ISLR | Chapter 3 Exercises

Marshall McQuillen

1/11/2018

Conceptual

1

The null hypothesis that the P-values in Table 3.4 (reproduced below) are calculated on is that at least one of the coefficients to all the variables in the model (TV, Radio and Newspaper) are zero. That is…

$$H_0: \quad \beta_{tv},\ \beta_{radio}\ or\ \beta_{newspaper} = 0$$

and the alternate hypothesis is that at least one of the coefficients is non-zero…

$$H_A: \quad \beta_{tv},\ \ \beta_{radio}\ \ or\ \ \beta_{newspaper}0$$

The conclusions that can be drawn from the P-values in the table are that *TV* and *Radio* have some relationship with *Sales* with practically complete certainty (the probability of observing those t-statistic's by chance under the null hypotheis is less than 0.01%). On the other hand, we would expect to observe a t-statistic greater than or equal to that of *Newspaper's* 85.99% of the time, and thus we fail to reject the null hypothesis that *Newspaper* has any effect on *Sales.*

Variable	Coefficient	Std_Error	t_statistic	p_value
Intercept	2.939	0.3119	9.42	< 0.0001
TV	0.046	0.0014	32.81	< 0.0001
radio	0.189	0.0086	21.89	< 0.0001
Newspaper	-0.001	0.0059	-0.18	0.8599

2

The main difference between the *KNN classifier* and *KNN regression* methods is that the *KNN classifier* outputs a qualitative prediction and *KNN regression* ouputs a quantitative prediction.

Mathematically, *KNN classification* takes the K nearest training observations to test obersvation x_0, and takes a majority vote on which class x_0 will be. For example, if you set $K = 5$, and you have two possible classes, *A* or *B*, *KNN classifcation* takes the 5 training observations closest to your test observation x_0, say 3 *A*'s and 2 *B*'s, and classifies x_0 as *A* because there are more *A*'s in the 5 nearest neighbors than *B*'s.

KNN regression takes the *average* of the K nearest neighbors' *quantitative output.* For example, again using the example where $K = 5$, if the 5 training observations closest to x_0 have respective response values of 16, 22, 14, 24 and 18, then *KNN regression* takes their average and gives the test observation x_0 that value…

$$\frac{16 + 22 + 14 + 24 + 18}{5} = 18.8 = y_0$$

3

To answer this question, the first step I took was to write out and simplify the model:

$$\hat{y_i} \;=\; \hat{\beta_o} \;+\; \hat{\beta_1}X_1 \;+\; \hat{\beta_2}X_2 \;+\; \hat{\beta_3}X_3 \;+\; \hat{\beta_4}X_4 \;+\; \hat{\beta_5}X_5$$

Which, rewritten to express the interaction terms is:

$$\hat{y_i} \;=\; \hat{\beta_o} \;+\; \hat{\beta_1}X_1 \;+\; \hat{\beta_2}X_2 \;+\; \hat{\beta_3}X_3 \;+\; \hat{\beta_4}X_1X_2 \;+\; \hat{\beta_5}X_1X_3$$

Now, X_3 is a binary variable, which means we can "split" this equation into two separate equations, based on the value of X_3, where a value of 1 = Female and 0 = Male. Thus:

$$\hat{y_i} \;=\; \hat{\beta_o} \;+\; \hat{\beta_1}X_1 \;+\; \hat{\beta_2}X_2 \;+\; \hat{\beta_3} \;+\; \hat{\beta_4}X_1X_2 \;+\; \hat{\beta_5}X_1X_3 \quad if\ x_i\ is\ Female$$

$$\hat{y_i} \;=\; \hat{\beta_o} \;+\; \hat{\beta_1}X_1 \;+\; \hat{\beta_2}X_2 \;+\; 0 \;+\; \hat{\beta_4}X_1X_2 \;+\; \hat{\beta_5}X_1X_3 \quad if\ x_i\ is\ Male$$

X_5 is an interaction term between Gender and GPA, so if Gender = Male (that is to say $X_3 = 0$) we can rewrite the above Male equation as:

$$\hat{y_i} \;=\; \hat{\beta_o} \;+\; \hat{\beta_1}X_1 \;+\; \hat{\beta_2}X_2 \;+\; 0 \;+\; \hat{\beta_4}X_1X_2 \;+\; 0 \quad if\ x_i\ is\ Male$$

And the Female equation, where $X_3 = 1$, becomes:

$$\hat{y_i} \;=\; \hat{\beta_o} \;+\; \hat{\beta_1}X_1 \;+\; \hat{\beta_2}X_2 \;+\; \hat{\beta_3} \;+\; \hat{\beta_4}X_1X_2 \;+\; \hat{\beta_5}X_1 \quad if\ x_i\ is\ Female$$

Finally, we can move a few things around to get our final equations:

$$\hat{y_i} \;=\; \hat{\beta_o} \;+\; X_1\left(\hat{\beta_1} \;+\; \hat{\beta_4}X_2\right) \;+\; \hat{\beta_2}X_2 \;+\; \hat{\beta_3} \;+\; \hat{\beta_5}X_1 \quad if\ x_i\ is\ Female$$

$$\hat{y_i} \;=\; \hat{\beta_o} \;+\; X_1\left(\hat{\beta_1} \;+\; \hat{\beta_4}X_2\right) \;+\; \hat{\beta_2}X_2 \quad if\ x_i\ is\ Male$$

Simplified:

$$\hat{y_i} \;=\; \begin{cases} \hat{\beta_o} \;+\; \tilde{\beta_1}X_1 \;+\; \hat{\beta_2}X_2 \;+\; \hat{\beta_3} \;+\; \hat{\beta_5}X_1, & if\ x_i \;=\; Female \\ \hat{\beta_o} \;+\; \tilde{\beta_1}X_1 \;+\; \hat{\beta_2}X_2, & if\ x_i \;=\; Male \end{cases} \qquad where\ \tilde{\beta_1} \;=\; \hat{\beta_1} \;+\; \hat{\beta_4}X_2$$

- **A**.

$$\hat{y_i} \;=\; \begin{cases} 50 \;+\; \tilde{\beta_1}X_1 \;+\; 0.07X_2 \;+\; 35 \;+\; (-10)X_1, & if\ x_i \;=\; Female \\ 50 \;+\; \tilde{\beta_1}X_1 \;+\; 0.07X_2, & if\ x_i \;=\; Male \end{cases} \qquad where\ \tilde{\beta_1} \;=\; 20 \;+\; 0.01X_2$$

Once simplified down to the above two equations, it isn't necessary to put in multiple testing values to see that the correct answer is *iii:* **for a fixed value of IQ and GPA, males earn more on average than females provided that the GPA is high enough.** This is easy to see because the equations have the exact same ouput through the third term in the equation ($0.07X_2$). The caveat *..provided that the GPA is high enough.* is necessary because GPA (X_1) needs to be high enough (> 3.5, to be exact) in order to offset addition of 35 in the female equation. If GPA < 3.5, the prediction for females will be 25, 15 and 5 higher than males for GPA values of 1, 2 and 3 respectively. However, for GPA values that are greater than 3.5, the fifth term in the Female equation becomes greater than the fourth term (35), leading to a decrease in the Female prediction relative to the Male prediction.

- **B**.

$$\hat{y_i} \;=\; 50 \;+\; (4)\,(20 \;+\; 0.01(110)) \;+\; 0.07(110) \;+\; 35 \;+\; (-10)(4) \;=\; 137.1 \;(\; 137,100\; dollars\;)$$

- **C**. False. Without knowing the standard error for the GPA/IQ interaction coefficient, we can't say that there is little evidence for the interaction. The value of the coefficent itself does not lend any information about how confident we are in that value. In order to find this out, we would use the standard error of the coefficient to construct a confidence interval for said coefficient. If that confidence interval contains 0, than there might not be an interaction at all.

4

- **A**. Given the true relationship between X and Y is linear, and two models are fit (model_1 being simple linear regression and model_2 being cubic regression), we would expect model_2 to have a lower (better) RSS on the training data, since there is a negative linear relationship between the flexibility in learning methods used, and the training error rate (the more flexible the model, the more of the irreducible error will be explained away in the training data, masking itself as a better model).
- **B**. Using the test RSS, model_1 (linear) would outperform model_2 (cubic), because, in this case, model_1 represents the true relationship between X and Y perfectly, and the only variance left is that of the irreducible error, which can't be predicted, so the model is "perfect" in the sense that everything that can be modeled is accounted for.
- **C**. Given that the true relationship between X and Y is not linear, and we don't know how far from linear it is, once again, we would expect the cubic regression to have a lower training RSS than the simple linear model. The cubic (more flexible) learning method will be able to "predict" responses that are closer to the training responses because it doesn't make the assumption that the unknown $f(x)_{true}$ is linear.
- **D**. Given that the true relationship between X and Y is not linear, and we don't know how far it is from linear, we don't have enough information to definitively say which model will have a lower test RSS. Whichever model (simple linear regression or cubic regression) is closer to the true relationship between X and Y will have the lower test RSS, however since we don't know which is closer, we can't say (I image this is the main challenge of most statistical models.)

5

$$\hat{y_i} \;=\; x_i\hat{\beta} \quad \textit{and} \quad \hat{\beta} \;=\; \frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i'=1}^{n} x_{i'}^2}$$

$$\hat{y_i} \;=\; x_i \frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i'=1}^{n} x_{i'}^2}$$

$$\hat{y_i} \;=\; \left(\frac{x_i}{1}\right) \frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i'=1}^{n} x_{i'}^2}$$

$$\hat{y_i} \;=\; \sum_{i=1}^{n} \left(\frac{x_i}{1}\right) \left(\frac{\sum_{i=1}^{n} x_{i'} y_{i'}}{\sum_{k=1}^{n} x_k^2}\right)$$

$$\hat{y_i} \;=\; \sum_{i=1}^{n} \left(\frac{x_i}{1}\right) \left(\frac{y_i}{1}\right) \left(\frac{\sum_{i=1}^{n} x_{i'}}{\sum_{k=1}^{n} x_k^2}\right)$$

$$\hat{y_i} \;=\; \sum_{i=1}^{n} \left(\frac{y_i}{1}\right) \left(\frac{\sum_{i=1}^{n} x_i x_{i'}}{\sum_{k=1}^{n} x_k^2}\right)$$

$$\hat{y_i} \;=\; \sum_{i=1}^{n} \left(\frac{\sum_{i=1}^{n} x_i x_{i'}}{\sum_{k=1}^{n} x_k^2}\right) y_{i'}$$

$$\hat{y_i} \;=\; \sum_{i=1}^{n} a_{i'} y_{i'} \quad where \quad a_{i'} \;=\; \left(\frac{\sum_{i=1}^{n} x_i x_{i'}}{\sum_{k=1}^{n} x_k^2}\right)$$

6

We start with the minimizers for $\hat{\beta}_1$ and $\hat{\beta}_0$ and the general structure of *f(x)* under the simple linear regression model:

$$\hat{\beta}_0 \;=\; \bar{y} - \hat{\beta}_1 \bar{x}$$

$$\hat{y_i} \;=\; \hat{\beta}_0 \;+\; \hat{\beta}_1 x_i$$

Re-arranging the equation for $\hat{\beta}_0$ in order to solve for $\bar{x}$ we get:

$$\bar{x} \;=\; \frac{-\hat{\beta}_0 \;+\; \bar{y}}{\hat{\beta}_1}$$

Substituting $\bar{x}$ in for x_i, we can re-write the general structure of *f(x)* as:

$$\hat{y_i} \;=\; \hat{\beta}_0 \;+\; \hat{\beta}_1 \left(\frac{-\hat{\beta}_0 \;+\; \bar{y}}{\hat{\beta}_1}\right)$$

We can then cancel the $\hat{\beta}_1$'s and simply to:

$$\hat{y_i} \;=\; \hat{\beta}_0 \;+\; \left(-\hat{\beta}_0 \;+\; \bar{y}\right)$$

Which then simplifies to show us that:

$$\hat{y_i} \;=\; \bar{y}$$

Where $\bar{y} \;=\; 0 \;=\; y_i$.

7

All summations are from $i = 1 \;\; to \;\; n$

$$Given: \quad R^2 \;=\; \frac{\sum(y_i - y_i)^2 - \sum(y_i - \hat{y_i})^2}{\sum(y_i - y_i)^2} \quad and \quad Cor(X,Y) \;=\; \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2}\sqrt{\sum(y_i - \bar{y})^2}}$$

Looking at the numerator of R^2, we also know that the total sum of squares SS_{tot} is equal to the residual sum of squares SS_{res} plus the regression sum of squares SS_{reg} (an incredible graphic illustrating this can be found on the fourth slide at this link):

$$SS_{tot} \;=\; SS_{res} \;+\; SS_{reg}$$

Which can be re-written as:

$$\sum(y_i - \bar{y})^2 \;=\; \sum(y_i - \hat{y_i})^2 \;+\; \sum(\hat{y_i} - \bar{y})^2$$

Which can the be re-arranged as:

$$\sum(y_i - \bar{y})^2 \;-\; \sum(y_i - \hat{y_i})^2 \;=\; \sum(\hat{y_i} - \bar{y})^2$$

The left side of the above equation is the same as the numerator in R^2, so we can substitute in the right side and begin the proof:

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

We also know the equation for $\hat{y_i}$ as:

$$\hat{y_i} \;=\; \hat{\beta}_0 \;+\; \hat{\beta}_1 x_i$$

Which, substituted in for $\hat{y_i}$ is:

$$R^2 \;=\; \frac{\sum(\hat{\beta}_0 \;+\; \hat{\beta}_1 x_i - \bar{y})^2}{\sum(y_i - y_i)^2}$$

We know the minimizers for $\hat{\beta}_0$ and $\hat{\beta}_1$ to be:

$$\hat{\beta}_0 \;=\; \bar{y} - \hat{\beta}_1 \bar{x} \quad and \quad \hat{\beta}_1 \;=\; \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sum(x_i - \bar{x})^2}$$

Which can be substituted in to get:

$$R^2 \;=\; \frac{\sum(\bar{y} - \hat{\beta}_1 \bar{x} \;+\; \hat{\beta}_1 x_i - \bar{y})^2}{\sum(y_i - y_i)^2}$$

$$R^2 \;=\; \frac{\sum(\hat{\beta}_1 x_i \;-\; \hat{\beta}_1 \bar{x_i})^2}{\sum(y_i - y_i)^2}$$

$$R^2 = \frac{\sum(\hat{\beta_1}^2 x_i^2 - 2\hat{\beta_1}^2 \bar{x} x_i + \hat{\beta_1}^2 \bar{x_i}^2)}{\sum(y_i - y_i)^2}$$

$$R^2 = \frac{\hat{\beta_1}^2 \sum(x_i^2 - 2\bar{x} x_i + \bar{x_i}^2)}{\sum(y_i - y_i)^2}$$

$$R^2 = \frac{\hat{\beta_1}^2 \sum(x_i - \bar{x})^2}{\sum(y_i - y_i)^2}$$

Substitute in $\hat{\beta_1}$ squared to get:

$$R^2 = \frac{\left(\sum(x_i - \bar{x})(y_i - \bar{y})\right)^2 \sum(x_i - \bar{x})^2}{\left(\sum(x_i - \bar{x})^2\right)^2 \sum(y_i - y_i)^2}$$

$$R^2 = \frac{\left(\sum(x_i - \bar{x})(y_i - \bar{y})\right)^2}{\sum(x_i - \bar{x})^2 \sum(y_i - y_i)^2}$$

Which, if we square $Cor(X, Y)$, is now equal to the above equation:

$$Cor(X,Y)^2 = \left(\frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2}\sqrt{\sum(y_i - \bar{y})^2}}\right)^2 = \frac{\left(\sum(x_i - \bar{x})(y_i - \bar{y})\right)^2}{\sum(x_i - \bar{x})^2 \sum(y_i - y_i)^2} = R^2$$

Applied

8

- **A**.

```
library(ISLR)
attach(Auto)
fit <- lm(mpg ~ horsepower)
summary(fit)
```

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

- *A-1.* With a P-value of 2e−16, we can say that there is a statistically significant relationship between horsepower and mpg.
- *A-2.* While there is a relationship betweent the predictor and the response, it isn't overly strong; only 60% of the variance in mpg is explained by horsepower.
- *A-3.* The relationship is negative

```
coef(fit)[2]
```

```
## horsepower
## -0.1578447
```

- *A-4.*

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

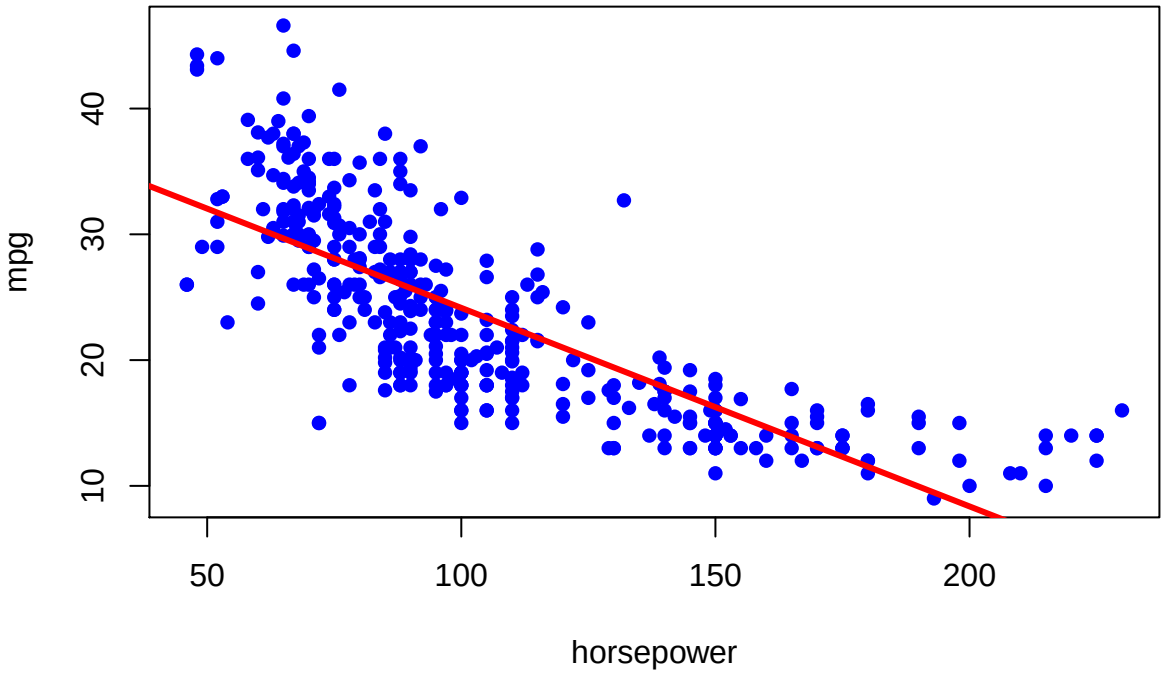
```
##        fit      lwr      upr
## 1 24.46708 23.97308 24.96108
```

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

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

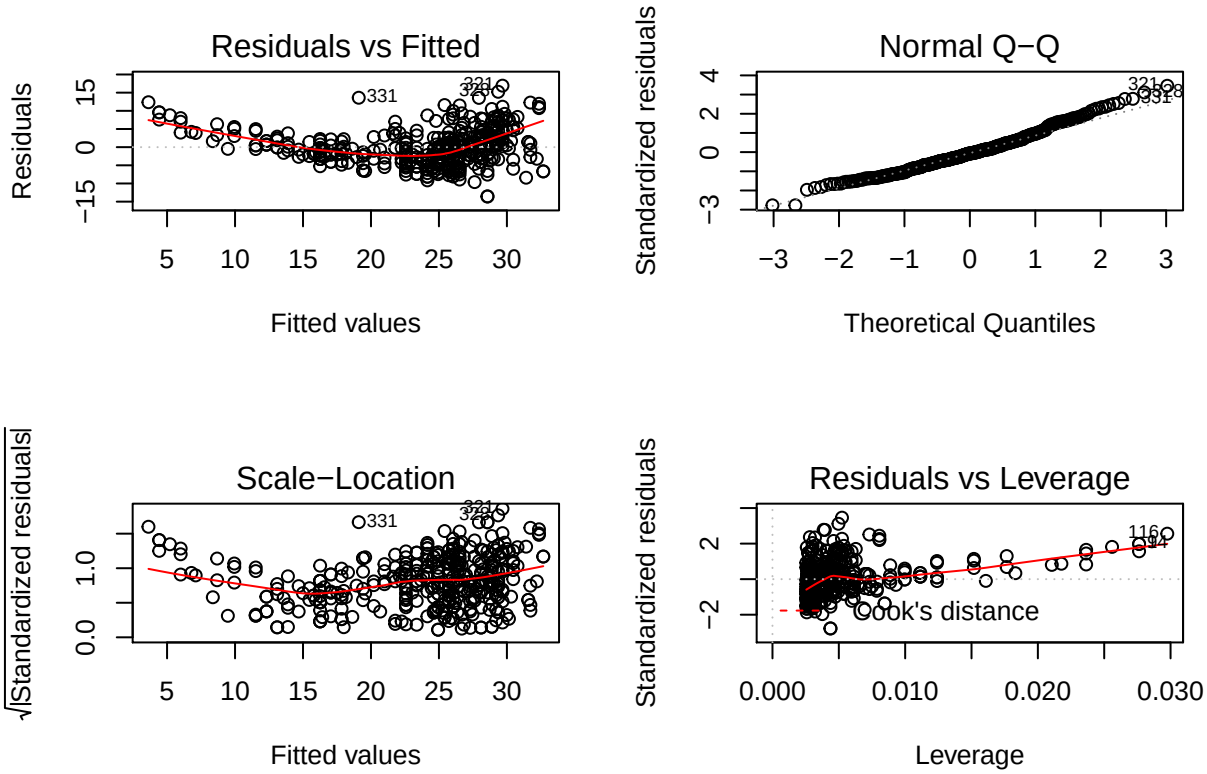
- **B.**

```
plot(horsepower, mpg, pch = 16, col = "blue")
abline(fit, lwd = 3, col = "red")
```



- **C.**

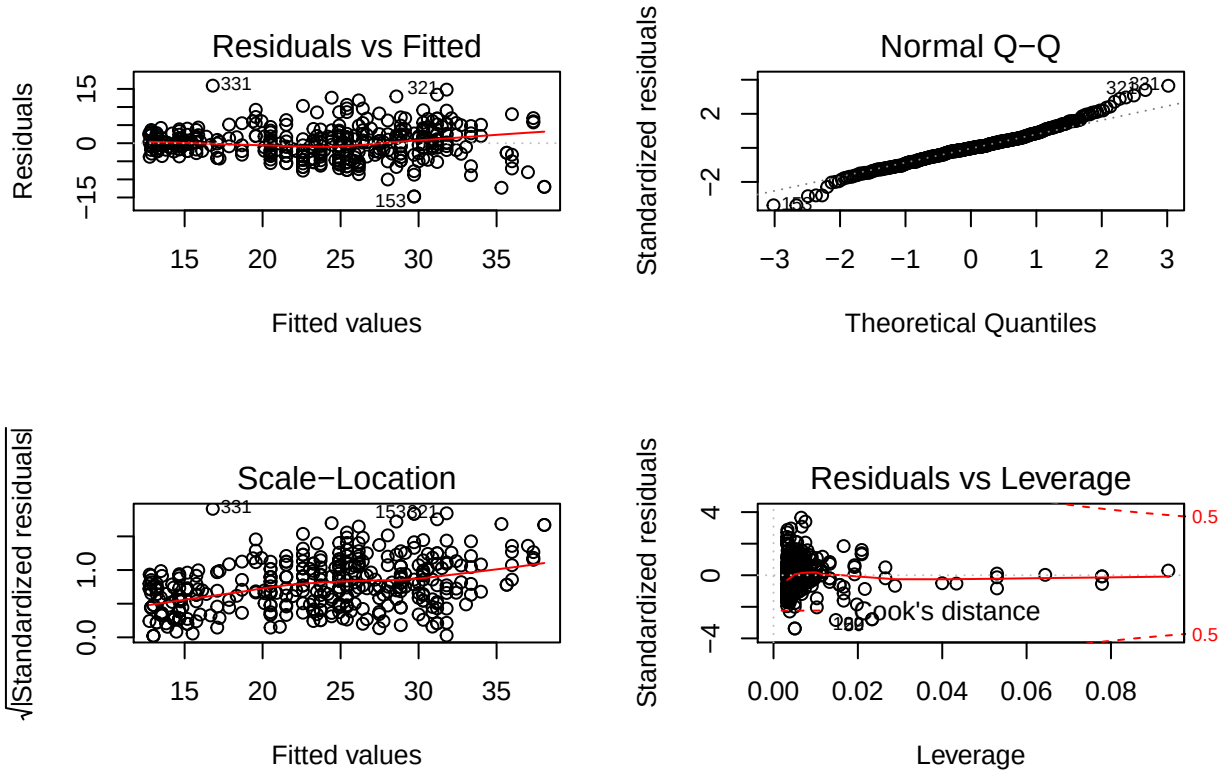
```
par(mfrow = c(2,2))
plot(fit)
```



We can see from the Residuals vs. Leverage plot that no single observation has a huge amount of influence on the regression line; Cooks Distance is not observable within the plot. However, looking at the Residuals vs. Fitted, we can see some evidence of non-linearity; observations with low and high fitted values have positive residuals, while those in the middle of the fitted value scale tend to be negative. This suggests that a model that is quadratic to some degree might be a better model. Looking back at the plot in **B**, we can see that a quadratic fit does seem to fit the data better than a linear model.

Looking at the Residuals vs. Fitted Values plot below (after adding a quadratic term to the model), this does seem to even out the Residuals to some extent.

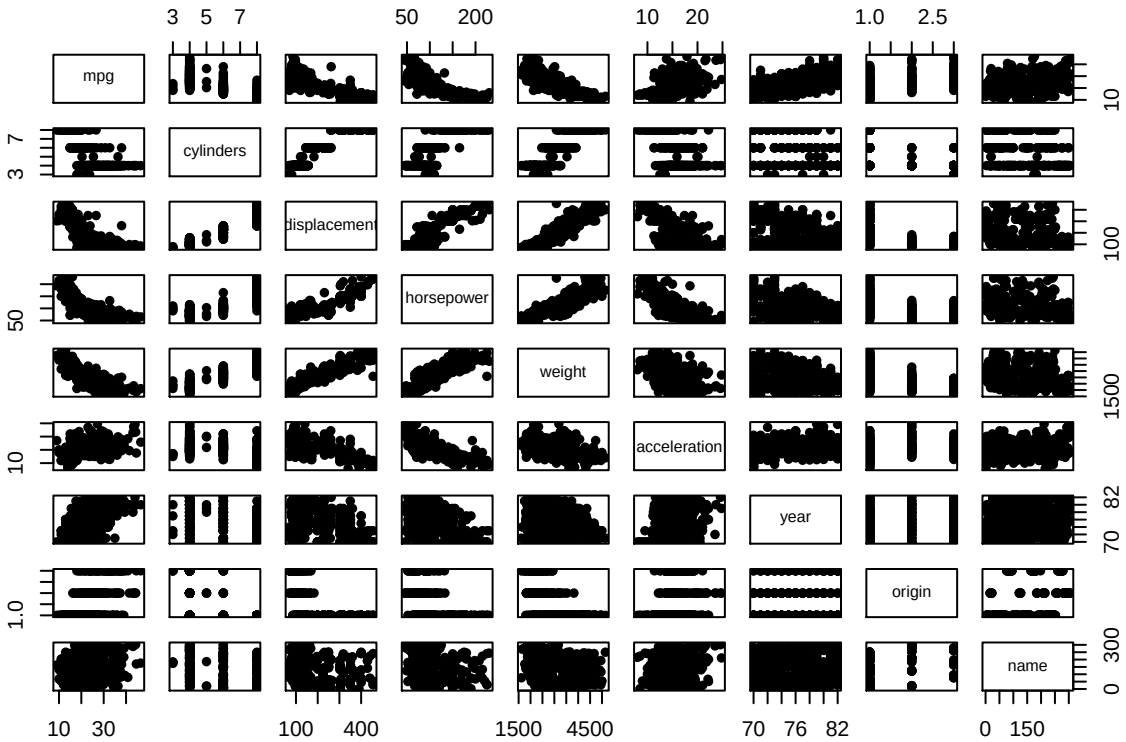
```
par(mfrow = c(2,2))
fit2 <- lm(mpg ~ horsepower + I(horsepower^2))
plot(fit2)
```



9

- **A**.

```
pairs(Auto, pch = 16)
```



- **B.**

```
dat <- subset(Auto, select = -name)
cors <- cor(dat)
cors
```

```
##                     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.**

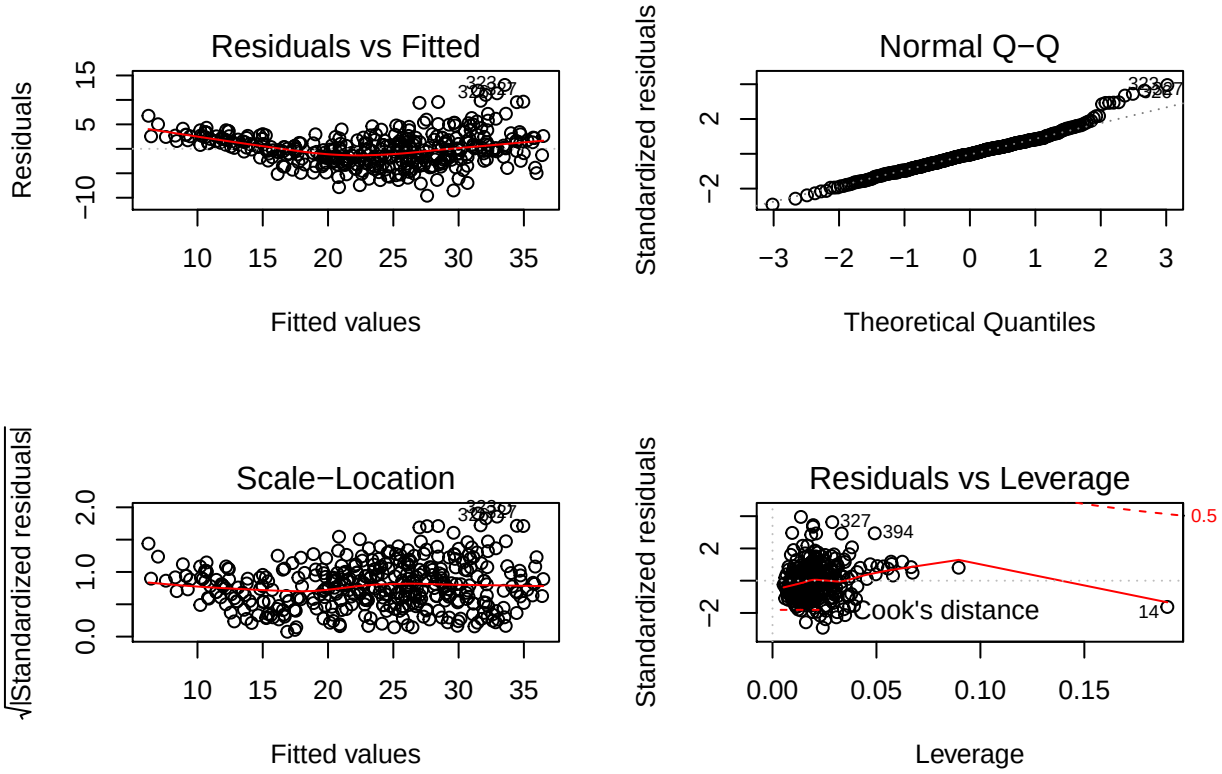
```
fit <- lm(mpg ~ ., data = dat)
summary(fit)
```

```
##
## Call:
## lm(formula = mpg ~ ., data = dat)
##
## 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
```

- *C-1.* Judging from the F-statisic of 252, and a corresponding P-value of effectively 0, we can say that there is a statistically significant relationship between the predictors (as a whole) and the response.

- *C-2.* All variables except acceleration, horsepower and cylinders have statistically significant relationships with the response.
- *C-3.* 0.75, the coefficient for the year variable suggests that, *holding all other (measured) variables constant*, we would see a 0.75 increase in mpg with a one unit increase in year. In other words, we would expect a vehicle produced in 1981 to have a 0.75 increase in mpg over the exact same vehicle, but produced in 1980.
- **D**.

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



Looking at the Residuals vs. Fitted Values plot, it seems that our model is a relatively good fit to the data; the smooth fit to the residuals is slightly concave, indicating there might be a quadratic relationship between one of the variables and the response. In addition, the residuals appear slightly heteroscedastic, which would require further investigation and a possible transformation of one of the variables.

The Residual plots as well as the Normal Q-Q plot suggest that observations 326, 327 and 323 are outliers, having greater values than we might expect given the Normal assumption (the right tail of the residual distribution is heavier than we would expect under the Normal assumption). In addition, observation 14 has the highest leverage on our model, although not a huge amount of influence since the residual isn't overly extreme.

- **E**.

```
fit3 <- lm(mpg ~ displacement*weight, data = dat)
fit4 <- lm(mpg ~ horsepower*displacement, data = dat)
summary(fit4)
```

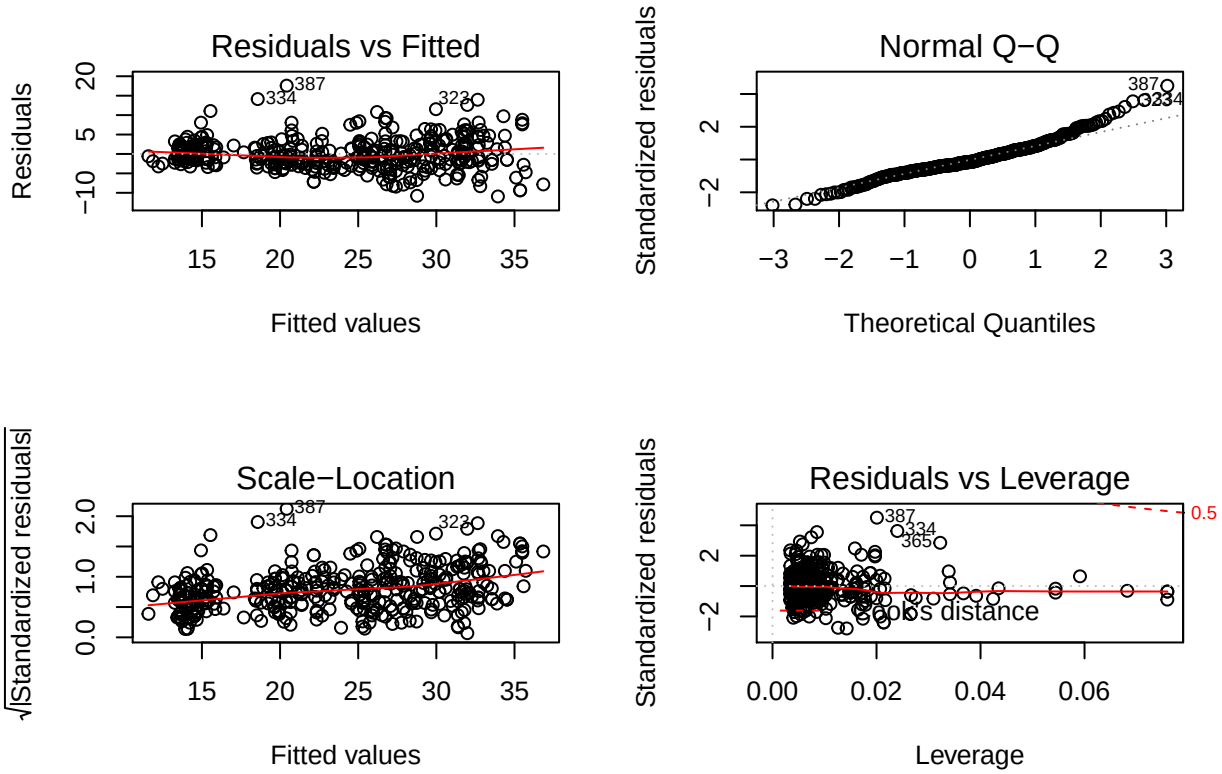
```
##
## Call:
## lm(formula = mpg ~ horsepower * displacement, data = dat)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.9391  -2.3373  -0.5816   2.1698  17.5771
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              5.305e+01  1.526e+00   34.77   <2e-16 ***
## horsepower              -2.343e-01  1.959e-02  -11.96   <2e-16 ***
## displacement            -9.805e-02  6.682e-03  -14.67   <2e-16 ***
## horsepower:displacement  5.828e-04  5.193e-05   11.22   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.944 on 388 degrees of freedom
## Multiple R-squared:  0.7466, Adjusted R-squared:  0.7446
## F-statistic:   381 on 3 and 388 DF,  p-value: < 2.2e-16
```

```
summary(fit3)
```

```
##
## Call:
## lm(formula = mpg ~ displacement * weight, data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.8664  -2.4801  -0.3355   1.8071  17.9429
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)          5.372e+01  1.940e+00  27.697  < 2e-16 ***
## displacement        -7.831e-02  1.131e-02  -6.922 1.85e-11 ***
## weight              -8.931e-03  8.474e-04 -10.539  < 2e-16 ***
## displacement:weight  1.744e-05  2.789e-06   6.253 1.06e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.097 on 388 degrees of freedom
## Multiple R-squared:  0.7265, Adjusted R-squared:  0.7244
## F-statistic: 343.6 on 3 and 388 DF,  p-value: < 2.2e-16
```

```
par(mfrow = c(2,2))
plot(fit4)
```



By building a couple models with interaction terms between some highly correlated variables, we can see that there is statistically significant evidence against the additive assumption of the linear model. The model with the interaction between displacement and horsepower explains more variance in mpg than the other model.

- **F**. Since there seems to be statistical evidence that the additve assumption of the linear model is incorrect, keeping the horsepower and displacement interaction term in our model would be prudent. In addition, there seems to be slight heteroscedasticity in our residuals. This can be seen by both the slight upward trend in the Scale-Location plot, as well as the increasing spread in the Residuals vs. Fitted Values plot.

The final model includes all the variable except cyclinders, in addition to the horsepower and displacement interaction term. I used a log transformation of the response in an attempt to reduce the heteroscedasticity.

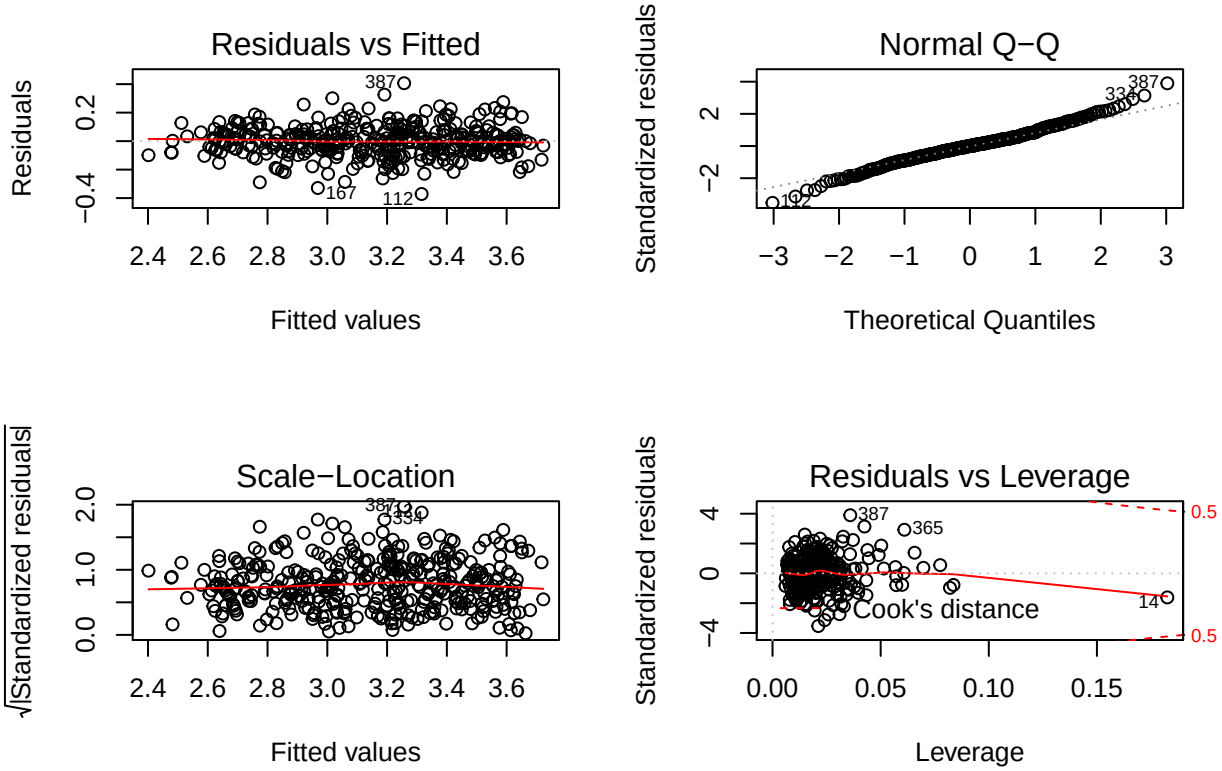
(My understanding of the interpretation of the each $\hat{\beta}_j$ when a log transformation is used on the response variable is that a one unit increase in X results in a $(100 * \hat{\beta}_j)\%$ change in Y)

```
par(mfrow = c(2,2))
fit <- lm(log(mpg + 1) ~ horsepower*displacement + (.-cylinders), data = dat)
summary(fit)
```

```
##
## Call:
## lm(formula = log(mpg + 1) ~ horsepower * displacement + (. -
##     cylinders), data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.37124 -0.06293  0.00044  0.05640  0.40648
##
## Coefficients:
```

```
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 2.245e+00  1.567e-01  14.322  < 2e-16 ***
## horsepower                 -6.052e-03  7.310e-04  -8.279 2.08e-15 ***
## displacement               -1.797e-03  3.018e-04  -5.954 5.89e-09 ***
## weight                     -1.578e-04  2.357e-05  -6.694 7.73e-11 ***
## acceleration               -8.694e-03  3.299e-03  -2.635  0.00874 **
## year                        2.803e-02  1.627e-03  17.222  < 2e-16 ***
## origin                      2.087e-02  9.060e-03   2.304  0.02176 *
## horsepower:displacement     1.356e-05  1.646e-06   8.242 2.72e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1062 on 384 degrees of freedom
## Multiple R-squared:  0.8948, Adjusted R-squared:  0.8929
## F-statistic: 466.6 on 7 and 384 DF,  p-value: < 2.2e-16
```

```
plot(fit)
```



10

- **A**.

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

- **B**.
- *Price* - The coefficient for Price can be interpreted as: A one unit increase in Price can be expected to result in a decrease in Sales of 54 units *on average*.
- *UrbanYes* - The coefficient for UrbanYes can be interpreted as: If the store is located in an Urban area,

there will be an expected decrease in Sales of 22 units *on average*.

- *USYes* - The coefficient for USYes can be interpreted as: If the store is located in the US, there will be an expected increase in Sales of 1,201 units *on average*.

```
summary(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
```

- **C**.

$$\hat{y}_i = 13.04 - 0.054 Price \begin{cases} -0.022, & if\ Urban = Yes\ and\ US = No \\ +1.201, & if\ Urban = No\ and\ US = Yes \\ -0.022 + 1.201, & if\ Urban = Yes\ and\ US = Yes \\ +0, & Otherwise \end{cases}$$

- **D**. Price and US have P-values low enough to suggest statistical significance, and therefore we can reject the null hypothesis for those predictors. However, Urban has a very high P-value, and therefore we can not reject the null hypothesis for that predictor.

- **E**.

```
fit2 <- lm(Sales ~ Price + US, data = Carseats)
```

- **F**. Both linear models fit the data very similarly, however relatively poorly; the R^2 value of 0.24 tells us that both models only explain 24% of the variation in the response. While the RSE is low at first glance, Sales are measured in *thousands* of units. Therefore, the RSE of roughly 2.47 translates to the predicted value of *Sales* being off by an average of 2,470 units *on average*.

```
summary(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
```

```
summary(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
```

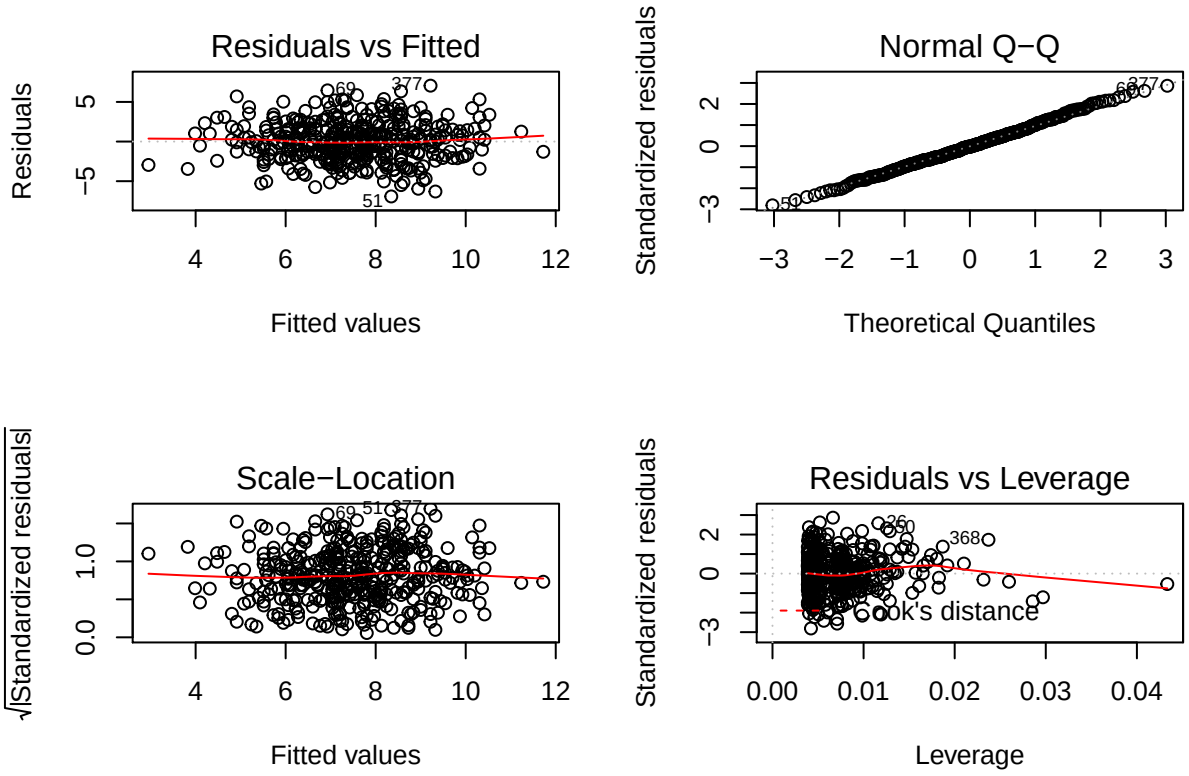
- **G**.

```
confint(fit2)
```

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

- **H**. Observations 377 and 69 could be considered outliers under certain circumstances, however, considering the large scale of the residuals, in my opinion they aren't extreme enough to warrant removal from the data set. In addition, looking at the Residuals vs. Leverage plot, no observations seem to influence the regression line more than others.

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



11

```
set.seed(1)
x <- rnorm(100)
y <- 2*x + rnorm(100)
```

- **A**.

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

The P-value of the coefficient for X suggests statistical significance, and the null hypthosis that $H_0 : \beta = 0$ is rejected.

- **B**.

```
fit2 <- lm(x ~ y + 0)
summary(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
```

Looking at the model of X regressed onto Y, once again the P-value suggests that there is statistical evidence of relationship between Y and X, and the null hypothesis $H_0 : \beta = 0$ is rejected.

- **C**. The results from (a) and (b) show that the equation:

$$\hat{y_i} = \hat{\beta}_1 X_1 + \epsilon \quad = \quad \hat{x_i} = \frac{1}{\hat{\beta}_1}(Y_1 - \epsilon)$$

- **D**. (Note: I'm doing this one step at a time, so the simplification will be long-winded.)

$$Given: \quad \hat{\beta} = \frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i'=1}^{n} x_{i'}^2} \quad and \quad SE(\beta) = \sqrt{\frac{\sum_{i=1}^{n} \left(y_i - x_i\hat{\beta}\right)^2}{(n-1)\sum_{i'=1}^{n} x_{i'}^2}} \quad and \quad t = \frac{\hat{\beta}}{SE\left(\hat{\beta}\right)}$$

$$t = \left(\frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i'=1}^{n} x_{i'}^2}\right) / \left(\frac{\sqrt{\sum_{i=1}^{n} \left(y_i - x_i\hat{\beta}\right)^2}}{\sqrt{(n-1)\sum_{i'=1}^{n} x_{i'}^2}}\right)$$

$$t = \left(\frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i'=1}^{n} x_{i'}^2}\right) \cdot \left(\frac{\sqrt{(n-1)\sum_{i'=1}^{n} x_{i'}^2}}{\sqrt{\sum_{i=1}^{n} \left(y_i - x_i\hat{\beta}\right)^2}}\right)$$

$$t = \frac{\left(\sum_{i=1}^{n} x_i y_i\right)\left(\sqrt{n-1}\right)\left(\sqrt{\sum_{i'=1}^{n} x_{i'}^2}\right)}{\left(\sum_{i'=1}^{n} x_{i'}^2\right)\left(\sqrt{\sum_{i=1}^{n} \left(y_i - x_i\hat{\beta}\right)^2}\right)}$$

$$t = \frac{\left(\sum_{i=1}^{n} x_i y_i\right)\left(\sqrt{n-1}\right)\left(\sqrt{\sum_{i'=1}^{n} x_{i'}^2}\right)}{\left(\sum_{i'=1}^{n} x_{i'}^2\right)\left(\sqrt{\sum_{i=1}^{n} y_i^2 - 2\hat{\beta} x_i y_i + \hat{\beta}^2 x_i^2}\right)}$$

$$t = \frac{\left(\sum_{i=1}^{n} x_i y_i\right)\left(\sqrt{n-1}\right)\left(\sqrt{\sum_{i'=1}^{n} x_{i'}^2}\right)}{\left(\sum_{i'=1}^{n} x_{i'}^2\right)\left(\sqrt{\sum_{i=1}^{n} y_i^2 - 2\hat{\beta} \sum_{i=1}^{n} x_i y_i + \hat{\beta}^2 \sum_{i=1}^{n} x_i^2}\right)}$$

$$t = \frac{\left(\sum_{i=1}^{n} x_i y_i\right)\left(\sqrt{n-1}\right)\left(\sqrt{\sum_{i'=1}^{n} x_{i'}^2}\right)}{\left(\sum_{i'=1}^{n} x_{i'}^2\right)\left(\sqrt{\sum_{i=1}^{n} y_i^2 - 2\frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i'=1}^{n} x_{i'}^2} \sum_{i=1}^{n} x_i y_i + \left(\frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i'=1}^{n} x_{i'}^2}\right)^2 \sum_{i=1}^{n} x_i^2}\right)}$$

$$t = \frac{\left(\sum_{i=1}^{n} x_i y_i\right)\left(\sqrt{n-1}\right)\left(\sqrt{\sum_{i'=1}^{n} x_{i'}^2}\right)}{\left(\sum_{i'=1}^{n} x_{i'}^2\right)\left(\sqrt{\sum_{i=1}^{n} y_i^2 - 2\frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i'=1}^{n} x_{i'}^2} \sum_{i=1}^{n} x_i y_i + \frac{\left(\sum_{i=1}^{n} x_i y_i\right)^2}{\left(\sum_{i'=1}^{n} x_{i'}^2\right)^2} \sum_{i=1}^{n} x_i^2}\right)}$$

$$t = \frac{\left(\sum_{i=1}^{n} x_i y_i\right)\left(\sqrt{n-1}\right)\left(\sqrt{\sum_{i'=1}^{n} x_{i'}^2}\right)}{\left(\sum_{i'=1}^{n} x_{i'}^2\right)\left(\sqrt{\sum_{i=1}^{n} y_i^2 - 2\frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i'=1}^{n} x_{i'}^2} \sum_{i=1}^{n} x_i y_i + \frac{\left(\sum_{i=1}^{n} x_i y_i\right)^2}{\sum_{i'=1}^{n} x_{i'}^2}}\right)}$$

$$t = \frac{\left(\sum_{i=1}^{n} x_i y_i\right)\left(\sqrt{n-1}\right)\left(\sqrt{\sum_{i'=1}^{n} x_{i'}^2}\right)}{\left(\sum_{i'=1}^{n} x_{i'}^2\right)\left(\sqrt{\frac{\sum_{i=1}^{n} y_i^2 \sum_{i=1}^{n} x_i^2}{\sum_{i'=1}^{n} x_{i'}^2} - 2\frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i'=1}^{n} x_{i'}^2} \sum_{i=1}^{n} x_i y_i + \frac{\left(\sum_{i=1}^{n} x_i y_i\right)^2}{\sum_{i'=1}^{n} x_{i'}^2}}\right)}$$

$$t = \frac{\left(\sum_{i=1}^{n} x_i y_i\right)\left(\sqrt{n-1}\right)\left(\sqrt{\sum_{i'=1}^{n} x_{i'}^2}\right)}{\left(\sum_{i'=1}^{n} x_{i'}^2\right)\left(\sqrt{\frac{\sum_{i=1}^{n} y_i^2 \sum_{i=1}^{n} x_i^2 - 2\sum_{i=1}^{n} x_i y_i \sum_{i=1}^{n} x_i y_i + \left(\sum_{i=1}^{n} x_i y_i\right)^2}{\sum_{i'=1}^{n} x_{i'}^2}}\right)}$$

$$t = \frac{\left(\sum_{i=1}^{n} x_i y_i\right)\left(\sqrt{n-1}\right)\left(\sqrt{\sum_{i'=1}^{n} x_{i'}^2}\right)}{\left(\sum_{i'=1}^{n} x_{i'}^2\right)\left(\sqrt{\frac{\sum_{i=1}^{n} y_i^2 \sum_{i=1}^{n} x_i^2 - 2\left(\sum_{i=1}^{n} x_i y_i\right)^2 + \left(\sum_{i=1}^{n} x_i y_i\right)^2}{\sum_{i'=1}^{n} x_{i'}^2}}\right)}$$

$$t = \frac{\left(\sum_{i=1}^{n} x_i y_i\right)\left(\sqrt{n-1}\right)\left(\sqrt{\sum_{i'=1}^{n} x_{i'}^2}\right)}{\left(\sum_{i'=1}^{n} x_{i'}^2\right)\left(\sqrt{\frac{\sum_{i=1}^{n} y_i^2 \sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i y_i\right)^2}{\sum_{i'=1}^{n} x_{i'}^2}}\right)}$$

$$t = \frac{\left(\sum_{i=1}^{n} x_i y_i\right)\left(\sqrt{n-1}\right)\left(\sqrt{\sum_{i'=1}^{n} x_{i'}^2}\right)\left(\sqrt{\sum_{i'=1}^{n} x_{i'}^2}\right)}{\left(\sum_{i'=1}^{n} x_{i'}^2\right)\sqrt{\sum_{i=1}^{n} y_i^2 \sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i y_i\right)^2}}$$

$$t = \frac{\sum_{i=1}^{n} x_i y_i \left(\sqrt{n-1}\right)}{\sqrt{\left(\sum_{i=1}^{n} y_i^2\right)\left(\sum_{i=1}^{n} x_i^2\right) - \left(\sum_{i=1}^{n} x_i y_i\right)^2}}$$

```
new_t_numerator <- sum(x*y) * sqrt(length(x) - 1)
new_t_denominator <- sqrt(sum(y*y) * sum(x*x) - sum(x*y)^2)
new_t <- new_t_numerator/new_t_denominator
round(new_t, digits = 2)
```

```
## [1] 18.73
```

- **E**. Given the above equation of:

$$t = \frac{\sum_{i=1}^{n} x_i y_i \left(\sqrt{n-1}\right)}{\sqrt{\left(\sum_{i=1}^{n} y_i^2\right)\left(\sum_{i=1}^{n} x_i^2\right) - \left(\sum_{i=1}^{n} x_i y_i\right)^2}}$$

it is clear to see that if I switch every x_i with a y_i, and vice versa, the equation becomes:

$$t = \frac{\sum_{i=1}^{n} y_i x_i \left(\sqrt{n-1}\right)}{\sqrt{\left(\sum_{i=1}^{n} x_i^2\right)\left(\sum_{i=1}^{n} y_i^2\right) - \left(\sum_{i=1}^{n} y_i x_i\right)^2}}$$

We can then plug all the variable in using R and get the same result.

```
new_t_numerator <- sum(y*x) * sqrt(length(y) - 1)
new_t_denominator <- sqrt(sum(x*x) * sum(y*y) - sum(y*x)^2)
new_t <- new_t_numerator/new_t_denominator
round(new_t, digits = 2)
```

```
## [1] 18.73
```

- **F**.

```
fit3 <- lm(y ~ x)
fit4 <- lm(x ~ y)
summary(fit3)$coef[2,3]
```

```
## [1] 18.5556
```

```
summary(fit4)$coef[2,3]
```

```
## [1] 18.5556
```

12

- **A**. The coefficient estimate for $\hat{\beta}$ when Y is regressed onto X will be the same as the coefficient estimate for $\hat{\beta}$ when X is regressed onto Y only when the summation of the squares of Y are equal to the summation of the squares of X.
- **B**. The coefficient $\hat{\beta}$ when Y is regressed onto X is 4.85 which does not equal the coefficient $\hat{\beta}$ when X is regressed onto Y.

```
x <- rnorm(100)
y <- 5*x
fit <- lm(y ~ x + 0)
coef(fit)
```

```
## x
## 5
```

```
fit2 <- lm(x ~ y + 0)
coef(fit2)
```

```
##   y
## 0.2
```

- **C**. By building each y_i so that some are the exact same as x_i, and others are the negative values of each x_i, we ensure that $X^2 = Y^2$. This is shown below, also showing that the coefficients are the same.

```
set.seed(10000)
x <- rnorm(100)
y_squared <- x^2
y <- -(sqrt(y_squared))
fit <- lm(y ~ x + 0)

fit2 <- lm(x ~ y + 0)

coef(fit)
```

```
##          x
## -0.1586549
```

```
coef(fit2)
```

```
##          y
## -0.1586549
```

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

```
## [1] 111.086
```

```
sum(x^2)
```

```
## [1] 111.086
```

13

- **A**.

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

- **B**.

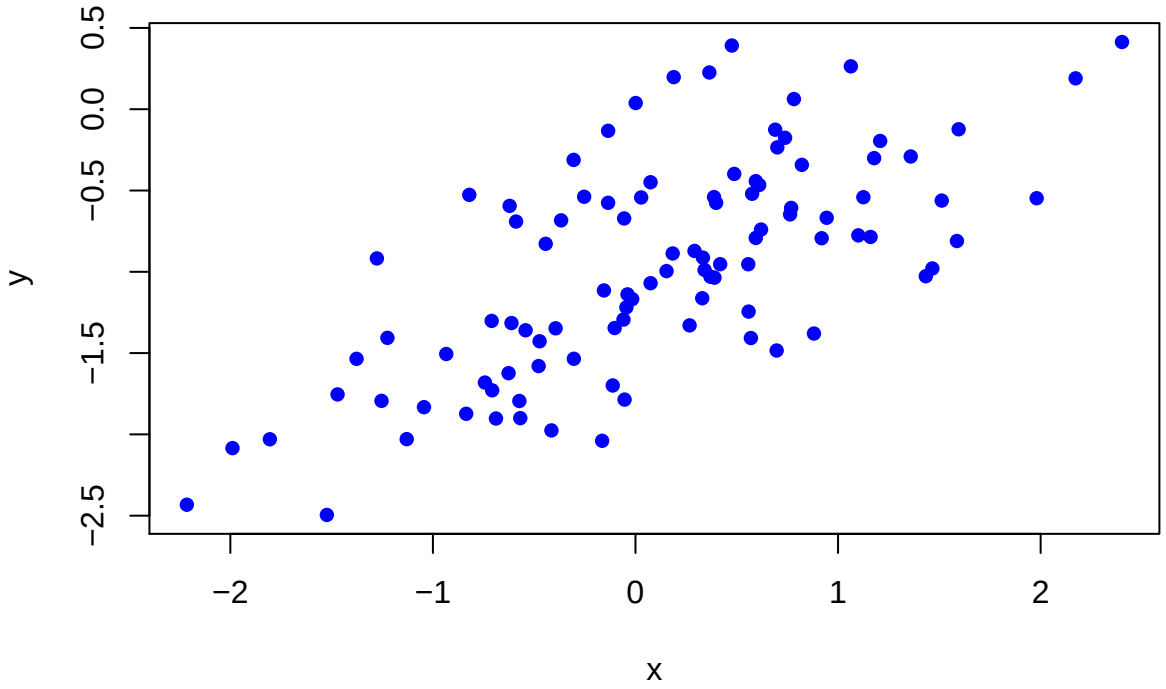
```
eps <- rnorm(100, sd = sqrt(0.25))
```

- **C**. The vector Y will be 100 observations long (the same as X). In the model given, $\hat{\beta}_0 = -1$ and $\hat{\beta}_1 = 0.5$

```
y <- -1 + 0.5*x + eps
```

- **D**. The plot below shows a scatterplot of X and Y. While a linear relationship is known, and somewhat visible, we can see that there is a decent amount of variance in Y for any given value of X.

```
plot(x, y, pch = 16, col = "blue")
```



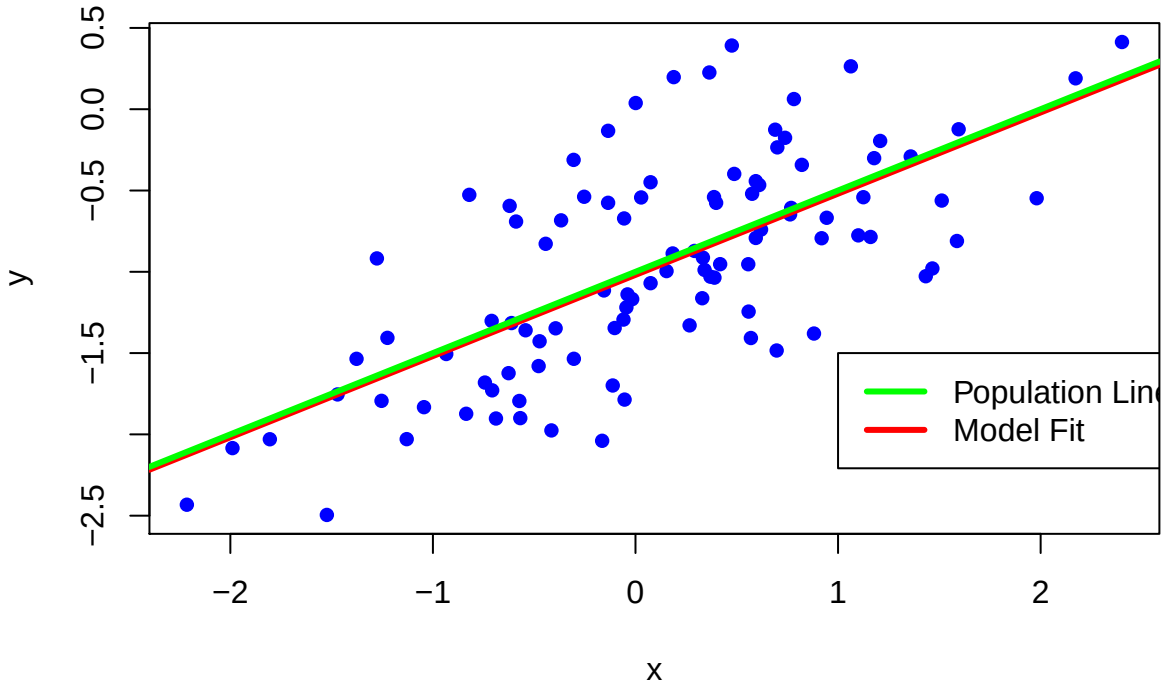
- **E**. The estimates $\hat{\beta}_0$ and $\hat{\beta}_1$ are both very close to their true values; $\hat{\beta}_0$ (if rounded to three decimal places) is only 18 thousandth's off of its true value and $\hat{\beta}_1$ (also rounded to three decimal places) is only 1 thousandth off from its true value.

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

```
##
## 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
```

- **F**.

```
plot(x, y, pch = 16, col = "blue")
abline(fit, lwd = 3, col = "red")
abline(a = -1, b = 0.5, lwd = 3, col = "green")
legend(x = 1, y = -1.5, legend = c("Population Line", "Model Fit"),
       col = c("green", "red"), lwd = 3)
```



- **G**. With a P-value of 0.164, there is no statistical evidence for the quadratic transformation of X with an alpha level of even 0.1.

```
fit2 <- lm(y ~ x + I(x^2))
summary(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
```

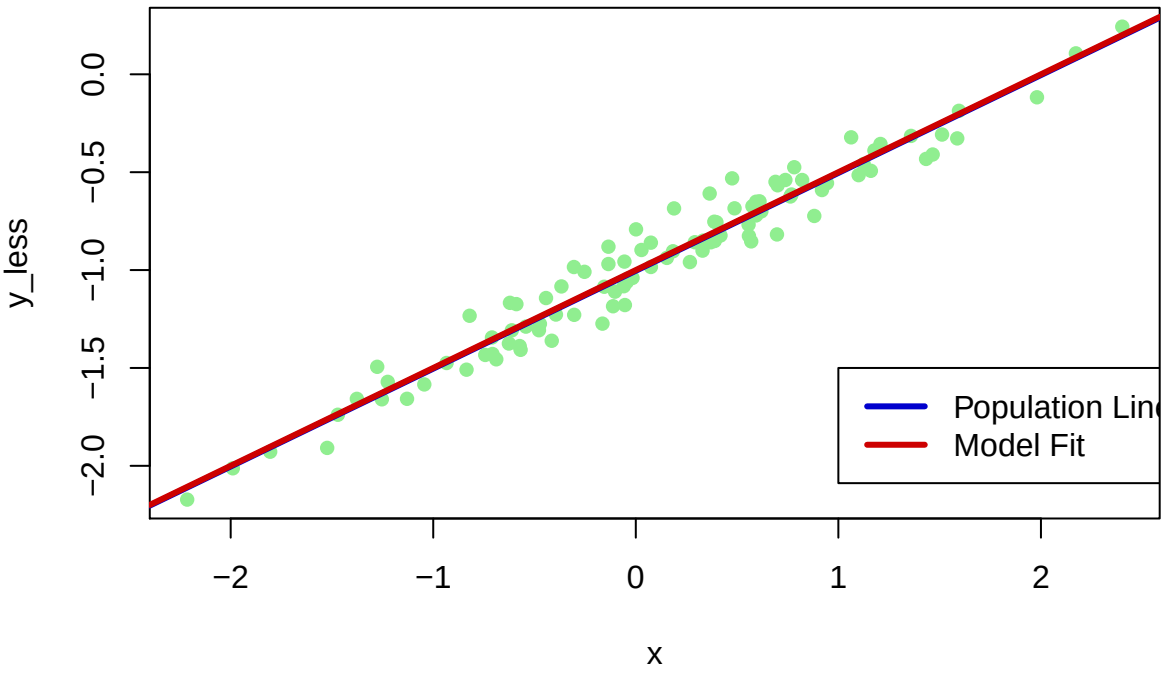
- **H**. Comparing the results from summary(fit_less_noise) to summary(fit), we can see that, as expected, the coefficients become more accurate, their t-statistics more extreme and therefore their P-values more significant. In addition, reducing the error decreases the RSE, increases the R^2, and increases the extremity of the F-statistic.

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

```r
# B
eps_less <- rnorm(100, sd = 0.1)

# C
y_less <- -1 + 0.5*x + eps_less

# D
plot(x, y_less, pch = 16, col = "light green")

# E
fit_less_noise <- lm(y_less ~ x)
summary(fit_less_noise)
```

```
##
## Call:
## lm(formula = y_less ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.18768 -0.06138 -0.01395  0.05394  0.23462
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.003769   0.009699  -103.5   <2e-16 ***
## x            0.499894   0.010773    46.4   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09628 on 98 degrees of freedom
## Multiple R-squared:  0.9565, Adjusted R-squared:  0.956
## F-statistic:  2153 on 1 and 98 DF,  p-value: < 2.2e-16
```

```r
# F
plot(x, y_less, pch = 16, col = "light green")
abline(fit_less_noise, lwd = 3, col = "blue3")
abline(a = -1, b = 0.5, lwd = 3, col = "red3")
legend(x = 1, y = -1.5, legend = c("Population Line", "Model Fit"),
       col = c("blue3", "red3"), lwd = 3)
```



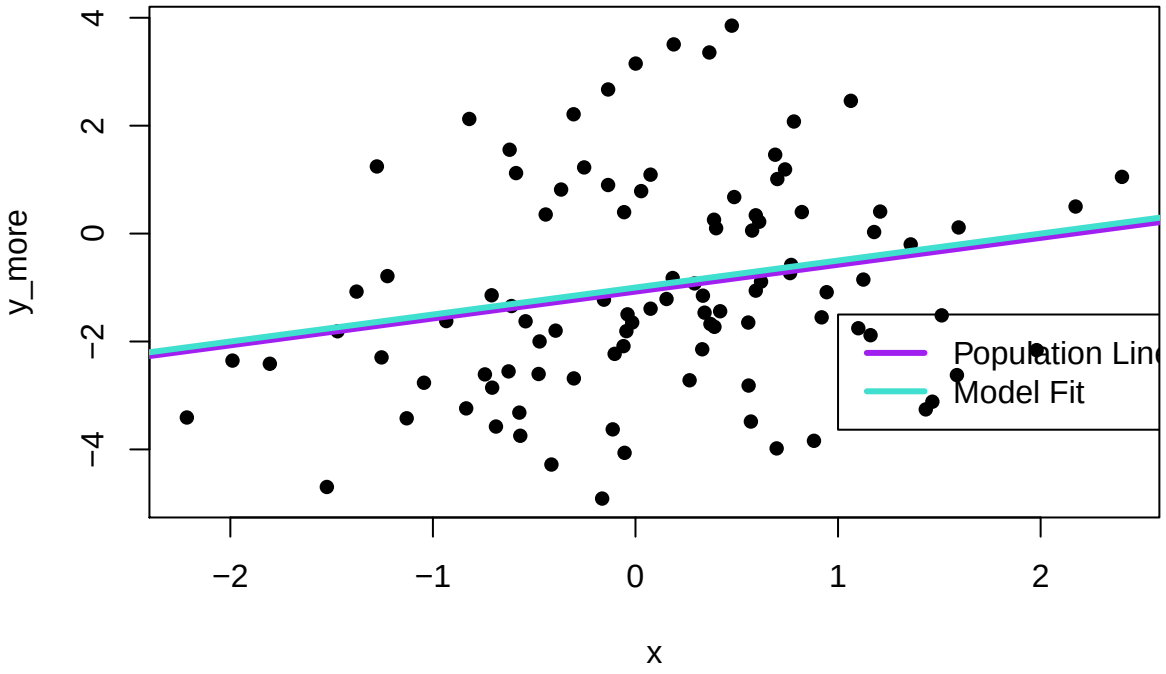
- **I**. As expected, increasing the variance in the response makes the coefficients slightly further away from their true values, therefore increasing their P-values. The RSE and the R^2 increase and decrease, respectively.

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

# B
eps_more <- rnorm(100, sd = 2)

# C
y_more <- -1 + 0.5*x + eps_more

# D
plot(x, y_more, pch = 16, col = "black")

# E
fit_more_noise <- lm(y_more ~ x)
summary(fit_more_noise)
```

```
##
## Call:
## lm(formula = y_more ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.754 -1.228 -0.279  1.079  4.692
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.0754     0.1940  -5.544  2.5e-07 ***
## x             0.4979     0.2155   2.311   0.0229 *
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.926 on 98 degrees of freedom
## Multiple R-squared:  0.05167,    Adjusted R-squared:  0.042
## F-statistic:  5.34 on 1 and 98 DF,  p-value: 0.02294
```

```
# F
plot(x, y_more, pch = 16, col = "black")
abline(fit_more_noise, lwd = 3, col = "purple")
abline(a = -1, b = 0.5, lwd = 3, col = "turquoise")
legend(x = 1, y = -1.5, legend = c("Population Line", "Model Fit"),
       col = c("purple", "turquoise"), lwd = 3)
```



- **J**. As expected, the 95% confidence interval becomes increasingly wider as the variance in the error term increases.

```
confint(fit_less_noise)
```

```
##                 2.5 %     97.5 %
## (Intercept) -1.0230161 -0.9845224
## x            0.4785159  0.5212720
```

```
confint(fit)
```

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

```
confint(fit_more_noise)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.46032149 -0.6904490
## x            0.07031765  0.9254408
```

14

- **A**.

$$\beta_0 = 2,\ \beta_1 = 2 \quad and \quad \beta_2 = 0.3$$

$$Y = 2 + 2X_1 + 0.3X_2 + \epsilon$$

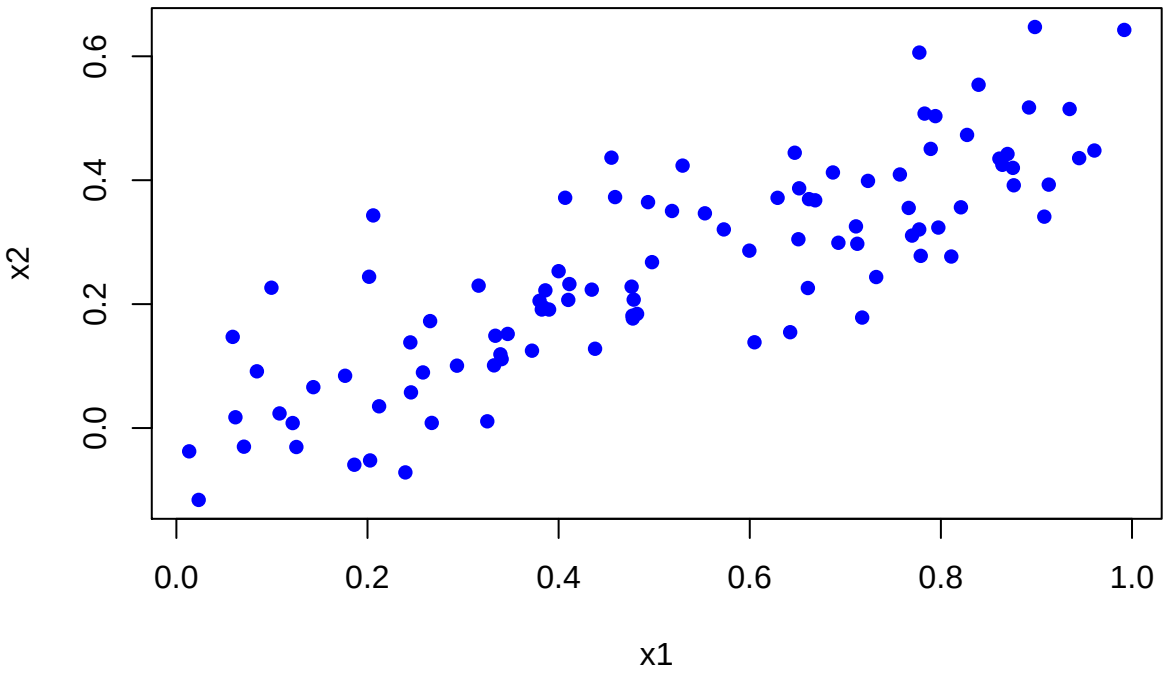
```
set.seed(1)
x1 <- runif(100)
x2 <- 0.5*x1 + rnorm(100)/10
y <- 2 + 2*x1 + 0.3*x2 + rnorm(100)
```

- **B**.

```
cor(x1,x2)
```

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

```
plot(x1, x2, pch = 16, col = "blue")
```



- **C**. Looking at the summary for the model fit to the data with the correlated terms, we can see that $\hat{\beta}_0$ is close to its true value (2) with a highly significant P-value. However, both $\hat{\beta}_1$ and $\hat{\beta}_2$ have increasingly less significant P-values, as expected in the presence of collinearity. While we can still reject that null hypothesis that $\hat{\beta}_1 = 0$, given an alpha level of 0.95, if we did not know that the true value for $\hat{\beta}_2 = 2$, and we were making decisions about our model based on the summary below, we would not be able reject the null hypothesis that $H_0 : \hat{\beta}_2 = 0$.

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

```
##
## 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**. In the absence of collinear terms, both the estimates for $\hat{\beta}_0$ and $\hat{\beta}_1$ are far closer to their true values. In addition, their P-values are far more significant, and the null hypothesis that $\hat{\beta}_1 = 0$ can be rejected.

```
fit1 <- lm(y ~ x1)
summary(fit1)
```

```
##
## 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**. We can still reject the null hypothesis that $\hat{\beta}_1 = 0$.

```
fit2 <- lm(y ~ x2)
summary(fit2)
```

```
##
## 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
```

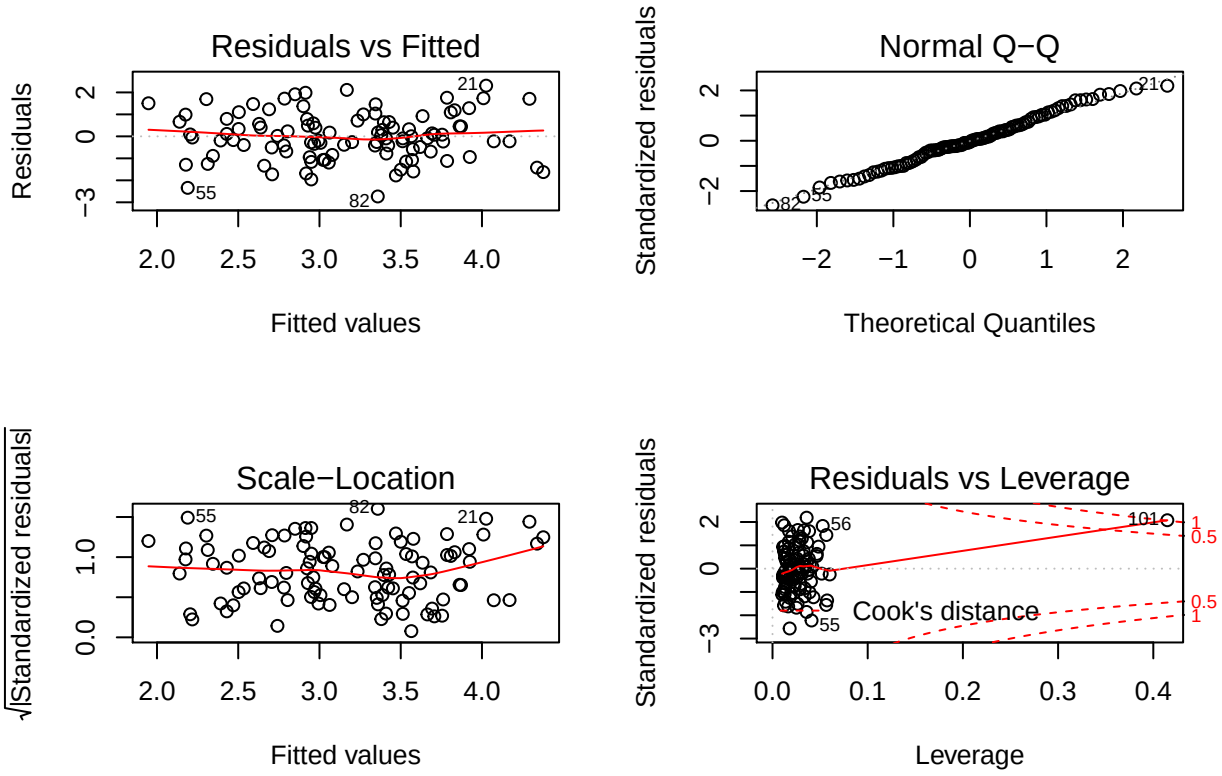
- **F**. While the results do contradict each other (in the model with both terms, we would conclude that $\hat{\beta}_2$ does not have enough evidence to be included in the model, in the model with X_2 as the predictor, we **would** include it in the model), it is to be expected. When two collinear predictors are included in the model, it can be hard to tell *which* predictor is responible for the variance in the response, and therefore we (and our software), have less accurate estimates for the coefficients.
- **G**.

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

After adding the additional data point to the data set and re-fitting the models, the following changes occured in each of the models:

- *fit1 (y ~ x1 + x2)* - The additional data point, being an outlier and having a lot of leverage on the regression line, had a large affect on the model with both X_1 and X_2 as predictors, as seen in the Residuals vs. Leverage plot below (point 101). In addition, we would now reject the null hypothesis that $\hat{\beta}_2 = 0$ and not be able to reject the null hypothesis that $\hat{\beta}_1 = 0$ (opposite of the model without the additional data point).

```
fit1 <- lm(y ~ x1 + x2)
par(mfrow = c(2,2))
plot(fit1)
```

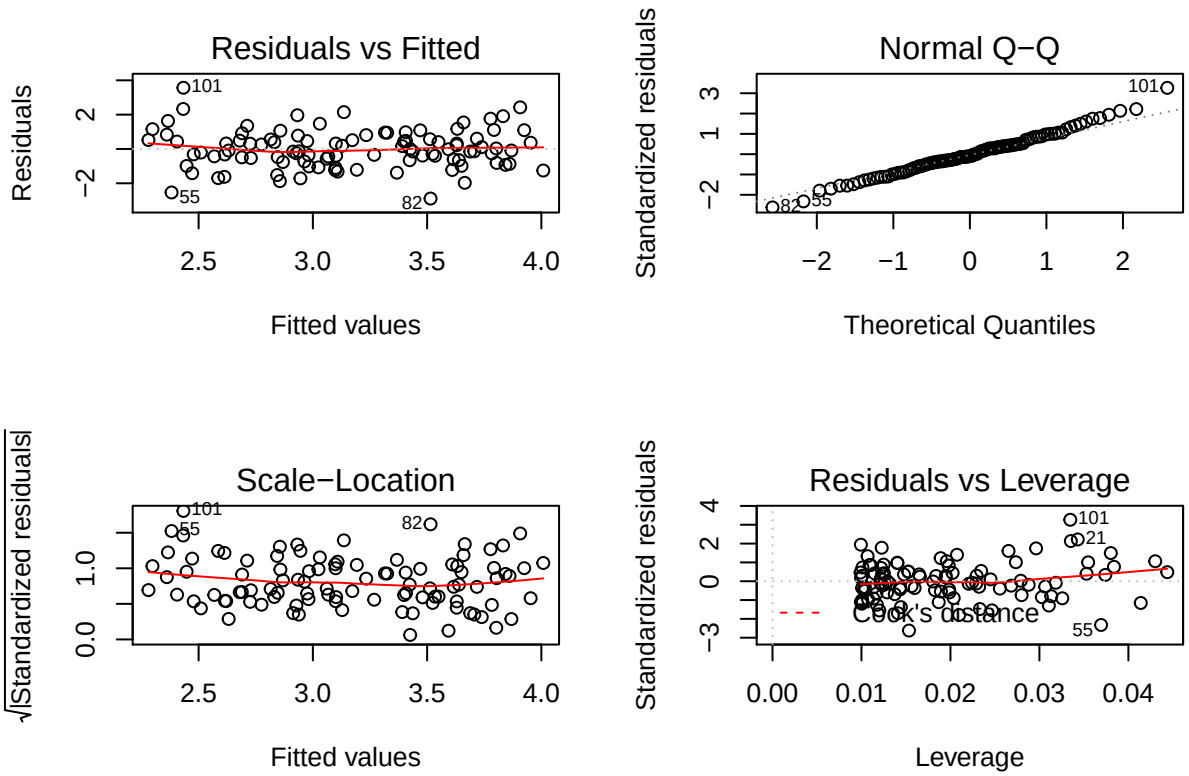



```
summary(fit1)
```

```
##
## 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
```

- *fit2 ($y \sim x1$)* - In the model that only includes X_1 as a predictor, the new data point has significantly less influence than the model with both predictors. Again, looking at the Residual vs. Leverage plot, we can see that the smooth fit to the residuals starts to tilt slightly upward, however not to the extent that it did in the previous model. While the data point is an outlier in this model, it wouldn't quite be considered a high leverage point since there are a few other data points having higher leverage statistics than it does.

```r
fit2 <- lm(y ~ x1)
par(mfrow = c(2,2))
plot(fit2)
```

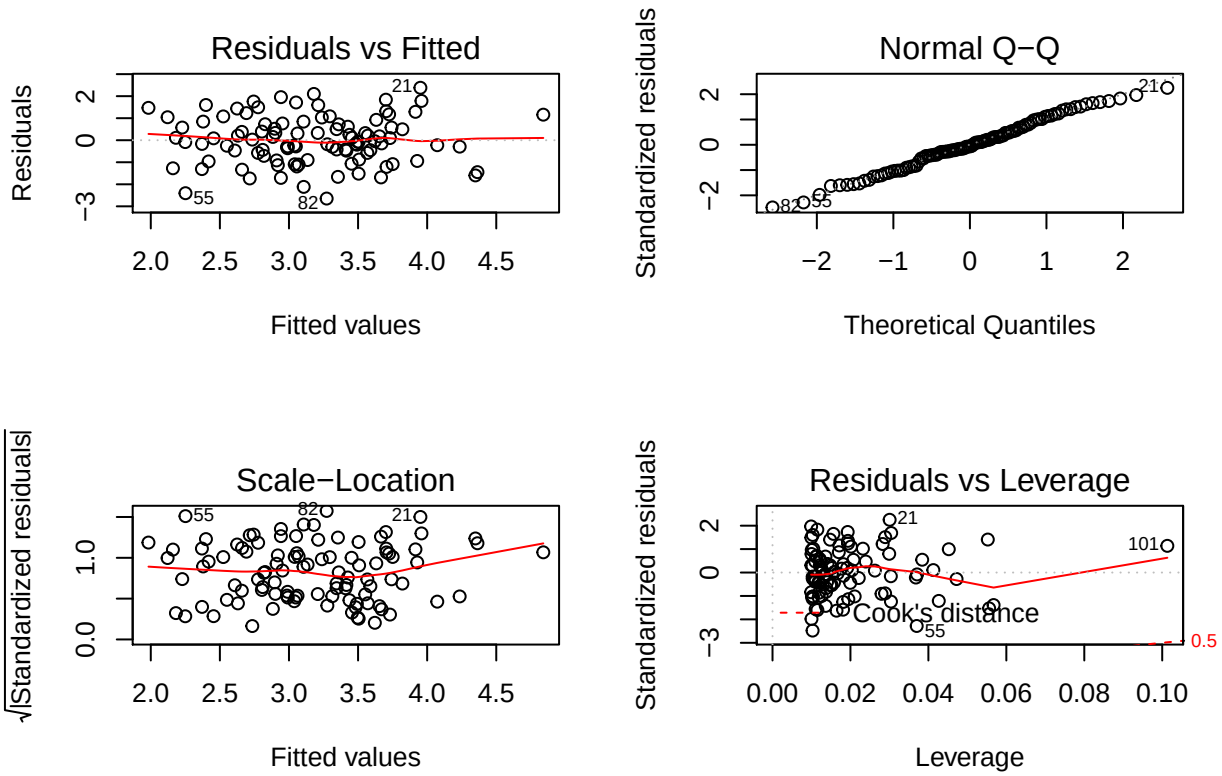



```r
summary(fit2)
```

```
##
## 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
```

- *fit3 ($y \sim x2$)* - We can see that the new data point has a lot of leverage on the regression line with only X_2 as the predictor. However, since the residual isn't overly extreme, it doesn't influence the line that much.

```r
fit3 <- lm(y ~ x2)
par(mfrow = c(2,2))
plot(fit3)
```




```r
summary(fit3)
```

```
##
## 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
```

15

- **A**. When each predictor is used in simple linear regression, "chas" or "Charles River Dummy Variable" is the only predictor that does not have statistically significant evidence to reject the null hypothesis that $H_0 : \hat{\beta}_1 = 0$. (I'm choosing to omit creating plots for each individual model in order to avoid the report being too long)

```
library(MASS)
attach(Boston)

for (i in 1:ncol(Boston)) {
    print(names(Boston)[i])
    print(summary(lm(crim ~ Boston[[i]])))
}
```

```
## [1] "crim"
```

```
## Warning in summary.lm(lm(crim ~ Boston[[i]])): essentially perfect fit:
## summary may be unreliable
```

```
##
## Call:
## lm(formula = crim ~ Boston[[i]])
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -1.768e-13  1.310e-16  3.800e-16  4.690e-16  8.488e-15
##
## Coefficients:
##               Estimate Std. Error    t value Pr(>|t|)
## (Intercept) -2.527e-15  3.818e-16 -6.618e+00 9.33e-11 ***
## Boston[[i]]  1.000e+00  4.096e-17  2.441e+16  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.918e-15 on 504 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 5.96e+32 on 1 and 504 DF,  p-value: < 2.2e-16
##
## [1] "zn"
##
## Call:
## lm(formula = crim ~ Boston[[i]])
##
## 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 ***
## Boston[[i]] -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
##
## [1] "indus"
##
## Call:
## lm(formula = crim ~ Boston[[i]])
##
## 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 **
## Boston[[i]]  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
##
## [1] "chas"
##
## Call:
## lm(formula = crim ~ Boston[[i]])
##
## 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 ***
## Boston[[i]]  -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
##
## [1] "nox"
##
## Call:
## lm(formula = crim ~ Boston[[i]])
##
## 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 ***
## Boston[[i]]   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
##
## [1] "rm"
##
## Call:
## lm(formula = crim ~ Boston[[i]])
##
## 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 ***
## Boston[[i]]   -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
##
## [1] "age"
##
## Call:
## lm(formula = crim ~ Boston[[i]])
##
## 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 ***
## Boston[[i]]  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
##
## [1] "dis"
##
## Call:
## lm(formula = crim ~ Boston[[i]])
##
```

```
## 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 ***
## Boston[[i]]  -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
##
## [1] "rad"
##
## Call:
## lm(formula = crim ~ Boston[[i]])
##
## 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 ***
## Boston[[i]]  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
##
## [1] "tax"
##
## Call:
## lm(formula = crim ~ Boston[[i]])
##
## 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 ***
## Boston[[i]]  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
```

```
##
## [1] "ptratio"
##
## Call:
## lm(formula = crim ~ Boston[[i]])
##
## 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 ***
## Boston[[i]]   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
##
## [1] "black"
##
## Call:
## lm(formula = crim ~ Boston[[i]])
##
## 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 ***
## Boston[[i]] -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
##
## [1] "lstat"
##
## Call:
## lm(formula = crim ~ Boston[[i]])
##
## 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 ***
## Boston[[i]]  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
##
## [1] "medv"
##
## Call:
## lm(formula = crim ~ Boston[[i]])
##
## 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 ***
## Boston[[i]] -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**. When a multiple linear regression model is fit using all of the variable to predict "crim," the only variables for which we can reject the null hypothesis that $H_0 : \hat{\beta}_j = 0$ are zn, nox, dis, rad, black, lstat and medv. Since this contradicts the results when each predictor is used separately, I would bet that some of the variables are collinear.

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

```
##
## 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
```

- **C**. As you can see from the plot below, almost all of the variables have relatively similar values for their coefficient pairs; the x-axis value corresponds to the predictor's coefficient when used in simple linear regression and the y-axis value corresponds to its coefficient when all the variables are used to predict crim, the response. The only variable that has a drastic difference in the values for its coefficient is "nox," which has a negative value (-10.3) when fit with all other variables and a positive value (31) when fit alone.

```
library(ggplot2)
```

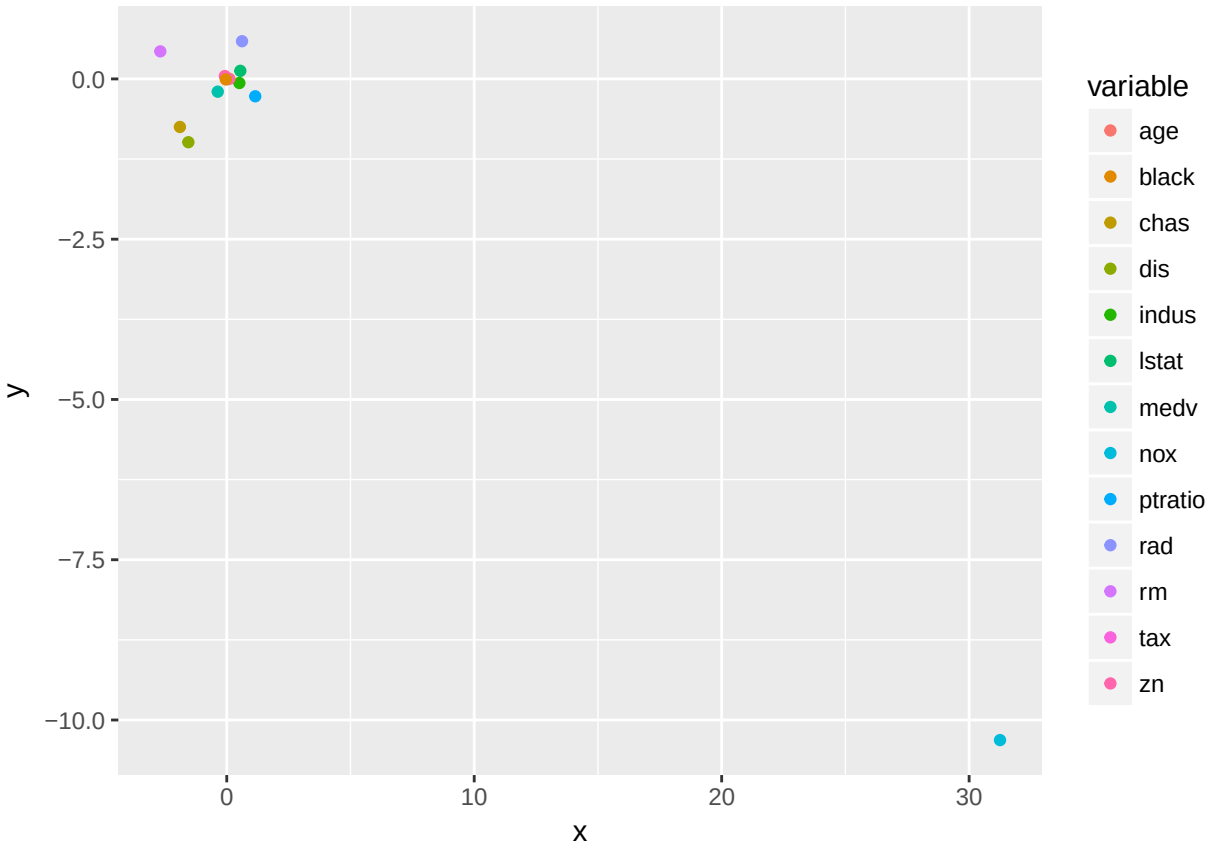
```
##
## Attaching package: 'ggplot2'
```

```
## The following object is masked from 'Auto':
##
##     mpg
```

```
x <- NULL
for (i in 1:ncol(Boston)) {
    fit <- lm(crim ~ Boston[[i]])
    summary(fit)
    x1 <- coef(fit)[[2]]
    x <- c(x, x1)
}
```

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

```
coef_df <- data.frame(variable = names(multi_fit$coefficients),x = x,
                      y = multi_fit$coefficients)
coef_df <- coef_df[2:nrow(coef_df),]

qplot(x,y, data = coef_df, col = variable)
```



- **D**. There seems to be statistically significant evidence that indus, nox, age, dis, ptratio and medv all have a non-linear relationship with the response, crim, based on the P-values for their respective quadratic and cubic terms.

```
for (i in 1:ncol(Boston)) {
    print(names(Boston)[i])
    print(summary(lm(crim ~ Boston[[i]] + I(Boston[[i]]^2) + I(Boston[[i]]^3))))
}
```

```
## [1] "crim"
```

```
## Warning in summary.lm(lm(crim ~ Boston[[i]] + I(Boston[[i]]^2) +
## I(Boston[[i]]^3))): essentially perfect fit: summary may be unreliable
```

```
##
## Call:
## lm(formula = crim ~ Boston[[i]] + I(Boston[[i]]^2) + I(Boston[[i]]^3))
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -1.766e-13  1.890e-16  3.520e-16  5.940e-16  8.688e-15
##
## Coefficients:
##                     Estimate Std. Error    t value Pr(>|t|)
## (Intercept)       -2.527e-15  4.224e-16 -5.983e+00 4.17e-09 ***
## Boston[[i]]        1.000e+00  1.401e-16  7.138e+15  < 2e-16 ***
## I(Boston[[i]]^2) -9.330e-19  6.603e-18 -1.410e-01    0.888
## I(Boston[[i]]^3) -1.271e-20  6.441e-20 -1.970e-01    0.844
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.913e-15 on 502 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 1.989e+32 on 3 and 502 DF,  p-value: < 2.2e-16
##
## [1] "zn"
##
## Call:
## lm(formula = crim ~ Boston[[i]] + I(Boston[[i]]^2) + I(Boston[[i]]^3))
##
## 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 ***
## Boston[[i]]      -3.322e-01  1.098e-01  -3.025  0.00261 **
## I(Boston[[i]]^2)  6.483e-03  3.861e-03   1.679  0.09375 .
## I(Boston[[i]]^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
##
## [1] "indus"
##
## Call:
## lm(formula = crim ~ Boston[[i]] + I(Boston[[i]]^2) + I(Boston[[i]]^3))
##
## 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 *
## Boston[[i]]      -1.9652129  0.4819901  -4.077 5.30e-05 ***
## I(Boston[[i]]^2)  0.2519373  0.0393221   6.407 3.42e-10 ***
## I(Boston[[i]]^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
##
## [1] "chas"
##
## Call:
## lm(formula = crim ~ Boston[[i]] + I(Boston[[i]]^2) + I(Boston[[i]]^3))
```

```
##
## 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 ***
## Boston[[i]]       -1.8928     1.5061  -1.257    0.209
## I(Boston[[i]]^2)       NA         NA      NA       NA
## I(Boston[[i]]^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
##
## [1] "nox"
##
## Call:
## lm(formula = crim ~ Boston[[i]] + I(Boston[[i]]^2) + I(Boston[[i]]^3))
##
## 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 ***
## Boston[[i]]      -1279.37     170.40  -7.508 2.76e-13 ***
## I(Boston[[i]]^2)  2248.54     279.90   8.033 6.81e-15 ***
## I(Boston[[i]]^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
##
## [1] "rm"
##
## Call:
## lm(formula = crim ~ Boston[[i]] + I(Boston[[i]]^2) + I(Boston[[i]]^3))
##
## 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 .
## Boston[[i]]      -39.1501    31.3115  -1.250   0.2118
## I(Boston[[i]]^2)   4.5509     5.0099   0.908   0.3641
```

```
## I(Boston[[i]]^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
##
## [1] "age"
##
## Call:
## lm(formula = crim ~ Boston[[i]] + I(Boston[[i]]^2) + I(Boston[[i]]^3))
##
## 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
## Boston[[i]]       2.737e-01  1.864e-01   1.468  0.14266
## I(Boston[[i]]^2) -7.230e-03  3.637e-03  -1.988  0.04738 *
## I(Boston[[i]]^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
##
## [1] "dis"
##
## Call:
## lm(formula = crim ~ Boston[[i]] + I(Boston[[i]]^2) + I(Boston[[i]]^3))
##
## 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 ***
## Boston[[i]]      -15.5543     1.7360  -8.960  < 2e-16 ***
## I(Boston[[i]]^2)   2.4521     0.3464   7.078 4.94e-12 ***
## I(Boston[[i]]^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
##
## [1] "rad"
##
```

```
## Call:
## lm(formula = crim ~ Boston[[i]] + I(Boston[[i]]^2) + I(Boston[[i]]^3))
##
## 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
## Boston[[i]]       0.512736   1.043597   0.491    0.623
## I(Boston[[i]]^2) -0.075177   0.148543  -0.506    0.613
## I(Boston[[i]]^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
##
## [1] "tax"
##
## Call:
## lm(formula = crim ~ Boston[[i]] + I(Boston[[i]]^2) + I(Boston[[i]]^3))
##
## 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
## Boston[[i]]      -1.533e-01  9.568e-02  -1.602    0.110
## I(Boston[[i]]^2)  3.608e-04  2.425e-04   1.488    0.137
## I(Boston[[i]]^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
##
## [1] "ptratio"
##
## Call:
## lm(formula = crim ~ Boston[[i]] + I(Boston[[i]]^2) + I(Boston[[i]]^3))
##
## 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 **
## Boston[[i]]      -82.36054   27.64394  -2.979  0.00303 **
## I(Boston[[i]]^2)   4.63535    1.60832   2.882  0.00412 **
## I(Boston[[i]]^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
##
## [1] "black"
##
## Call:
## lm(formula = crim ~ Boston[[i]] + I(Boston[[i]]^2) + I(Boston[[i]]^3))
##
## 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)       1.826e+01  2.305e+00   7.924  1.5e-14 ***
## Boston[[i]]      -8.356e-02  5.633e-02  -1.483    0.139
## I(Boston[[i]]^2)  2.137e-04  2.984e-04   0.716    0.474
## I(Boston[[i]]^3) -2.652e-07  4.364e-07  -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
##
## [1] "lstat"
##
## Call:
## lm(formula = crim ~ Boston[[i]] + I(Boston[[i]]^2) + I(Boston[[i]]^3))
##
## 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
## Boston[[i]]      -0.4490656  0.4648911  -0.966   0.3345
## I(Boston[[i]]^2)  0.0557794  0.0301156   1.852   0.0646 .
## I(Boston[[i]]^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
##
## [1] "medv"
##
## Call:
## lm(formula = crim ~ Boston[[i]] + I(Boston[[i]]^2) + I(Boston[[i]]^3))
```

```
##
## 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 ***
## Boston[[i]]      -5.0948305  0.4338321 -11.744  < 2e-16 ***
## I(Boston[[i]]^2)  0.1554965  0.0171904   9.046  < 2e-16 ***
## I(Boston[[i]]^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
```