Assignment 2 IDS 575

Talish Barmare

10/6/2019

Ex. 2.8 Compare the classification performance of linear regression and k– nearest neighbor classification on the zipcode data. In particular, consider only the 2's and 3's, and k = 1, 3, 5, 7 and 15. Show both the training and test error for each choice. The zipcode data are available from the book website www-stat.stanford.edu/ElemStatLearn.

Solution:

Post building our models and getting our predictions, the below table compares the performance of our Linear regression model with our KNN models with the respect K values.

Error Rate Table

K	KNN.Train	KNN.Test	LR.Train	LR.Test
1	0.000000000	0.02472527	0.005759539	0.04120879
3	0.005039597	0.02472527	0.005759539	0.04120879
7	0.005039597	0.03021978	0.005759539	0.04120879
5	0.005759539	0.03021978	0.005759539	0.04120879
15	0.009359251	0.03846154	0.005759539	0.04120879

Observations:

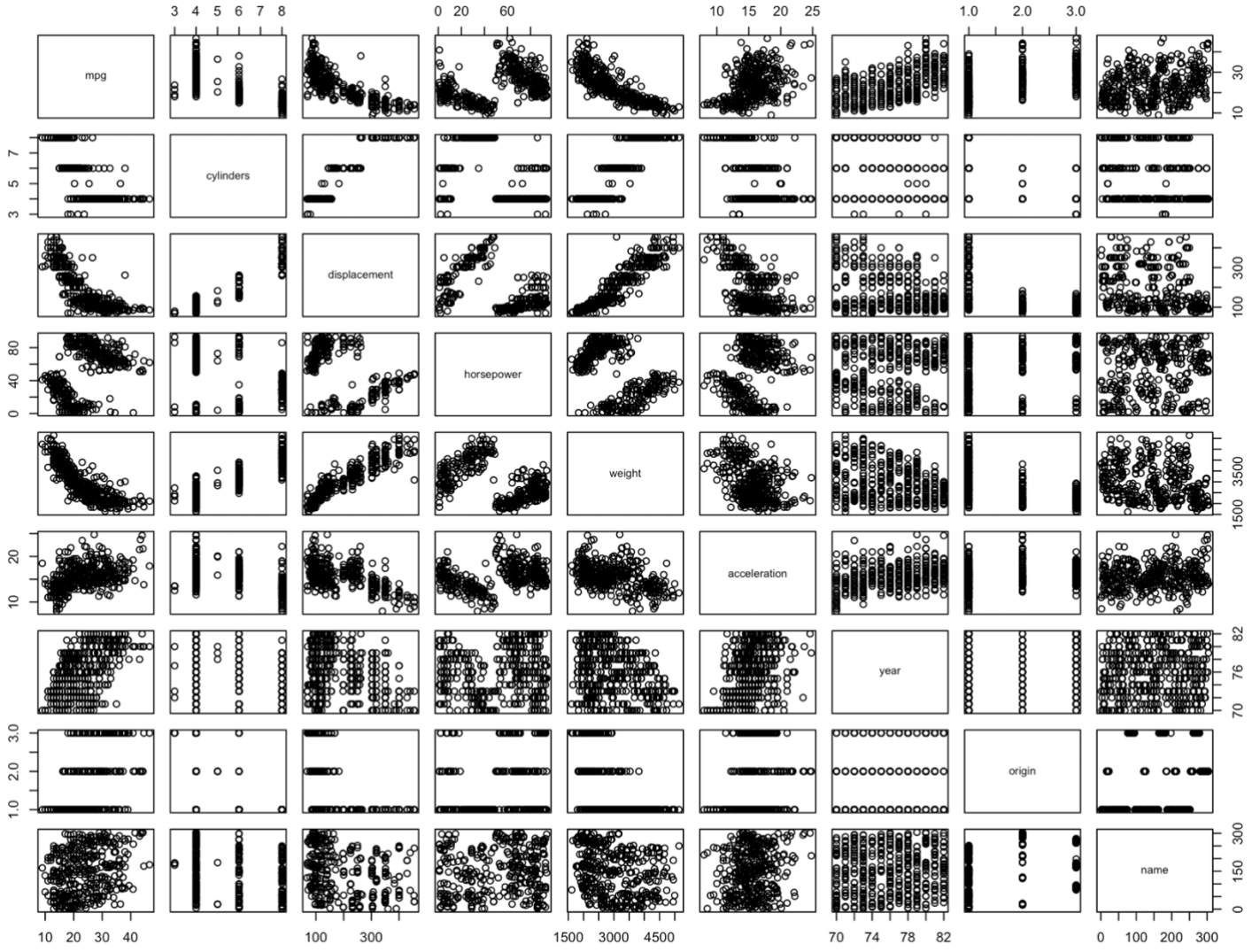
1. In this case, the **k-NN with small k values outperforms linear regression**.
2. In case of k-NN procedures, **the smaller k gives better performance**. This is because of the **Curse of Dimensionality**

3. With 256 features, the data points are spread out so far that often their 'nearest neighbors' are actually not very near them.

Q9. This question involves the use of multiple linear regression on the "Auto" data set.

a) Produce a scatterplot matrix which include all the variables in the data set.

```
auto <- read.csv("Auto.csv")
pairs(auto)
```



b) Compute the matrix of correlations between the variables using the function cor(). You will need to exclude the "name" variable, which is qualitative.

Solution:

The function *names()* was used to determine the name of the variables in the Auto data set. Once the name of the variables were known the qualitative variable, name was removed to compute the correlations matrices for all the variables.

cor(Auto[0:8])

```
> cor(Auto[0:8])
                     mpg  cylinders displacement horsepower     weight acceleration       year     origin
mpg           1.0000000 -0.7776175   -0.8051269 -0.7784268 -0.8322442    0.4233285  0.5805410  0.5652088
cylinders    -0.7776175  1.0000000    0.9508233  0.8429834  0.8975273   -0.5046834 -0.3456474 -0.5689316
displacement -0.8051269  0.9508233    1.0000000  0.8972570  0.9329944   -0.5438005 -0.3698552 -0.6145351
horsepower   -0.7784268  0.8429834    0.8972570  1.0000000  0.8645377   -0.6891955 -0.4163615 -0.4551715
weight       -0.8322442  0.8975273    0.9329944  0.8645377  1.0000000   -0.4168392 -0.3091199 -0.5850054
acceleration  0.4233285 -0.5046834   -0.5438005 -0.6891955 -0.4168392    1.0000000  0.2903161  0.2127458
year          0.5805410 -0.3456474   -0.3698552 -0.4163615 -0.3091199    0.2903161  1.0000000  0.1815277
origin        0.5652088 -0.5689316   -0.6145351 -0.4551715 -0.5850054    0.2127458  0.1815277  1.0000000
```

c)Use the lm() function to perform a multiple linear regression with "mpg" as the response and all other variables except "name" as the predictors. Use the summary() function to print the results. Comment on the output. For instance :

fit <- lm(mpg ~ . - name, data = auto)

summary(fit)

```
> summary(fit)

Call:
lm(formula = mpg ~ . - name, data = Auto)

Residuals:
    Min      1Q  Median      3Q     Max
-9.5903 -2.1565 -0.1169  1.8690 13.0604

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  -17.218435   4.644294  -3.707  0.00024 ***
cylinders     -0.493376   0.323282  -1.526  0.12780
displacement   0.019896   0.007515   2.647  0.00844 **
horsepower    -0.016951   0.013787  -1.230  0.21963
weight        -0.006474   0.000652  -9.929  < 2e-16 ***
acceleration   0.080576   0.098845   0.815  0.41548
year           0.750773   0.050973  14.729  < 2e-16 ***
origin         1.426141   0.278136   5.127 4.67e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.328 on 384 degrees of freedom
Multiple R-squared:  0.8215,    Adjusted R-squared:  0.8182
F-statistic: 252.4 on 7 and 384 DF,  p-value: < 2.2e-16
```

i. Is there a relationship between the predictors and the response?

To determine the relationship between the predictors and the target variable (mpg), it is essential to must the null hypothesis H0:βi=0 for all values of I (the number of predictors).

The output above illustrates that with a F-statistic with the p-value = <.001, we reject the null hypothesis that the predictors are equal to zero; thus there a statistically significant relationship between the predictors and the response variables.

ii. Which predictors appear to have a statistically significant relationship to the response?

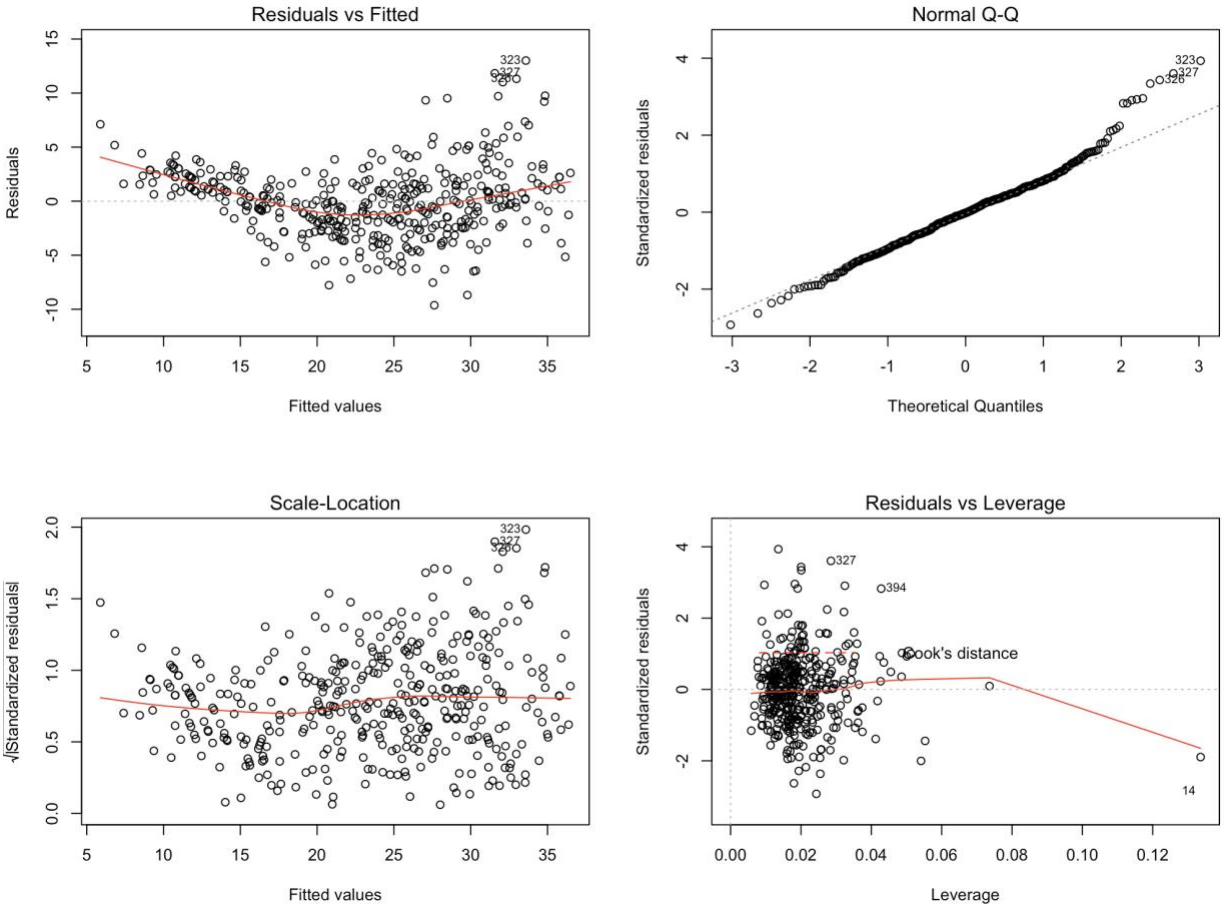
The predictor variables that appear to have statistically significant relationship to the response, mpg are the predictors variables with Pr(>|t|) are less than the level of significance, α = 0.05. The predictors that fit this criteria are displacement (p-value= p-value= 0.0283), weight(p-value=<.001), year(p-value=<.001) and origin(p-value=<.001).

iii. What does the coefficient for the year variable suggest?

The coefficient of the "year" variable suggests that the average effect of an increase of 1 year is an increase of 0.750773 in "mpg" (all other predictors remaining constant). In other words, cars become more fuel efficient every year by almost 1 mpg / year.

d)Use the plot() function to produce diagnostic plots of the linear regression fit. Comment on any problems you see with #the fit. Do the residual plots suggest any unusually large outliers ? Does the leverage plots identify any observations #with unusually high leverages ?

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



As before, the plot of residuals versus fitted values indicates the presence of mild non linearity in the data. The plot of standardized residuals versus leverage indicates the presence of a few outliers (higher than 2 or lower than -2) and one high leverage point 14.

e) Use the * and : symbols to fit linear regression models with interaction effects. Do any interactions appear to be #statistically significant ?

From the correlation matrix, we obtained the two highest correlated pairs and used them in picking interaction effects.

interaction <- lm(mpg~ cylinders*horsepower+displacement*weight+acceleration*horsepower ,data= Auto)

summary(interaction)

```
> summary(interaction)

Call:
lm(formula = mpg ~ cylinders * horsepower + displacement * weight +
    acceleration * horsepower, data = Auto)

Residuals:
     Min       1Q   Median       3Q      Max
-11.7093  -2.1721  -0.4586   1.7839  16.7986

Coefficients:
                          Estimate Std. Error t value Pr(>|t|)
(Intercept)              6.567e+01  7.197e+00   9.125  < 2e-16 ***
cylinders               -3.070e+00  1.035e+00  -2.968 0.003189 **
horsepower              -2.435e-01  8.174e-02  -2.979 0.003077 **
displacement            -4.299e-02  1.648e-02  -2.609 0.009429 **
weight                  -3.405e-03  1.421e-03  -2.396 0.017039 *
acceleration             5.307e-02  2.483e-01   0.214 0.830877
cylinders:horsepower     3.094e-02  8.685e-03   3.562 0.000414 ***
displacement:weight      6.292e-06  4.398e-06   1.431 0.153346
horsepower:acceleration -3.787e-03  2.581e-03  -1.467 0.143221
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.849 on 383 degrees of freedom
Multiple R-squared:  0.7618,    Adjusted R-squared:  0.7568
F-statistic: 153.1 on 8 and 383 DF,  p-value: < 2.2e-16
```

The only interaction that statistically significant in this example is the interaction between the variables cylinders and horsepower with a p-value = 0.000414.

f)Try a few different transformations of the variables, such as logX, $X^{-}\sqrt{}$, X2. Comment on your findings.

The variable that were chosen to be transformed were horsepower and acceleration because these variables were not statistically significant in the original model.

par(mfrow=c(2,3))

plot(log(Auto$horsepower),Auto$mpg)

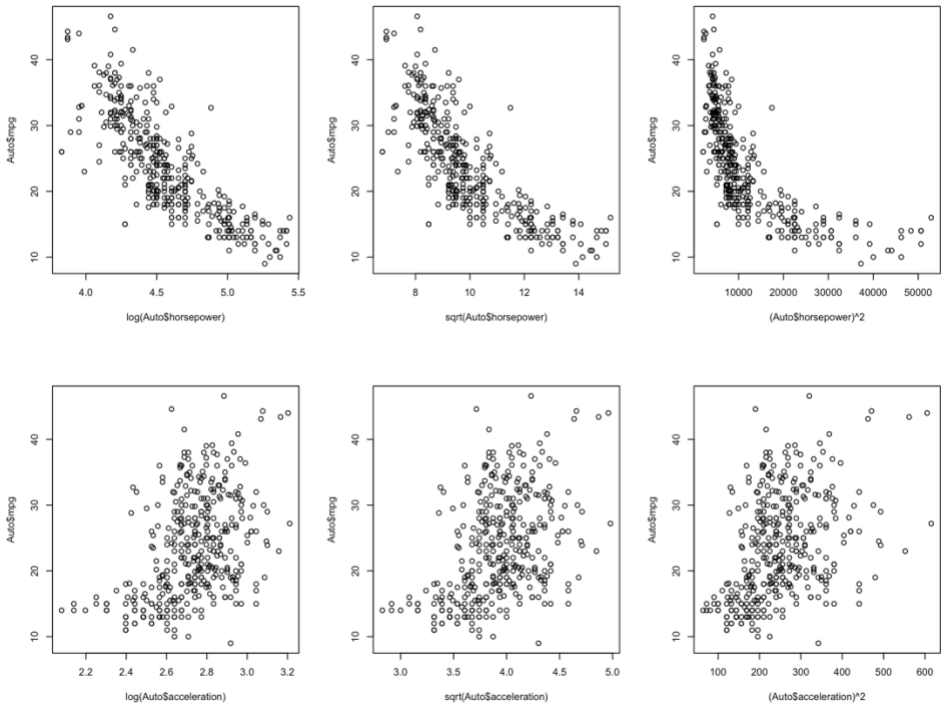
plot(sqrt(Auto$horsepower),Auto$mpg)

plot((Auto$horsepower)^2,Auto$mpg)

plot(log(Auto$acceleration),Auto$mpg)

plot(sqrt(Auto$acceleration),Auto$mpg)

plot((Auto$acceleration)^2,Auto$mpg)



It seems that the log transformation of horsepower gives the most linear looking plot.

Reviewing the plots of each of the transformation verses the response variable mpg; the only transformation that resulted in strongest most linear relationship is the log transformation of horsepower. The transformations of the acceleration variable do not appear to improve the linear relationship or the strength of the relationship between this variable and mpg.

Q15.

This problem involves the "Boston" data set, which we saw in the lab for this chapter. We will now try to predict per capita crime rate using the other variables in this data set. In other words, per capita crime rate is the response, and the other variables are the predictors.

a) For each predictor, fit a simple linear regression model to predict the response. Describe your results. In which of the #models is there a statistically significant association between the predictor and the response? Create some plots to back up your assertions.

library(MASS)

attach(Boston)

Linear models for all variables

lm.zn <- lm(crim ~ zn)

summary(lm.zn)

```
Call:
lm(formula = crim ~ zn)

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
```

lm.indus <- lm(crim ~ indus)

summary(lm.indus)

```
Call:
lm(formula = crim ~ indus)

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
```

---

chas <- as.factor(chas)

lm.chas <- lm(crim ~ chas)

summary(lm.chas)

```
> summary(lm.chas)

Call:
lm(formula = crim ~ chas)

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 ***
chas1        -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
```

lm.nox <- lm(crim ~ nox)

summary(lm.nox)

```
> summary(lm.nox)

Call:
lm(formula = crim ~ nox)

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
```

lm.rm <- lm(crim ~ rm)

summary(lm. rm)

```
> summary(lm.rm)

Call:
lm(formula = crim ~ rm)

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
```

lm.age <- lm(crim ~ age)

summary(lm.age)

```
> summary(lm.age)

Call:
lm(formula = crim ~ age)

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
```

lm.dis <- lm(crim ~ dis)

summary(lm.dis)

```
> summary(lm.dis)

Call:
lm(formula = crim ~ dis)

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
```

lm.rad <- lm(crim ~ rad)

summary(lm.rad)

```
> summary(lm.rad)

Call:
lm(formula = crim ~ rad)

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
```

lm.tax <- lm(crim ~ tax)

summary(lm.tax)

```
> summary(lm.tax)

Call:
lm(formula = crim ~ tax)

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
```

lm.ptratio <- lm(crim ~ ptratio)

summary(lm.ptratio)

```
> summary(lm.ptratio)

Call:
lm(formula = crim ~ ptratio)

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
```

lm.black <- lm(crim ~ black)

summary(lm.black)

```
Call:
lm(formula = crim ~ black)

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
```

lm.lstat <- lm(crim ~ lstat)

summary(lm.lstat)

```
> summary(lm.lstat)

Call:
lm(formula = crim ~ lstat)

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
```

lm.medv <- lm(crim ~ medv)

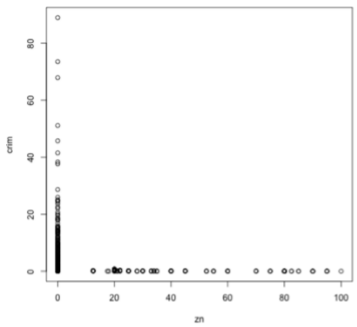
summary(lm.medv)

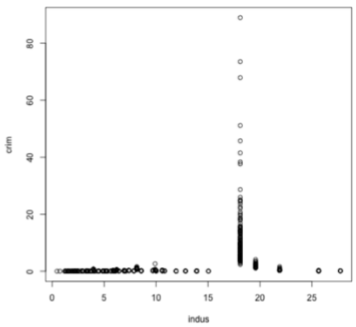
```
> summary(lm.medv)

Call:
lm(formula = crim ~ medv)

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
```

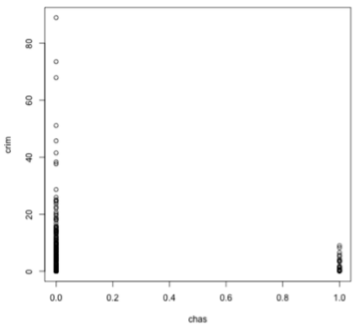
To find which predictors are significant, we have to test H0:β1=0. All predictors have a p-value less than 0.05 except “chas”, so we may conclude that there is a statistically significant association between each predictor and the response except for the “chas” predictor.

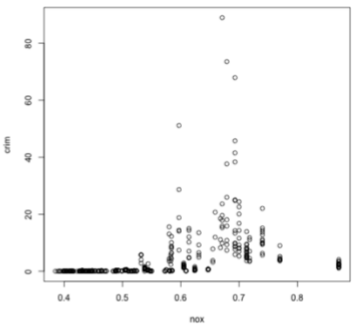
Few Plots:

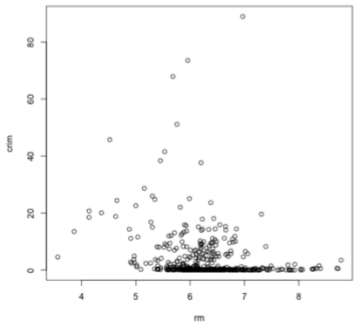
plot(crim ~ . - crim, data = Boston)

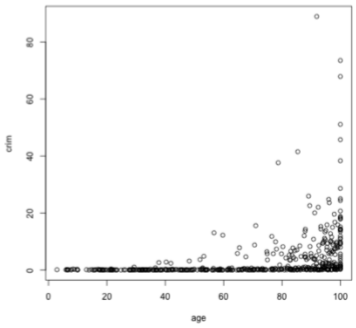


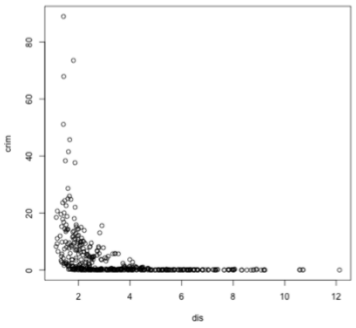


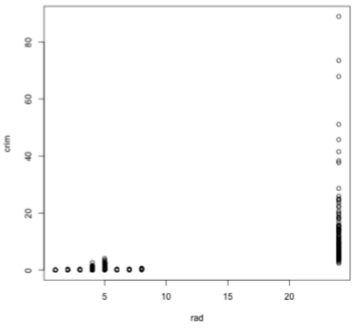


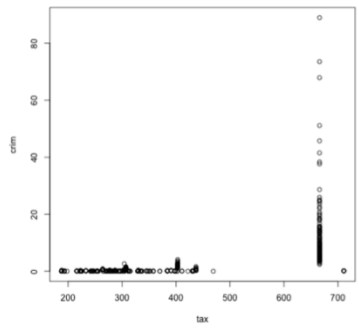


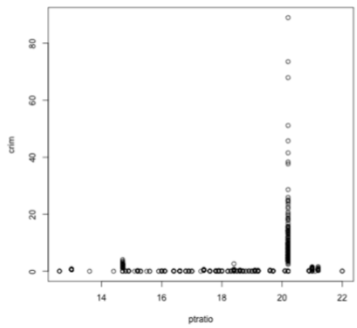


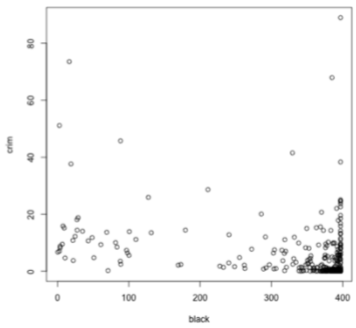


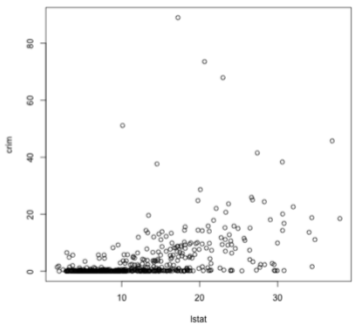


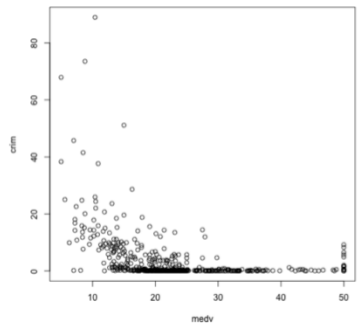












b)Fit a multiple regression model to predict the response using all of the predictors. Describe your results. For which predictors can we reject the null hypothesis H0:βj=0 ?

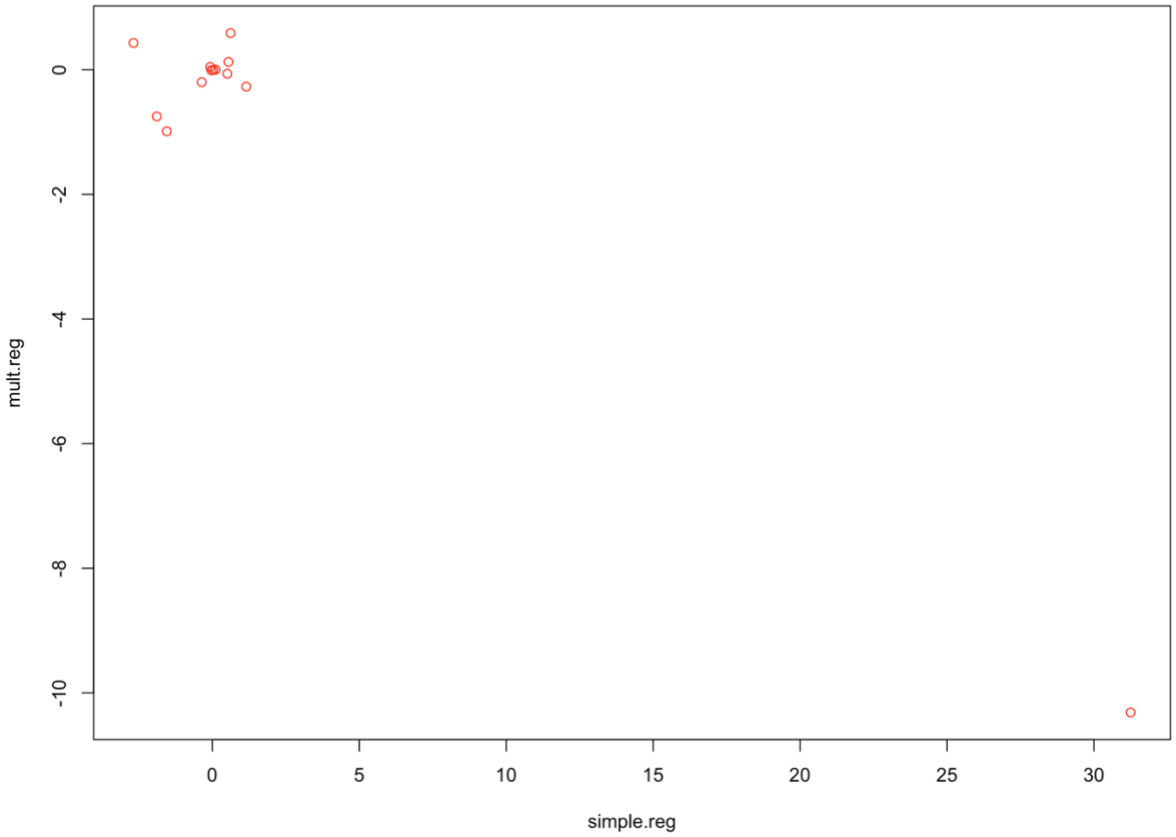
lm.all <- lm(crim ~ ., data = Boston)

summary(lm.all)

```
> summary(lm.all)

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
```

When fitting a multiple regression model, only a small number of variables are found to be statistically signficant: dis and rad at the .001 level, medv at the .01 level, and zn and black at the .05 level. For every other variable, we now #fail to reject the null hypothesis. R-squared is also much higher using a multiple regression model than any of the #predictors on their own, meaning we better explain more of the variance in the outcome.

c)How do your results from (a) compare to your results from (b) ? Create a plot displaying the univariate regression coefficients from (a) on the x-axis, and the multiple regression coefficients from (b) on the y-axis. That is, each predictor is displayed as a single point on the plot. Its coefficient in a simple linear regression model is shown on the x-axis, and its coefficient estimate in the multiple linear regression model is shown on the y-axis.



There is a difference between the simple and multiple regression coefficients. This difference is due to the fact that in the simple regression case, the slope term represents the average effect of an increase in the predictor, ignoring other predictors. In contrast, in the multiple regression case, the slope term represents the average effect of an increase in the predictor, while holding other predictors fixed. It does make sense for the multiple regression to suggest no relationship between the response and some of the predictors while the simple linear regression implies the opposite because the correlation between the predictors show some strong relationships between some of the predictors.

Examine the correlation matrix

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

So for example, when “age” is high there is a tendency in “dis” to be low, hence in simple linear regression which only examines “crim” versus “age”, we observe that higher values of “age” are associated with higher values of “crim”, even though “age” does not actually affect “crim”. So “age” is a surrogate for “dis”; “age” gets credit for the effect of “dis” on “crim”.

d)Is there evidence of non-linear association between any of the predictors and the response ? To answer this question, for each predictor X, fit a model of the form

Y=β0+β1X+β2X2+β3X3+ε.

lm.zn2 <- lm(crim ~ poly(zn, 3))

summary(lm.zn2)

```
> summary(lm.zn2)

Call:
lm(formula = crim ~ poly(zn, 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)    3.6135     0.3722   9.709  < 2e-16 ***
poly(zn, 3)1 -38.7498     8.3722  -4.628  4.7e-06 ***
poly(zn, 3)2  23.9398     8.3722   2.859  0.00442 **
poly(zn, 3)3 -10.0719     8.3722  -1.203  0.22954
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.372 on 502 degrees of freedom
Multiple R-squared:  0.05824,   Adjusted R-squared:  0.05261
F-statistic: 10.35 on 3 and 502 DF,  p-value: 1.281e-06
```

lm.indus2 <- lm(crim ~ poly(indus, 3))

summary(lm.indus2)

```
> summary(lm.indus2)

Call:
lm(formula = crim ~ poly(indus, 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)        3.614      0.330  10.950  < 2e-16 ***
poly(indus, 3)1   78.591      7.423  10.587  < 2e-16 ***
poly(indus, 3)2  -24.395      7.423  -3.286  0.00109 **
poly(indus, 3)3  -54.130      7.423  -7.292  1.2e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.423 on 502 degrees of freedom
Multiple R-squared:  0.2597,    Adjusted R-squared:  0.2552
F-statistic: 58.69 on 3 and 502 DF,  p-value: < 2.2e-16
```

lm.nox2 <- lm(crim ~ poly(nox, 3))

summary(lm.nox2)

```
> summary(lm.nox2)

Call:
lm(formula = crim ~ poly(nox, 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)     3.6135     0.3216  11.237  < 2e-16 ***
poly(nox, 3)1  81.3720     7.2336  11.249  < 2e-16 ***
poly(nox, 3)2 -28.8286     7.2336  -3.985 7.74e-05 ***
poly(nox, 3)3 -60.3619     7.2336  -8.345 6.96e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.234 on 502 degrees of freedom
Multiple R-squared:  0.297,     Adjusted R-squared:  0.2928
F-statistic: 70.69 on 3 and 502 DF,  p-value: < 2.2e-16
```

lm.rm2 <- lm(crim ~ poly(rm, 3))

summary(lm.rm2)

```
> summary(lm.rm2)

Call:
lm(formula = crim ~ poly(rm, 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)    3.6135     0.3703   9.758  < 2e-16 ***
poly(rm, 3)1 -42.3794     8.3297  -5.088 5.13e-07 ***
poly(rm, 3)2  26.5768     8.3297   3.191  0.00151 **
poly(rm, 3)3  -5.5103     8.3297  -0.662  0.50858
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.33 on 502 degrees of freedom
Multiple R-squared:  0.06779,   Adjusted R-squared:  0.06222
F-statistic: 12.17 on 3 and 502 DF,  p-value: 1.067e-07
```

lm.age2 <- lm(crim ~ poly(age, 3))

summary(lm.age2)

```
> summary(lm.age2)

Call:
lm(formula = crim ~ poly(age, 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)     3.6135     0.3485  10.368  < 2e-16 ***
poly(age, 3)1  68.1820     7.8397   8.697  < 2e-16 ***
poly(age, 3)2  37.4845     7.8397   4.781 2.29e-06 ***
poly(age, 3)3  21.3532     7.8397   2.724  0.00668 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.84 on 502 degrees of freedom
Multiple R-squared:  0.1742,    Adjusted R-squared:  0.1693
F-statistic: 35.31 on 3 and 502 DF,  p-value: < 2.2e-16
```

lm.dis2 <- lm(crim ~ poly(dis, 3))

summary(lm.dis2)

```
> summary(lm.dis2)

Call:
lm(formula = crim ~ poly(dis, 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)     3.6135     0.3259  11.087  < 2e-16 ***
poly(dis, 3)1 -73.3886     7.3315 -10.010  < 2e-16 ***
poly(dis, 3)2  56.3730     7.3315   7.689 7.87e-14 ***
poly(dis, 3)3 -42.6219     7.3315  -5.814 1.09e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.331 on 502 degrees of freedom
Multiple R-squared:  0.2778,    Adjusted R-squared:  0.2735
F-statistic: 64.37 on 3 and 502 DF,  p-value: < 2.2e-16
```

lm.rad2 <- lm(crim ~ poly(rad, 3))

summary(lm.rad2)

```
> summary(lm.rad2)

Call:
lm(formula = crim ~ poly(rad, 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)     3.6135     0.2971  12.164  < 2e-16 ***
poly(rad, 3)1 120.9074     6.6824  18.093  < 2e-16 ***
poly(rad, 3)2  17.4923     6.6824   2.618  0.00912 **
poly(rad, 3)3   4.6985     6.6824   0.703  0.48231
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.682 on 502 degrees of freedom
Multiple R-squared:    0.4,     Adjusted R-squared:  0.3965
F-statistic: 111.6 on 3 and 502 DF,  p-value: < 2.2e-16
```

lm.tax2 <- lm(crim ~ poly(tax, 3))

summary(lm.tax2)

```
> summary(lm.tax2)

Call:
lm(formula = crim ~ poly(tax, 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)     3.6135     0.3047  11.860  < 2e-16 ***
poly(tax, 3)1 112.6458     6.8537  16.436  < 2e-16 ***
poly(tax, 3)2  32.0873     6.8537   4.682 3.67e-06 ***
poly(tax, 3)3  -7.9968     6.8537  -1.167    0.244
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.854 on 502 degrees of freedom
Multiple R-squared:  0.3689,    Adjusted R-squared:  0.3651
F-statistic:  97.8 on 3 and 502 DF,  p-value: < 2.2e-16
```

lm.ptratio2 <- lm(crim ~ poly(ptratio, 3))

summary(lm.ptratio2)

```
> summary(lm.ptratio2)

Call:
lm(formula = crim ~ poly(ptratio, 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)          3.614      0.361  10.008  < 2e-16 ***
poly(ptratio, 3)1   56.045      8.122   6.901 1.57e-11 ***
poly(ptratio, 3)2   24.775      8.122   3.050  0.00241 **
poly(ptratio, 3)3  -22.280      8.122  -2.743  0.00630 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.122 on 502 degrees of freedom
Multiple R-squared:  0.1138,    Adjusted R-squared:  0.1085
F-statistic: 21.48 on 3 and 502 DF,  p-value: 4.171e-13
```

lm.black2 <- lm(crim ~ poly(black, 3))

summary(lm.black2)

```
> summary(lm.black2)

Call:
lm(formula = crim ~ poly(black, 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)       3.6135     0.3536  10.218   <2e-16 ***
poly(black, 3)1 -74.4312     7.9546  -9.357   <2e-16 ***
poly(black, 3)2   5.9264     7.9546   0.745    0.457
poly(black, 3)3  -4.8346     7.9546  -0.608    0.544
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.955 on 502 degrees of freedom
Multiple R-squared:  0.1498,    Adjusted R-squared:  0.1448
F-statistic: 29.49 on 3 and 502 DF,  p-value: < 2.2e-16
```

lm.lstat2 <- lm(crim ~ poly(lstat, 3))

summary(lm.lstat2)

```
> summary(lm.lstat2)

Call:
lm(formula = crim ~ poly(lstat, 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)       3.6135     0.3392  10.654   <2e-16 ***
poly(lstat, 3)1  88.0697     7.6294  11.543   <2e-16 ***
poly(lstat, 3)2  15.8882     7.6294   2.082   0.0378 *
poly(lstat, 3)3 -11.5740     7.6294  -1.517   0.1299
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.629 on 502 degrees of freedom
Multiple R-squared:  0.2179,    Adjusted R-squared:  0.2133
F-statistic: 46.63 on 3 and 502 DF,  p-value: < 2.2e-16
```

lm.medv2 <- lm(crim ~ poly(medv, 3))

summary(lm.medv2)

```
> summary(lm.medv2)

Call:
lm(formula = crim ~ poly(medv, 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)      3.614      0.292  12.374  < 2e-16 ***
poly(medv, 3)1 -75.058      6.569 -11.426  < 2e-16 ***
poly(medv, 3)2  88.086      6.569  13.409  < 2e-16 ***
poly(medv, 3)3 -48.033      6.569  -7.312 1.05e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.569 on 502 degrees of freedom
Multiple R-squared:  0.4202,    Adjusted R-squared:  0.4167
F-statistic: 121.3 on 3 and 502 DF,  p-value: < 2.2e-16
```

With the variables indus, nox, dis, ptracio, and medv, there is evidence of a non-linear relationship, as each of these variables squared and cubed terms is found to be statistically significant (we reject the null hypothesis that the coefficients on these exponentiated variables are zero). Age also appears to have a non-linear relationship, and once squared-age and cubed-age are brought into the model, linear age becomes statistically insignificant.

For every other variable, we do not find evidence of a non-linear relationship between the predictor and outcome variables.