Linear Regression

Contents

3.7.8

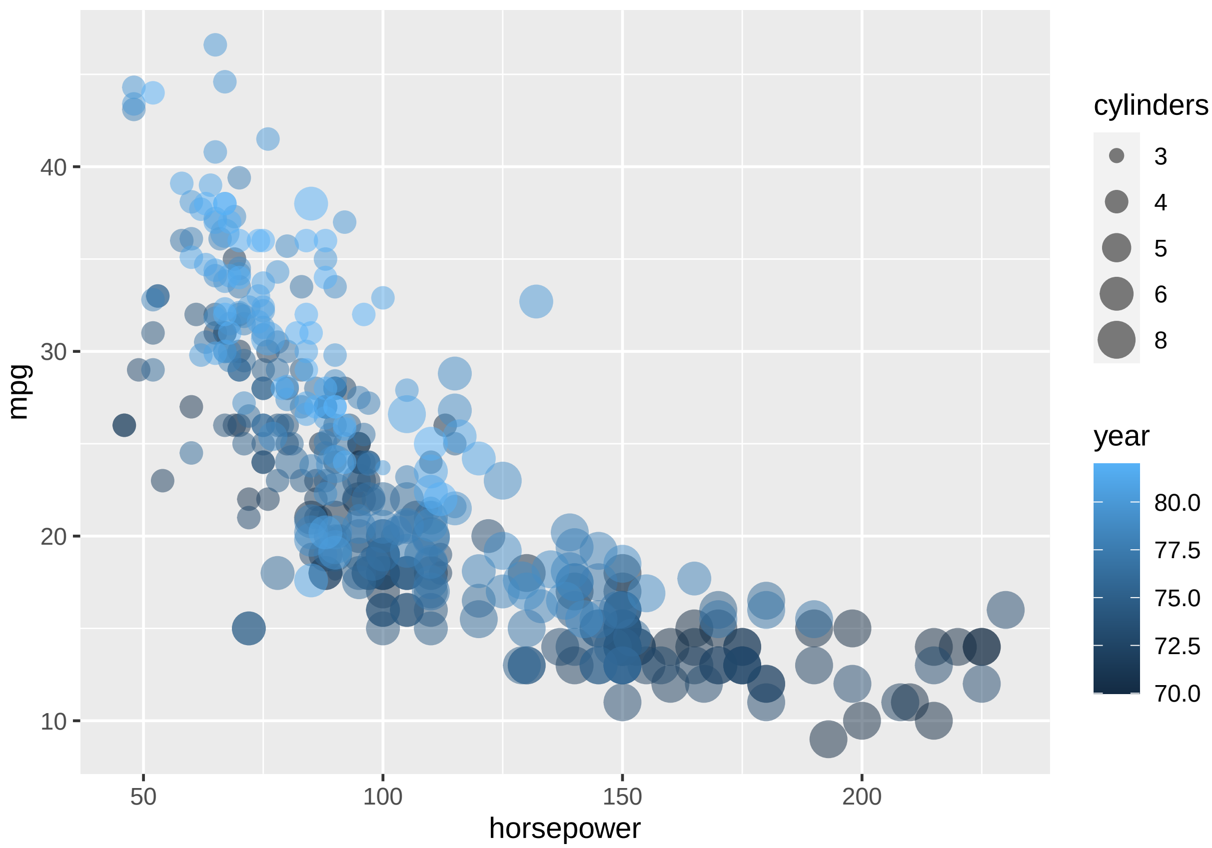
Loading the required data

```
data1 <- Auto
data1$cylinders <- as.factor(data1$cylinders)
data1$origin <- as.factor(data1$origin)
data1$name <- as.character(data1$name)
kable(head(data1, 10))
```

mpg	cylinders	displacement	horsepower	weight	acceleration	year	origin	name
18	8	307	130	3504	12.0	70	1	chevrolet chevelle malibu
15	8	350	165	3693	11.5	70	1	buick skylark 320
18	8	318	150	3436	11.0	70	1	plymouth satellite
16	8	304	150	3433	12.0	70	1	amc rebel sst
17	8	302	140	3449	10.5	70	1	ford torino
15	8	429	198	4341	10.0	70	1	ford galaxie 500
14	8	454	220	4354	9.0	70	1	chevrolet impala
14	8	440	215	4312	8.5	70	1	plymouth fury iii
14	8	455	225	4425	10.0	70	1	pontiac catalina
15	8	390	190	3850	8.5	70	1	amc ambassador dpl

(a)

```
ggplot(data1, aes(x = horsepower, y = mpg, color = year, size = cylinders))+
  geom_point(alpha = 0.5)
```



We see that there is some sort of linear relatin between horsepower and mpg. We may however need to transform the variables in order to make the relationship more linear.

Linear Model of mpg and horsepower

```
l1 <- lm(mpg ~ horsepower, data = data1)
summary(l1)
```

```
##
## Call:
## lm(formula = mpg ~ horsepower, data = data1)
##
## 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
```

Prediction for horse power with prediction interval

```r
predict(l1, newdata = data.frame(horsepower = c(98)), interval = 'prediction')
```

```
##        fit     lwr      upr
## 1 24.46708 14.8094 34.12476
```

Prediction for horse power with confidence interval

```r
predict(l1, newdata = data.frame(horsepower = c(98)), interval = 'confidence')
```

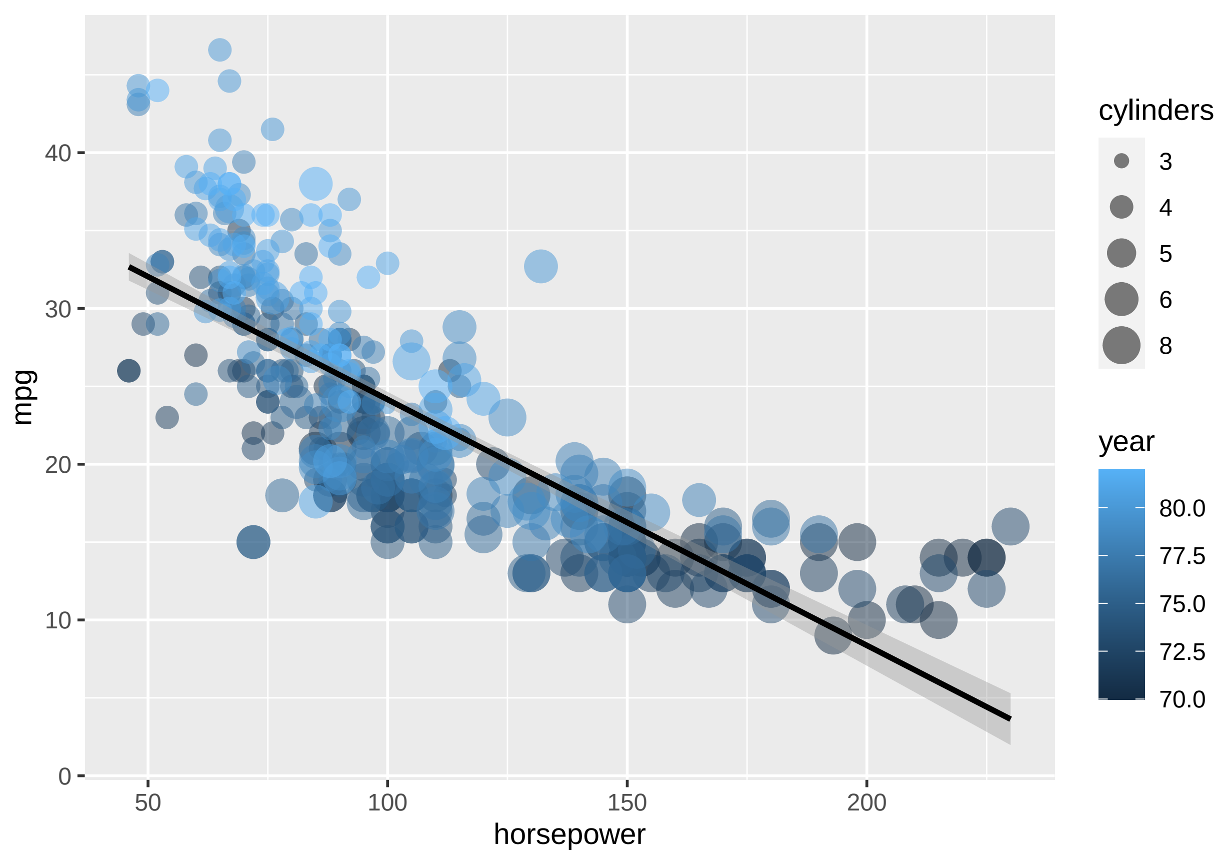
```
##        fit      lwr      upr
## 1 24.46708 23.97308 24.96108
```

Comments:/ 1. There is a relationship between mpg and horsepower.
2. The relationship is very statistically significant.
3. The relationship is negative.
4. The predicted mpg for a horsepower of 98 is 24.47. The 95% Prediction Interval is (14.81,34,12) and 95% Confidence Interval is (23.97, 24.96).

(b)

```r
ggplot(data = data1, aes(x = horsepower, y = mpg)) +
  geom_point(alpha = 0.5,aes(color = year, size = cylinders)) +
  geom_smooth(method = lm, color = 'black')
```

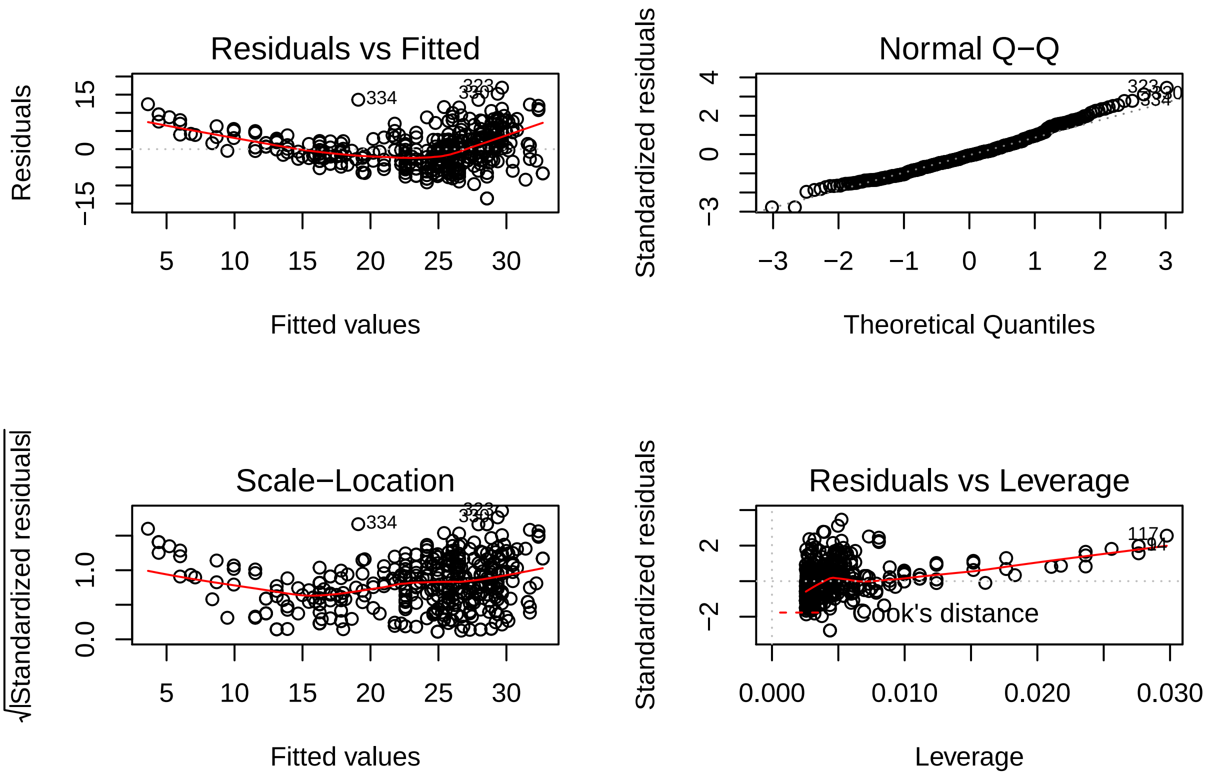
```
## `geom_smooth()` using formula 'y ~ x'
```



(c)

Diagnostic Plots

```
par(mfrow = c(2,2))
plot(l1)
```



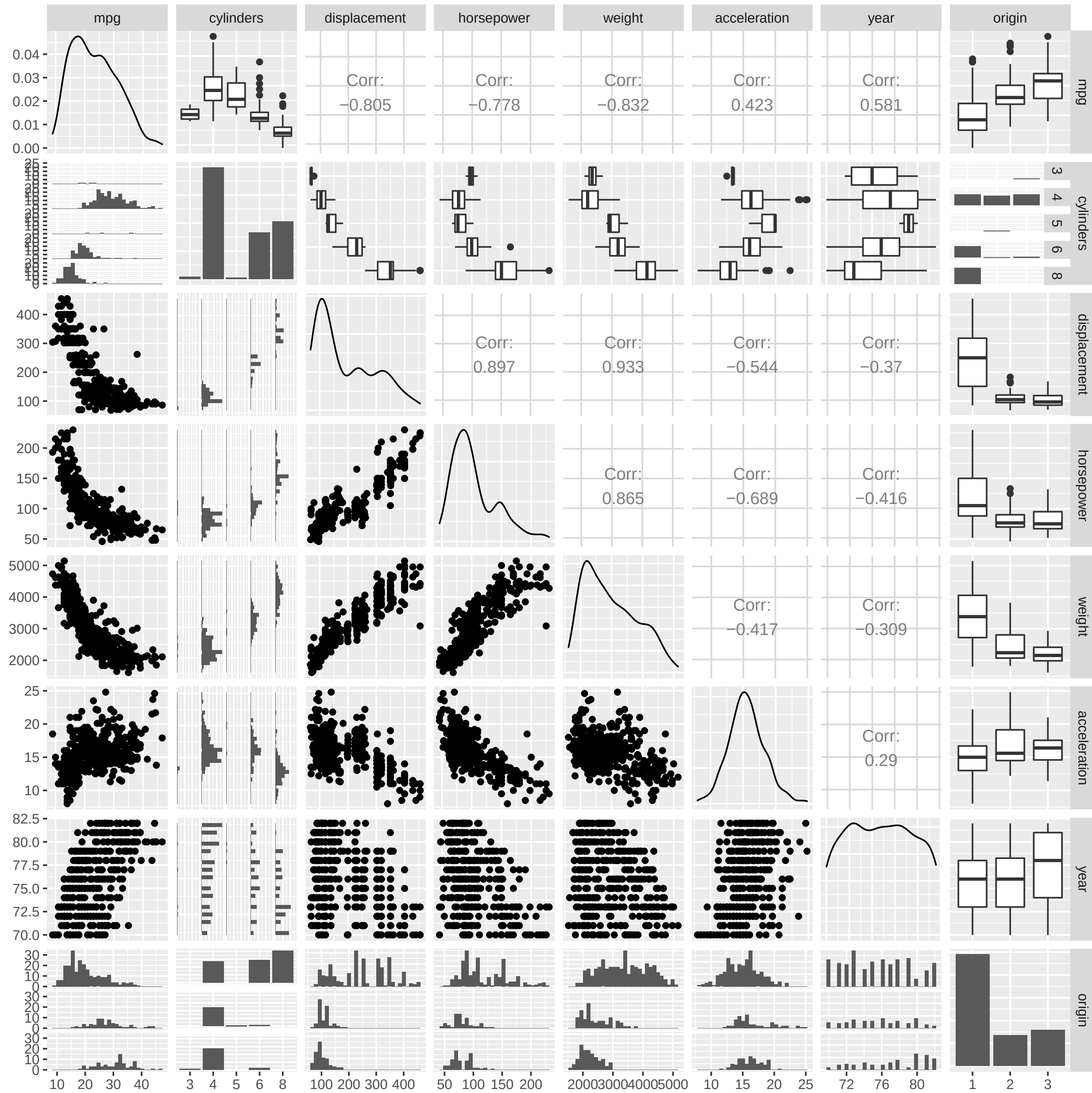
Comments:
1. The Residuals vs. Fitted Values plot shows that there might be non linear pattern in the residuals.
2. The Residuals vs Leverage plot shows that there are some bad leverage points that we might have to check.

3.7.9

We will be using the same dataset as above. ## (a) ## **Scatterplots for the data**

```
ggpairs(data1[-c(9)],progress = FALSE)
```

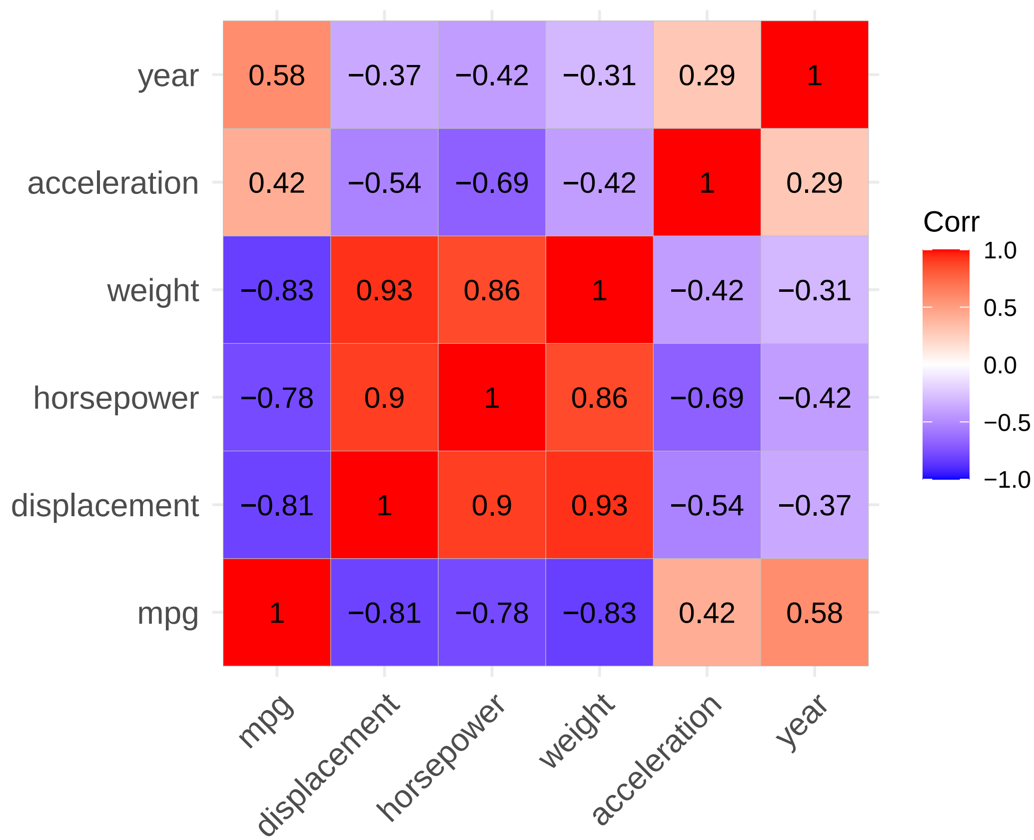
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



(b)

Correlation plot

```r
corr <- cor(data1[-c(2,8,9)])
ggcorrplot::ggcorrplot(corr, lab = T)
```



(c)

Multiple Linear Regression Model

```r
l2 <- lm(mpg ~ ., data = data1[-c(9)])
summary(l2)
```

```
##
## Call:
## lm(formula = mpg ~ ., data = data1[-c(9)])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.6797 -1.9373 -0.0678  1.6711 12.7756
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.208e+01  4.541e+00  -4.862 1.70e-06 ***
## cylinders4    6.722e+00  1.654e+00   4.064 5.85e-05 ***
## cylinders5    7.078e+00  2.516e+00   2.813  0.00516 **
## cylinders6    3.351e+00  1.824e+00   1.837  0.06701 .
## cylinders8    5.099e+00  2.109e+00   2.418  0.01607 *
## displacement  1.870e-02  7.222e-03   2.590  0.00997 **
## horsepower   -3.490e-02  1.323e-02  -2.639  0.00866 **
```

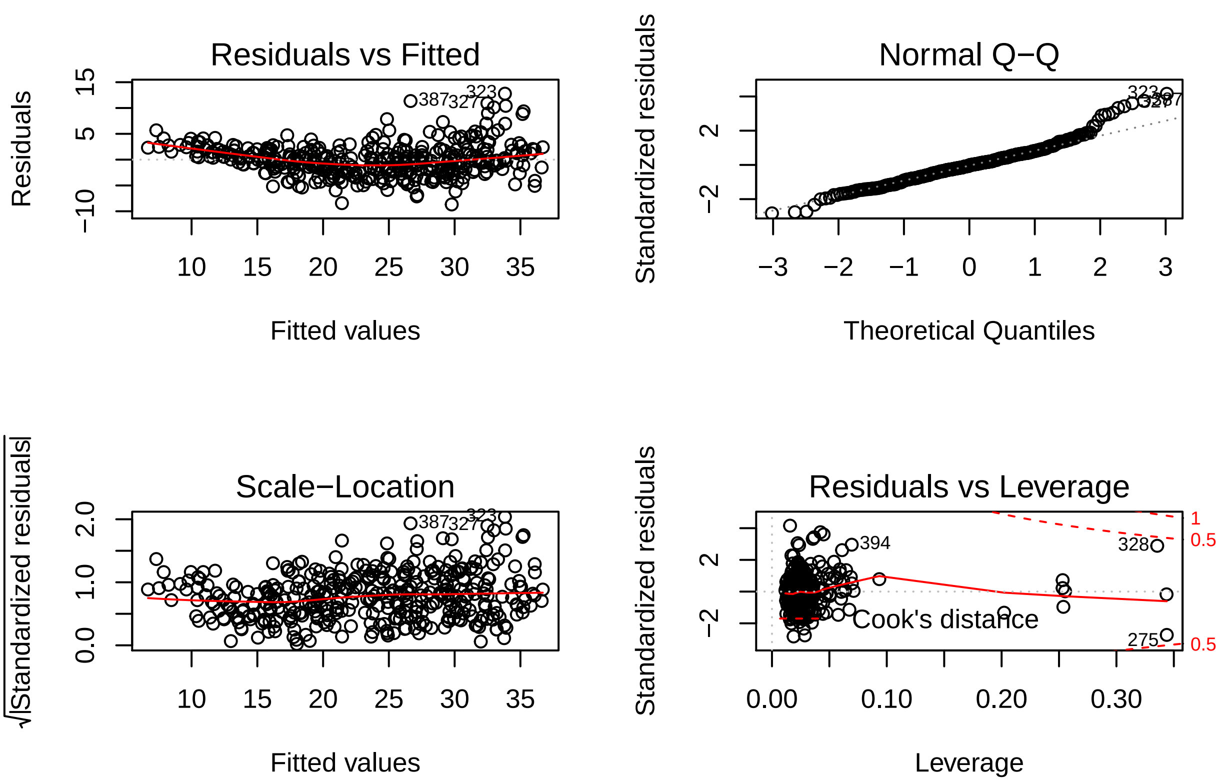
```
## weight       -5.780e-03  6.315e-04  -9.154  < 2e-16 ***
## acceleration  2.598e-02  9.304e-02   0.279  0.78021
## year          7.370e-01  4.892e-02  15.064  < 2e-16 ***
## origin2       1.764e+00  5.513e-01   3.200  0.00149 **
## origin3       2.617e+00  5.272e-01   4.964 1.04e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.098 on 380 degrees of freedom
## Multiple R-squared:  0.8469, Adjusted R-squared:  0.8425
## F-statistic: 191.1 on 11 and 380 DF,  p-value: < 2.2e-16
```

Comments:
1. Yes, thre is a relationship between the response and predictors.
2. Except cylinders6 and acceleration, all the other predictors are statistically significant.
3. A unit increase in the year of the car increases the mpg by 0.737 units on average, keeping everything else constant.

(d)

Diagnostic plots

```r
par(mfrow = c(2,2))
plot(l2)
```



Comments:
1. Looking at the Residuals vs Fitted Values plot, we still see slight nonlinearity.
2. There are some bad leverage points like observation 328, 394, 275.
3. There are some problems at the upper end of the Q-Q plot. This might also be due to the leverage points.

(e)

Regression Model with interaction terms

```
l3 <- lm(mpg ~ . + cylinders:horsepower + displacement:weight + year:origin, data = data1[-c(9)])
summary(l3)
```

```
##
## Call:
## lm(formula = mpg ~ . + cylinders:horsepower + displacement:weight +
##     year:origin, data = data1[-c(9)])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.9800 -1.5349 -0.0955  1.2713 13.3345
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -2.604e+01  1.972e+01  -1.321 0.187417
## cylinders4            3.396e+01  1.922e+01   1.767 0.078061 .
## cylinders5            6.109e+01  2.121e+01   2.881 0.004195 **
## cylinders6            2.856e+01  1.940e+01   1.472 0.141809
## cylinders8            2.719e+01  1.941e+01   1.401 0.162037
## displacement         -4.876e-02  1.324e-02  -3.683 0.000265 ***
## horsepower            1.971e-01  1.933e-01   1.019 0.308677
## weight               -7.937e-03  1.109e-03  -7.159 4.34e-12 ***
## acceleration         -9.047e-02  9.198e-02  -0.984 0.325973
## year                  6.265e-01  5.814e-02  10.776  < 2e-16 ***
## origin2              -3.239e+01  9.080e+00  -3.567 0.000408 ***
## origin3              -1.494e+01  8.445e+00  -1.769 0.077663 .
## cylinders4:horsepower -2.767e-01  1.932e-01  -1.432 0.152845
## cylinders5:horsepower -5.951e-01  2.199e-01  -2.706 0.007117 **
## cylinders6:horsepower -2.294e-01  1.941e-01  -1.182 0.237964
## cylinders8:horsepower -2.168e-01  1.940e-01  -1.118 0.264430
## displacement:weight    1.489e-05  3.400e-06   4.379 1.55e-05 ***
## year:origin2           4.374e-01  1.190e-01   3.677 0.000271 ***
## year:origin3           2.098e-01  1.083e-01   1.938 0.053406 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.745 on 373 degrees of freedom
## Multiple R-squared:  0.882,  Adjusted R-squared:  0.8763
## F-statistic: 154.9 on 18 and 373 DF,  p-value: < 2.2e-16
```

Among the interaction terms, the interaction between displacement and weight, year and origin2 and cylinders5 and horsepower are statistically significant. The others can be excluded.

3.7.10

We will be using the Carseats data.

```
data2 <- Carseats
kable(head(data2,10))
```

Sales	CompPrice	Income	Advertising	Population	Price	ShelveLoc	Age	Education	Urban	US
9.50	138	73	11	276	120	Bad	42	17	Yes	Yes
11.22	111	48	16	260	83	Good	65	10	Yes	Yes
10.06	113	35	10	269	80	Medium	59	12	Yes	Yes
7.40	117	100	4	466	97	Medium	55	14	Yes	Yes
4.15	141	64	3	340	128	Bad	38	13	Yes	No
10.81	124	113	13	501	72	Bad	78	16	No	Yes
6.63	115	105	0	45	108	Medium	71	15	Yes	No
11.85	136	81	15	425	120	Good	67	10	Yes	Yes
6.54	132	110	0	108	124	Medium	76	10	No	No
4.69	132	113	0	131	124	Medium	76	17	No	Yes

(a)

Multiple Linear Regression to fit Sales with Price, Urban, US

```
l4 <- lm(Sales ~ Price + Urban + US, data = data2)
summary(l4)
```

```
##
## Call:
## lm(formula = Sales ~ Price + Urban + US, data = data2)
##
## 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)

Price: A unit increase in price of the carseats decreases the Sales by -0.054 units on average, all else constant.

Urban: An urban store has a sale 0f 0.022 lower than a non-urban store on average, all else constant.
US: A US store has a sale of 1.3 higher than a non-US store on average, all else constant.

(c)

Mathematically, the model is:

$$Sales = \beta_0 + \beta_1 Price + \beta_2 UrbanYes + \beta_3 USYes$$

(d)

We can reject the null hypothesis of $H_0 : \beta_j = 0$ for UrbanYes Variable. The p-value is higher so we faile to reject the null and hence the variable is not statistically significant.

(e)

Reduced Model

```r
l5 <- lm(Sales ~ Price + US, data = data2)
summary(l5)
```

```
##
## Call:
## lm(formula = Sales ~ Price + US, data = data2)
##
## 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)

The model are not a very good fit to the data, as the R-squared statistics is very low(around 0.23). We will also check the RSE below.

```r
summary(l5)$sigma
```

```
## [1] 2.469397
```

From the RSE, we see that ther model is not a perfect fit.

(g)

Confidence Interval for coefficients.

```r
confint(l5)
```

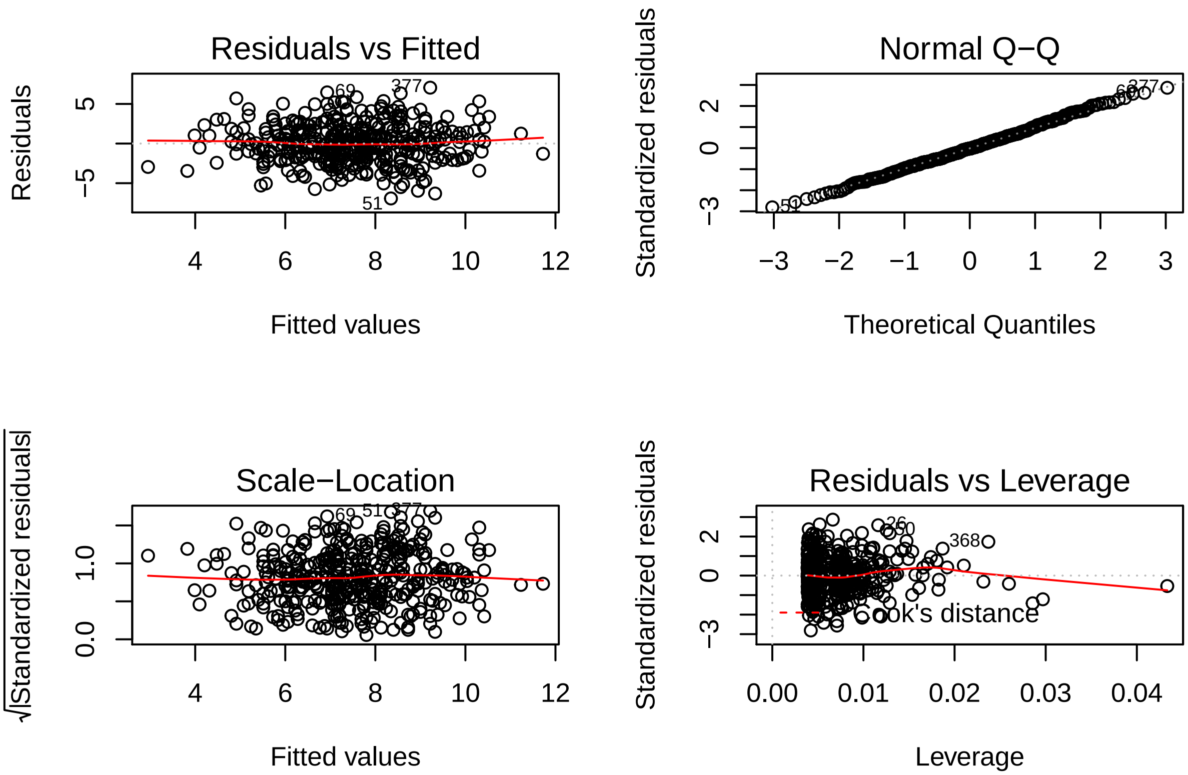
```
##                   2.5 %      97.5 %
## (Intercept) 11.79032020 14.27126531
## Price       -0.06475984 -0.04419543
## USYes        0.69151957  1.70776632
```

The above output gives the confidence Interval.

(h)

Diagnostic Plots

```r
par(mfrow = c(2,2))
plot(l5)
```



Yes, there are some points which might have high leverage points. Namely, 26, 368, 377, 69 and 51. We must check these observations.

3.7.11

We will be generating our own dataset in this section.

Generating the dataset

```
set.seed(1)
x <- rnorm(100)
y <- 2*x + rnorm(100)
data3 <- data.frame(y,x)
kable(head(data3, 10))
```

y	x
-1.8732743	-0.6264538
0.4094025	0.1836433
-2.5821789	-0.8356286
3.3485904	1.5952808
0.0044309	0.3295078
0.1263505	-0.8204684
1.6915656	0.4874291
2.3868236	0.7383247
1.5357481	0.5757814
1.0713993	-0.3053884

(a)

Linear Regression y onto x without an intercept

```r
l6 <- lm(y ~ x + 0)
summary(l6)
```

```
##
## 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
```

We can see the coeff. estimate, std. error, t-value and p-value in the summary above. We see that the coeff x is statistically significant.

(b)

Linear Regression x onto y without intercept

```r
l7 <- lm(x ~ y + 0)
summary(l7)
```

```
##
## 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
```

We can see the coeff. estimate, std. error, t-value and p-value in the summary above. We see that the coeff y is statistically significant.

(c)

In both the above regression models, the R-squared value is the same. That means these to variables explain the same amount of variation found in the data.

(d)

```r
t <- sqrt(length(x) - 1) * sum(x*y) / sqrt(sum(x^2)*sum(y^2) - (sum(x*y))^2)
print(paste0('T-statistic: ',t))
```

```
## [1] "T-statistic: 18.7259319374486"
```

Hence, the given formula gives the T-statistic.

(e)

Yes, the test statistic is the same. Changing the order of regression changes the definition of x and y in the formula, but the result obtained is the same.

(f)

Regression of y onto x with intercept

```r
l8 <- lm(y ~ x)
summary(l8)
```

```
##
## 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
```

Regression of x onto y

```r
l9 <- lm(x ~ y)
summary(l9)
```

```
##
## 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
```

As we can see, the test statistic is the same.

3.7.13

In this section, we will be simulating the dataset.

(a)

```r
set.seed(1)
x <- rnorm(100)
head(x,5)
```

```
## [1] -0.6264538  0.1836433 -0.8356286  1.5952808  0.3295078
```

(b)

```
eps <- rnorm(100,0,0.25)
head(eps,5)
```

```
## [1] -0.15509167  0.01052897 -0.22773041  0.03950719 -0.16364616
```

(c)

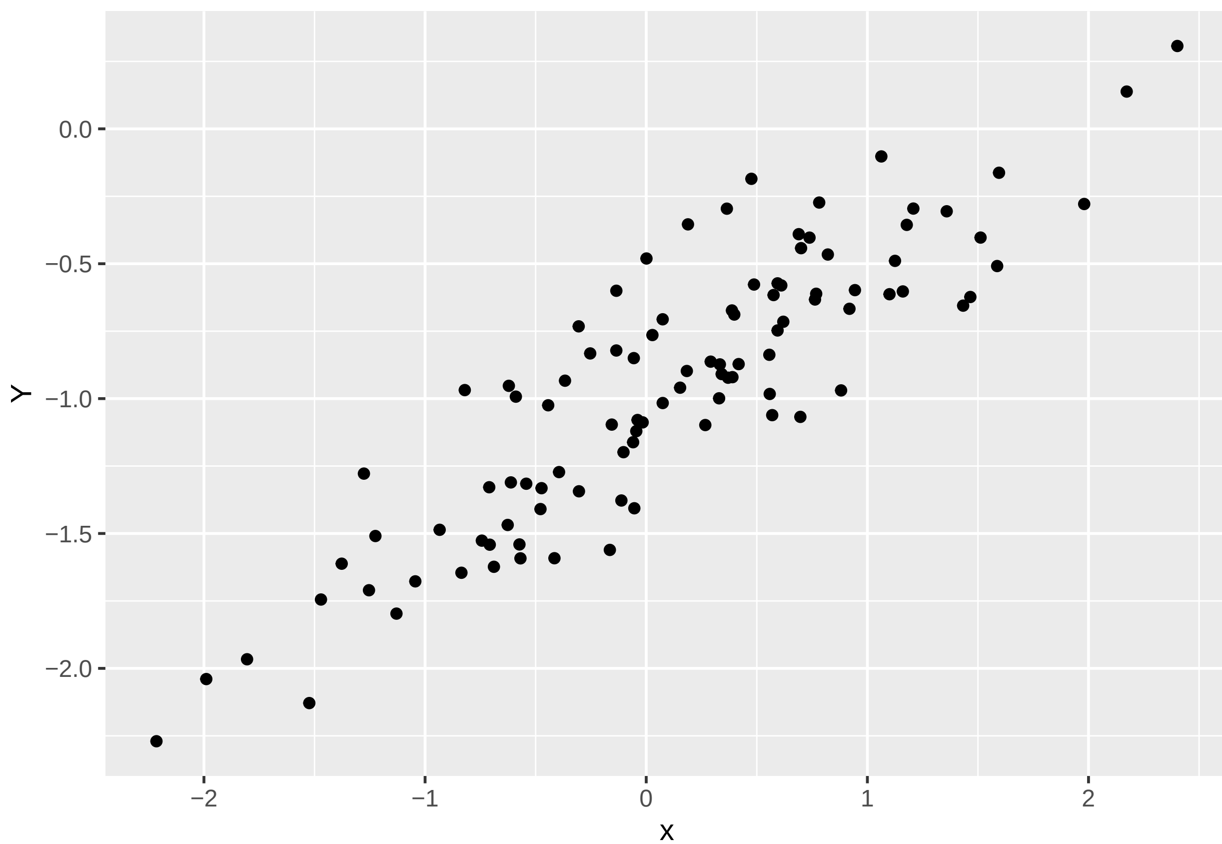
```
Y <- -1 + 0.5*x + eps
head(Y,5)
```

```
## [1] -1.4683186 -0.8976494 -1.6455447 -0.1628524 -0.9988923
```

The length of the vector Y is 100. The $\beta_0 = -1$ and $\beta_1 = 0.5$

(d)

```
ggplot(data = data.frame(x,Y),aes(x = x, y = Y))+
  geom_point()
```



There is a linear relationship between x and Y.

(e)

Creating the Linear Model

```r
l10 <- lm(Y~x)
summary(l10)
```

```
##
## Call:
## lm(formula = Y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.46921 -0.15344 -0.03487  0.13485  0.58654
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.00942    0.02425  -41.63   <2e-16 ***
## x            0.49973    0.02693   18.56   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2407 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
```

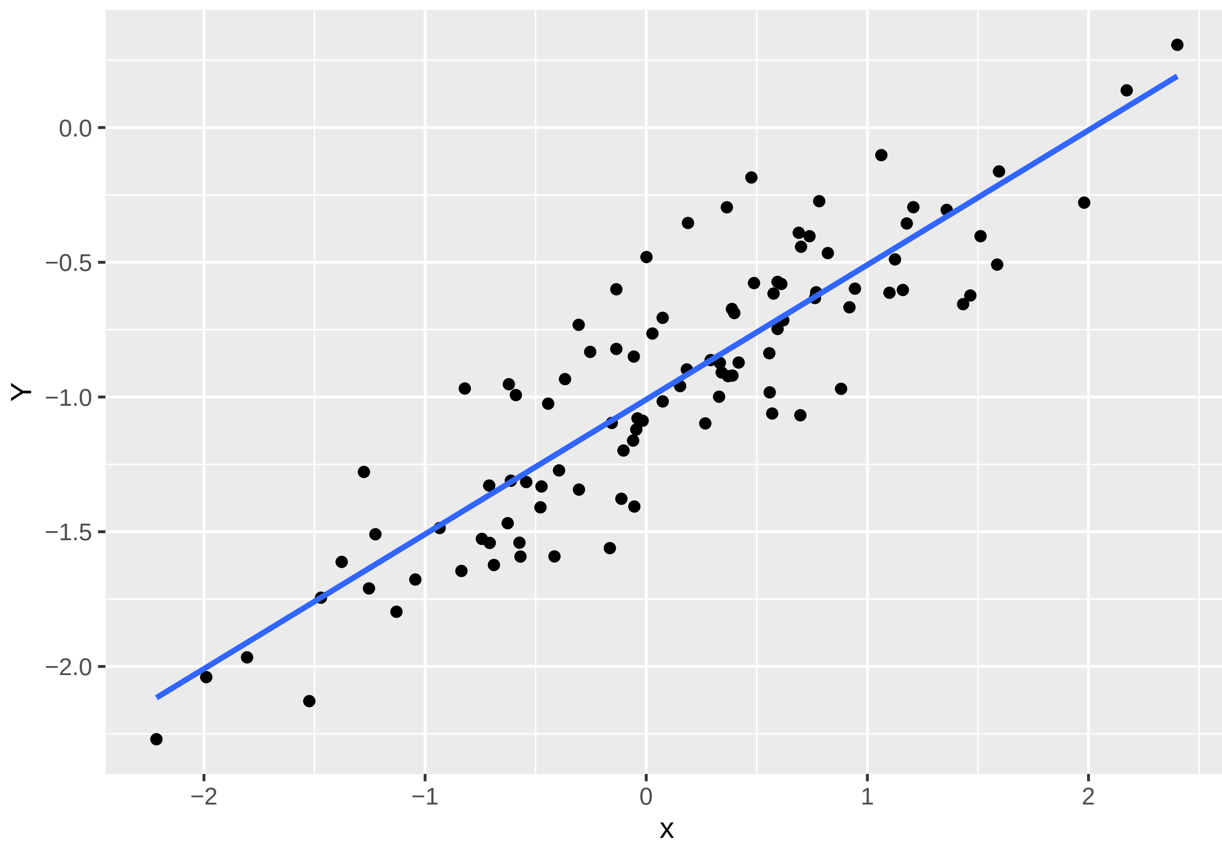
Here, $\hat{\beta}_0 = -1.0092$ and $\hat{\beta}_1 = 0.49973$.

The estimated coefficient is nearly equal to the actual coefficients.

(f)

```r
ggplot(data = data.frame(x,Y),aes(x = x, y = Y))+
  geom_point() +
  geom_smooth(method = lm, se = F)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



(g)

```
l11 <- lm(Y ~ x + I(x^2))
summary(l11)
```

```
##
## Call:
## lm(formula = Y ~ x + I(x^2))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.4913 -0.1563 -0.0322  0.1451  0.5675
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.98582    0.02941 -33.516   <2e-16 ***
## x            0.50429    0.02700  18.680   <2e-16 ***
## I(x^2)      -0.02973    0.02119  -1.403    0.164
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2395 on 97 degrees of freedom
## Multiple R-squared:  0.7828, Adjusted R-squared:  0.7784
## F-statistic: 174.8 on 2 and 97 DF,  p-value: < 2.2e-16
```

There is no evidence that the polynomial model improves the model because the x^2 variable is not statistically significant.

(h)

```r
set.seed(1)
x1 <- rnorm(100)
eps1 <- rnorm(100,0,0.005)
Y1 <- -1 + 0.5*x1 + eps1
head(data.frame(Y1,x1),5)
```

```
##           Y1         x1
## 1 -1.3163287 -0.6264538
## 2 -0.9079678  0.1836433
## 3 -1.4223689 -0.8356286
## 4 -0.2015695  1.5952808
## 5 -0.8385190  0.3295078
```

The length of the vector Y is 100. The $\beta_0 = -1$ and $\beta_1 = 0.5$

Creating the Linear Model

```r
l12 <- lm(Y1~x1)
summary(l12)
```

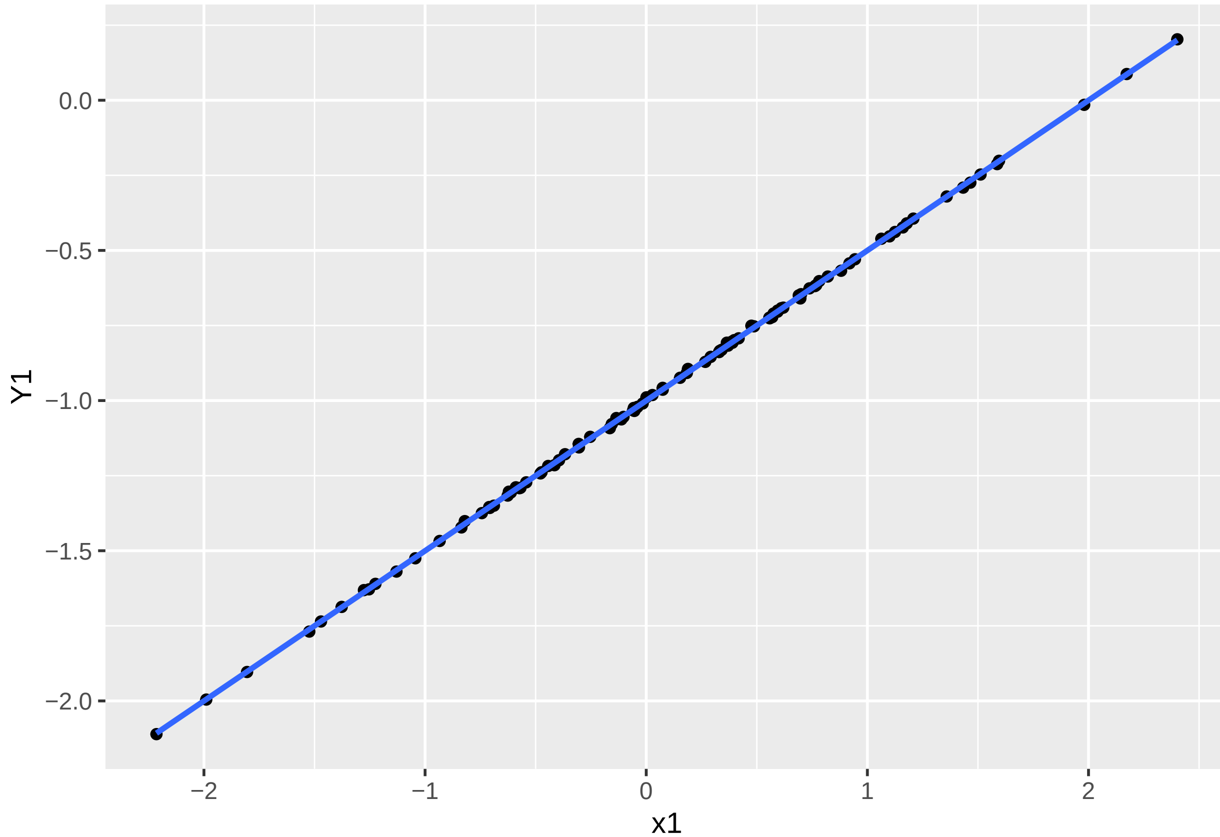
```
##
## Call:
## lm(formula = Y1 ~ x1)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0093842 -0.0030688 -0.0006975  0.0026970  0.0117309
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.0001885  0.0004849 -2062.5   <2e-16 ***
## x1           0.4999947  0.0005386   928.3   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.004814 on 98 degrees of freedom
## Multiple R-squared:  0.9999, Adjusted R-squared:  0.9999
## F-statistic: 8.617e+05 on 1 and 98 DF,  p-value: < 2.2e-16
```

Here, $\hat{\beta}_0 = -1.0001885$ and $\hat{\beta}_1 = 0.4999947$.

The estimated coefficient is nearly equal to the actual coefficients.

```r
ggplot(data = data.frame(x1,Y1),aes(x = x1, y = Y1))+
  geom_point() +
  geom_smooth(method = lm, se = F)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Creating the polynomial Model

```r
l13 <- lm(Y1 ~ x1 + I(x1^2))
summary(l11)
```

```
##
## Call:
## lm(formula = Y ~ x + I(x^2))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.4913 -0.1563 -0.0322  0.1451  0.5675
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.98582    0.02941 -33.516   <2e-16 ***
## x            0.50429    0.02700  18.680   <2e-16 ***
## I(x^2)      -0.02973    0.02119  -1.403    0.164
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2395 on 97 degrees of freedom
## Multiple R-squared:  0.7828, Adjusted R-squared:  0.7784
## F-statistic: 174.8 on 2 and 97 DF,  p-value: < 2.2e-16
```

There is no evidence that the polynomial model improves the model because the x^2 variable is not statistically significant.

{i}

```
set.seed(1)
x2 <- rnorm(100)
eps2 <- rnorm(100,0,0.5)
Y2 <- -1 + 0.5*x2 + eps2
head(data.frame(Y2,x2),5)
```

```
##           Y2         x2
## 1 -1.6234102 -0.6264538
## 2 -0.8871204  0.1836433
## 3 -1.8732751 -0.8356286
## 4 -0.1233452  1.5952808
## 5 -1.1625384  0.3295078
```

The length of the vector Y2 is 100. The $\beta_0 = -1$ and $\beta_1 = 0.5$

Creating the Linear Model

```
l14 <- lm(Y2~x2)
summary(l14)
```

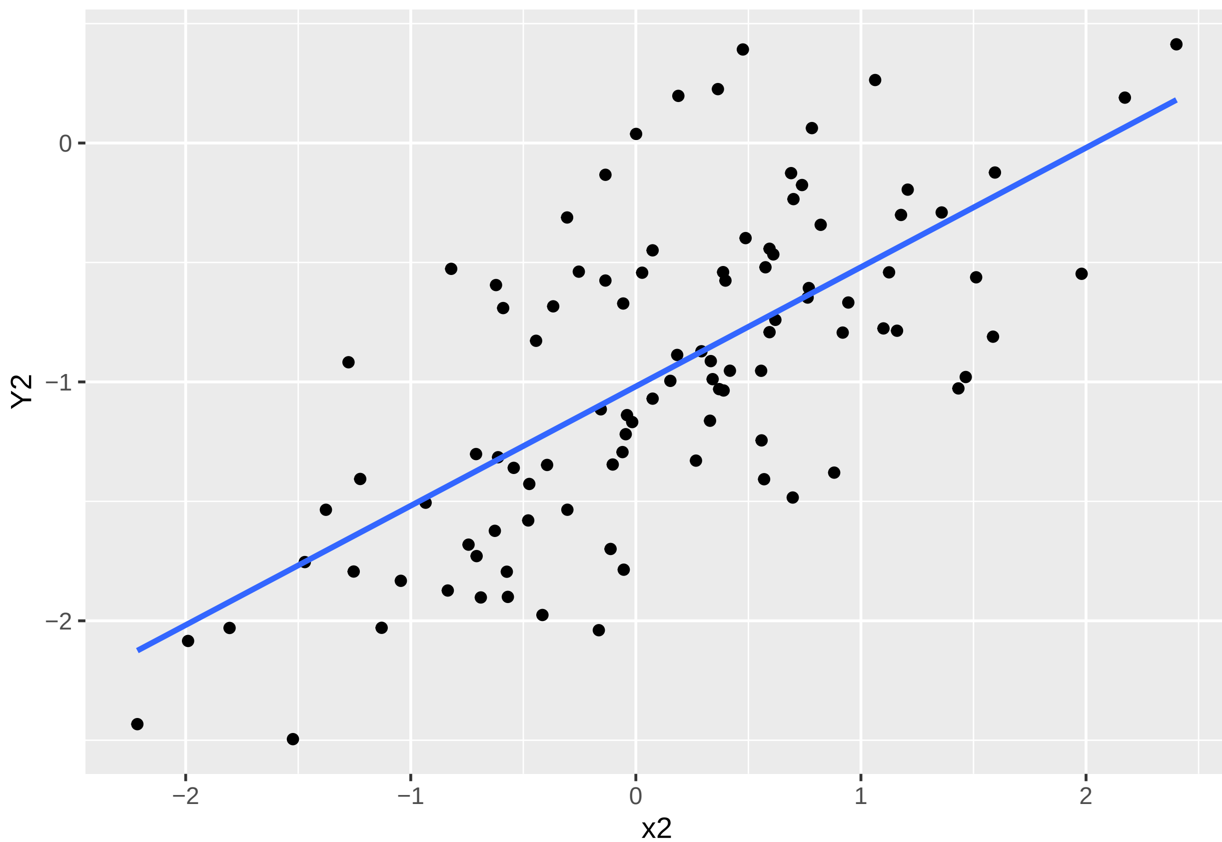
```
##
## Call:
## lm(formula = Y2 ~ x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.93842 -0.30688 -0.06975  0.26970  1.17309
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.01885    0.04849 -21.010  < 2e-16 ***
## x2           0.49947    0.05386   9.273 4.58e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4814 on 98 degrees of freedom
## Multiple R-squared:  0.4674, Adjusted R-squared:  0.4619
## F-statistic: 85.99 on 1 and 98 DF,  p-value: 4.583e-15
```

Here, $\hat{\beta}_0 = -1.01885$ and $\hat{\beta}_1 = 0.49947$.

The estimated coefficient is nearly equal to the actual coefficients.

```
ggplot(data = data.frame(x2,Y2),aes(x = x2, y = Y2))+
  geom_point() +
  geom_smooth(method = lm, se = F)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Creating the polynomial Model

```r
l15<- lm(Y2 ~ x2 + I(x2^2))
summary(l15)
```

```
##
## Call:
## lm(formula = Y2 ~ x2 + I(x2^2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.98252 -0.31270 -0.06441  0.29014  1.13500
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.97164    0.05883 -16.517  < 2e-16 ***
## x2           0.50858    0.05399   9.420  2.4e-15 ***
## I(x2^2)     -0.05946    0.04238  -1.403    0.164
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.479 on 97 degrees of freedom
## Multiple R-squared:  0.4779, Adjusted R-squared:  0.4672
## F-statistic:  44.4 on 2 and 97 DF,  p-value: 2.038e-14
```

There is no evidence that the polynomial model improves the model because the x^2 variable is not statistically significant.

(j)

confidence interval for original data

```
confint(l10)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.0575402 -0.9613061
## x            0.4462897  0.5531801
```

confidence interval for noisier data

```
confint(l14)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.1150804 -0.9226122
## x2           0.3925794  0.6063602
```

confidence interval for less noisier data

```
confint(l12)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.0011508 -0.9992261
## x1           0.4989258  0.5010636
```

The lower the noise, the smaller the confidence interval.

3.7.14

(a)

```
x1 <- runif(100)
x2 <- 0.5*x1 + rnorm(100)/10
y <- 2 + 2*x1 + 0.3*x2 + rnorm(100)
```

The true form of the linear model is:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2$$

The true regression coeffiecients are:
1. $\beta_0 = 2$
2. $\beta_1 = 2$
3. $\beta_2 = 0.3$

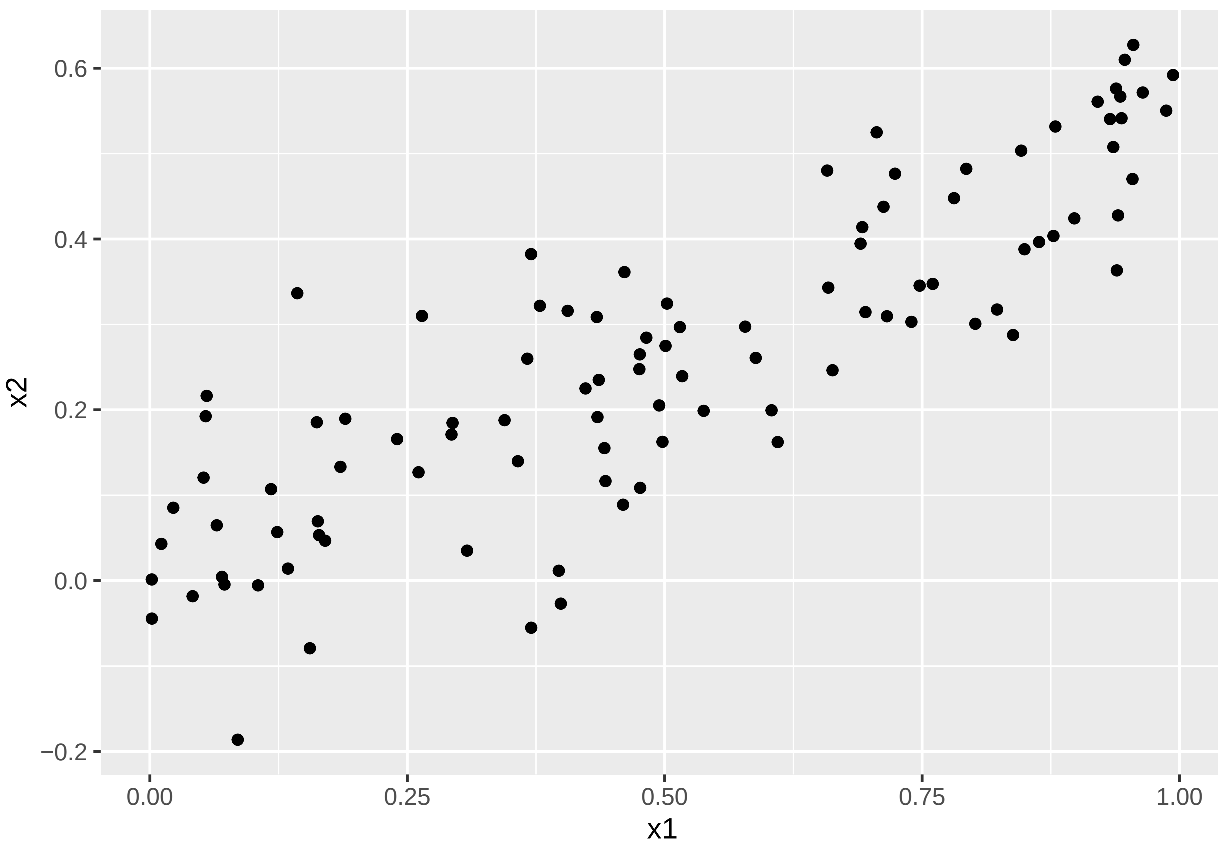
(b)

```r
print(paste0('correlation between x1 and x2 ',round(cor(x1,x2),4)))
```

```
## [1] "correlation between x1 and x2 0.8514"
```

x_1 and x_2 have highly positive correlation.

```r
ggplot(data.frame(x1,x2), aes(x = x1, y = x2))+
  geom_point()
```



(c)

Linear Model

```r
l16 <- lm(y ~ x1 + x2)
summary(l16)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.91594 -0.57900 -0.01157  0.68557  1.97436
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.0533     0.2056   9.989   <2e-16 ***
## x1            1.6336     0.6656   2.454   0.0159 *
## x2            0.5588     1.0914   0.512   0.6098
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.057 on 97 degrees of freedom
## Multiple R-squared:  0.2399, Adjusted R-squared:  0.2242
## F-statistic: 15.31 on 2 and 97 DF,  p-value: 1.668e-06
```

The variable x2 is not statistically significant. We can reject the null hypothers $H_0 : \beta_0 = 0$ but not the other one.

(d)

Linear Model using only x1.

```r
l17 <- lm(y ~ x1)
summary(l17)
```

```
##
## Call:
## lm(formula = y ~ x1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.99977 -0.53567 -0.01094  0.71087  1.93670
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.0535     0.2048   10.03  < 2e-16 ***
## x1            1.9237     0.3479    5.53 2.65e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.053 on 98 degrees of freedom
## Multiple R-squared:  0.2379, Adjusted R-squared:  0.2301
## F-statistic: 30.58 on 1 and 98 DF,  p-value: 2.655e-07
```

Yes we can reject the null hypothesis.

(e)

Linear Model using only x2.

```r
l18 <- lm(y ~ x2)
summary(l18)
```

```
##
## Call:
## lm(formula = y ~ x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.06128 -0.67275 -0.02065  0.77313  2.44900
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.2796     0.1884  12.101  < 2e-16 ***
## x2            2.8392     0.5870   4.837 4.91e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.084 on 98 degrees of freedom
## Multiple R-squared:  0.1927, Adjusted R-squared:  0.1845
## F-statistic: 23.39 on 1 and 98 DF,  p-value: 4.905e-06
```

Yes, we can reject the null hypothesis.

(f)

Yes, the results in (c) - (e) contradicts each other. As we see, in (c) we showed that x2 was not significant but in (e) we see that it is significant.

(g)

```r
x1 <- c(x1,0.1)
x2 <- c(x2, 0.8)
y <- c(y,6)

l19 <- lm(y ~ x1 + x2)
summary(l19)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.80004 -0.68053 -0.07887  0.73521  2.19254
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.1192     0.2091  10.134   <2e-16 ***
```

```
## x1            0.6925     0.5606   1.235   0.2197
## x2            2.1966     0.8907   2.466   0.0154 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.084 on 98 degrees of freedom
## Multiple R-squared:  0.2386, Adjusted R-squared:  0.2231
## F-statistic: 15.36 on 2 and 98 DF,  p-value: 1.578e-06
```

The variable x1 is not statistically significant. We can reject the null hypothers $H_0 : \beta_1 = 0$ but not the other one.

(d)

Linear Model using only x1.

```
l20 <- lm(y ~ x1)
summary(l20)
```

```
##
## Call:
## lm(formula = y ~ x1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0436 -0.5743 -0.0156  0.6860  3.6523
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.1715     0.2133  10.180  < 2e-16 ***
## x1            1.7623     0.3641   4.841 4.77e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.112 on 99 degrees of freedom
## Multiple R-squared:  0.1914, Adjusted R-squared:  0.1832
## F-statistic: 23.43 on 1 and 99 DF,  p-value: 4.77e-06
```

Yes we can reject the null hypothesis.

(e)

Linear Model using only x2.

```
l21 <- lm(y ~ x2)
summary(l21)
```

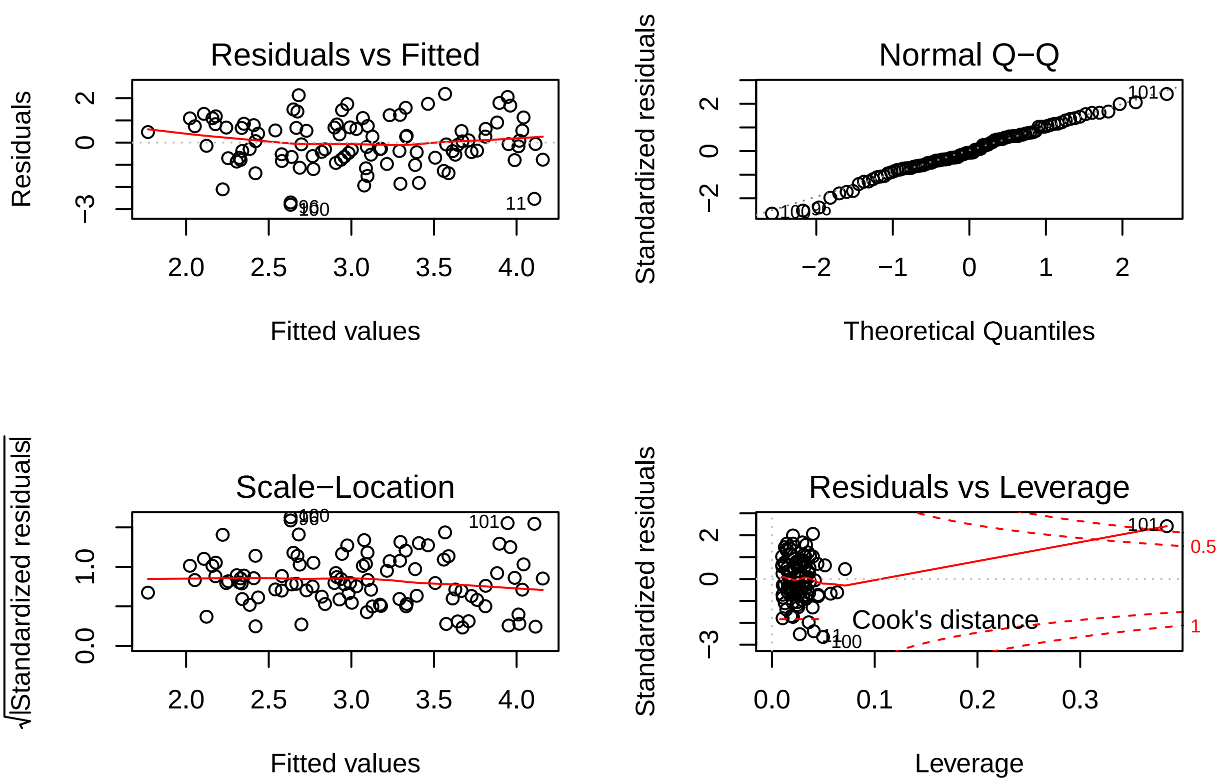
```
##
## Call:
## lm(formula = y ~ x2)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.06488 -0.74915 -0.07163  0.79722  2.41474
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.2380     0.1861  12.023  < 2e-16 ***
## x2            3.0480     0.5656   5.389 4.81e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.087 on 99 degrees of freedom
## Multiple R-squared:  0.2268, Adjusted R-squared:  0.219
## F-statistic: 29.04 on 1 and 99 DF,  p-value: 4.814e-07
```

Yes, we can reject the null hypothesis.

The result are the same as in previous question. There is a contradiction.

```
par(mfrow = c(2,2))
plot(l19)
```



Yes, it seems that the point is a outlier and a bad leverage point.