Rachel Prokopius

Regression Analysis

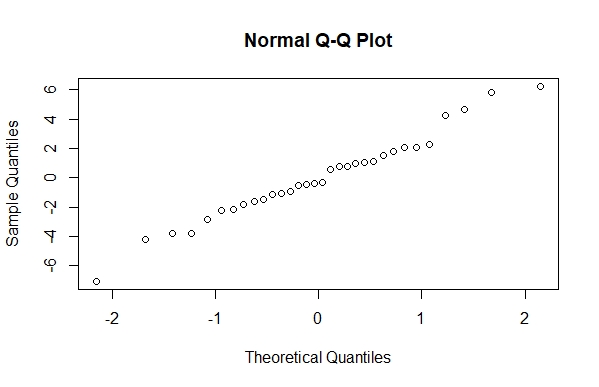
Homework 3

26 March 2020

**Question 4.4**

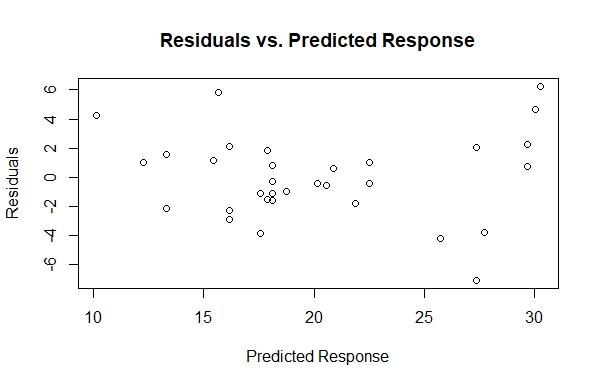
1. Figure 4.4a: The following is a normal probability plot of residuals for the following multiple regression model that relates gasoline mileage to engine displacement (x1)and the number of carburetor barrels the engine contains (x6):

**y= -0.05315 [miles/(cubic inches\*gallon)](x1) + 0.95922 [miles/(barrels\*gallon)](x6) + 32.88455 miles per gallon**



The normal probability plot of residuals, when the model follows the rules of normal distribution, independent distribution and equal variance, should have all the residuals fall on a straight line. As can be seen in the above plot, the residuals on the two ends of the normal probability plot do not fall on the relatively straight line that the middle portion of residuals falls on. This indicates that there may be outliers in the data set and that further analysis of the residuals must be conducted to determine if the potential outliers can be managed with transformation of the data set or, depending on the circumstances, removed from the data set entirely.

1. Figure 4.4b: The following is a plot of the residuals from the multiple linear regression from part a above versus the y-values the constructed model predicts.

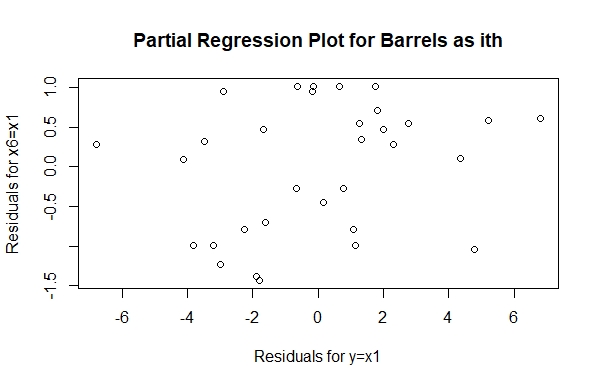


If the data set is normally distributed, independently distributed and has a constant variance, there will be no pattern in the data points in the constructed residuals vs. predicted response plot. The plot points seem to be fairly randomly distributed except for the rightmost points, which begin to have more-extreme residual values. This, similar to part a, indicates that there may be outliers in the data set and that further analysis of the residuals must be conducted to determine if the potential outliers can be managed with transformation of the data set or, depending on the circumstances, removed from the data set entirely.

1. Figure 4.4c: The following is a partial regression plot of residuals for the number of carburetor barrels (x6) explained by the engine displacement (x1) versus the residuals for the gas mileage (y) explained by the engine displacement (x1):

**y= -0.4736 [miles/(cubic inches\*gallon)](x1) + 33.72268 miles per gallon**

**(x6)= 0.006034 [miles/(cubic inches\*gallon)](x1) + 0.873755 miles per gallon**

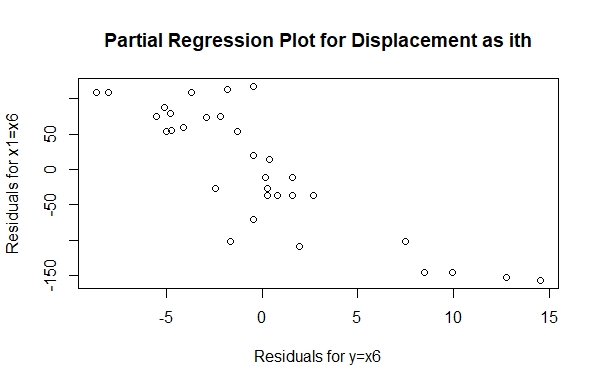


This partial regression plot treats the number of carburetor barrels (x6) as the ith term that is removed from the model, and then the model is re-run and its residuals are plotted against the residuals of the model with the ith term as the response variable. The plot helps to determine if the removal of the ith term (here, the number of carburetor barrels) greatly affects the outcome of the model. The fairly random pattern of the residuals plotted here suggests that the ith term doesn’t explain much additional variance after accounting for the tested regressors (here, engine displacement) and the ith term being removed does not have a substantial effect on the model and may not be a necessary regressor to include in the model.

Figure 4.4d: The following is a partial regression plot of residuals for the engine displacement (x1) explained by the number of carburetor barrels (x6) versus the residuals for the gas mileage (y) explained by the number of carburetor barrels (x6):

**y= -2.867 [miles/(barrels\*gallon)](x6) + 27.658 miles per gallon**

**(x1)= 71.98 [miles/(barrels\*gallon)](x6) + 98.34 miles per gallon**



This partial regression plot treats the engine displacement (x1) as the ith term that is removed from the model, and then the model is re-run and its residuals are plotted against the residuals of the model with the ith term as the response variable. The plot helps to determine if the removal of the ith term (here, the engine displacement) greatly affects the outcome of the model. The fairly distinct pattern of the residuals plotted here suggests that the ith term explains much of the model’s variance after accounting for the tested regressors (here, number of carburetor barrels) and the ith term being removed does have a substantial effect on the model and should remain in the model.

1. Table 4.4a: the following is a table containing the studentized residuals for the multiple regression model that relates gasoline mileage to engine displacement (x1)and the number of carburetor barrels the engine contains (x6).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| 0.272 | -0.390 | -0.198 | 0.731 | -0.640 | -0.747 | -0.138 |
| **8** | **9** | **10** | **11** | **12** | **13** | **14** |
| 0.201 | 1.667 | 0.266 | -0.564 | 2.234 | -1.445 | -0.138 |
| **15** | **16** | **17** | **18** | **19** | **20** | **21** |
| **-2.450** | **-0.325** | **1.519** | **0.543** | **-0.111** | **-0.512** | **0.344** |
| **22** | **23** | **24** | **25** | **26** | **27** | **28** |
| 2.028 | 0.392 | 0.799 | 0.707 | 0.350 | -1.323 | 0.627 |
| **29** | **30** | **31** | **32** |  |  |  |
| -0.782 | -1.002 | -1.328 | -0.379 |  |  |  |

Table 4.4b: the following is a table containing the R-student residuals for the multiple regression model that relates gasoline mileage to engine displacement (x1)and the number of carburetor barrels the engine contains (x6).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| 0.267 | -0.384 | -0.195 | 0.725 | -0.634 | -0.741 | -0.135 |
| **8** | **9** | **10** | **11** | **12** | **13** | **14** |
| 0.197 | 1.723 | 0.261 | -0.557 | 2.413 | -1.473 | -0.151 |
| **15** | **16** | **17** | **18** | **19** | **20** | **21** |
| -2.703 | -0.320 | 1.555 | 0.536 | -0.109 | -0.506 | 0.339 |
| **22** | **23** | **24** | **25** | **26** | **27** | **28** |
| 2.151 | 0.386 | 0.794 | 0.701 | 0.345 | -1.341 | 0.621 |
| **29** | **30** | **31** | **32** |  |  |  |
| -0.777 | -1.002 | -1.347 | -0.373 |  |  |  |

The studentized and R-student residuals are scaled residuals that can indicate if the data set is normally distributed and has a constant variance. If the absolute value of any of these residuals is greater than 3, it could indicate an outlier in the data set. An absolute value of a studentized residual or a R-student residual that is greater than 3 indicates the associated data point’s y-coordinate is likely an outlier. Based on the above data, none of the data points have a y-coordinate outlier because none of the scaled residuals has an absolute value greater than 3.

**R-code and output for Question 4.4**

> #Question 4.4

> #Choose the file needed

> B3=read.csv(file.choose())

> #Extract values from certain columns in the dataset

> B3y=B3[,1]

> B3displace=B3[,2]

> B3barrels=B3[,7]

> #Create the multiregression model

> multreg=lm(B3y~B3displace+B3barrels)

> multreg

Call:

lm(formula = B3y ~ B3displace + B3barrels)

Coefficients:

(Intercept) B3displace B3barrels

32.88455 -0.05315 0.95922

> #Get the unstandardized residuals of the model

> residmodel=resid(multreg)

> residmodel

1 2 3 4 5 6 7

0.7802408 -1.1197592 -0.5568569 2.1018346 -1.8155486 -2.1364691 -0.4058856

8 9 10 11 12 13 14

0.5916921 4.6643489 0.7470121 -1.6197592 6.2304992 -4.2147457 -0.4316755

15 16 17 18 19 20 21

-7.0623234 -0.9524012 4.2423910 1.5535309 -0.3197592 -1.4920385 1.0141144

22 23 24 25 26 27 28

5.8001636 1.1276242 2.2470121 2.0376766 0.9964843 -3.8024685 1.8279615

29 30 31 32

-2.2481654 -2.8781654 -3.8182825 -1.0882825

> #Create a normal probability plot of the unstandardized residuals

> qqnorm(residmodel)

> #Get predicted values from the multiregression model

> predictmultreg=predict(multreg)

> predictmultreg

1 2 3 4 5 6 7 8

18.11976 18.11976 20.55686 16.14817 21.88555 13.33647 22.52589 20.87831

9 10 11 12 13 14 15 16

30.03565 29.65299 18.11976 30.26950 25.71475 20.13168 27.36232 18.75240

17 18 19 20 21 22 23 24

10.14761 13.33647 18.11976 17.90204 22.52589 15.66984 15.46238 29.65299

25 26 27 28 29 30 31 32

27.36232 12.27352 27.70247 17.90204 16.14817 16.14817 17.58828 17.58828

> #Plot residuals vs the predicted response

> plot(predictmultreg,residmodel)

> #Plot residuals vs the predicted response

> plot(predictmultreg,residmodel, main= "Residuals vs. Predicted Response")

> #Plot residuals vs the predicted response

> plot(predictmultreg,residmodel, main= "Residuals vs. Predicted Response", x= "Predicted Response", y="Residuals")

> #Plot residuals vs the predicted response

> plot(predictmultreg,residmodel, main= "Residuals vs. Predicted Response", x="Predicted Response", y="Residuals")

> #Plot residuals vs the predicted response

> plot(predictmultreg,residmodel, main= "Residuals vs. Predicted Response", xlab ="Predicted Response", ylab="Residuals")

> #Plot partial regression models

> plot(resid1,residbarrelsith,main="Partial Regression Plot for Barrels as ith", xlab= "Residuals for y=x1", ylab= "Residuals for y=x6")

> lm.displacement=lm(B3y~B3displace)

> lm.displacement

Call:

lm(formula = B3y ~ B3displace)

Coefficients:

(Intercept) B3displace

33.72268 -0.04736

> #Get residuals for displacement model

> resid1=resid(lm.displacement)

> #Create partial regression for barrels as ith

> lm.barrelsith=lm(B3barrels~B3displace)

> lm.barrelsith

Call:

lm(formula = B3barrels ~ B3displace)

Coefficients:

(Intercept) B3displace

0.873755 0.006034

> #Get residuals for barrels as ith

> residbarrelsith=resid(lm.barrelsith)

> #Plot partial regression models

> plot(resid1,residbarrelsith,main="Partial Regression Plot for Barrels as ith", xlab= "Residuals for y=x1", ylab= "Residuals for y=x6")

> #Create partial regression for barrels

> #Plot partial regression models

> plot(resid1,residbarrelsith,main="Partial Regression Plot for Barrels as ith", xlab= "Residuals for y=x1", ylab= "Residuals for x6=x1")

> #Create partial regression for barrels

> lm.barrels=lm(B3y~B3barrels)

> lm.barrels

Call:

lm(formula = B3y ~ B3barrels)

Coefficients:

(Intercept) B3barrels

27.658 -2.867

> #Get residuals for displacement as ith

> resid2=resid(lm.barrels)

> #Create partial regression for displacement as ith

> lm.displaceith=lm(B3displace~B3barrels)

> lm.displaceith

Call:

lm(formula = B3displace ~ B3barrels)

Coefficients:

(Intercept) B3barrels

98.34 71.98

> #Get residuals for displacement as ith

> residdisplaceith=resid(lm.displaceith)

> #Plot partial regression models

> plot(resid2,residdisplaceith,main="Partial Regression Plot for Displacement as ith", "xlab= Residuals for y=x6", ylab= "Residuals for x1=x6")

> #Plot partial regression models

> plot(resid2,residdisplaceith,main="Partial Regression Plot for Displacement as ith", xlab= "Residuals for y=x6", ylab= "Residuals for x1=x6")

> #Studentized residuals (standardized)

> standr=rstandard(multreg)

> standr

1 2 3 4 5 6 7

0.2717882 -0.3900557 -0.1980105 0.7313729 -0.6404817 -0.7469016 -0.1376161

8 9 10 11 12 13 14

0.2007271 1.6672464 0.2656477 -0.5642251 2.2343937 -1.4445983 -0.1539895

15 16 17 18 19 20 21

-2.4499447 -0.3253417 1.5186561 0.5431086 -0.1113846 -0.5121263 0.3438369

22 23 24 25 26 27 28

2.0277182 0.3917877 0.7990680 0.7068771 0.3504291 -1.3231450 0.6274283

29 30 31 32

-0.7822915 -1.0015118 -1.3281597 -0.3785506

> #Student residuals (studentized)

> studentr=rstudent(multreg)

> studentr

1 2 3 4 5 6 7

0.2674019 -0.3842810 -0.1946983 0.7253733 -0.6338410 -0.7410735 -0.1352667

8 9 10 11 12 13 14

0.1973731 1.7229000 0.2613455 -0.5574801 2.4130447 -1.4734791 -0.1513731

15 16 17 18 19 20 21

-2.7032887 -0.3202682 1.5553718 0.5363974 -0.1094707 -0.5055101 0.3385474

22 23 24 25 26 27 28

2.1507428 0.3859964 0.7939593 0.7006450 0.3450656 -1.3412473 0.6207433

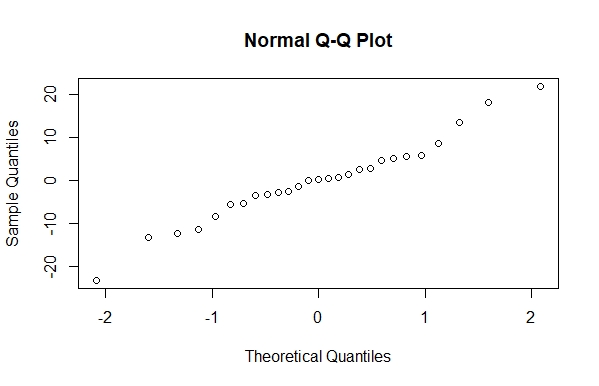
29 30 31 32

-0.7769267 -1.0015660 -1.3466593 -0.3728890

**Question 4.13**

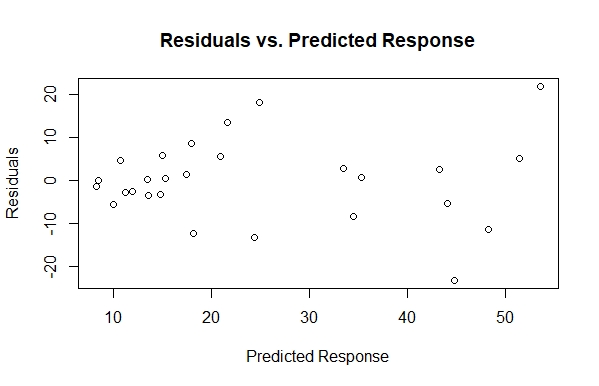
1. Figure 4.13a: The following is a normal probability plot of residuals for the following multiple regression model that relates carbon dioxide levels to total solvent (x6) and hydrogen consumption (x7):

**y= 0.01852 [CO2/(total solvent)](x6) + 2.18572 [CO2/(hydrogen consumption)](x7) + 2.52663 CO2**



The normal probability plot of residuals, when the model follows the rules of normal distribution, independent distribution and equal variance, should have all the residuals fall on a straight line. As can be seen in the above plot, the residuals on the two ends of the normal probability plot do not fall on the relatively straight line that the middle portion of residuals falls on. This indicates that there may be outliers in the data set and that further analysis of the residuals must be conducted to determine if the potential outliers can be managed with transformation of the data set or, depending on the circumstances, removed from the data set entirely.

1. Figure 4.13b: The following is a plot of the residuals from the multiple linear regression from part a above versus the y-values the constructed model predicts.

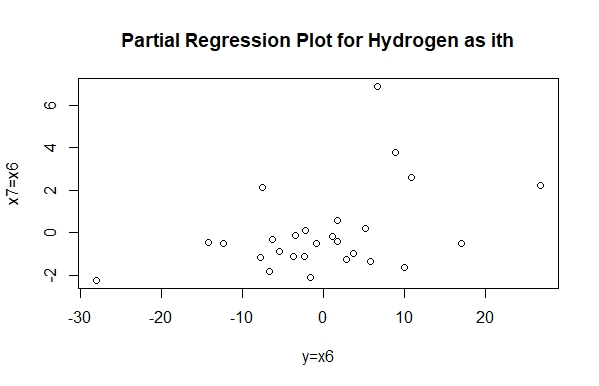


If the data set is normally distributed, independently distributed and has a constant variance, there will be no pattern in the data points in the constructed residuals vs. predicted response plot. The plot points seem to be fairly randomly distributed except for the leftmost points which are clustered around 0, and the rightmost points which begin to have more-extreme residual values. This, similar to part a, indicates that there may be outliers in the data set and that further analysis of the residuals must be conducted to determine if the potential outliers can be managed with transformation of the data set or, depending on the circumstances, removed from the data set entirely.

1. Figure 4.13c: The following is a partial regression plot of residuals for the solvent level (x6) explained by the hydrogen consumption (x7) versus the residuals for the levels of C02 (y) explained by the solvent level (x6):

**y= 0.01939 [CO2/(total solvent)](x6) + 6.14427 CO2**

**[CO2/(hydrogen consumption)](x7)= 0.0003993 [CO2/(total solvent)](x6) + 1.65513 CO2**

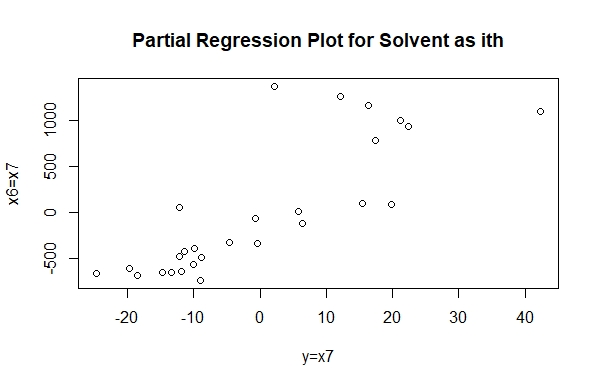


This partial regression plot treats the hydrogen consumption (x7) as the ith term that is removed from the model, and then the model is re-run and its residuals are plotted against the residuals of the model with the ith term as the response variable. The plot helps to determine if the removal of the ith term (here, the hydrogen consumption) greatly affects the outcome of the model. The fairly random pattern of the residuals plotted here suggests that the ith term doesn’t explain much additional variance after accounting for the tested regressors (here, total solvent) and the ith term being removed does not have a substantial effect on the model and may not be a necessary regressor to include in the model.

1. Figure 4.13d: The following is a partial regression plot of residuals for the hydrogen consumption (x7) explained by the solvent level (x6) versus the residuals for the levels of C02 (y) explained by the hydrogen consumption (x7):

**y= 3.113 [CO2/(total solvent)](x6) + 18.387 CO2**

**[CO2/(total solvent)](x6) = 50.06 [CO2/(hydrogen consumption)](x7) + 856.30 CO2**



This partial regression plot treats the total solvent (x6) as the ith term that is removed from the model, and then the model is re-run and its residuals are plotted against the residuals of the model with the ith term as the response variable. The plot helps to determine if the removal of the ith term (here, the total solvent) greatly affects the outcome of the model. The general positive correlation between the x and y variables plotted here suggests that the ith term explains much of the model’s variance after accounting for the tested regressors (here, hydrogen consumption) and the ith term being removed does have an effect on the model and should remain in the model.

1. Table 4.13a: the following is a table containing the studentized residuals for the multiple regression model that relates C02 levels to solvent level (x6)and hydrogen consumption (x7).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| -1.243 | 0.026 | -0.362 | 0.007 | 0.313 | 0.579 | 0.159 |
| **8** | **9** | **10** | **11** | **12** | **13** | **14** |
| -1.300 | 0.497 | 0.575 | 0.911 | 0.604 | -0.138 | 0.280 |
| **15** | **16** | **17** | **18** | **19** | **20** | **21** |
| 1.868 | 0.046 | -2.629 | 1.412 | -1.195 | -0.279 | -0.331 |
| **22** | **23** | **24** | **25** | **26** | **27** |  |
| -0.253 | -0.584 | -0.574 | -1.365 | 2.476 | 0.072 |  |

Table 4.13b: the following is a table containing the R-student residuals for the multiple regression model that relates C02 levels to solvent level (x6)and hydrogen consumption (x7).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| -1.258 | 0.026 | -0.355 | 0.007 | 0.308 | 0.571 | 0.156 |
| **8** | **9** | **10** | **11** | **12** | **13** | **14** |
| -1.320 | 0.489 | 0.567 | 0.908 | 0.596 | -0.135 | 0.274 |
| **15** | **16** | **17** | **18** | **19** | **20** | **21** |
| 1.978 | 0.045 | -3.050 | 1.444 | -1.207 | -0.273 | -0.324 |
| **22** | **23** | **24** | **25** | **26** | **27** |  |
| -0.248 | -0.576 | -0.565 | -1.391 | 2.809 | 0.070 |  |

The studentized and R-student residuals are scaled residuals that can indicate if the data set is normally distributed and has a constant variance. If the absolute value of any of these residuals is greater than 3, it could indicate an outlier in the data set. An absolute value of a studentized residual or a R-student residual that is greater than 3 indicates the associated data point’s y-coordinate is likely an outlier. Based on the above data, there is one data point where the R-student residual (which tends to be higher than the corresponding studentized residual) has an absolute value greater than 3, which has been colored red. This is likely one of the points on the end of the normal probability plot of residuals, Figure 4.13a, and indicates that review of the data set, the collection of that particular point, whether the data point was collected correctly and accurately, etc. could be warranted and if it is found to have been collected or recorded incorrectly, it may be removed from the data set.

1. The PRESS statistic for the full model, **y= 0.01852 [CO2/(total solvent)](x6) + 2.18572 [CO2/(hydrogen consumption)](x7) + 2.52663 CO2**, is:

**3388.589**

The PRESS statistic for the partial model with only total solvent as a regressor, **y= 3.113 [CO2/(total solvent)](x6) + 18.387 CO2**, is:

**3692.852**

A lower PRESS statistic is ideal because it indicates that the influential points in the data set are not drastically altering the fit of the model. With this in mind, the full model has a lower PRESS statistic and suggests that the influential points are not affecting the fit of the model as much as they are affecting the fit of the partial model with only total solvent. So, even though previous portions of this question have suggested that hydrogen consumption may not be necessary to explain much variance in the model, it may be helping to lower the influence of some of the potential outliers in the data set, as seen by the model with hydrogen consumption as a regressor having a lower PRESS statistic than the model without hydrogen consumption as a regressor.

**R-code and output for Question 4.13**

#Question 4.13

> #Choose the file needed

> B5=read.csv(file.choose())

> B5y=B5[,1]

> B5solvent=B5[,7]

> B5hydrogen=B5[,8]

> #Create the multiregression model

> multreg2=lm(B5y~B5solvent+B5hydrogen)

> multreg2

Call:

lm(formula = B5y ~ B5solvent + B5hydrogen)

Coefficients:

(Intercept) B5solvent B5hydrogen

2.52663 0.01852 2.18572

> residmodel2=resid(multreg2)

> residmodel2

1 2 3 4 5 6

-11.30225909 0.25116016 -3.47936258 0.07070517 2.92739440 5.59187264

7 8 9 10 11 12

1.54350850 -12.20504177 4.73491492 5.13287213 8.77783207 5.81538115

13 14 15 16 17 18

-1.30482106 2.61385273 18.16733530 0.44327305 -23.20361196 13.57531727

19 20 21 22 23 24

-8.40074999 -2.64381047 -3.18765539 -2.42493207 -5.56542292 -5.26343724

25 26 27

-13.27814078 21.96824651 0.64557933

> qqnorm(residmodel2)

> predictmultreg2=predict(multreg2)

> predictmultreg2

1 2 3 4 5 6 7

48.282259 13.488840 13.559363 8.459295 33.492606 20.998127 17.526491

8 9 10 11 12 13 14

18.165042 10.785085 51.477128 17.942168 14.984619 8.294821 43.316147

15 16 17 18 19 20 21

24.922665 15.346727 44.803612 21.614683 34.540750 11.243810 14.817655

22 23 24 25 26 27

12.014932 9.985423 44.153437 24.468141 53.651753 35.384421

> #Plot residuals vs the predicted response

> plot(predictmultreg2,residmodel2, main= "Residuals vs. Predicted Response", xlab ="Predicted Response", ylab="Residuals")

> lm.solvent=lm(B5y~B5solvent)

> lm.solvent

Call:

lm(formula = B5y ~ B5solvent)

Coefficients:

(Intercept) B5solvent

6.14427 0.01939

> #Get residuals for solvent

> residsolv=resid(lm.solvent)

> residsolv

1 2 3 4 5 6

-12.3611646 -0.8390371 -5.3968188 -2.4074844 -1.6458498 2.8432724

7 8 9 10 11 12

1.1327670 -7.5649341 5.2368935 10.8353306 5.8070350 3.6705571

13 14 15 16 17 18

-3.7523735 1.7431764 17.0876664 1.6686783 -28.0809601 10.0652775

19 20 21 22 23 24

6.6595291 -6.6264329 -3.4478694 -2.1868879 -6.2616680 -7.7527771

25 26 27

-14.2448441 26.8816234 8.9372952

> #Create partial regression for hydrogen as ith term

> lm.hydrogenith=lm(B5hydrogen~B5solvent)

> lm.hydrogenith

Call:

lm(formula = B5hydrogen ~ B5solvent)

Coefficients:

(Intercept) B5solvent

1.6551268 0.0003993

> #Get residuals for hydrogen as ith term

> residhydrogenith=resid(lm.hydrogenith)

> #Plot partial regression models

> plot(residsolv,residhydrogenith,main="Partial Regression Plot for Hydrogen as ith", xlab= "y=x6", ylab= "x7=x6")

> #Create partial regression for solvent as ith

> lm.hydrogen=lm(B5y~B5hydrogen)

> lm.hydrogen

Call:

lm(formula = B5y ~ B5hydrogen)

Coefficients:

(Intercept) B5hydrogen

18.387 3.113

> #Get residuals for hydrogen

> residhydr=resid(lm.hydrogen)

> #Create partial regression for solvent as ith

> lm.solventith=lm(B5solvent~B5hydrogen)

> lm.solventith

Call:

lm(formula = B5solvent ~ B5hydrogen)

Coefficients:

(Intercept) B5hydrogen

856.30 50.06

> #Get residuals for solvent as ith

> residsolventith=resid(lm.solventith)

> #Plot partial regression models

> plot(residhydr,residsolventith,main="Partial Regression Plot for Solvent as ith", xlab= "y=x7", ylab= "x6=x7")

> #Studentized residuals (standardized)

> standrmultreg2=rstandard(multreg2)

> standrmultreg2

1 2 3 4 5 6

-1.242735355 0.026102658 -0.361898040 0.007457631 0.313478720 0.578825298

7 8 9 10 11 12

0.159282072 -1.300235724 0.497199502 0.574963608 0.911147912 0.603683702

13 14 15 16 17 18

-0.137684822 0.279830896 1.868041787 0.046130670 -2.629102138 1.412244137

19 20 21 22 23 24

-1.195433348 -0.278794389 -0.330545550 -0.253459007 -0.583838529 -0.573686737

25 26 27

-1.364824899 2.475852736 0.071678399

> #Student residuals (studentized)

> studentrmultreg2=rstudent(multreg2)

> studentrmultreg2

1 2 3 4 5 6

-1.257708959 0.025553430 -0.355248919 0.007300619 0.307508628 0.570635154

7 8 9 10 11 12

0.156010872 -1.320207492 0.489257244 0.566774752 0.907802911 0.595511792

13 14 15 16 17 18

-0.134839133 0.274387049 1.978168260 0.045161391 -3.050197925 1.443801278

19 20 21 22 23 24

-1.206742116 -0.273367398 -0.324325007 -0.248455177 -0.575648346 -0.565498547

25 26 27

-1.391166115 2.808824399 0.070176723

> #PRESS Statistic for full model

> #Getting hat values

> PRESS(multreg2)

> #PRESS statistic for solvent model

> #Getting hat values

> PRESS(lm.solvent)

> #PRESS Statistic for full model

> library(PRESS)

> #PRESS Statistic for full model

> library(qpcR)

> #Getting hat values

> PRESS(multreg2)

.........10.........20.......

$stat

[1] 3388.589

$residuals

[1] -13.45848563 0.26719164 -3.70746381 0.07747379 3.30627549

[6] 5.90122104 1.61893314 -13.64297439 5.14225177 6.34343383

[11] 9.31525283 6.17226925 -1.43095284 2.95062730 18.91847336

[16] 0.47283797 -29.34022692 14.47017284 -16.75470630 -2.89562212

[21] -3.37594538 -2.60927605 -6.03241869 -6.15863288 -13.81722249

[26] 27.48263506 0.78384669

$P.square

[1] 0.5694357

>

> #PRESS statistic for solvent model

> #Getting hat values

> PRESS(lm.solvent)

.........10.........20.......

$stat

[1] 3692.852

$residuals

[1] -14.6799951 -0.8903291 -5.7056788 -2.6027357 -1.7745784 2.9532214

[7] 1.1876979 -8.0658600 5.6842868 12.3896219 6.0486567 3.8579419

[13] -4.0614491 1.9643909 17.7508384 1.7742594 -33.4823557 10.4527406

[19] 6.9554084 -7.0126154 -3.6510033 -2.3528473 -6.7799228 -8.9409996

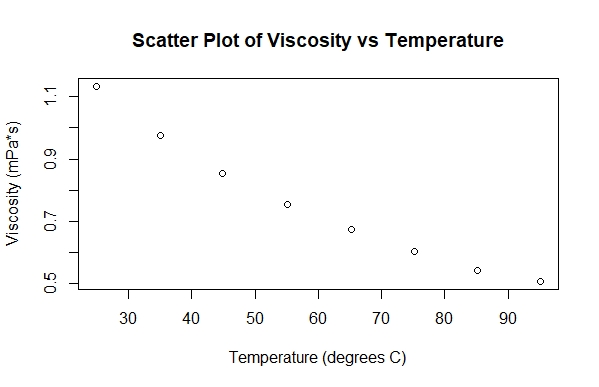
[25] -14.7942449 31.7040526 9.2917877

$P.square

[1] 0.5307751

**Question 5.1:**

1. Figure 5.1a: The following is a scatter plot of the viscosity of toluene-tetralin blends when the molar fraction of toluene is 0.4 (mPa\*s) versus temperature (degrees C):



The data set does not appear to be completely linear. It appears that a power function may be more suitable to fit the data set.

1. The straight-line model is:

**y= -0.008758[(degrees C)-1](x) + 1.281511 mPa\*s**

The summary statistics for the straight-line model are as follows:

Call:

lm(formula = Viscosity ~ Temp)

Residuals:

Min 1Q Median 3Q Max

-0.043955 -0.035863 -0.009305 0.019900 0.069559

Coefficients:

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

(Intercept) 1.2815107 0.0468683 27.34 1.58e-07 \*\*\*

Temp -0.0087578 0.0007284 -12.02 2.01e-05 \*\*\*

---

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

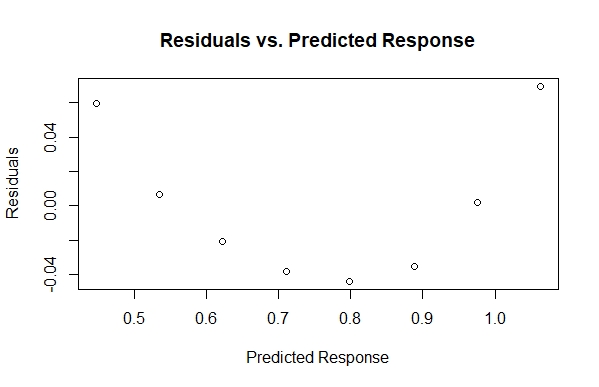
Residual standard error: 0.04743 on 6 degrees of freedom

Multiple R-squared: 0.9602, Adjusted R-squared: 0.9535

F-statistic: 144.6 on 1 and 6 DF, p-value: 2.007e-05

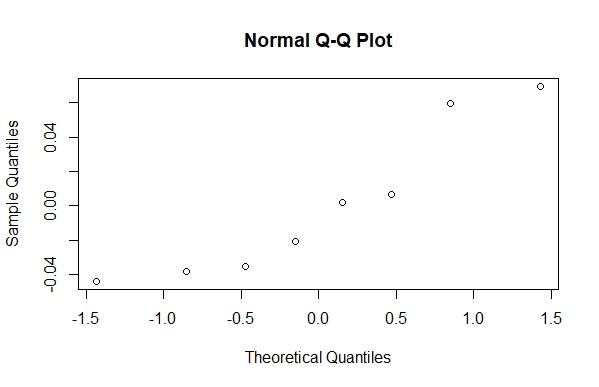
The summary statistics of the model indicate that there is a significant relationship between the two variables (p<0.05), and the R2 value is relatively high (0.9602).

Figure 5.1b: The following is a plot of the residuals from the straight-line model from part a above versus the y-values the constructed model predicts.



If the data set has a constant variance, there will be no pattern in the data points in the constructed residuals vs. predicted response plot. There is a clear pattern to Figure 5.1b, indicating that transformation of the data set is needed to attempt to normalize it so that the model adheres to the assumptions of normal distribution, independent distribution and constant variance.

Figure 5.1c: The following is a normal probability plot of residuals for the straight-line model that relates the viscosity of toluene-tetralin blends when the molar fraction of toluene is 0.4 (mPa\*s) to temperature (degrees C).

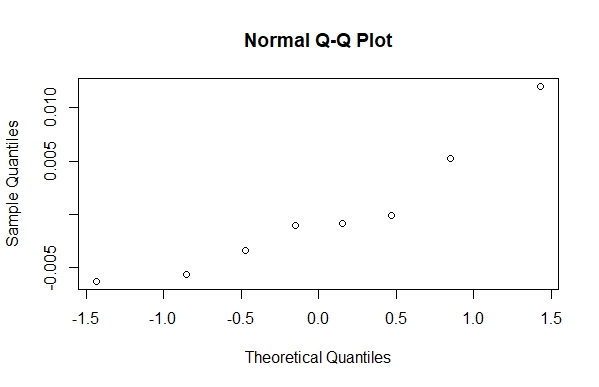


The normal probability plot of residuals, when the model follows the rules of normal distribution, independent distribution and equal variance, should have all the residuals fall on a straight line. As can be seen in the above plot, the residuals do not fall on a straight line. This indicates that there is a lack of normalcy in the data set.

Because the straight-line model has been found to predict and plot values that do not have a constant variance and are also not normally distributed, then it is likely that the data set would fit better with a model that transforms both the x and y variables. However, the transformation that best fit the model was a log transformation of just temperature.

Figure 5.1d: The following is a normal probability plot of residuals for the model that relates the viscosity of toluene-tetralin blends when the molar fraction of toluene is 0.4 (mPa\*s) to log-transformed temperature (degrees C).

**y= -0.4762[(degrees C)-1][log(x)] + 2.6651 mPa\*s**



The normal probability plot of residuals, when the model follows the rules of normal distribution, independent distribution and equal variance, should have all the residuals fall on a straight line. As can be seen in the above plot, the residuals fall on a slightly straighter line than the residuals for the straight-line plot illustrated in Figure 5.1c. Therefore, it suggests that the log-transformed temperature model may be a better model.

The summary statistics for the log-transformed temperature model are as follows:

Call:

lm(formula = Viscosity ~ logT)

Residuals:

Min 1Q Median 3Q Max

-0.0062609 -0.0039317 -0.0009493 0.0012340 0.0119894

Coefficients:

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

(Intercept) 2.66506 0.02154 123.72 1.88e-11 \*\*\*

logT -0.47622 0.00534 -89.17 1.34e-10 \*\*\*

---

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

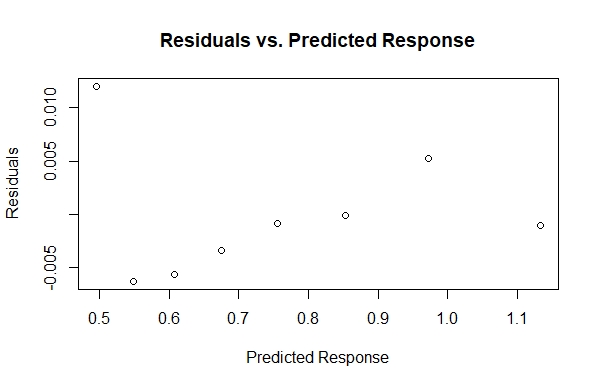
Residual standard error: 0.006525 on 6 degrees of freedom

Multiple R-squared: 0.9992, Adjusted R-squared: 0.9991

F-statistic: 7952 on 1 and 6 DF, p-value: 1.34e-10

The log-transformed temperature model has a slightly higher Rsquared value than the straight-line model, which suggests that this model is a better fit for the data set than the straight-line model.

Figure 5.1e: The following is a plot of the residuals from the log-transformed temperature model versus the y-values the constructed model predicts.



Though there is still a pattern in the residuals vs. predicted response graph, the pattern is less so than in Figure 5.1b with the straight-line model. This suggests that the new model has a more-constant variance than the straight-line model and may be a better fit for the data set.

The PRESS statistic for the straight-line model, **y= -0.008758[(degrees C)-1](x) + 1.281511 mPa\*s**, is:

**0.0317038**

The PRESS statistic for the log-transformed temperature model, **y= -0.4762[(degrees C)-1][log(x)] + 2.6651 mPa\*s**, is:

**0.0005062481**

A lower PRESS statistic is ideal because it indicates that the influential points in the data set are not drastically altering the fit of the model. With this in mind, the log-transformed temperature model has a much lower PRESS statistic and suggests that the influential points are not affecting the fit of the model as much as they are affecting the fit of the straight-line model.

**Rcode and output for 5.1**

> #Question 5.1

> #Choose the file needed

> Data=read.csv(file.choose())

> Temp=Data[,1]

> Viscosity=Data[,2]

> plot(Temp,Viscosity, main= "Scatter Plot of Viscosity vs Temperature", xlab= "Temperature (degrees C)", ylab= "Viscosity (mPa\*s)")

>

> #Fit a straight-line model

> lm.TV=lm(Viscosity~Temp)

> lm.TV

Call:

lm(formula = Viscosity ~ Temp)

Coefficients:

(Intercept) Temp

1.281511 -0.008758

> summary(lm.TV)

Call:

lm(formula = Viscosity ~ Temp)

Residuals:

Min 1Q Median 3Q Max

-0.043955 -0.035863 -0.009305 0.019900 0.069559

Coefficients:

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

(Intercept) 1.2815107 0.0468683 27.34 1.58e-07 \*\*\*

Temp -0.0087578 0.0007284 -12.02 2.01e-05 \*\*\*

---

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

Residual standard error: 0.04743 on 6 degrees of freedom

Multiple R-squared: 0.9602, Adjusted R-squared: 0.9535

F-statistic: 144.6 on 1 and 6 DF, p-value: 2.007e-05

>

> #Get the unstandardized residuals of the model

> residlm.TV=resid(lm.TV)

> residlm.TV

1 2 3 4 5 6

0.069559123 0.002213130 -0.035084428 -0.043954639 -0.038200633 -0.020822409

7 8

0.006655816 0.059634040

>

> #Get predicted values from the model

> predictlm.TV=predict(lm.TV)

> predictlm.TV

1 2 3 4 5 6 7

1.0634409 0.9749869 0.8882844 0.7989546 0.7105006 0.6229224 0.5353442

8

0.4477660

> plot(predictlm.TV,residlm.TV, main= "Residuals vs. Predicted Response", xlab ="Predicted Response", ylab="Residuals")

>

> #Create a normal probability plot of the unstandardized residuals

> qqnorm(residlm.TV)

>

> #Fit a transformed model

> NewV=1/Viscosity

> NewT=1/Temp

> new.TV=lm(NewV~NewT)

> new.TV

Call:

lm(formula = NewV ~ NewT)

Coefficients:

(Intercept) NewT

2.13 -35.44

> summary(new.TV)

Call:

lm(formula = NewV ~ NewT)

Residuals:

Min 1Q Median 3Q Max

-0.16839 -0.11460 -0.04574 0.14245 0.21332

Coefficients:

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

(Intercept) 2.1298 0.1381 15.418 4.71e-06 \*\*\*

NewT -35.4418 6.2444 -5.676 0.00129 \*\*

---

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

Residual standard error: 0.1668 on 6 degrees of freedom

Multiple R-squared: 0.843, Adjusted R-squared: 0.8168

F-statistic: 32.21 on 1 and 6 DF, p-value: 0.001288

>

> #Get the unstandardized residuals of the model

> residnew.TV=resid(new.TV)

> residnew.TV

1 2 3 4 5 6

0.17618223 -0.09384076 -0.16838769 -0.16206525 -0.09877829 0.00235886

7 8

0.13120669 0.21332420

>

> #Get predicted values from the model

> predictnew.TV=predict(new.TV)

> predictnew.TV

1 2 3 4 5 6 7

0.7064303 1.1171727 1.3404458 1.4865686 1.5862095 1.6584948 1.7138118

8

1.7575075

> plot(predictnew.TV,residnew.TV, main= "Residuals vs. Predicted Response", xlab ="Predicted Response", ylab="Residuals")

>

> #Create a normal probability plot of the unstandardized residuals

> qqnorm(residnew.TV)

>

> new2.TV=lm(Viscosity~NewT)

> new2.TV

Call:

lm(formula = Viscosity ~ NewT)

Coefficients:

(Intercept) NewT

0.3272 21.3980

> summary(new2.TV)

Call:

lm(formula = Viscosity ~ NewT)

Residuals:

Min 1Q Median 3Q Max

-0.053594 -0.038442 0.003594 0.038797 0.049393

Coefficients:

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

(Intercept) 0.32724 0.03711 8.819 0.000118 \*\*\*

NewT 21.39800 1.67739 12.757 1.42e-05 \*\*\*

---

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

Residual standard error: 0.04481 on 6 degrees of freedom

Multiple R-squared: 0.9644, Adjusted R-squared: 0.9585

F-statistic: 162.7 on 1 and 6 DF, p-value: 1.424e-05

>

> #Get the unstandardized residuals of the model

> residnew2.TV=resid(new2.TV)

> residnew2.TV

1 2 3 4 5 6

-0.053594387 0.038591638 0.049392923 0.039414630 0.016872909 -0.009684775

7 8

-0.036387134 -0.044605804

>

> #Get predicted values from the model

> predictnew2.TV=predict(new2.TV)

> predictnew2.TV

1 2 3 4 5 6 7

1.1865944 0.9386084 0.8038071 0.7155854 0.6554271 0.6117848 0.5783871

8

0.5520058

> plot(predictnew2.TV,residnew2.TV, main= "Residuals vs. Predicted Response", xlab ="Predicted Response", ylab="Residuals")

>

> #Create a normal probability plot of the unstandardized residuals

> qqnorm(residnew2.TV)

>

> #Fit a transformed model

> NegNewV=-(1/Viscosity)

> new3.TV=lm(NegNewV~Temp)

> new3.TV

Call:

lm(formula = NegNewV ~ Temp)

Coefficients:

(Intercept) Temp

-0.47015 -0.01582

> summary(new3.TV)

Call:

lm(formula = NegNewV ~ Temp)

Residuals:

Min 1Q Median 3Q Max

-0.026867 -0.005316 0.003054 0.009932 0.017416

Coefficients:

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

(Intercept) -0.4701470 0.0165118 -28.47 1.24e-07 \*\*\*

Temp -0.0158216 0.0002566 -61.66 1.22e-09 \*\*\*

---

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

Residual standard error: 0.01671 on 6 degrees of freedom

Multiple R-squared: 0.9984, Adjusted R-squared: 0.9982

F-statistic: 3802 on 1 and 6 DF, p-value: 1.223e-09

>

> #Get the unstandardized residuals of the model

> residnew3.TV=resid(new3.TV)

> residnew3.TV

1 2 3 4 5

-0.0185065524 0.0005726285 0.0084807509 0.0174163549 0.0142870765

6 7 8

-0.0009189438 -0.0268672631 0.0055359483

>

> #Get predicted values from the model

> predictnew3.TV=predict(new3.TV)

> predictnew3.TV

1 2 3 4 5 6 7

-0.864106 -1.023905 -1.180539 -1.341920 -1.501718 -1.659935 -1.818151

8

-1.976368

> plot(predictnew3.TV,residnew3.TV, main= "Residuals vs. Predicted Response", xlab ="Predicted Response", ylab="Residuals")

>

> #Create a normal probability plot of the unstandardized residuals

> qqnorm(residnew3.TV)

>

> plot(Temp,NegNewV, main= "Scatter Plot of Viscosity vs Temperature", xlab= "Temperature (degrees C)", ylab= "1/(Viscosity (mPa\*s))")

>

> V2=Viscosity^2

> T2=Temp^2

> logV=log(Viscosity)

> logT=log(Temp)

> expT=exp(Temp)

> #Fit a transformed model

>

> new4.TV=lm(Viscosity~logT)

> new4.TV

Call:

lm(formula = Viscosity ~ logT)

Coefficients:

(Intercept) logT

2.6651 -0.4762

> summary(new4.TV)

Call:

lm(formula = Viscosity ~ logT)

Residuals:

Min 1Q Median 3Q Max

-0.0062609 -0.0039317 -0.0009493 0.0012340 0.0119894

Coefficients:

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

(Intercept) 2.66506 0.02154 123.72 1.88e-11 \*\*\*

logT -0.47622 0.00534 -89.17 1.34e-10 \*\*\*

---

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

Residual standard error: 0.006525 on 6 degrees of freedom

Multiple R-squared: 0.9992, Adjusted R-squared: 0.9991

F-statistic: 7952 on 1 and 6 DF, p-value: 1.34e-10

> #Get the unstandardized residuals of the model

> residnew4.TV=resid(new4.TV)

> residnew4.TV

1 2 3 4 5

-0.0010760302 0.0052677267 -0.0001105788 -0.0008225182 -0.0033699250

6 7 8

-0.0056171500 -0.0062608953 0.0119893708

> #Get predicted values from the model

> predictnew4.TV=predict(new4.TV)

> predictnew4.TV

1 2 3 4 5 6 7

1.1340760 0.9719323 0.8533106 0.7558225 0.6756699 0.6077171 0.5482609

8

0.4954106

> plot(predictnew4.TV,residnew4.TV, main= "Residuals vs. Predicted Response", xlab ="Predicted Response", ylab="Residuals")

> #Create a normal probability plot of the unstandardized residuals

> qqnorm(residnew4.TV)

> #Create a normal probability plot of the unstandardized residuals

> qqnorm(residlm.TV)

> #PRESS Statistic for full model

> library(qpcR)

> PRESS(lm.TV)

........

$stat

[1] 0.0317038

$residuals

[1] 0.119300354 0.003045870 -0.042753901 -0.050572869 -0.043967543

[6] -0.025357640 0.009163949 0.102060305

$P.square

[1] 0.906421

> PRESS(new4.TV)

........

$stat

[1] 0.0005062481

$residuals

[1] -0.0023851069 0.0071539011 -0.0001306136 -0.0009400218 -0.0039355350

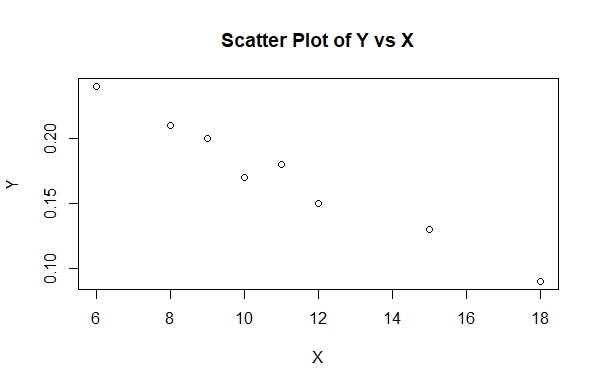
[6] -0.0069289021 -0.0083655791 0.0177482208

$P.square

[1] 0.9985057

**Question 5.4**

1. Figure 5.4a: The following is a scatter plot of the y-values vs. the x-values:



The data set does not appear to be completely linear, especially with the left cluster of points. It appears that a power function may be more suitable to fit the data set.

1. The straight-line model is:

**y= -0.01212(x) + 0.30610 mPa\*s**

The summary statistics for the straight-line model are as follows:

Call:

lm(formula = y ~ x)

Residuals:

Min 1Q Median 3Q Max

-0.014887 -0.002008 0.002539 0.005948 0.007235

Coefficients:

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

(Intercept) 0.3061025 0.0101963 30.02 9.06e-08 \*\*\*

x -0.0121216 0.0008715 -13.91 8.61e-06 \*\*\*

---

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

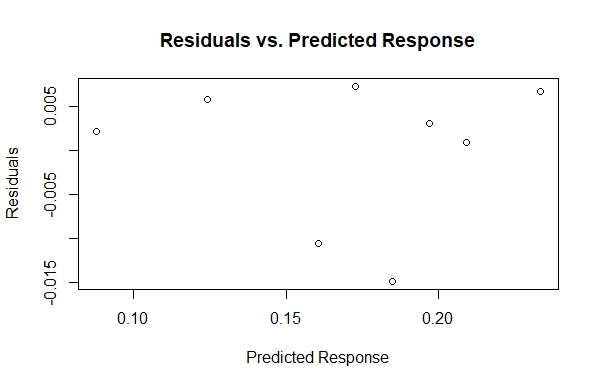
Residual standard error: 0.008925 on 6 degrees of freedom

Multiple R-squared: 0.9699, Adjusted R-squared: 0.9649

F-statistic: 193.4 on 1 and 6 DF, p-value: 8.606e-06

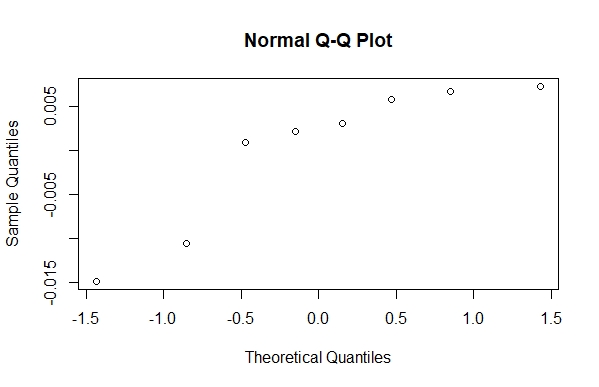
The summary statistics of the model indicate that there is a significant relationship between the two variables (p<0.05), and the R2 value is relatively high (0.9699).

Figure 5.4b: The following is a plot of the residuals from the straight-line model from part a above versus the y-values the constructed model predicts.



If the data set has a constant variance, there will be no pattern in the data points in the constructed residuals vs. predicted response plot. There isn’t necessarily a distinct pattern to Figure 5.4b, but it isn’t completely random either.

Figure 5.4c: The following is a normal probability plot of residuals for the model that relates y versus x.



The normal probability plot of residuals, when the model follows the rules of normal distribution, independent distribution and equal variance, should have all the residuals fall on a straight line. As can be seen in the above plot, the residuals do not fall on a straight line. This indicates that there is a lack of normalcy in the data set.

Because the straight-line model has been found to predict and plot values that do not have a constant variance and are also not normally distributed, then it is likely that the data set would fit better with a model that transforms both the x and y variables. However, the best tested model was log-transformed y versus x.

The log-transformed y versus x model is as follows:

**log(y)= -0.07916(x) - 0.92303**

The summary statistics for the straight-line model are as follows:

Call:

lm(formula = logy ~ x)

Residuals:

Min 1Q Median 3Q Max

-0.06008 -0.03619 -0.01428 0.03705 0.07896

Coefficients:

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

(Intercept) -0.923032 0.066268 -13.93 8.53e-06 \*\*\*

x -0.079157 0.005664 -13.97 8.37e-06 \*\*\*

---

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

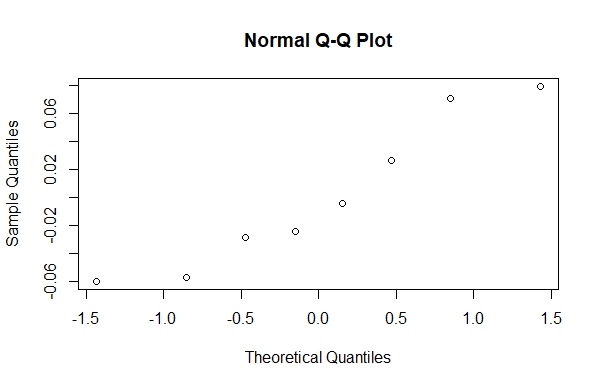
Residual standard error: 0.05801 on 6 degrees of freedom

Multiple R-squared: 0.9702, Adjusted R-squared: 0.9652

F-statistic: 195.3 on 1 and 6 DF, p-value: 8.37e-06

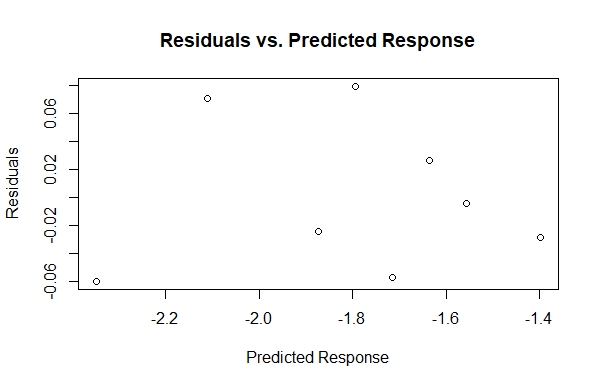
The summary statistics of the model indicate that there is a significant relationship between the two variables (p<0.05), and the R2 value is relatively high (0.9702).

Figure 5.4d: The following is a normal probability plot of residuals for the model that relates the log-transformed y versus x.



The normal probability plot of residuals, when the model follows the rules of normal distribution, independent distribution and equal variance, should have all the residuals fall on a straight line. As can be seen in the above plot, the residuals fall on a slightly straighter line than the residuals for the straight-line plot illustrated in Figure 5.4c. Therefore, it suggests that the logy-transformed model may be a better model.

Figure 5.1e: The following is a plot of the residuals from the logy-transformed model versus the y-values the constructed model predicts.



Though there is still a pattern in the residuals vs. predicted response graph, the pattern is less so than in Figure 5.4b with the straight-line model. This suggests that the new model has a more-constant variance than the straight-line model and may be a better fit for the data set.

The PRESS statistic for the straight-line model, **y= -0.01212(x) + 0.30610 mPa\*s**, is:

**0.0007285009**

The PRESS statistic for the logy-transformed model, **log(y)= -0.07916(x) - 0.92303**, is:

**0.04577235**

A lower PRESS statistic is ideal because it indicates that the influential points in the data set are not drastically altering the fit of the model. With this in mind, the logy-transformed model has a much higher PRESS statistic and suggests that the influential points are affecting the fit of this model more than the straight-line model.

Taking into account that the transformations of the model were marginally better looking at the plots, the Rsquared value of the transformed model was only a little higher than that of the straight-line model, and the PRESS statistic for the transformed model was higher than the PRESS statistic for the straight-line mode, it may be that the straight-line model is the best fit for the data and it does not require transformation. Or, perhaps it requires more in-depth transformations than are attempted here. Based on the transformations attempted here (the others can be seen in the Rcode, but the Rsquared values dropped so much that they were not further analyzed), the straight-line model is the best fit for the data set.

**Rcode and output for 5.4**

View(Data)

> x=Data[,3]

> y=Data[,4]

> plot(x,y, main= "Scatter Plot of Y vs X", xlab= "X", ylab= "Y")

#Fit a straight-line model

> lm.xy=lm(y~x)

> lm.xy

Call:

lm(formula = y ~ x)

Coefficients:

(Intercept) x

0.30610 -0.01212

> summary(lm.xy)

Call:

lm(formula = y ~ x)

Residuals:

Min 1Q Median 3Q Max

-0.014887 -0.002008 0.002539 0.005948 0.007235

Coefficients:

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

(Intercept) 0.3061025 0.0101963 30.02 9.06e-08 \*\*\*

x -0.0121216 0.0008715 -13.91 8.61e-06 \*\*\*

---

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

Residual standard error: 0.008925 on 6 degrees of freedom

Multiple R-squared: 0.9699, Adjusted R-squared: 0.9649

F-statistic: 193.4 on 1 and 6 DF, p-value: 8.606e-06

> #Get the unstandardized residuals of the model

> residlm.xy=resid(lm.xy)

> residlm.xy

1 2 3 4 5

-0.0148867700 0.0057210965 0.0020858164 -0.0106436234 0.0029916567

6 7 8

0.0008700834 0.0072348033 0.0066269368

> #Get predicted values from the model

> predictlm.xy=predict(lm.xy)

> predictlm.xy

1 2 3 4 5 6 7

0.18488677 0.12427890 0.08791418 0.16064362 0.19700834 0.20912992 0.17276520

8

0.23337306

> plot(predictlm.xy,residlm.xy, main= "Residuals vs. Predicted Response", xlab ="Predicted Response", ylab="Residuals")

> #Create a normal probability plot of the unstandardized residuals

> qqnorm(residlm.xy)

> #Fit a transformed model

> Newx=1/x

> Newy=1/y

> new.xy=lm(Newy~Newx)

> new.xy

Call:

lm(formula = Newy ~ Newx)

Coefficients:

(Intercept) Newx

11.61 -52.62

> summary(new.TV)

Call:

lm(formula = NewV ~ NewT)

Residuals:

Min 1Q Median 3Q Max

-0.16839 -0.11460 -0.04574 0.14245 0.21332

Coefficients:

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

(Intercept) 2.1298 0.1381 15.418 4.71e-06 \*\*\*

NewT -35.4418 6.2444 -5.676 0.00129 \*\*

---

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

Residual standard error: 0.1668 on 6 degrees of freedom

Multiple R-squared: 0.843, Adjusted R-squared: 0.8168

F-statistic: 32.21 on 1 and 6 DF, p-value: 0.001288

> #Get the unstandardized residuals of the model

> residnew.xy=resid(new.xy)

> residnew.xy

1 2 3 4 5 6 7

-0.4672344 -0.4114302 2.4226563 -0.5599959 -0.7648704 -0.2720696 -1.2724364

8

1.3253806

> #Get predicted values from the model

> predictnew.xy=predict(new.xy)

> predictnew.xy

1 2 3 4 5 6 7 8

6.349587 8.103738 8.688455 7.226663 5.764870 5.033974 6.827992 2.841286

> plot(predictnew.xy,residnew.xy, main= "Residuals vs. Predicted Response", xlab ="Predicted Response", ylab="Residuals")

> #Create a normal probability plot of the unstandardized residuals

> qqnorm(residnew.xy)

> #Fit a transformed model

> logy=log(y)

> new2.xy=lm(logy~Newx)

> new2.xy

Call:

lm(formula = logy ~ Newx)

Coefficients:

(Intercept) Newx

-2.600 7.967

> summary(new2.xy)

Call:

lm(formula = logy ~ Newx)

Residuals:

Min 1Q Median 3Q Max

-0.25098 -0.01766 0.03475 0.05855 0.16053

Coefficients:

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

(Intercept) -2.5996 0.1659 -15.67 4.28e-06 \*\*\*

Newx 7.9666 1.5775 5.05 0.00233 \*\*

---

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

Residual standard error: 0.1466 on 6 degrees of freedom

Multiple R-squared: 0.8096, Adjusted R-squared: 0.7778

F-statistic: 25.51 on 1 and 6 DF, p-value: 0.002333

> #Get the unstandardized residuals of the model

> residnew2.xy=resid(new2.xy)

> residnew2.xy

1 2 3 4 5 6

0.03094421 0.02823266 -0.25097464 0.03855729 0.10494566 0.04308898

7 8

0.16052602 -0.15532017

> #Get predicted values from the model

> predictnew2.xy=predict(new2.xy)

> predictnew2.xy

1 2 3 4 5 6 7

-1.802901 -2.068453 -2.156971 -1.935677 -1.714384 -1.603737 -1.875324

8

-1.271796

> plot(predictnew2.xy,residnew2.xy, main= "Residuals vs. Predicted Response", xlab ="Predicted Response", ylab="Residuals")

> #Create a normal probability plot of the unstandardized residuals

> qqnorm(residnew2.xy)

> #Fit a transformed model

> new3.xy=lm(logy~x)

> new3.xy

Call:

lm(formula = logy ~ x)

Coefficients:

(Intercept) x

-0.92303 -0.07916

> summary(new3.xy)

Call:

lm(formula = logy ~ x)

Residuals:

Min 1Q Median 3Q Max

-0.06008 -0.03619 -0.01428 0.03705 0.07896

Coefficients:

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

(Intercept) -0.923032 0.066268 -13.93 8.53e-06 \*\*\*

x -0.079157 0.005664 -13.97 8.37e-06 \*\*\*

---

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

Residual standard error: 0.05801 on 6 degrees of freedom

Multiple R-squared: 0.9702, Adjusted R-squared: 0.9652

F-statistic: 195.3 on 1 and 6 DF, p-value: 8.37e-06

> #Get the unstandardized residuals of the model

> residnew3.xy=resid(new3.xy)

> residnew3.xy

1 2 3 4 5 6

-0.057353208 0.070168713 -0.060084523 -0.024201988 0.026008541 -0.004358477

7 8

0.078962388 -0.029141447

> #Get predicted values from the model

> predictnew3.xy=predict(new3.xy)

> predictnew3.xy

1 2 3 4 5 6 7

-1.714604 -2.110390 -2.347861 -1.872918 -1.635446 -1.556289 -1.793761

8

-1.397975

> plot(predictnew3.xy,residnew3.xy, main= "Residuals vs. Predicted Response", xlab ="Predicted Response", ylab="Residuals")

> #Create a normal probability plot of the unstandardized residuals

> qqnorm(residnew3.xy)

> New2y=1/y

> New2x=1/x

> new4.xy=lm(New2y~x)

> new4.xy

Call:

lm(formula = New2y ~ x)

Coefficients:

(Intercept) x

0.2171 0.5517

> summary(new4.xy)

Call:

lm(formula = New2y ~ x)

Residuals:

Min 1Q Median 3Q Max

-0.80003 -0.31920 -0.01964 0.27119 0.96372

Coefficients:

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

(Intercept) 0.21710 0.75381 0.288 0.783032

x 0.55168 0.06443 8.562 0.000139 \*\*\*

---

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

Residual standard error: 0.6598 on 6 degrees of freedom

Multiple R-squared: 0.9243, Adjusted R-squared: 0.9117

F-statistic: 73.31 on 1 and 6 DF, p-value: 0.0001393

> #Get the unstandardized residuals of the model

> residnew4.xy=resid(new4.xy)

> residnew4.xy

1 2 3 4 5 6 7

0.1484251 -0.8000327 0.9637231 -0.1706262 -0.1822453 0.1313420 -0.7300548

8

0.6394689

> #Get predicted values from the model

> predictnew4.xy=predict(new4.xy)

> predictnew4.xy

1 2 3 4 5 6 7

5.733928 8.492340 10.147388 6.837293 5.182245 4.630563 6.285610

8

3.527198

> plot(predictnew4.xy,residnew4.xy, main= "Residuals vs. Predicted Response", xlab ="Predicted Response", ylab="Residuals")

> #Create a normal probability plot of the unstandardized residuals

> qqnorm(residnew4.xy)

> new5.xy=lm(y~New2x)

> new5.xy

Call:

lm(formula = y ~ New2x)

Coefficients:

(Intercept) New2x

0.04252 1.28850

> summary(new5.xy)

Call:

lm(formula = y ~ New2x)

Residuals:

Min 1Q Median 3Q Max

-0.0241051 -0.0053471 0.0008406 0.0083893 0.0203417

Coefficients:

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

(Intercept) 0.04252 0.01815 2.343 0.057605 .

New2x 1.28850 0.17256 7.467 0.000298 \*\*\*

---

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

Residual standard error: 0.01604 on 6 degrees of freedom

Multiple R-squared: 0.9028, Adjusted R-squared: 0.8867

F-statistic: 55.76 on 1 and 6 DF, p-value: 0.0002977

> #Get the unstandardized residuals of the model

> residnew5.xy=resid(new5.xy)

> residnew5.xy

1 2 3 4 5

-0.0013720175 0.0015781491 -0.0241051287 0.0001030658 0.0143112603

6 7 8

0.0064153575 0.0203416643 -0.0172723507

> #Get predicted values from the model

> predictnew5.xy=predict(new5.xy)

> predictnew5.xy

1 2 3 4 5 6 7

0.1713720 0.1284219 0.1141051 0.1498969 0.1856887 0.2035846 0.1596583

8

0.2572724

> plot(predictnew5.xy,residnew5.xy, main= "Residuals vs. Predicted Response", xlab ="Predicted Response", ylab="Residuals")

> #Create a normal probability plot of the unstandardized residuals

> qqnorm(residnew5.xy)

> new6.xy=lm(New2y~New2x)

> new6.xy

Call:

lm(formula = New2y ~ New2x)

Coefficients:

(Intercept) New2x

11.61 -52.62

> summary(new6.xy)

Call:

lm(formula = New2y ~ New2x)

Residuals:

Min 1Q Median 3Q Max

-1.2724 -0.6112 -0.4393 0.1273 2.4227

Coefficients:

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

(Intercept) 11.612 1.504 7.719 0.000248 \*\*\*

New2x -52.625 14.303 -3.679 0.010341 \*

---

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

Residual standard error: 1.329 on 6 degrees of freedom

Multiple R-squared: 0.6929, Adjusted R-squared: 0.6417

F-statistic: 13.54 on 1 and 6 DF, p-value: 0.01034

> #Get the unstandardized residuals of the model

> residnew6.xy=resid(new6.xy)

> residnew6.xy

1 2 3 4 5 6 7

-0.4672344 -0.4114302 2.4226563 -0.5599959 -0.7648704 -0.2720696 -1.2724364

8

1.3253806

> #Get predicted values from the model

> predictnew6.xy=predict(new6.xy)

> predictnew6.xy

1 2 3 4 5 6 7 8

6.349587 8.103738 8.688455 7.226663 5.764870 5.033974 6.827992 2.841286

> plot(predictnew6.xy,residnew6.xy, main= "Residuals vs. Predicted Response", xlab ="Predicted Response", ylab="Residuals")

> #Create a normal probability plot of the unstandardized residuals

> qqnorm(residnew6.xy)

> #Get the unstandardized residuals of the model

> residnew3.xy=resid(new3.xy)

> residnew3.xy

1 2 3 4 5 6

-0.057353208 0.070168713 -0.060084523 -0.024201988 0.026008541 -0.004358477

7 8

0.078962388 -0.029141447

> #Get predicted values from the model

> predictnew3.xy=predict(new3.xy)

> predictnew3.xy

1 2 3 4 5 6 7

-1.714604 -2.110390 -2.347861 -1.872918 -1.635446 -1.556289 -1.793761

8

-1.397975

> plot(predictnew3.xy,residnew3.xy, main= "Residuals vs. Predicted Response", xlab ="Predicted Response", ylab="Residuals")

> #Create a normal probability plot of the unstandardized residuals

> qqnorm(residnew3.xy)

> #PRESS Statistic for full model

> library(qpcR)

> PRESS(lm.xy)

........

$stat

[1] 0.0007285009

$residuals

[1] -0.017251381 0.007817590 0.004915730 -0.012266484 0.003595989

[6] 0.001112805 0.008269755 0.010610687

$P.square

[1] 0.9541463

> PRESS(new3.xy)

........

$stat

[1] 0.04577235

$residuals

[1] -0.066463178 0.095882004 -0.141603692 -0.027892126 0.031262415

[6] -0.005574332 0.090258097 -0.046659683

$P.square

[1] 0.9324214