Chapter 3 Exercises

From *ISLR*

Jacob Zeiher

May 17, 2018

Contents

Conceptual

Problem 1

The p values given in Table 3.4 refer to the highest level of confidence with which we can reject the null hypothesis that $\beta_i = 0, i = 0, 1, 2, 3$ where β_i is the coefficient on intercept, TV, radio, and newspaper, respectively. From these p-values, we can conclude that intercept, TV, and radio all have a significant relationship with the outcome variables, holding the other variables constant. Since the coefficient on newspaper has a p-value of 0.8599 there is not a significant relationship between newspaper and the outcome variable.

Problem 2

The KNN classifier is a classification method while the KNN regression is a regression method. The KNN classifier makes a classification based on the classification of the K closest data points. Similarly, the KNN regression assigns a predicted value based on the average value of the K nearest data points. Hence, the outcome variable for the KNN classifier is categorical while the outcome variable for the KNN regression is quantitative.

Problem 3

Part a

We have

$$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 GPA + \hat{\beta}_2 IQ + \hat{\beta}_3 FEMALE + \hat{\beta}_4 (GPA \times IQ) + \hat{\beta}_5 (GPA \times FEMALE).$$

For males this equation becomes

$$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 GPA + \hat{\beta}_2 IQ + \hat{\beta}_4 (GPA \times IQ) = 50 + 20GPA + 0.07IQ + 0.01 (GPA \times IQ),$$

and for females it becomes

$$\hat{Y} = \left(\hat{\beta}_0 + \hat{\beta}_3\right) + \left(\hat{\beta}_1 + \hat{\beta}_5\right) GPA + \hat{\beta}_2 IQ + \hat{\beta}_4 (GPA \times IQ) = 85 + 10GPA + +0.07IQ + 0.01 (GPA \times IQ).$$

Hence, option ii is correct: *For a fixed value of IQ and GPA, males earn more on average than females provided that the GPA is high enough.*

Part b

Using the previous equation, we would have

$$\hat{Y} = 85 + 10(4) + 0.07(110) + 0.01(110 \times 4) = 131.7$$

Part c

False. To determine statistical significance, we need to compare $t = \frac{\hat{\beta}}{se\hat{\beta}}$ to a t-distribution with $n - 2$ degrees of freedom. We cannot determine statistical significance based solely on the *magnitude* of the coefficient. We must also take into consideration its estimated standard error.

Problem 4

Part a

Without knowing more details about the training data, it is difficult to know which training RSS is lower between linear or cubic. However, as the true relationship between X and Y is linear, we may expect the least squares line to be close to the true regression line, and consequently the RSS for the linear regression may be lower than for the cubic regression. Moreover, the training RSS for the cubic regression will be lower than the linear regression because adding additional regressors has to decrease training RSS.

Part b

If the additional predictors lead to overfitting, the testing RSS could be worse (higher) for the cubic regression fit

Part c

The cubic regression fit should produce a better RSS on the training set because it can adjust for the non-linearity.

Part d

Similar to training RSS, the cubic regression fit should produce a better RSS on the testing set because it can adjust for the non-linearity.

Problem 5

We have

$$\hat{y}_i = x_i \frac{\sum_{j=1}^n x_j y_j}{\sum_{k=1}^n x_k^2} = \sum_{j=1}^n \frac{x_j y_j x_i}{\sum_{k=1}^n x_k^2} = \sum_{j=1}^n \frac{x_j x_i}{\sum_{k=1}^n x_k^2} y_j,$$

so

$$a_j = \frac{x_i x_j}{\sum_{k=1}^n x_k^2}.$$

Problem 6

Note that $\bar{y} = \frac{1}{n}\sum y_i = \frac{1}{n}\sum(\hat{y}_i + \hat{\epsilon}_i)$. The estimates of $\hat{y}_i$ are given by

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i + \hat{\epsilon}_i.$$

Multiplying both sides by $\frac{1}{n}$ and summing from $i = 1, \ldots, n$, we get

$$\frac{1}{n}\hat{y}_i = \bar{y} = \frac{1}{n}\sum(\hat{\beta}_0 + \hat{\beta}_1 x_i + \hat{\epsilon}_i) = \hat{\beta}_0 + \hat{\beta}_1 \bar{x},$$

where the $\hat{\epsilon}_i$ sum to 0. Hence, $\bar{y} = \hat{\beta}_0 + \hat{\beta}_1\bar{x}$, so the regression equation passes through $(\bar{x}, \bar{y})$.

Problem 7

First, show that in simple linear regression,

$$\sum_{i=1}^n (y_i - \bar{y})^2 = \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \sum_{i=1}^n (\hat{y}_i - \bar{y})^2. \tag{1}$$

Obviously,

$$(y_i - \bar{y}) = (y_i + \hat{y}_i) + (\hat{y}_i - \bar{y}).$$

Summing the square of both sides over all observations,

$$\sum_{i=1}^n (y_i - \bar{y})^2 = \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \sum_{i=1}^n (\hat{y}_i - \bar{y})^2 + \sum_{i=1}^n 2(y_i - \hat{y}_i)(\hat{y}_i - \bar{y}).$$

Need to show

$$\sum_{i=1}^n 2(y_i - \hat{y}_i)(\hat{y}_i - \bar{y}) = 0.$$

In simple linear regression,

$$\hat{y}_i = \hat{\alpha} + \hat{\beta} x_i,$$
$$\bar{y} = \hat{\alpha} + \hat{\beta}\bar{x},$$

and

$$\hat{\beta} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}.$$

Then

$$\hat{y}_i - \bar{y} = (\hat{\alpha} + \hat{\beta}x_i) - (\hat{\alpha} + \hat{\beta}\bar{x}) = \hat{\beta}(x_i - \bar{x}),$$

and

$$y_i - \hat{y}_i = (y_i - \bar{y}) - (\hat{y}_i - \bar{y}) = (y_i - \bar{y}) - \hat{\beta}(x_i - \bar{x}).$$

Then we have

$$\begin{aligned}
\sum_{i=1}^{n} 2(\hat{y}_i - \bar{y})(y_i - \hat{y}_i) &= 2\hat{\beta}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \hat{y}_i) \\
&= 2\hat{\beta}\sum_{i=1}^{n}(x_i - \bar{x})[(y_i - \bar{y}) - \hat{\beta}(x_i - \bar{x})] \\
&= 2\hat{\beta}\left[\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}) - \sum_{i=1}^{n}\hat{\beta}(x_i - \bar{x})^2\right] \\
&= 2\hat{\beta}\left[\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}) - \sum_{i=1}^{n}\left[(x_i - \bar{x})^2\sum_{j=1}^{n}\frac{(x_j - \bar{x})(y_j - \bar{y})}{(x_j - \bar{x})^2}\right]\right] \\
&= 2\hat{\beta}\left[\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}) - \sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})\right] \\
&= 2\hat{\beta}(0) = 0.
\end{aligned}$$

Hence, (1) holds in simple linear regression.

Now need to show that

$$R^2 = \mathrm{Cov}^2(X, Y). \tag{2}$$

By (1) we have that

$$\begin{aligned}
R^2 &= \frac{TSS - RSS}{TSS} \qquad (3) \\
&= \frac{\sum_{i=1}^{n}(y_i - \bar{y})^2 - \sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \qquad (4) \\
&= \frac{\sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}. \qquad (5)
\end{aligned}$$

Since $\bar{y} = \hat{\alpha} + \hat{\beta}\bar{x}$, we have $\hat{\alpha} = \bar{y} - \hat{\beta}\bar{x}$. Then

$$
\begin{aligned}
\sum_{i=1}^{n}(\hat{y}_i-\bar{y})^2 &= \sum_{i=1}^{n}(\hat{\alpha}+\hat{\beta}x_i-\bar{y})^2 \\
&= \sum_{i=1}^{n}(\bar{y}-\hat{\beta}\bar{x}+\hat{\beta}x_i-\bar{y})^2 \\
&= \sum_{i=1}^{n}(\hat{\beta}(x_i-\bar{x})^2 \\
&= \hat{\beta}^2\sum_{i=1}^{n}(x_i-\bar{x})^2 \\
&= \frac{\left[\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})\right]^2\sum_{i=1}^{n}(x_i-\bar{x})^2}{\left[\sum_{i=1}^{n}(x_i-\bar{x})^2\right]^2} \\
&= \frac{\left[\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})\right]^2}{\sum_{i=1}^{n}(x_i-\bar{x})^2}.
\end{aligned}
$$

Then (5) becomes

$$
\begin{aligned}
R^2 &= \frac{\sum_{i=1}^{n}(\hat{y}_i-\bar{y})^2}{\sum_{i=1}^{n}(y_i-\bar{y})^2} \\
&= \frac{\left[\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})\right]^2}{\sum_{i=1}^{n}(x_i-\bar{x})^2\sum_{i=1}^{n}(y_i-\bar{y})^2} \\
&= \left[\frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i-\bar{y})^2}}\right]^2 \\
&= \mathrm{Cov}^2(X,Y).
\end{aligned}
$$

Applied

Problem 8

Part a

```r
#Import libraries
  library(ISLR)
#Fit regression
  lm.fit1 <- lm(mpg~horsepower, data=Auto)
  summary(lm.fit1)
```

```
##
## 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. Since an F-statistics of 599.7 on 1 and 390 degrees of freedom has p-value of <2.2e-16, we can reject the null hypothesis of no relationship between the response and predictor and conclude there is a relationship.

ii. The model has an R^2 of 0.6059, so we can conclude there is a pretty strong relationship between the response and predictors. Recall that the R^2 is interpreted as the amount of variation in the response variable (in this case mpg) that is explained by the model.

iii. Since the coefficient on horsepower is negative, the relationship between the predictor and response variable is negative. That is, as horsepower increase, mpg will decrease, on average.

iv.

```r
predict(lm.fit1,data.frame(horsepower=98),interval="confidence")
```

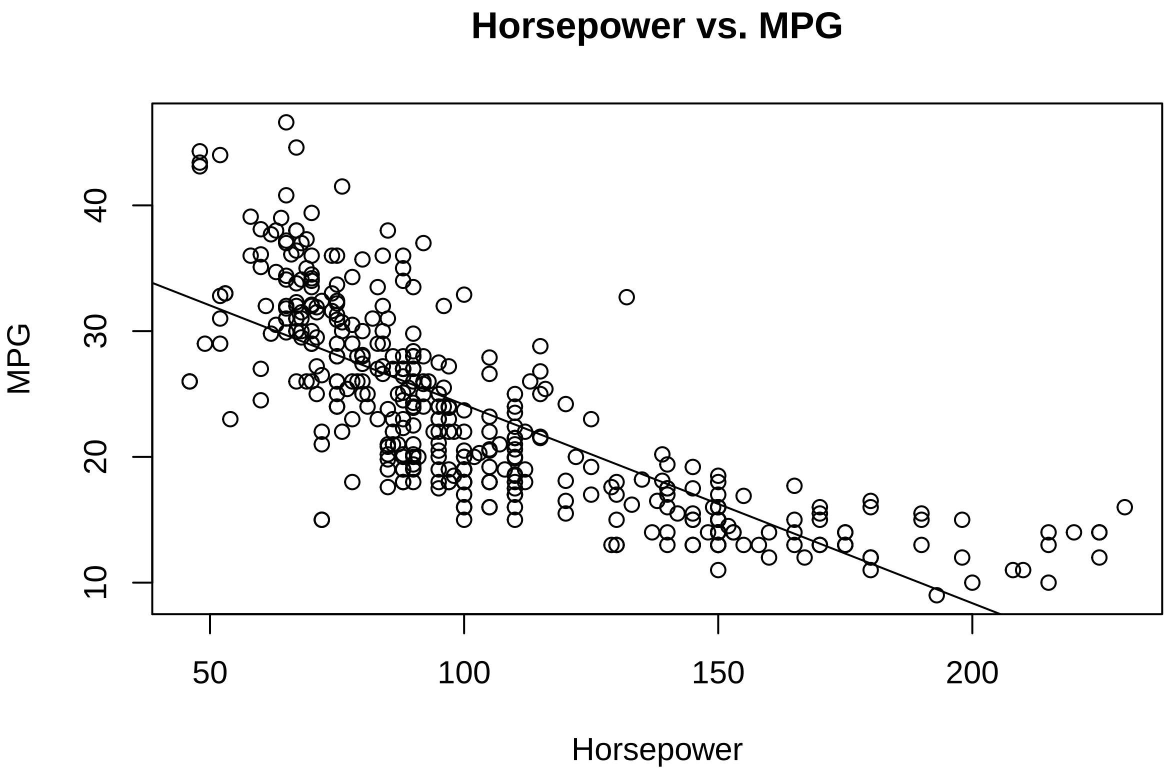
```
##        fit      lwr      upr
## 1 24.46708 23.97308 24.96108
```

```r
predict(lm.fit1,data.frame(horsepower=98),interval="prediction")
```

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

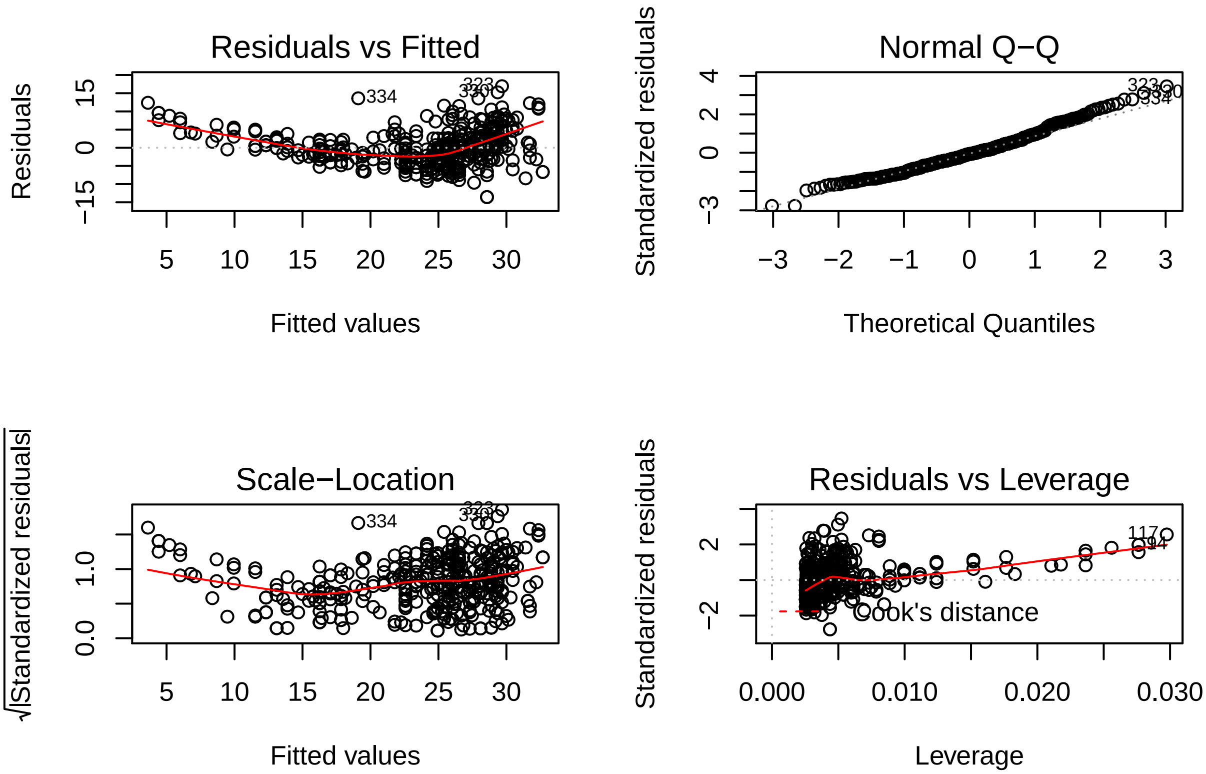
Part b

```r
#Plot regression with fit
  plot(Auto$horsepower,Auto$mpg,ylab="MPG",xlab="Horsepower",main="Horsepower vs. MPG")
  abline(lm.fit1)
```



Part c

```
#Plot regression diagnostics
  par(mfrow=c(2,2))
  plot(lm.fit1)
```

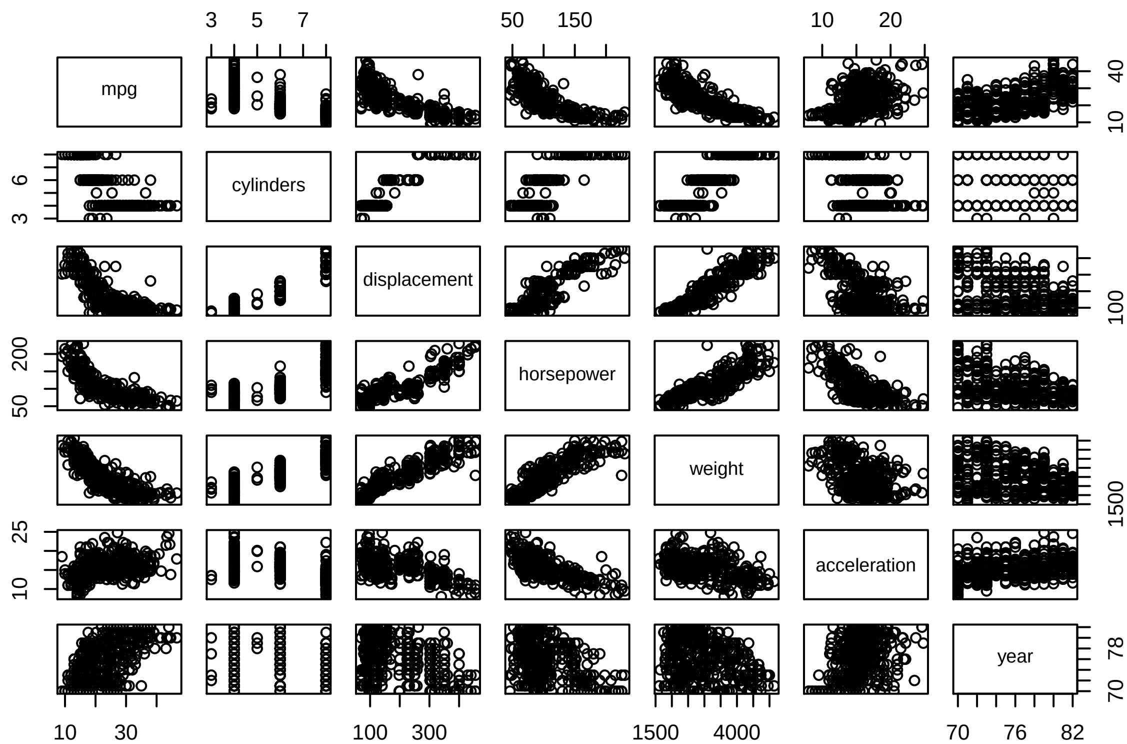


The plots of horsepower vs. mpg and the plot of residual vs fitted values both indicate there is some non-linearity in the data. The plot of standardized residuals vs leverage indicate there are some outliers and high leverage points.

Problem 9

Part a

```
#Import library
  library(ISLR)
#Produce scatterplot matrix
  pairs(Auto[1:7])
```



Part b

```r
#Produce correlation matrix
  cor(Auto[1:8])
```

```
##                     mpg  cylinders displacement horsepower     weight
## mpg           1.0000000 -0.7776175   -0.8051269 -0.7784268 -0.8322442
## cylinders    -0.7776175  1.0000000    0.9508233  0.8429834  0.8975273
## displacement -0.8051269  0.9508233    1.0000000  0.8972570  0.9329944
## horsepower   -0.7784268  0.8429834    0.8972570  1.0000000  0.8645377
## weight       -0.8322442  0.8975273    0.9329944  0.8645377  1.0000000
## acceleration  0.4233285 -0.5046834   -0.5438005 -0.6891955 -0.4168392
## year          0.5805410 -0.3456474   -0.3698552 -0.4163615 -0.3091199
## origin        0.5652088 -0.5689316   -0.6145351 -0.4551715 -0.5850054
##              acceleration       year     origin
## mpg             0.4233285  0.5805410  0.5652088
## cylinders      -0.5046834 -0.3456474 -0.5689316
## displacement   -0.5438005 -0.3698552 -0.6145351
## horsepower     -0.6891955 -0.4163615 -0.4551715
## weight         -0.4168392 -0.3091199 -0.5850054
## acceleration    1.0000000  0.2903161  0.2127458
## year            0.2903161  1.0000000  0.1815277
## origin          0.2127458  0.1815277  1.0000000
```

Part c

```r
#Convert origin variable into factor
  Auto$origin <- factor(Auto$origin, levels=c(1,2,3),labels=c("American","European","Japanese"))
```

```r
#Fit multiple linear regression model
  lm.fit1 <- lm(mpg~.-name, data=Auto)
  summary(lm.fit1)
```

```
##
## Call:
## lm(formula = mpg ~ . - name, data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.0095 -2.0785 -0.0982  1.9856 13.3608
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -1.795e+01  4.677e+00  -3.839 0.000145 ***
## cylinders      -4.897e-01  3.212e-01  -1.524 0.128215
## displacement    2.398e-02  7.653e-03   3.133 0.001863 **
## horsepower     -1.818e-02  1.371e-02  -1.326 0.185488
## weight         -6.710e-03  6.551e-04 -10.243  < 2e-16 ***
## acceleration    7.910e-02  9.822e-02   0.805 0.421101
## year            7.770e-01  5.178e-02  15.005  < 2e-16 ***
## originEuropean  2.630e+00  5.664e-01   4.643 4.72e-06 ***
## originJapanese  2.853e+00  5.527e-01   5.162 3.93e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.307 on 383 degrees of freedom
## Multiple R-squared:  0.8242, Adjusted R-squared:  0.8205
## F-statistic: 224.5 on 8 and 383 DF,  p-value: < 2.2e-16
```

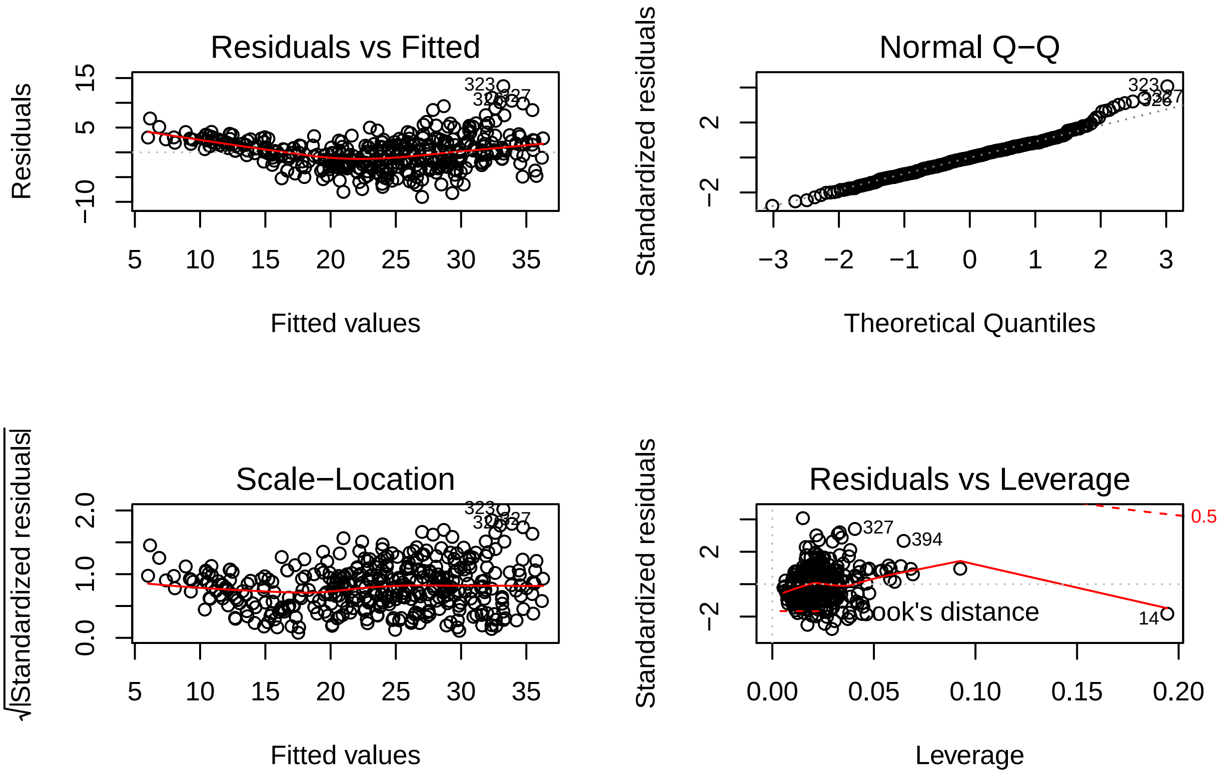
i. Since an F-statistics of 224.5 on 8 and 383 degrees of freedom has p-value of <2.2e-16, we can reject the null hypothesis of no relationship between the response and predictor and conclude there is a relationship.

ii. The displacement, weight, year, originEuropean, and originaJapanese variables appear to have a statistically significant relationship with the response variable (mpg). The cylinders, horsepower, and accelearation variables do not have statistically significant relationships with the response variable.

iii. The coefficient on the year variable is highly statistically significant and positive, indicating cars have become more fuel efficient over time. In particular, cars gain 0.75 mpg per year, on average.

Part d

```r
#Plot regression diagnostics
  par(mfrow=c(2,2))
  plot(lm.fit1)
```



The regression diagnostic plots indicate the presence of some outliers, in particular observations 323, 327, and 326. The plots also indicate the presence of a high leverage point in observation 14.

Part e

```
#Fit multiple linear regression model with interaction terms
  lm.fit2 <- lm(mpg~.-name+cylinders:displacement+cylinders:horsepower, data=Auto)
  summary(lm.fit2)
```

```
##
## Call:
## lm(formula = mpg ~ . - name + cylinders:displacement + cylinders:horsepower,
##     data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.0246 -1.6646 -0.0235  1.3480 11.8755
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            11.1039607  5.0443246   2.201 0.028315 *
## cylinders              -4.2385487  0.4618816  -9.177  < 2e-16 ***
## displacement           -0.0006610  0.0177105  -0.037 0.970245
## horsepower             -0.3086123  0.0419368  -7.359 1.15e-12 ***
## weight                 -0.0040492  0.0006376  -6.351 6.10e-10 ***
## acceleration           -0.1644937  0.0935930  -1.758 0.079628 .
## year                    0.7515100  0.0460129  16.333  < 2e-16 ***
## originEuropean          1.4429494  0.5273240   2.736 0.006503 **
## originJapanese          1.8105669  0.5228537   3.463 0.000595 ***
```

```
## cylinders:displacement  0.0002336  0.0025871   0.090 0.928089
## cylinders:horsepower    0.0391074  0.0059838   6.535 2.04e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.931 on 381 degrees of freedom
## Multiple R-squared:  0.8626, Adjusted R-squared:  0.859
## F-statistic: 239.2 on 10 and 381 DF,  p-value: < 2.2e-16
```

The interaction between cylinders and horsepower appears to be highly significant.

Part f

```r
#Fit multiple linear regression model with transformation
  lm.fit3 <- lm(mpg~horsepower+I(horsepower^2), data=Auto)
  summary(lm.fit3)
```

```
##
## Call:
## lm(formula = mpg ~ horsepower + I(horsepower^2), data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.7135  -2.5943  -0.0859   2.2868  15.8961
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     56.9000997  1.8004268   31.60   <2e-16 ***
## horsepower      -0.4661896  0.0311246  -14.98   <2e-16 ***
## I(horsepower^2)  0.0012305  0.0001221   10.08   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.374 on 389 degrees of freedom
## Multiple R-squared:  0.6876, Adjusted R-squared:  0.686
## F-statistic:   428 on 2 and 389 DF,  p-value: < 2.2e-16
```

The coefficient on the square of the horsepower variable indicates nonlinearity in the relationship between horsepower and mpg.

Problem 10

Part a

```r
#Import library
  library(ISLR)
#Fit multiple regression model
  lm.fit1 <- lm(Sales~Price+Urban+US, data=Carseats)
  summary(lm.fit1)
```

```
##
## 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
```

Part b

- Coefficient on price: for a unit increase in the price of the carseat, sales will decrease by $0.05 * 1000 = 50$, on average.
- Coefficient on Urban: being sold in an urban area decreases the carseat's sales by $0.02 * 1000 = 20$, on average.
- Coefficient on US: being sold in the United States increases the carseat's sales by 1200, on average.

Part c

The model in equation form:

$$sales_i = \beta_0 + \beta_1 price_i + \beta_2 Urban_i + \beta_3 US_i + \epsilon_i.$$

Part d

We can reject the null hypothesis that $\beta_j = 0$ for the intercept, price, and US.

Part e

```r
#Fit smaller multiple regression model
  lm.fit2 <- lm(Sales~Price+US, data=Carseats)
  summary(lm.fit2)
```

```
##
## 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
```

Part f

Both models fit the data about the same. They both have an R^2 of 0.2393, but the smaller model has a higher adjusted R^2. Moreover, the smaller model has a slighly higher F-statistic and a slightly smaller RSE. All these facts indicate the smaller model fits the data slightly better than the original model.

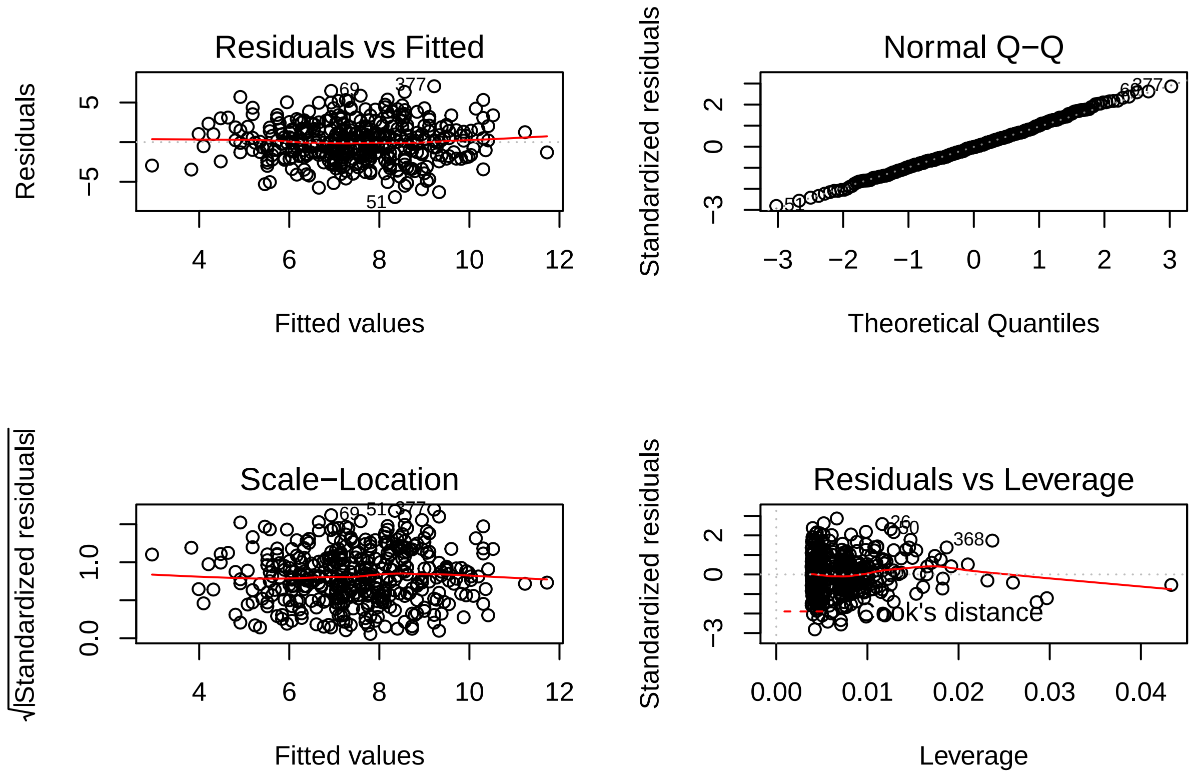
Part g

```r
#Confidence intervals for coefficients
  confint(lm.fit2)
```

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

Part h

```r
#Plot regression diagnostics
  par(mfrow=c(2,2))
  plot(lm.fit2)
```



The regression diagnostic plots do not indicate the presence of any outliers (defined at ± 2 standard errors). The plots do, however, indicate the presence of a couple high leverage points.

Problem 11

Part a

```
set.seed(1)
x <- rnorm(100)
y <- 2*x+rnorm(100)
#Fit simple linear regression of y onto x without intercept
  lm.fit1 <- lm(y~x+0)
  summary(lm.fit1)
```

```
##
## 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
```

According to the summary, we have $\hat{\beta} = 1.9939$, $se(\hat{\beta}) = 0.1065$, a t-statistic of 18.73, and a p-value of $< 2e - 16$. These results allow us to reject the null hypothesis that $\beta = 0$.

Part b

```r
#Fit simple linear regression of x onto y without intercept
  lm.fit2 <- lm(x~y+0)
  summary(lm.fit2)
```

```
##
## 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
```

According to the summary, we have $\hat{\beta} = -0.2368$, $se(\hat{\beta}) = 0.02089$, a t-statistic of 18.73, and a p-value of $< 2e - 16$. These results allow us to reject the null hypothesis that $\beta = 0$.

Part c

We obtain the same p-value in both regression. This reflects the fact that the data come from the same line. We can write $Y = 2X + \epsilon$ as $X = \frac{1}{2}(Y - \epsilon)$.

Part d

To show algebraicly, note that

$$
\begin{aligned}
t &= \frac{\hat{\beta}}{se(\hat{\beta})} \\
&= \frac{\frac{\sum x_i y_i}{\sum x_i^2}}{\sqrt{\frac{\sum (y_i - x_i\hat{\beta})^2}{(n-1)\sum x_i^2}}} \\
&= \frac{\sqrt{n-1}\sum x_i y_i}{\sqrt{\sum x_i^2}\sqrt{\sum (y_i - x_i\hat{\beta})^2}} \\
&= \frac{\sqrt{n-1}\sum x_i y_i}{\sqrt{\sum x_i^2 \sum y_i^2 - \sum x_i^2 \hat{\beta}\left(2\sum x_i y_i - \hat{\beta}\sum x_i^2\right)}} \\
&= \frac{\sqrt{n-1}\sum x_i y_i}{\sqrt{\sum x_i^2 \sum y_i^2 - \sum x_i y_i \left(2\sum x_i y_i - \sum x_i y_i\right)}} \\
&= \frac{\sqrt{n-1}\sum x_i y_i}{\sqrt{\sum x_i^2 \sum y_i^2 - \left(\sum x_i y_i\right)^2}}.
\end{aligned}
$$

```r
#Numerically verify above equation
  n <- length(x)
  t <- sqrt(n - 1)*(x %*% y)/sqrt(sum(x^2) * sum(y^2) - (x %*% y)^2)
  as.numeric(t)
```

```
## [1] 18.72593
```

This is the exact same t-value as in the previous two regressions.

Part e

From the previous equation, we see the formula for the t-statistic is symmetric with respect to the ordering of x and y. That is, we can swap the value of x and y in the above equation and the equation remains the same. Hence, for the t-statistic for x y is the same as the t-statistic for y x.

Part f

```r
#Fit simple linear regression of y onto x without intercept
  lm.fit3 <- lm(y~x)
  summary(lm.fit3)
```

```
##
## 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
```

```r
#Reverse variable order
  lm.fit4 <- lm(x~y)
  summary(lm.fit4)
```

```
##
## 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 the previous two summaries, we see the t-statistic for testing that $\beta_1 = 0$ in the simple linear regression models is the same for both y x and x y.

Problem 12

Part a

The coefficient is the same if and only if

$$\frac{\sum_{i=1}^n x_i y_i}{\sum_{j=1}^n x_j^2} = \frac{\sum_{i=1}^n x_i y_i}{\sum_{j=1}^n y_j^2}.$$

That is, we need

$$\sum_{j=1}^n y_j^2 = \sum_{j=1}^n x_j^2.$$

Part b

```r
#Example where coefficients for Y~X and X~Y are not the same
  set.seed(1)
  x <- rnorm(100)
  y <- 2*x
```

```r
#Show sum of squares is different
  sum(x^2)
```

```
## [1] 81.05509
```

```r
  sum(y^2)
```

```
## [1] 324.2204
```

```r
#Fit first model
  lm.fit1 <- lm(y~x+0)
  summary(lm.fit1)
```

```
## Warning in summary.lm(lm.fit1): essentially perfect fit: summary may be
## unreliable

##
## Call:
## lm(formula = y ~ x + 0)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -3.739e-15 -5.130e-17 -1.200e-18  3.250e-17  2.639e-16
##
## Coefficients:
##   Estimate Std. Error   t value Pr(>|t|)
## x 2.00e+00   4.29e-17 4.662e+16   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.862e-16 on 99 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 2.174e+33 on 1 and 99 DF,  p-value: < 2.2e-16
```

```r
#Fit second model
  lm.fit2 <- lm(x~y+0)
  summary(lm.fit2)
```

```
## Warning in summary.lm(lm.fit2): essentially perfect fit: summary may be
## unreliable

##
## Call:
## lm(formula = x ~ y + 0)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -1.870e-15 -2.567e-17 -6.200e-19  1.624e-17  1.320e-16
##
## Coefficients:
##    Estimate Std. Error   t value Pr(>|t|)
## y 5.000e-01  1.072e-17 4.662e+16   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.931e-16 on 99 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
```

```
## F-statistic: 2.174e+33 on 1 and 99 DF,  p-value: < 2.2e-16
```

The coefficients are clearly different.

Part c

```r
#Example where coefficients for Y~X and X~Y are the same
  set.seed(1)
  x <- rnorm(100)
  y <- -sample(x,100) #Just re-order the values in x and multiply by -1
#Show sum of squares is different
  sum(x^2)
```

```
## [1] 81.05509
```

```r
  sum(y^2)
```

```
## [1] 81.05509
```

```r
#Fit first model
  lm.fit3 <- lm(y~x+0)
  summary(lm.fit3)
```

```
##
## Call:
## lm(formula = y ~ x + 0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.3926 -0.6877 -0.1027  0.5124  2.2315
##
## Coefficients:
##   Estimate Std. Error t value Pr(>|t|)
## x -0.02148    0.10048  -0.214    0.831
##
## Residual standard error: 0.9046 on 99 degrees of freedom
## Multiple R-squared:  0.0004614,  Adjusted R-squared:  -0.009635
## F-statistic: 0.0457 on 1 and 99 DF,  p-value: 0.8312
```

```r
#Fit second model
  lm.fit4 <- lm(x~y+0)
  summary(lm.fit4)
```

```
##
## Call:
## lm(formula = x ~ y + 0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.2400 -0.5154  0.1213  0.6788  2.3959
##
## Coefficients:
##   Estimate Std. Error t value Pr(>|t|)
## y -0.02148    0.10048  -0.214    0.831
##
## Residual standard error: 0.9046 on 99 degrees of freedom
```

```
## Multiple R-squared:  0.0004614,  Adjusted R-squared:  -0.009635
## F-statistic: 0.0457 on 1 and 99 DF,  p-value: 0.8312
```

As we can see in the summaries above the coefficients are the same.

Problem 13

Part a

```r
#Generate x
  set.seed(1)
  x <- rnorm(100)
```

Part b

```r
#Generate eps
  eps <- rnorm(100, mean=0,sd=sqrt(0.25))
```

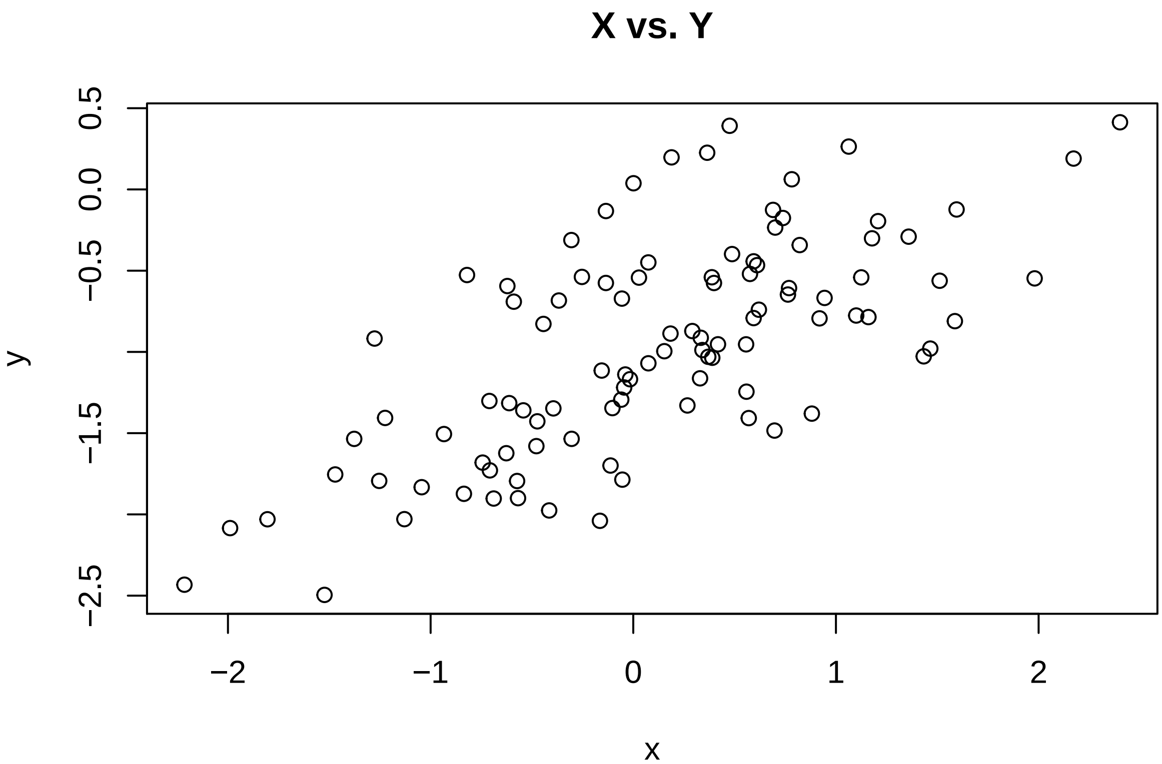
Part c

```r
#Generate y
  y <- -1 + (0.5*x) + eps
```

The vector y has length 100. This model has $\beta_0 = -1$ and $\beta_1 = 0.5$.

Part d

```r
#Plot x vs y
  plot(x,y,xlab="x",ylab="y",main="X vs. Y")
```



Appears to be a linear relationship between x and y.

Part e

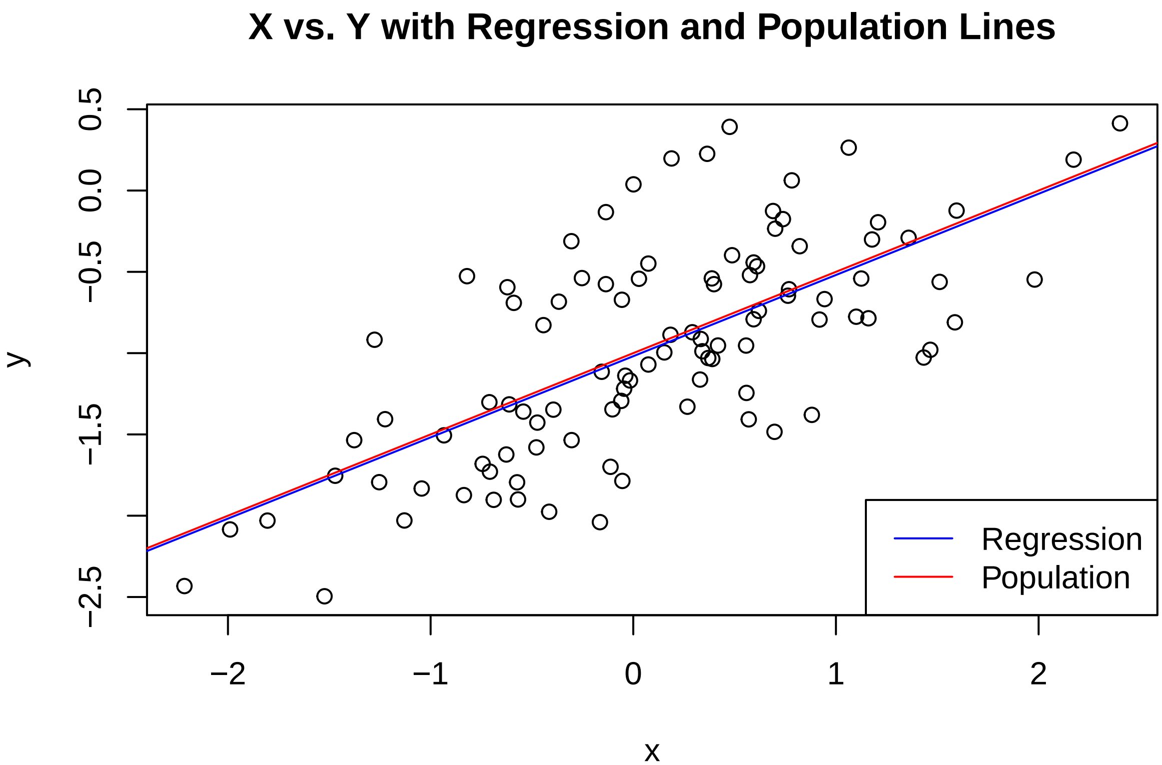
```r
#Fit the least squares linear model
  lm.fit1 <- lm(y~x)
  summary(lm.fit1)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## 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 ***
## x            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
```

From the model above, we have $\hat{\beta}_0 = 1.01885$ and $\hat{\beta}_1 = 0.49947$. These estimates are very close to the true β_0 and β_1.

Part f

```r
#Plot x vs y with regression and population lines
  plot(x,y,xlab="x",ylab="y",main="X vs. Y with Regression and Population Lines")
  abline(lm.fit1,col="blue")
  abline(-1,0.5, col="red")
  legend("bottomright",c("Regression","Population"), col=c("blue","red"), lty=c(1,1))
```



Part g

```r
#Fit regression with quadratic term
  lm.fit2 <- lm(y~x+I(x^2))
  summary(lm.fit2)
```

```
##
## Call:
## lm(formula = y ~ x + I(x^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 ***
## x            0.50858    0.05399   9.420  2.4e-15 ***
## I(x^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
```

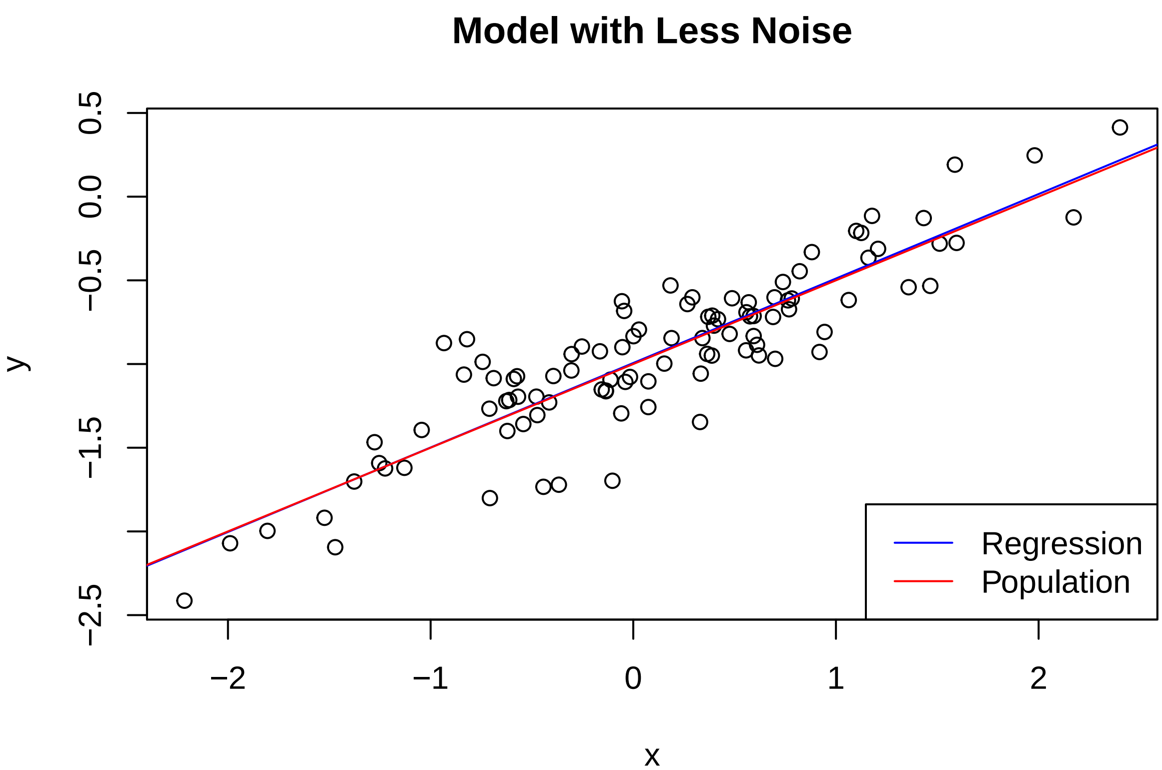
There is little evidence the addition of the quadratic term improves the fit of the model. The adjusted R^2 barely increases, and the coefficient on the quadratic term is not statistically significant.

Part h

```r
#Generate data with less noise
  eps2 <- rnorm(100, mean=0,sd=sqrt(0.05))
  y2 <- -1 + (0.5*x) + eps2
#Fit the least squares linear model
  lm.fit3 <- lm(y2~x)
  summary(lm.fit3)
```

```
##
## Call:
## lm(formula = y2 ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.65162 -0.10785 -0.01014  0.14518  0.59067
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.99388    0.02341  -42.45   <2e-16 ***
## x            0.50473    0.02601   19.41   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2324 on 98 degrees of freedom
## Multiple R-squared:  0.7936, Adjusted R-squared:  0.7915
## F-statistic: 376.7 on 1 and 98 DF,  p-value: < 2.2e-16
```

```r
#Plot x vs y with regression and population lines
  plot(x,y2,xlab="x",ylab="y",main="Model with Less Noise")
  abline(lm.fit3,col="blue")
  abline(-1,0.5, col="red")
  legend("bottomright",c("Regression","Population"), col=c("blue","red"), lty=c(1,1))
```



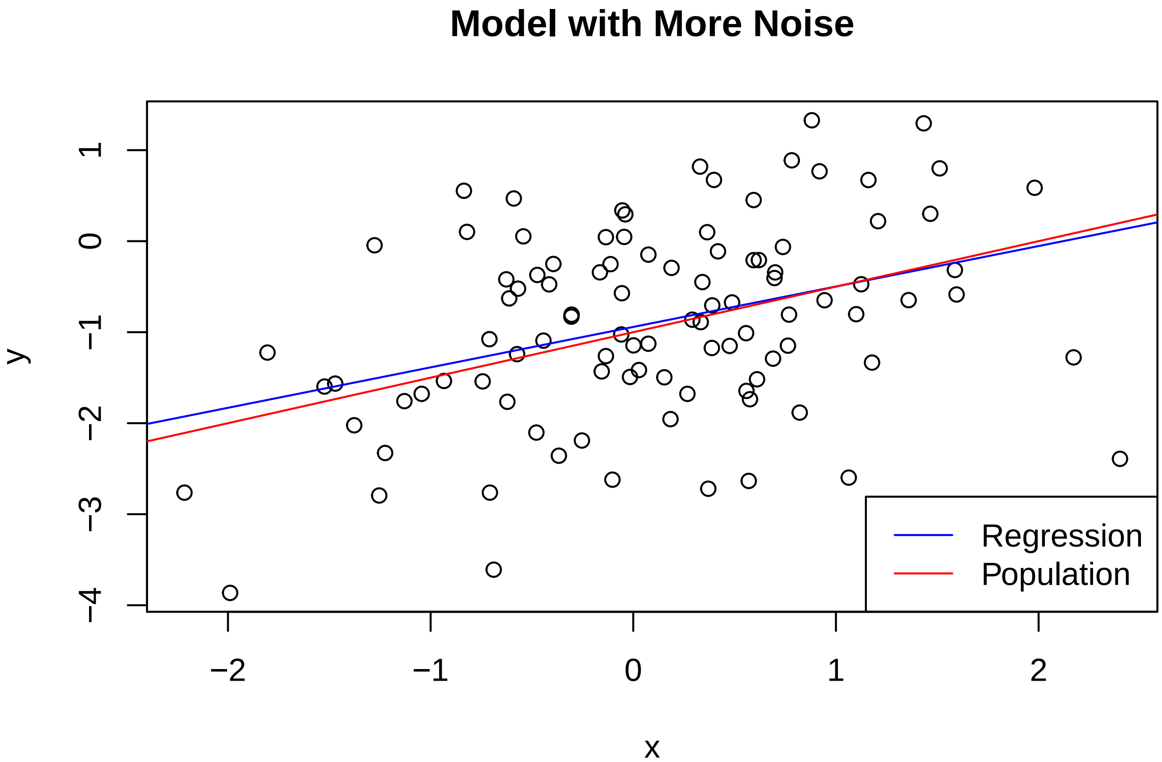
The regression line and coefficients are closer to the population values than in the original model.

Part i

```
#Generate data with more noise
  eps3 <- rnorm(100, mean=0)
  y3 <- -1 + (0.5*x) + eps3
#Fit the least squares linear model
  lm.fit4 <- lm(y3~x)
  summary(lm.fit4)
```

```
##
## Call:
## lm(formula = y3 ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.51626 -0.54525 -0.03776  0.67289  1.87887
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.9423     0.1003  -9.397 2.47e-15 ***
## x             0.4443     0.1114   3.989 0.000128 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9955 on 98 degrees of freedom
## Multiple R-squared:  0.1397, Adjusted R-squared:  0.1309
## F-statistic: 15.91 on 1 and 98 DF,  p-value: 0.000128
```

```r
#Plot x vs y with regression and population lines
  plot(x,y3,xlab="x",ylab="y",main="Model with More Noise")
  abline(lm.fit4,col="blue")
  abline(-1,0.5, col="red")
  legend("bottomright",c("Regression","Population"), col=c("blue","red"), lty=c(1,1))
```



The regression line and coefficients are less close to the population values than in the original model.

Part j

```r
#Coefficient confidence intervals for original model
  confint(lm.fit1)
```

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

```r
#Coefficient confidence intervals for model with less noise
  confint(lm.fit3)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.0403415 -0.9474188
## x            0.4531269  0.5563393
```

```r
#Coefficient confidence intervals for model with more noise
  confint(lm.fit4)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.1413399 -0.7433293
## x            0.2232721  0.6653558
```

The confidence intervals for the coefficients in the noisier data are larger than the confidence intervals for the coefficients in the original data. Likewise, the confidence intervals for the coefficients in the original data are larger than the confidence intervals for the coefficients in the less noisy data.

Problem 14

Part a

```
#Generate the data
  set.seed(1)
  x1 <- runif(100)
  x2 <- 0.5*x1 + rnorm(100)/10
  y <- 2 + 2*x1 + 0.3*x2 + rnorm(100)
```

The model has the form

$$\begin{aligned} y_i &= \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \epsilon_i \\ &= 2 + 2x_{1,i} + 0.3x_{2,i} + \epsilon_i. \end{aligned}$$

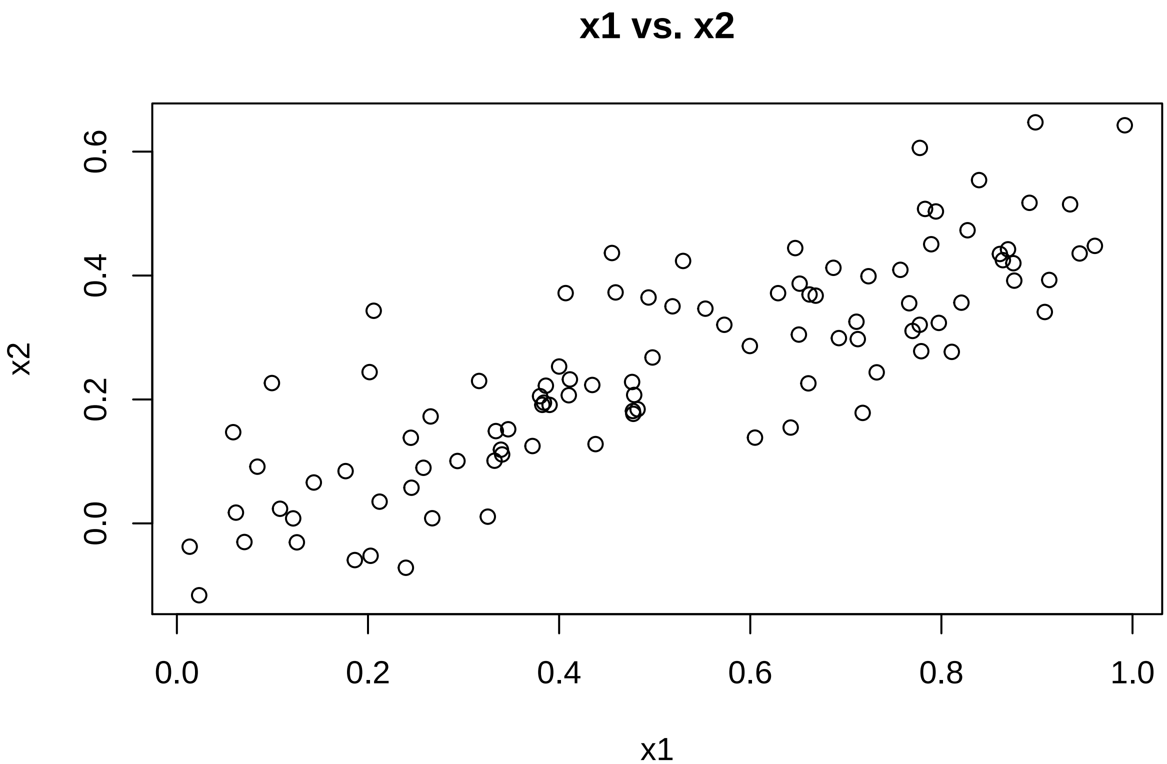
Hence, $\beta_0 = 2$, $\beta_1 = 2$, and $\beta_2 = 0.3$.

Part b

```
#Examine correlation
  cor(x1,x2)
```

```
## [1] 0.8351212
```

```
  plot(x1,x2,xlab="x1",ylab="x2",main="x1 vs. x2")
```




```r
  lm.fit1 <- lm(y~x1+x2)
  summary(lm.fit1)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8311 -0.7273 -0.0537  0.6338  2.3359
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.1305     0.2319   9.188 7.61e-15 ***
## x1            1.4396     0.7212   1.996   0.0487 *
## x2            1.0097     1.1337   0.891   0.3754
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.056 on 97 degrees of freedom
## Multiple R-squared:  0.2088, Adjusted R-squared:  0.1925
## F-statistic:  12.8 on 2 and 97 DF,  p-value: 1.164e-05
```

Part c

```r
#Fit linear model
  lm.fit1 <- lm(y~x1+x2)
  summary(lm.fit1)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8311 -0.7273 -0.0537  0.6338  2.3359
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.1305     0.2319   9.188 7.61e-15 ***
## x1            1.4396     0.7212   1.996   0.0487 *
## x2            1.0097     1.1337   0.891   0.3754
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.056 on 97 degrees of freedom
## Multiple R-squared:  0.2088, Adjusted R-squared:  0.1925
## F-statistic:  12.8 on 2 and 97 DF,  p-value: 1.164e-05
```

From the regression, we have $\hat{\beta}_0 = 2.1305$, $\hat{\beta}_1 = 1.4396$, and $\hat{\beta}_2 = 1.0097$. These results are pretty far from the true values of $\hat{\beta}_0$, $\hat{\beta}_1$, and $\hat{\beta}_2$. We can only reject the null that $\beta_1 = 0$ at the 10% significance level. We cannot rejec the null that $\beta_2 = 0$.

Part d

```r
#Simple linear regression using just x1
  lm.fit2 <- lm(y~x1)
  summary(lm.fit2)
```

```
##
## Call:
## lm(formula = y ~ x1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.89495 -0.66874 -0.07785  0.59221  2.45560
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.1124     0.2307   9.155 8.27e-15 ***
## x1            1.9759     0.3963   4.986 2.66e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.055 on 98 degrees of freedom
## Multiple R-squared:  0.2024, Adjusted R-squared:  0.1942
## F-statistic: 24.86 on 1 and 98 DF,  p-value: 2.661e-06
```

In the model using just x1 as a predictor, we can reject the null that $\beta_1 = 0$ with a high degree of confidence ($p < 0.001$).

Part e

```r
#Simple linear regression using just x2
  lm.fit3 <- lm(y~x2)
  summary(lm.fit3)
```

```
##
## Call:
## lm(formula = y ~ x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.62687 -0.75156 -0.03598  0.72383  2.44890
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.3899     0.1949   12.26  < 2e-16 ***
## x2            2.8996     0.6330    4.58 1.37e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.072 on 98 degrees of freedom
## Multiple R-squared:  0.1763, Adjusted R-squared:  0.1679
## F-statistic: 20.98 on 1 and 98 DF,  p-value: 1.366e-05
```

In the model using just x2 as a predictor, we can reject the null that $\beta_2 = 0$ with a high degree of confidence ($p < 0.001$).

Part f

The results of the previous three regressions do not contradict each other. Without the presence of other predictors, both β_1 and β_2 are statistically significant. In the presence of other predictors, β_2 is no longer statistically significant.

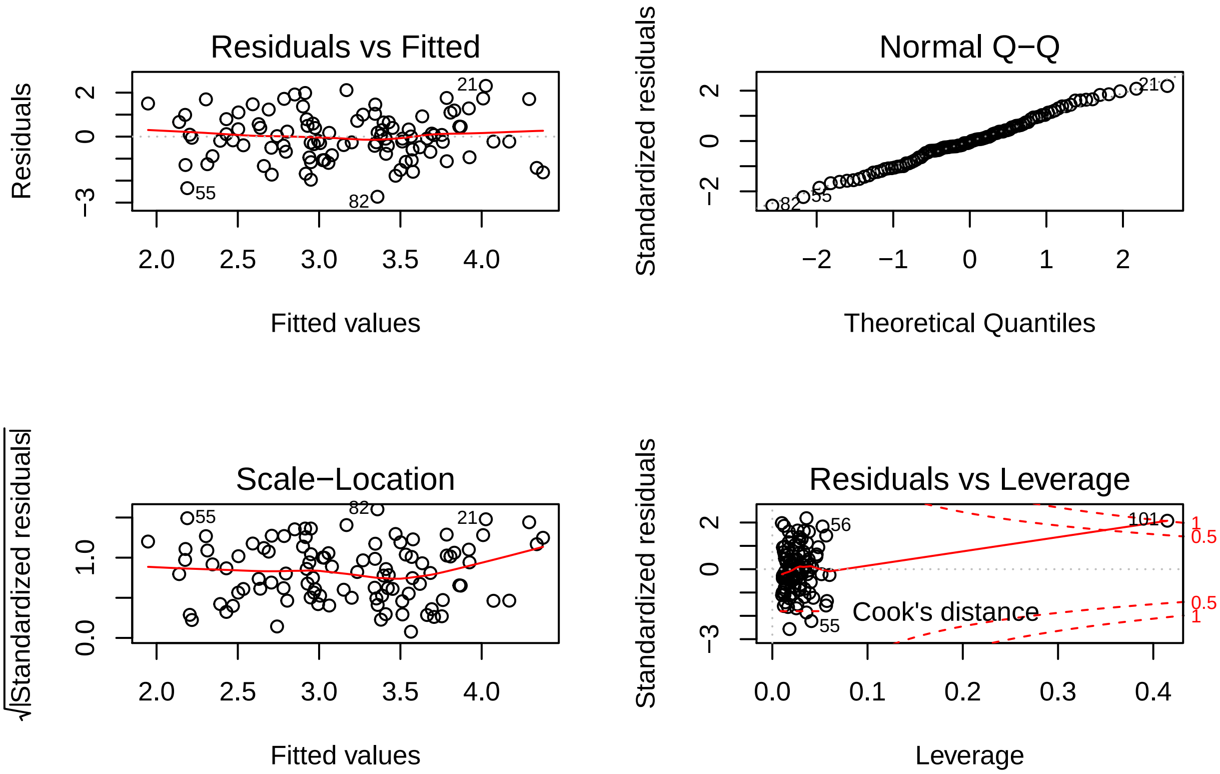
Part g

```r
x1 <- c(x1, 0.1)
x2 <- c(x2, 0.8)
y <- c(y, 6)
par(mfrow=c(2,2))
# regression with both x1 and x2
  lm.fit4 <- lm(y~x1+x2)
  summary(lm.fit4)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.73348 -0.69318 -0.05263  0.66385  2.30619
##
## Coefficients:
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.2267     0.2314   9.624 7.91e-16 ***
## x1             0.5394     0.5922   0.911  0.36458
## x2             2.5146     0.8977   2.801  0.00614 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.075 on 98 degrees of freedom
## Multiple R-squared:  0.2188, Adjusted R-squared:  0.2029
## F-statistic: 13.72 on 2 and 98 DF,  p-value: 5.564e-06
```

```r
plot(lm.fit4)
```

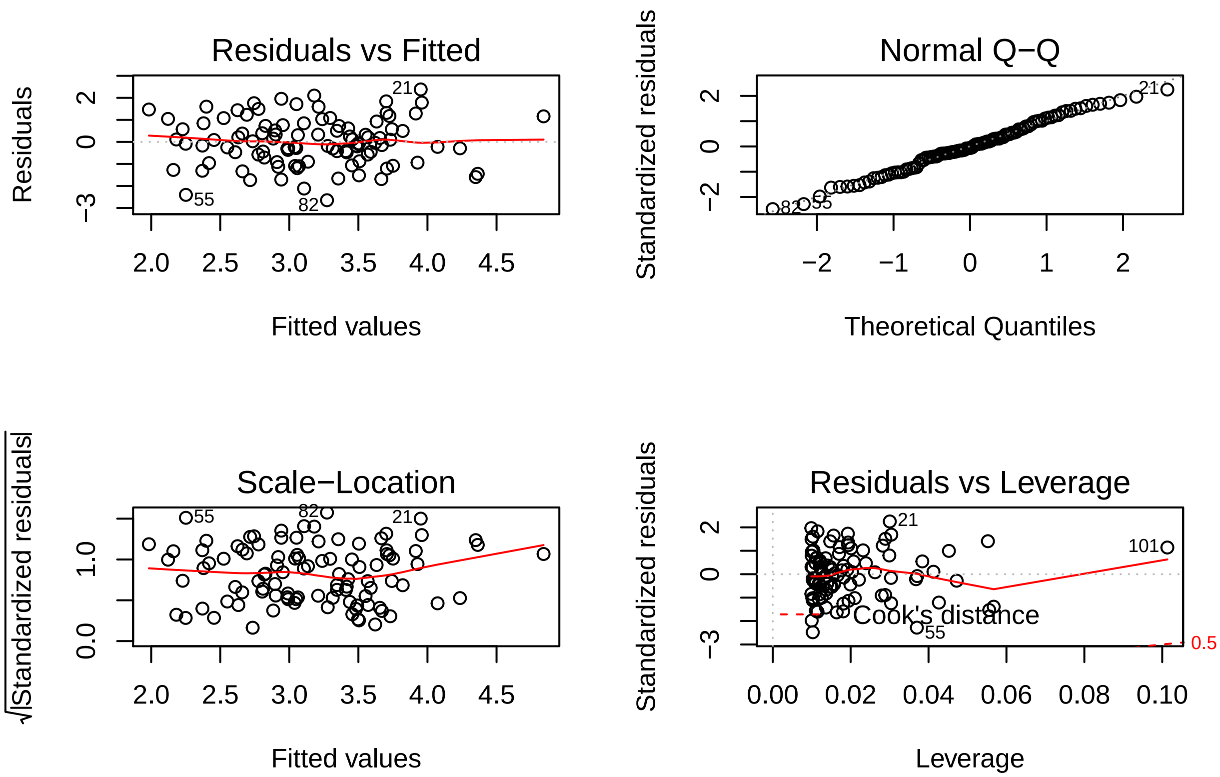



```r
# regression with x1 only
lm.fit5 <- lm(y~x2)
summary(lm.fit5)
```

```
##
## Call:
## lm(formula = y ~ x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.64729 -0.71021 -0.06899  0.72699  2.38074
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.3451     0.1912  12.264  < 2e-16 ***
## x2            3.1190     0.6040   5.164 1.25e-06 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.074 on 99 degrees of freedom
## Multiple R-squared:  0.2122, Adjusted R-squared:  0.2042
## F-statistic: 26.66 on 1 and 99 DF,  p-value: 1.253e-06
```

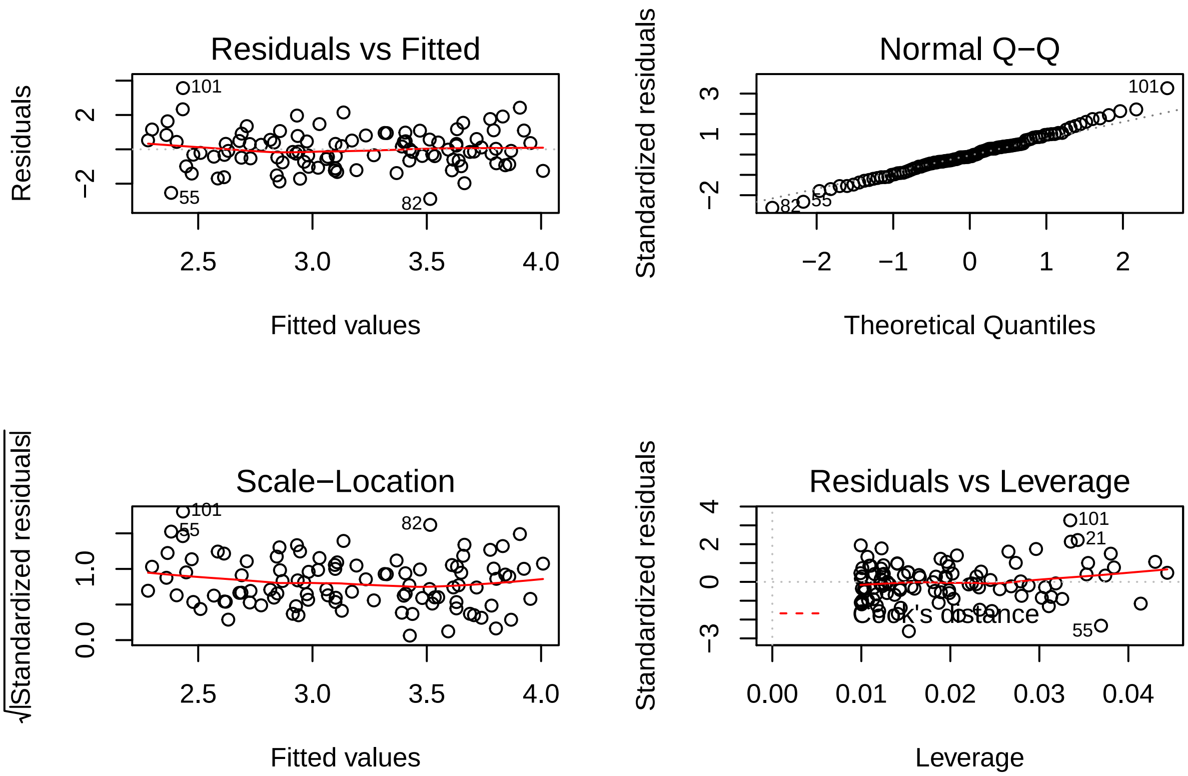
```r
plot(lm.fit5)
```




```r
# regression with x2 only
  lm.fit6 <- lm(y~x1)
  summary(lm.fit6)
```

```
##
## Call:
## lm(formula = y ~ x1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8897 -0.6556 -0.0909  0.5682  3.5665
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.2569     0.2390   9.445 1.78e-15 ***
## x1            1.7657     0.4124   4.282 4.29e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.111 on 99 degrees of freedom
## Multiple R-squared:  0.1562, Adjusted R-squared:  0.1477
## F-statistic: 18.33 on 1 and 99 DF,  p-value: 4.295e-05
```

```r
plot(lm.fit6)
```



The new point is an outlier for all three models (though not quite as bad in the model with just x2), and it is an outlier in the model with just x1 and the model with just x2.

- In the model with x1 and x2, the residuals vs leverage plot shows the new observation as being high-leverage.
- In the model with just x1, the new point has high leverage but does not cause issues because it is not an outlier for x1 or y.
- In the model with just x2, the new point has high leverage but does not cause major issues because it falls close to the regression line.

Problem 15

Part a

```r
#Import the data
library(MASS)
names(Boston)
```

```
##  [1] "crim"    "zn"      "indus"   "chas"    "nox"     "rm"      "age"
##  [8] "dis"     "rad"     "tax"     "ptratio" "black"   "lstat"   "medv"
```

```r
#Fit the univariate models
fit1 <- lm(crim~zn, data=Boston)
summary(fit1)
```

```
##
## Call:
```

```
## lm(formula = crim ~ zn, data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.429 -4.222 -2.620  1.250 84.523
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.45369    0.41722  10.675  < 2e-16 ***
## zn          -0.07393    0.01609  -4.594 5.51e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.435 on 504 degrees of freedom
## Multiple R-squared:  0.04019,    Adjusted R-squared:  0.03828
## F-statistic:  21.1 on 1 and 504 DF,  p-value: 5.506e-06
```

```r
fit2 <- lm(crim~indus, data=Boston)
summary(fit2)
```

```
##
## Call:
## lm(formula = crim ~ indus, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -11.972  -2.698  -0.736   0.712  81.813
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.06374    0.66723  -3.093  0.00209 **
## indus        0.50978    0.05102   9.991  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.866 on 504 degrees of freedom
## Multiple R-squared:  0.1653, Adjusted R-squared:  0.1637
## F-statistic: 99.82 on 1 and 504 DF,  p-value: < 2.2e-16
```

```r
fit3 <- lm(crim~chas, data=Boston)
summary(fit3)
```

```
##
## Call:
## lm(formula = crim ~ chas, data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.738 -3.661 -3.435  0.018 85.232
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.7444     0.3961   9.453   <2e-16 ***
## chas         -1.8928     1.5061  -1.257    0.209
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.597 on 504 degrees of freedom
## Multiple R-squared:  0.003124,	Adjusted R-squared:  0.001146
## F-statistic: 1.579 on 1 and 504 DF,  p-value: 0.2094
```

```r
fit4 <- lm(crim~nox, data=Boston)
summary(fit4)
```

```
##
## Call:
## lm(formula = crim ~ nox, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -12.371  -2.738  -0.974   0.559  81.728
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -13.720      1.699  -8.073 5.08e-15 ***
## nox           31.249      2.999  10.419  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.81 on 504 degrees of freedom
## Multiple R-squared:  0.1772, Adjusted R-squared:  0.1756
## F-statistic: 108.6 on 1 and 504 DF,  p-value: < 2.2e-16
```

```r
fit5 <- lm(crim~rm, data=Boston)
summary(fit5)
```

```
##
## Call:
## lm(formula = crim ~ rm, data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.604 -3.952 -2.654  0.989 87.197
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   20.482      3.365   6.088 2.27e-09 ***
## rm            -2.684      0.532  -5.045 6.35e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.401 on 504 degrees of freedom
## Multiple R-squared:  0.04807,	Adjusted R-squared:  0.04618
## F-statistic: 25.45 on 1 and 504 DF,  p-value: 6.347e-07
```

```r
fit6 <- lm(crim~age, data=Boston)
summary(fit6)
```

```
##
## Call:
## lm(formula = crim ~ age, data = Boston)
```

```
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.789 -4.257 -1.230  1.527 82.849
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.77791    0.94398  -4.002 7.22e-05 ***
## age          0.10779    0.01274   8.463 2.85e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.057 on 504 degrees of freedom
## Multiple R-squared:  0.1244, Adjusted R-squared:  0.1227
## F-statistic: 71.62 on 1 and 504 DF,  p-value: 2.855e-16
```

```r
fit7 <- lm(crim~dis, data=Boston)
summary(fit7)
```

```
##
## Call:
## lm(formula = crim ~ dis, data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.708 -4.134 -1.527  1.516 81.674
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.4993     0.7304  13.006   <2e-16 ***
## dis          -1.5509     0.1683  -9.213   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.965 on 504 degrees of freedom
## Multiple R-squared:  0.1441, Adjusted R-squared:  0.1425
## F-statistic: 84.89 on 1 and 504 DF,  p-value: < 2.2e-16
```

```r
fit8 <- lm(crim~rad, data=Boston)
summary(fit8)
```

```
##
## Call:
## lm(formula = crim ~ rad, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.164  -1.381  -0.141   0.660  76.433
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.28716    0.44348  -5.157 3.61e-07 ***
## rad          0.61791    0.03433  17.998  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 6.718 on 504 degrees of freedom
## Multiple R-squared:  0.3913, Adjusted R-squared:   0.39
## F-statistic: 323.9 on 1 and 504 DF,  p-value: < 2.2e-16
```

```r
fit9 <- lm(crim~tax, data=Boston)
summary(fit9)
```

```
##
## Call:
## lm(formula = crim ~ tax, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -12.513  -2.738  -0.194   1.065  77.696
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -8.528369   0.815809  -10.45   <2e-16 ***
## tax          0.029742   0.001847   16.10   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.997 on 504 degrees of freedom
## Multiple R-squared:  0.3396, Adjusted R-squared:  0.3383
## F-statistic: 259.2 on 1 and 504 DF,  p-value: < 2.2e-16
```

```r
fit10 <- lm(crim~ptratio, data=Boston)
summary(fit10)
```

```
##
## Call:
## lm(formula = crim ~ ptratio, data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -7.654 -3.985 -1.912  1.825 83.353
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -17.6469     3.1473  -5.607 3.40e-08 ***
## ptratio       1.1520     0.1694   6.801 2.94e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.24 on 504 degrees of freedom
## Multiple R-squared:  0.08407,    Adjusted R-squared:  0.08225
## F-statistic: 46.26 on 1 and 504 DF,  p-value: 2.943e-11
```

```r
fit11 <- lm(crim~black, data=Boston)
summary(fit11)
```

```
##
## Call:
## lm(formula = crim ~ black, data = Boston)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.756  -2.299  -2.095  -1.296  86.822
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 16.553529   1.425903  11.609   <2e-16 ***
## black       -0.036280   0.003873  -9.367   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.946 on 504 degrees of freedom
## Multiple R-squared:  0.1483, Adjusted R-squared:  0.1466
## F-statistic: 87.74 on 1 and 504 DF,  p-value: < 2.2e-16
```

```r
fit12 <- lm(crim~lstat, data=Boston)
summary(fit12)
```

```
##
## Call:
## lm(formula = crim ~ lstat, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.925  -2.822  -0.664   1.079  82.862
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.33054    0.69376  -4.801 2.09e-06 ***
## lstat        0.54880    0.04776  11.491  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.664 on 504 degrees of freedom
## Multiple R-squared:  0.2076, Adjusted R-squared:  0.206
## F-statistic:   132 on 1 and 504 DF,  p-value: < 2.2e-16
```

```r
fit13 <- lm(crim~medv, data=Boston)
summary(fit13)
```

```
##
## Call:
## lm(formula = crim ~ medv, data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.071 -4.022 -2.343  1.298 80.957
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 11.79654    0.93419   12.63   <2e-16 ***
## medv        -0.36316    0.03839   -9.46   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 7.934 on 504 degrees of freedom
## Multiple R-squared:  0.1508, Adjusted R-squared:  0.1491
## F-statistic: 89.49 on 1 and 504 DF,  p-value: < 2.2e-16
```

Each predictor has a statistically significant association with the response except for the chas variable.

Part b

```r
#Fit multiple regression model
  fit14 <- lm(crim~., data=Boston)
  summary(fit14)
```

```
##
## Call:
## lm(formula = crim ~ ., data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.924 -2.120 -0.353  1.019 75.051
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  17.033228   7.234903   2.354 0.018949 *
## zn            0.044855   0.018734   2.394 0.017025 *
## indus        -0.063855   0.083407  -0.766 0.444294
## chas         -0.749134   1.180147  -0.635 0.525867
## nox         -10.313535   5.275536  -1.955 0.051152 .
## rm            0.430131   0.612830   0.702 0.483089
## age           0.001452   0.017925   0.081 0.935488
## dis          -0.987176   0.281817  -3.503 0.000502 ***
## rad           0.588209   0.088049   6.680 6.46e-11 ***
## tax          -0.003780   0.005156  -0.733 0.463793
## ptratio      -0.271081   0.186450  -1.454 0.146611
## black        -0.007538   0.003673  -2.052 0.040702 *
## lstat         0.126211   0.075725   1.667 0.096208 .
## medv         -0.198887   0.060516  -3.287 0.001087 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.439 on 492 degrees of freedom
## Multiple R-squared:  0.454,  Adjusted R-squared:  0.4396
## F-statistic: 31.47 on 13 and 492 DF,  p-value: < 2.2e-16
```

In the multiple regression model, we can rejec the null for zn, nox, dis, rad, black, lstat, and medv.

Part c

In the multople regression model, fewer predictors have a significant association with the response.

Part d

```r
#Examine non-linearities
# skip chas because it's a factor variable
  summary(lm(crim~poly(zn,3), data=Boston))      # 1,2
```

```
##
## Call:
## lm(formula = crim ~ poly(zn, 3), data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.821 -4.614 -1.294  0.473 84.130
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.6135     0.3722   9.709  < 2e-16 ***
## poly(zn, 3)1 -38.7498     8.3722  -4.628  4.7e-06 ***
## poly(zn, 3)2  23.9398     8.3722   2.859  0.00442 **
## poly(zn, 3)3 -10.0719     8.3722  -1.203  0.22954
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.372 on 502 degrees of freedom
## Multiple R-squared:  0.05824,    Adjusted R-squared:  0.05261
## F-statistic: 10.35 on 3 and 502 DF,  p-value: 1.281e-06
```

```r
  summary(lm(crim~poly(indus,3), data=Boston))   # 1,2,3
```

```
##
## Call:
## lm(formula = crim ~ poly(indus, 3), data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -8.278 -2.514  0.054  0.764 79.713
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)        3.614      0.330  10.950  < 2e-16 ***
## poly(indus, 3)1   78.591      7.423  10.587  < 2e-16 ***
## poly(indus, 3)2  -24.395      7.423  -3.286  0.00109 **
## poly(indus, 3)3  -54.130      7.423  -7.292  1.2e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.423 on 502 degrees of freedom
## Multiple R-squared:  0.2597, Adjusted R-squared:  0.2552
## F-statistic: 58.69 on 3 and 502 DF,  p-value: < 2.2e-16
```

```r
  summary(lm(crim~poly(nox,3), data=Boston))     # 1,2,3
```

```
##
## Call:
## lm(formula = crim ~ poly(nox, 3), data = Boston)
##
## Residuals:
```

```
##    Min     1Q Median     3Q    Max
## -9.110 -2.068 -0.255  0.739 78.302
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.6135     0.3216  11.237  < 2e-16 ***
## poly(nox, 3)1  81.3720     7.2336  11.249  < 2e-16 ***
## poly(nox, 3)2 -28.8286     7.2336  -3.985 7.74e-05 ***
## poly(nox, 3)3 -60.3619     7.2336  -8.345 6.96e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.234 on 502 degrees of freedom
## Multiple R-squared:  0.297,  Adjusted R-squared:  0.2928
## F-statistic: 70.69 on 3 and 502 DF,  p-value: < 2.2e-16
```

```
summary(lm(crim~poly(rm,3), data=Boston))      # 1,2
```

```
##
## Call:
## lm(formula = crim ~ poly(rm, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.485  -3.468  -2.221  -0.015  87.219
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.6135     0.3703   9.758  < 2e-16 ***
## poly(rm, 3)1 -42.3794     8.3297  -5.088 5.13e-07 ***
## poly(rm, 3)2  26.5768     8.3297   3.191  0.00151 **
## poly(rm, 3)3  -5.5103     8.3297  -0.662  0.50858
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.33 on 502 degrees of freedom
## Multiple R-squared:  0.06779,    Adjusted R-squared:  0.06222
## F-statistic: 12.17 on 3 and 502 DF,  p-value: 1.067e-07
```

```
summary(lm(crim~poly(age,3), data=Boston))      # 1,2,3
```

```
##
## Call:
## lm(formula = crim ~ poly(age, 3), data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.762 -2.673 -0.516  0.019 82.842
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.6135     0.3485  10.368  < 2e-16 ***
## poly(age, 3)1  68.1820     7.8397   8.697  < 2e-16 ***
## poly(age, 3)2  37.4845     7.8397   4.781 2.29e-06 ***
## poly(age, 3)3  21.3532     7.8397   2.724  0.00668 **
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.84 on 502 degrees of freedom
## Multiple R-squared:  0.1742, Adjusted R-squared:  0.1693
## F-statistic: 35.31 on 3 and 502 DF,  p-value: < 2.2e-16
```

```r
summary(lm(crim~poly(dis,3), data=Boston))     # 1,2,3
```

```
##
## Call:
## lm(formula = crim ~ poly(dis, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.757  -2.588   0.031   1.267  76.378
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)       3.6135     0.3259  11.087  < 2e-16 ***
## poly(dis, 3)1   -73.3886     7.3315 -10.010  < 2e-16 ***
## poly(dis, 3)2    56.3730     7.3315   7.689 7.87e-14 ***
## poly(dis, 3)3   -42.6219     7.3315  -5.814 1.09e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.331 on 502 degrees of freedom
## Multiple R-squared:  0.2778, Adjusted R-squared:  0.2735
## F-statistic: 64.37 on 3 and 502 DF,  p-value: < 2.2e-16
```

```r
summary(lm(crim~poly(rad,3), data=Boston))     # 1,2
```

```
##
## Call:
## lm(formula = crim ~ poly(rad, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.381  -0.412  -0.269   0.179  76.217
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)       3.6135     0.2971  12.164  < 2e-16 ***
## poly(rad, 3)1   120.9074     6.6824  18.093  < 2e-16 ***
## poly(rad, 3)2    17.4923     6.6824   2.618  0.00912 **
## poly(rad, 3)3     4.6985     6.6824   0.703  0.48231
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.682 on 502 degrees of freedom
## Multiple R-squared:    0.4,  Adjusted R-squared:  0.3965
## F-statistic: 111.6 on 3 and 502 DF,  p-value: < 2.2e-16
```

```r
summary(lm(crim~poly(tax,3), data=Boston))     # 1,2
```

```
##
```

```
## Call:
## lm(formula = crim ~ poly(tax, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.273  -1.389   0.046   0.536  76.950
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.6135     0.3047  11.860  < 2e-16 ***
## poly(tax, 3)1 112.6458     6.8537  16.436  < 2e-16 ***
## poly(tax, 3)2  32.0873     6.8537   4.682 3.67e-06 ***
## poly(tax, 3)3  -7.9968     6.8537  -1.167    0.244
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.854 on 502 degrees of freedom
## Multiple R-squared:  0.3689, Adjusted R-squared:  0.3651
## F-statistic:  97.8 on 3 and 502 DF,  p-value: < 2.2e-16
```

```
summary(lm(crim~poly(ptratio,3), data=Boston)) # 1,2,3
```

```
##
## Call:
## lm(formula = crim ~ poly(ptratio, 3), data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.833 -4.146 -1.655  1.408 82.697
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)           3.614      0.361  10.008  < 2e-16 ***
## poly(ptratio, 3)1    56.045      8.122   6.901 1.57e-11 ***
## poly(ptratio, 3)2    24.775      8.122   3.050  0.00241 **
## poly(ptratio, 3)3   -22.280      8.122  -2.743  0.00630 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.122 on 502 degrees of freedom
## Multiple R-squared:  0.1138, Adjusted R-squared:  0.1085
## F-statistic: 21.48 on 3 and 502 DF,  p-value: 4.171e-13
```

```
summary(lm(crim~poly(black,3), data=Boston))   # 1
```

```
##
## Call:
## lm(formula = crim ~ poly(black, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.096  -2.343  -2.128  -1.439  86.790
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)       3.6135     0.3536  10.218   <2e-16 ***
## poly(black, 3)1 -74.4312     7.9546  -9.357   <2e-16 ***
## poly(black, 3)2   5.9264     7.9546   0.745    0.457
## poly(black, 3)3  -4.8346     7.9546  -0.608    0.544
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.955 on 502 degrees of freedom
## Multiple R-squared:  0.1498, Adjusted R-squared:  0.1448
## F-statistic: 29.49 on 3 and 502 DF,  p-value: < 2.2e-16
```

```r
summary(lm(crim~poly(lstat,3), data=Boston))   # 1,2
```

```
##
## Call:
## lm(formula = crim ~ poly(lstat, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.234  -2.151  -0.486   0.066  83.353
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)       3.6135     0.3392  10.654   <2e-16 ***
## poly(lstat, 3)1  88.0697     7.6294  11.543   <2e-16 ***
## poly(lstat, 3)2  15.8882     7.6294   2.082   0.0378 *
## poly(lstat, 3)3 -11.5740     7.6294  -1.517   0.1299
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.629 on 502 degrees of freedom
## Multiple R-squared:  0.2179, Adjusted R-squared:  0.2133
## F-statistic: 46.63 on 3 and 502 DF,  p-value: < 2.2e-16
```

```r
summary(lm(crim~poly(medv,3), data=Boston))    # 1,2,3
```

```
##
## Call:
## lm(formula = crim ~ poly(medv, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -24.427  -1.976  -0.437   0.439  73.655
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)       3.614      0.292  12.374  < 2e-16 ***
## poly(medv, 3)1  -75.058      6.569 -11.426  < 2e-16 ***
## poly(medv, 3)2   88.086      6.569  13.409  < 2e-16 ***
## poly(medv, 3)3  -48.033      6.569  -7.312 1.05e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.569 on 502 degrees of freedom
## Multiple R-squared:  0.4202, Adjusted R-squared:  0.4167
```

```
## F-statistic: 121.3 on 3 and 502 DF,  p-value: < 2.2e-16
```

Yes, there is evidence for a non-linear relationship between the predictor and response for several variables in the dataset.