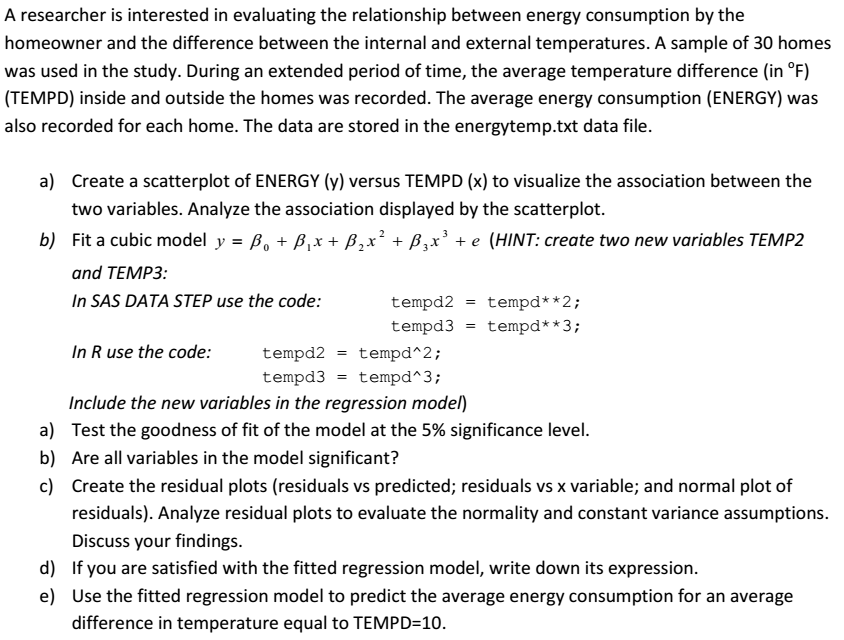
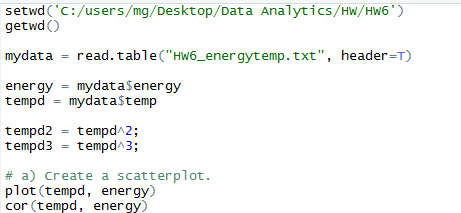
**HW6: Multiple Linear Regression Analysis**Note: every step you use R, you should provide the snapshots of your R commands and R outputs

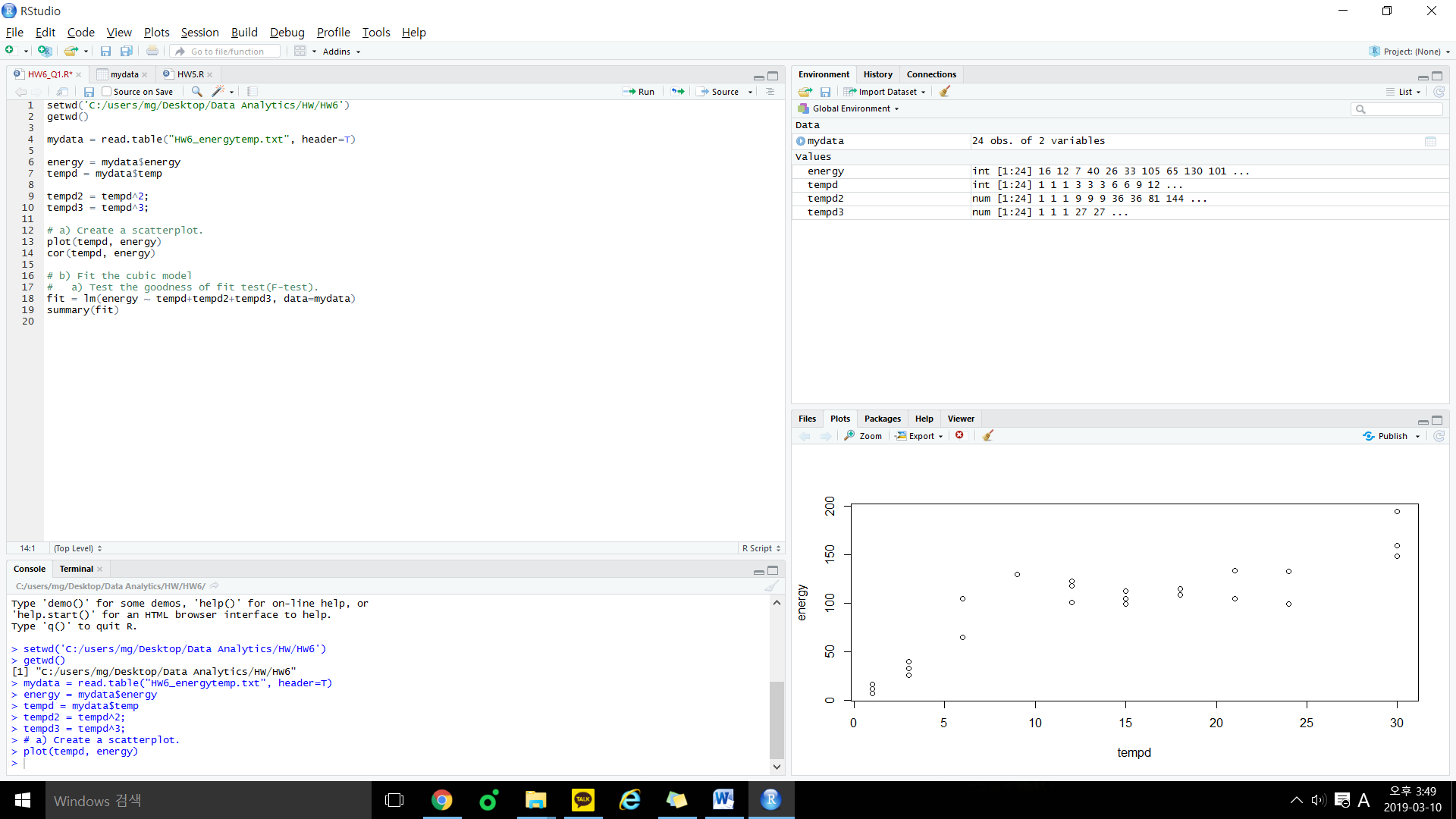
**Problem 1 [30]**

 In R, you should use, new = data.frame (tempd=c(10), tempd2=c(100), tempd3=c(1000)),   
 and then use the predict() function in R to produce predictions and confidence interval

f) By using influence.measures() function to identify whether there are influential points that can affect your final model. Use cook’s distance as the metric to identify the influential points

**a)**

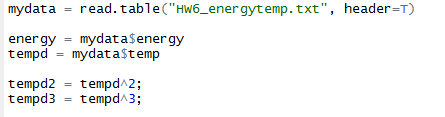




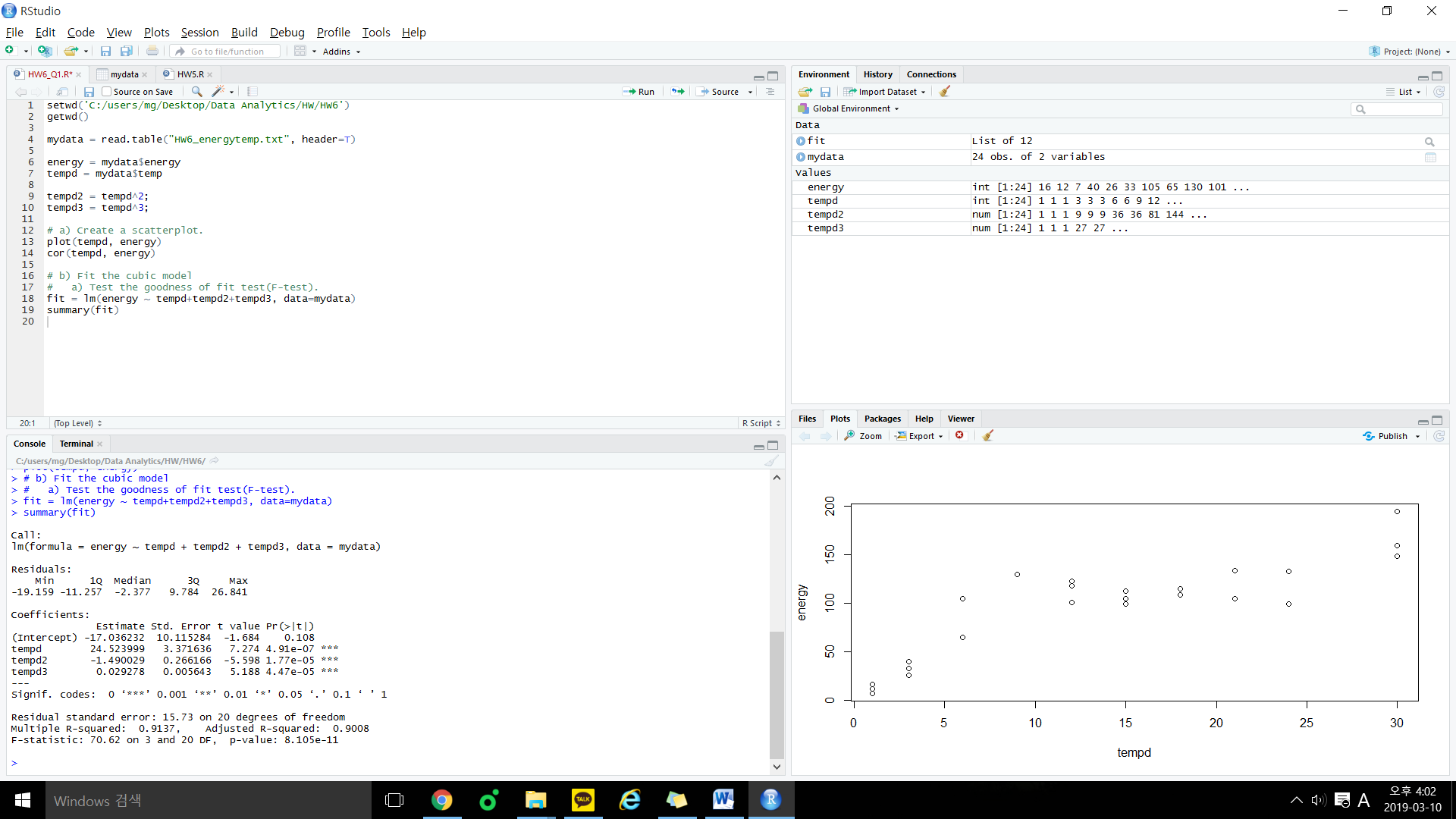
It looks like not a linear relationship. It looks like 3rd order relationship.

So we need a transformation of x variables.

**b)**



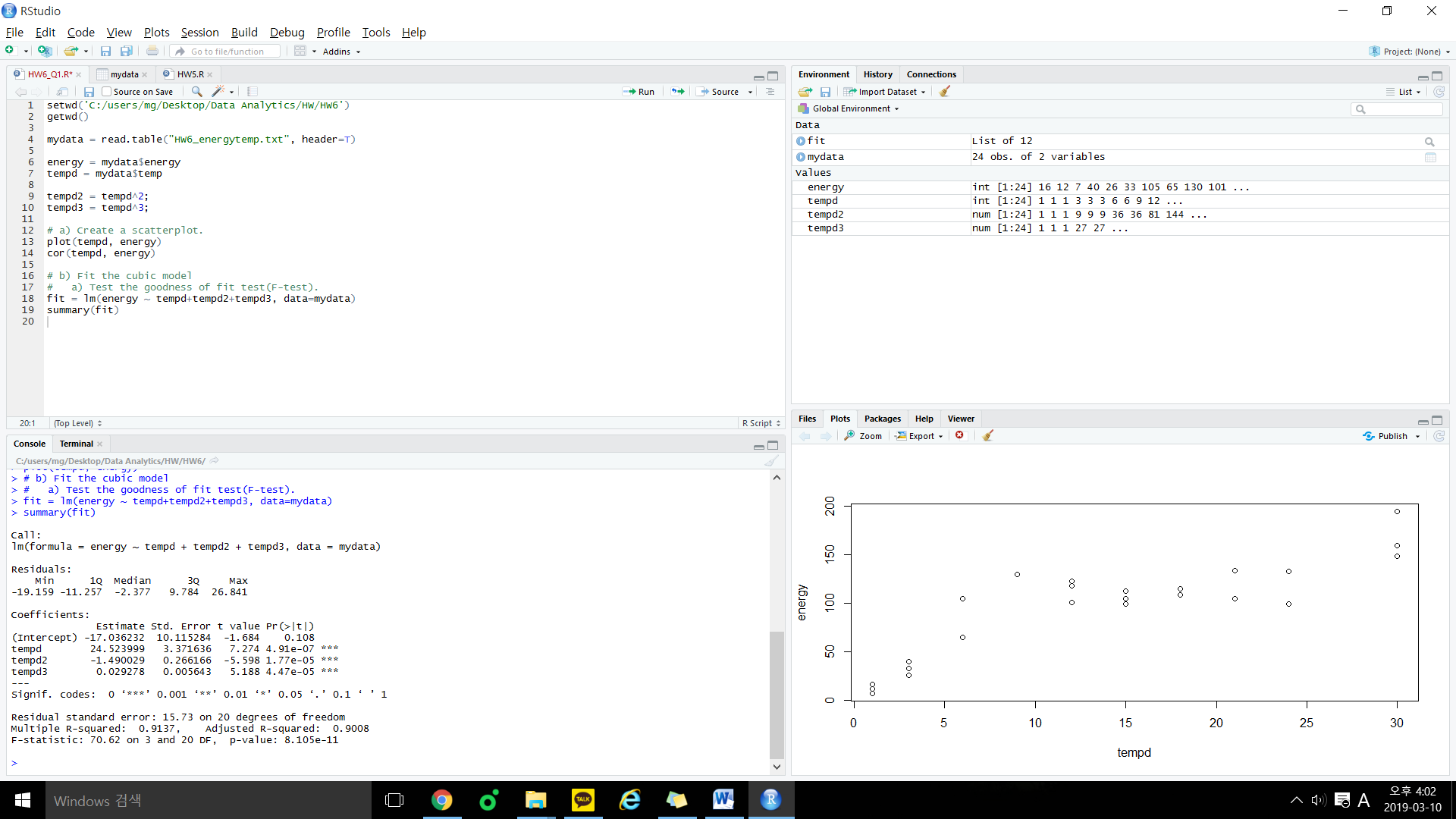
**a)**



Since P-value is less than 0.05, At 95% confidence level, I can say that at least one x variable has significant linear relationship with y, and it can affect the value of the y.

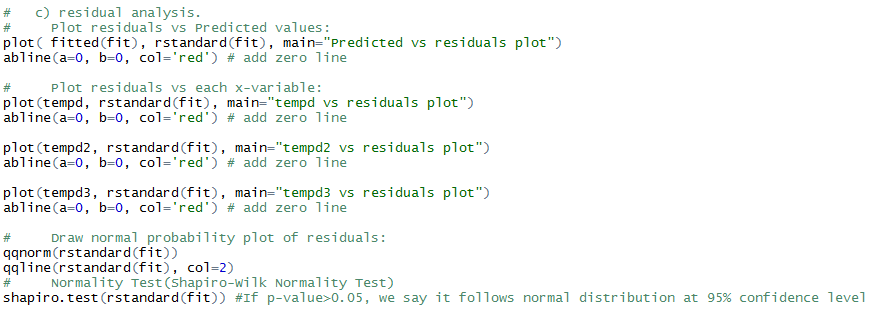
**b)**

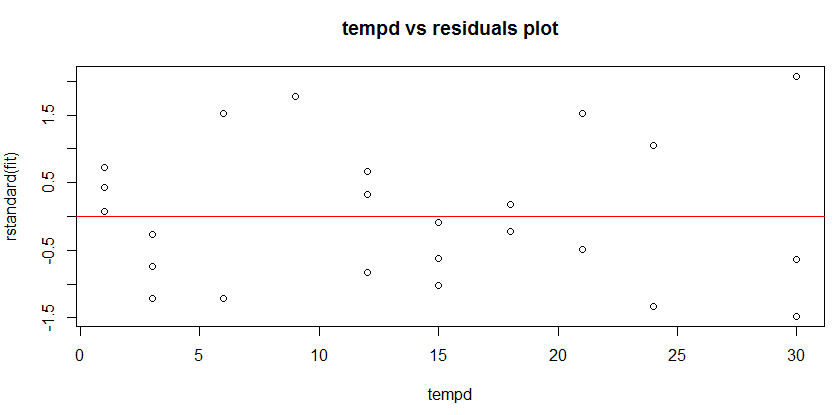
If we want to know all variables in the model is significant, we should conduct a individual parameter test.

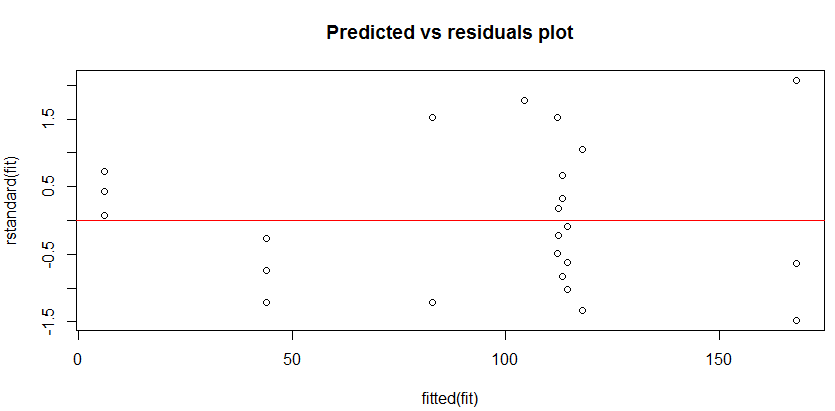


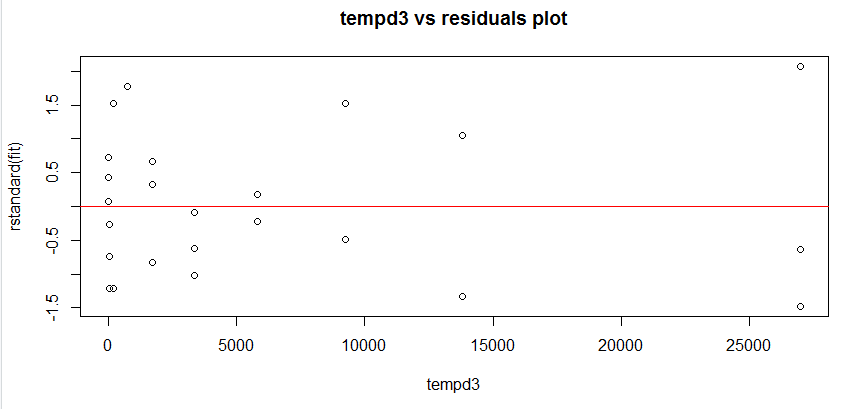
Since P-value of all of x variables is less than 0.05, we know that all of x variables are significant.

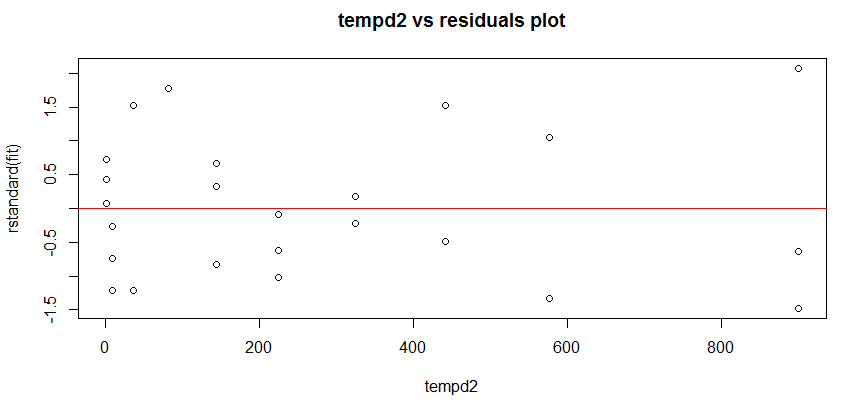
**c)**

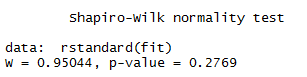


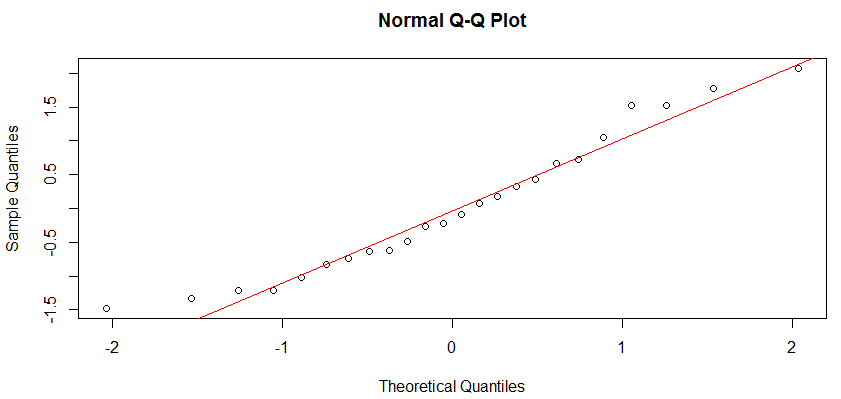










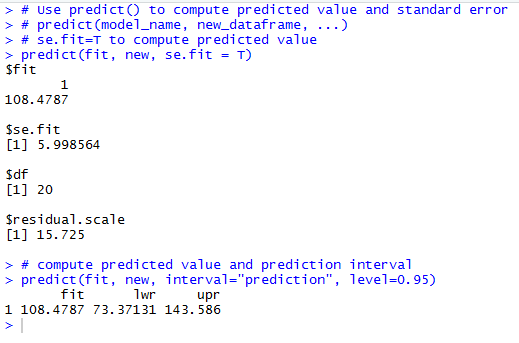


Spread is quite constant. It seems like transformation on y is not necessary.

And if you see the Normal Q-Q Plot, we can see points are follow the linear line but I am not sure the part below the -1.5. So I conducted Normality test and got a result which is normal distribution of residuals at 95% confidence level.

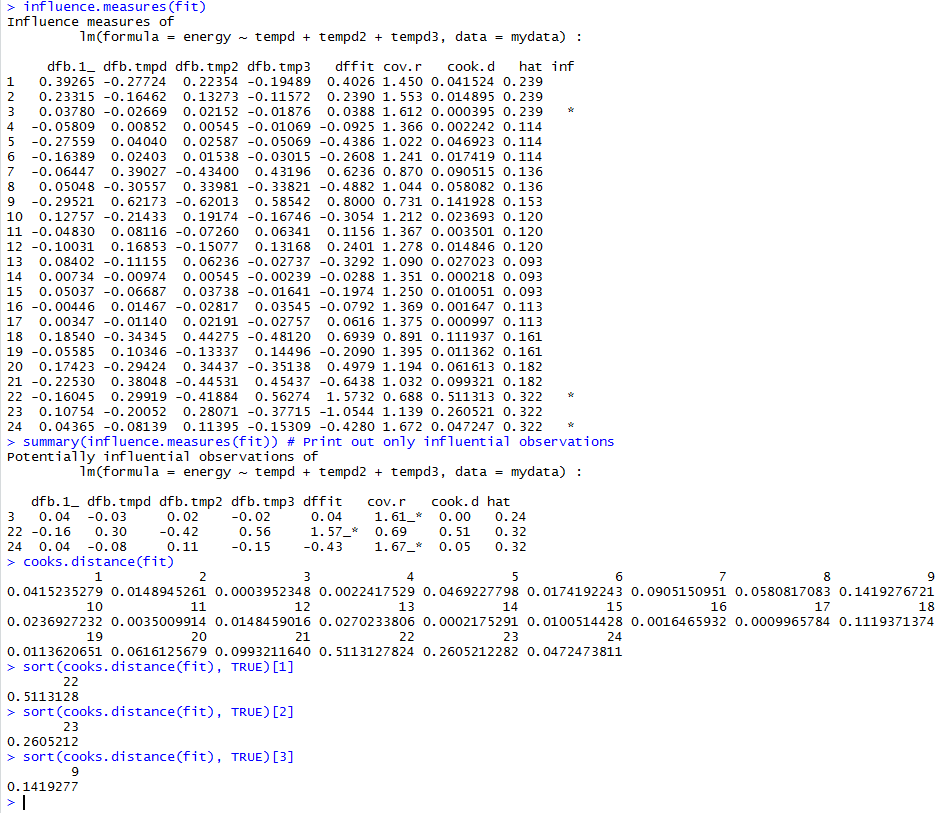
**d)**

**e)**



Prediction = 108.4787, Confidence Interval = [73.371, 143.586]

**f)**



I could find influential points with cooks.distance.

But the data is small, so I cannot remove all of them.

Therefore, considering the size of the data, i chose two indexes with a cooks.distance value that meets the criteria(> 4 / n = 4 / 24 = 0.167)

**OUTPUT of Q1**

setwd('C:/users/mg/Desktop/Data Analytics/HW/HW6')

getwd()

mydata = read.table("HW6\_energytemp.txt", header=T)

energy = mydata$energy

tempd = mydata$temp

tempd2 = tempd^2;

tempd3 = tempd^3;

# a) Create a scatterplot.

plot(tempd, energy)

cor(tempd, energy)

# b) Fit the cubic model

# a) Test the goodness of fit test(F-test).

# b) Are all variables in the model significant?

fit = lm(energy ~ tempd+tempd2+tempd3, data=mydata)

summary(fit)

# c) residual analysis.

# Plot residuals vs Predicted values:

plot( fitted(fit), rstandard(fit), main="Predicted vs residuals plot")

abline(a=0, b=0, col='red') # add zero line

# Plot residuals vs each x-variable:

plot(tempd, rstandard(fit), main="tempd vs residuals plot")

abline(a=0, b=0, col='red') # add zero line

plot(tempd2, rstandard(fit), main="tempd2 vs residuals plot")

abline(a=0, b=0, col='red') # add zero line

plot(tempd3, rstandard(fit), main="tempd3 vs residuals plot")

abline(a=0, b=0, col='red') # add zero line

# Draw normal probability plot of residuals:

qqnorm(rstandard(fit))

qqline(rstandard(fit), col=2)

# Normality Test(Shapiro-Wilk Normality Test)

shapiro.test(rstandard(fit)) #If p-value>0.05, we say it follows normal distribution at 95% confidence level

# e) Use the fitted regression model to predict the average energy consumption for an average differnece

# in temperature equal to TEMPD = 10

# and use the predict() function in R to produce predictions and confidence interval

# Prediction for one certain data point.

# Create new data frame containing xvalues for prediction.

new = data.frame (tempd=c(10), tempd2=c(100), tempd3=c(1000))

# Use predict() to compute predicted value and standard error

# predict(model\_name, new\_dataframe, ...)

# se.fit=T to compute predicted value

predict(fit, new, se.fit = T)

# compute predicted value and prediction interval

predict(fit, new, interval="prediction", level=0.95)

# f) By using influence.measures() function to identify whether there are influential points that

# can affect your final model. Use cook’s distance as the metric to identify the influential points

influence.measures(fit)

summary(influence.measures(fit)) # Print out only influential observations

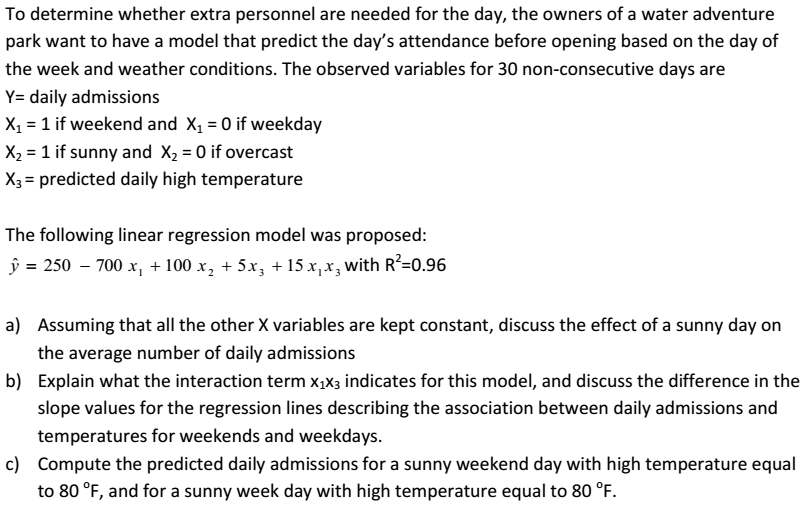
cooks.distance(fit)

sort(cooks.distance(fit), TRUE)[1]

sort(cooks.distance(fit), TRUE)[2]

sort(cooks.distance(fit), TRUE)[3]

**Problem 2 [15]**



1. If the weather is sunny, daily admissions will increase by 100.
2. Interaction term x1x3 indicates that, on weekends, predicted daily high temperature affects daily admissions. On the other hand, on weekdays, predicted daily high temperature does not affects daily admissions.

The daily admission for weekdays and weekends has a partial slope.

In the case of slope of predicted daily high temperature,

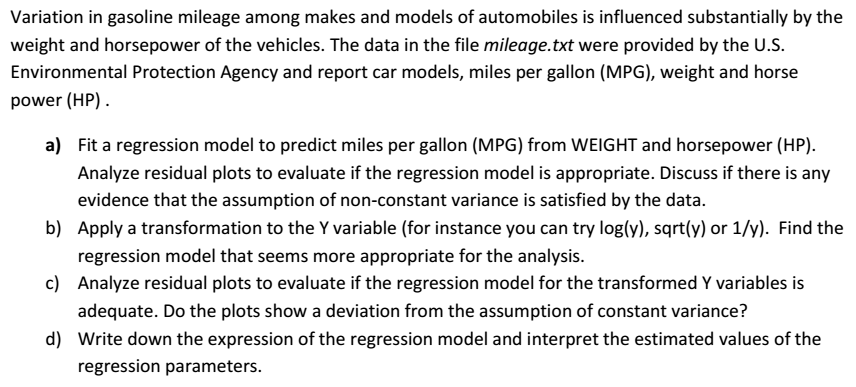
slope of predicted daily high temperature of weekdays is equal to 5.

slope of predicted daily high temperature of weekends is equal to 20.

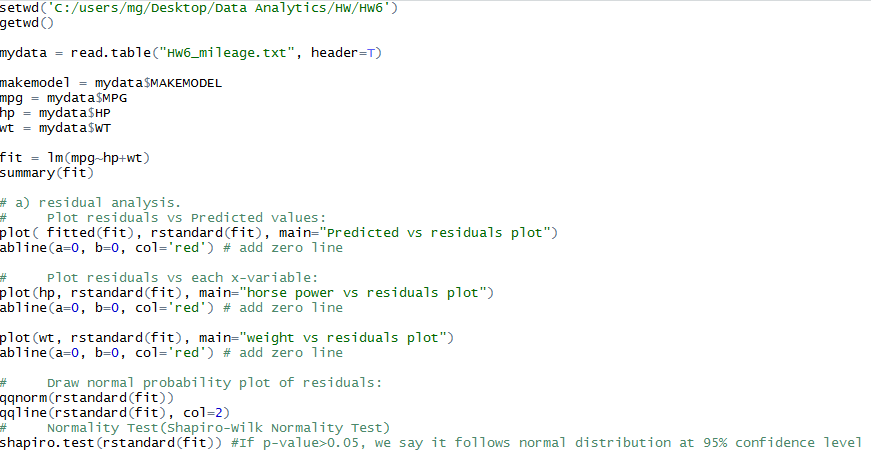
Sunny weekend day with high temperature equal to 80F) X1 = 1, x2 = 1, x3 = 80

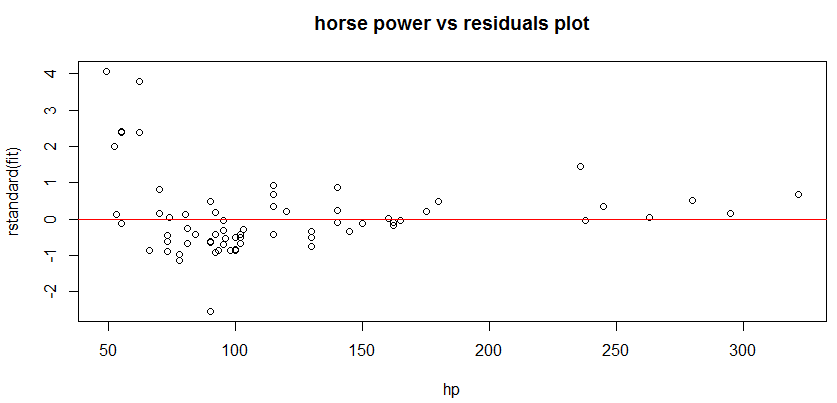
Sunny week day with high temperature equal to 80F) X1 = 0, x2 = 1, x3 = 80.

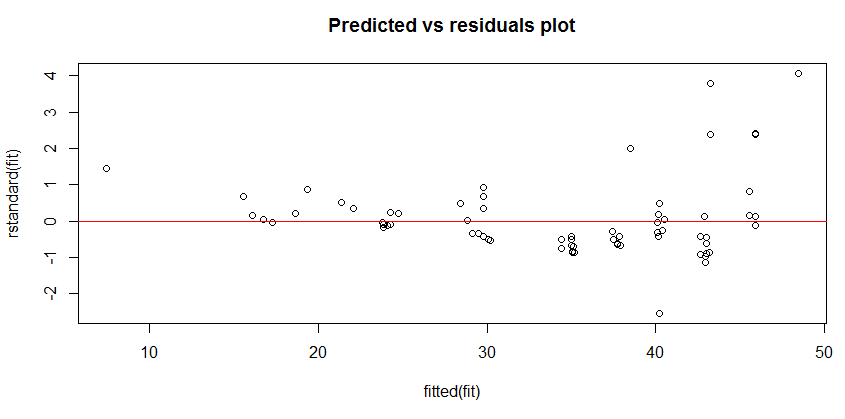
**Problem 3 [25]**

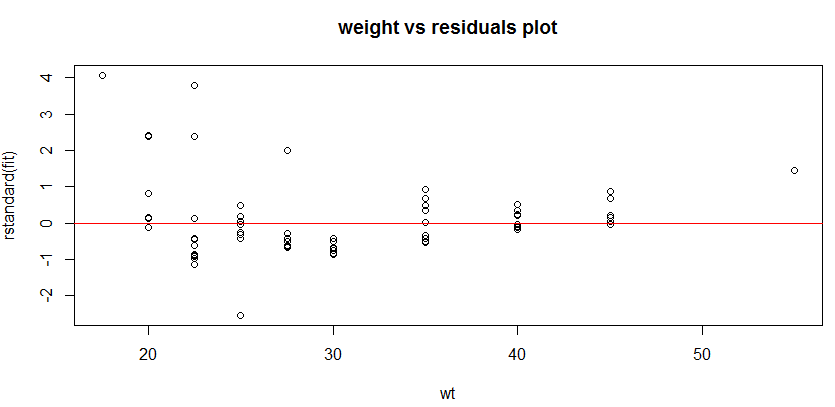


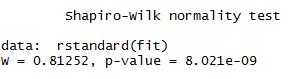
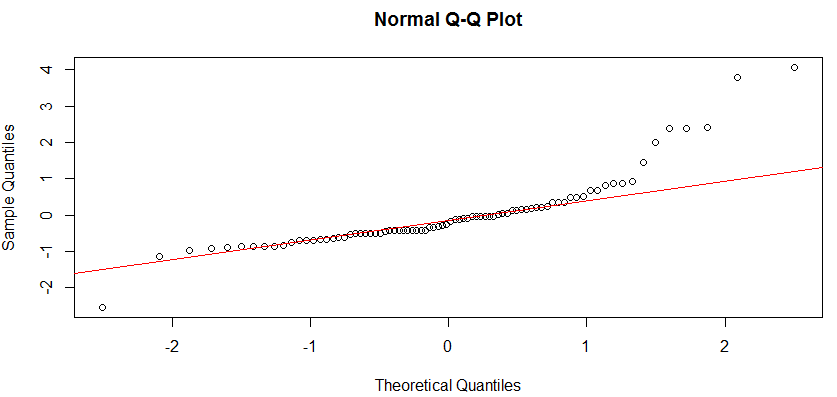
**a)**











(Y = MPG, X1 = WT, X2 = HP)

I can see the increasing pattern of the ‘Predicted vs residuals’ plot.

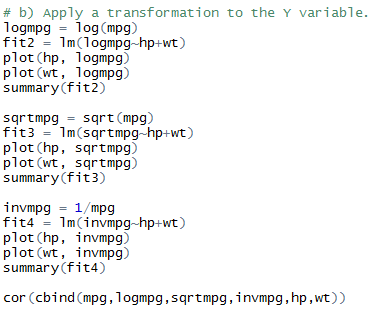
On the other hand, i can’t see any patterns but i can see a few outliers from the plots which are comparing x variables with residuals plot.

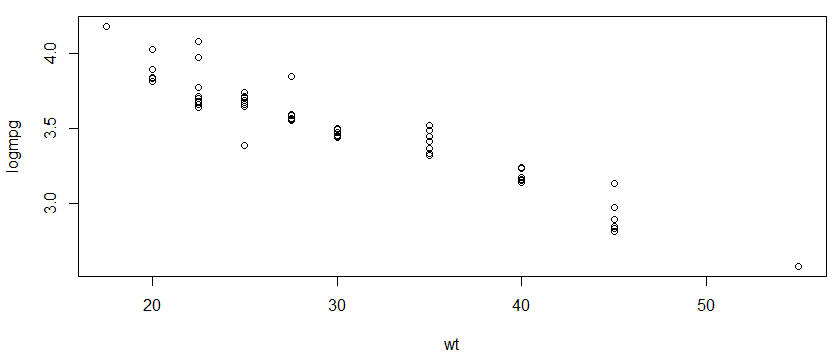
And I am not sure that variables follow normal distribution through QQ plot, so I conducted Normality test.

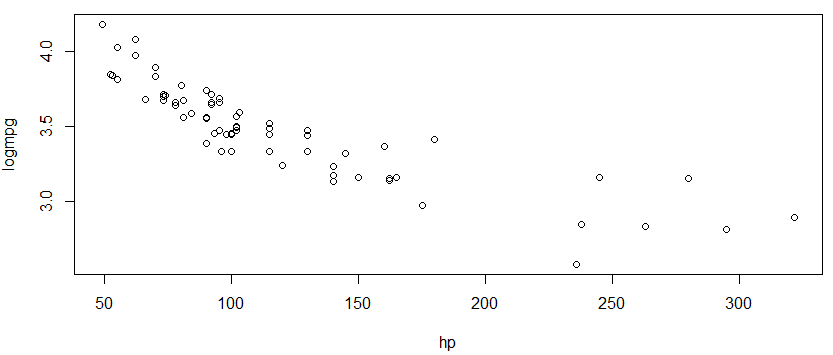
As a result, P-value is smaller than 0.05, so it doesn’t follows normal distribution at 95% confidence level.

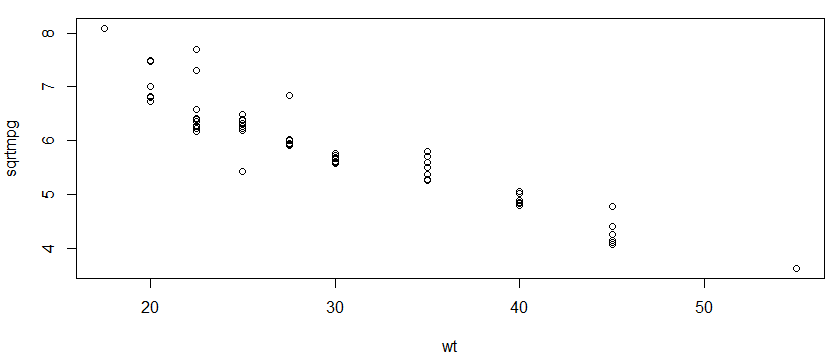
Therefore, regression models are not appropriate.

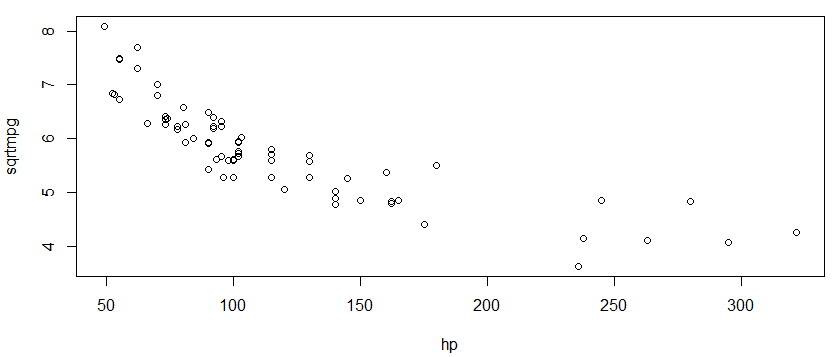
**b)**

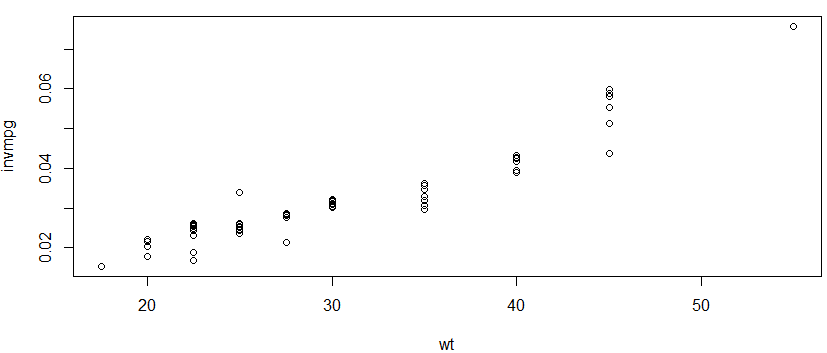


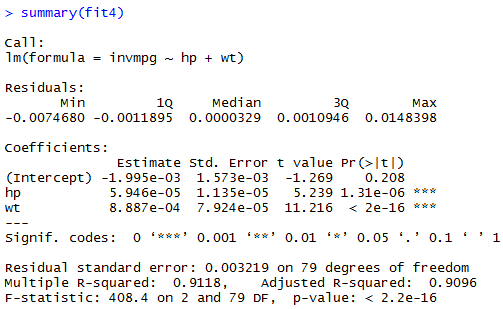
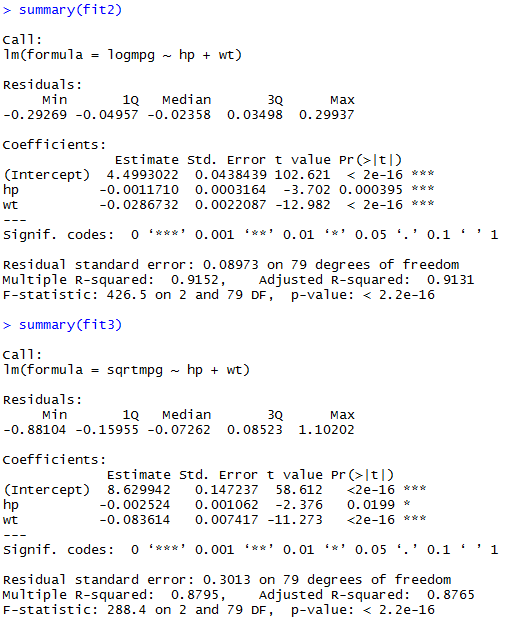
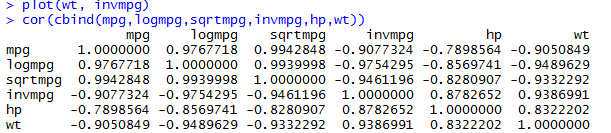
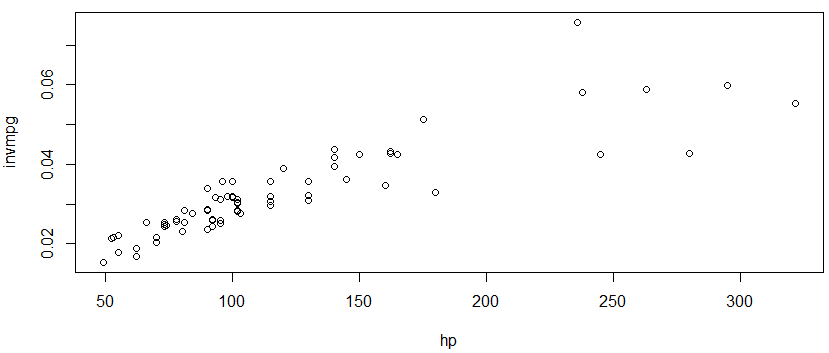










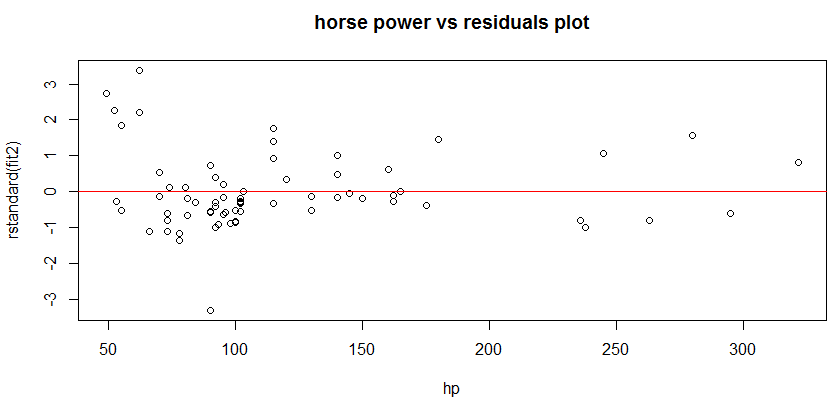
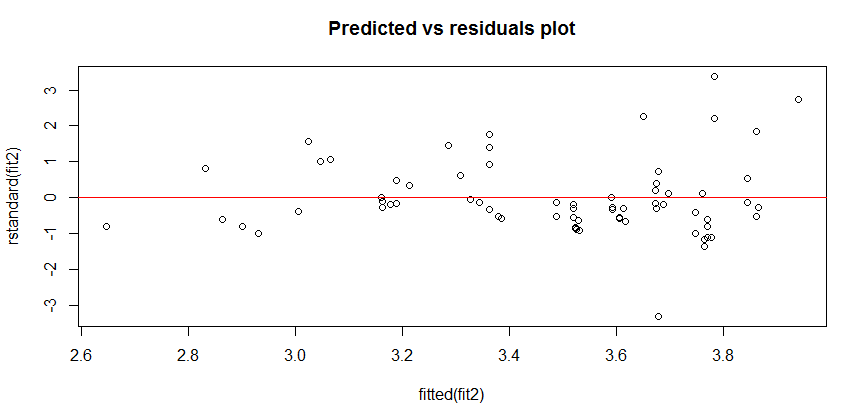
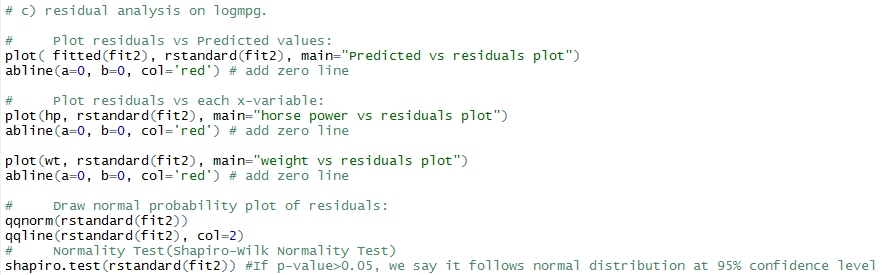


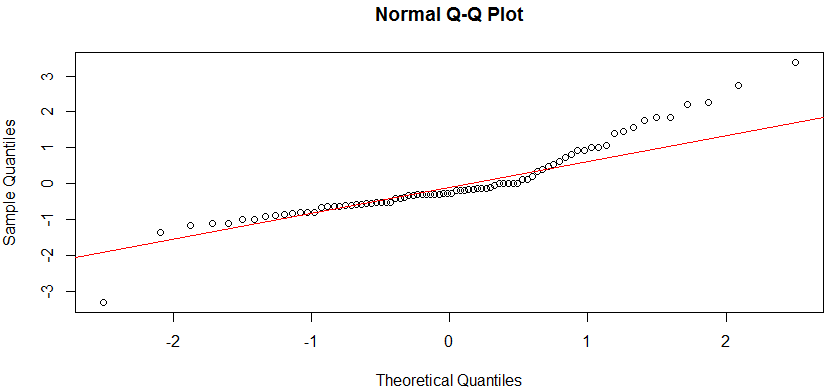
I made transformations of y variable which are logmpg = log(mpg), sqrtmpg = sqrt(mpg), invmpg = 1/mpg.

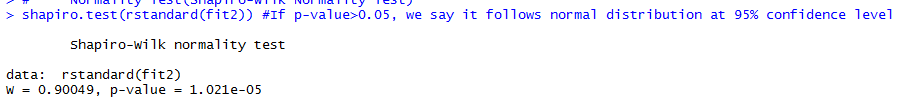
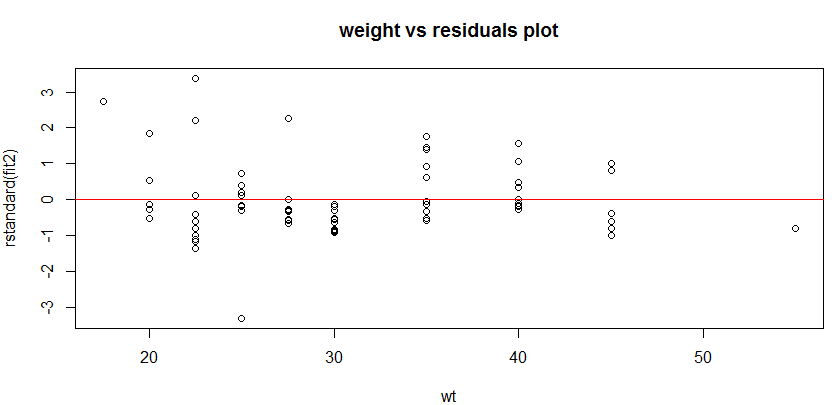
Then I compared plots and correlation.

When considering plots, correlation and adj-R2, I chose logmpg.

**c)**



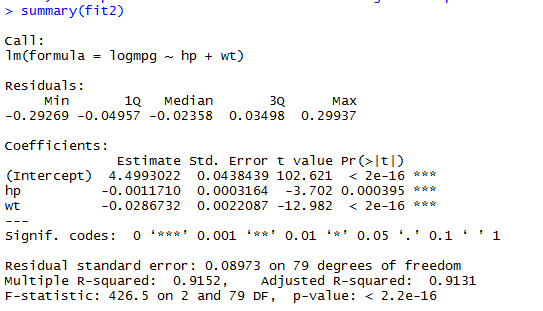




I can see a few outliers in plots but also I can see a constant variance from each plots.

Given QQ plot and Normality test, this model does not follow the normal distribution.

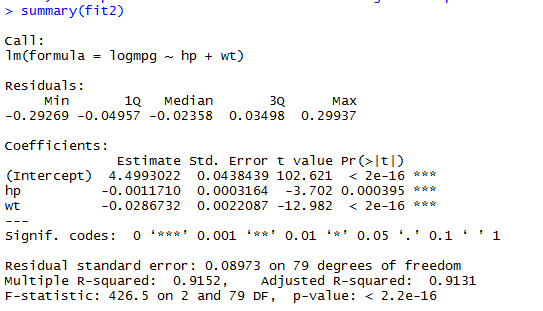
**d)**

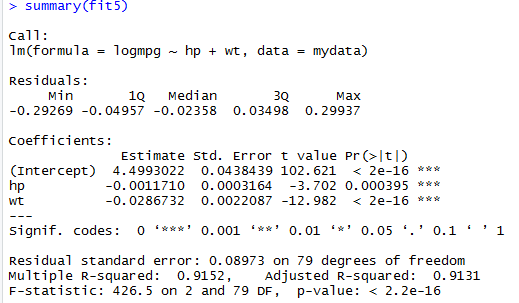


(Y = log(MPG), X1 = WT, X2 = HP)

Y intercept value is equal to 4.50, regression parameter of X1 is equal to -0.001 and regression parameter of X2 is equal to -0.03.

**e)**





Given adj-R2 value, Both models are the same.

e). Use the step function to adopt both backward and stepwise forward selection to build a   
 new model, compare the new model with the previous model in terms of the adj-R2 value.   
 Hint, you can use the step function, but set direction=”both”

**OUTPUT of Q3**

setwd('C:/users/mg/Desktop/Data Analytics/HW/HW6')

getwd()

mydata = read.table("HW6\_mileage.txt", header=T)

makemodel = mydata$MAKEMODEL

mpg = mydata$MPG

hp = mydata$HP

wt = mydata$WT

fit = lm(mpg~hp+wt)

summary(fit)

# a) residual analysis.

# Plot residuals vs Predicted values:

plot( fitted(fit), rstandard(fit), main="Predicted vs residuals plot")

abline(a=0, b=0, col='red') # add zero line

# Plot residuals vs each x-variable:

plot(hp, rstandard(fit), main="horse power vs residuals plot")

abline(a=0, b=0, col='red') # add zero line

plot(wt, rstandard(fit), main="weight vs residuals plot")

abline(a=0, b=0, col='red') # add zero line

# Draw normal probability plot of residuals:

qqnorm(rstandard(fit))

qqline(rstandard(fit), col=2)

# Normality Test(Shapiro-Wilk Normality Test)

shapiro.test(rstandard(fit)) #If p-value>0.05, we say it follows normal distribution at 95% confidence level

# b) Apply a transformation to the Y variable.

logmpg = log(mpg)

fit2 = lm(logmpg~hp+wt)

plot(hp, logmpg)

plot(wt, logmpg)

summary(fit2)

sqrtmpg = sqrt(mpg)

fit3 = lm(sqrtmpg~hp+wt)

plot(hp, sqrtmpg)

plot(wt, sqrtmpg)

summary(fit3)

invmpg = 1/mpg

fit4 = lm(invmpg~hp+wt)

plot(hp, invmpg)

plot(wt, invmpg)

summary(fit4)

cor(cbind(mpg,logmpg,sqrtmpg,invmpg,hp,wt))

# c) residual analysis on logmpg.

# Plot residuals vs Predicted values:

plot( fitted(fit2), rstandard(fit2), main="Predicted vs residuals plot")

abline(a=0, b=0, col='red') # add zero line

# Plot residuals vs each x-variable:

plot(hp, rstandard(fit2), main="horse power vs residuals plot")

abline(a=0, b=0, col='red') # add zero line

plot(wt, rstandard(fit2), main="weight vs residuals plot")

abline(a=0, b=0, col='red') # add zero line

# Draw normal probability plot of residuals:

qqnorm(rstandard(fit2))

qqline(rstandard(fit2), col=2)

# Normality Test(Shapiro-Wilk Normality Test)

shapiro.test(rstandard(fit2)) #If p-value>0.05, we say it follows normal distribution at 95% confidence level

# d) Interpret te estimated values of the regression parameters

summary(fit2)

# e) Use the step() function.

base = lm(logmpg~hp, data=mydata)

full = lm(logmpg~hp+wt, data=mydata)

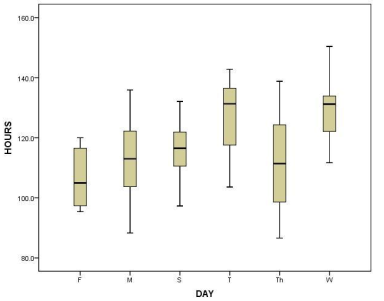
fit5 = step(base, scope=list(upper=full, lower=~1), direction="both", trace=T)

summary(fit5)

**Problem 4 [30]**

This time, let’s use the “HW5\_clerical\_Q2.txt” data.

A store manager noticed that the busiest days for clerical staff are Wednesdays and Tuesdays. See enclosed box plot. The manage tries to compare the group means in hours by different days



**a).** [5] Observe the box plot. Can you confirm that the hours in Tuesday is the highest? Why?

We usually use mean or median to represent a set of quantitative data. So, Given the median value, you can see that Tuesday and Wednesday are the highest.

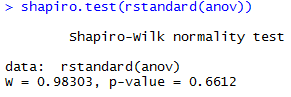
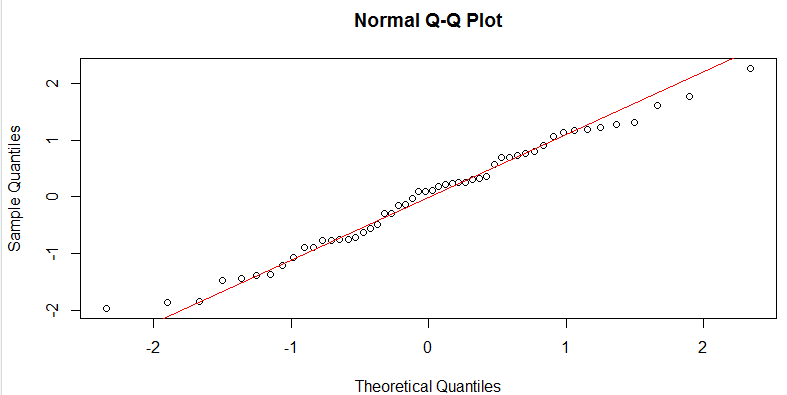
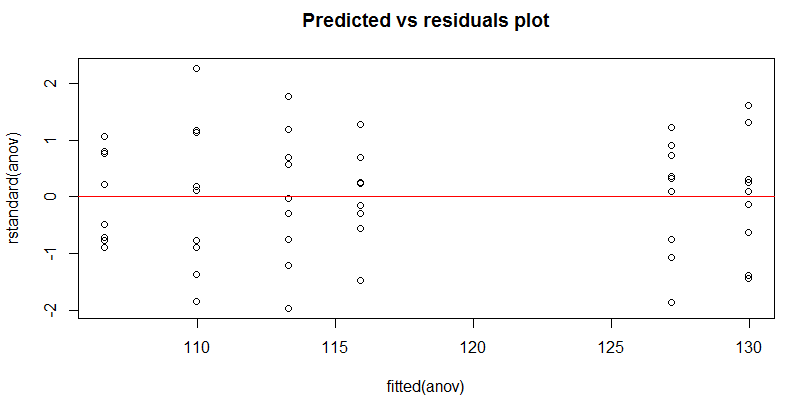
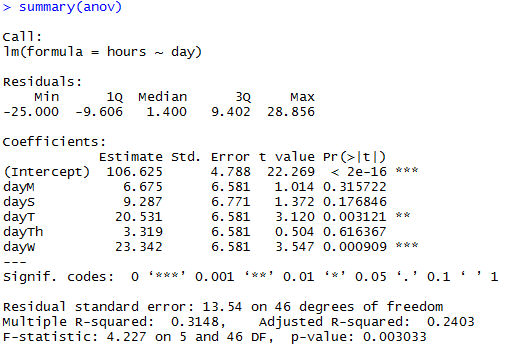
However, since some boxes are very large in variance, we can see that q2 does not represent a whole peace of data well.

**b).** [5] Write down your hypothesis in the ANOVA to compare the group means in hours by different days.

**Ho : The average number of hours of each days are all equal.**

**Ha : Not all the averages are equal.**

**c).** [10]Using R to build the ANOVA regression model, and help the manager to make the decision whether the group means in hours or different days are the same or not.



Because the p-value is less than 0.05, the null hypothesis is rejected and the alternative hypothesis is adopted. Therefore, not all the averages are equal.

**d).** [10] Try to interpret the coefficients you got in the ANOVA regression model from part c).

The individual parameter test on Monday, Saturday, Thursday is not significant, so it indicates that the average number of hours on Monday, Thursday, Friday and Saturday are not quite different.

But Tuesday and Wednesday are a bit different.

Tuesdays and Wednesdays are approximately 20 hours longer than Fridays.

**OUTPUT of Q4**

setwd('C:/users/mg/Desktop/Data Analytics/HW/HW6')

getwd()

data4 = read.table("HW6\_clerical\_Q2.txt", header=T)

hours=data4$HOURS

day=data4$DAY

plot(hours~day)

table(day)

summary(data4)

# Hours of each days

mon=data4[which(day=='M'),]

tus=data4[which(day=='T'),]

wed=data4[which(day=='W'),]

thu=data4[which(day=='Th'),]

fri=data4[which(day=='F'),]

sat=data4[which(day=='S'),]

anov=lm(hours ~ day)

summary(anov)

# c) residual analysis.

# Plot residuals vs Predicted values:

plot( fitted(anov), rstandard(anov), main="Predicted vs residuals plot")

abline(a=0, b=0, col='red') # add zero line

# Draw normal probability plot of residuals:

qqnorm(rstandard(anov))

qqline(rstandard(anov), col=2)

# Normality Test(Shapiro-Wilk Normality Test)

shapiro.test(rstandard(anov))