Brauchle_HW2

3a. Provided the GPA is high enough, males earn more on average than women.
3b. (Salary) = 50 + 20 * 4.0 + 0.07 * 110 + 35 + 0.01 * (4.0*110) - 10* (4.0) = 137.1
3c. False, because we would need information on the standard error of the interaction to find a probability of the hypothesis β_i = 0.

8.

```
#8a i - iii
auto_slm <- lm(mpg ~ horsepower, data = Auto)
summary(auto_slm)
```

```
##
## 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
##   (5 observations deleted due to missingness)
## Multiple R-squared:  0.6059, Adjusted R-squared:  0.6049
## F-statistic: 599.7 on 1 and 390 DF,  p-value: < 2.2e-16
```

```
#8aiv
predict (auto_slm ,data.frame(horsepower = 98), interval="confidence")
```

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

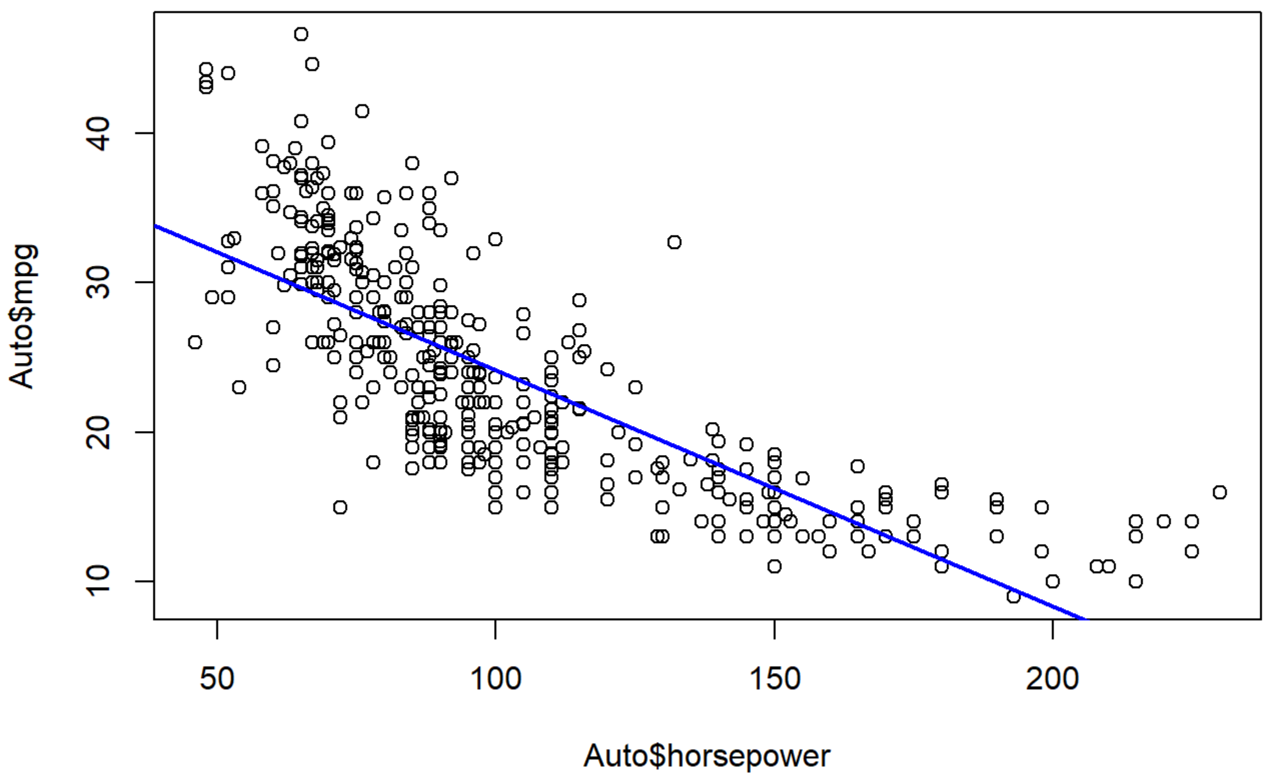
```
predict (auto_slm ,data.frame(horsepower = 98), interval="prediction")
```

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

8a. i. There is a linear relationship between horsepower and mpg ($p < 0.05$).
ii. The relationship is not particularly strong with a R-squared of 0.1767.
iii.The relationship between horsepower and mpg is negative. With every one unit increase in horsepower, mpg

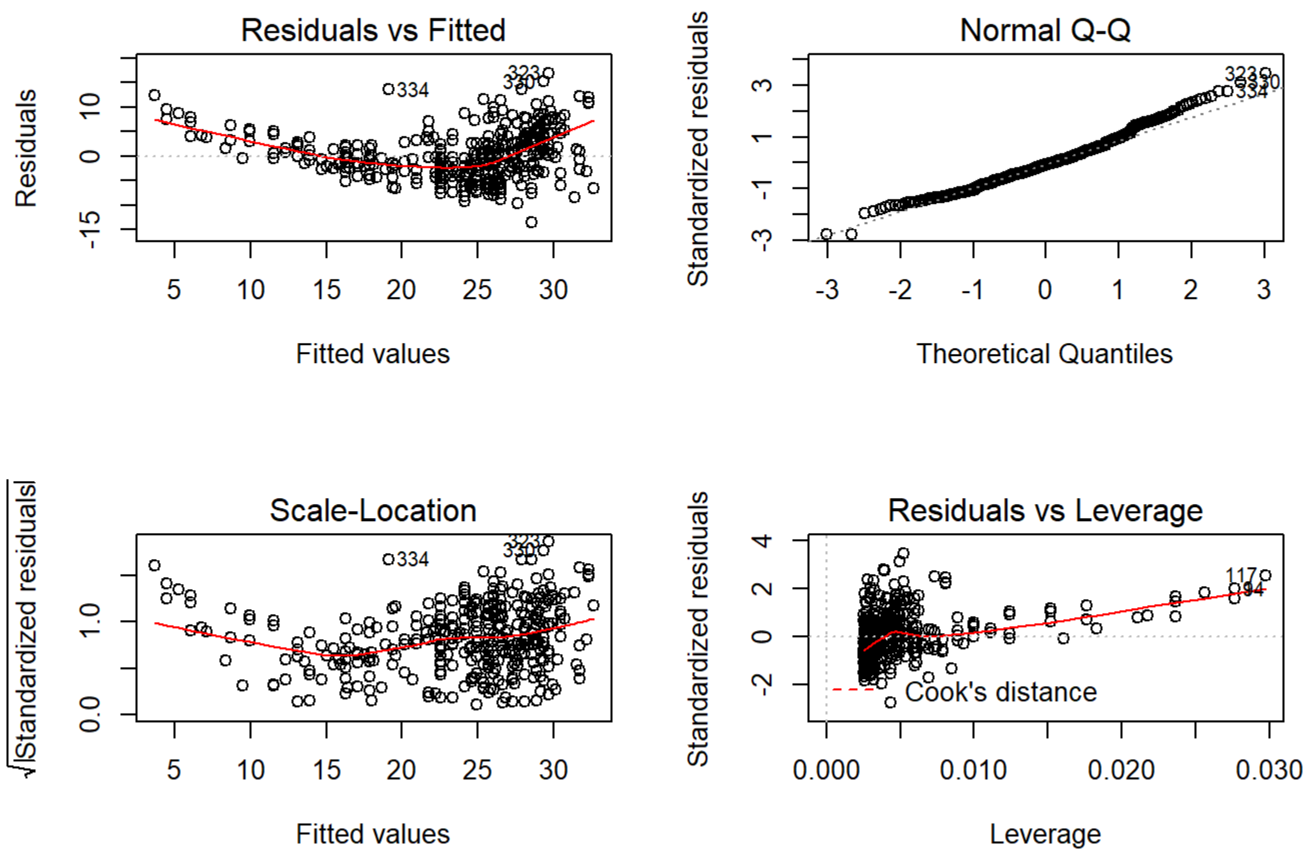
decreases by 0.158.
iv. A horsepower of 98 will have a predicted mpg of 24.47 mpg. The 95% confidence interval is (23.97308, 24.96108) and the 95% prediction interval is (14.8094, 34.12476).

```
#8b
plot(Auto$horsepower, Auto$mpg)
abline(auto_slm, lwd = 2, col = "blue")
```



8b. A line fits the data pretty well, but we do see the data curve at high values of horsepower.

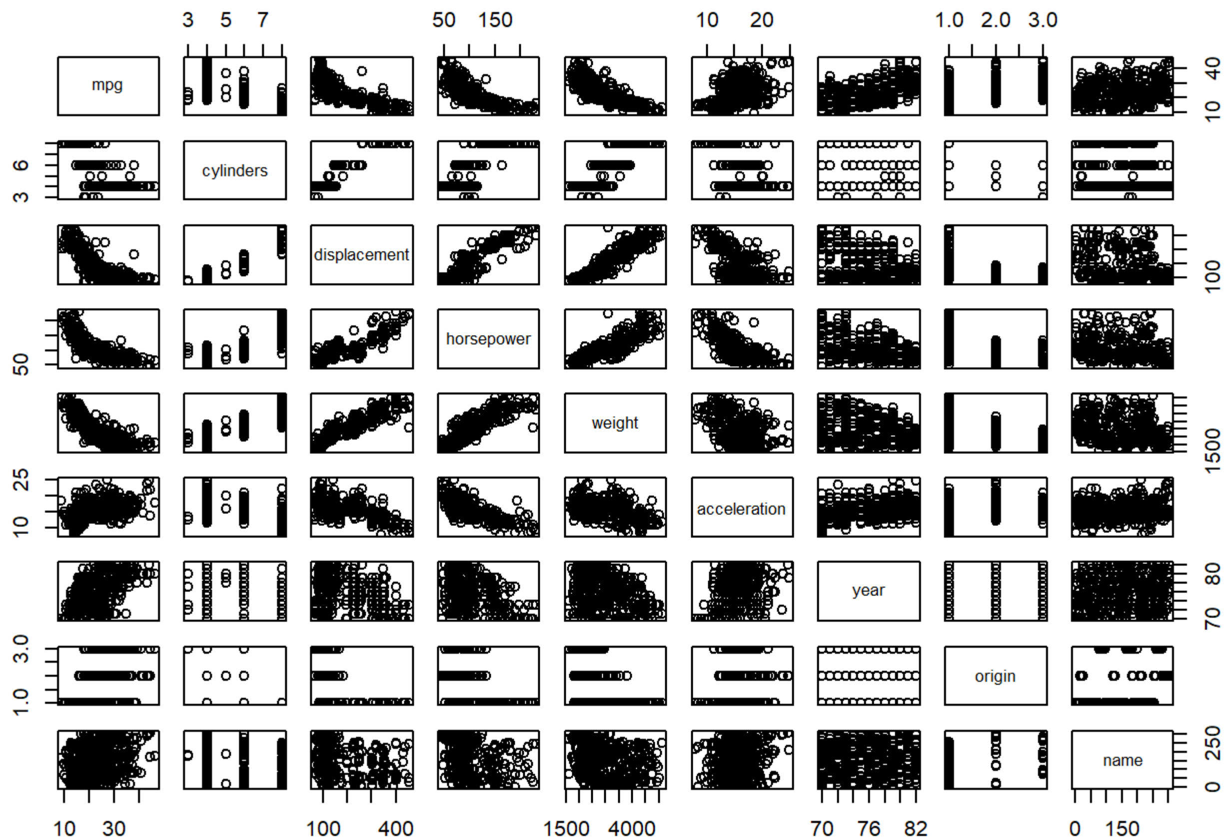
```
#8c
par(mfrow=c(2,2))
plot(auto_slm)
```



8c. The residuals v fitted plot shows funneling and a curvilinear pattern. There also appear to be observations with a significant pull, particularly observation 117, based on the residuals vs. leverage plot.

9.

```
#9a

pairs(Auto)
```



9a. There appears to be relationships between mpg and displacement, horsepower, and weight. They appear to be mostly linear, but with a slight curve.

```
#9b
auto_noname <- Auto %>% dplyr::select(-name)
cor(auto_noname, use = "complete.obs")
```

```
##                     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
```

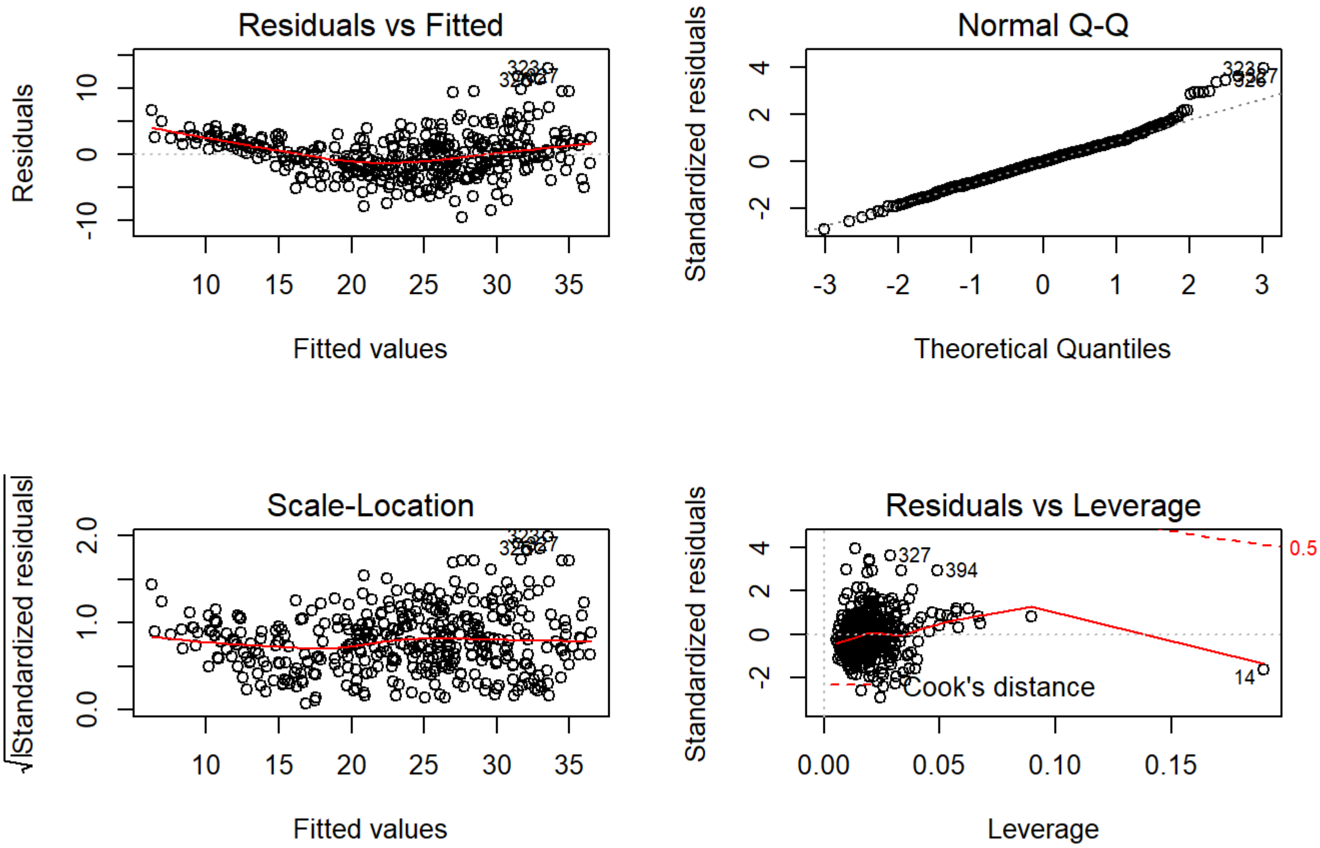
9b. Considering |r| > 0.8 as having a strong linear relationship, mpg has a strong linear relationship with displacement and weight. Similarly, cylinders also has a strong linear relationship with displacement, horsepower, and weight. Displacement also has a strong relationship with horsepower and weight. Horsepower also has a strong relationship with weight.

```
#9c
all_auto_lm <- lm(mpg ~ cylinders + displacement + horsepower + weight + acceleration + year + o
rigin, data = Auto)
summary(all_auto_lm)
```

```
##
## Call:
## lm(formula = mpg ~ cylinders + displacement + horsepower + weight +
##     acceleration + year + origin, 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
##   (5 observations deleted due to missingness)
## Multiple R-squared:  0.8215, Adjusted R-squared:  0.8182
## F-statistic: 252.4 on 7 and 384 DF,  p-value: < 2.2e-16
```

9c. i. The overall model is significant (F = 252.4, p << 0.05), and it has a high Adjusted R-squared of 81.82%. Therefore, the predictors appear to have a strong linear relationship with mpg.
ii. Displacement, Weight, Year, and Origin appear to have a significant linear relationship with mpg.
iii. For every one year, mpg increased by 0.751 mpg.

```
#9d
par(mfrow=c(2,2))
plot(all_auto_lm)
```



9d. The Residuals vs Fitted plot shows that the relationship does appear to have a curvilinear pattern, and the leverage plot points to observation 14 as having unusually high leverage.

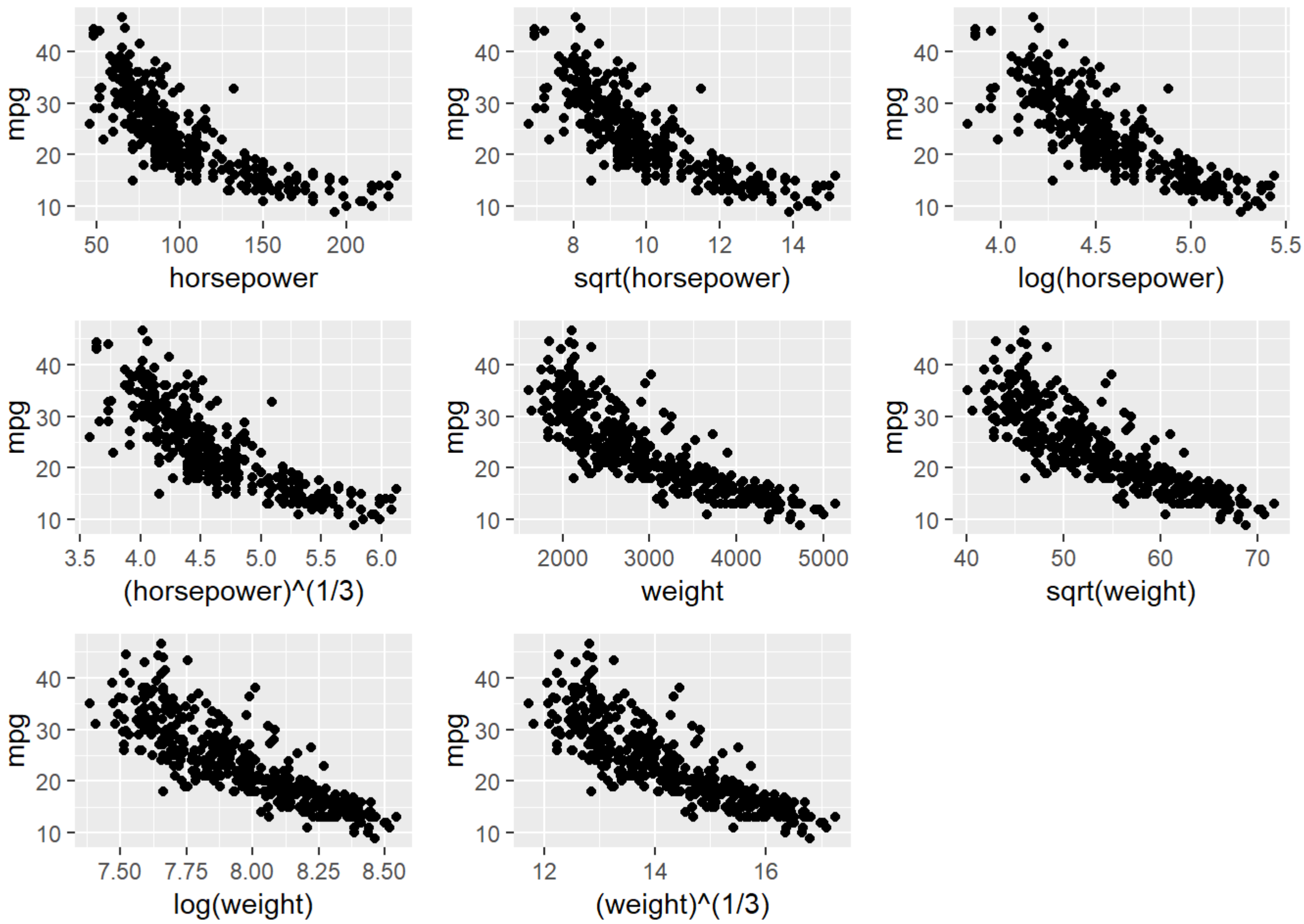
```
int_model_1 <- lm(mpg ~ cylinders + displacement + horsepower + weight + acceleration + year + o
rigin + cylinders:weight + displacement*weight, data = Auto)
summary(int_model_1)
```

```
##
## Call:
## lm(formula = mpg ~ cylinders + displacement + horsepower + weight +
##     acceleration + year + origin + cylinders:weight + displacement *
##     weight, data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.8292 -1.8274 -0.1202  1.6231 12.1608
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -6.360e+00  6.038e+00  -1.053 0.292864
## cylinders            4.592e-01  1.519e+00   0.302 0.762535
## displacement        -7.319e-02  2.374e-02  -3.083 0.002195 **
## horsepower          -3.274e-02  1.240e-02  -2.640 0.008619 **
## weight              -1.031e-02  1.631e-03  -6.320 7.31e-10 ***
## acceleration         6.511e-02  8.864e-02   0.735 0.463044
## year                 7.851e-01  4.559e-02  17.223  < 2e-16 ***
## origin               5.530e-01  2.649e-01   2.088 0.037488 *
## cylinders:weight    -1.016e-04  4.429e-04  -0.229 0.818721
## displacement:weight  2.401e-05  6.152e-06   3.902 0.000113 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.967 on 382 degrees of freedom
##   (5 observations deleted due to missingness)
## Multiple R-squared:  0.8588, Adjusted R-squared:  0.8554
## F-statistic: 258.1 on 9 and 382 DF,  p-value: < 2.2e-16
```

9e. The two models show that cylinders and weight do not have an interaction ($t = -0.229$, $p > 0.05$), but that displacement and weight does have an interaction, meaning that weight affects the slope of displacemnet on mpg ($t = 3.90$, $p < 0.05$). In the second model we see that cylinders and horsepower have a significant interaction ($t = 2.95$, $p < 0.05$), as does horsepower and weight ($t = 2.83$, $p < 0.05$). The second model also has a higher Adjusted R-squared.

```
p1 <- Auto %>% ggplot(aes(horsepower, mpg)) + geom_point()
p2 <- Auto %>% ggplot(aes(sqrt(horsepower), mpg)) + geom_point()
p3 <- Auto %>% ggplot(aes(log(horsepower), mpg)) + geom_point()
p4 <- Auto %>% ggplot(aes((horsepower)^(1/3), mpg)) + geom_point()

p5 <- Auto %>% ggplot(aes(weight, mpg)) + geom_point()
p6 <- Auto %>% ggplot(aes(sqrt(weight), mpg)) + geom_point()
p7 <- Auto %>% ggplot(aes(log(weight), mpg)) + geom_point()
p8 <- Auto %>% ggplot(aes((weight)^(1/3), mpg)) + geom_point()

grid.arrange(p1, p2, p3, p4, p5, p6, p7, p8)
```

```
## Warning: Removed 5 rows containing missing values (geom_point).

## Warning: Removed 5 rows containing missing values (geom_point).

## Warning: Removed 5 rows containing missing values (geom_point).

## Warning: Removed 5 rows containing missing values (geom_point).
```



9f. From the plots above we see that a log transformation of horsepower and a cuberoot transformation of weight would make the data the most linear.

14.

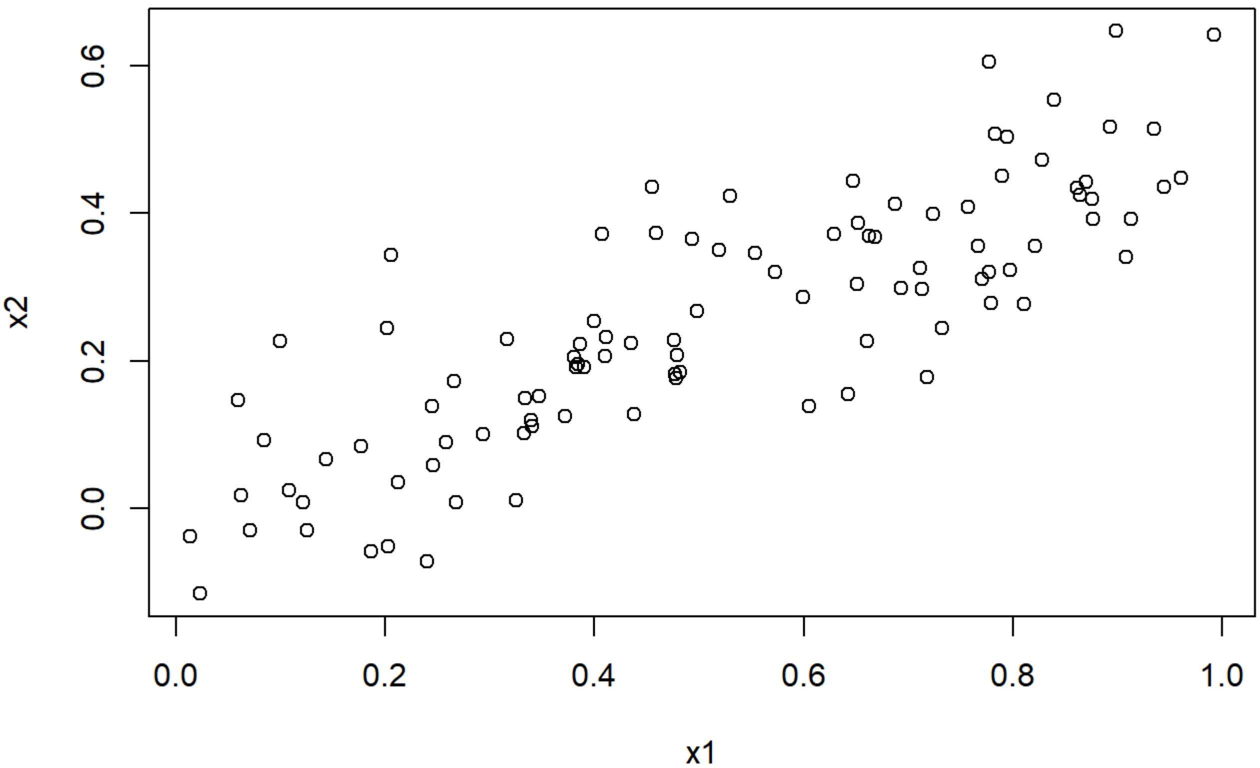
```
#14a
set.seed(1)
x1=runif (100)
x2=0.5*x1+rnorm (100)/10
y=2+2*x1+0.3*x2+rnorm (100)
```

14a. The form of the linear model is $Y = 2 + 2X_1 + 0.3X_2 + \epsilon$, where ϵ ~ N(0,1). The correlation coefficients are as follows: β_0 = 2, β_1 = 2, β_2 = 0.3.

```
# 14b
cor(x1, x2)
```

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

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



14b. X_1 and X_2 have a correlation of 0.835, and the scatterplot confirms a strong, positive linear trend.

```
#14c
col_lm <- lm(y ~ x1 + x2)
summary(col_lm)
```

```
##
## 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
```

14c. $\hat{B_0}$ = 2.13, $\hat{B_1}$ = 1.44, and $\hat{B_0}$ = 1.01, which is quite different from the true values. The null hypothesis $\beta_1 = 0$ can be rejected (t = 1.996, p < 0.05). However, the null hypothesis $\beta_2 = 0$ can not be rejected (t = 0.891, p > 0.05).

```
#14d
col_lm_2 <- lm(y ~ x1)
summary(col_lm_2)
```

```
##
## 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
```

14d. $\hat{B_0}$ = 2.11 and $\hat{B_1}$ = 1.98, which is closer to the true values. We can also reject the null hypothesis $\beta_1 = 0$ with t = 4.986, p << 0.05.

```r
#14e
col_lm_3 <- lm(y ~ x2)
summary(col_lm_3)
```

```
##
## 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
```

14e. $\hat{B}_0$ = 2.39 and $\hat{B}_1$ = 2.90. We can also reject the null hypothesis $\beta_1 = 0$ with t = 4.58, p << 0.05.
14f. The results do not contradict each other because X_1 and X_2 are correlated, thus reducing the power of the test. This makes it more difficult to understand how they are individually related to the dependent variable, and also increases the standard error.

```r
# 14g
x1=c(x1, 0.1)
x2=c(x2, 0.8)
y=c(y,6)

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

```
##
## 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
```

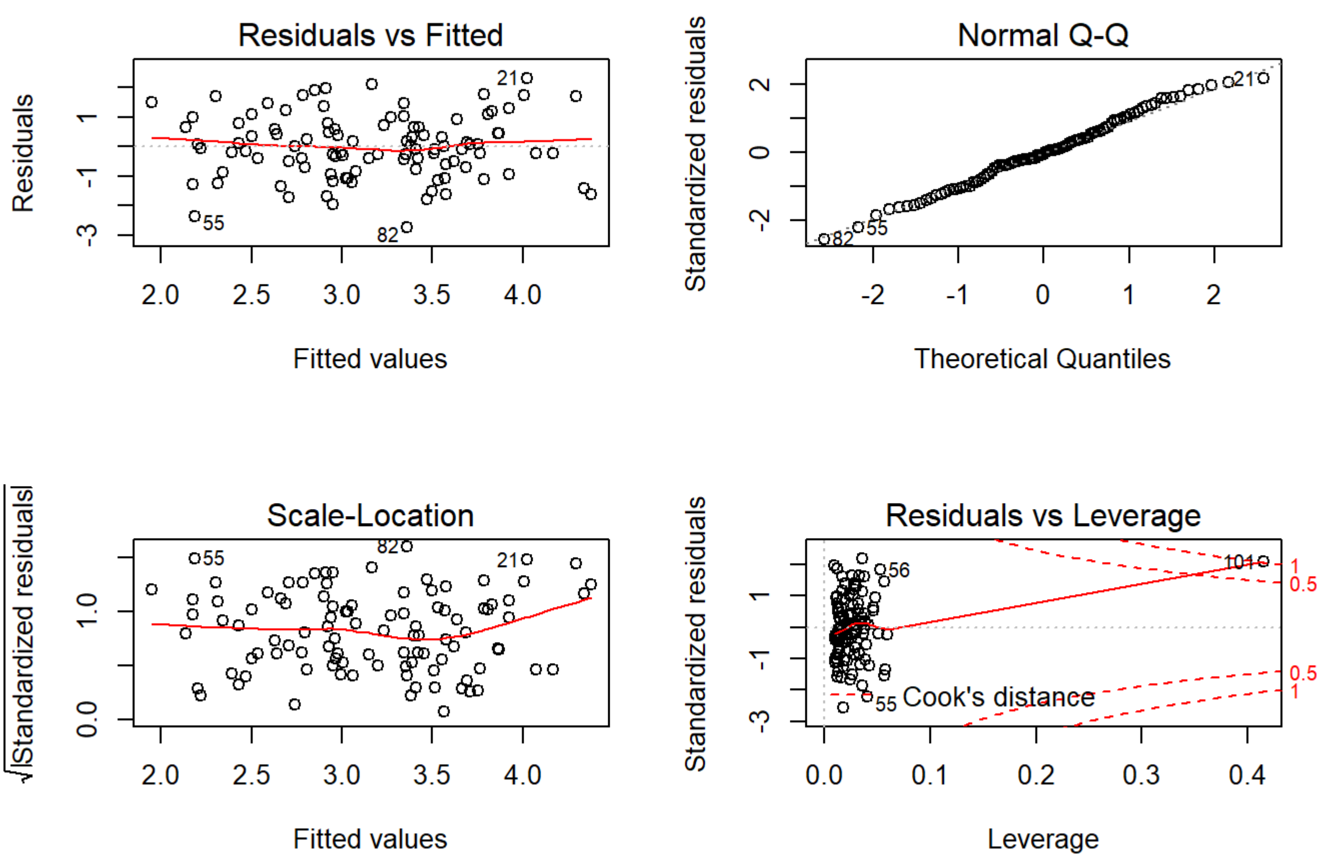
```
col_lm_2 <- lm(y ~ x1)
summary(col_lm_2)
```

```
##
## 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
```

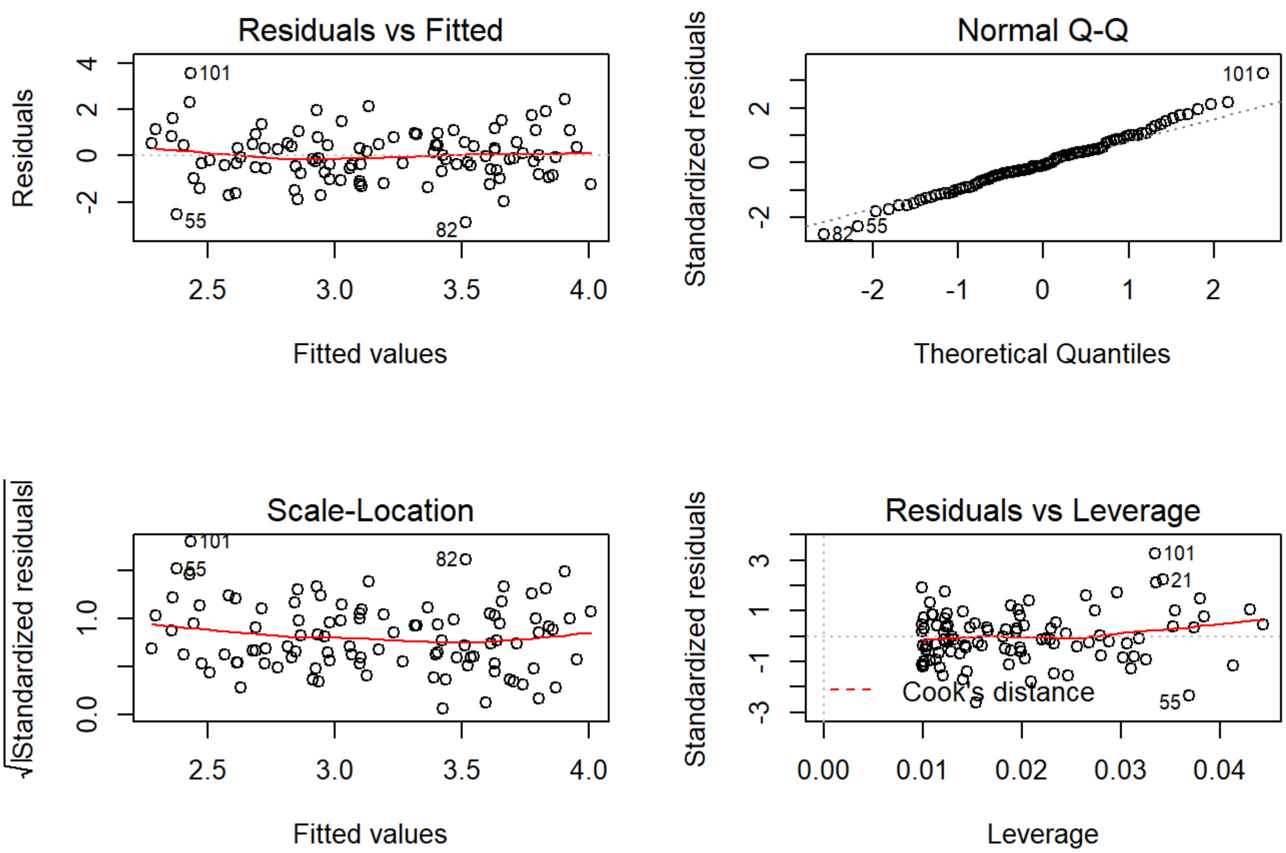
```
col_lm_3 <- lm(y ~ x2)
summary(col_lm_3)
```

```
##
## 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
```

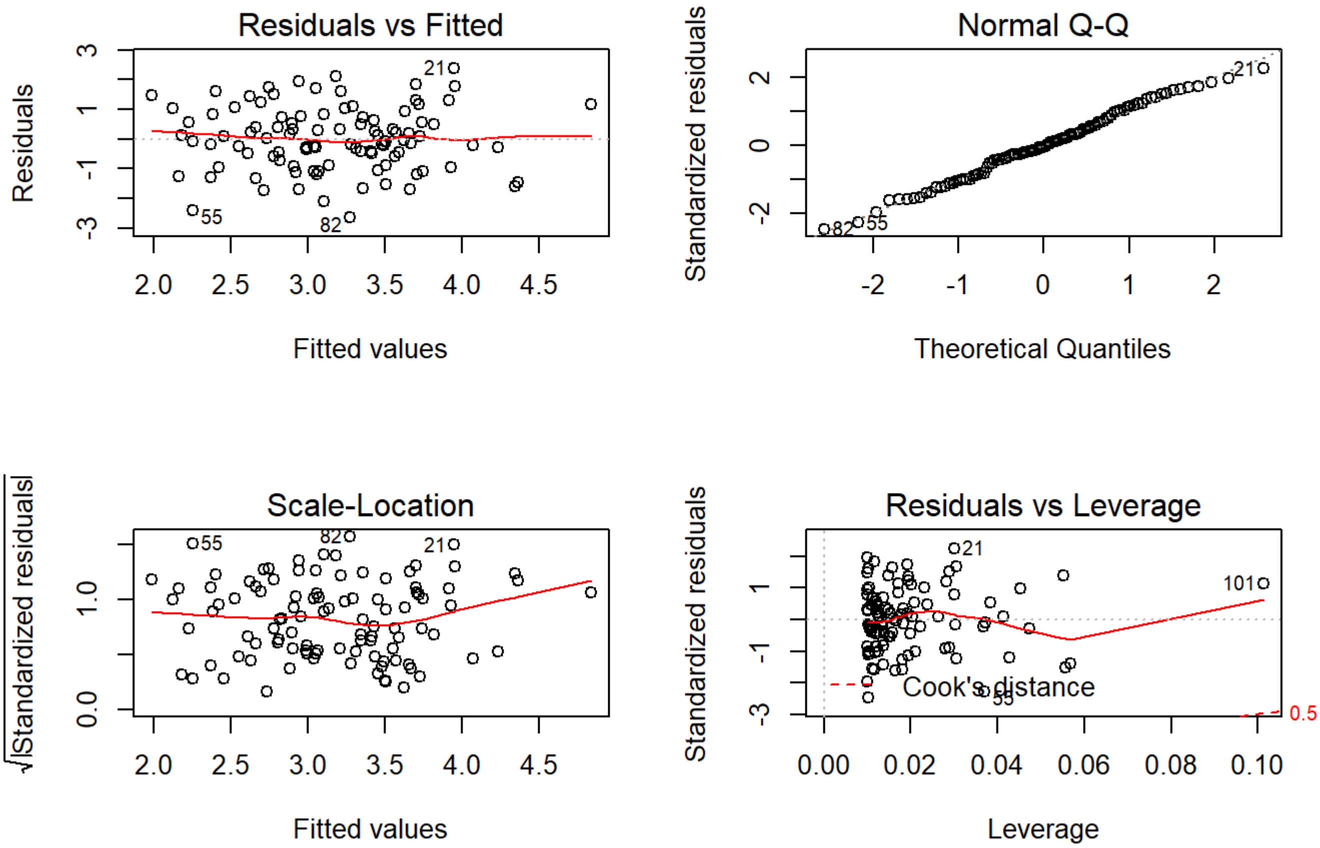
```
par(mfrow=c(2,2))
plot(col_lm)
```




```
par(mfrow=c(2,2))
plot(col_lm_2)
```




```
par(mfrow=c(2,2))
plot(col_lm_3)
```



g. In the full model, the new point reduces the slope of both X_1 and X_2; further, X_2 is now a significant predictor, and not X_1 as before. For the model with only X_1 as a predictor, $\hat{B}_1$ = 1.57 and it is still significant (t = 3.69, p < 0.05). For the model with only X_2 as a predictor, $\hat{B}_1$ = 3.30 and it is still significant (t = 5.70, p < 0.05). This new observation is a high leverage point and an outlier for the full model, and just has high leverage for the model with only X_2 as a predictor.

```r
library(MASS)
```

```
## Warning: package 'MASS' was built under R version 3.5.2
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```r
Boston
```

15.

```r
names(Boston)
```

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

```
lapply(c("zn", "indus", "nox", "chas", "rm", "age", "dis", "rad", "tax", "ptratio", "black", "ls
tat", "medv"),

       function(var) {

           formula    <- as.formula(paste("crim ~", var))
           boston_lm <- lm(formula, data = Boston)
           summary(boston_lm)
       })
```

```
## [[1]]
## 
## Call:
## lm(formula = formula, data = Boston)
## 
## Residuals:
##    Min     1Q Median     3Q    Max 
## -4.429 -4.222 -2.620  1.250 84.523 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  4.45369    0.41722  10.675  < 2e-16 ***
## zn          -0.07393    0.01609  -4.594 5.51e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 8.435 on 504 degrees of freedom
## Multiple R-squared:  0.04019,    Adjusted R-squared:  0.03828 
## F-statistic:  21.1 on 1 and 504 DF,  p-value: 5.506e-06
## 
## 
## [[2]]
## 
## Call:
## lm(formula = formula, data = Boston)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -11.972  -2.698  -0.736   0.712  81.813 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -2.06374    0.66723  -3.093  0.00209 ** 
## indus        0.50978    0.05102   9.991  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 7.866 on 504 degrees of freedom
## Multiple R-squared:  0.1653, Adjusted R-squared:  0.1637 
## F-statistic: 99.82 on 1 and 504 DF,  p-value: < 2.2e-16
## 
## 
## [[3]]
## 
## Call:
## lm(formula = formula, data = Boston)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -12.371  -2.738  -0.974   0.559  81.728 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  -13.720      1.699  -8.073 5.08e-15 ***
## nox           31.249      2.999  10.419  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.81 on 504 degrees of freedom
## Multiple R-squared:  0.1772, Adjusted R-squared:  0.1756
## F-statistic: 108.6 on 1 and 504 DF,  p-value: < 2.2e-16
##
##
## [[4]]
##
## Call:
## lm(formula = formula, data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.738 -3.661 -3.435  0.018 85.232
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.7444     0.3961   9.453   <2e-16 ***
## chas         -1.8928     1.5061  -1.257    0.209
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.597 on 504 degrees of freedom
## Multiple R-squared:  0.003124,   Adjusted R-squared:  0.001146
## F-statistic: 1.579 on 1 and 504 DF,  p-value: 0.2094
##
##
## [[5]]
##
## Call:
## lm(formula = formula, data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.604 -3.952 -2.654  0.989 87.197
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   20.482      3.365   6.088 2.27e-09 ***
## rm            -2.684      0.532  -5.045 6.35e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.401 on 504 degrees of freedom
## Multiple R-squared:  0.04807,    Adjusted R-squared:  0.04618
## F-statistic: 25.45 on 1 and 504 DF,  p-value: 6.347e-07
##
##
## [[6]]
##
```

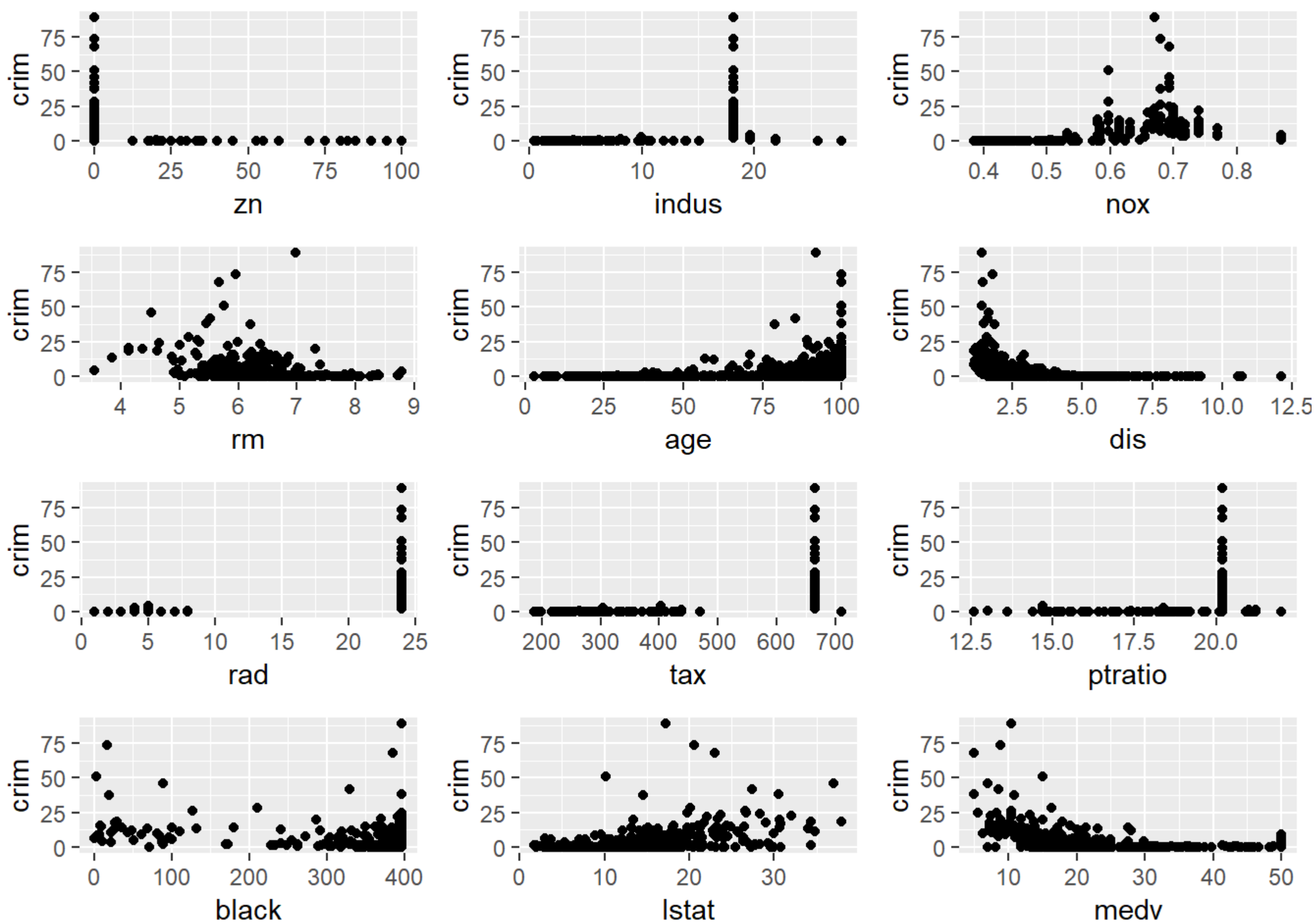
```
## Call:
## lm(formula = formula, data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.789 -4.257 -1.230  1.527 82.849
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.77791    0.94398  -4.002 7.22e-05 ***
## age          0.10779    0.01274   8.463 2.85e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.057 on 504 degrees of freedom
## Multiple R-squared:  0.1244, Adjusted R-squared:  0.1227
## F-statistic: 71.62 on 1 and 504 DF,  p-value: 2.855e-16
##
##
## [[7]]
##
## Call:
## lm(formula = formula, data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.708 -4.134 -1.527  1.516 81.674
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.4993     0.7304  13.006   <2e-16 ***
## dis          -1.5509     0.1683  -9.213   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.965 on 504 degrees of freedom
## Multiple R-squared:  0.1441, Adjusted R-squared:  0.1425
## F-statistic: 84.89 on 1 and 504 DF,  p-value: < 2.2e-16
##
##
## [[8]]
##
## Call:
## lm(formula = formula, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.164  -1.381  -0.141   0.660  76.433
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.28716    0.44348  -5.157 3.61e-07 ***
## rad          0.61791    0.03433  17.998  < 2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.718 on 504 degrees of freedom
## Multiple R-squared:  0.3913, Adjusted R-squared:   0.39
## F-statistic: 323.9 on 1 and 504 DF,  p-value: < 2.2e-16
##
##
## [[9]]
##
## Call:
## lm(formula = formula, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -12.513  -2.738  -0.194   1.065  77.696
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -8.528369   0.815809  -10.45   <2e-16 ***
## tax          0.029742   0.001847   16.10   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.997 on 504 degrees of freedom
## Multiple R-squared:  0.3396, Adjusted R-squared:  0.3383
## F-statistic: 259.2 on 1 and 504 DF,  p-value: < 2.2e-16
##
##
## [[10]]
##
## Call:
## lm(formula = formula, data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -7.654 -3.985 -1.912  1.825 83.353
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -17.6469     3.1473  -5.607 3.40e-08 ***
## ptratio       1.1520     0.1694   6.801 2.94e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.24 on 504 degrees of freedom
## Multiple R-squared:  0.08407,    Adjusted R-squared:  0.08225
## F-statistic: 46.26 on 1 and 504 DF,  p-value: 2.943e-11
##
##
## [[11]]
##
## Call:
## lm(formula = formula, data = Boston)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.756  -2.299  -2.095  -1.296  86.822
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 16.553529   1.425903  11.609   <2e-16 ***
## black       -0.036280   0.003873  -9.367   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.946 on 504 degrees of freedom
## Multiple R-squared:  0.1483, Adjusted R-squared:  0.1466
## F-statistic: 87.74 on 1 and 504 DF,  p-value: < 2.2e-16
##
##
## [[12]]
##
## Call:
## lm(formula = formula, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.925  -2.822  -0.664   1.079  82.862
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.33054    0.69376  -4.801 2.09e-06 ***
## lstat        0.54880    0.04776  11.491  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.664 on 504 degrees of freedom
## Multiple R-squared:  0.2076, Adjusted R-squared:  0.206
## F-statistic:   132 on 1 and 504 DF,  p-value: < 2.2e-16
##
##
## [[13]]
##
## Call:
## lm(formula = formula, data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.071 -4.022 -2.343  1.298 80.957
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 11.79654    0.93419   12.63   <2e-16 ***
## medv        -0.36316    0.03839   -9.46   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.934 on 504 degrees of freedom
```

```
## Multiple R-squared:  0.1508, Adjusted R-squared:  0.1491
## F-statistic: 89.49 on 1 and 504 DF,  p-value: < 2.2e-16
```

```
p1 <- Boston %>% ggplot(aes(zn, crim)) + geom_point()
p2 <- Boston %>% ggplot(aes(indus, crim)) + geom_point()
p3 <- Boston %>% ggplot(aes(nox, crim)) + geom_point()
p4 <- Boston %>% ggplot(aes(rm, crim)) + geom_point()
p5 <- Boston %>% ggplot(aes(age, crim)) + geom_point()
p6 <- Boston %>% ggplot(aes(dis, crim)) + geom_point()
p7 <- Boston %>% ggplot(aes(rad, crim)) + geom_point()
p8 <- Boston %>% ggplot(aes(tax, crim)) + geom_point()
p9 <- Boston %>% ggplot(aes(ptratio, crim)) + geom_point()
p10 <- Boston %>% ggplot(aes(black, crim)) + geom_point()
p11 <- Boston %>% ggplot(aes(lstat, crim)) + geom_point()
p12 <- Boston %>% ggplot(aes(medv, crim)) + geom_point()
grid.arrange(p1, p2, p3, p4, p5, p6, p7, p8, p9, p10, p11, p12)
```



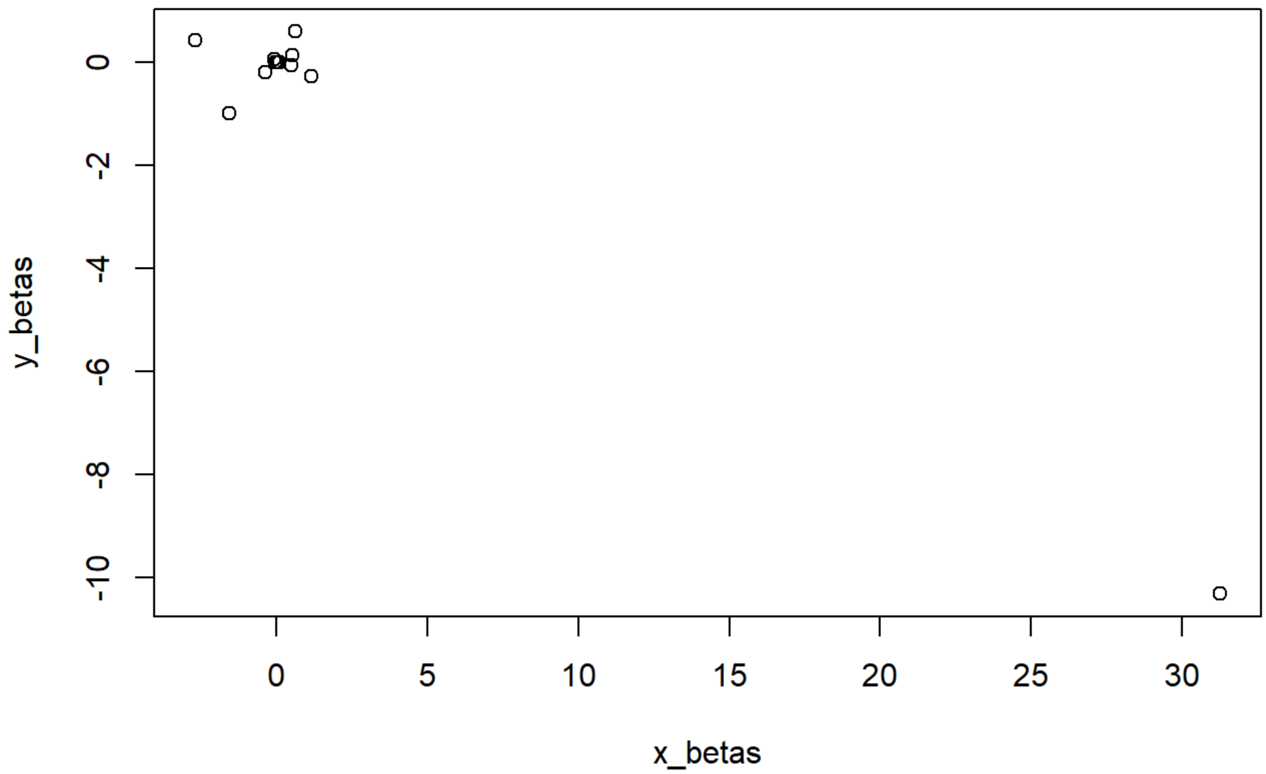
15a. zn, indus, nox, rm, age, dis, rad, tax, ptratio, black, lstat, medv are all significant predictors of crime. The scatterplots show that a lot of the variables have outliers. However, of note, dis appears to be negatively related with crime and lstat, rm, and age are positively correlated with crime.

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

```
##
## 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
```

15b. In the full model, we see that only zn, dis, rad, black, and medv are significantly related to crime and we can reject the null hypothesis β_i = 0.

```
#15c zn, indus, nox, rm, age, dis, rad, tax, ptratio, black, lstat, medv

indvar <- c("zn", "indus", "nox", "rm", "age", "dis", "rad", "tax", "ptratio", "black", "lstat",
 "medv")
x_betas <- lapply(indvar, function(dv) {
    boston_lm <- lm(crim ~ get(dv), data = Boston)
    boston_lm$coefficients[2]
})


y_betas <- full_crime_lm$coefficients
y_betas <- y_betas[-c(1,4)]

plot(x_betas, y_betas)
```



15c. Fewer of the variables are significantly related to crime in the full model than in the univariate regressions. As multicollinearity reduces power, this may be a reason fewer are significant. The plot also shows that the betas are not the same, which may be related to the interpretation of a multiple regression vs. a simple regression. In simple regression, we do not take other predictors into account and get the base average increase of a dependent variable given the predictor. in contrast, a multiple regression gives the average increase in the dependent variable *while holding the other variables constant.*

```
#15d
lm_zn <- lm(crim ~ poly(zn, 3), data = Boston)




indvar <- c("zn", "indus", "nox", "rm", "age", "dis", "rad", "tax", "ptratio", "black", "lstat",
 "medv")
lapply(indvar, function(var) {
    boston_lm <- lm(crim ~ poly(get(var), 3), data = Boston)
    summary(boston_lm)

})
```

```
## [[1]]
##
## Call:
## lm(formula = crim ~ poly(get(var), 3), data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.821 -4.614 -1.294  0.473 84.130
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)           3.6135     0.3722   9.709  < 2e-16 ***
## poly(get(var), 3)1 -38.7498     8.3722  -4.628  4.7e-06 ***
## poly(get(var), 3)2  23.9398     8.3722   2.859  0.00442 **
## poly(get(var), 3)3 -10.0719     8.3722  -1.203  0.22954
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.372 on 502 degrees of freedom
## Multiple R-squared:  0.05824,    Adjusted R-squared:  0.05261
## F-statistic: 10.35 on 3 and 502 DF,  p-value: 1.281e-06
##
##
## [[2]]
##
## Call:
## lm(formula = crim ~ poly(get(var), 3), data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -8.278 -2.514  0.054  0.764 79.713
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)            3.614      0.330  10.950  < 2e-16 ***
## poly(get(var), 3)1    78.591      7.423  10.587  < 2e-16 ***
## poly(get(var), 3)2   -24.395      7.423  -3.286  0.00109 **
## poly(get(var), 3)3   -54.130      7.423  -7.292  1.2e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.423 on 502 degrees of freedom
## Multiple R-squared:  0.2597, Adjusted R-squared:  0.2552
## F-statistic: 58.69 on 3 and 502 DF,  p-value: < 2.2e-16
##
##
## [[3]]
##
## Call:
## lm(formula = crim ~ poly(get(var), 3), data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
```

```
## -9.110 -2.068 -0.255  0.739 78.302
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.6135     0.3216  11.237  < 2e-16 ***
## poly(get(var), 3)1  81.3720     7.2336  11.249  < 2e-16 ***
## poly(get(var), 3)2 -28.8286     7.2336  -3.985 7.74e-05 ***
## poly(get(var), 3)3 -60.3619     7.2336  -8.345 6.96e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.234 on 502 degrees of freedom
## Multiple R-squared:  0.297,  Adjusted R-squared:  0.2928
## F-statistic: 70.69 on 3 and 502 DF,  p-value: < 2.2e-16
##
##
## [[4]]
##
## Call:
## lm(formula = crim ~ poly(get(var), 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.485  -3.468  -2.221  -0.015  87.219
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.6135     0.3703   9.758  < 2e-16 ***
## poly(get(var), 3)1 -42.3794     8.3297  -5.088 5.13e-07 ***
## poly(get(var), 3)2  26.5768     8.3297   3.191  0.00151 **
## poly(get(var), 3)3  -5.5103     8.3297  -0.662  0.50858
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.33 on 502 degrees of freedom
## Multiple R-squared:  0.06779,    Adjusted R-squared:  0.06222
## F-statistic: 12.17 on 3 and 502 DF,  p-value: 1.067e-07
##
##
## [[5]]
##
## Call:
## lm(formula = crim ~ poly(get(var), 3), data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.762 -2.673 -0.516  0.019 82.842
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.6135     0.3485  10.368  < 2e-16 ***
## poly(get(var), 3)1  68.1820     7.8397   8.697  < 2e-16 ***
## poly(get(var), 3)2  37.4845     7.8397   4.781 2.29e-06 ***
## poly(get(var), 3)3  21.3532     7.8397   2.724  0.00668 **
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.84 on 502 degrees of freedom
## Multiple R-squared:  0.1742, Adjusted R-squared:  0.1693
## F-statistic: 35.31 on 3 and 502 DF,  p-value: < 2.2e-16
##
##
## [[6]]
##
## Call:
## lm(formula = crim ~ poly(get(var), 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.757  -2.588   0.031   1.267  76.378
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.6135     0.3259  11.087  < 2e-16 ***
## poly(get(var), 3)1 -73.3886     7.3315 -10.010  < 2e-16 ***
## poly(get(var), 3)2  56.3730     7.3315   7.689 7.87e-14 ***
## poly(get(var), 3)3 -42.6219     7.3315  -5.814 1.09e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.331 on 502 degrees of freedom
## Multiple R-squared:  0.2778, Adjusted R-squared:  0.2735
## F-statistic: 64.37 on 3 and 502 DF,  p-value: < 2.2e-16
##
##
## [[7]]
##
## Call:
## lm(formula = crim ~ poly(get(var), 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.381  -0.412  -0.269   0.179  76.217
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.6135     0.2971  12.164  < 2e-16 ***
## poly(get(var), 3)1 120.9074     6.6824  18.093  < 2e-16 ***
## poly(get(var), 3)2  17.4923     6.6824   2.618  0.00912 **
## poly(get(var), 3)3   4.6985     6.6824   0.703  0.48231
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.682 on 502 degrees of freedom
## Multiple R-squared:    0.4,  Adjusted R-squared:  0.3965
## F-statistic: 111.6 on 3 and 502 DF,  p-value: < 2.2e-16
##
##
```

```
## [[8]]
##
## Call:
## lm(formula = crim ~ poly(get(var), 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.273  -1.389   0.046   0.536  76.950
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)           3.6135     0.3047  11.860  < 2e-16 ***
## poly(get(var), 3)1 112.6458     6.8537  16.436  < 2e-16 ***
## poly(get(var), 3)2  32.0873     6.8537   4.682 3.67e-06 ***
## poly(get(var), 3)3  -7.9968     6.8537  -1.167    0.244
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.854 on 502 degrees of freedom
## Multiple R-squared:  0.3689, Adjusted R-squared:  0.3651
## F-statistic:  97.8 on 3 and 502 DF,  p-value: < 2.2e-16
##
##
## [[9]]
##
## Call:
## lm(formula = crim ~ poly(get(var), 3), data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.833 -4.146 -1.655  1.408 82.697
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)            3.614      0.361  10.008  < 2e-16 ***
## poly(get(var), 3)1    56.045      8.122   6.901 1.57e-11 ***
## poly(get(var), 3)2    24.775      8.122   3.050  0.00241 **
## poly(get(var), 3)3   -22.280      8.122  -2.743  0.00630 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.122 on 502 degrees of freedom
## Multiple R-squared:  0.1138, Adjusted R-squared:  0.1085
## F-statistic: 21.48 on 3 and 502 DF,  p-value: 4.171e-13
##
##
## [[10]]
##
## Call:
## lm(formula = crim ~ poly(get(var), 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.096  -2.343  -2.128  -1.439  86.790
```

```
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)            3.6135     0.3536  10.218   <2e-16 ***
## poly(get(var), 3)1 -74.4312     7.9546  -9.357   <2e-16 ***
## poly(get(var), 3)2   5.9264     7.9546   0.745    0.457
## poly(get(var), 3)3  -4.8346     7.9546  -0.608    0.544
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.955 on 502 degrees of freedom
## Multiple R-squared:  0.1498, Adjusted R-squared:  0.1448
## F-statistic: 29.49 on 3 and 502 DF,  p-value: < 2.2e-16
##
##
## [[11]]
##
## Call:
## lm(formula = crim ~ poly(get(var), 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.234  -2.151  -0.486   0.066  83.353
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)            3.6135     0.3392  10.654   <2e-16 ***
## poly(get(var), 3)1  88.0697     7.6294  11.543   <2e-16 ***
## poly(get(var), 3)2  15.8882     7.6294   2.082   0.0378 *
## poly(get(var), 3)3 -11.5740     7.6294  -1.517   0.1299
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.629 on 502 degrees of freedom
## Multiple R-squared:  0.2179, Adjusted R-squared:  0.2133
## F-statistic: 46.63 on 3 and 502 DF,  p-value: < 2.2e-16
##
##
## [[12]]
##
## Call:
## lm(formula = crim ~ poly(get(var), 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -24.427  -1.976  -0.437   0.439  73.655
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)             3.614      0.292  12.374  < 2e-16 ***
## poly(get(var), 3)1   -75.058      6.569 -11.426  < 2e-16 ***
## poly(get(var), 3)2    88.086      6.569  13.409  < 2e-16 ***
## poly(get(var), 3)3   -48.033      6.569  -7.312 1.05e-12 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.569 on 502 degrees of freedom
## Multiple R-squared:  0.4202, Adjusted R-squared:  0.4167
## F-statistic: 121.3 on 3 and 502 DF,  p-value: < 2.2e-16
```

15d. All of the models significantly predicted crime, and the Adjusted R-squared of the polynomial models is also generally greater than for the simple linear regression.