

Chapter 3 Exercises

Adrien Osakwe

Conceptual

1)

Table 3.4 shows basic stats from the linear model for the estimation of the number of units sold given the advertising budgets for (the predictors): *TV, radio and newspaper*. The null hypotheses the p-values correspond to are that there is no link between the budgets of each form of advertisement and the number of units sold (\Rightarrow the coefficients of the predictors are equal to 0).

Based on this relationship, the p-values indicate that the null hypotheses for both the TV and radio budgets' relationship to sales can be rejected. However, the null hypothesis for newspaper must be accepted as its p-value is much larger than 0.05. We would therefore conclude that TV and radio budgets are two predictors worth using to estimate sales.

2)

The KNN classifier will be used to estimate a qualitative response, (a category). This functions as a conditional probability where we find the k nearest neighbors to our observation and determine what proportion of the neighbors are in a given class. i.e: If $k-1/k$ of our observation's neighbors are in class Green, then we would classify our observation as green with a conditional probability of $k-1/k$.

The KNN regression method is used to estimate quantitative responses, where the estimation of $f(x)$ will be the average response of the k nearest-neighbors . i.e: if the average of our observation's neighborhood response is 8, then our regression model with k neighbors will estimate the response as 8.

3)

The linear model can be written as follows:

$$y = 50 + 20\text{GPA} + 0.07\text{IQ} + 35\text{Level} + 0.01(\text{GPA} \times \text{IQ}) - 10(\text{GPA} \times \text{Level})$$

We can simplify the model as the following: Level = 1: College graduate $y = 85 + 10\text{GPA} + 0.07\text{IQ} + 0.01(\text{GPA} \times \text{IQ})$ Level = 0: High School Graduate $y = 50 + 20\text{GPA} + 0.07\text{IQ} + 0.01(\text{GPA} \times \text{IQ})$

\rightarrow These models differ at two coefficients: the intercept is 35 units larger for a college graduate and the GPA coefficient is twice as large for the high school graduate

- A) Based on the different coefficients in the two instances in the model, the student with the higher salary is dependent on GPA. Because the intercept is larger for the college instance, the college graduate will have a higher salary for low GPAs. However, when GPA is 3.5, both students have the same salary ($10\text{GPA} = 35$, which equals the difference in the intercept). Therefore, $\text{GPA} > 3.5$ will lead to a higher salary for a high school graduate.

The correct answer is therefore (iii)

- B) The salary will be 137100 dollars. What did they study? How is it so high?!! Who knows...
- C) This is false. Evidence for an interaction effect would be based on the t-statistic and resulting p-value for the interaction's coefficient to know if we can accept the null hypothesis that there is no interaction effect.

4)

- There is not enough information to know as the cubic regression may enable the model to overfit more to any noise in the data. On the other hand, since the linear model is closer to the truth, it may also have a lower RSS.
- We expect it to be lower for the linear regression as its assumption are closest to the true trend, reducing reducible error. The cubic model is more likely to have overfit, increasing reducible error.
- There is not enough information as the true trend may not be cubic either, in which case RSS will also be larger for the cubic regression. If anything, the size of the RSS for either may help give a clue of what kind of non-linear relationship is the truth.
- As in c, little can be said here without knowing what the true relationship is. Therefore, it is most likely higher in the regression furthest from the truth.

5)

Note: I really need to review LaTeX...

$$y(i) = x(i) \sum (x(j)y(j)) / \sum (x^2(i'))$$

$$y(i) = \sum (x(i)x(j)y(j)) / \sum (x^2(i'))$$

$$y(i) = \sum (((x(i)x(j)) / \sum (x^2(i'))) y(j))$$

$$y(i) = \sum (a(j)y(j))$$

$$a(j) = x(i)(j) / \sum (x^2(i'))$$

6)

$$y = b_0 + b_1x$$

$$x = x(\wedge)$$

$$y = b_0 + b_1x(\wedge)$$

$$b_0 = y(\wedge) - b_1x(\wedge)$$

$$y = y(\wedge)$$

7)

```
setwd(getwd())
```

$$R^2 = 1 - \frac{RSS}{TSS}$$

$$\bar{x} = \bar{y} = 0$$

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum y_i^2}$$

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$$

$$\circ \bar{x} = \bar{y} = 0 \Rightarrow \beta_0 = 0$$

$$\hat{y} = \hat{\beta}_1 x$$

$$\beta_1 = \frac{\sum x_i y_i}{\sum x_i^2}$$

$$R^2 = 1 - \frac{\sum \left(y_i - \frac{\sum x_j y_j}{\sum x_j^2} \cdot x_i \right)^2}{\sum y_i^2}$$

$$R^2 = 1 - \frac{\left(\sum y_i^2 - 2 \sum y_i \cdot \left(\frac{\sum x_j y_j}{\sum x_j^2} \right) \cdot x_i + \sum \left(\frac{\sum x_j y_j}{\sum x_j^2} \right)^2 x_i^2 \right)}{\sum y_i^2}$$

$$R^2 = \frac{2 \sum y_i \cdot \left(\frac{\sum x_j y_j}{\sum x_j^2} \right) \cdot x_i - \left(\frac{\sum (x_j y_j)^2}{\sum x_j^4} \right) \cdot x_i^2}{\sum y_i^2}$$

$$\frac{2 \frac{\sum (x_i y_i)^2}{\sum x_i^2} - \frac{\sum (x_i y_i)^2}{\sum x_i^2}}{\sum y_i^2}$$

$$\hookrightarrow R^2 = \frac{\sum (x_i y_i)^2}{\sum x_i^2 \sum y_i^2}$$

Note: $\bar{x} = \bar{y} = 0$

$$\text{Cor}(X, Y) = \frac{\sum x_i y_i}{\sqrt{\sum x_i^2 \cdot \sum y_i^2}}$$

$$\hookrightarrow \text{Cor}^2(X, Y) = R^2$$

Applied

Packages

```
library(ISLR2)
```

```
## Warning: package 'ISLR2' was built under R version 4.1.2
```

8)

A)

- i) The summary of `lm.fit` shows that there is a relationship between mpg and horsepower (very small p-value).
- ii) According to the R^2 value, horsepower accounts for 60% of the variance in mpg. Note: I saw in the solutions manual afterwards that we can also check the percentage error (RSE/mean response) to evaluate relationship strength.
- iii) Negative relationship
- iv) Prediction: $\text{mpg} \sim 24.47$, $\text{CI} = [23.97-24.96]$, $\text{PI} = [14.81-34.12]$ (rounded to 2 d.p.)

```
# Part A
```

```
lm.fit <- lm(mpg ~ horsepower, data = Auto)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = mpg ~ horsepower, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.5710  -3.2592  -0.3435   2.7630  16.9240
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  39.935861   0.717499   55.66  <2e-16 ***
## horsepower  -0.157845   0.006446  -24.49  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.906 on 390 degrees of freedom
## Multiple R-squared:  0.6059, Adjusted R-squared:  0.6049
## F-statistic: 599.7 on 1 and 390 DF, p-value: < 2.2e-16
```

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

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

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

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

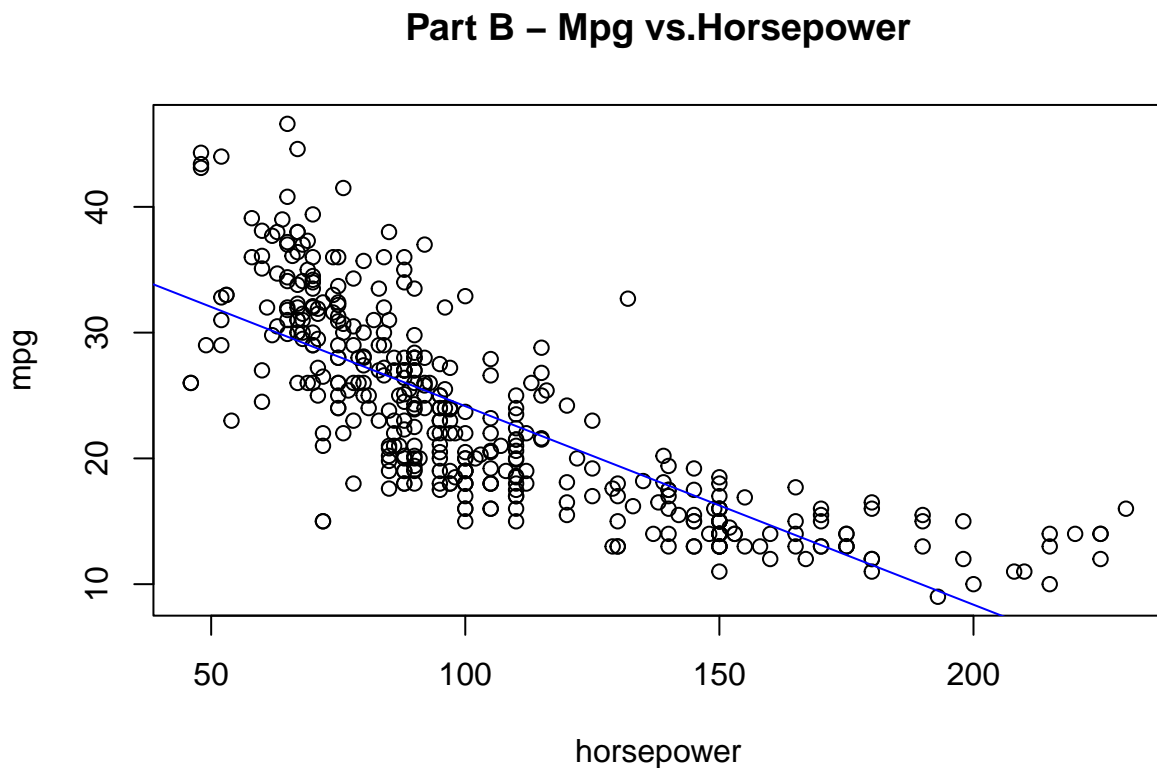
B)

See plot below

```
# Part B  
attach(Auto)  
plot(horsepower,mpg) + title("Part B - Mpg vs.Horsepower")
```

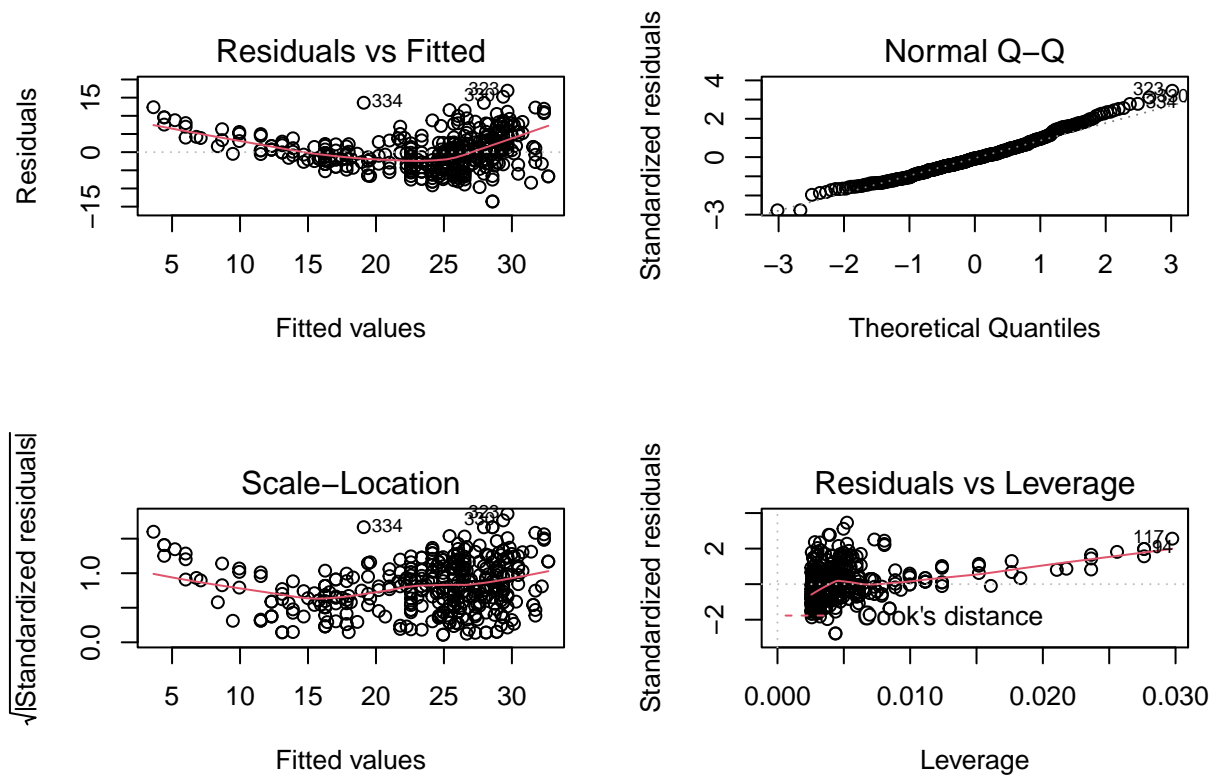
```
## integer(0)
```

```
abline(lm.fit, col = "blue")
```



- C) Based on the residuals v. fitted value plot, we can see that the true relationship is not linear, indicating that some form of transformation will be required to best fit the data. There is not a large fat tail in the Q-Q plot which indicates that the responses are not very non-normal. The remaining plots highlight one or two observations that are outliers.

```
# Part C
par(mfrow = c(2,2))
plot(lm.fit)
```

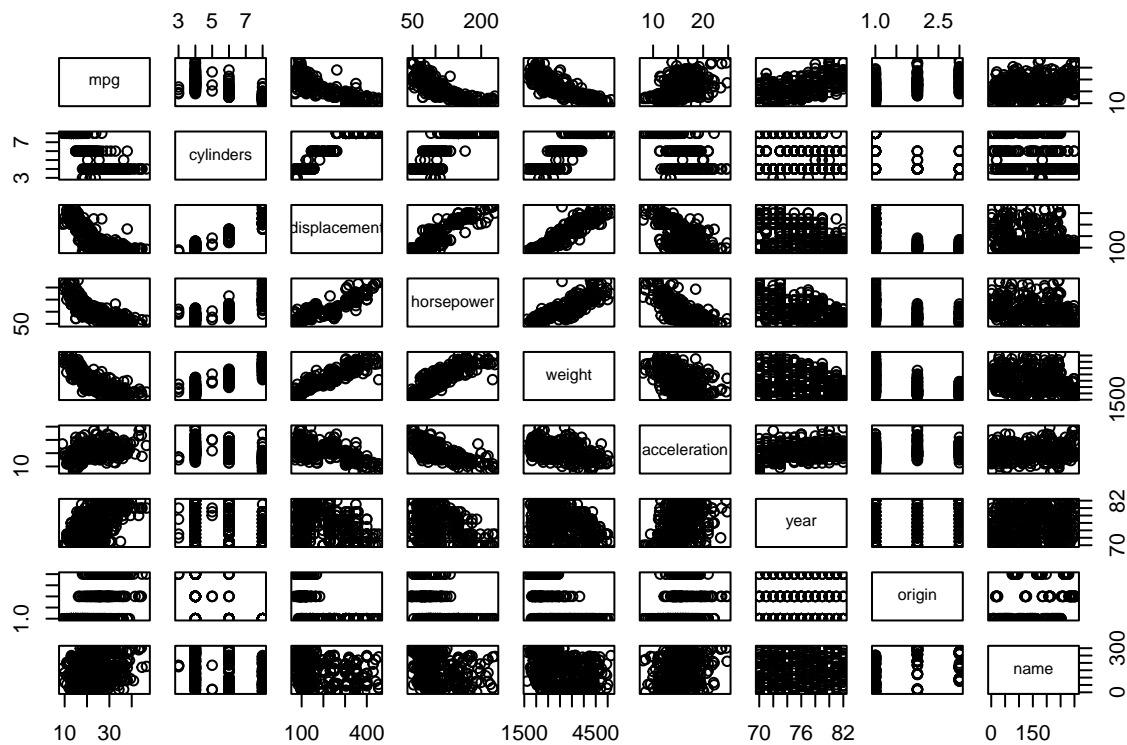


9)

A)

See plot below.

```
#Part A
pairs(Auto)
```



B) See matrix below.

#Part B

```
mat.cor <- cor(Auto[,!(names(Auto) == "name")])
data.frame(mat.cor)
```

```
##           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
```

C) i) The F-statistic is below 0.05 which confirms we can reject the null hypothesis.

- ii) Based on the p-values, only some of the predictors are shown to have a relationship with the response. These are: displacement, weight, year, and origin.
- iii) The coefficient for year suggests that as the year increases, the mpg increases. This is reassuring to see as I would be concerned if more technologically advanced cars were incapable of a better mileage than a model T...

#Part C

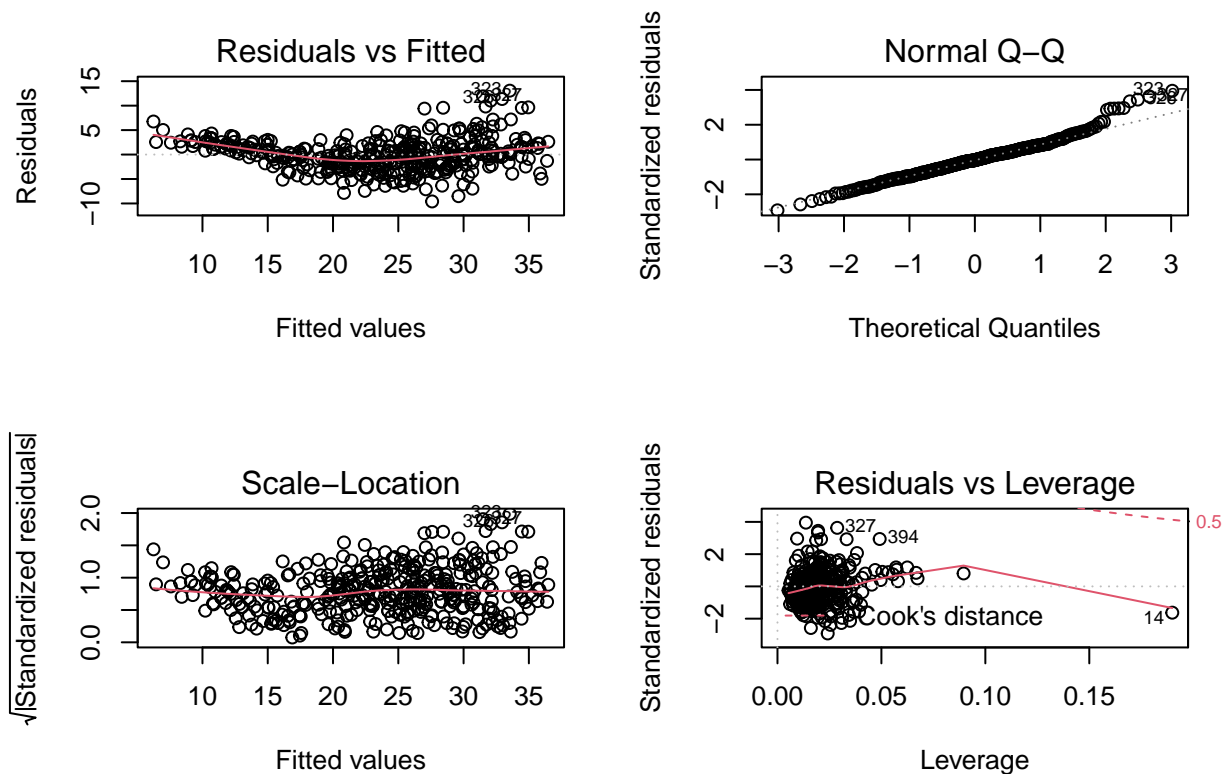
```
lm.fit <- lm(mpg ~ . - name, data = Auto)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = mpg ~ . - name, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -9.5903 -2.1565 -0.1169  1.8690 13.0604
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -17.218435   4.644294  -3.707  0.00024 ***
## cylinders     -0.493376   0.323282  -1.526  0.12780
## displacement  0.019896   0.007515   2.647  0.00844 **
## horsepower   -0.016951   0.013787  -1.230  0.21963
## weight       -0.006474   0.000652  -9.929 < 2e-16 ***
## acceleration  0.080576   0.098845   0.815  0.41548
## year          0.750773   0.050973  14.729 < 2e-16 ***
## origin        1.426141   0.278136   5.127 4.67e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.328 on 384 degrees of freedom
## Multiple R-squared:  0.8215, Adjusted R-squared:  0.8182
## F-statistic: 252.4 on 7 and 384 DF,  p-value: < 2.2e-16
```

- D) The residuals vs leverage plot shows observation 14 to be a considerable outlier compared to the rest of the dataset. Observations 323, 326 and 327 are also shown to be outliers based on the other three plots. The residuals vs. fitted plot seems to have a very light non-linearity. Q-Q plot shows a fat tail indicating some of the data is non-normal.

#Part D

```
par(mfrow = c(2,2))
plot(lm.fit)
```



E) Checked for interactions effects between cylinders and weight as well as horsepower and acceleration. Both interactions are shown to have a significant relationship. In addition this model reduces RSE and increases R^2 . An anova analysis reveals that this model is a significant improvement of the previous one.

```
#Part E
lm.fit2 <- lm(mpg ~ . - name + cylinders:weight + horsepower:acceleration, data = Auto)
summary(lm.fit2)
```

```
##
## Call:
## lm(formula = mpg ~ . - name + cylinders:weight + horsepower:acceleration,
##     data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.3499  -1.6685  -0.0141   1.5009  12.0571
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -7.4837124   5.5164273  -1.357  0.175702
## cylinders      -4.0179263   0.5878598  -6.835 3.26e-11 ***
## displacement  -0.0048855   0.0075637  -0.646  0.518716
## horsepower     0.0806652   0.0236283   3.414  0.000709 ***
## weight        -0.0120372   0.0012008 -10.024 < 2e-16 ***
```

```
## acceleration      0.7885834  0.1514762   5.206 3.16e-07 ***
## year              0.7814074  0.0447175  17.474 < 2e-16 ***
## origin            0.5833654  0.2552730   2.285 0.022845 *
## cylinders:weight   0.0013150  0.0001633   8.051 1.05e-14 ***
## horsepower:acceleration -0.0092849  0.0016782  -5.533 5.86e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.912 on 382 degrees of freedom
## Multiple R-squared:  0.864, Adjusted R-squared:  0.8608
## F-statistic: 269.7 on 9 and 382 DF, p-value: < 2.2e-16
```

```
anova(lm.fit,lm.fit2)
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ (cylinders + displacement + horsepower + weight + acceleration +
##   year + origin + name) - name
## Model 2: mpg ~ (cylinders + displacement + horsepower + weight + acceleration +
##   year + origin + name) - name + cylinders:weight + horsepower:acceleration
##   Res.Df    RSS Df Sum of Sq    F    Pr(>F)
## 1      384 4252.2
## 2      382 3238.4  2    1013.8 59.794 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

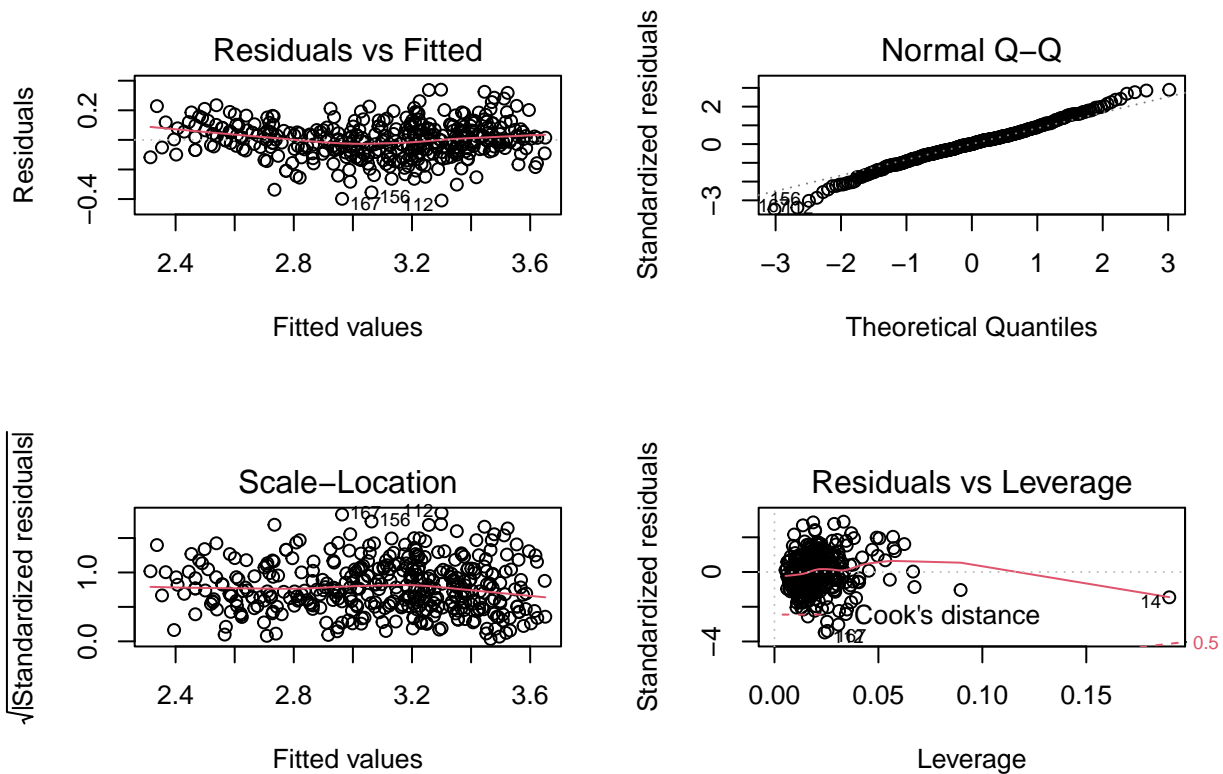
F) Log-transforming mpg seems to better fit the data. RSE is very small and the residuals v. fitted plot shows no trend. A square root transform also reduces R^2 , however the non-linear trend in the residuals vs. fitted plot shows the model does not fit the data as well as the log transformation. See plots below

```
#Part F
log.fit <- lm(log(mpg) ~ . - name, data = Auto)
summary(log.fit)
```

```
##
## Call:
## lm(formula = log(mpg) ~ . - name, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.40955 -0.06533  0.00079  0.06785  0.33925
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.751e+00  1.662e-01  10.533 < 2e-16 ***
## cylinders    -2.795e-02  1.157e-02  -2.415  0.01619 *
## displacement  6.362e-04  2.690e-04   2.365  0.01852 *
## horsepower   -1.475e-03  4.935e-04  -2.989  0.00298 **
## weight       -2.551e-04  2.334e-05 -10.931 < 2e-16 ***
## acceleration -1.348e-03  3.538e-03  -0.381  0.70339
## year         2.958e-02  1.824e-03  16.211 < 2e-16 ***
## origin       4.071e-02  9.955e-03   4.089 5.28e-05 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1191 on 384 degrees of freedom
## Multiple R-squared:  0.8795, Adjusted R-squared:  0.8773
## F-statistic: 400.4 on 7 and 384 DF,  p-value: < 2.2e-16
```

```
par(mfrow = c(2,2))
plot(log.fit)
```

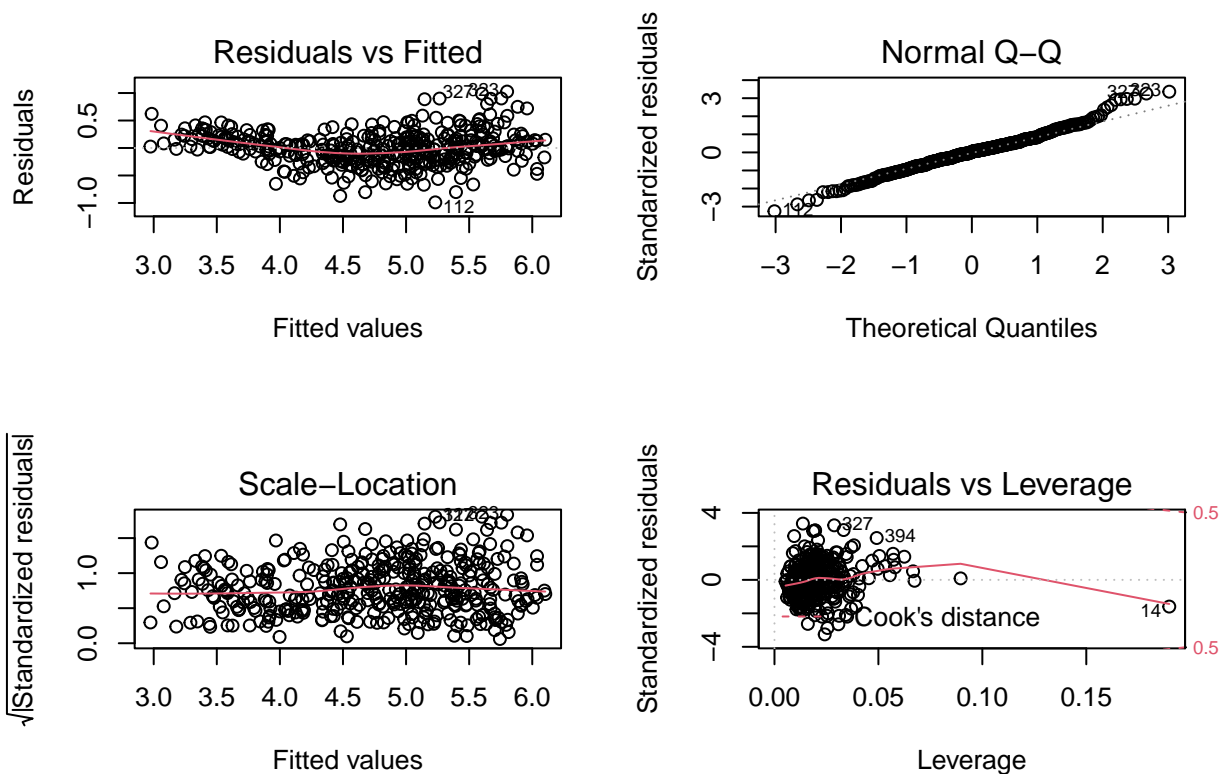


```
sqrfit <- lm(sqrt(mpg) ~ . - name, data = Auto)
summary(sqrfit)
```

```
##
## Call:
## lm(formula = sqrt(mpg) ~ . - name, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.98891 -0.18946  0.00505  0.16947  1.02581
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.075e+00  4.290e-01   2.506  0.0126 *
## cylinders     -5.942e-02  2.986e-02  -1.990  0.0474 *
```

```
## displacement 1.752e-03 6.942e-04 2.524 0.0120 *
## horsepower -2.512e-03 1.274e-03 -1.972 0.0493 *
## weight -6.367e-04 6.024e-05 -10.570 < 2e-16 ***
## acceleration 2.738e-03 9.131e-03 0.300 0.7644
## year 7.381e-02 4.709e-03 15.675 < 2e-16 ***
## origin 1.217e-01 2.569e-02 4.735 3.09e-06 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3074 on 384 degrees of freedom
## Multiple R-squared: 0.8561, Adjusted R-squared: 0.8535
## F-statistic: 326.3 on 7 and 384 DF, p-value: < 2.2e-16
```

```
par(mfrow = c(2,2))
plot(sqrt.fit)
```



10)

A)

```
lm.fit <- lm(Sales ~ Price + Urban + US, data = Carseats)
summary(lm.fit)
```

```
##
```

```
## Call:
## lm(formula = Sales ~ Price + Urban + US, data = Carseats)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.9206 -1.6220 -0.0564  1.5786  7.0581
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 13.043469   0.651012  20.036 < 2e-16 ***
## Price       -0.054459   0.005242 -10.389 < 2e-16 ***
## UrbanYes    -0.021916   0.271650  -0.081  0.936
## USYes       1.200573    0.259042   4.635 4.86e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.472 on 396 degrees of freedom
## Multiple R-squared:  0.2393, Adjusted R-squared:  0.2335
## F-statistic: 41.52 on 3 and 396 DF, p-value: < 2.2e-16
```

B) See above for the summary statistics of the model.

Based on the summary, the F-statistic indicates that we can reject the null hypothesis that all coefficients are 0. The individual p-values show that Urban is the only predictor to not be related to sales. The coefficient of Price tells us that a one unit increase in price will reduce sales by 0.05 units. The coefficient of US tells us that being in the US increases sales by 1.2 units.

C) If in the US: $\text{Sales} = 14.2 - 0.05\text{Price}$ Else: $\text{Sales} = 13 - 0.05\text{Price}$

D) As mentioned in B, the null hypothesis can be rejected for Price and US.

E)

```
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
```

F) The summary stats for both models tell us that the predictors in both cases only account for about 23% of the variance in sales. Based off this alone, we will likely have to use other predictors and/or interaction effects to have a better fit.

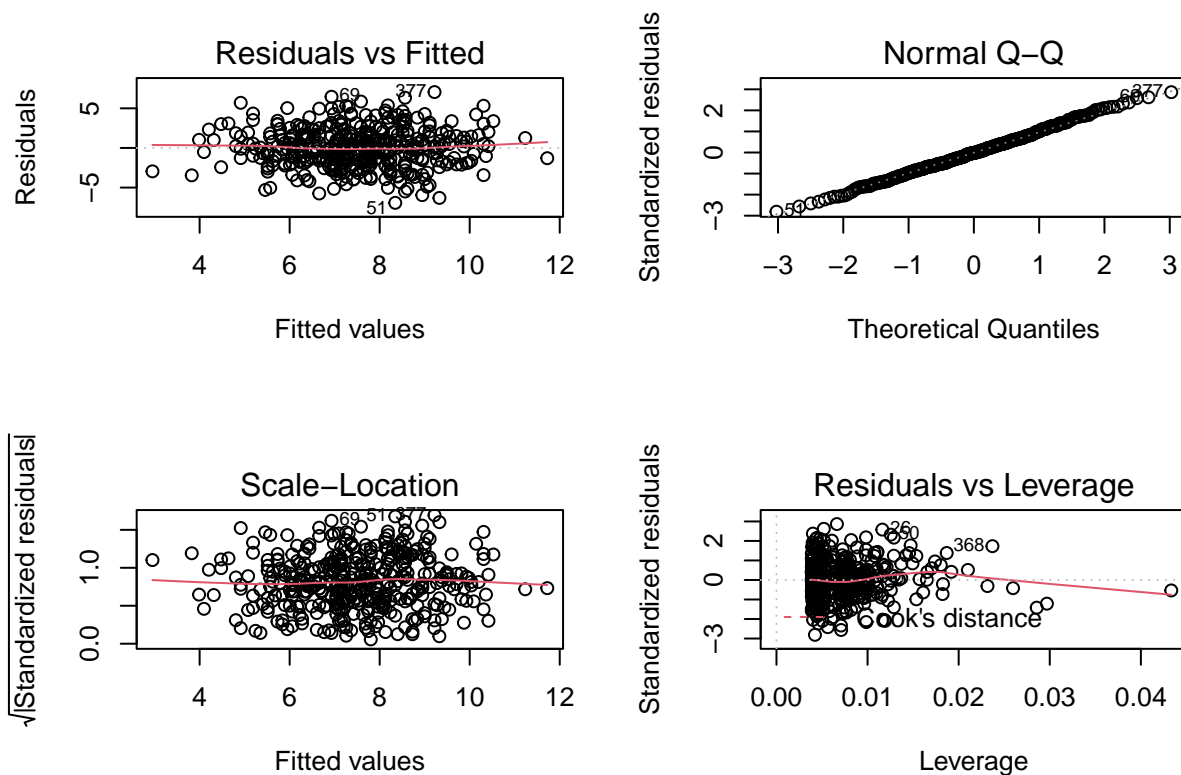
G)

```
confint(lm.fit2)
```

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

H) The diagnostics show some evidence for one outlier (377) and one case of high leverage.

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



11)

A) Coefficients: x

Estimate SE t value Pr(>|t|)

1.9939 0.1065 18.73 <2e-16

These results show that the null hypothesis can be rejected as the p-value is below 0.05.

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

```
##
## 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
```

B) Coefficients: y

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

0.39111 0.02089 18.73 <2e-16

Much like before, we can reject the null hypothesis as p-value is below 0.05.

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

```
##
## 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
```


- C) Both results have the same t-statistic and p-value. This is expected as both models are producing the same line (we simply inverted the roles of predictor and response).
- D)

11) d)

$$t = \hat{\beta} / SE(\hat{\beta})$$

$$\hat{\beta} = \sum x_i y_i / \sum x_i^2$$

$$SE(\hat{\beta}) = \sqrt{\frac{\sum (y_i - x_i \hat{\beta})^2}{(n-1) \sum x_i^2}}$$

$$\text{note: } \sum (y_i - x_i \hat{\beta})^2 = RSS = \frac{\sum y_i^2 \cdot \sum x_i^2 - (\sum x_i y_i)^2}{\sum x_i^2} *$$

$$\hookrightarrow SE(\hat{\beta}) = \sqrt{\frac{\sum y_i^2 \cdot \sum x_i^2 - (\sum x_i y_i)^2}{(n-1) \cdot (\sum x_i^2)^2}} = \frac{\sqrt{\sum y_i^2 \cdot \sum x_i^2 - (\sum x_i y_i)^2}}{\sqrt{n-1} \cdot \sum x_i^2}$$

$$t = \hat{\beta} / SE(\hat{\beta}) = \frac{\sum x_i y_i / \sum x_i^2}{\frac{\sqrt{\sum y_i^2 \cdot \sum x_i^2 - (\sum x_i y_i)^2}}{\sqrt{n-1} \cdot \sum x_i^2}}$$

$$t = \frac{\sqrt{n-1} \cdot \sum x_i y_i}{\sqrt{\sum y_i^2 \cdot \sum x_i^2 - (\sum x_i y_i)^2}}$$

* RSS derivation:

$$RSS = \sum (y_i - x_i \hat{\beta})^2$$

$$= \sum y_i^2 - 2\hat{\beta} \sum x_i y_i + \hat{\beta}^2 \sum x_i^2$$

$$= \sum y_i^2 - 2 \frac{\sum x_i y_i \cdot \sum x_i y_i}{\sum x_i^2} + \frac{(\sum x_i y_i)^2}{\sum x_i^2}$$

$$= \sum y_i^2 - \frac{(\sum x_i y_i)^2}{\sum x_i^2} = \frac{\sum y_i^2 \sum x_i^2 - (\sum x_i y_i)^2}{\sum x_i^2}$$

```
true_t <- 18.73
test_t <- (sqrt(length(x)-1)*sum(x %>% y))/(sqrt(sum(y**2)*sum(x**2)-sum(x %>% y)**2))
true_t == round(test_t,2)
```

```
## [1] TRUE
```

E) Based on the derived formula, and that there are n observation of x and n observations of y, we know that the t-statistic is the same for regressions of x onto y and y onto x. This is because the formula shows that the values of x and y undergo the exact same transformations in the formula whether they are the response or the predictor. Therefore their exact role in the regression is irrelevant as it will lead to the same value.

F)

```
set.seed(1)
x <- rnorm(100)
y <- 2*x + rnorm(100)
lm.fit <- lm(x ~ y)
lm.fit2 <- lm(y ~ x)
round(summary(lm.fit)$coefficients[,3][2],2) == round(summary(lm.fit2)$coefficients[,3][2],2)
```

```
##      y
## TRUE
```

12)

A) The equation for the coefficients are:

y onto x:

$$\text{Sum}(x \% \% y) / \text{sum}(x^{**}2)$$

x onto y:

$$\text{Sum}(y \% \% x) / \text{sum}(y^{**}2)$$

Based on the above, the conditions are the same if x and y are equal.

B)

```
set.seed(1)
x <- rnorm(100)
y <- 2*x + rnorm(100)
lm.fit <- lm(x ~ y)
lm.fit2 <- lm(y ~ x)
round(lm.fit$coefficients[2],2) == round(lm.fit2$coefficients[2],2)
```

```
##      y
## FALSE
```

C)

```

set.seed(1)
x <- rnorm(100)
y <- x
lm.fit <- lm(x ~ y)
lm.fit2 <- lm(y ~ x)
round(lm.fit$coefficients[2],2) == round(lm.fit2$coefficients[2],2)

##      y
## TRUE

```

13)

```

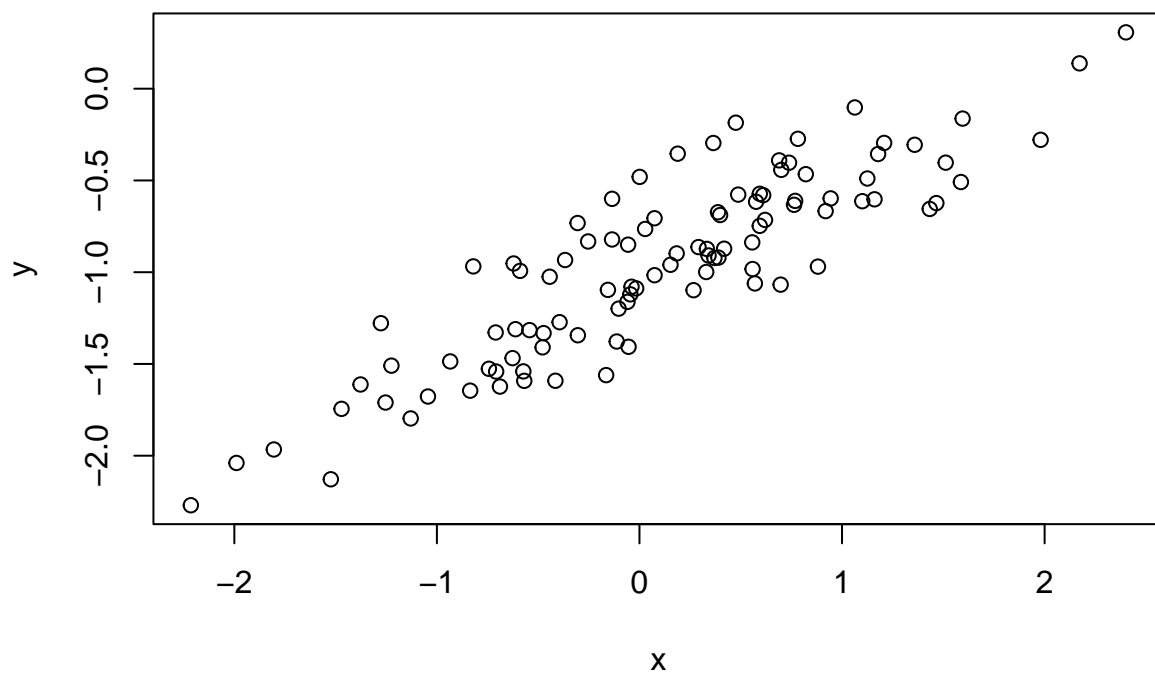
set.seed(1)
x <- rnorm(100)
eps <- rnorm(100,0,0.25)
y <- -1 + 0.5*x + eps

```

C) Y has a length of 100 observations, with B0 as -1 and B1 as 0.5.

D) Scatterplot (see below) shows a positive linear correlation between x and y.

```
plot(x,y)
```



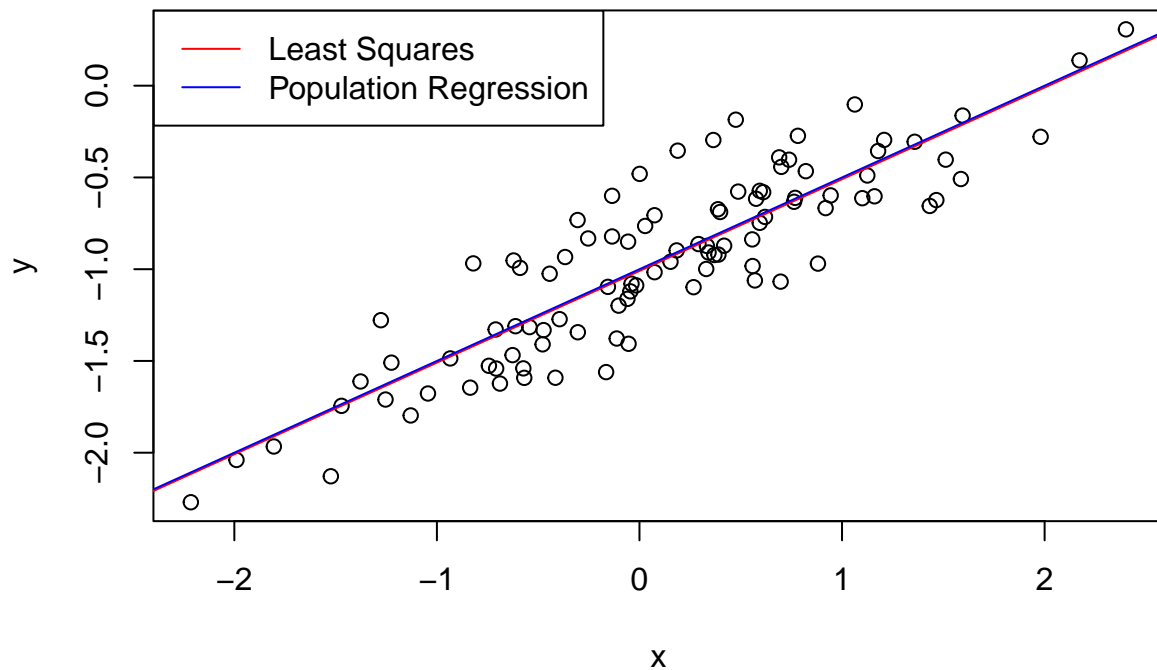
E) The least squares linear model (below) finds coefficients that closely resemble their true values.

```
lm.fit <- lm(y ~ x)
summary(lm.fit)

##
## 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
```

F) We can see below that the model seems to fit the data very well as it matches the population regression line.

```
plot(x,y)
abline(lm.fit, col = "red")
abline(-1,0.5, col = "blue")
legend("topleft",c("Least Squares", "Population Regression"), col = c("red", "blue"), lty = c(1,1))
```



G) Running anova on the two models indicates that there is no significant improvement when including X^2 in the model.

```
lm.fit2 <- lm(y ~ x + I(x**2))
anova(lm.fit,lm.fit2)
```

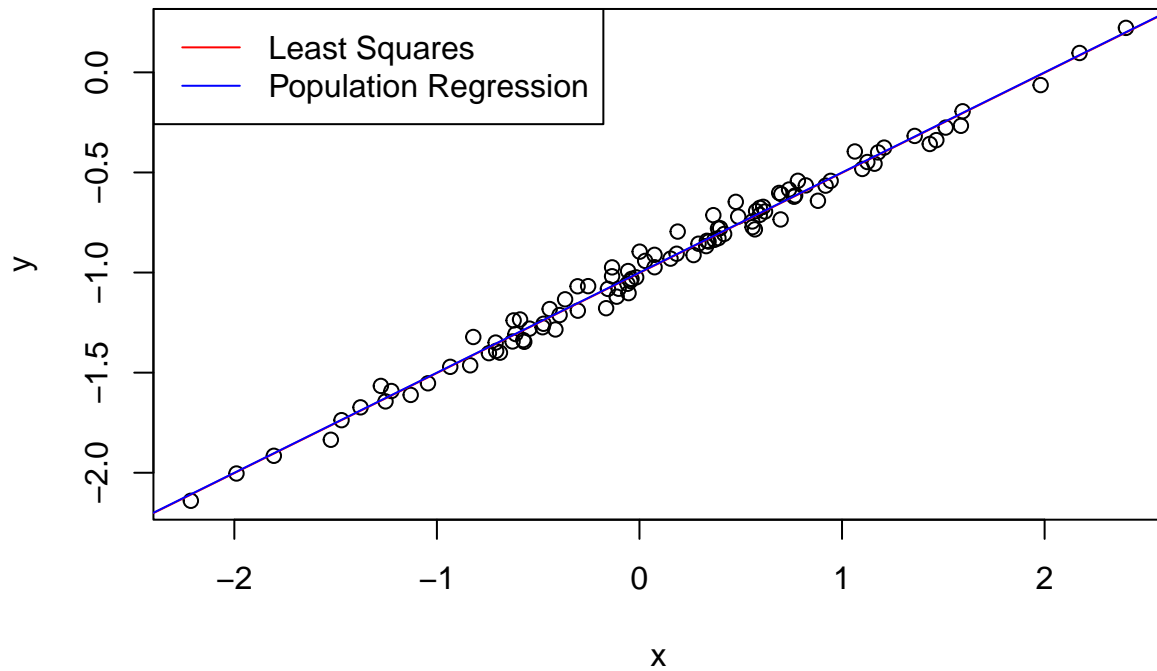
```
## Analysis of Variance Table
##
## Model 1: y ~ x
## Model 2: y ~ x + I(x^2)
##   Res.Df    RSS Df Sum of Sq    F Pr(>F)
## 1      98 5.6772
## 2      97 5.5643   1   0.11291 1.9682 0.1638
```

H) Reduced the noise of data by generating the epsilon vector using a normal distribution with a smaller standard deviation.

This, as expected, generates a model that matches the trend of the test data much more than the model used on the noisier data.

```
set.seed(1)
x <- rnorm(100)
eps <- rnorm(100,0,0.05)
y <- -1 + 0.5*x + eps
lm.fit2 <- lm(y ~ x)
plot(x,y)
```

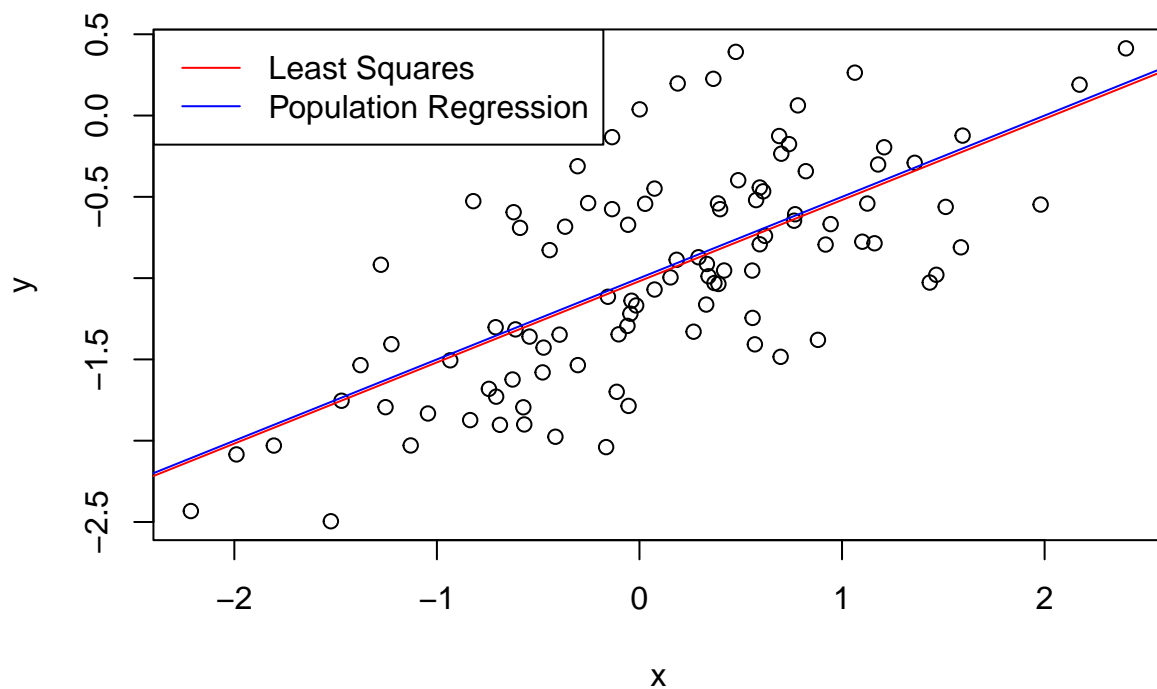
```
abline(lm.fit2, col = "red")
abline(-1,0.5, col = "blue")
legend("topleft",c("Least Squares", "Population Regression"), col = c("red", "blue"), lty = c(1,1))
```



I) Increasing standard deviation in the normal distribution used for ϵ increases noise.

As expected, there is much more deviation between the observed responses and what our model predicts.

```
set.seed(1)
x <- rnorm(100)
eps <- rnorm(100,0,0.5)
y <- -1 + 0.5*x + eps
lm.fit3 <- lm(y ~ x)
plot(x,y)
abline(lm.fit3, col = "red")
abline(-1,0.5, col = "blue")
legend("topleft",c("Least Squares", "Population Regression"), col = c("red", "blue"), lty = c(1,1))
```



J)

As seen below, the range of our CI increases with noise for beta1.

```
confint(lm.fit)
```

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

```
confint(lm.fit2)
```

```
##                2.5 %    97.5 %
## (Intercept) -1.0115080 -0.9922612
## x           0.4892579  0.5106360
```

```
confint(lm.fit3)
```

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

14)

A) The model has the form:

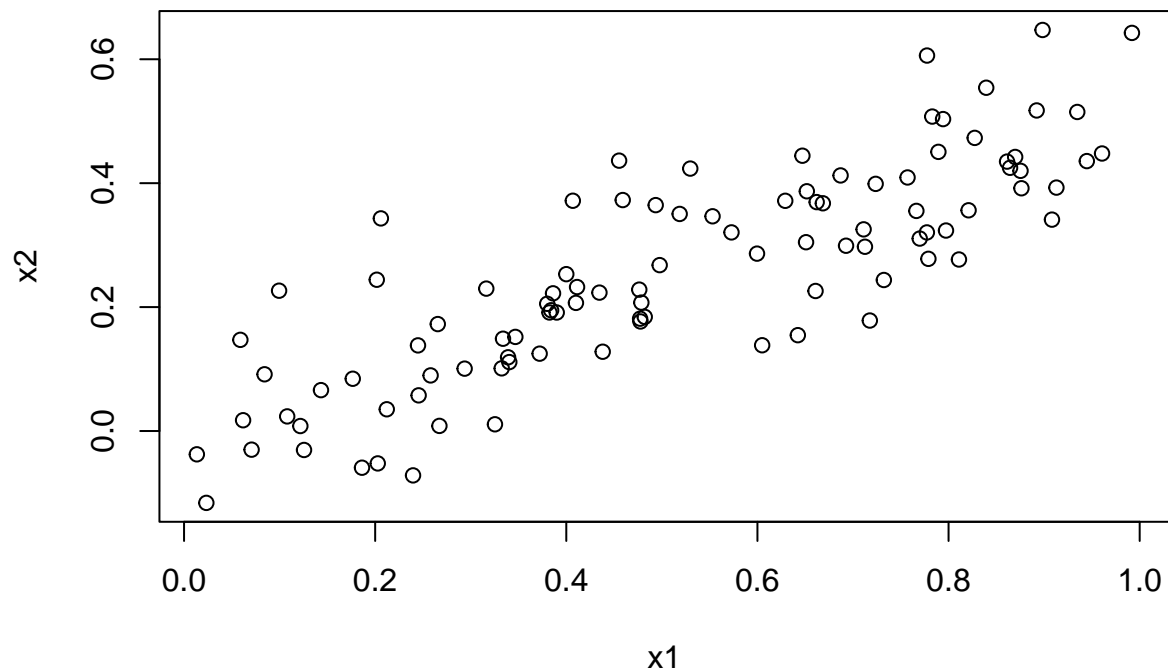
$$y = 2 + 2x_1 + 0.3x_2 + e$$

$$b_0 = 2, b_1 = 2, b_2 = 0.3$$

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

B) x_1 and x_2 share a positive correlation (every unit increase in x_1 increases x_2 by 0.5 units)

```
plot(x1,x2)
```



C) F-statistic indicates that we can reject the null hypothesis and conclude that there is a link between the variables and y . The p values from the t-stats tell us that we can reject the null hypothesis for x_1 but must accept it for x_2 as its p-value is greater than 0.05.

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

```
##
## 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
```

D) Removing x_2 barely changes R^2 and the RSE and causes the F-statistic to increase. We can still reject the null hypothesis that $b_1 = 0$.

```
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
```

E) When we use x_2 instead of x_1 , we can also reject the null hypothesis for b_1 , giving more evidence that these two variables are collinear.

```
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
```

F) At first glance, one would say that the results are contradictory. However, given what we know about collinearity and that we know x_2 depends on x_1 , these results make sense as x_2 provides little information to the model when x_1 is already present but is meaningful if it is not.

G) This mismeasured data point now masks the importance of x_1 and unmaskes that of x_2 in the MLR.

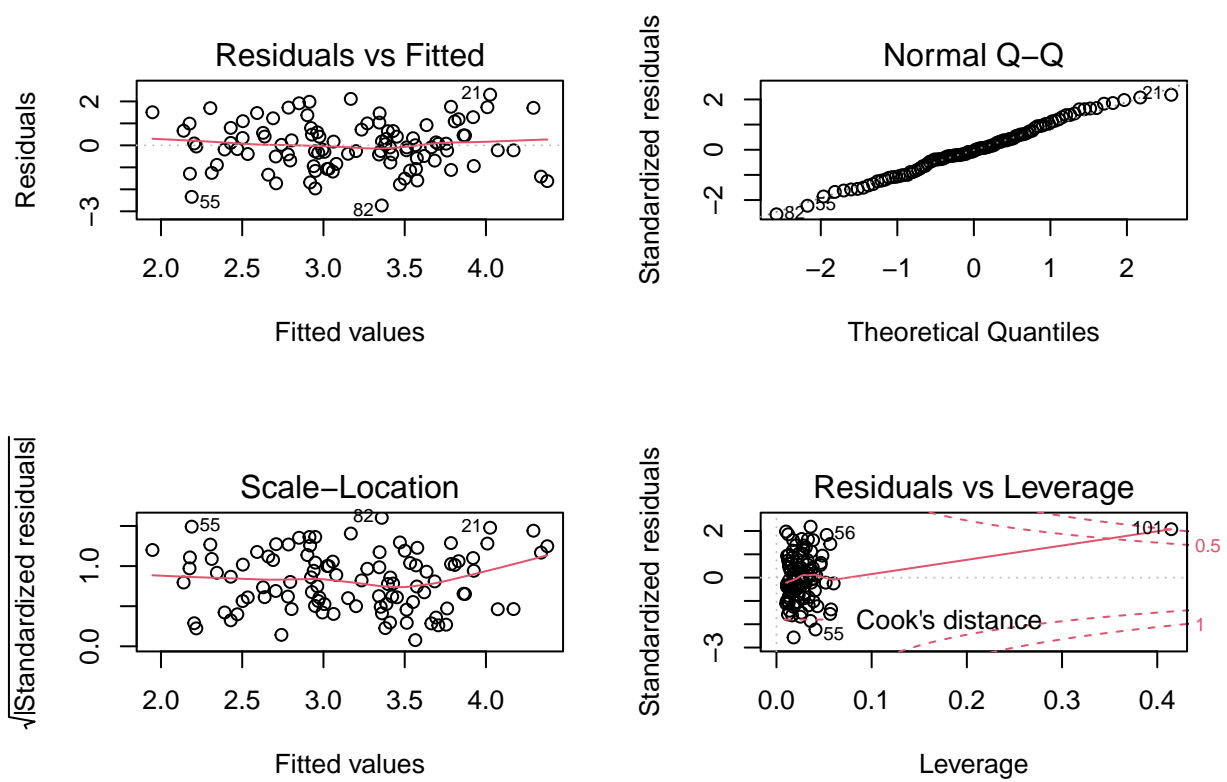
Diagnostics of the three new models show the observation to be an outlier and /or a high leverage point.

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

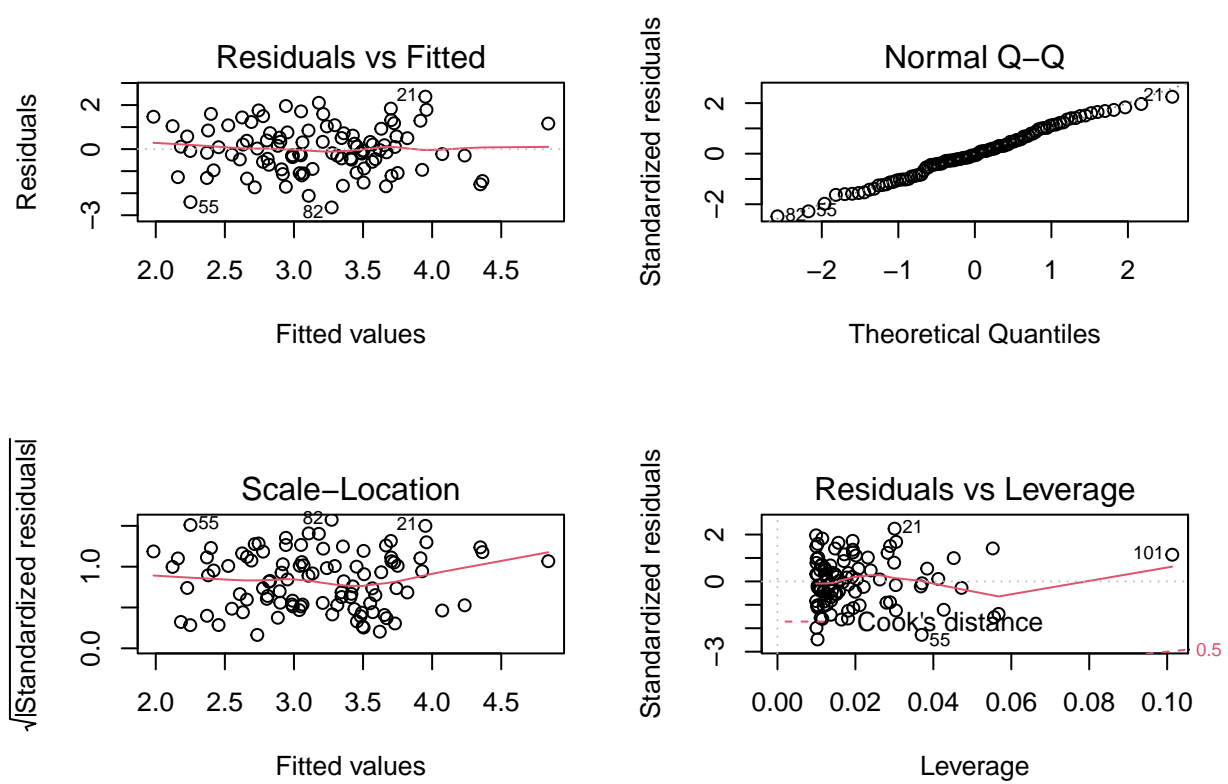
```
lm.fit4 <- lm(y ~ x1 + x2)
summary(lm.fit4)
```

```
##
## 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
```

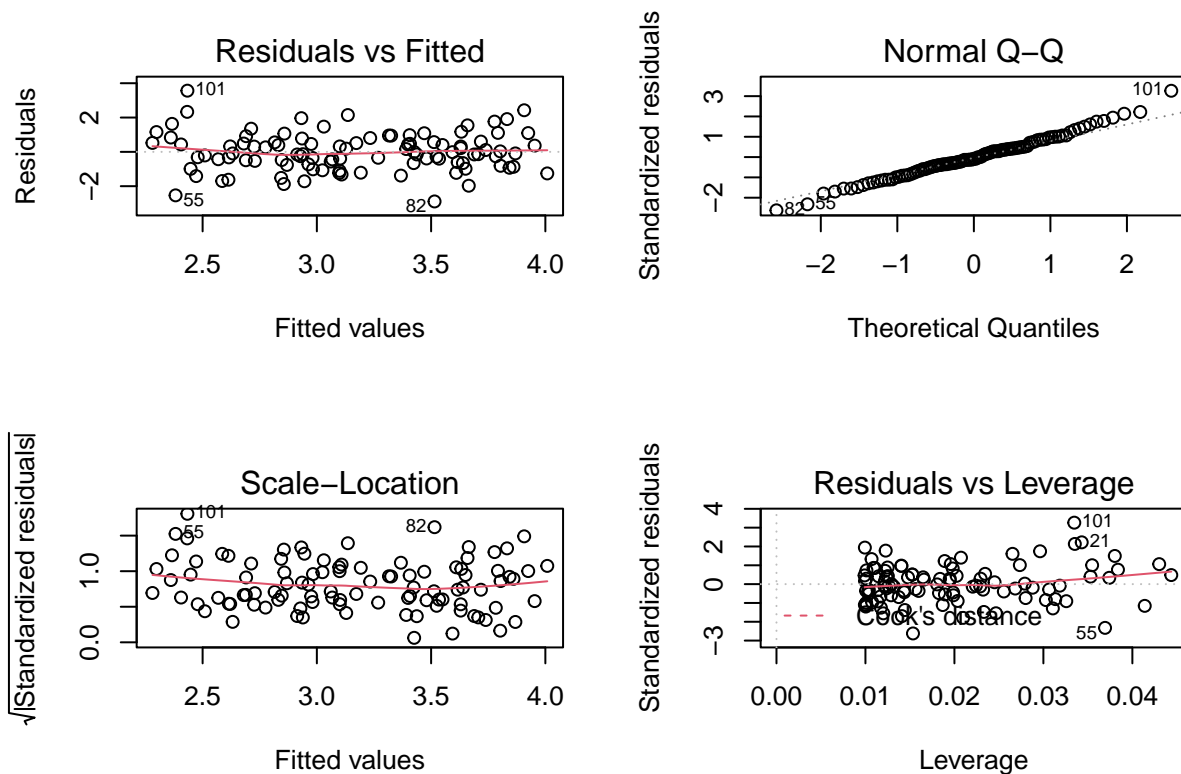
```
lm.fit5 <- lm(y ~ x2)
lm.fit6 <- lm(y ~ x1)
par(mfrow = c(2,2))
plot(lm.fit4)
```



```
par(mfrow = c(2,2))
plot(lm.fit5)
```



```
par(mfrow = c(2,2))
plot(lm.fit6)
```



15)

A) Based on the summary stats, all of the variables can have their null hypothesis rejected except for the predictor “chas” for which the p-values for the corresponding t and f statistics are above 0.05.

The predictor “rad” explained the most variance.

```
models <- lapply(Boston[,-1],function(a) lm(crim ~ a,data = Boston))
```

```
lapply(models,function(a) summary(a))
```

```
## $zn
##
## Call:
## lm(formula = crim ~ a, 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 ***
## a           -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
##
##
## $indus
##
## Call:
## lm(formula = crim ~ a, 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 **
## a           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
##
##
## $chas
##
## Call:
## lm(formula = crim ~ a, 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 ***
## a            -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
##
##
## $nox
##
## Call:
## lm(formula = crim ~ a, 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 ***
## a              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
##
##
## $rm
##
## Call:
## lm(formula = crim ~ a, 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 ***
## a             -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
##
##
## $age
##
## Call:
## lm(formula = crim ~ a, 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 ***
## a             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
##
##
## $dis
##
## Call:
## lm(formula = crim ~ a, 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 ***
## a            -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
##
##
## $rad
##
## Call:
## lm(formula = crim ~ a, 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 ***
## a             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
##
##
## $tax
##
## Call:
## lm(formula = crim ~ a, 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 ***
## a           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
##
##
## $ptratio
##
## Call:
## lm(formula = crim ~ a, 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 ***
## a           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
##
##
## $lstat
##
## Call:
## lm(formula = crim ~ a, 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 ***
## a           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
##

```

```
##
## $medv
##
## Call:
## lm(formula = crim ~ a, 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 ***
## a           -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
```

B) Based on the summary stats, we can reject the null hypothesis for the coefficients of: zn,dis,rad and medv.

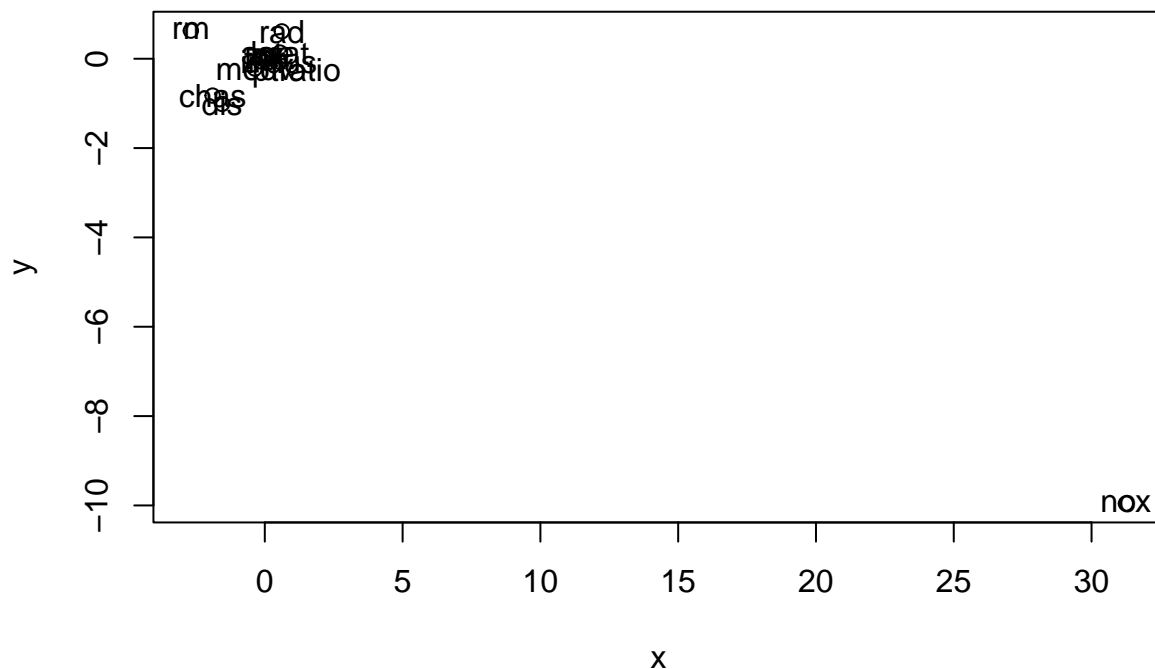
```
mlr.fit <- lm(crim ~., data = Boston)
summary(mlr.fit)
```

```
##
## Call:
## lm(formula = crim ~ ., data = Boston)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -8.534 -2.248 -0.348  1.087 73.923
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 13.7783938  7.0818258   1.946 0.052271 .
## zn           0.0457100  0.0187903   2.433 0.015344 *
## indus       -0.0583501  0.0836351  -0.698 0.485709
## chas        -0.8253776  1.1833963  -0.697 0.485841
## nox         -9.9575865  5.2898242  -1.882 0.060370 .
## rm           0.6289107  0.6070924   1.036 0.300738
## age         -0.0008483  0.0179482  -0.047 0.962323
## dis         -1.0122467  0.2824676  -3.584 0.000373 ***
## rad          0.6124653  0.0875358   6.997 8.59e-12 ***
## tax         -0.0037756  0.0051723  -0.730 0.465757
## ptratio     -0.3040728  0.1863598  -1.632 0.103393
## lstat        0.1388006  0.0757213   1.833 0.067398 .
## medv        -0.2200564  0.0598240  -3.678 0.000261 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 6.46 on 493 degrees of freedom
## Multiple R-squared:  0.4493, Adjusted R-squared:  0.4359
## F-statistic: 33.52 on 12 and 493 DF,  p-value: < 2.2e-16
```

C) For most predictors, our plot is showing a similar coefficient. However, one predictor, nox, has drastically different coefficients. This is an indicator of a possible interaction effect with nox and another variable. As the slr only consider the effect of the one predictor, and the mlr considers the effect of one predictor at fixed values of the other predictors, the model would not have been able to account for this interaction as is. We could look at a matrix of correlations to see if there is additional evidence there (need to remove chas which is a qualitative predictor). Sure enough there are some high correlations for some of these predictors.

```
x <- sapply(models, function(a) summary(a)$coefficients[2,1])
y <- summary(mlr.fit)$coefficients[-1,1]
plot(x,y)
text(x,y, labels = names(models))
```



```
cor(Boston[-c(1,4)])
```

```
##           zn      indus      nox      rm      age      dis
## zn      1.0000000 -0.5338282 -0.5166037  0.3119906 -0.5695373  0.6644082
## indus  -0.5338282  1.0000000  0.7636514 -0.3916759  0.6447785 -0.7080270
## nox    -0.5166037  0.7636514  1.0000000 -0.3021882  0.7314701 -0.7692301
## rm      0.3119906 -0.3916759 -0.3021882  1.0000000 -0.2402649  0.2052462
```

```
## age      -0.5695373  0.6447785  0.7314701 -0.2402649  1.0000000 -0.7478805
## dis      0.6644082 -0.7080270 -0.7692301  0.2052462 -0.7478805  1.0000000
## rad     -0.3119478  0.5951293  0.6114406 -0.2098467  0.4560225 -0.4945879
## tax     -0.3145633  0.7207602  0.6680232 -0.2920478  0.5064556 -0.5344316
## ptratio -0.3916785  0.3832476  0.1889327 -0.3555015  0.2615150 -0.2324705
## lstat   -0.4129946  0.6037997  0.5908789 -0.6138083  0.6023385 -0.4969958
## medv     0.3604453 -0.4837252 -0.4273208  0.6953599 -0.3769546  0.2499287
##          rad      tax      ptratio      lstat      medv
## zn      -0.3119478 -0.3145633 -0.3916785 -0.4129946  0.3604453
## indus    0.5951293  0.7207602  0.3832476  0.6037997 -0.4837252
## nox      0.6114406  0.6680232  0.1889327  0.5908789 -0.4273208
## rm      -0.2098467 -0.2920478 -0.3555015 -0.6138083  0.6953599
## age      0.4560225  0.5064556  0.2615150  0.6023385 -0.3769546
## dis     -0.4945879 -0.5344316 -0.2324705 -0.4969958  0.2499287
## rad      1.0000000  0.9102282  0.4647412  0.4886763 -0.3816262
## tax      0.9102282  1.0000000  0.4608530  0.5439934 -0.4685359
## ptratio  0.4647412  0.4608530  1.0000000  0.3740443 -0.5077867
## lstat    0.4886763  0.5439934  0.3740443  1.0000000 -0.7376627
## medv    -0.3816262 -0.4685359 -0.5077867 -0.7376627  1.0000000
```

D) Briefly skimming over the results shows certain predictors to have polynomial coefficients whose null hypothesis were rejected. Some of these are nox, indus, ptratio and medv.

```
models2 <- lapply(Boston[,-1],function(a) lm(crim ~ a + I(a**2) + I(a**3),data = Boston))
lapply(models2,function(a) summary(a))
```

```
## $zn
##
## Call:
## lm(formula = crim ~ a + I(a^2) + I(a^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)  4.846e+00  4.330e-01  11.192  < 2e-16 ***
## a           -3.322e-01  1.098e-01  -3.025  0.00261 **
## I(a^2)        6.483e-03  3.861e-03   1.679  0.09375 .
## I(a^3)       -3.776e-05  3.139e-05  -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
##
##
## $indus
##
## Call:
## lm(formula = crim ~ a + I(a^2) + I(a^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)  3.6625683  1.5739833   2.327  0.0204 *
## a           -1.9652129  0.4819901  -4.077 5.30e-05 ***
## I(a^2)        0.2519373  0.0393221   6.407 3.42e-10 ***
## I(a^3)       -0.0069760  0.0009567  -7.292 1.20e-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
##
##
## $chas
##
## Call:
## lm(formula = crim ~ a + I(a^2) + I(a^3), data = Boston)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.738 -3.661 -3.435  0.018 85.232
##
## Coefficients: (2 not defined because of singularities)
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.7444      0.3961   9.453 <2e-16 ***
## a            -1.8928      1.5061  -1.257  0.209
## I(a^2)         NA         NA      NA      NA
## I(a^3)         NA         NA      NA      NA
## ---
## 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
##
##
## $nox
##
## Call:
## lm(formula = crim ~ a + I(a^2) + I(a^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)  233.09      33.64   6.928 1.31e-11 ***
```

```

## a          -1279.37      170.40   -7.508 2.76e-13 ***
## I(a^2)      2248.54      279.90    8.033 6.81e-15 ***
## I(a^3)     -1245.70      149.28   -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
##
##
## $rm
##
## Call:
## lm(formula = crim ~ a + I(a^2) + I(a^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)  112.6246    64.5172   1.746  0.0815 .
## a           -39.1501    31.3115  -1.250  0.2118
## I(a^2)         4.5509     5.0099   0.908  0.3641
## I(a^3)        -0.1745     0.2637  -0.662  0.5086
## ---
## 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
##
##
## $age
##
## Call:
## lm(formula = crim ~ a + I(a^2) + I(a^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) -2.549e+00  2.769e+00  -0.920  0.35780
## a            2.737e-01  1.864e-01   1.468  0.14266
## I(a^2)       -7.230e-03  3.637e-03  -1.988  0.04738 *
## I(a^3)        5.745e-05  2.109e-05   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
##
##
## $dis
##
## Call:
## lm(formula = crim ~ a + I(a^2) + I(a^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)  30.0476     2.4459  12.285 < 2e-16 ***
## a          -15.5543     1.7360  -8.960 < 2e-16 ***
## I(a^2)         2.4521     0.3464   7.078 4.94e-12 ***
## I(a^3)        -0.1186     0.0204  -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
##
##
## $rad
##
## Call:
## lm(formula = crim ~ a + I(a^2) + I(a^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) -0.605545     2.050108  -0.295   0.768
## a             0.512736     1.043597   0.491   0.623
## I(a^2)       -0.075177     0.148543  -0.506   0.613
## I(a^3)        0.003209     0.004564   0.703   0.482
##
## 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
##
##
## $tax
##
## Call:
## lm(formula = crim ~ a + I(a^2) + I(a^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)  1.918e+01  1.180e+01  1.626  0.105
## a           -1.533e-01  9.568e-02 -1.602  0.110
## I(a^2)       3.608e-04  2.425e-04  1.488  0.137
## I(a^3)      -2.204e-07  1.889e-07 -1.167  0.244
##
## 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
##
##
## $ptratio
##
## Call:
## lm(formula = crim ~ a + I(a^2) + I(a^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) 477.18405  156.79498   3.043  0.00246 **
## a          -82.36054   27.64394  -2.979  0.00303 **
## I(a^2)       4.63535    1.60832   2.882  0.00412 **
## I(a^3)      -0.08476    0.03090  -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
##
##
## $lstat
##
## Call:
## lm(formula = crim ~ a + I(a^2) + I(a^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)  1.2009656  2.0286452   0.592  0.5541
## a          -0.4490656  0.4648911  -0.966  0.3345
## I(a^2)       0.0557794  0.0301156   1.852  0.0646 .
## I(a^3)      -0.0008574  0.0005652  -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
##
##
## $medv
##
## Call:
## lm(formula = crim ~ a + I(a^2) + I(a^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) 53.1655381  3.3563105  15.840 < 2e-16 ***
## a          -5.0948305  0.4338321 -11.744 < 2e-16 ***
## I(a^2)       0.1554965  0.0171904   9.046 < 2e-16 ***
## I(a^3)      -0.0014901  0.0002038  -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
```