and performed variable selection similar to the methods that were discussed in our class.

```
## First we imported the Energy Efficiency Data Set using R
library("readxl")
data_init <- read_excel('ENB2012_data.xlsx')
## We removed the variable y1 from the data since we are not using it in our
analysis
data_1 <- data_init[,-9]
## We use the pairs function to observe the relationship between the
variables
pairs(data_1)


## Based on the pairs function results we make the attributes X6-X8 because
their
## relationship with other variables appears to be discrete, thus we
transform them
## into categorical varialbes as follows:
data_1$X6 <- as.factor(data_1$X6)
data_1$X7 <- as.factor(data_1$X7)
data_1$X8 <- as.factor(data_1$X8)

## We now define two models, a null model using all the parameters and a null
model
## The full model uses all the variables X1-X8 to predict cooling load (y2)
lm_y2_full <- lm(Y2~., data=data_1)
lm_y2_null <- lm(Y2~1, data=data_1)

## We will now use AIC and BIC with various step functions to select the
optimal
## combination of variables for our model
## We first use AIC with backwards step selection to select variables for our
model
fit_back_aic = step(lm_y2_full, direction = "backward",trace=0)
## We now observe the results of this variable selection procedure
summary(fit_back_aic)

##
## Call:
## lm(formula = Y2 ~ X1 + X2 + X3 + X5 + X7, data = data_1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.7507 -1.7566 -0.2843  1.3521 11.2930
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  96.353299  20.616056   4.674 3.50e-06 ***
## X1          -70.787707  11.144919  -6.352 3.67e-10 ***
## X2           -0.088245   0.018495  -4.771 2.20e-06 ***
## X3            0.044682   0.007201   6.205 8.97e-10 ***
## X5            4.283843   0.366091  11.702  < 2e-16 ***
## X70.1         3.229292   0.502530   6.426 2.31e-10 ***
## X70.25        5.186375   0.502530  10.321  < 2e-16 ***
## X70.4         7.205167   0.502530  14.338  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.178 on 760 degrees of freedom
## Multiple R-squared:  0.8894, Adjusted R-squared:  0.8884
## F-statistic: 873.1 on 7 and 760 DF,  p-value: < 2.2e-16
```

```
## The AIC with backwards step selection chooses the attributes
X1,X2,X3,X5,X7
## as the best predictors for Y2, it has an adjusted R^2 of 0.8884

## We now use BIC with backwards selection to select variables for our model
n = nrow(data_1)
fit_back_bic = step(lm_y2_full, direction = "backward", k=log(n),trace=0)
summary(fit_back_bic)
```

```
##
## Call:
## lm(formula = Y2 ~ X1 + X2 + X3 + X5 + X7, data = data_1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.7507 -1.7566 -0.2843  1.3521 11.2930
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  96.353299  20.616056   4.674 3.50e-06 ***
## X1          -70.787707  11.144919  -6.352 3.67e-10 ***
## X2           -0.088245   0.018495  -4.771 2.20e-06 ***
## X3            0.044682   0.007201   6.205 8.97e-10 ***
## X5            4.283843   0.366091  11.702  < 2e-16 ***
## X70.1         3.229292   0.502530   6.426 2.31e-10 ***
## X70.25        5.186375   0.502530  10.321  < 2e-16 ***
## X70.4         7.205167   0.502530  14.338  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.178 on 760 degrees of freedom
## Multiple R-squared:  0.8894, Adjusted R-squared:  0.8884
## F-statistic: 873.1 on 7 and 760 DF,  p-value: < 2.2e-16
```

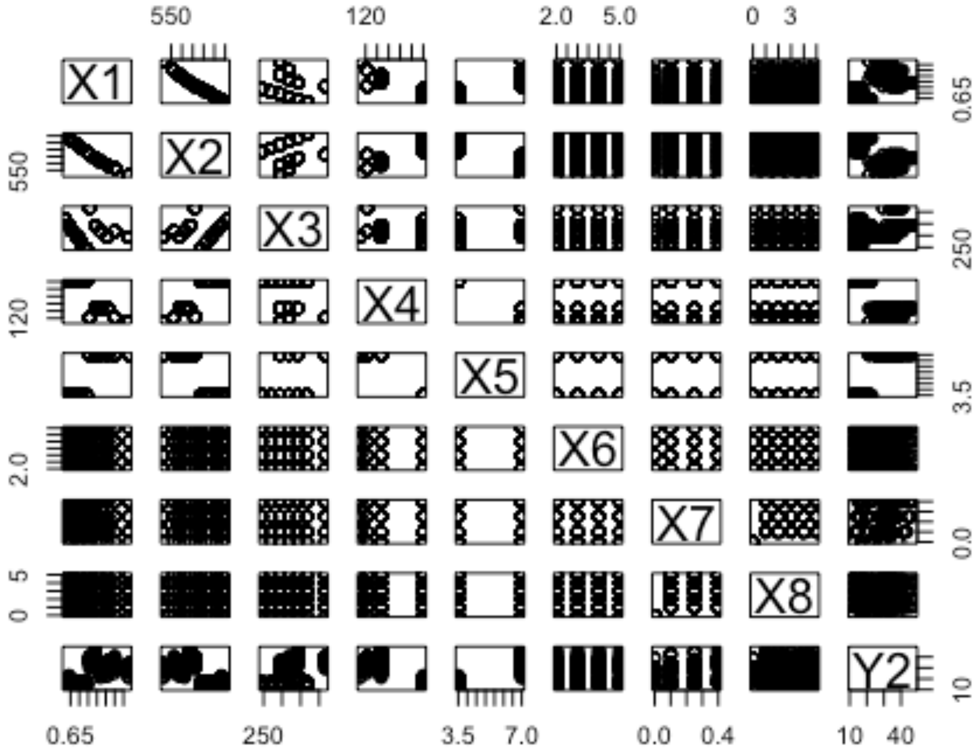
```r
## The BIC with backwards step selection chooses the attributes
X1,X2,X3,X5,X7
## as the best predictors for Y2, it has an adjusted R^2 of 0.8884

## We now use AIC with forward selection to select variables for our model
fit_forw_aic = step(lm_y2_null, scope = Y2 ~ X1+X2+X3+X4+X5+X6+X7+X8,
direction = "forward",trace=0)
summary(fit_forw_aic)
```

```
##
## Call:
## lm(formula = Y2 ~ X5 + X7 + X1 + X4 + X2, data = data_1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.7507 -1.7566 -0.2843  1.3521 11.2930
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  96.35330   20.61606   4.674 3.50e-06 ***
## X5            4.28384    0.36609  11.702  < 2e-16 ***
## X70.1         3.22929    0.50253   6.426 2.31e-10 ***
## X70.25        5.18637    0.50253  10.321  < 2e-16 ***
## X70.4         7.20517    0.50253  14.338  < 2e-16 ***
## X1          -70.78771   11.14492  -6.352 3.67e-10 ***
## X4           -0.08936    0.01440  -6.205 8.97e-10 ***
## X2           -0.04356    0.01384  -3.149   0.0017 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.178 on 760 degrees of freedom
## Multiple R-squared:  0.8894, Adjusted R-squared:  0.8884
## F-statistic: 873.1 on 7 and 760 DF,  p-value: < 2.2e-16
```

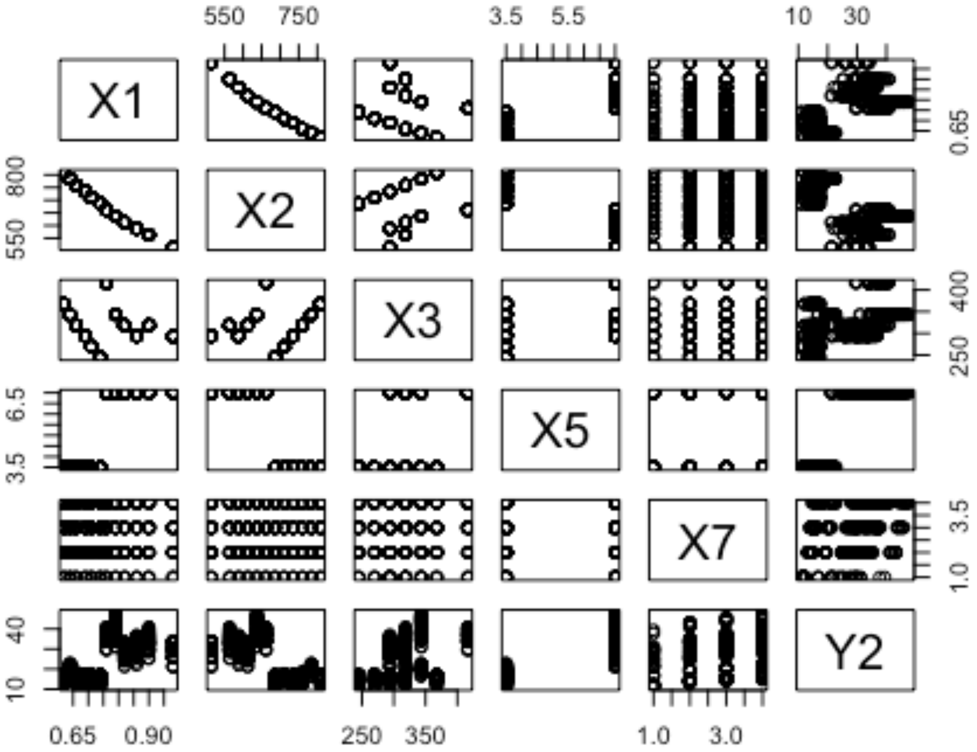
```r
## The AIC with forwards step selection chooses the attributes X1,X2,X3,X5,X7
## as the best predictors for Y2, it has an adjusted R^2 of 0.8884

## We now use AIC with both directions selection to select variables for our
model
fit_final = step(lm_y2_full, scope = Y2 ~ X1+X2+X3+X4+X5+X6+X7+X8, direction
= "both",trace=0)
summary(fit_final)
```

```
##
## Call:
## lm(formula = Y2 ~ X1 + X2 + X3 + X5 + X7, data = data_1)
##
## Residuals:
```

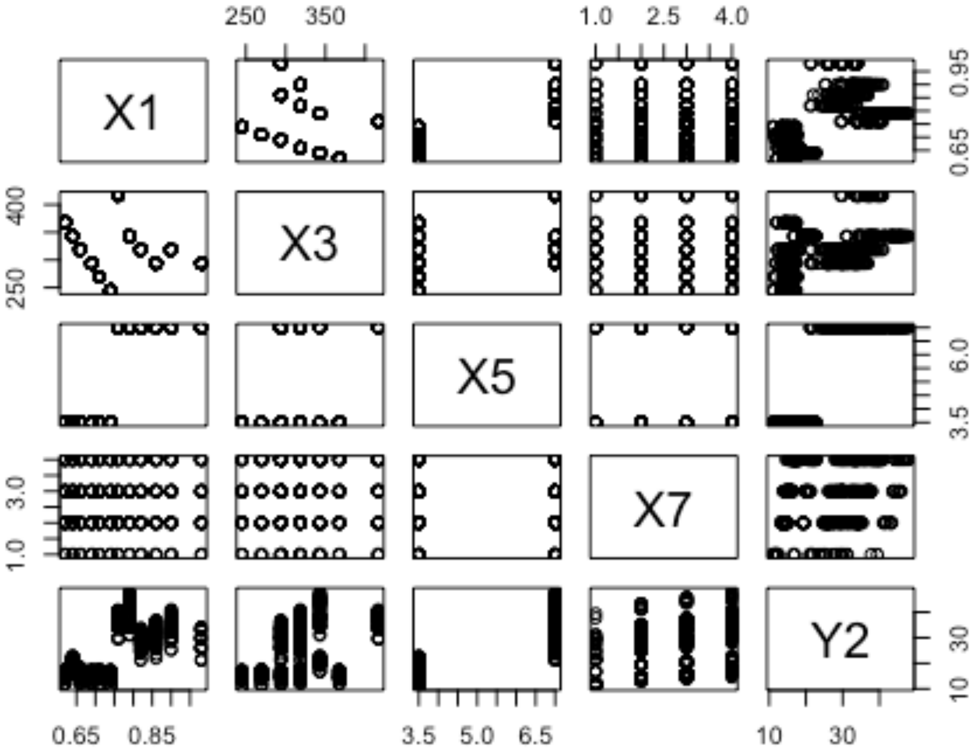
```
##     Min      1Q  Median      3Q     Max 
## -7.7507 -1.7566 -0.2843  1.3521 11.2930 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  96.353299  20.616056   4.674 3.50e-06 ***
## X1          -70.787707  11.144919  -6.352 3.67e-10 ***
## X2           -0.088245   0.018495  -4.771 2.20e-06 ***
## X3            0.044682   0.007201   6.205 8.97e-10 ***
## X5            4.283843   0.366091  11.702  < 2e-16 ***
## X70.1         3.229292   0.502530   6.426 2.31e-10 ***
## X70.25        5.186375   0.502530  10.321  < 2e-16 ***
## X70.4         7.205167   0.502530  14.338  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 3.178 on 760 degrees of freedom
## Multiple R-squared:  0.8894, Adjusted R-squared:  0.8884 
## F-statistic: 873.1 on 7 and 760 DF,  p-value: < 2.2e-16

## The AIC with both directions step selection chooses the attributes 
X1,X2,X3,X5,X7
## as the best predictors for Y2, it has an adjusted R^2 of 0.8884

## Now it appears that all of the stepwise variable selection methods above 
using
## AIC and BIC agree on using the attributes X1,X2,X3,X5,X7 to predict the 
response variable Y2

## Thus we define our final model as Y2 ~ X1+X2+X3+X5+X7, we will next check 
model
## interactions as follows

## Now we will check if there exists interaction or collinearity between the
## variables in our final model as follows

## We first check the relationships using the pairs function
## install.packages("faraway")
library(faraway)
```



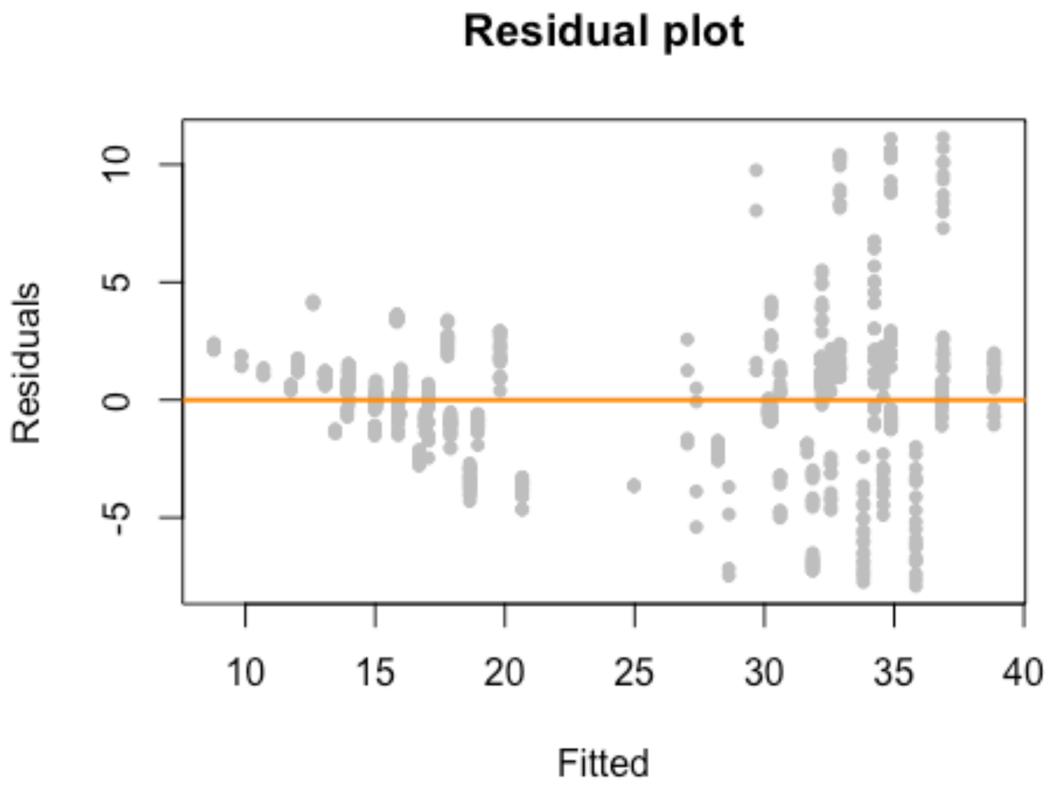
```r
## We use the pairs function to check relationship between the model variables
data_reduced = data_1[,-c(4,6,8)]
pairs(data_reduced)
```



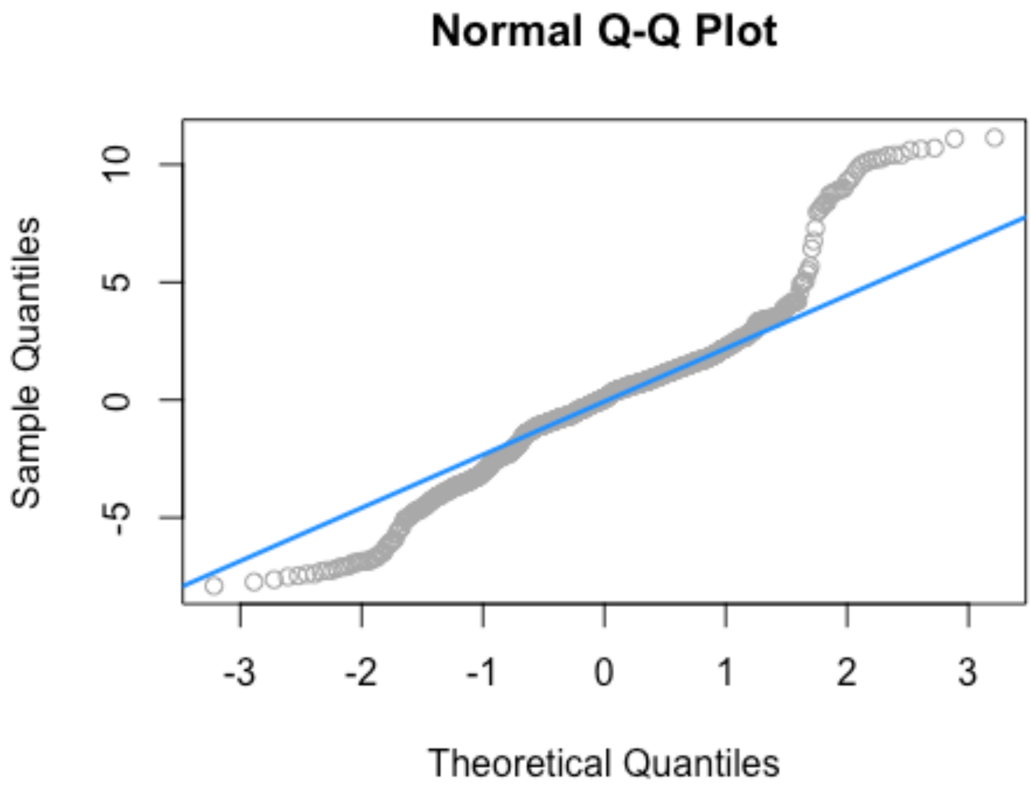
```
## It appears there is collinearity between variables so we check using the
VIF function
## In particular it appears that x1 is strongly correlated with x2
## Using the VIF functions over a 10 VIF indicates collinearity
vif(fit_final)

##          X1         X2         X3         X5      X70.1      X70.25
X70.4
## 105.524054 201.531134   7.492984  31.205474   4.125000   4.125000
4.125000

## Thus we remove x2 from the model and test the model with attributes
X1,X3,X5,X7
lm_final_2= lm(Y2~X1+X3+X5+X7,data=data_1)
data_reduced = data_1[,-c(2,4,6,8)]
## We now check the pairs graph and the vif function of the new model
pairs(data_reduced)
```



```
## The pairs graph shows there is less collinearity between the variables as
desired
vif(lm_final_2)

##        X1        X3        X5     X70.1    X70.25     X70.4
## 9.250283 3.161934 9.626103 4.125000 4.125000 4.125000

## All of the VIF values of the attributes are below 10 so there is no major
collinearity

## Now that we have done variable selection and analyzed to variable
interaction
## To obtain a final model with attributes X1,X3,X5,X7 we now test the three
## model assumptions for multiple linear regression : linearity, normality,
and
## equal variance of the residuals

## To check the model assumptions we first plot the rediduals to check the
equal
## variance and linearity assumptions of the model
plot(resid(lm_final_2)~fitted(lm_final_2), col = "grey", pch = 20, xlab =
"Fitted", ylab = "Residuals", main = "Residual plot")
abline(h = 0, col = "darkorange", lwd = 2)
```



```
## from the residual plot it appears that the linearity assumption is
violated because
## the residuals are not equally spread out accross the x axis, there
## appears to be a gap in the middle of the plot
## also from the residual plot it appears that the equal variance assumption
is
## violate because the residuals take very small values with smaller fitted
values
## and much larger residual values when the fitted value is larger

## We now use the qqplot to check if the normality assumption holds
qqnorm(resid(lm_final_2), main = "Normal Q-Q Plot", col = "darkgrey")
qqline(resid(lm_final_2), col = "dodgerblue", lwd = 2)
```



```
## from the qqplot it appears that the normality assumption is violated
## because of the very large tails at either end of the plot that do not
match
## a typical normal distribution

## We will now perform two more rigorous tests to verify the model
assumptions the bp test ## and the shapiro test
library(lmtest)

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## We perform the bptest for the equal variance assumption
bptest(lm_final_2)

##
##  studentized Breusch-Pagan test
##
```

```
## data:  lm_final_2
## BP = 198.46, df = 6, p-value < 2.2e-16

## The BP test has a very small p values so we reject the null hypothesis
that
## the residuals have equal variance, thus the equal variance assumption does
## not hold

## We now perform the shapiro test to check if the normality assumption holds
shapiro.test(resid(lm_final_2))

##
##  Shapiro-Wilk normality test
##
## data:  resid(lm_final_2)
## W = 0.94465, p-value = 2.457e-16

## The p-values is very small thus we reject the null hypothesis,
## Thus we can conclude that the normality assumption does not hold for our
model

## Thus we have determined that all three of the model assumptions for our
final
## model do not hold.


## Now using our model including 4 variables found in performing AIC and BIC
we will split the data into training and testing data to train the model for
prediction
## The predictors we are including are X1, X3, X5 and X7.
set.seed(8)
n = 768

## We will simply split the data 50/50.
idx <- sample(n,round(0.5*n),replace=FALSE)

## Here we are splitting the data up into their respective datasets. We have
Y as the response variable Y2 and X is the matrix containing all
## the predictor data from our original dataset including X1-X8.
X = model.matrix(Y2~.,data_1)[, -1] #the first column (for intercept) is
eliminated
y = data_1$Y2


## Training set
y_tr <- y[idx]
X_tr <- X[idx,]

## Testing set
y_ts <- y[-idx]
```

```
X_ts <- X[-idx,]

## In this next section we are going to see if the adjustments we made to the
model improved our models.
## We will look at the MSE of the full model versus the reduced model as well
as compare the r-squared values using numerical predictors versus using
categorical variables.

## We will begin with comparing the MSE of the full and reduced model.
##Now we are going to train the models including all the predictors
lm_train_mod_all = lm(Y2~., data_1[idx,])

## Here we are predicting the new values using the trained model we just
produced on the testing data.
pred_ls_all = predict(lm_train_mod_all, newdata=data_1[-idx,])

## Warning in predict.lm(lm_train_mod_all, newdata = data_1[-idx, ]):
prediction
## from a rank-deficient fit may be misleading

mse_ls_all  <- mean((pred_ls_all-y_ts)^2)
print(paste("The MSE of our model with all predictors is: ", mse_ls_all))

## [1] "The MSE of our model with all predictors is:  9.10228185541025"

## We observe a MSE of 10.2948314823287

## Next we will perform the same prediction test and calculate the MSE using
our model with only 4 predictors.
lm_train_mod = lm(Y2~X1+X3+X5+X7, data_1[idx,])
pred_ls = predict(lm_train_mod, newdata=data_1[-idx,])
mse_ls <- mean((pred_ls-y_ts)^2)
print(paste("The MSE of our improved model for prediction is: ", mse_ls))

## [1] "The MSE of our improved model for prediction is:  9.19462101543937"

##The reduced model produced an MSE of 10.3794565342607 which is higher than
the MSE of the full model by a sliver.


## Now we will quickly compare the models before and after converting
variables into cateogrical variables.
## The model chosen using AIC and BIC is Y2~X1+X3+X5+X7.
## Fitting the categorical model and viewing its output summary
mod_cat<-lm(Y2~X1+X3+X5+X7, data=data_1)
print(paste("The adjusted r-squared with categorical variables: ",
summary(mod_cat)$adj.r.squared))

## [1] "The adjusted r-squared with categorical variables:
0.885193132595794"
```

```r
# #The X7 predictor was previously changed to a categorical variable here we
are changing it back to a numeric variable before fitting the model
data_1$X7 <- as.numeric(data_1$X7)
mod_num<-lm(Y2~X1+X3+X5+X7, data=data_1)
print(paste("The adjusted r-squared with numerical variables: ",
summary(mod_num)$adj.r.squared))
```

```
## [1] "The adjusted r-squared with numerical variables:  0.88477359265234"
```

```r
## The adjusted r-squared values show that changing X7 to a cateogrical
variable has not significantly impacted our model but has improved it by a
small fraction.
```

Summary of the Output and Results

After performing our multiple linear regression analysis on the Energy Efficiency Data Set we have defined our model to predict cooling load in terms of attributes X1-X8 to be the model with Y2~X1+X3+X5+X7. We used AIC and BIC with stepwise selection to pick these attributes. We tested the model assumptions of normality, equal variance and linearity using residual plots, qqplots, the bptest and the shapiro test and found that all of the model assumptions did not hold. We tested the final model using VIF and found that there was no significant interaction between the terms in the model. Finally we compared the MSE of the full model and the reduced model to capture the improvements made to the model. We can view a summary of the of the model along with its goodness of fit as follows as follows:

```r
summary(lm_final_2)
```

```
##
## Call:
## lm(formula = Y2 ~ X1 + X3 + X5 + X7, data = data_1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.9036 -1.5945  0.0465  1.4520 11.1417
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.040951   2.929190  -0.355    0.722
## X1          -19.996672   3.346586  -5.975 3.53e-09 ***
## X3            0.018562   0.004744   3.913 9.94e-05 ***
## X5            5.736372   0.206216  27.817  < 2e-16 ***
## X70.1         3.229292   0.509666   6.336 4.03e-10 ***
## X70.25        5.186375   0.509666  10.176  < 2e-16 ***
## X70.4         7.205167   0.509666  14.137  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.223 on 761 degrees of freedom
## Multiple R-squared:  0.8861, Adjusted R-squared:  0.8852
## F-statistic: 986.6 on 6 and 761 DF,  p-value: < 2.2e-16
```

```r
AIC(lm_final_2)

## [1] 3986.252

BIC(lm_final_2)

## [1] 4023.403

summary(lm_final_2)$r.squared

## [1] 0.8860912

summary(lm_final_2)$adj.r.squared

## [1] 0.8851931

print(paste("The MSE of our model with all predictors is: ", mse_ls_all))

## [1] "The MSE of our model with all predictors is:  9.10228185541025"

print(paste("The MSE of our improved model for prediction is: ", mse_ls))

## [1] "The MSE of our improved model for prediction is:  9.19462101543937"

print(paste("The adjusted r-squared with categorical variables: ",
summary(mod_cat)$adj.r.squared))

## [1] "The adjusted r-squared with categorical variables:
0.885193132595794"

print(paste("The adjusted r-squared with numerical variables: ",
summary(mod_num)$adj.r.squared))

## [1] "The adjusted r-squared with numerical variables:  0.88477359265234"
```

Thus we have found that the R^2 value of our model is 0.8861 and the adjusted R^2 value of the model is 0.8852. THE AIC is 3986.252 and the BIC is 4023.403, from the summary we can see that all of the values are statistically significant because fo their small p values and from the VIF analysis since all of the VIF values are below 10 we can conclude there is no significant colinearity in the model. However we have that all three model assumptions are violated when we verify them graphically and statistically.

Discussion of the findings and inferences

From the results of our multiple regression analysis we have found a multiple linear regression model, Y2~X1+X3+X5+X7 which can predict the 88.52% of the change in the response variable, however our model violates all three of the model assumptions: linearity, equal variance and normality for residuals. Our goal was to find a multiple regression model to predict cooling load from the attributes X1-X8 in the Energy Efficiency Data Set. We were able to find a model with a good R^2 by using stepwise variable selection with AIC and BIC, and reduce collinearity using VIF, however we were unable to find a model that does not violate the assumptions of the model, thus we have not completed our goal for the multiple regression analysis. Overall by using the tools we learned in class to perform a multiple regression analysis on the data we where unable to obtain a model that performed the goals we set out to do. Perhaps there exist other statistical methods beyond those learned in class that would be better suited to take the data given to us and create a model to predict the response variable cooling load from the given attributes. To conclude the tools given to us were not sufficient to create a regression model which can predict cooling load from the Energy Efficiency Data Set.

Limitations and further questions raised by the study

From our analysis and discussion of the multiple linear regression analysis we can see that our model does not satisfy the model assumptions. We would like to see if there are other statistical techniques which could produce a multiple linear regression model in which all three model assumptions, linearity, equal variance and normality can hold. Also the R^2 value of the model was 0.8852, that means that aproximately 88.5% of the variation in y2 can be explained by the model, thus we know that our model is not perfect and we would seek to improve upon the model to find one with a better goodness of fit. Perhaps there are models or statistical techniques which could produce a superior model to for the Energy Efficiency data set. When we did a literature review, we found that there were other statistical methods to predict the value of y2 using machine learning techniques such as random forests and decision trees which obtained a R^2 value of into the high ninety percent. This brings us to the question of how to obtain the optimal model for predicting our response variable and what we could do to improve our model, and also determine if there are other models much better suited to solving our problem.