

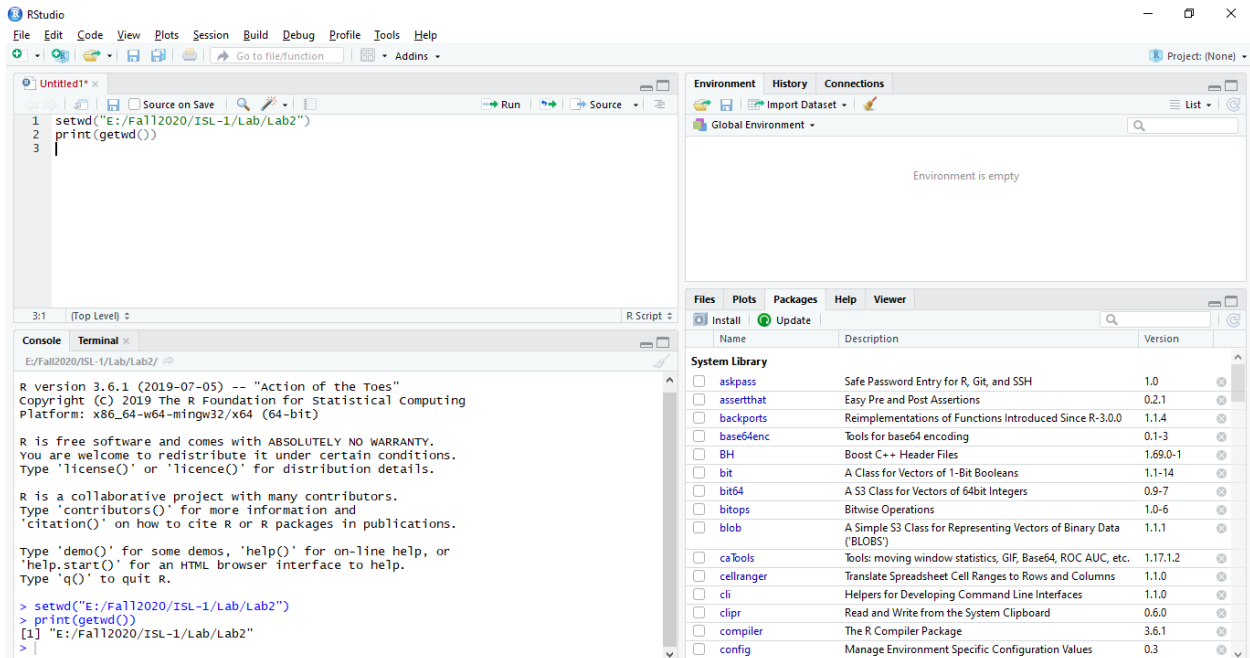
Introduction to Statistical Learning- Lab2

Name: Sandeep Reddy Salkuti

Student id: 16296868

Email: sswf7@umsystem.edu

1. View the video at the following URL and install R
<https://www.youtube.com/watch?v=5ONFqIk3RFg> You may download the R Code for Labs and the Data Sets to use from the textbook website. <http://www-bcf.usc.edu/~gareth/ISL/>



```
1 setwd("E:/Fall2020/ISL-1/Lab/Lab2")
2 print(getwd())
3
```

R version 3.6.1 (2019-07-05) -- "Action of the Toes"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

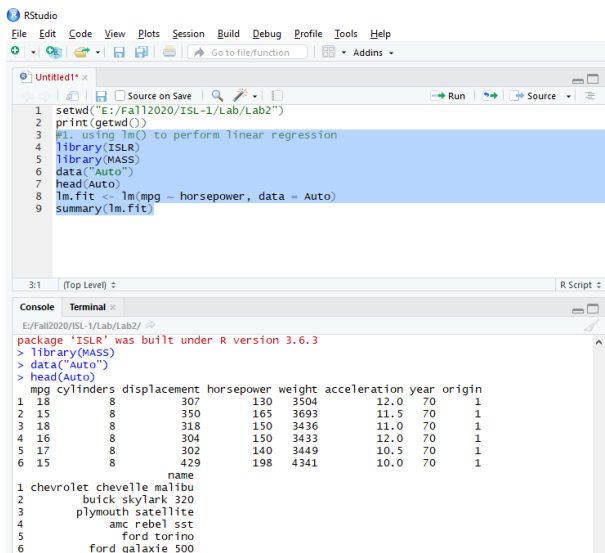
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

```
> setwd("E:/Fall2020/ISL-1/Lab/Lab2")
> print(getwd())
[1] "E:/Fall2020/ISL-1/Lab/Lab2"
```

2. a) Use the `lm()` function to perform a simple linear regression with `mpg` as the response and `horsepower` as the predictor. Use the `summary()` function to print the results



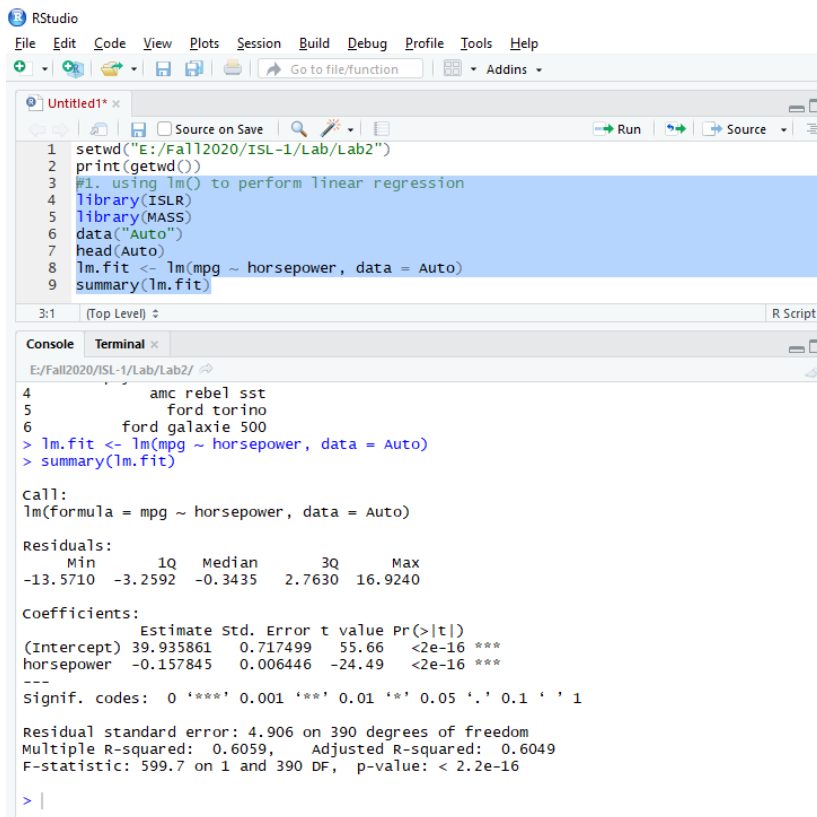
```
1 setwd("E:/Fall2020/ISL-1/Lab/Lab2")
2 print(getwd())
3 #1. using lm() to perform linear regression
4 library(ISLR)
5 library(MASS)
6 data("Auto")
7 head(Auto)
8 lm.fit <- lm(mpg ~ horsepower, data = Auto)
9 summary(lm.fit)
```

package 'ISLR' was built under R version 3.6.3

```
> library(MASS)
> data("Auto")
> head(Auto)
  mpg cylinders displacement horsepower weight acceleration year origin
1  18         8         307         130   3504         12.0    70     1
2  15         8         350         165  3693         11.5    70     1
3  18         8         318         150  3436         11.0    70     1
4  16         8         304         150  3433         12.0    70     1
5  17         8         302         140  3449         10.5    70     1
6  15         8         429         198  4341         10.0    70     1
```

name
1 chevrolet chevelle malibu
2 buick skylark 320
3 plymouth satellite
4 amc rebel sst
5 ford torino
6 ford galaxie 500

Applying linear regression model



```
1 setwd("E:/Fall2020/ISL-1/Lab/Lab2")
2 print(getwd())
3 #1. using lm() to perform linear regression
4 library(ISLR)
5 library(MASS)
6 data("Auto")
7 head(Auto)
8 lm.fit <- lm(mpg ~ horsepower, data = Auto)
9 summary(lm.fit)
```

Console

```
E:/Fall2020/ISL-1/Lab/Lab2/
4      amc rebel sst
5      ford torino
6      ford galaxie 500
> lm.fit <- lm(mpg ~ horsepower, data = Auto)
> summary(lm.fit)

Call:
lm(formula = mpg ~ horsepower, data = Auto)

Residuals:
    Min       1Q   Median       3Q      Max
-13.5710  -3.2592  -0.3435   2.7630  16.9240

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  39.935861   0.717499   55.66  <2e-16 ***
horsepower  -0.157845   0.006446  -24.49  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.906 on 390 degrees of freedom
Multiple R-squared:  0.6059,    Adjusted R-squared:  0.6049
F-statistic: 599.7 on 1 and 390 DF,  p-value: < 2.2e-16

> |
```

i. Is there a relationship between the predictor and the response?

The p-values from the above for the regression coefficients are nearly zero ($2e-16$). This shows statistical significance, which in turn means that there exists a relationship.

ii. How strong is the relationship between the predictor and the response?

The R^2 value indicates that about 61% of the variation in the response variable (mpg) is due to the predictor variable (horsepower).

iii. Is the relationship between the predictor and the response positive or negative?

The regression coefficient for 'horsepower' is negative. Hence, the relationship is negative.

iv. What is the predicted mpg associated with a horsepower of 98? What are the associated 95% confidence and prediction intervals?

```

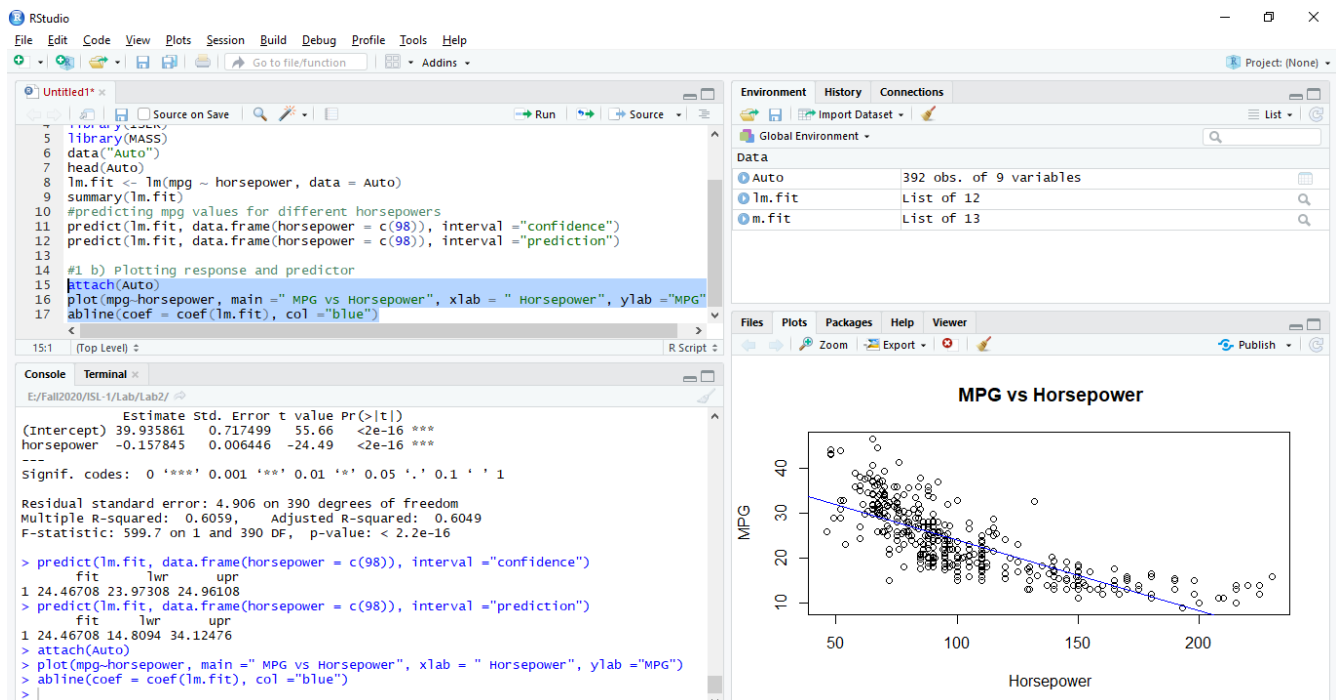
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins
Untitled1* x
Source on Save Run Source
1 setwd("E:/Fall2020/ISL-1/Lab/Lab2")
2 print(getwd())
3 #1. a) using lm() to perform linear regression
4 library(ISLR)
5 library(MASS)
6 data("Auto")
7 head(Auto)
8 lm.fit <- lm(mpg ~ horsepower, data = Auto)
9 summary(lm.fit)
10 #predicting mpg values for different horsepowers
11 predict(lm.fit, data.frame(horsepower = c(98)), interval = "confidence")
12 predict(lm.fit, data.frame(horsepower = c(98)), interval = "prediction")
10:1 (Top Level) R Script
Console Terminal x
E:/Fall2020/ISL-1/Lab/Lab2/
-13.5710 -3.2592 -0.3435 2.7630 16.9240
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  39.935861   0.717499   55.66  <2e-16 ***
horsepower   -0.157845   0.006446  -24.49  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.906 on 390 degrees of freedom
Multiple R-squared:  0.6059,    Adjusted R-squared:  0.6049
F-statistic: 599.7 on 1 and 390 DF,  p-value: < 2.2e-16

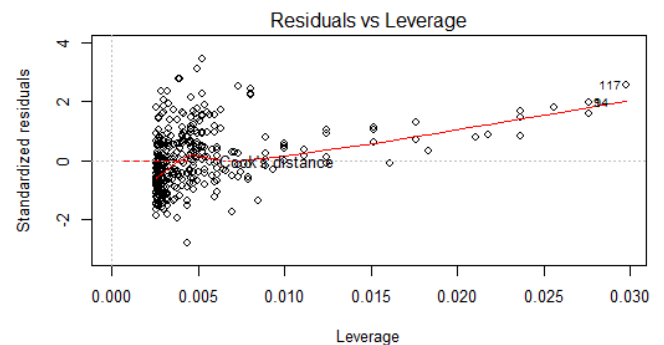
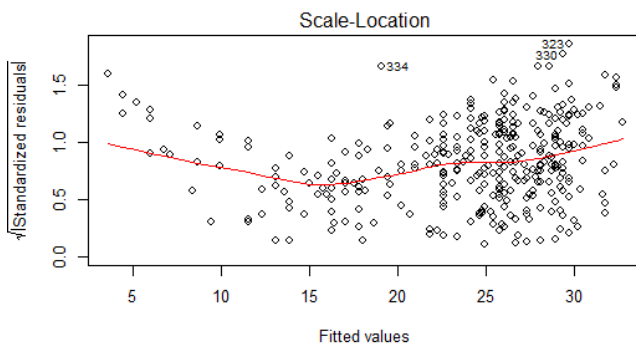
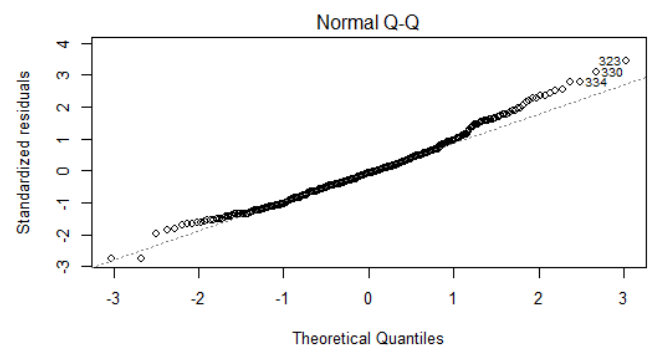
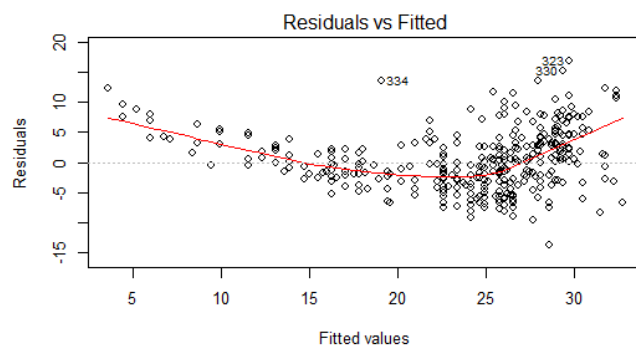
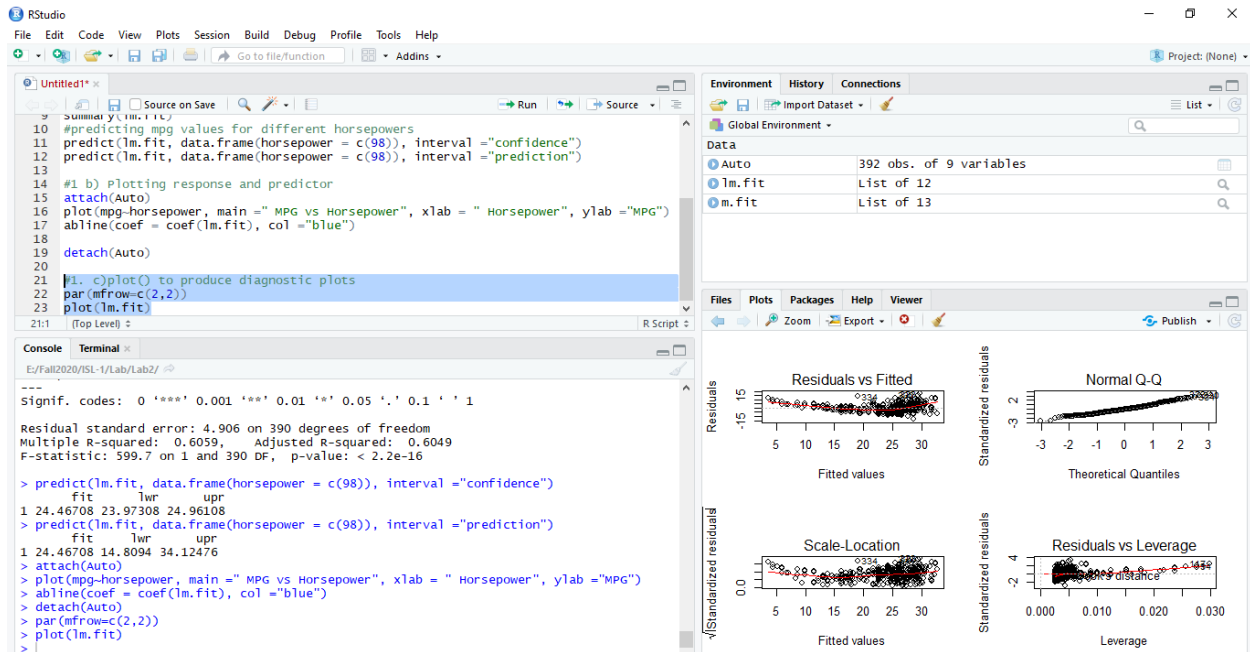
> predict(lm.fit, data.frame(horsepower = c(98)), interval = "confidence")
      fit      lwr      upr
1 24.46708 23.97308 24.96108
> predict(lm.fit, data.frame(horsepower = c(98)), interval = "prediction")
      fit      lwr      upr
1 24.46708 14.8094 34.12476
>

```

(b) Plot the response and the predictor. Use the abline() function to display the least squares regression line.



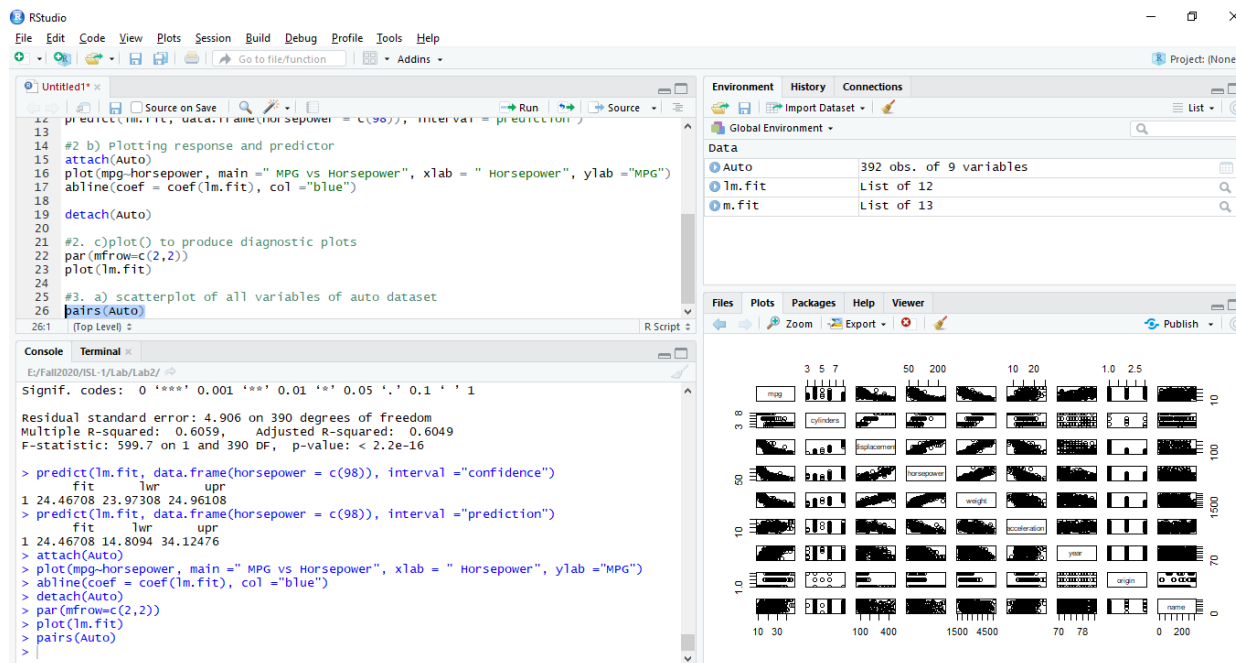
(c) Use the `plot()` function to produce diagnostic plots of the least squares regression fit. Comment on any problems you see with the fit.

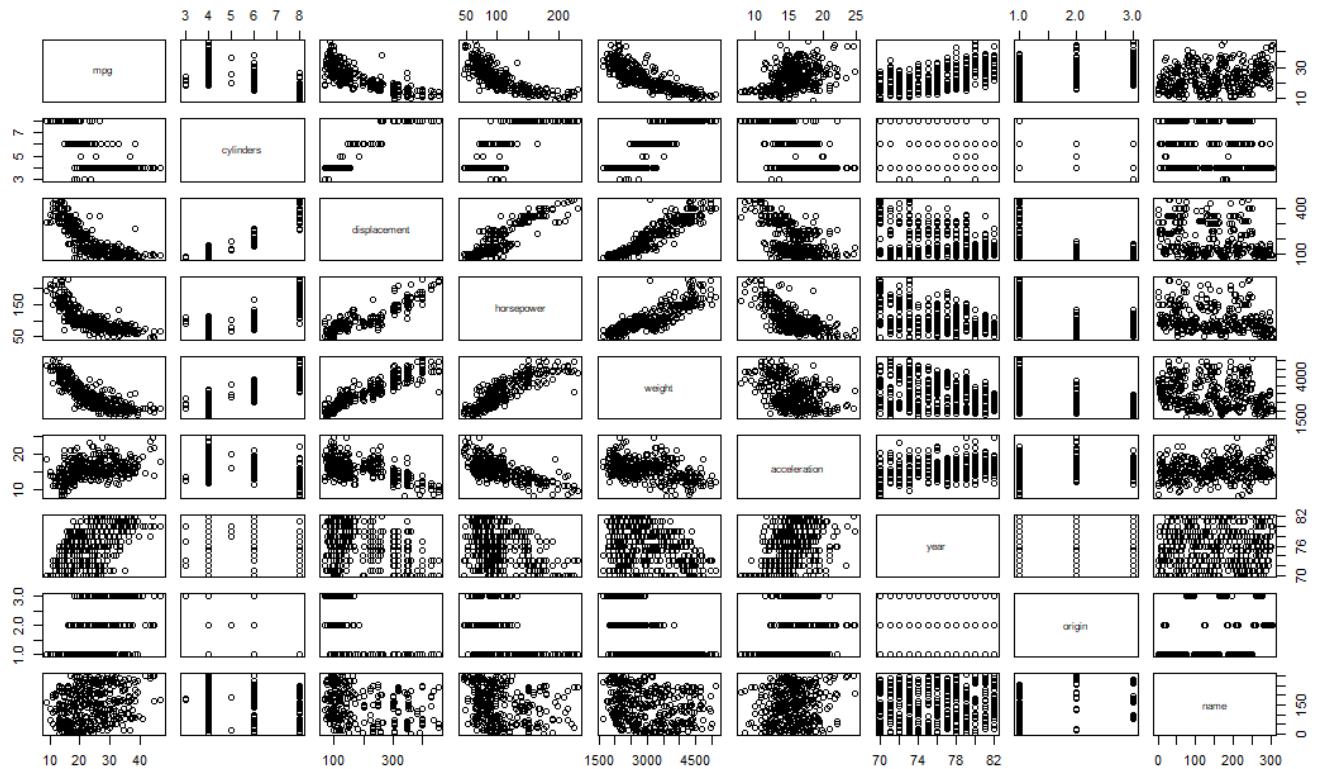


The first plot shows a pattern (U-shaped) between the residuals and the fitted values. This indicates a non-linear relationship between the predictor and response variables. The second plot shows that the residuals are normally distributed. The third plot shows that the variance of the errors is constant. Finally, the fourth plot indicates that there are no leverage points in the data.

3. This question involves the use of multiple linear regression on the Auto data set.

(a) Produce a scatterplot matrix which includes all of the variables in the data set.





b) Compute the matrix of correlations between the variables using the function `cor()`. You will need to exclude the name variable, which is qualitative.

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

```

#2. b) Plotting response and predictor
14 attach(Auto)
15 plot(mpg~horsepower, main = "MPG vs Horsepower", xlab = "Horsepower", ylab = "MPG")
16 abline(coef = coef(lm.fit), col = "blue")
17 detach(Auto)
18
19 #2. c) plot() to produce diagnostic plots
20 par(mfrow=c(2,2))
21 plot(lm.fit)
22
23 #3. a) scatterplot of all variables of auto dataset
24 pairs(Auto)
25 #3. b) correlation between variables using cor()
26 cor(Auto[, names(Auto) != "name"])
27
28:1 (Top Level)
R Script

```

```

> cor(Auto[, names(Auto) != "name"])
      mpg      cylinders displacement horsepower      weight acceleration
mpg      1.0000000 -0.7776175 -0.8051269 -0.7784268 -0.8322442  0.4233285
cylinders -0.7776175  1.0000000  0.9508233  0.8429834  0.8975273 -0.5046834
displacement -0.8051269  0.9508233  1.0000000  0.8972570  0.9329944 -0.5438005
horsepower -0.7784268  0.8429834  0.8972570  1.0000000  0.8645377 -0.6891955
weight -0.8322442  0.8975273  0.9329944  0.8645377  1.0000000 -0.4168392
acceleration 0.4233285 -0.5046834 -0.5438005 -0.6891955 -0.4168392  1.0000000
year      0.5805410 -0.3456474 -0.3698552 -0.4163615 -0.3091199  0.2903161
origin     0.5652088 -0.5689316 -0.6145351 -0.4551715 -0.5850054  0.2127458
      year      origin
mpg      0.5805410  0.5652088
cylinders -0.3456474 -0.5689316
displacement -0.3698552 -0.6145351
horsepower -0.4163615 -0.4551715
weight -0.3091199 -0.5850054
acceleration 0.2903161  0.2127458
year      1.0000000  0.1815277
origin     0.1815277  1.0000000

```

(c) Use the `lm()` function to perform a multiple linear regression with `mpg` as the response and all other variables except `name` as the predictors. Use the `summary()` function to print the results.

```

10 detach(Auto)
19
20
21 #2. c)plot() to produce diagnostic plots
22 par(mfrow=c(2,2))
23 plot(lm.fit)
24
25 #3. a) scatterplot of all variables of auto dataset
26 pairs(Auto)
27 #3. b) correlation between variables using cor()
28 cor(Auto[, names(Auto) != "name"])
29
30 #3.c) use lm() to perform multiple linear regression
31 model = lm(mpg ~. -name, data = Auto)
32 summary(model)
31:1 (Top Level)
R Script

```

```

> model = lm(mpg ~. -name, data = Auto)
> summary(model)

Call:
lm(formula = mpg ~ . - name, data = Auto)

Residuals:
    Min       1Q   Median       3Q      Max
-9.5903 -2.1565 -0.1169  1.8690 13.0604

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -17.218435   4.644294  -3.707  0.00024 ***
cylinders    -0.493376   0.323282  -1.526  0.12780
displacement  0.019896   0.007515   2.647  0.00844 **
horsepower   -0.016951   0.013787  -1.230  0.21963
weight       -0.006474   0.000652  -9.929 < 2e-16 ***
acceleration  0.080576   0.098845   0.815  0.41548
year          0.750773   0.050973  14.729 < 2e-16 ***
origin        1.426141   0.278136   5.127  4.67e-07 ***

```

i. Is there a relationship between the predictors and the response?

Yes, there is. However, some predictors do not have a statistically significant effect on the response. R-squared value implies that 82% of the changes in the response can be explained by the predictors in this regression model.

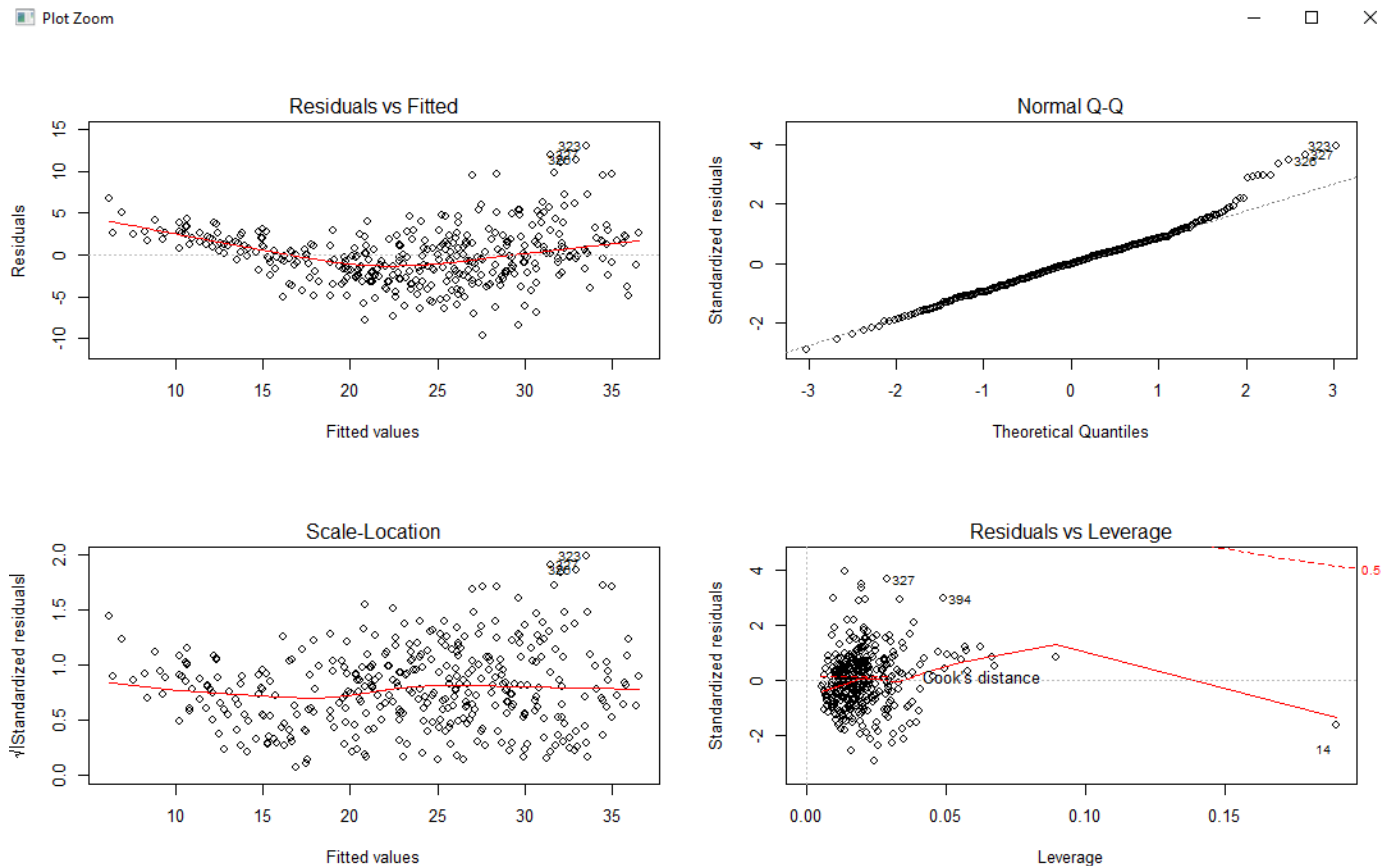
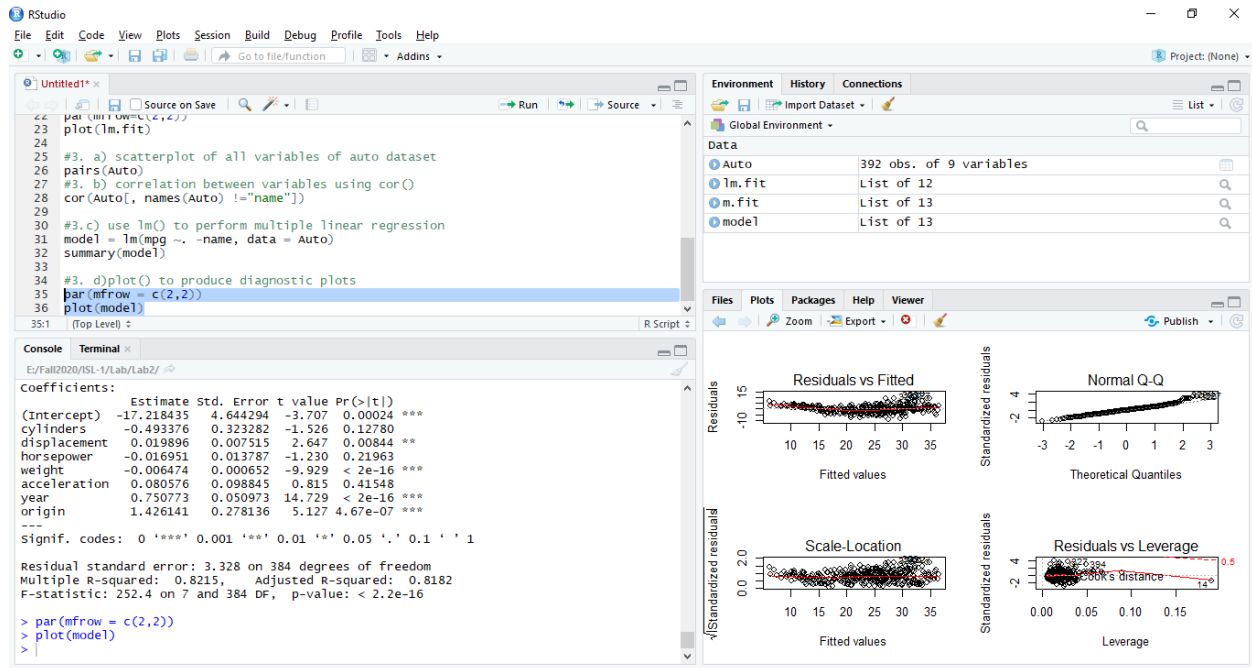
ii. Which predictors appear to have a statistically significant relationship to the response?

Displacement, weight, year, origin have significant relationship to the response.

iii. What does the coefficient for the year variable suggest?

When every other predictor held constant, the mpg value increases with each year that passes. Specifically, mpg increase by 1.43 each year.

(d) Use the `plot()` function to produce diagnostic plots of the linear regression fit. Comment on any problems you see with the fit. Do the residual plots suggest any unusually large outliers? Does the leverage plot identify any observations with unusually high leverage?



1. The first graph shows that there is a non-linear relationship between the response and the predictors
2. The second graph shows that the residuals are normally distributed and right skewed

3. The third graph shows that the constant variance of error assumption is not true for this model
4. The fourth graphs shows that there are no leverage points. However, there on observation that stands out as a potential leverage point

(e) Use the * and : symbols to fit linear regression models with interaction effects. Do any interactions appear to be statistically significant?

The screenshot shows the RStudio interface. The script editor contains the following code:

```

31 model = lm(mpg ~ . - name, data = Auto)
32 summary(model)
33
34 #3. d) plot() to produce diagnostic plots
35 par(mfrow = c(2,2))
36 plot(model)
37 #3. e) Use symbols to fit linear regression
38 model = lm(mpg ~ . - name + displacement:weight, data = Auto)
39 summary(model)

```

The console output shows the results of the model fit:

```

Call:
lm(formula = mpg ~ . - name + displacement:weight, data = Auto)

Residuals:
    Min       1Q   Median       3Q      Max
-9.9027 -1.8092 -0.0946  1.5549 12.1687

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -5.389e+00  4.301e+00  -1.253   0.2109
cylinders      1.175e-01  2.943e-01   0.399   0.6899
displacement  -6.837e-02  1.104e-02  -6.193 1.52e-09 ***
horsepower    -3.280e-02  1.238e-02  -2.649  0.0084 **
weight        -1.064e-02  7.136e-04 -14.915 < 2e-16 ***
acceleration   6.724e-02  8.805e-02   0.764   0.4455
year           7.852e-01  4.553e-02  17.246 < 2e-16 ***
origin         5.610e-01  2.622e-01   2.139  0.0331 *
displacement:weight  2.269e-05  2.257e-06  10.054 < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.964 on 383 degrees of freedom
Multiple R-squared:  0.8588,    Adjusted R-squared:  0.8558
F-statistic: 291.1 on 8 and 383 DF,  p-value: < 2.2e-16

```

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

Untitled1*

Source on Save Run Source

```

38 model = lm(mpg ~ . - name + displacement:weight, data = Auto)
39 summary(model)
40
41 model = lm(mpg ~ . - name + displacement:cylinders + displacement:weight + acceleration:horsepower, data=Auto)
42 summary(model)
43:1 (Top Level)

```

R Script

Console Terminal

E:/Fall2020/ISL-1/Lab/Lab2/

```

> model = lm(mpg ~ . - name + displacement:cylinders + displacement:weight + acceleration:horsepower, data=Auto)
> summary(model)

Call:
lm(formula = mpg ~ . - name + displacement:cylinders + displacement:weight +
    acceleration:horsepower, data = Auto)

Residuals:
    Min       1Q   Median       3Q      Max
-9.3344 -1.6333  0.0188  1.4740 11.9723

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.725e+01  5.328e+00  -3.237  0.00131 ***
cylinders      6.354e-01  6.106e-01   1.041  0.29870
displacement  -6.805e-02  1.337e-02  -5.088  5.68e-07 ***
horsepower     6.026e-02  2.601e-02   2.317  0.02105 *
weight        -8.864e-03  1.097e-03  -8.084  8.43e-15 ***
acceleration   6.257e-01  1.592e-01   3.931  0.00010 ***
year          7.845e-01  4.470e-02  17.549  < 2e-16 ***
origin        4.668e-01  2.595e-01   1.799  0.07284 .
cylinders:displacement -1.337e-03  2.726e-03  -0.490  0.62415
displacement:weight  2.071e-05  3.638e-06   5.694  2.49e-08 ***
horsepower:acceleration -7.467e-03  1.784e-03  -4.185  3.55e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.905 on 381 degrees of freedom
Multiple R-squared:  0.865,    Adjusted R-squared:  0.8615
F-statistic: 244.2 on 10 and 381 DF,  p-value: < 2.2e-16

```

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

Untitled1*

Source on Save Run Source

```

42 summary(model)
43
44 model = lm(mpg ~ . - name + displacement:cylinders + displacement:weight + year:origin + acceleration:horsepower, data=Auto)
45 summary(model)
46:1 (Top Level)

```

R Script

Console Terminal

E:/Fall2020/ISL-1/Lab/Lab2/

```

> summary(model)

Call:
lm(formula = mpg ~ . - name + displacement:cylinders + displacement:weight +
    year:origin + acceleration:horsepower, data = Auto)

Residuals:
    Min       1Q   Median       3Q      Max
-8.6504 -1.6476  0.0381  1.4254 12.7893

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.287e+00  9.074e+00   0.583  0.560429
cylinders     4.249e-01  6.079e-01   0.699  0.485011
displacement  -7.322e-02  1.334e-02  -5.490  7.38e-08 ***
horsepower     5.252e-02  2.586e-02   2.031  0.042913 *
weight        -8.689e-03  1.086e-03  -7.998  1.54e-14 ***
acceleration   5.796e-01  1.582e-01   3.665  0.000283 ***
year          5.116e-01  9.976e-02   5.129  4.66e-07 ***
origin       -1.220e+01  4.161e+00  -2.933  0.003560 **
cylinders:displacement -4.368e-04  2.712e-03  -0.161  0.872156
displacement:weight  1.992e-05  3.608e-06   5.522  6.21e-08 ***
year:origin    1.630e-01  5.341e-02   3.051  0.002440 **
horsepower:acceleration -6.735e-03  1.781e-03  -3.781  0.000181 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.874 on 380 degrees of freedom
Multiple R-squared:  0.8683,    Adjusted R-squared:  0.8644
F-statistic: 227.7 on 11 and 380 DF,  p-value: < 2.2e-16

```

The screenshot shows the RStudio interface. The source editor contains the following R code:

```

46
47 model = lm(mpg ~ . - name - cylinders - acceleration + year:origin + displacement:weight +
48             displacement:weight + acceleration:horsepower + acceleration:weight, data=Auto)
49 summary(model)

```

The console shows the output of the `summary(model)` command:

```

> model = lm(mpg ~ . - name - cylinders - acceleration + year:origin + displacement:weight +
+             displacement:weight + acceleration:horsepower + acceleration:weight, data=Auto)
> summary(model)

Call:
lm(formula = mpg ~ . - name - cylinders - acceleration + year:origin +
    displacement:weight + displacement:weight + acceleration:horsepower +
    acceleration:weight, data = Auto)

Residuals:
    Min       1Q   Median       3Q      Max
-9.5074 -1.6324  0.0599  1.4577 12.7376

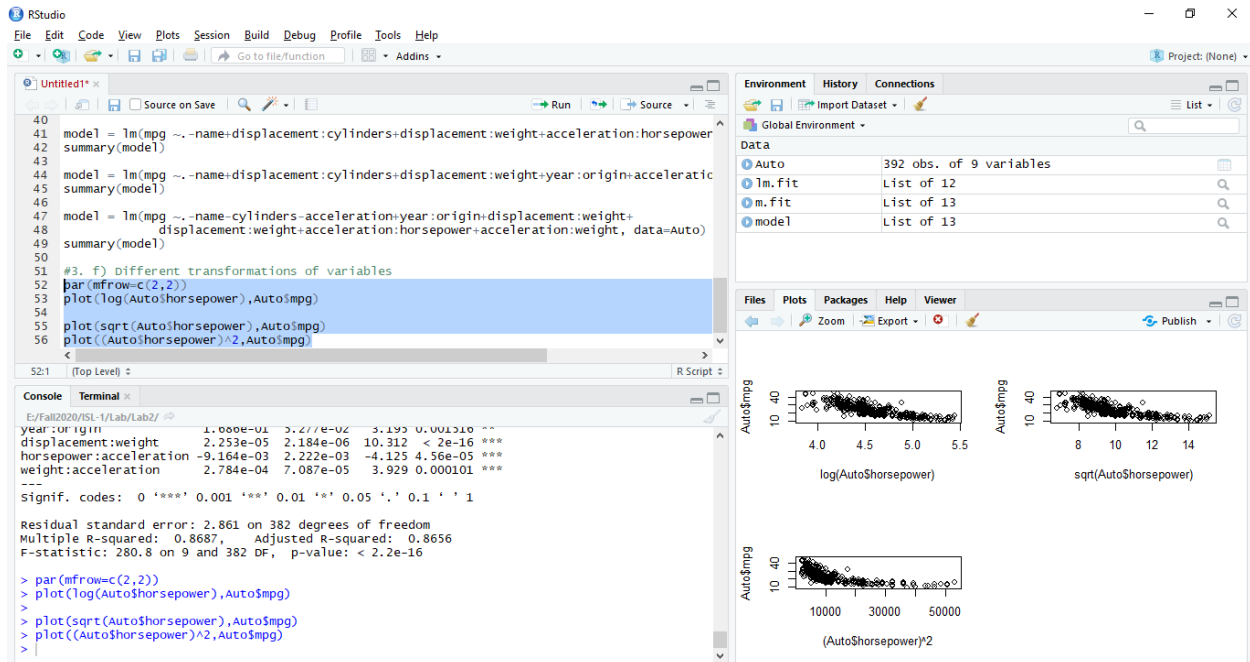
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.868e+01  7.796e+00   2.396  0.017051 *
displacement -7.794e-02  9.026e-03  -8.636 < 2e-16 ***
horsepower   8.719e-02  3.167e-02   2.753  0.006183 **
weight      -1.350e-02  1.287e-03 -10.490 < 2e-16 ***
year         4.911e-01  9.825e-02   4.998  8.83e-07 ***
origin      -1.262e+01  4.109e+00  -3.071  0.002288 **
year:origin   1.686e-01  5.277e-02   3.195  0.001516 **
displacement:weight  2.253e-05  2.184e-06  10.312 < 2e-16 ***
horsepower:acceleration -9.164e-03  2.222e-03  -4.125  4.56e-05 ***
weight:acceleration  2.784e-04  7.087e-05   3.929  0.000101 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.861 on 382 degrees of freedom
Multiple R-squared:  0.8687,    Adjusted R-squared:  0.8656

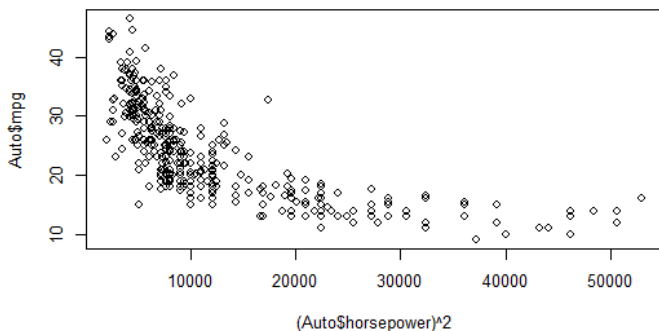
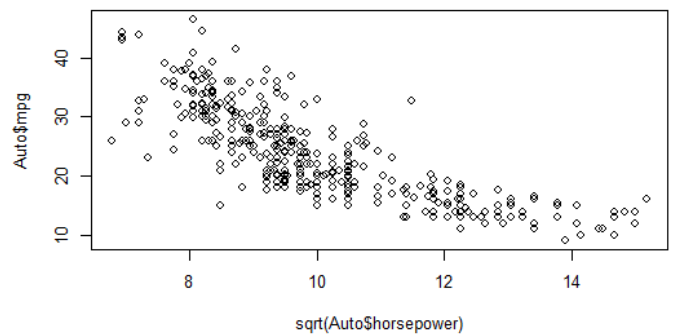
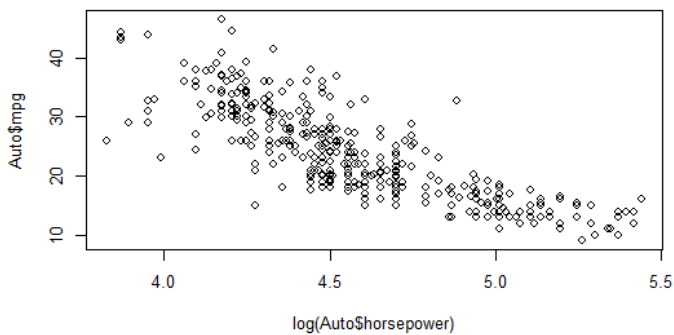
```

From all the 4 models, the last model is the only one with all variables being significant. And, based on results from a few trials not show here, it is very likely that it is the best combination of predictors and interaction terms. The R-squared statistics estimates that 87% of the changes in the response can be explained by this particular set of predictors (single and interaction.) A higher value was not obtained from the trials.

(f) Try a few different transformations of the variables, such as $\log(X)$, \sqrt{X} , X^2 . Comment on your findings.



Plot Zoom



4. This question should be answered using the Carseats data set.

(a) Fit a multiple regression model to predict Sales using Price, Urban, and US

```

#4. a) Multiple regression model to predict sales
?Carseats
head(Carseats)
str(Carseats)
data(Carseats)
lm.fit=lm(Sales ~ Price + Urban + US, data = Carseats)
summary(lm.fit)

```

```

$ Education : num  17 10 12 14 13 16 15 10 10 17 ...
$ Urban      : Factor w/ 2 levels "No","Yes": 2 2 2 2 2 1 2 2 1 1 ...
$ US         : Factor w/ 2 levels "No","Yes": 2 2 2 2 1 2 1 2 1 2 ...
> data(Carseats)
> lm.fit=lm(Sales ~ Price + Urban + US, data = Carseats)
> summary(lm.fit)

Call:
lm(formula = Sales ~ Price + Urban + US, data = Carseats)

Residuals:
    Min       1Q   Median       3Q      Max
-6.9206 -1.6220 -0.0564  1.5786  7.0581

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 13.043469   0.651012  20.036 < 2e-16 ***
Price       -0.054459   0.005242 -10.389 < 2e-16 ***
UrbanYes    -0.021916   0.271650  -0.081  0.936
USYes       1.200573    0.259042   4.635 4.86e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.472 on 396 degrees of freedom
Multiple R-squared:  0.2393,    Adjusted R-squared:  0.2335
F-statistic: 41.52 on 3 and 396 DF,  p-value: < 2.2e-16

```

(b) Provide an interpretation of each coefficient in the model. Be careful some of the variables in the model are qualitative!

Price:

“Price” variable may be interpreted by saying that the average effect of a price increase of 1 dollar is a decrease of 54.4588492 units in sales all other predictors remaining fixed.

Urban:

“Urban” variable may be interpreted by saying that on average the unit sales in urban location are 21.9161508 units less than in rural location all other predictors remaining fixed.

US:

“US” variable may be interpreted by saying that on average the unit sales in a US store are 1200.5726978 units more than in a non US store all other predictors remaining fixed.

(c) Write out the model in equation form, being careful to handle the qualitative variables properly.

Model in equation form can be written as:

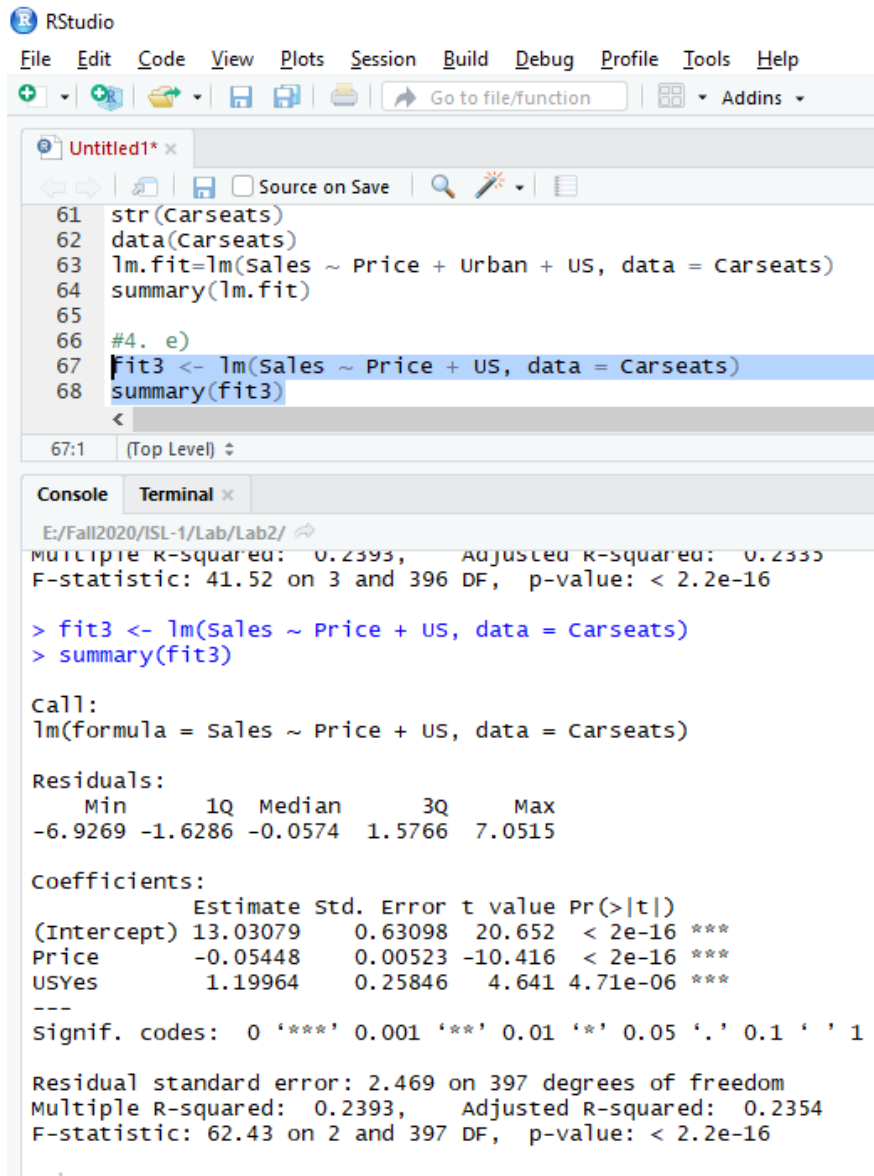
$$\text{Sales} = 13.0434689 + (-0.0544588) \times \text{Price} + (-0.0219162) \times \text{Urban} + (1.2005727) \times \text{US} + \epsilon$$

If store is in urban then urban=1 otherwise urban=0 If store is in us then us=1 otherwise us=0.

(d) For which of the predictors can you reject the null hypothesis $H_0 : \beta_j = 0$

For US and Price variables we can reject null hypothesis

(e) On the basis of your response to the previous question, fit a smaller model that only uses the predictors for which there is evidence of association with the outcome.



The screenshot shows the RStudio environment. The source editor contains the following R code:

```
61 str(Carseats)
62 data(Carseats)
63 lm.fit=lm(Sales ~ Price + Urban + US, data = Carseats)
64 summary(lm.fit)
65
66 #4. e)
67 fit3 <- lm(Sales ~ Price + US, data = Carseats)
68 summary(fit3)
```

The console shows the output of the code:

```
E:/Fall2020/ISL-1/Lab/Lab2/
Multiple R-squared:  0.2393,    Adjusted R-squared:  0.2333
F-statistic: 41.52 on 3 and 396 DF,  p-value: < 2.2e-16

> fit3 <- lm(Sales ~ Price + US, data = Carseats)
> summary(fit3)

Call:
lm(formula = Sales ~ Price + US, data = Carseats)

Residuals:
    Min       1Q   Median       3Q      Max
-6.9269 -1.6286 -0.0574  1.5766  7.0515

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  13.03079    0.63098   20.652 < 2e-16 ***
Price       -0.05448    0.00523  -10.416 < 2e-16 ***
USYes        1.19964    0.25846   4.641 4.71e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.469 on 397 degrees of freedom
Multiple R-squared:  0.2393,    Adjusted R-squared:  0.2354
F-statistic: 62.43 on 2 and 397 DF,  p-value: < 2.2e-16
```

(f) How well do the models in (a) and (e) fit the data?

By looking R^2 values for smaller model is better than bigger model. Variability of 24% is shown by the model.

(g) Using the model from (e), obtain 95% confidence intervals for the coefficient(s).

```

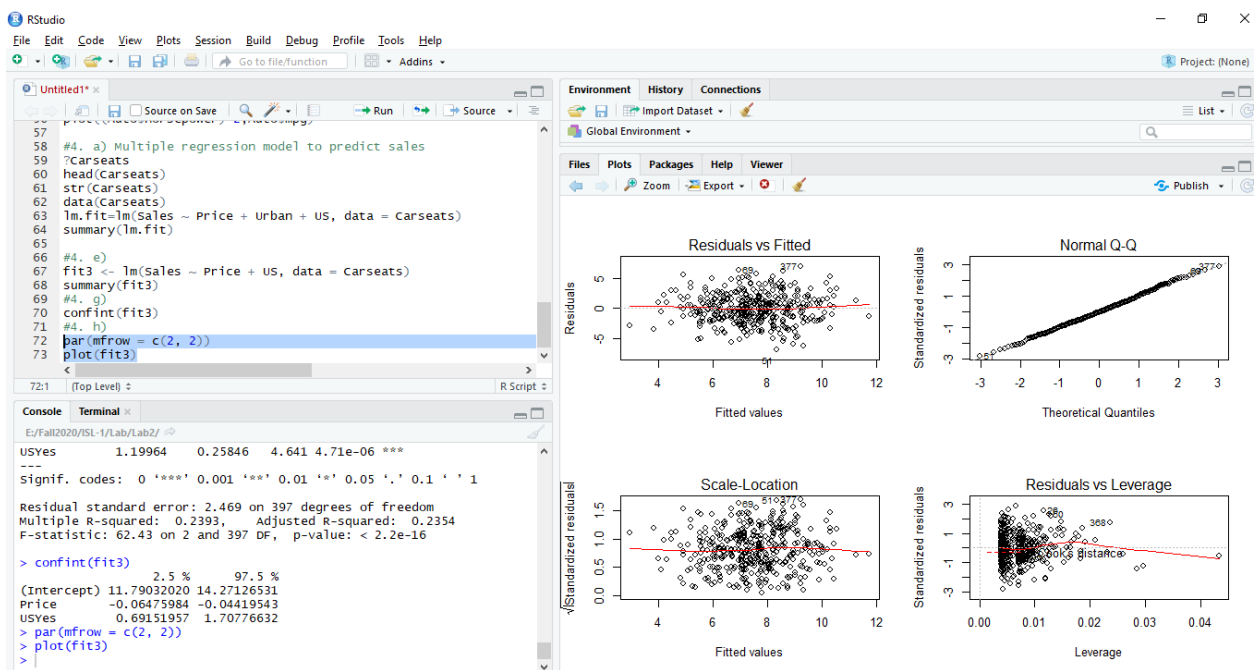
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
+ - Go to file/function Addins
Untitled1* x
54 price ~ log(horsepower) + log(mpg)
55 plot(sqrt(Auto$horsepower), Auto$mpg)
56 plot((Auto$horsepower)^2, Auto$mpg)
57
58 #4. a) Multiple regression model to predict sales
59 ?Carseats
60 head(Carseats)
61 str(Carseats)
62 data(Carseats)
63 lm.fit = lm(Sales ~ Price + Urban + US, data = Carseats)
64 summary(lm.fit)
65
66 #4. e)
67 fit3 <- lm(Sales ~ Price + US, data = Carseats)
68 summary(fit3)
69 #4. g)
70 confint(fit3)
71
70:1 (Top Level)
Console Terminal
E:/Fall2020/ISL-1/Lab/Lab2/
(Intercept) 13.03079 0.63098 20.652 < 2e-16 ***
Price -0.05448 0.00523 -10.416 < 2e-16 ***
USYes 1.19964 0.25846 4.641 4.71e-06 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.469 on 397 degrees of freedom
Multiple R-squared: 0.2393, Adjusted R-squared: 0.2354
F-statistic: 62.43 on 2 and 397 DF, p-value: < 2.2e-16

> confint(fit3)
2.5 % 97.5 %
(Intercept) 11.79032020 14.27126531
Price -0.06475984 -0.04419543
USYes 0.69151957 1.70776632
> |

```

(h) Is there evidence of outliers or high leverage observations in the model from (e)?

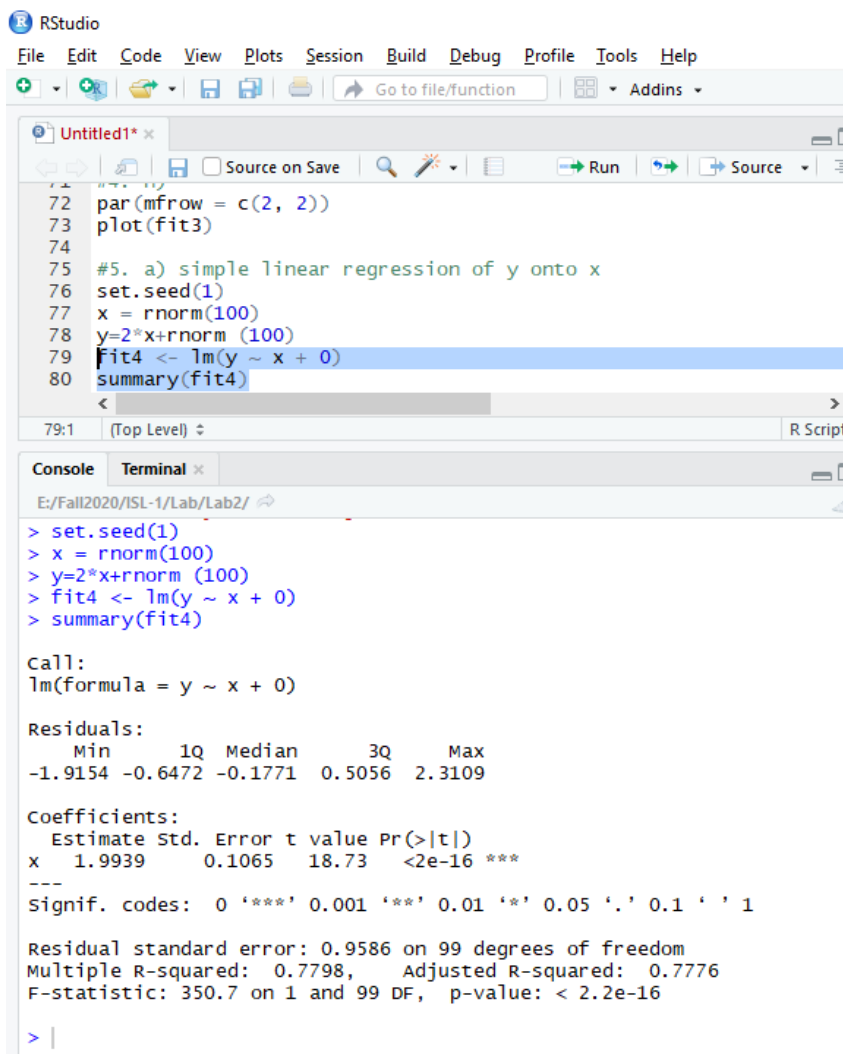


The plots of residual vs leverage indicates that there are few outliers and some leverage points.

5. In this problem we will investigate the t-statistic for the null hypothesis $H_0 : \beta = 0$ in simple linear regression without an intercept. To begin, we generate a predictor x and a response y as follows.

```
> set.seed(1)
> x = rnorm(100)
> y=2*x+rnorm (100)
```

(a) Perform a simple linear regression of y onto x , without an intercept. Report the coefficient estimate $\hat{\beta}$, the standard error of this coefficient estimate, and the t-statistic and p-value associated with the null hypothesis $H_0 : \beta = 0$. Comment on these results. (You can perform regression without an intercept using the command `lm(y ~ x + 0)`.)



The screenshot shows the RStudio interface. The script editor contains the following code:

```
72 par(mfrow = c(2, 2))
73 plot(fit3)
74
75 #5. a) simple linear regression of y onto x
76 set.seed(1)
77 x = rnorm(100)
78 y=2*x+rnorm (100)
79 fit4 <- lm(y ~ x + 0)
80 summary(fit4)
```

The console output shows the results of the regression:

```
> set.seed(1)
> x = rnorm(100)
> y=2*x+rnorm (100)
> fit4 <- lm(y ~ x + 0)
> summary(fit4)

Call:
lm(formula = y ~ x + 0)

Residuals:
    Min       1Q   Median       3Q      Max
-1.9154 -0.6472 -0.1771  0.5056  2.3109

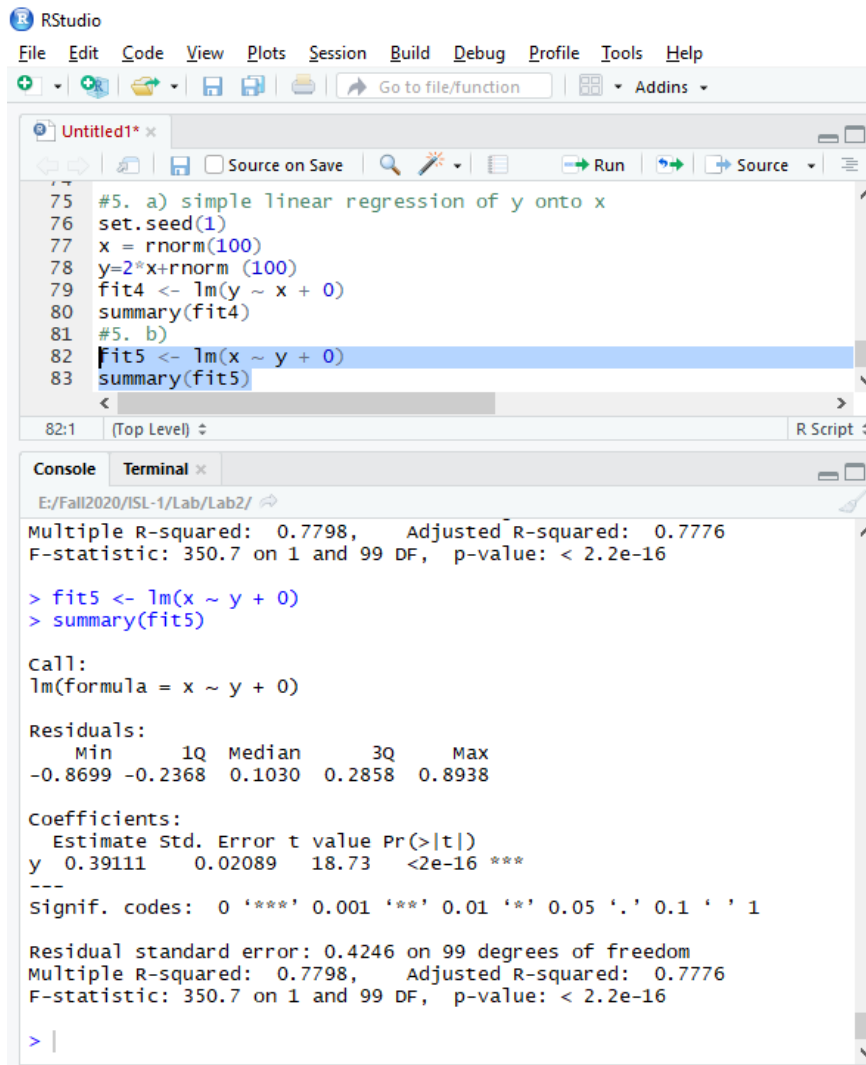
Coefficients:
      Estimate Std. Error t value Pr(>|t|)
x      1.9939     0.1065   18.73  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9586 on 99 degrees of freedom
Multiple R-squared:  0.7798,    Adjusted R-squared:  0.7776
F-statistic: 350.7 on 1 and 99 DF,  p-value: < 2.2e-16

> |
```

Based on these results, the predictor 'x' is statistically significant for estimating the response variable 'y'.

(b) Now perform a simple linear regression of x onto y without an intercept, and report the coefficient estimate, its standard error, and the corresponding t-statistic and p-values associated with the null hypothesis $H_0 : \beta = 0$. Comment on these results.



The screenshot shows the RStudio interface. The source editor contains the following R code:

```

75 #5. a) simple linear regression of y onto x
76 set.seed(1)
77 x = rnorm(100)
78 y=2*x+rnorm (100)
79 fit4 <- lm(y ~ x + 0)
80 summary(fit4)
81 #5. b)
82 fit5 <- lm(x ~ y + 0)
83 summary(fit5)

```

The console shows the output of the R script:

```

Multiple R-squared:  0.7798,    Adjusted R-squared:  0.7776
F-statistic: 350.7 on 1 and 99 DF,  p-value: < 2.2e-16

> fit5 <- lm(x ~ y + 0)
> summary(fit5)

Call:
lm(formula = x ~ y + 0)

Residuals:
    Min       1Q   Median       3Q      Max
-0.8699 -0.2368  0.1030  0.2858  0.8938

Coefficients:
      Estimate Std. Error t value Pr(>|t|)
y  0.39111    0.02089   18.73  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

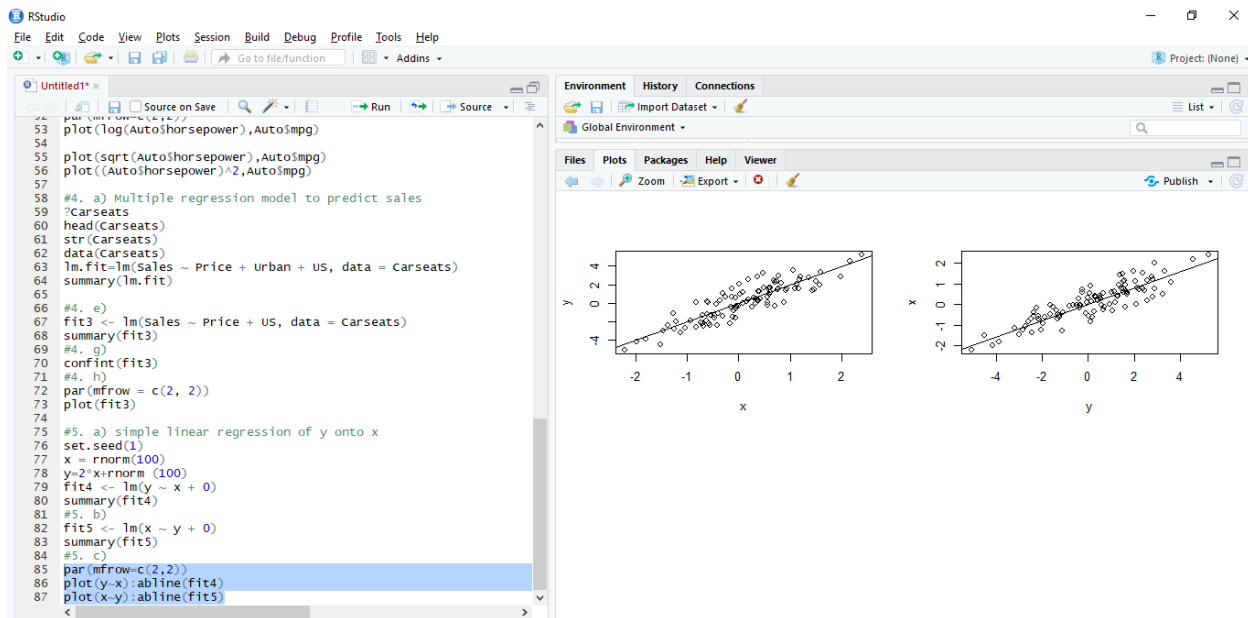
Residual standard error: 0.4246 on 99 degrees of freedom
Multiple R-squared:  0.7798,    Adjusted R-squared:  0.7776
F-statistic: 350.7 on 1 and 99 DF,  p-value: < 2.2e-16

> |

```

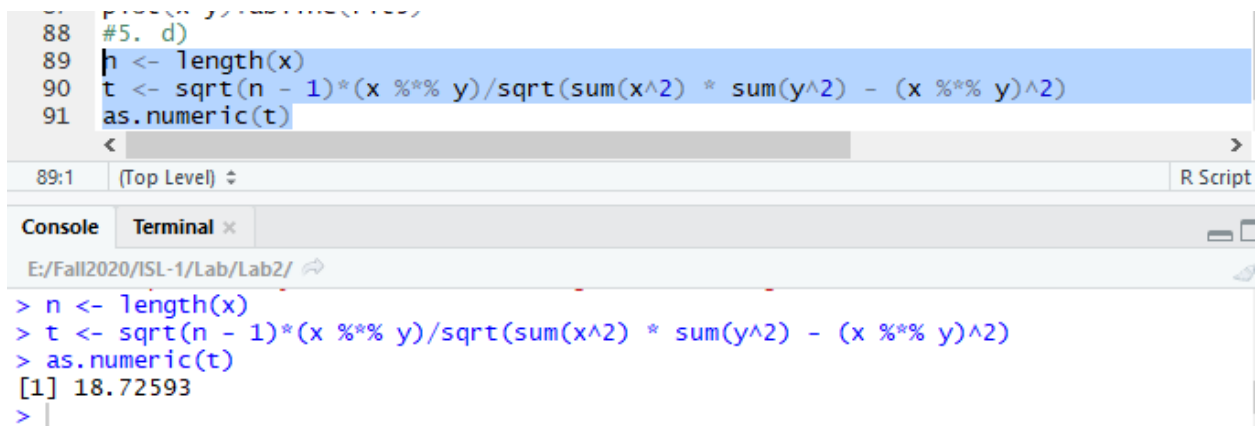
Same observation as the previous model. The predictor variable is significance.

(c) What is the relationship between the results obtained in (a) and (b)?



We obtain the same value for the t-statistic and consequently the same value for the corresponding p-value. Both results in (a) and (b) reflect the same line created in (a). In other words, $y = 2x + \epsilon$ could also be written $x = 0.5(y - \epsilon)$.

(d) For the regression of Y onto X without an intercept, the t-statistic for $H_0 : \beta = 0$ takes the form $\beta / SE(\hat{\beta})$, where $\hat{\beta}$ is given by (3.38), and where $SE(\hat{\beta}) = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (y_i - x_i \hat{\beta})^2}$. (These formulas are slightly different from those given in Sections 3.1.1 and 3.1.2, since here we are performing regression without an intercept.) Show algebraically, and confirm numerically in R, that the t-statistic can be written as $(\sqrt{n-1} \sum_{i=1}^n x_i y_i) / \sqrt{(\sum_{i=1}^n x_i^2)(\sum_{i=1}^n y_i^2) - (\sum_{i=1}^n x_i y_i)^2}$.



We may see that t-value is same as the t-statistic in above 5b.

(e) Using the results from (d), argue that the t-statistic for the regression of y onto x is the same as the t-statistic for the regression of x onto y.

It is clear that if we replace x_i by y_i in the formula for the t-statistic, the result will be the same.

(f) In R, show that when regression is performed with an intercept, the t-statistic for $H_0 : \beta_1 = 0$ is the same for the regression of y onto x as it is for the regression of x onto y

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
[Icons] Go to file/function Addins

Untitled1* x
[Icons] Source on Save [Icons] Run [Icons]

85 par(mfrow=c(2,2))
86 plot(y~x):abline(fit4)
87 plot(x~y):abline(fit5)
88 #5. d)
89 n <- length(x)
90 t <- sqrt(n - 1)*(x %>% y)/sqrt(sum(x^2) * sum(y^2) - (x %>% y)^2)
91 as.numeric(t)
92
93 #5. f)
94 fit6 <- lm(y ~ x)
95 summary(fit6)
96 <
97 (Top Level) :
```

```
Console Terminal x
E:/Fall2020/ISL-1/Lab/Lab2/
> fit6 <- lm(y ~ x)
> summary(fit6)

Call:
lm(formula = y ~ x)

Residuals:
    Min       1Q   Median       3Q      Max
-1.8768 -0.6138 -0.1395  0.5394  2.3462

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.03769    0.09699   -0.389   0.698
x             1.99894    0.10773   18.556 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9628 on 98 degrees of freedom
Multiple R-squared:  0.7784,    Adjusted R-squared:  0.7762
F-statistic: 344.3 on 1 and 98 DF,  p-value: < 2.2e-16
```

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
[Icons] Go to file/function Addins

Untitled1* x
[Icons] Source on Save [Icons] Run [Icons]

88 #5. d)
89 n <- length(x)
90 t <- sqrt(n - 1)*(x %>% y)/sqrt(sum(x^2) * sum(y^2) - (x %>% y)^2)
91 as.numeric(t)
92
93 #5. f)
94 fit6 <- lm(y ~ x)
95 summary(fit6)
96
97 fit7 <- lm(x ~ y)
98 summary(fit7)
99 <
100 (Top Level) :
```

```
Console Terminal x
E:/Fall2020/ISL-1/Lab/Lab2/
> fit7 <- lm(x ~ y)
> summary(fit7)

Call:
lm(formula = x ~ y)

Residuals:
    Min       1Q   Median       3Q      Max
-0.90848 -0.28101  0.06274  0.24570  0.85736

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.03880    0.04266    0.91   0.365
y             0.38942    0.02099   18.56 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4249 on 98 degrees of freedom
Multiple R-squared:  0.7784,    Adjusted R-squared:  0.7762
F-statistic: 344.3 on 1 and 98 DF,  p-value: < 2.2e-16
> |
```

The summary tables shows that the t-value are the same (by approximation).

