**Project 2 – CSC-423**

**Joseph Berkowitz**

**8/1/2015**

**Part A. Flour Dataset (40 pts.)**

**Use the**[**flour dataset**](http://facweb.cdm.depaul.edu/sjost/csc423/projects/flour.txt)**to do these problems. This dataset is based on a mill that grinds grain into flour.  Each time a batch of flour is ground a certain number of pounds of flower is produces (independent variable Weight).  This flour is put into bags for shipping.  The number of bags needed is the dependent variable NBags.**

1. **Create and print a SAS dataset or R dataframe named flour.**

> #Reading in flour data into R

> flour <- read.table("c:/DataSets/flour.txt", header=T)

> print(flour)

Weight NBags

1 5050 100

2 10249 205

3 20000 450

4 7420 150

5 24685 500

6 10206 200

7 7325 150

8 4958 100

9 7162 150

10 24000 500

11 4900 100

12 14501 300

13 28000 600

14 17002 400

15 16100 400

> #Create Weight and NBags variables

> Weight <- flour$Weight

> NBags <- flour$NBags

> print(Weight)

[1] 5050 10249 20000 7420 24685 10206 7325 4958 7162 24000 4900 14501

[13] 28000 17002 16100

> print(NBags)

[1] 100 205 450 150 500 200 150 100 150 500 100 300 600 400 400

1. **Use SAS or R to compute the means and standard deviations for weight and nbags. Also compute the correlation between weight and nbags.**

Mean of Weight: 13437.2

Mean of NBags: 287

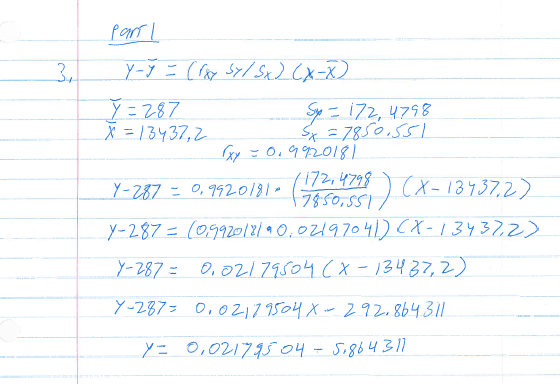
Standard Deviation of Weight: 7850.551

Standard Deviation of NBags: 172.4798

Correlation between Weight and NBags: 0.9920181

1. **Compute the regression model by hand using the formula**

**y - y = (rxy sy / sx)(x - x)**



1. **Use SAS or R to find the simple linear regression model for predicting nbags from weight. Compare your hand calculations in Question A3 to the simple linear regression model obtained by SAS or R.**

> regmodel <- lm(NBags ~ Weight, data=flourFrame)

> print(summary(regmodel))

Call:

lm(formula = NBags ~ Weight, data = flourFrame)

Residuals:

Min 1Q Median 3Q Max

-32.146 -11.349 -4.201 -0.582 54.964

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -5.8643573 11.8557460 -0.495 0.629

Weight 0.0217950 0.0007684 28.366 4.45e-13 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 22.57 on 13 degrees of freedom

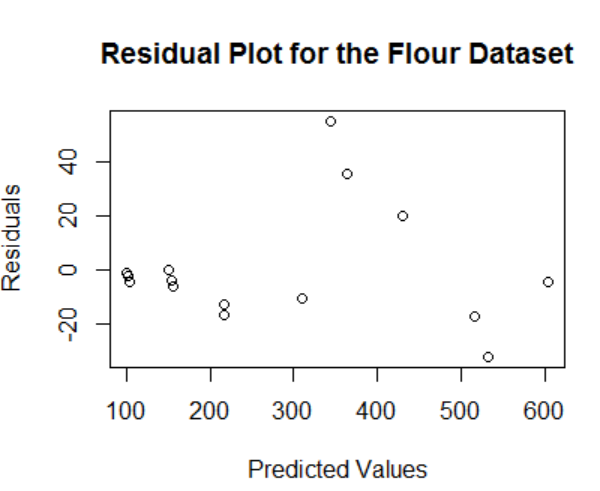
Multiple R-squared: 0.9841, Adjusted R-squared: 0.9829

F-statistic: 804.6 on 1 and 13 DF, p-value: 4.454e-13

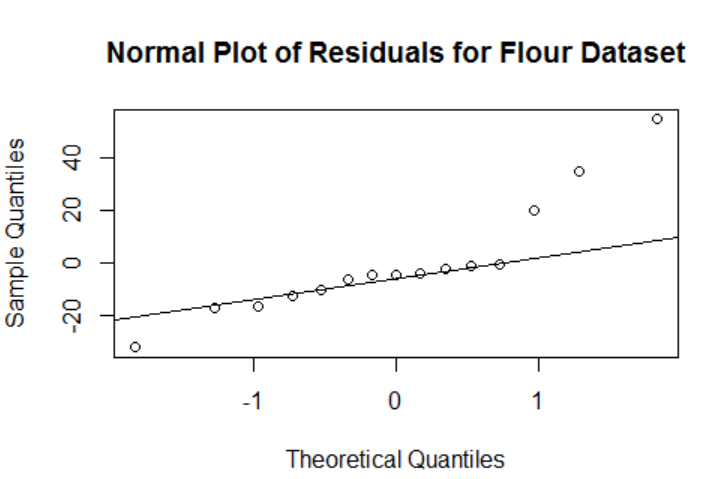
My analysis of the regression model by hand and R’s analysis are very similar. We both had the intercept at approximately -5.864, and the x coefficient to be 0.021795.

1. **For the simple linear regression model, create and interpret the residual plot and normal plot of the residuals.**

Residual Plot



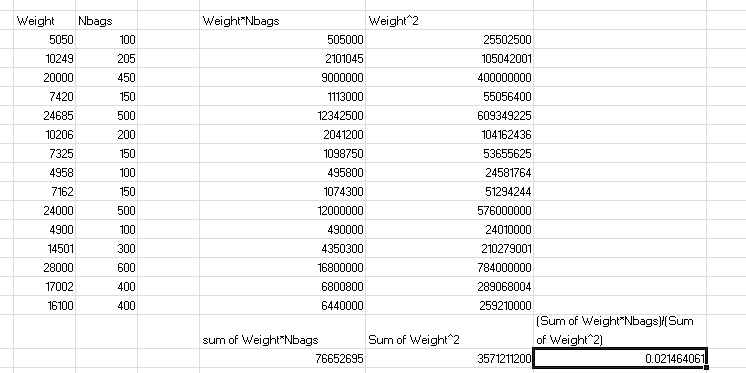
The residual plot looks to be fairly biased and homoscedastic, though it is difficult to tell, because nearly all of the values are on the lower half of the plot.



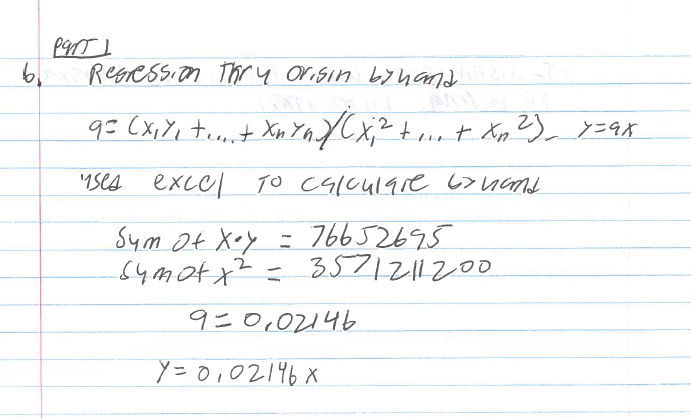
The normal plot appears to have thick tails.

1. **Compute the regression through the origin model by hand using the formula**

**a = (x1y1 + ... + xnyn) / (x12 + ... + xn2),     y = ax.**



Y = 0.02146061X



1. **Use SAS or R to find the regression through the origin model for predicting nbags from weight. Compare your hand calculation in Question B6 to the regression through the origin model obtained with SAS or R.**

> noint\_model <- lm(NBags ~ Weight + 0, data = flourFrame)

> print(summary(noint\_model))

Call:

lm(formula = NBags ~ Weight + 0, data = flourFrame)

Residuals:

Min 1Q Median 3Q Max

-29.840 -13.118 -7.224 -2.360 54.429

Coefficients:

Estimate Std. Error t value Pr(>|t|)

Weight 0.0214641 0.0003673 58.43 <2e-16 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

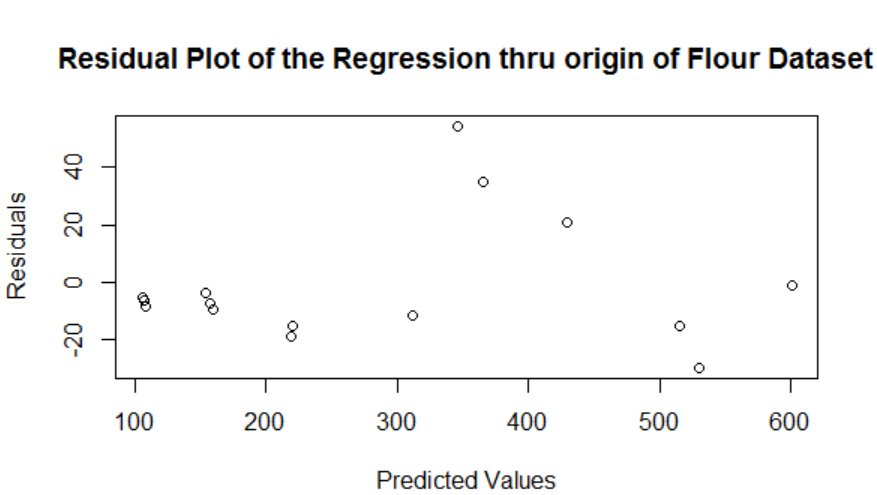
Residual standard error: 21.95 on 14 degrees of freedom

Multiple R-squared: 0.9959, Adjusted R-squared: 0.9956

F-statistic: 3414 on 1 and 14 DF, p-value: < 2.2e-16

The regression equation I obtained by hand and the model created by R are the same, as both have the x coefficient as 0.02164.

1. **For the regression through the origin model, create and interpret the residual plot and normal plot of the residuals.**



Once again, the residual plot appears to be fairly biased and homoscedastic.



The normal plot appears to have thick tails once again.

**Part B: Used Car Dataset (50 pts.)**

Collect the following used car data from the internet or elsewhere for at least 20 cars (or other vehicles like motorcycles or motorboats) of the same make and model: price, year, miles.

1. **Create and print the SAS dataset or R data frame called UsedCars.**

> usedCars <- read.table("c:/DataSets/UsedCars.txt", header = T)

> usedCarsFrame <- data.frame(year=usedCars$Year, miles=usedCars$Miles, price=usedCars$Price)

> print(usedCarsFrame)

year miles price

1 2013 34962 17990

2 2013 16249 16995

3 2012 36938 16295

4 2013 16174 15995

5 2011 41529 13995

6 2012 66088 13975

7 2013 43781 13972

8 2012 66309 12995

9 2013 53823 12994

10 2010 20658 12500

11 2011 60660 11995

12 2011 45920 11988

13 2012 36145 11646

14 2010 28800 11900

15 2011 63474 10995

16 2010 32245 10991

17 2012 113547 10990

18 2011 70406 10988

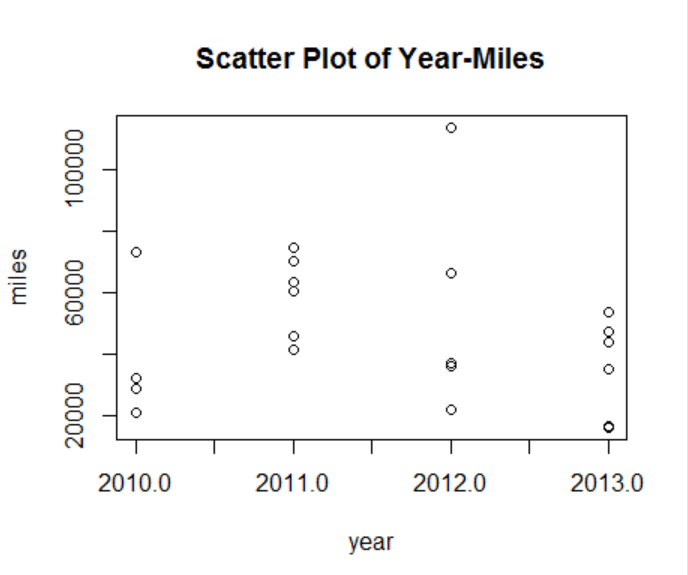
19 2011 74814 9995

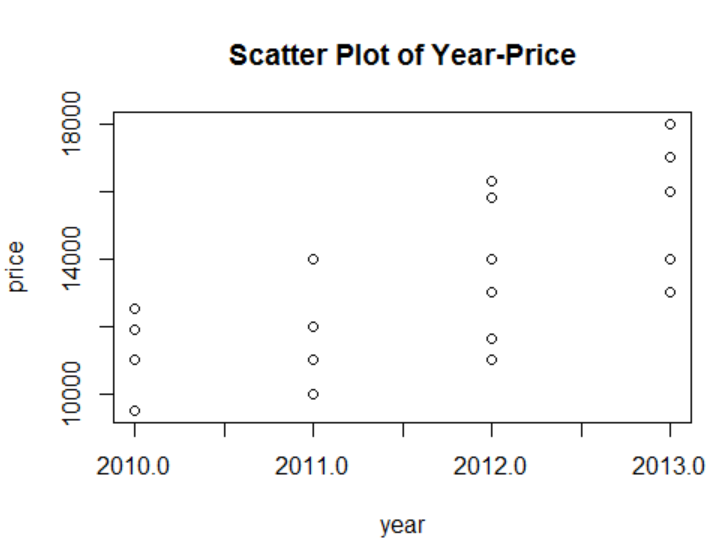
20 2010 73335 9499

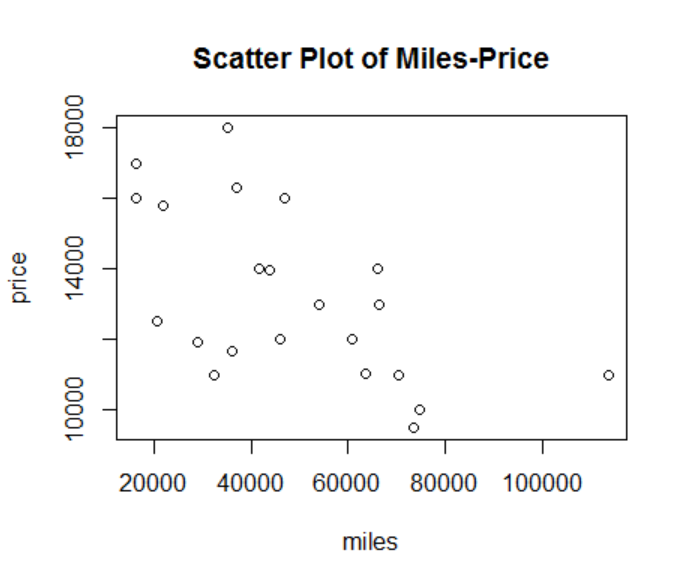
21 2013 46989 15988

22 2012 21907 15795

1. **Create the pairwise scatterplots of year, miles, and price.**



****

1. **Find the pairwise correlations of year, miles, and price with SAS or R. Interpret them.**

> cat("Correlation between year and miles: \n")

Correlation between year and miles:

> cor(year, miles)

[1] -0.1243684

> cat("Correlation between year and price: \n")

Correlation between year and price:

> cor(year, price)

[1] 0.7193466

> cat("Correlation between miles and price: \n")

Correlation between miles and price:

> cor(miles, price)

[1] -0.5823739

All of these correlations make sense to me:

In terms of a correlation between year and miles, it would make sense to be slightly negative – the newer a car (the higher the year is), the less likely it is for that car to have a lot of miles on it.

For the correlation between year and price, it is very positive, which makes sense too – the newer a car is, the higher the price should theoretically be.

Lastly, the correlation between miles and price being fairly negative makes sense too – the more miles on a car, the lower its price should be.

1. **Find the simple linear regression model price=year with SAS or R.**

> priceYrModel <- lm(price ~ year, data = usedCarsFrame)

> print(summary(priceYrModel))

Call:

lm(formula = price ~ year, data = usedCarsFrame)

Residuals:

Min 1Q Median 3Q Max

-2790.64 -1203.63 -2.01 1543.80 2622.11

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -3179773.7 689460.8 -4.612 0.000169 \*\*\*

year 1587.3 342.7 4.631 0.000161 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

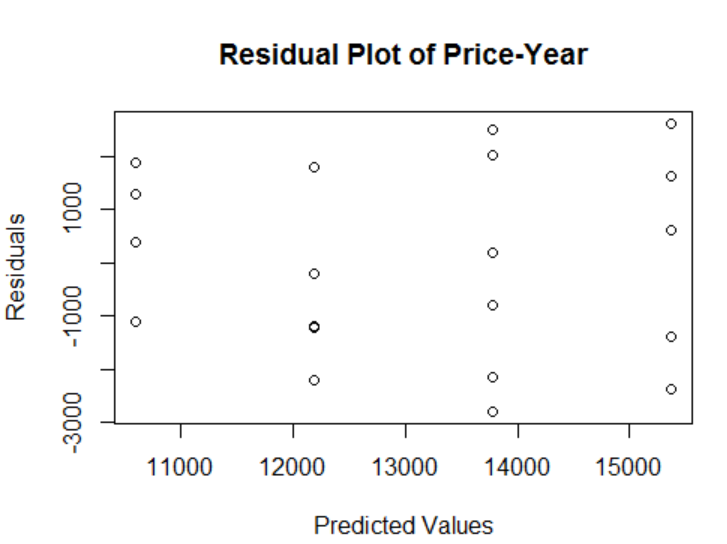
Residual standard error: 1717 on 20 degrees of freedom

Multiple R-squared: 0.5175, Adjusted R-squared: 0.4933

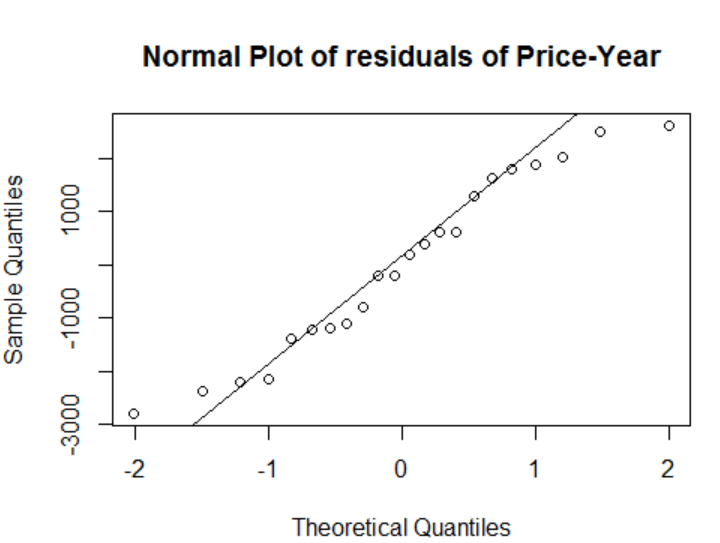
F-statistic: 21.45 on 1 and 20 DF, p-value: 0.0001612

Y = 1587.3x -3179773.7

1. **Create the residual plot residuals\*predicted and the normal plot of the residuals. Interpret these plots.**



This plot appears to be biased and homoscedastic, as the means of all the residuals in the smaller rectangles will not average out to 0.



This normal plot appears to be one that will have thin tails.

1. **Find the simple linear regression model price=miles with SAS or R.**

> priceMilesModel <- lm(price ~ miles, data = usedCarsFrame)

> print(summary(priceMilesModel))

Call:

lm(formula = price ~ miles, data = usedCarsFrame)

Residuals:

Min 1Q Median 3Q Max

-3168.9 -1573.7 248.3 1489.0 3990.9

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 1.607e+04 9.918e+02 16.202 5.76e-13 \*\*\*

miles -5.922e-02 1.848e-02 -3.204 0.00446 \*\*

---

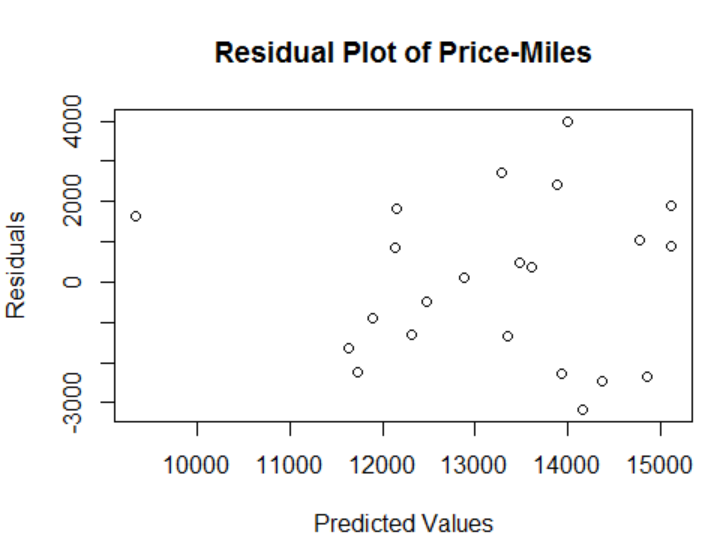
Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

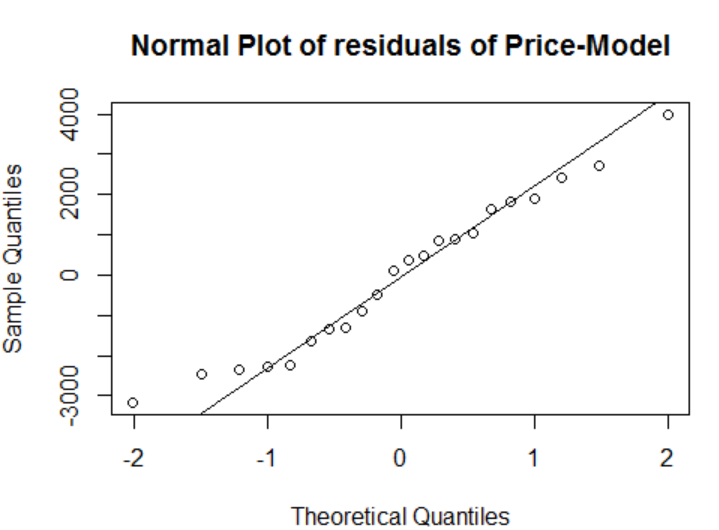
Residual standard error: 2009 on 20 degrees of freedom

Multiple R-squared: 0.3392, Adjusted R-squared: 0.3061

F-statistic: 10.26 on 1 and 20 DF, p-value: 0.004457

1. **Create the residual plot residuals\*predicted and normal plot of the residuals. Interpret these plots.**

  
 This residual plot appear to be very biased and heteroscedastic.



This normal plot also appears to have thin tails.

1. **Find the multiple linear regression model price=year miles with SAS or R.**

> multiModel <- lm(price ~ year + miles, data = usedCarsFrame)

> print(summary(multiModel))

Call:

lm(formula = price ~ year + miles, data = usedCarsFrame)

Residuals:

Min 1Q Median 3Q Max

-2708.4 -626.3 123.8 680.8 2125.5

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -2.901e+06 4.984e+05 -5.821 1.32e-05 \*\*\*

year 1.450e+03 2.477e+02 5.853 1.23e-05 \*\*\*

miles -5.091e-02 1.142e-02 -4.460 0.000269 \*\*\*

---

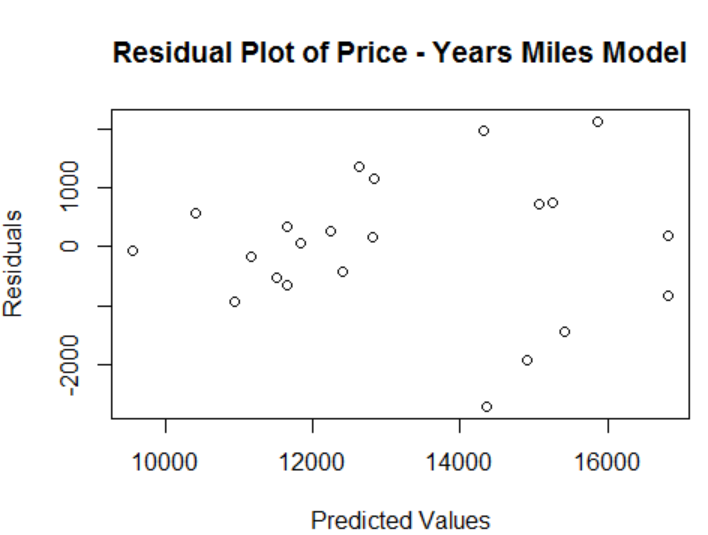
Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1231 on 19 degrees of freedom

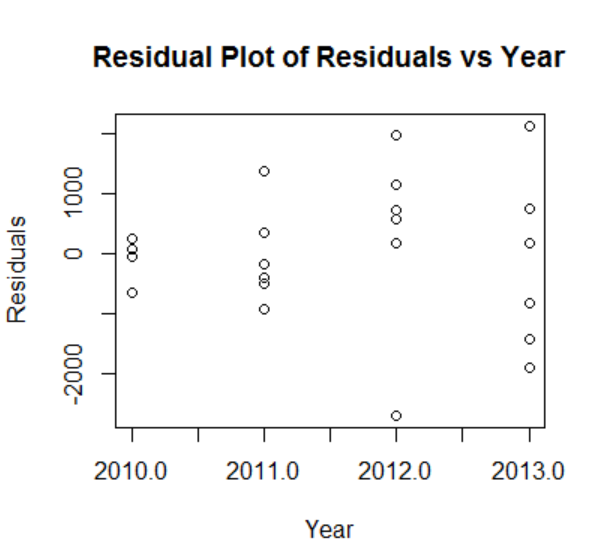
Multiple R-squared: 0.7642, Adjusted R-squared: 0.7394

F-statistic: 30.79 on 2 and 19 DF, p-value: 1.093e-06

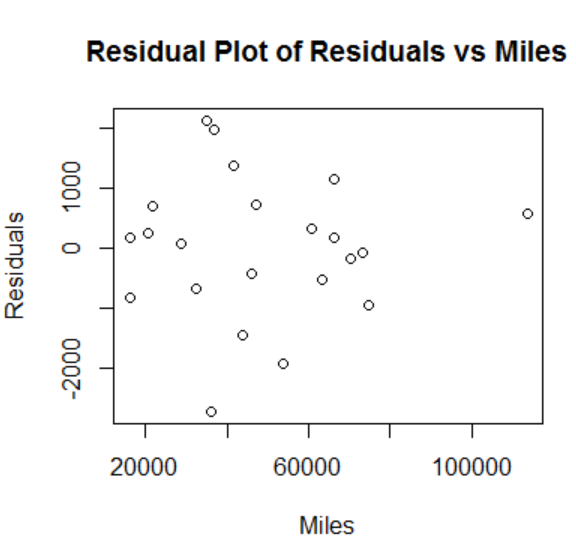
1. Create residual plots and the normal plot of the residuals.  Create these three residual plots: residuals\*predicted, residuals\*year, and residuals\*miles. Interpret these plots.



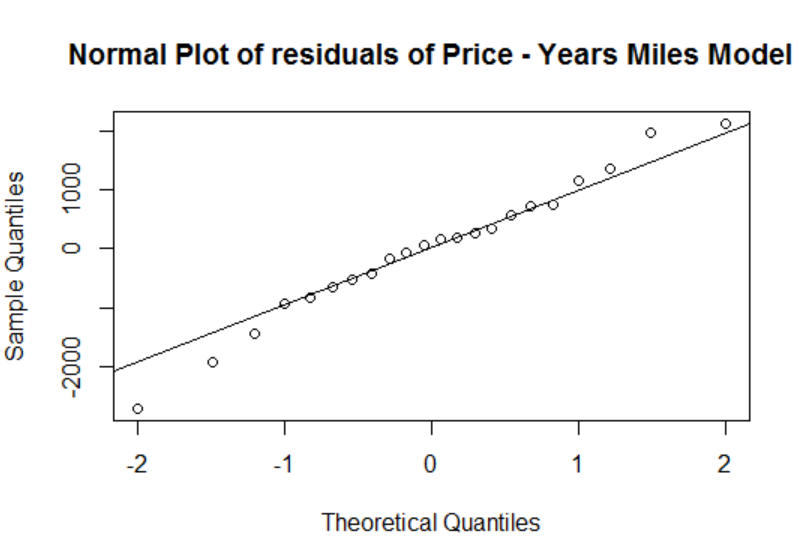
This residual plot appears to be fairly biased and heteroscedastic.



This residual plot appears to be biased, but homoscedastic.



This residual plot appears to be unbiased and heteroscedastic.

The normal plot of the residuals will appear to have thick tails.

1. In your opinion, which is the best regression model out of price=year, price=miles, price=year miles for predicting price.  Explain your answer.

In my opinion, the best regression model for predicting price out of all above would be the multiple regression model. This is because it has the highest multiple and adjusted R-squared values (0.7642 and 0.7394, respectively), compared to much lower values for the price-year and price-miles models. Though the residual plot and normal plot for this model wasn’t ideally unbiased and homoscedastic, it was closer to that than the residual plots for the other models.