1. This question involves the use of simple linear regression on the Auto data set.

(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. Comment on the output. For example:

CODE:

```
library(ISLR)
###Q1###
fit1 = lm(mpg ~ horsepower, data = Auto)
summary(fit1)
```

OUTPUT:
(1) Is there a relationship between the predictor and the response?

```
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
```

Yes. To do a hypothesis test, set H0 : B1 = 0 and H1 : B1 != 0. We know the p-value is very small, so we will reject the hypothesis, the predictor and the response have a relationship.

(2) How strong is the relationship between the predictor and the response?

R^2 = 0.6059. Almost 60% of the variability in “mpg” can be explained using “horsepower”.

(3) Is the relationship between the predictor and the response positive or negative?

The relationship between the predictor and the response is **negative** because the coefficient of horsepower is negative. As the more horsepower an automobile got, the less mpg the automobile will have.

(4) What is the predicted mpg associated with a horsepower of 98? What are the associated 95% confidence and prediction intervals?

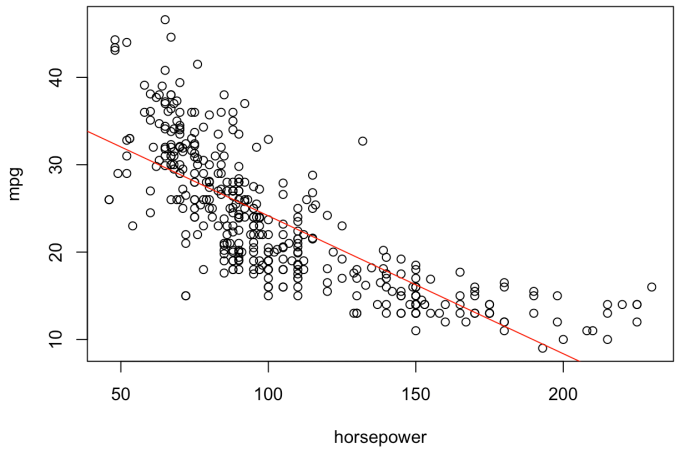
```
> #predicted mpg associated with a horsepower of 98
> predict(fit1, data.frame(horsepower = 98))
       1
24.46708
> #95% prediction intervals
> predict(fit1, data.frame(horsepower = 98), interval="prediction", level=0.95)
       fit     lwr      upr
1 24.46708 14.8094 34.12476
> #95% confidence intervals
> predict(fit1, data.frame(horsepower = 98), interval="confidence", level=0.95)
       fit      lwr      upr
1 24.46708 23.97308 24.96108
>
```

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

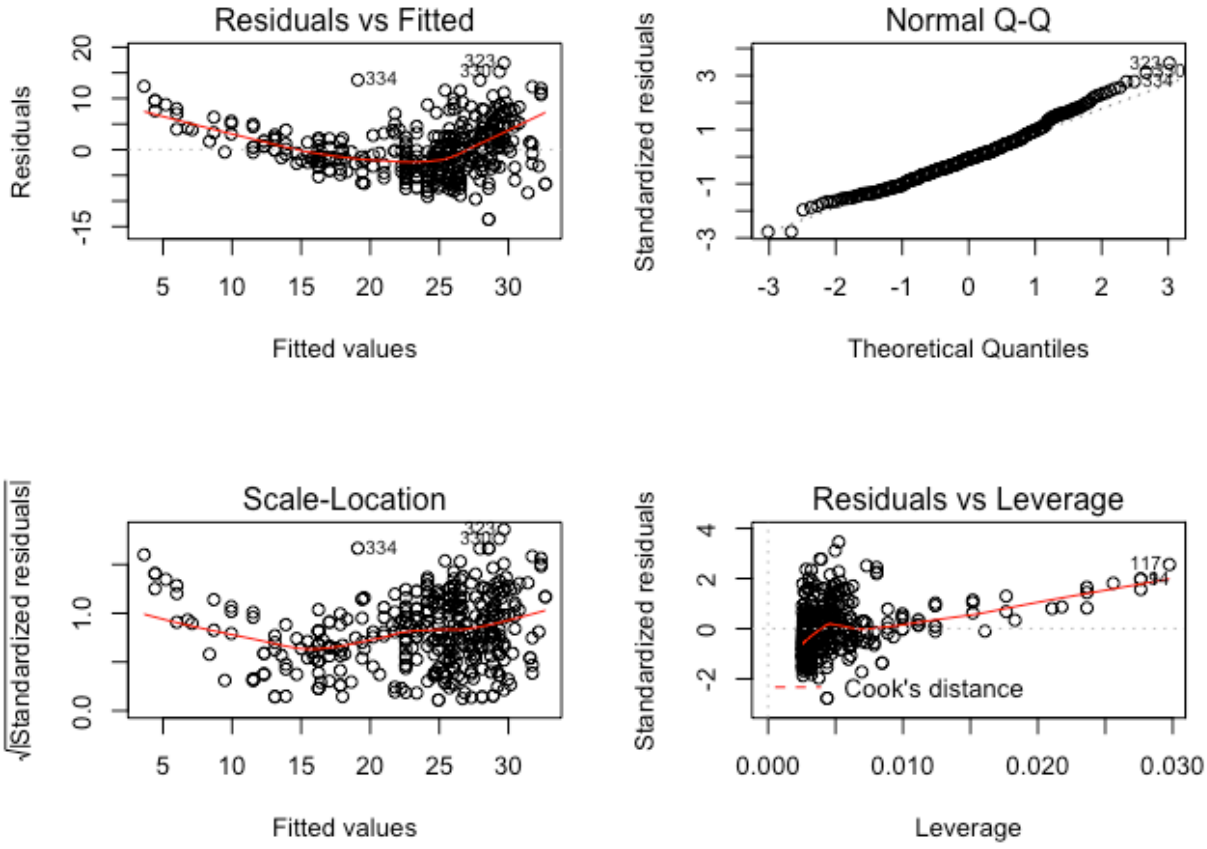
CODE:

```
plot(Auto$horsepower, Auto$mpg, xlab = "horsepower", ylab = "mpg")
abline(fit1, col = "red")
```

OUTPUT:



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



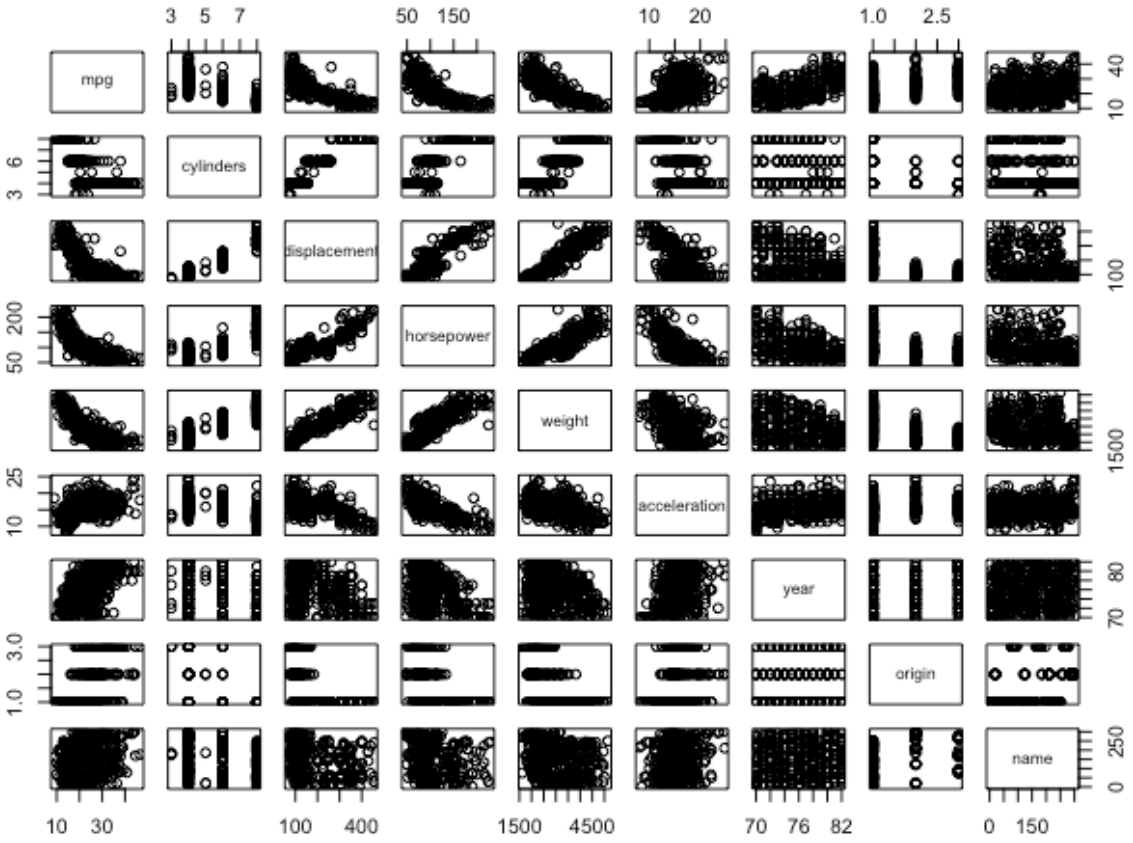
The plot "residuals VS fitted values" shows it is not a linearity in the data. The plot "standardized residuals VS leverage" shows there are some outliers points.

2. 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.

CODE: pairs(Auto)

OUTPUT:



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

CODE:

```
names(Auto)
cor(Auto[1:8])
```

OUTPUT:

```
> cor(Auto[1:8])
                     mpg  cylinders displacement horsepower     weight acceleration       year
mpg           1.0000000 -0.7776175   -0.8051269 -0.7784268 -0.8322442    0.4233285  0.5805410
cylinders    -0.7776175  1.0000000    0.9508233  0.8429834  0.8975273   -0.5046834 -0.3456474
displacement -0.8051269  0.9508233    1.0000000  0.8972570  0.9329944   -0.5438005 -0.3698552
horsepower   -0.7784268  0.8429834    0.8972570  1.0000000  0.8645377   -0.6891955 -0.4163615
weight       -0.8322442  0.8975273    0.9329944  0.8645377  1.0000000   -0.4168392 -0.3091199
acceleration  0.4233285 -0.5046834   -0.5438005 -0.6891955 -0.4168392    1.0000000  0.2903161
year          0.5805410 -0.3456474   -0.3698552 -0.4163615 -0.3091199    0.2903161  1.0000000
origin        0.5652088 -0.5689316   -0.6145351 -0.4551715 -0.5850054    0.2127458  0.1815277
                 origin
mpg           0.5652088
cylinders    -0.5689316
displacement -0.6145351
horsepower   -0.4551715
weight       -0.5850054
acceleration  0.2127458
year          0.1815277
origin        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. Comment on the output. For instance:

CODE:

```
fit2 = lm(mpg ~ . -name, Auto)
summary(fit2)
```

OUTPUT:

```
~/Desktop/CS465/Lab/Lab2/Lab2_Code/

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

Residual standard error: 3.328 on 384 degrees of freedom
Multiple R-squared:  0.8215,    Adjusted R-squared:  0.8182
F-statistic: 252.4 on 7 and 384 DF,  p-value: < 2.2e-16
```

(1) Is there a relationship between the predictors and the response?

Yes. To do a hypothesis test, set H0 : B1 = 0 and H1 : B1 != 0. We know the p-value is very small, so we will reject the hypothesis, the predictor and the response have a relationship.

(2) Which predictors appear to have a statistically significant relationship to the response?

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

(3) What does the coefficient for the year variable suggest?

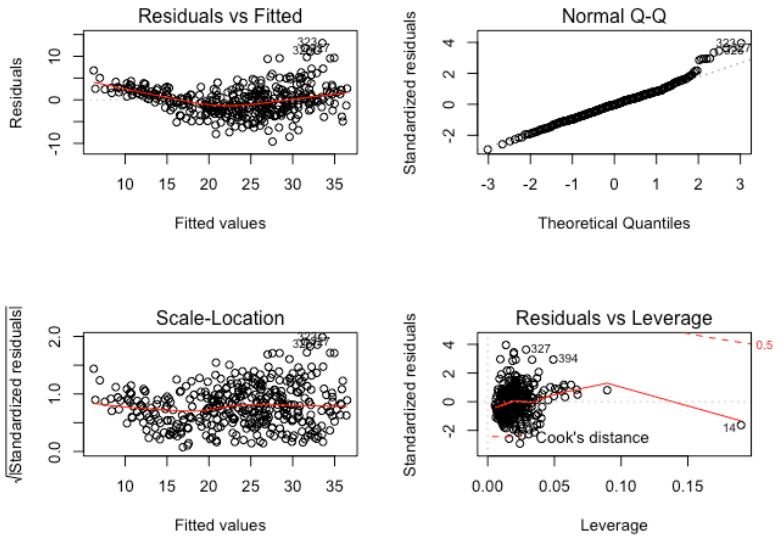
The coefficient of the year variable is **0.75** which means as the year increase 1, mpg will increase 0.75. **I would say automobiles will be more economic year by year.**

(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?

CODE:

```
par(mfrow = c(2, 2))
plot(fit2)
```

OUTPUT:



The plot "residuals VS fitted values" shows it is not a linearity in the data. The plot "standardized residuals VS leverage" shows there are some outliers points.

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

```
fit3 = lm(mpg ~ displacement * horsepower+displacement * weight, Auto); summary(fit3)
```

OUTPUT:

```
Residuals:
    Min      1Q  Median      3Q     Max
-11.305  -2.149  -0.387   1.872  16.447

Coefficients:
                          Estimate Std. Error t value Pr(>|t|)
(Intercept)              5.640e+01  2.007e+00  28.107  < 2e-16 ***
displacement            -7.864e-02  1.072e-02  -7.337 1.30e-12 ***
horsepower              -1.724e-01  2.760e-02  -6.247 1.10e-09 ***
weight                  -4.473e-03  1.347e-03  -3.320 0.000986 ***
displacement:horsepower  3.849e-04  9.128e-05   4.217 3.09e-05 ***
displacement:weight      6.739e-06  4.468e-06   1.508 0.132326
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.87 on 386 degrees of freedom
Multiple R-squared:  0.7573,    Adjusted R-squared:  0.7542
F-statistic: 240.9 on 5 and 386 DF,  p-value: < 2.2e-16
```

From the p-values, we can see that the interaction between displacement and horsepower is statistically significant, but the interaction between displacement and weight is not.

(f) Try a few different transformations of the variables, such as log(X), X, X .
Comment on your findings.
CODE:

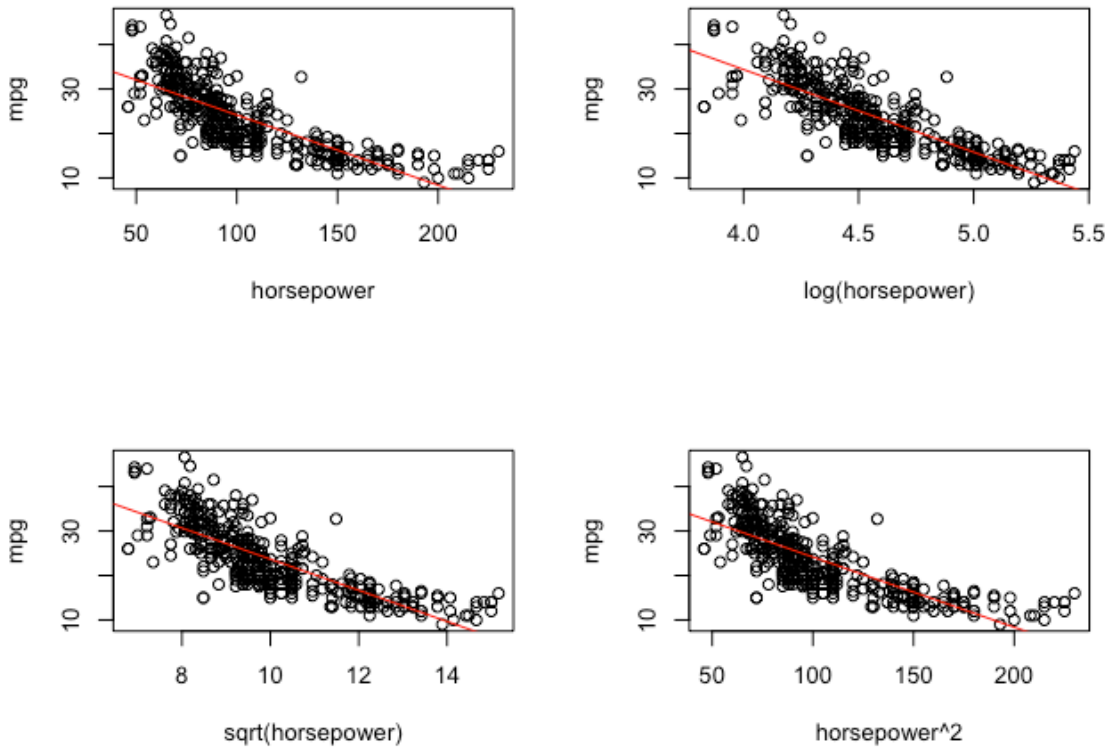
```
par(mfrow = c(2, 2))
plot(mpg ~ horsepower, Auto)
abline(fit1, col = "red")

fit_log = lm(mpg ~ log(horsepower), Auto)
plot(mpg ~ log(horsepower), Auto)
abline(fit_log, col = "red")

fit_sqrt = lm(mpg ~ sqrt(horsepower), Auto)
plot(mpg ~ sqrt(horsepower), Auto)
abline(fit_sqrt, col = "red")

fit_squa = lm(mpg ~ (horsepower)^2, Auto)
plot(mpg ~ (horsepower)^2, data = Auto, xlab = "horsepower^2")
abline(fit_squa, col = "red")
```

OUTPUT:



<u>It seems like the sort and ^2 don't change the plot a lot but the log transformation gives the most linear looking plot.</u>

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

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

CODE:

```
fit4 = lm(Sales ~ Price + Urban + US, data = Carseats)
summary(fit4)
```

OUTPUT:

```
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!

Coefficient of Price: -0.054456 (Prince increases, Sales decrease)
Coefficient of Urban: -0.021916 (Urban is yes, Sales decrease)
Coefficient of US: 1.200573 (US is yes, Sales increase)

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

Y = 13.043469 - 0.054456*Xp - 0.021916*Xurban + 1.200573*Xus + e

Where Xurban = 1 means yes; Xurban = 0 means no. Xus = 1 means yes; Xus = 0 means no;

(d) For which of the predictors can you reject the null hypothesis H0 : βj = 0

I am going to reject null hypothesis for Price and US because P-value of them is so small.

(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.

CODE:

```
fit5 = lm(Sales ~ Price + US, data = Carseats)
summary(fit5)
```

OUTPUT:

```
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?

The R^2 for the (e) model is better than the R^2 for the (a) model.

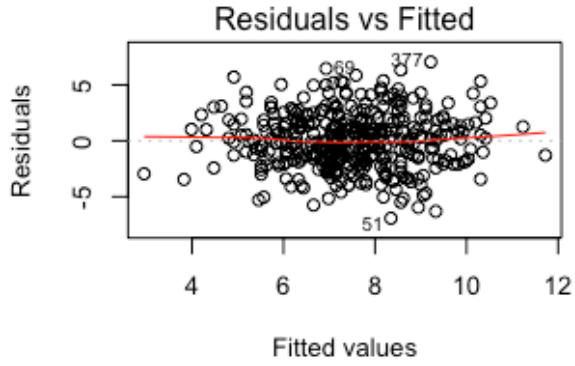
(g) Using the model from (e), obtain 95% confidence intervals for the coefficient(s).

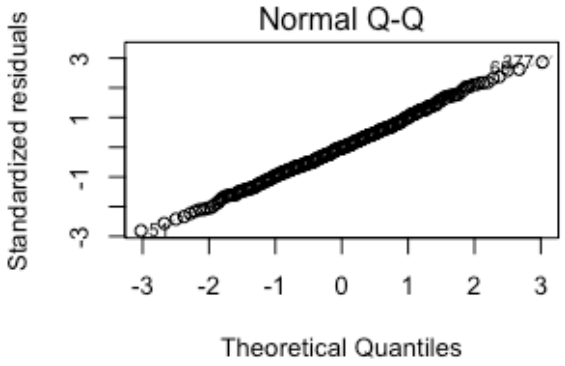
CODE&OUTPUT:

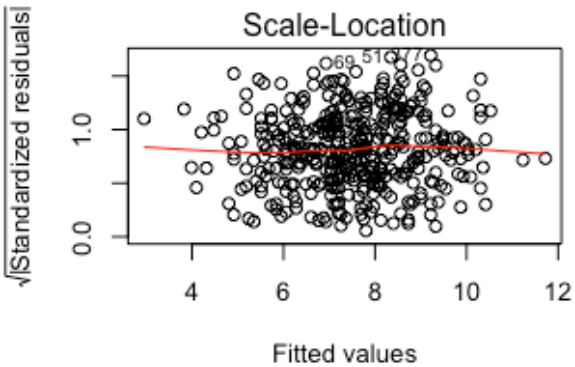
```
> confint(fit5, level = 0.95)
                  2.5 %      97.5 %
(Intercept) 11.79032020 14.27126531
Price       -0.06475984 -0.04419543
USYes        0.69151957  1.70776632
```

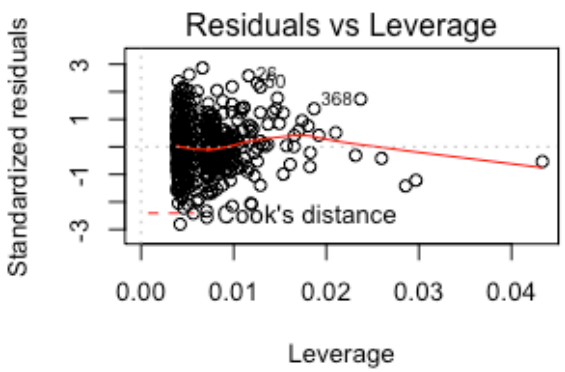
(h) Is there evidence of outliers or high leverage observations in the model from (e)?

According to the Residuals VS Leverage, there are a few outliers points.









4. In this problem we will investigate the t-statistic for the null hypothesis H0 : β = 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 β, the standard error of this coefficient estimate, and the t- statistic and p-value associated with the null hypothesis H0 : β = 0. Comment on these results.

CODE:

```
###Q4###
set.seed(1)
x = rnorm(100)
y=2*x+rnorm (100)
fit6 = lm(y ~ x + 0)
summary(fit6)
```

OUTPUT:

```
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
```

<u>According to the output, β = 1.9939, the standard error is 0.1065, the t is 18.73, the p-value is very small, we can reject the H0.</u>

(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 H0 : β = 0. Comment on these results.

CODE:

```
fit7 = lm(x ~ y + 0)
summary(fit7)
```

OUTPUT:

```
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
```

<u>According to the output, β = 0.39111, the standard error is 0.02089, the t is 18.73, the p-value is very small, we can reject the H0.</u>

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

The t values and p-value are same from (a) and (b). They are same line actually.

(d) Show algebraically, and confirm numerically in R

CODE:

```
len = length(x)
t = sqrt(len - 1)*(x %*% y)/sqrt(sum(x^2) * sum(y^2) - (x %*% y)^2)
print(as.numeric(t))
```

OUTPUT:

```
> print(as.numeric(t))
[1] 18.72593
>
```

t = 18.72593

(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 totally same if we replace x and y in the formal.

(f) In R, show that when regression is performed with an intercept, the t-statistic for H0 : β1 = 0. is the same for the regression of y onto x as it is for the regression of x onto y.
y onto x OUTPUT:

```
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
```

x onto y OUTPUT:

```
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 t-statistic for y onto x and x onto y are both equal to 18.56.

END

ALL CODE ARE ON MY GITHUB:

https://github.com/arthurmjt/CS465_Introduction-to-Statistical-Learning.git