Homework Machine Learning

Sushanth Ravichandran (sr56925)

2024-07-17

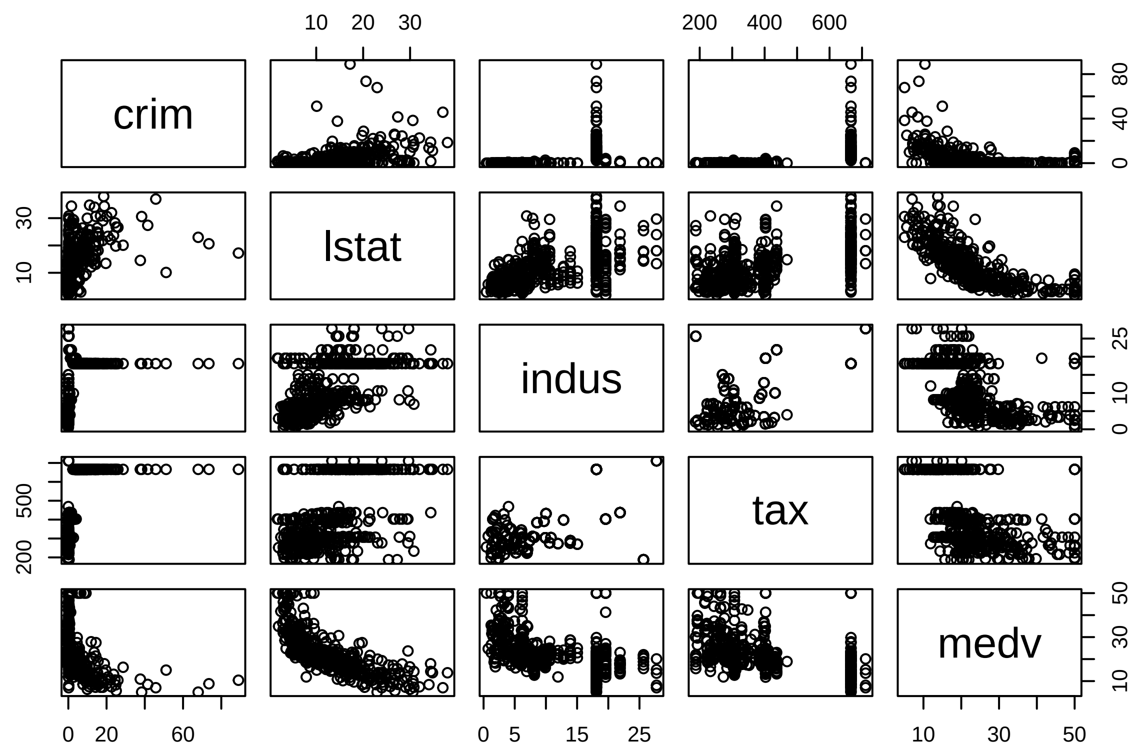
10.a This exercise involves the Boston housing data set. How many rows are in this data set? How many columns? What do the rows and columns represent?

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x dplyr::select() masks MASS::select()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
## Loading required package: lattice
##
##
## Attaching package: 'caret'
##
##
## The following object is masked from 'package:purrr':
##
##     lift
##
##
## randomForest 4.7-1.1
##
## Type rfNews() to see new features/changes/bug fixes.
##
##
## Attaching package: 'randomForest'
##
##
## The following object is masked from 'package:dplyr':
##
##     combine
##
##
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
## [1] 506  14
```

There are 506 rows of towns and 14 columns of predictors.

(b) Make some pairwise scatterplots of the predictors (columns) in this data set. Describe your findings.



The findings are as follows: • Lower status of the population percent has a negative linear relationship with median value of owner-occupied homes in $1000s. • proportion of non-retail business acres per town is inversely proportional to the median value of owner-occupied homes in $1000s. • median value of owner-occupied homes in $1000s has a positive linear relationship with lower status of the population (percent).

(c) Are any of the predictors associated with per capita crime rate? If so, explain the relationship.

```
##                 [,1]
## zn      -0.20046922
## indus    0.40658341
## chas    -0.05589158
## nox      0.42097171
## rm      -0.21924670
## age      0.35273425
## dis     -0.37967009
## rad      0.62550515
## tax      0.58276431
## ptratio  0.28994558
## black   -0.38506394
```

```
## lstat    0.45562148
## medv    -0.38830461
```

per capita crime rate by town has a negative linear relationship with medv, dis,rm, chas and black per capita crime rate by town has a strong positive linear relationship with indus, nox, rad, lstat and tax

(d) Do any of the census tracts of Boston appear to have particularly high crime rates? Tax rates? Pupil-teacher ratios? Comment on the range of each predictor.

```
## [1] "The range, mean and sd of per capita crime rate by town is:"

## [1]  0.00632 88.97620

## [1] 3.613524

## [1] 8.601545

## [1] "Range of full-value property-tax rate per $10,000."

## [1] 187 711

## [1] "The range of pupil-teacher ratio by town"

## [1] 12.6 22.0
```

The crime rate varies widely, ranging from nearly zero to 89.There are some tracts with very high crime rates due to a major deviation from the mean. The full value property tax rate ranges from 187 to 711 which shows high parity. Fourteen suburbs have a property tax rate higher than one standard deviation above the mean. The pupil-teacher ratio ranges from 12.6 to 22, indicating no suburbs with a high teacher-to-pupil ratio.

(e) How many of the census tracts in this data set bound the Charles river?

```
## [1] 35
```

35 census tracts in this data set bound the Charles river!

(f) What is the median pupil-teacher ratio among the towns in this data set?

```
## [1] 19.05
```

19.05 is the median pupil-teacher ratio

(g) Which census tract of Boston has lowest median value of owner occupied homes? What are the values of the other predictors for that census tract, and how do those values compare to the overall ranges for those predictors? Comment on your findings.

```
## [1] 399 406
```

```
##        crim zn indus chas   nox    rm age    dis rad tax ptratio  black lstat
## 399 38.3518  0  18.1    0 0.693 5.453 100 1.4896  24 666    20.2 396.90 30.59
## 406 67.9208  0  18.1    0 0.693 5.683 100 1.4254  24 666    20.2 384.97 22.98
##     medv
## 399    5
## 406    5
```

```
##          crim  zn indus chas   nox    rm   age     dis rad tax ptratio  black
## [1,]  0.00632   0  0.46    0 0.385 3.561   2.9  1.1296   1 187    12.6   0.32
## [2,] 88.97620 100 27.74    1 0.871 8.780 100.0 12.1265  24 711    22.0 396.90
##      lstat medv
## [1,]  1.73    5
## [2,] 37.97   50
```

Census tracts 399 and 406 have least median house value. There is a big difference in the criminal rates in the two suburbs Both pupil-teacher ratio and lower status of the population (percent) are close to their maximum values.

(h) In this data set, how many of the census tracts average more than seven rooms per dwelling? More than eight rooms per dwelling? Comment on the census tracts that average more than eight rooms per dwelling.

```
## [1] 64
```

```
## [1] 13
```

64 census tracts average more than 7 rooms per dwelling The number of census tracts with avg. no. of rooms >8 is 13. There are a lot of census tracts that average at ~8 rooms per dwelling due to the big delta between rm (>7 and >8) values.

```
## [1] "Summary of Boston dataset"
```

```
##       crim                 zn             indus            chas
##  Min.   : 0.00632   Min.   :  0.00   Min.   : 0.46   Min.   :0.00000
##  1st Qu.: 0.08205   1st Qu.:  0.00   1st Qu.: 5.19   1st Qu.:0.00000
##  Median : 0.25651   Median :  0.00   Median : 9.69   Median :0.00000
##  Mean   : 3.61352   Mean   : 11.36   Mean   :11.14   Mean   :0.06917
##  3rd Qu.: 3.67708   3rd Qu.: 12.50   3rd Qu.:18.10   3rd Qu.:0.00000
##  Max.   :88.97620   Max.   :100.00   Max.   :27.74   Max.   :1.00000
##       nox               rm             age              dis
##  Min.   :0.3850   Min.   :3.561   Min.   :  2.90   Min.   : 1.130
##  1st Qu.:0.4490   1st Qu.:5.886   1st Qu.: 45.02   1st Qu.: 2.100
##  Median :0.5380   Median :6.208   Median : 77.50   Median : 3.207
##  Mean   :0.5547   Mean   :6.285   Mean   : 68.57   Mean   : 3.795
##  3rd Qu.:0.6240   3rd Qu.:6.623   3rd Qu.: 94.08   3rd Qu.: 5.188
##  Max.   :0.8710   Max.   :8.780   Max.   :100.00   Max.   :12.127
##       rad              tax           ptratio          black
##  Min.   : 1.000   Min.   :187.0   Min.   :12.60   Min.   :  0.32
##  1st Qu.: 4.000   1st Qu.:279.0   1st Qu.:17.40   1st Qu.:375.38
##  Median : 5.000   Median :330.0   Median :19.05   Median :391.44
##  Mean   : 9.549   Mean   :408.2   Mean   :18.46   Mean   :356.67
##  3rd Qu.:24.000   3rd Qu.:666.0   3rd Qu.:20.20   3rd Qu.:396.23
```

```
## Max.   :24.000   Max.   :711.0   Max.   :22.00   Max.   :396.90
##      lstat            medv
## Min.   : 1.73   Min.   : 5.00
## 1st Qu.: 6.95   1st Qu.:17.02
## Median :11.36   Median :21.20
## Mean   :12.65   Mean   :22.53
## 3rd Qu.:16.95   3rd Qu.:25.00
## Max.   :37.97   Max.   :50.00
```

```
## [1] "Summary of Boston dataset with avg no. of rooms per dwelling> 8"
```

```
##       crim               zn            indus             chas
## Min.   :0.02009   Min.   : 0.00   Min.   : 2.680   Min.   :0.0000
## 1st Qu.:0.33147   1st Qu.: 0.00   1st Qu.: 3.970   1st Qu.:0.0000
## Median :0.52014   Median : 0.00   Median : 6.200   Median :0.0000
## Mean   :0.71879   Mean   :13.62   Mean   : 7.078   Mean   :0.1538
## 3rd Qu.:0.57834   3rd Qu.:20.00   3rd Qu.: 6.200   3rd Qu.:0.0000
## Max.   :3.47428   Max.   :95.00   Max.   :19.580   Max.   :1.0000
##       nox               rm             age              dis
## Min.   :0.4161   Min.   :8.034   Min.   : 8.40   Min.   :1.801
## 1st Qu.:0.5040   1st Qu.:8.247   1st Qu.:70.40   1st Qu.:2.288
## Median :0.5070   Median :8.297   Median :78.30   Median :2.894
## Mean   :0.5392   Mean   :8.349   Mean   :71.54   Mean   :3.430
## 3rd Qu.:0.6050   3rd Qu.:8.398   3rd Qu.:86.50   3rd Qu.:3.652
## Max.   :0.7180   Max.   :8.780   Max.   :93.90   Max.   :8.907
##       rad              tax           ptratio          black
## Min.   : 2.000   Min.   :224.0   Min.   :13.00   Min.   :354.6
## 1st Qu.: 5.000   1st Qu.:264.0   1st Qu.:14.70   1st Qu.:384.5
## Median : 7.000   Median :307.0   Median :17.40   Median :386.9
## Mean   : 7.462   Mean   :325.1   Mean   :16.36   Mean   :385.2
## 3rd Qu.: 8.000   3rd Qu.:307.0   3rd Qu.:17.40   3rd Qu.:389.7
## Max.   :24.000   Max.   :666.0   Max.   :20.20   Max.   :396.9
##      lstat           medv
## Min.   :2.47   Min.   :21.9
## 1st Qu.:3.32   1st Qu.:41.7
## Median :4.14   Median :48.3
## Mean   :4.31   Mean   :44.2
## 3rd Qu.:5.12   3rd Qu.:50.0
## Max.   :7.44   Max.   :50.0
```

The crime rate mean is at 0.7 in tracts where (rm>8) compared to the overall mean of 3.6 which shows there is is a lesser rate of crimes in these dwellings

The median house value is almost 2X in the (rm>8) compared to the overall dataset which shows that there are more expensive houses in these areas

The lower status of the population (percent) is much lower in the tracts with avg. no. of rooms >8 which shows the proportion of rich in these tracts is higher

2.)

Question 3 15. 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.

```
##       crim                zn             indus       chas         nox
##  Min.   : 0.00632   Min.   :  0.00   Min.   : 0.46   N:471   Min.   :0.3850
##  1st Qu.: 0.08205   1st Qu.:  0.00   1st Qu.: 5.19   Y: 35   1st Qu.:0.4490
##  Median : 0.25651   Median :  0.00   Median : 9.69           Median :0.5380
##  Mean   : 3.61352   Mean   : 11.36   Mean   :11.14           Mean   :0.5547
##  3rd Qu.: 3.67708   3rd Qu.: 12.50   3rd Qu.:18.10           3rd Qu.:0.6240
##  Max.   :88.97620   Max.   :100.00   Max.   :27.74           Max.   :0.8710
##        rm             age              dis              rad
##  Min.   :3.561   Min.   :  2.90   Min.   : 1.130   Min.   : 1.000
##  1st Qu.:5.886   1st Qu.: 45.02   1st Qu.: 2.100   1st Qu.: 4.000
##  Median :6.208   Median : 77.50   Median : 3.207   Median : 5.000
##  Mean   :6.285   Mean   : 68.57   Mean   : 3.795   Mean   : 9.549
##  3rd Qu.:6.623   3rd Qu.: 94.08   3rd Qu.: 5.188   3rd Qu.:24.000
##  Max.   :8.780   Max.   :100.00   Max.   :12.127   Max.   :24.000
##       tax          ptratio          black            lstat
##  Min.   :187.0   Min.   :12.60   Min.   :  0.32   Min.   : 1.73
##  1st Qu.:279.0   1st Qu.:17.40   1st Qu.:375.38   1st Qu.: 6.95
##  Median :330.0   Median :19.05   Median :391.44   Median :11.36
##  Mean   :408.2   Mean   :18.46   Mean   :356.67   Mean   :12.65
##  3rd Qu.:666.0   3rd Qu.:20.20   3rd Qu.:396.23   3rd Qu.:16.95
##  Max.   :711.0   Max.   :22.00   Max.   :396.90   Max.   :37.97
##       medv
##  Min.   : 5.00
##  1st Qu.:17.02
##  Median :21.20
##  Mean   :22.53
##  3rd Qu.:25.00
##  Max.   :50.00
```

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

```
##
```

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

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

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

```
## F-statistic: 25.45 on 1 and 504 DF,  p-value: 6.347e-07
```

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

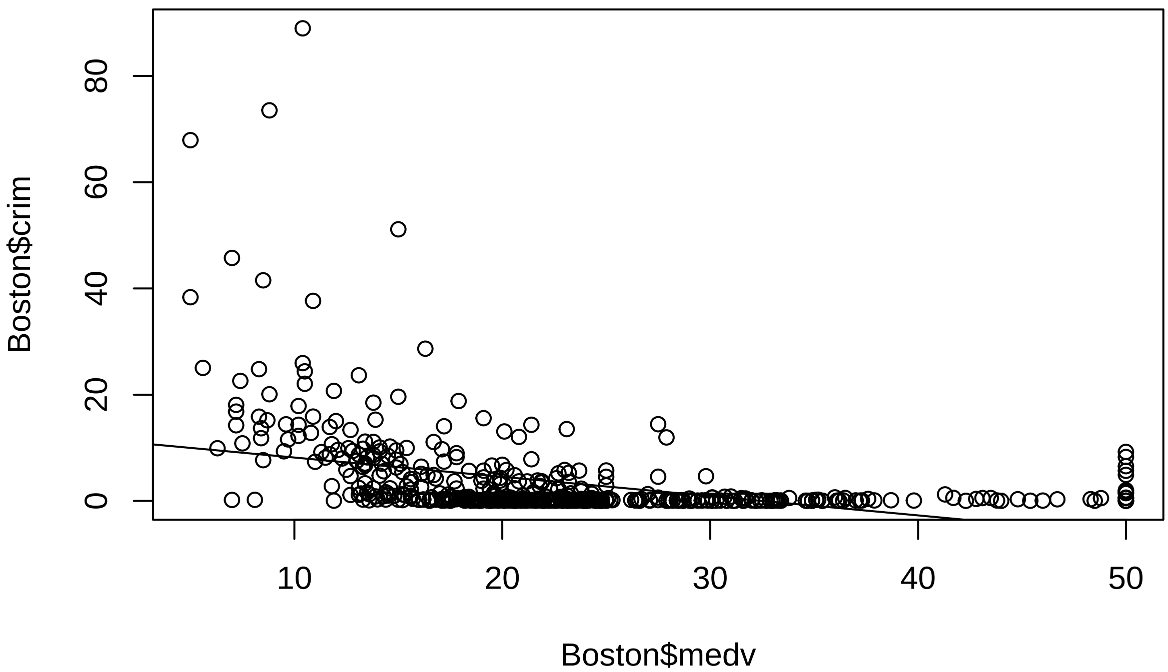
```
##
## Call:
## lm(formula = crim ~ rad, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.164  -1.381  -0.141   0.660  76.433
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.28716    0.44348  -5.157 3.61e-07 ***
## rad          0.61791    0.03433  17.998  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.718 on 504 degrees of freedom
## Multiple R-squared:  0.3913, Adjusted R-squared:   0.39
## F-statistic: 323.9 on 1 and 504 DF,  p-value: < 2.2e-16
```

```
##
## Call:
## lm(formula = crim ~ black, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.756  -2.299  -2.095  -1.296  86.822
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 16.553529   1.425903  11.609   <2e-16 ***
## black       -0.036280   0.003873  -9.367   <2e-16 ***
## ---
```

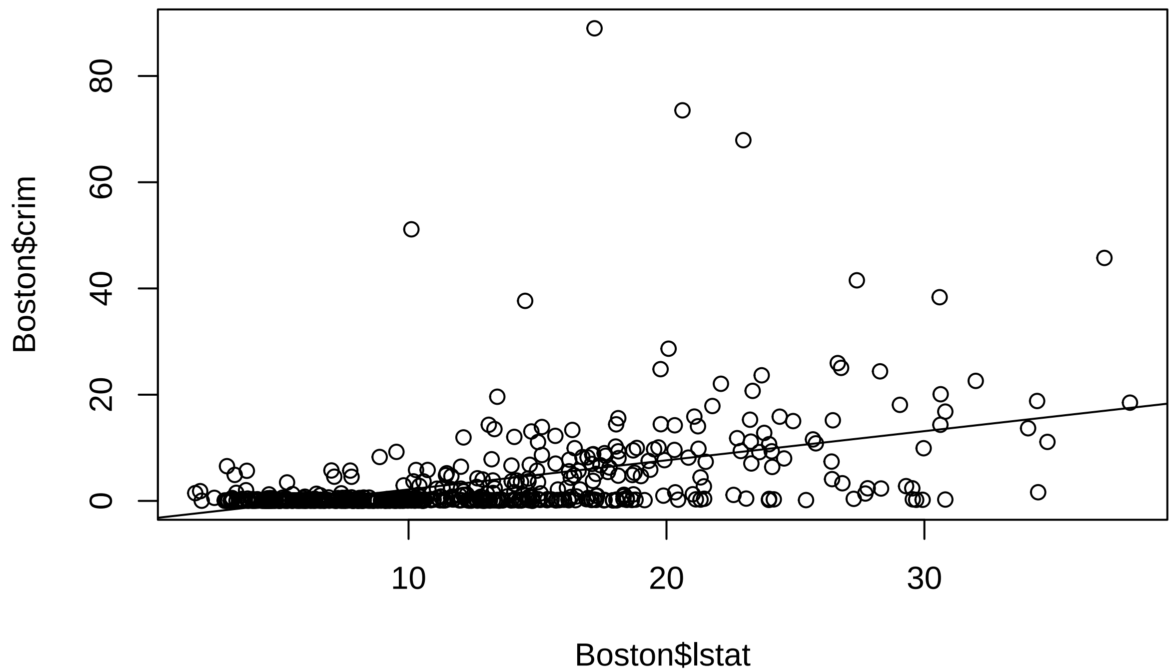
```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.946 on 504 degrees of freedom
## Multiple R-squared:  0.1483, Adjusted R-squared:  0.1466
## F-statistic: 87.74 on 1 and 504 DF,  p-value: < 2.2e-16
```

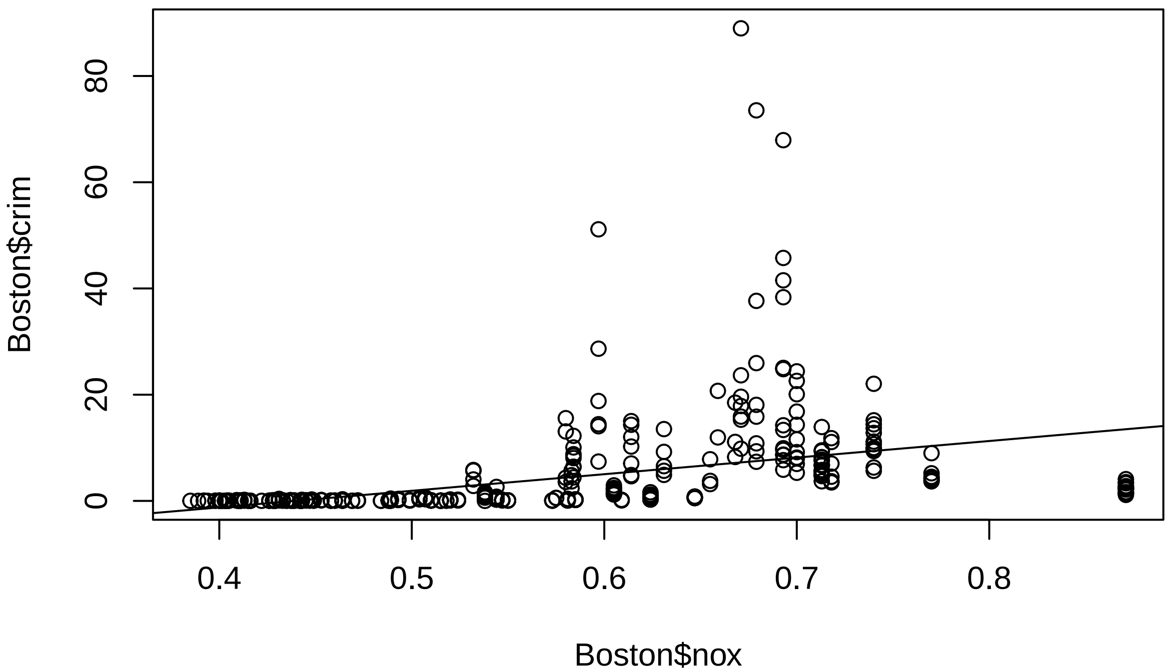
```
##
## Call:
## lm(formula = crim ~ lstat, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.925  -2.822  -0.664   1.079  82.862
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.33054    0.69376  -4.801 2.09e-06 ***
## lstat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

```
##
## Call:
## lm(formula = crim ~ medv, data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.071 -4.022 -2.343  1.298 80.957
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 11.79654    0.93419   12.63   <2e-16 ***
## medv        -0.36316    0.03839   -9.46   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.934 on 504 degrees of freedom
## Multiple R-squared:  0.1508, Adjusted R-squared:  0.1491
## F-statistic: 89.49 on 1 and 504 DF,  p-value: < 2.2e-16
```



Boston$crim
0
20
40
60
80
10
20
30
40
50
Boston$medv


Boston$crim
0
20
40
60
80
10
20
30
Boston$lstat



The linear regressions of per capita crime rate by town has been drawn against all other predictorss to find any linear relationship between these predictors

Based on the results, except for chas(tracts that bound the Charles river) all other predictors variables have an effect on predicting the per capita crime rate by town.

Criminal rate is inversely proportional to the median value of owner-occupied homes Criminal rate is directly proportional to lower status of the population (percent) These two are indicators that there are higher chances of crime in poorer neighbourhoods.

(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 : j = 0

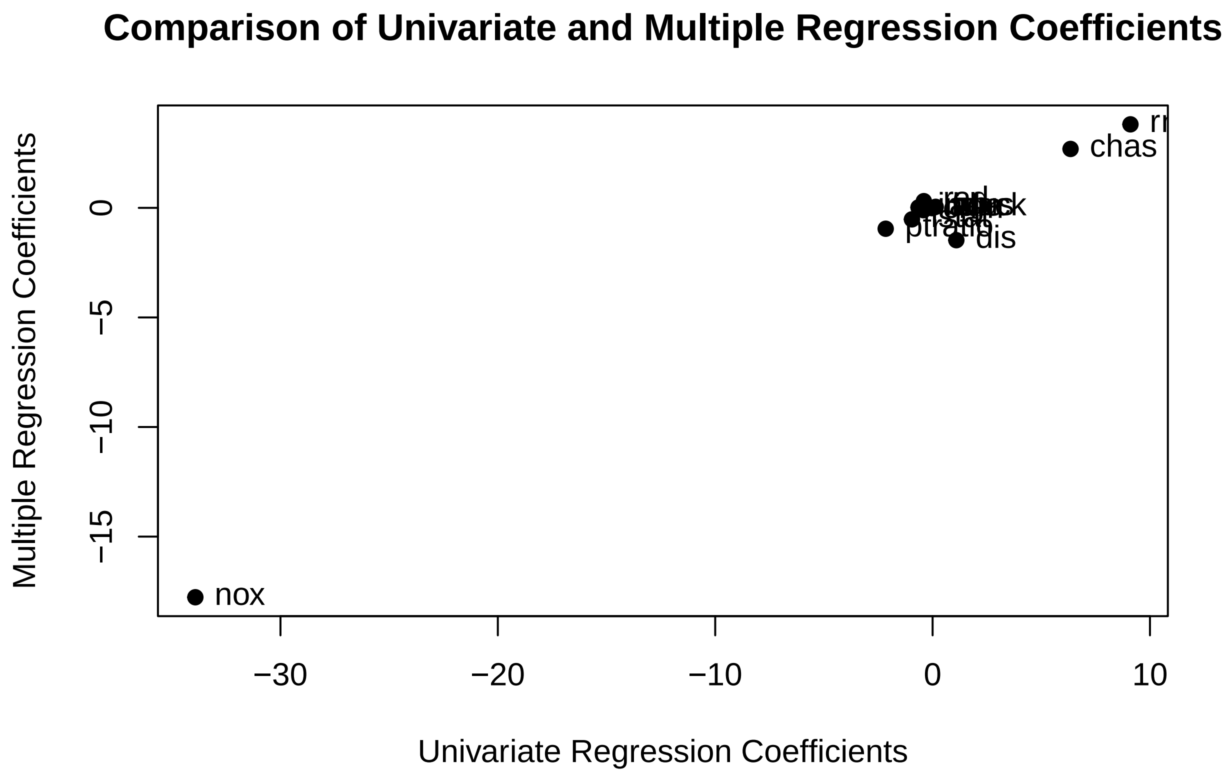
```
##
## Call:
## lm(formula = crim ~ ., data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.924 -2.120 -0.353  1.019 75.051
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  17.033228   7.234903   2.354 0.018949 *
## zn            0.044855   0.018734   2.394 0.017025 *
## indus        -0.063855   0.083407  -0.766 0.444294
## chasY        -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
```

dis, rad, medv, black & zn have the least p value (<0.05) as a result of which null hypothesis can be rejected (Rejecting the null hypothesis implies that these independent variables significantly contributes to the model in explaining the variance in the dependent variable(criminal rate)).

The results are different from simple regression since other variables are ignored in simple linear regression.

(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 in 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



The coefficients have changed while performing multiple linear regression as compared to single linear regression which shows that there is some form of pair wise interaction comes into play during multiple linear regression which affects the coeff.

(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 = 0 + 1X + 2X2 + 3X3 + .

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

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

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

```
##
## Call:
## lm(formula = crim ~ poly(rm, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.485  -3.468  -2.221  -0.015  87.219
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.6135     0.3703   9.758  < 2e-16 ***
## poly(rm, 3)1 -42.3794     8.3297  -5.088 5.13e-07 ***
```

```
## poly(rm, 3)2  26.5768     8.3297   3.191  0.00151 **
## poly(rm, 3)3  -5.5103     8.3297  -0.662  0.50858
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.33 on 502 degrees of freedom
## Multiple R-squared:  0.06779,    Adjusted R-squared:  0.06222
## F-statistic: 12.17 on 3 and 502 DF,  p-value: 1.067e-07
```

```
##
## Call:
## lm(formula = crim ~ poly(age, 3), data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.762 -2.673 -0.516  0.019 82.842
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)     3.6135     0.3485  10.368  < 2e-16 ***
## poly(age, 3)1  68.1820     7.8397   8.697  < 2e-16 ***
## poly(age, 3)2  37.4845     7.8397   4.781 2.29e-06 ***
## poly(age, 3)3  21.3532     7.8397   2.724  0.00668 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.84 on 502 degrees of freedom
## Multiple R-squared:  0.1742, Adjusted R-squared:  0.1693
## F-statistic: 35.31 on 3 and 502 DF,  p-value: < 2.2e-16
```

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

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

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

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

```
##
## Call:
## lm(formula = crim ~ poly(ptratio, 3), data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.833 -4.146 -1.655  1.408 82.697
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.614      0.361  10.008  < 2e-16 ***
## poly(ptratio, 3)1   56.045      8.122   6.901 1.57e-11 ***
## poly(ptratio, 3)2   24.775      8.122   3.050  0.00241 **
## poly(ptratio, 3)3  -22.280      8.122  -2.743  0.00630 **
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.122 on 502 degrees of freedom
## Multiple R-squared:  0.1138, Adjusted R-squared:  0.1085
## F-statistic: 21.48 on 3 and 502 DF,  p-value: 4.171e-13
```

```
##
## Call:
## lm(formula = crim ~ poly(black, 3), data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.096  -2.343  -2.128  -1.439  86.790
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)       3.6135     0.3536  10.218   <2e-16 ***
## poly(black, 3)1 -74.4312     7.9546  -9.357   <2e-16 ***
## poly(black, 3)2   5.9264     7.9546   0.745    0.457
## poly(black, 3)3  -4.8346     7.9546  -0.608    0.544
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.955 on 502 degrees of freedom
## Multiple R-squared:  0.1498, Adjusted R-squared:  0.1448
## F-statistic: 29.49 on 3 and 502 DF,  p-value: < 2.2e-16
```

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

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

```
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -24.427  -1.976  -0.437   0.439  73.655 
## 
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)    
## (Intercept)       3.614      0.292  12.374  < 2e-16 ***
## poly(medv, 3)1  -75.058      6.569 -11.426  < 2e-16 ***
## poly(medv, 3)2   88.086      6.569  13.409  < 2e-16 ***
## poly(medv, 3)3  -48.033      6.569  -7.312 1.05e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 6.569 on 502 degrees of freedom
## Multiple R-squared:  0.4202, Adjusted R-squared:  0.4167 
## F-statistic: 121.3 on 3 and 502 DF,  p-value: < 2.2e-16
```

Most predictors show evidence of a non-linear relationship (either quadratic or cubic) with the response variable, except for black (Bk, the proportion of blacks by town), which only shows a strong linear relationship.

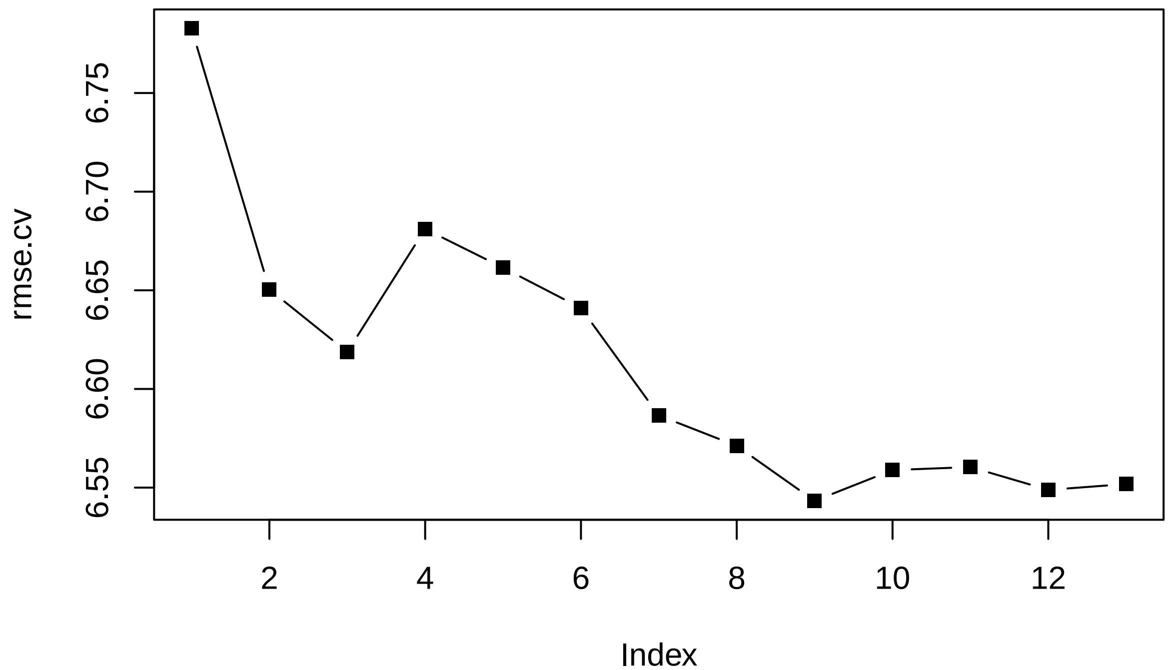
The squared term (degree 2) is significant for most predictors, including zn, indus, nox, rm, age, dis, rad, tax, ptratio, lstat, and medv, suggesting a non-linear (parabolic) relationship for these variables.

For predictors such as indus, nox, age, dis, tax, ptratio, and medv, the third-order polynomial terms are significant, indicating a more complex non-linear relationship with the response variable (crime rate).

3. Shrinkage and selection in linear models: Chapter 6: #11
4. We will now try to predict per capita crime rate in the Boston data set.

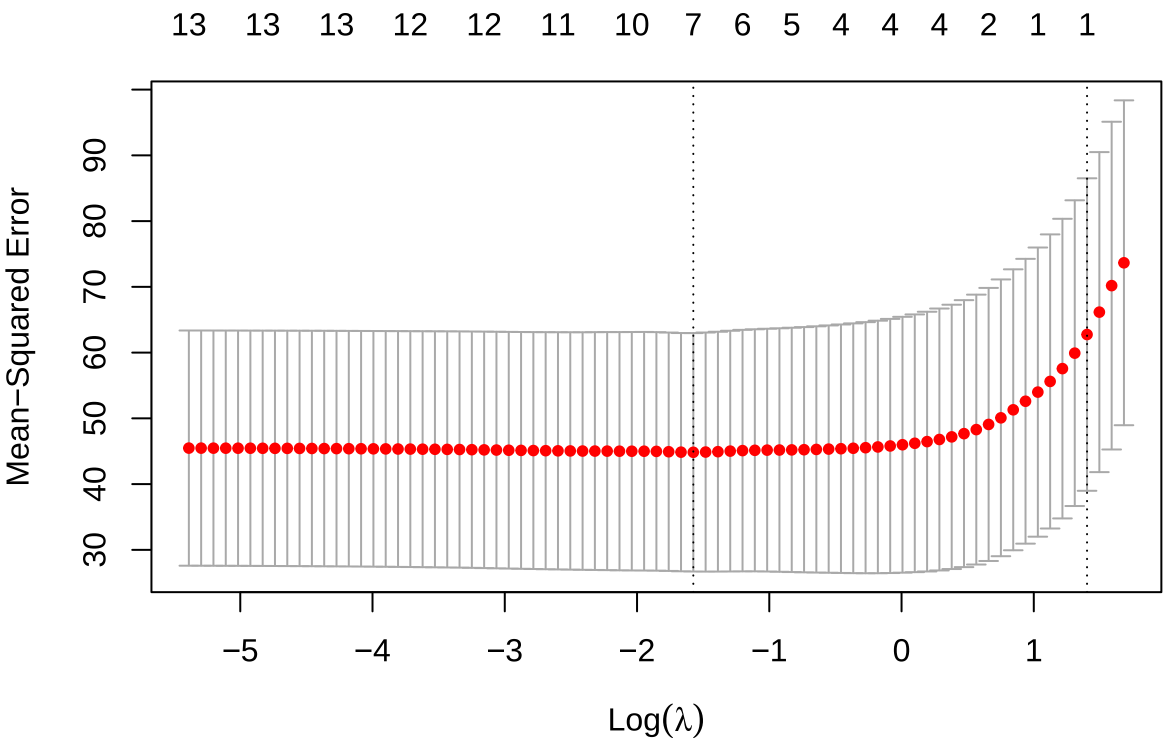
(a) Try out some of the regression methods explored in this chapter, such as best subset selection, the lasso, ridge regression, and PCR. Present and discuss results for the approaches that you consider.

```
## Loading required package: Matrix

## 
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':
## 
##     expand, pack, unpack

## Loaded glmnet 4.1-8
```



Best subset selection

```
## [1] 6.543281
```

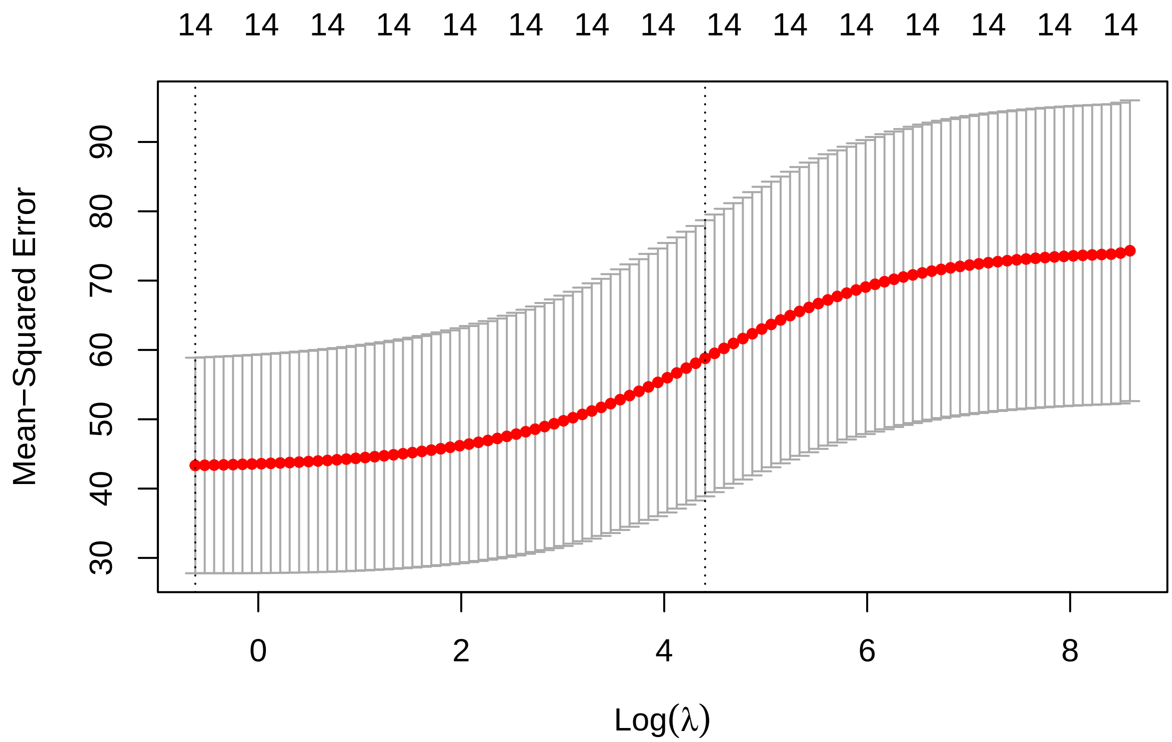


Lasso

```
## [1] "The RMSE is "
```

```
## [1] 7.921353
```

Ridge Regression




```
## 15 x 1 sparse Matrix of class "dgCMatrix"
##                       s1
## (Intercept)  1.378868104
## zn          -0.002955708
## indus        0.029308357
## chasN        0.152157898
## chasY       -0.152154852
## nox          1.877361697
## rm          -0.142466331
## age          0.006217963
## dis         -0.094695187
## rad          0.045930738
## tax          0.002085959
## ptratio      0.071079829
## black       -0.002603532
## lstat        0.035722766
## medv        -0.023418669
```

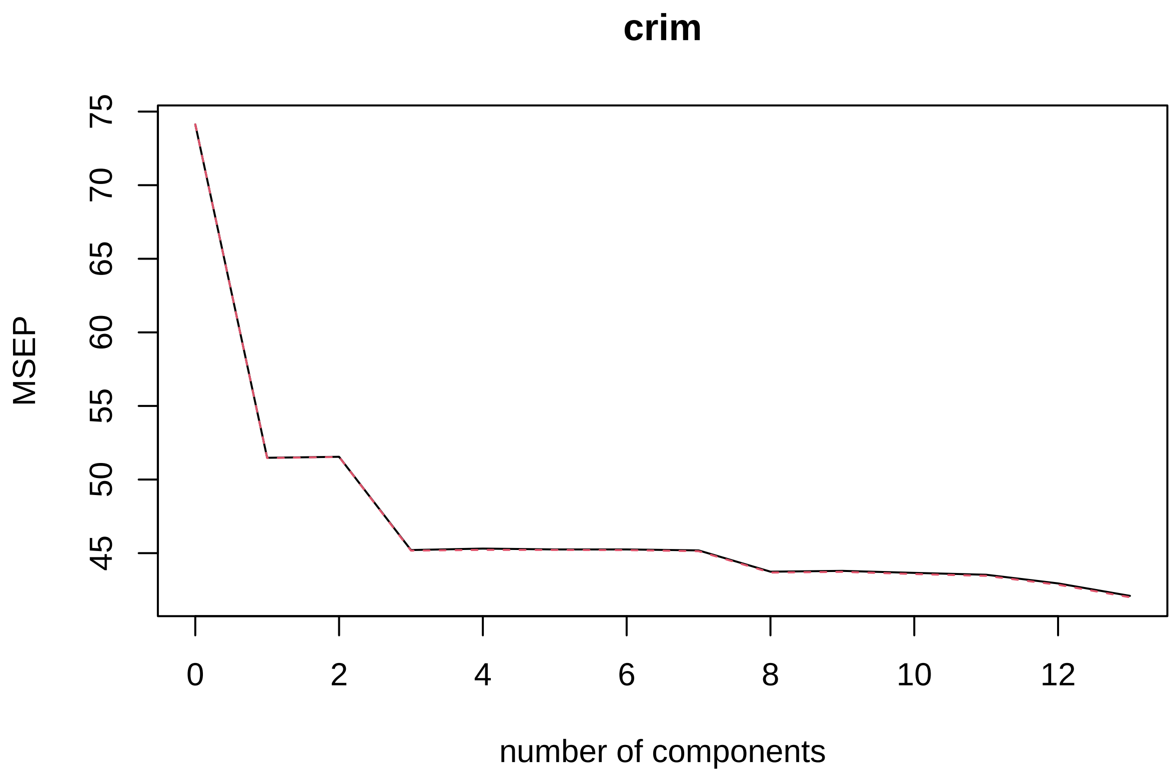
```
## [1] "The RMSE is "
```

```
## [1] 7.667762
```

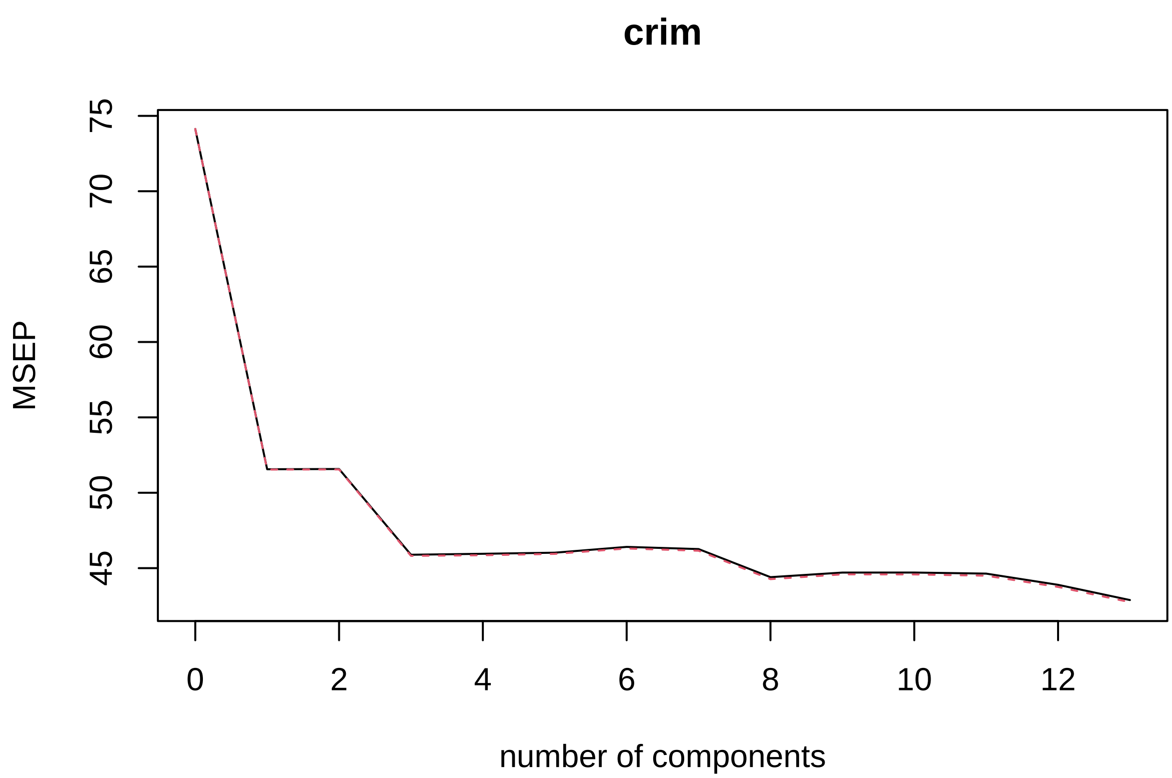
PCR

```
##
## Attaching package: 'pls'
```

```
## The following object is masked from 'package:caret':
##
##     R2

## The following object is masked from 'package:stats':
##
##     loadings

## Data:    X dimension: 506 13
##  Y dimension: 506 1
## Fit method: svdpc
## Number of components considered: 13
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## CV            8.61    7.175    7.180    6.724    6.731    6.727    6.727
## adjCV         8.61    7.174    7.179    6.721    6.725    6.724    6.724
##        7 comps  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## CV       6.722    6.614    6.618     6.607     6.598     6.553     6.488
## adjCV    6.718    6.609    6.613     6.602     6.592     6.546     6.481
##
## TRAINING: % variance explained
##       1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps
## X       47.70    60.36    69.67    76.45    82.99    88.00    91.14    93.45
## crim    30.69    30.87    39.27    39.61    39.61    39.86    40.14    42.47
##       9 comps  10 comps  11 comps  12 comps  13 comps
## X       95.40     97.04     98.46     99.52     100.0
## crim    42.55     42.78     43.04     44.13      45.4
```



crim
MSEP
75
70
65
60
55
50
45
0
2
4
6
8
10
12
number of components



Propose a model (or set of models) that seem to perform well on this data set, and justify your answer. Make sure that you are evaluating model performance using validation set error, cross validation, or some other reasonable alternative, as opposed to using training error.

(c) Does your chosen model involve all of the features in the data set? Why or why not?

b)Both the ridge and lasso model provide comparable RMSE results. The ridge model is better when we want to retian all predictors but in this case from our subset selection we have found that 9 predictors are optimal. Therefore Lasso regression is preferable when you believe that only a subset of predictors is relevant and you want to perform variable selection. It can produce simpler and more interpretable models by setting some coefficients to zero.

Interpretability: Lasso models are typically more interpretable compared to other methods because they perform variable selection, resulting in a simpler, more understandable model.

c.)No the chosen Lasso model results in a sparse model, including only 9 variables. This means that it excludes less influential variables, which simplifies the model and focuses on the most significant predictors.

Regression Trees: Chapter 8: #8 BUT: Use the Austin housing data posted to the course website (austin-housing.csv) instead of the dataset in the book. Use the following variables to generate predictions for log(latestPrice): latitude, longitude, hasAssociation, livingAreaSqFt, numOfBathrooms, numOfBedrooms. (See the description of the dataset in the individual prediciton project assignment.) When reporting your prediction errors, report them in terms of prices (not log prices).

(a) Split the data set into a training set and a test set.

```
##         streetAddress zipcode
```

```
## 1    14004 Chisos Trl   78717
## 2 14405 Laurinburg Dr   78717
## 3    14702 Menifee St   78725
## 4     15207 Lucian St   78725
## 5   12525 Verandah Ct   78726
## 6   12512 Verandah Ct   78726
##
## 1
## 2
## 3
## 4
## 5 Welcome to the Estates of Grandview Hills.  This elegant custom home is located on a cul-de-sac lo
## 6
##   latitude longitude garageSpaces hasAssociation hasGarage hasSpa hasView
## 1 30.49564 -97.79787            0           TRUE     FALSE  FALSE   FALSE
## 2 30.48878 -97.79490            2           TRUE      TRUE  FALSE   FALSE
## 3 30.23315 -97.58732            2          FALSE      TRUE  FALSE   FALSE
## 4 30.23824 -97.57833            2           TRUE      TRUE  FALSE   FALSE
## 5 30.42646 -97.85929            2           TRUE      TRUE  FALSE    TRUE
## 6 30.42596 -97.85841            0           TRUE     FALSE  FALSE   FALSE
##        homeType yearBuilt latestPrice latest_saledate latest_salemonth
## 1 Single Family      2008       400.0       1/10/2020                1
## 2 Single Family      2013       549.9       3/13/2018                3
## 3 Single Family      1999       240.0      12/31/2020               12
## 4 Single Family      2012       200.0       1/30/2018                1
## 5 Single Family      2004       875.0       11/9/2020               11
## 6 Single Family      2005       830.0       9/17/2019                9
##   latest_saleyear numOfPhotos numOfAccessibilityFeatures numOfAppliances
## 1            2020          20                          0               3
## 2            2018          69                          0               4
## 3            2020          10                          0               4
## 4            2018          33                          0               5
## 5            2020          38                          0               8
## 6            2019          37                          0               4
##   numOfParkingFeatures numOfPatioAndPorchFeatures numOfSecurityFeatures
## 1                    2                          0                     0
## 2                    3                          0                     0
## 3                    2                          2                     0
## 4                    2                          0                     0
## 5                    2                          4                     1
## 6                    1                          1                     3
##   numOfWaterfrontFeatures numOfWindowFeatures numOfCommunityFeatures
## 1                       0                   0                      0
## 2                       0                   0                      0
## 3                       0                   0                      0
## 4                       0                   0                      0
## 5                       0                   0                      0
## 6                       0                   1                      0
##   lotSizeSqFt livingAreaSqFt avgSchoolDistance avgSchoolRating avgSchoolSize
## 1      7666.0           2228          1.900000        8.333333          1481
## 2      8494.0           3494          3.300000        7.666667          1259
## 3      5183.0           1534          1.800000        3.000000          1457
## 4      8145.0           1652          1.966667        3.000000          1457
## 5     30056.4           3402          2.066667        7.000000          1277
```

```
## 6     19166.4            3573          2.000000         7.000000           1277
##   MedianStudentsPerTeacher numOfBathrooms numOfBedrooms numOfStories
## 1                       16              2             3            1
## 2                       14              5             4            2
## 3                       13              3             3            1
## 4                       13              2             3            1
## 5                       16              4             4            2
## 6                       16              5             4            2
```

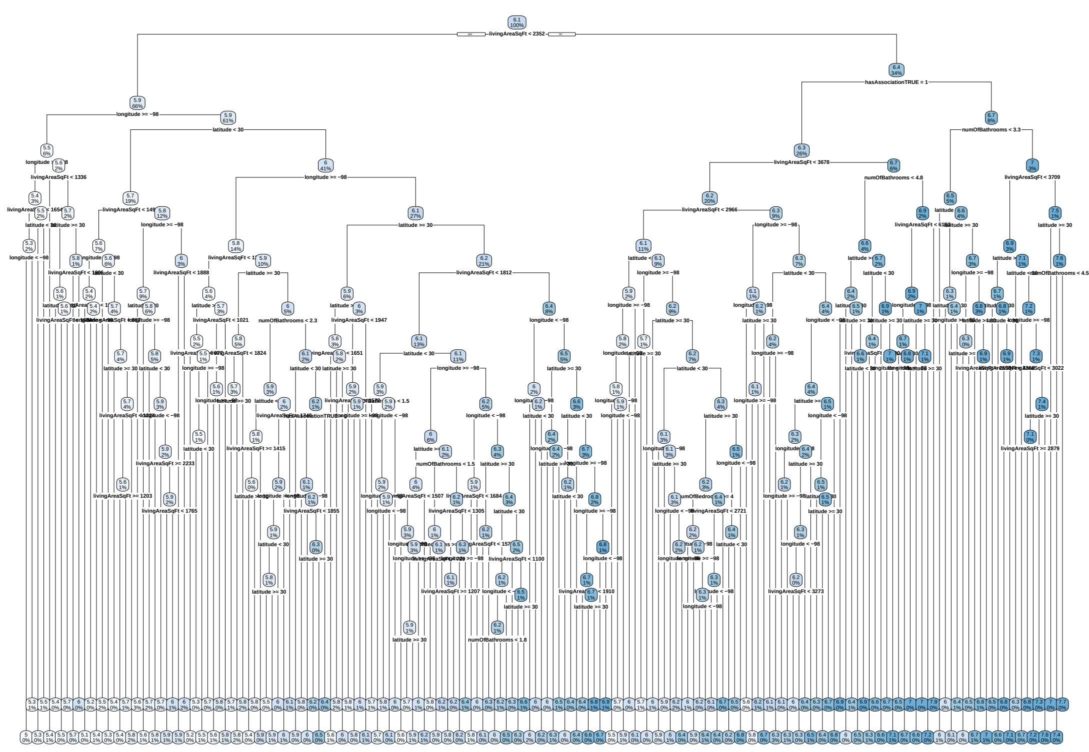
```
## [1] "The Dimensions of the train and test set are:"
```

```
## [1] 5429   35
```

```
## [1] 1355   35
```

Fit a regression tree to the training set. Plot the tree, and interpret the results. What test MSE do you obtain?

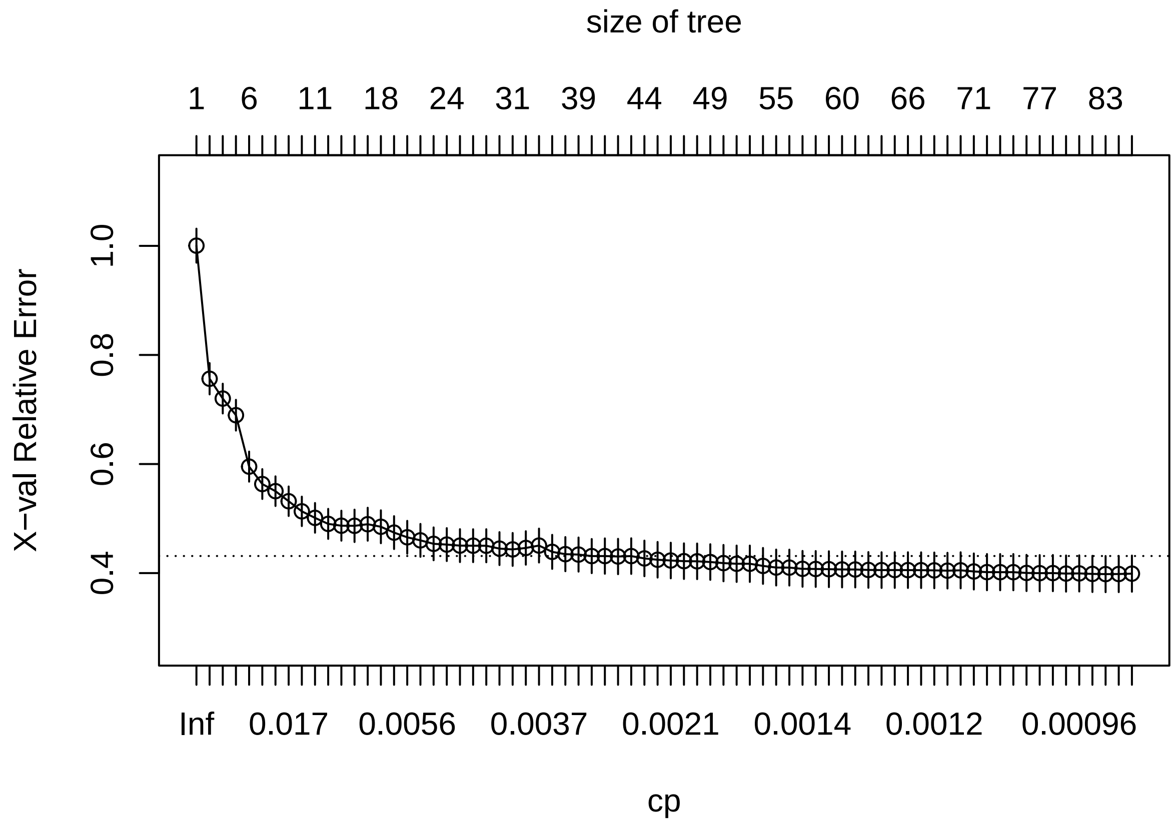
```
## Warning: labs do not fit even at cex 0.15, there may be some overplotting
```



```
## [1] "Test MSE is "
```

```
## [1] 71950.34
```

c.) Use cross-validation in order to determine the optimal level of tree complexity. Does pruning the tree im-



prove the test MSE?

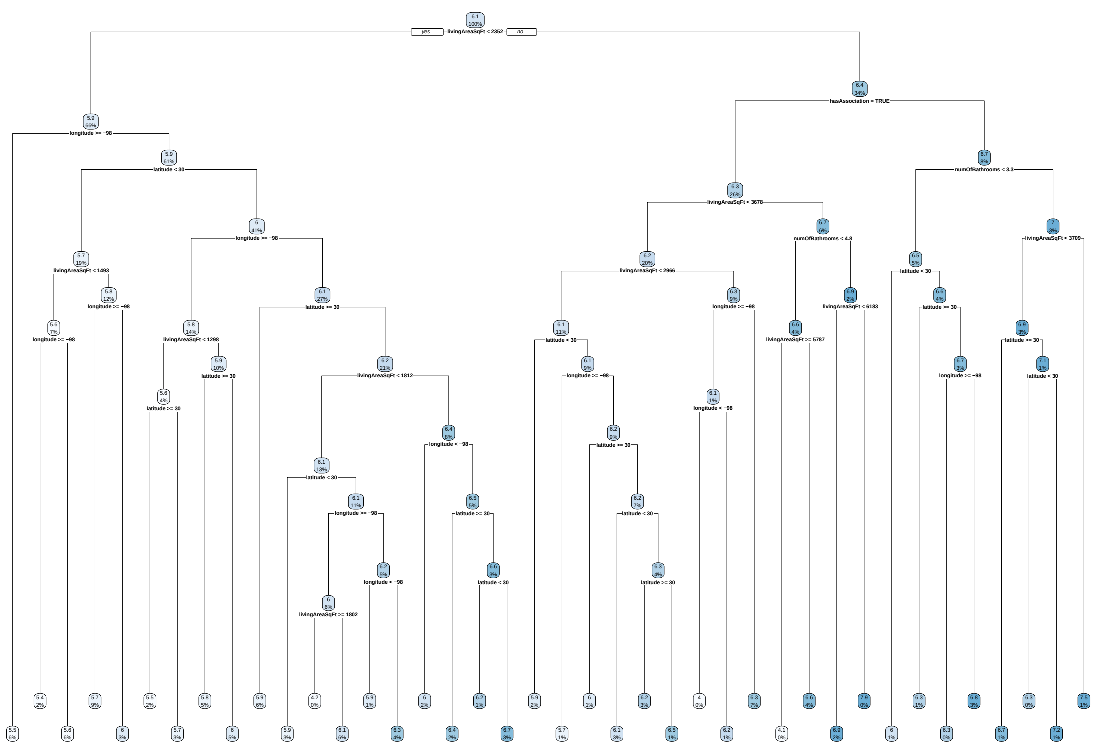
```
##
## Regression tree:
## rpart(formula = log_latestPrice ~ latitude + longitude + hasAssociation +
##     livingAreaSqFt + numOfBathrooms + numOfBedrooms, data = austin_train,
##     control = rpart.control(cp = 9e-04, minsplit = 5))
##
## Variables actually used in tree construction:
## [1] hasAssociation latitude       livingAreaSqFt longitude      numOfBathrooms
## [6] numOfBedrooms
##
## Root node error: 1395.8/5429 = 0.2571
##
## n= 5429
##
##            CP nsplit rel error  xerror     xstd
## 1  0.24643805      0   1.00000 1.00042 0.031187
## 2  0.04813819      1   0.75356 0.75636 0.028749
## 3  0.04265406      2   0.70542 0.72001 0.027326
## 4  0.03966120      3   0.66277 0.68952 0.028213
## 5  0.03378678      5   0.58345 0.59524 0.027610
## 6  0.01976545      6   0.54966 0.56330 0.027359
## 7  0.01764142      7   0.52990 0.55023 0.027188
## 8  0.01578030      8   0.51225 0.53178 0.027061
## 9  0.01456032      9   0.49647 0.51322 0.026998
## 10 0.00909126     10   0.48191 0.50128 0.027254
```

```
## 11 0.00785784      11   0.47282 0.49024 0.027528
## 12 0.00785161      13   0.45711 0.48691 0.027495
## 13 0.00710738      14   0.44925 0.48668 0.029667
## 14 0.00679441      16   0.43504 0.48962 0.030230
## 15 0.00654939      17   0.42825 0.48496 0.030191
## 16 0.00566256      18   0.42170 0.47423 0.030141
## 17 0.00560957      19   0.41603 0.46566 0.030157
## 18 0.00528276      20   0.41042 0.46002 0.030152
## 19 0.00485966      22   0.39986 0.45365 0.030132
## 20 0.00464212      23   0.39500 0.45228 0.030186
## 21 0.00462132      25   0.38571 0.45037 0.030170
## 22 0.00455603      26   0.38109 0.45024 0.030170
## 23 0.00414398      27   0.37654 0.45014 0.030319
## 24 0.00413909      29   0.36825 0.44489 0.030188
## 25 0.00391998      30   0.36411 0.44339 0.030180
## 26 0.00373403      32   0.35627 0.44608 0.030697
## 27 0.00361912      34   0.34880 0.45049 0.031075
## 28 0.00296798      35   0.34518 0.43893 0.031211
## 29 0.00289144      37   0.33925 0.43472 0.031302
## 30 0.00287754      38   0.33636 0.43398 0.031293
## 31 0.00275234      39   0.33348 0.43120 0.031281
## 32 0.00263643      40   0.33073 0.43122 0.032416
## 33 0.00260840      41   0.32809 0.43043 0.032402
## 34 0.00228177      42   0.32548 0.43117 0.032653
## 35 0.00221751      43   0.32320 0.42690 0.032662
## 36 0.00209047      44   0.32098 0.42445 0.032634
## 37 0.00204887      45   0.31889 0.42306 0.032679
## 38 0.00199048      46   0.31684 0.42190 0.032673
## 39 0.00197945      47   0.31485 0.42159 0.032677
## 40 0.00191796      48   0.31287 0.42028 0.032671
## 41 0.00189622      50   0.30904 0.41829 0.033345
## 42 0.00189181      51   0.30714 0.41713 0.033334
## 43 0.00184893      52   0.30525 0.41713 0.033334
## 44 0.00152097      53   0.30340 0.41314 0.032936
## 45 0.00150955      54   0.30188 0.41020 0.032939
## 46 0.00150032      55   0.30037 0.41007 0.032938
## 47 0.00139954      56   0.29887 0.40778 0.032849
## 48 0.00137916      57   0.29747 0.40752 0.032887
## 49 0.00137261      58   0.29609 0.40713 0.032887
## 50 0.00136648      59   0.29472 0.40666 0.032889
## 51 0.00134595      60   0.29335 0.40661 0.032889
## 52 0.00128533      62   0.29066 0.40578 0.032725
## 53 0.00127137      63   0.28937 0.40547 0.032734
## 54 0.00126195      64   0.28810 0.40557 0.032734
## 55 0.00125558      65   0.28684 0.40547 0.032734
## 56 0.00123892      66   0.28558 0.40533 0.032734
## 57 0.00122852      67   0.28435 0.40500 0.032735
## 58 0.00122099      68   0.28312 0.40448 0.032733
## 59 0.00120431      69   0.28190 0.40509 0.033085
## 60 0.00111147      70   0.28069 0.40299 0.033055
## 61 0.00106199      71   0.27958 0.40175 0.033193
## 62 0.00106187      73   0.27746 0.40169 0.033194
## 63 0.00105752      74   0.27639 0.40169 0.033194
## 64 0.00102038      75   0.27534 0.40026 0.033185
```

```
## 65 0.00099786     76   0.27432 0.39984 0.033188
## 66 0.00098874     77   0.27332 0.40002 0.033192
## 67 0.00096379     78   0.27233 0.39914 0.033169
## 68 0.00095072     79   0.27137 0.39942 0.033177
## 69 0.00094082     81   0.26946 0.39850 0.033170
## 70 0.00093276     82   0.26852 0.39817 0.033170
## 71 0.00090248     83   0.26759 0.39829 0.033169
## 72 0.00090000     84   0.26669 0.39898 0.033181
```

```
## [1] 0.3981728
```

```
##           CP       nsplit    rel error       xerror         xstd
##  0.002752337 39.000000000  0.333477897  0.431198440  0.031280920
```



```
## [1] "Test MSE with pruned tree is "
```

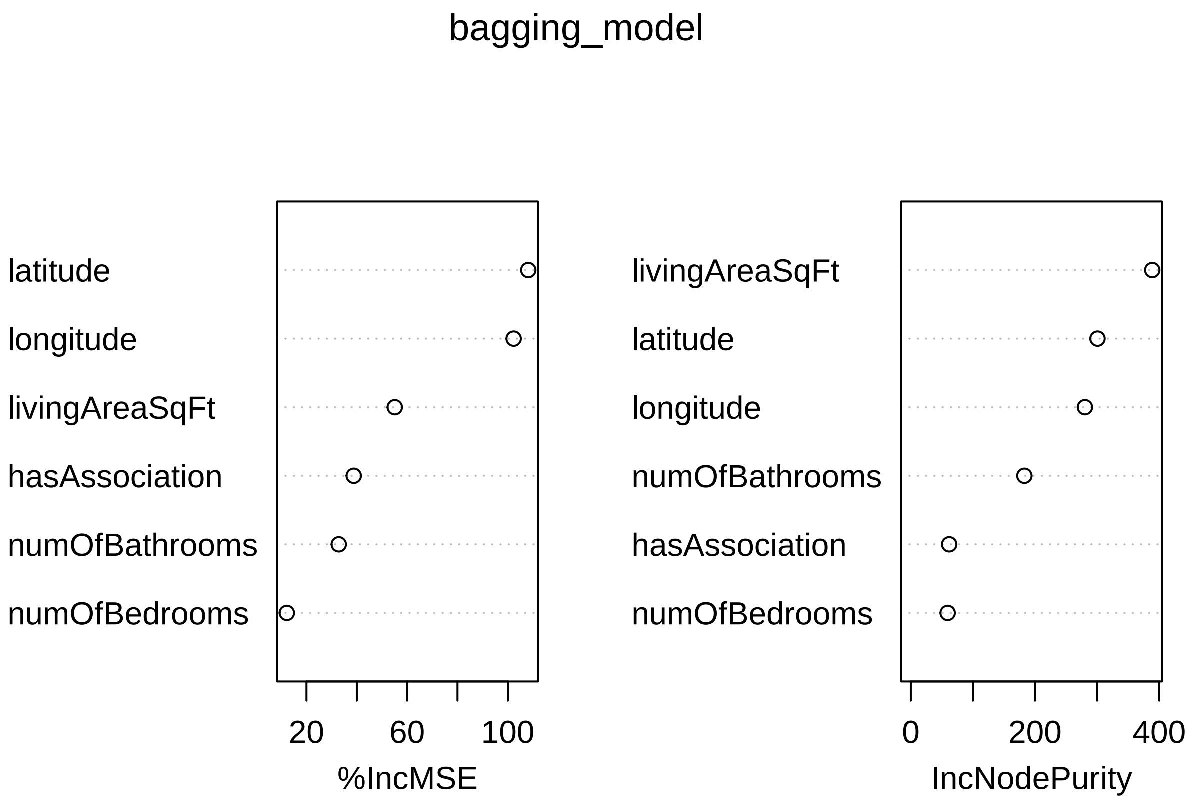
```
## [1] 76968.75
```

It does not seem like pruning the tree helps reduce the RMSE

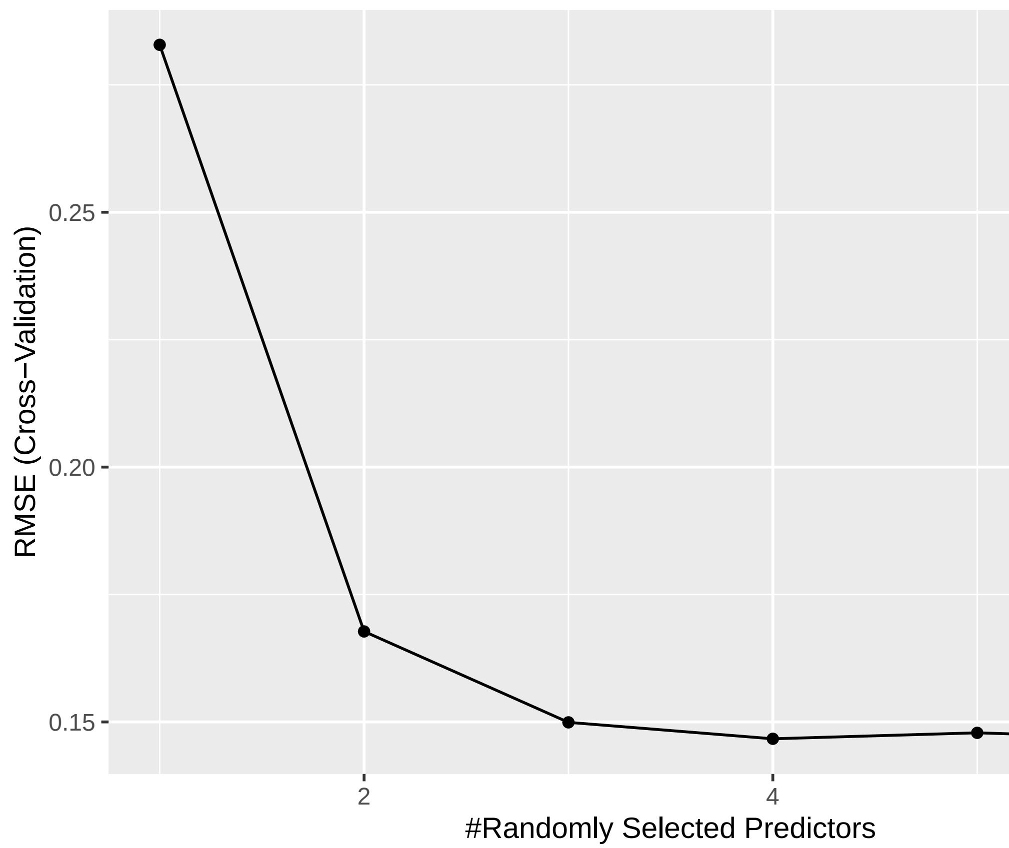
d.) Use the bagging approach in order to analyze this data. What test MSE do you obtain? Use the importance() function to determine which variables are most important

```
## Test MSE (Bagging): 72690.95
```

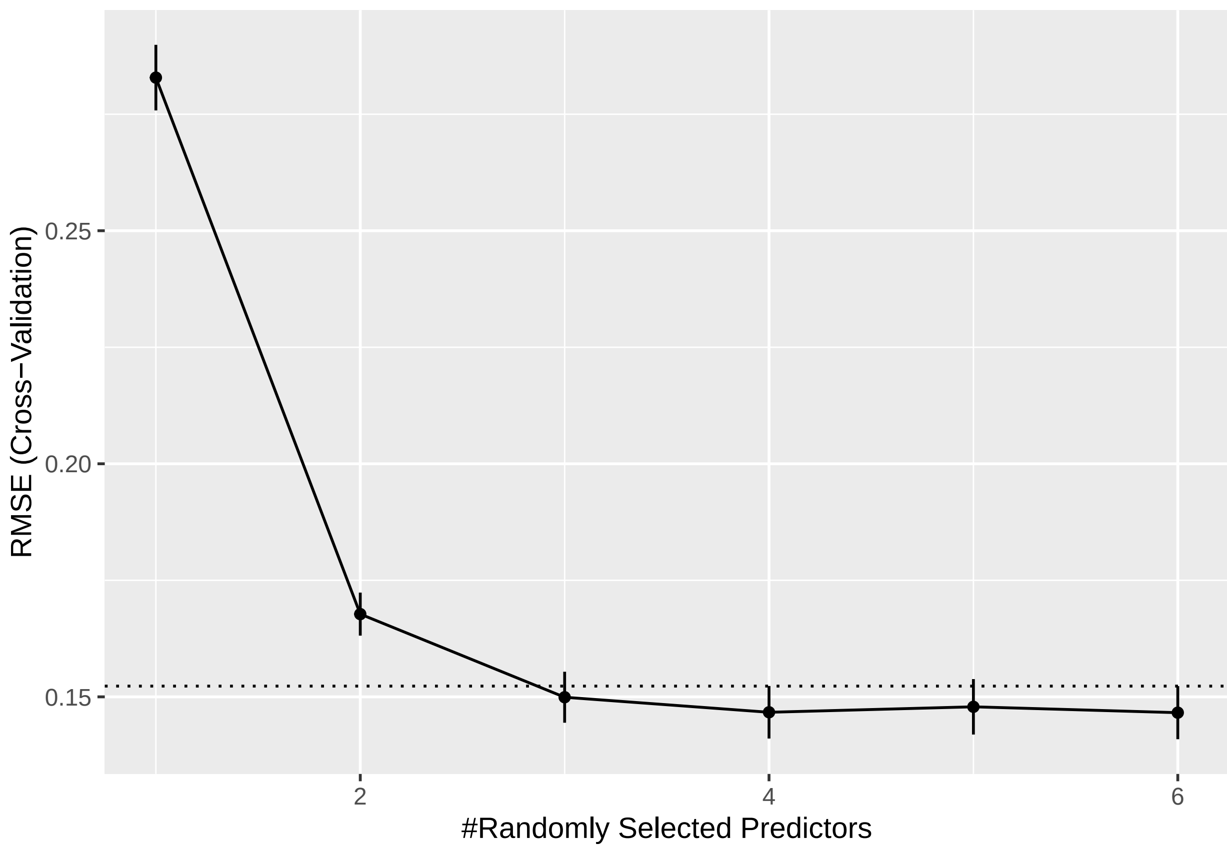
```
##                    %IncMSE IncNodePurity
## latitude        108.12231     300.60600
## longitude       102.31258     280.30849
## hasAssociation   38.78526      61.79132
## livingAreaSqFt   55.07664     388.77845
## numOfBathrooms   32.83544     182.85117
## numOfBedrooms    12.19858      59.33172
```



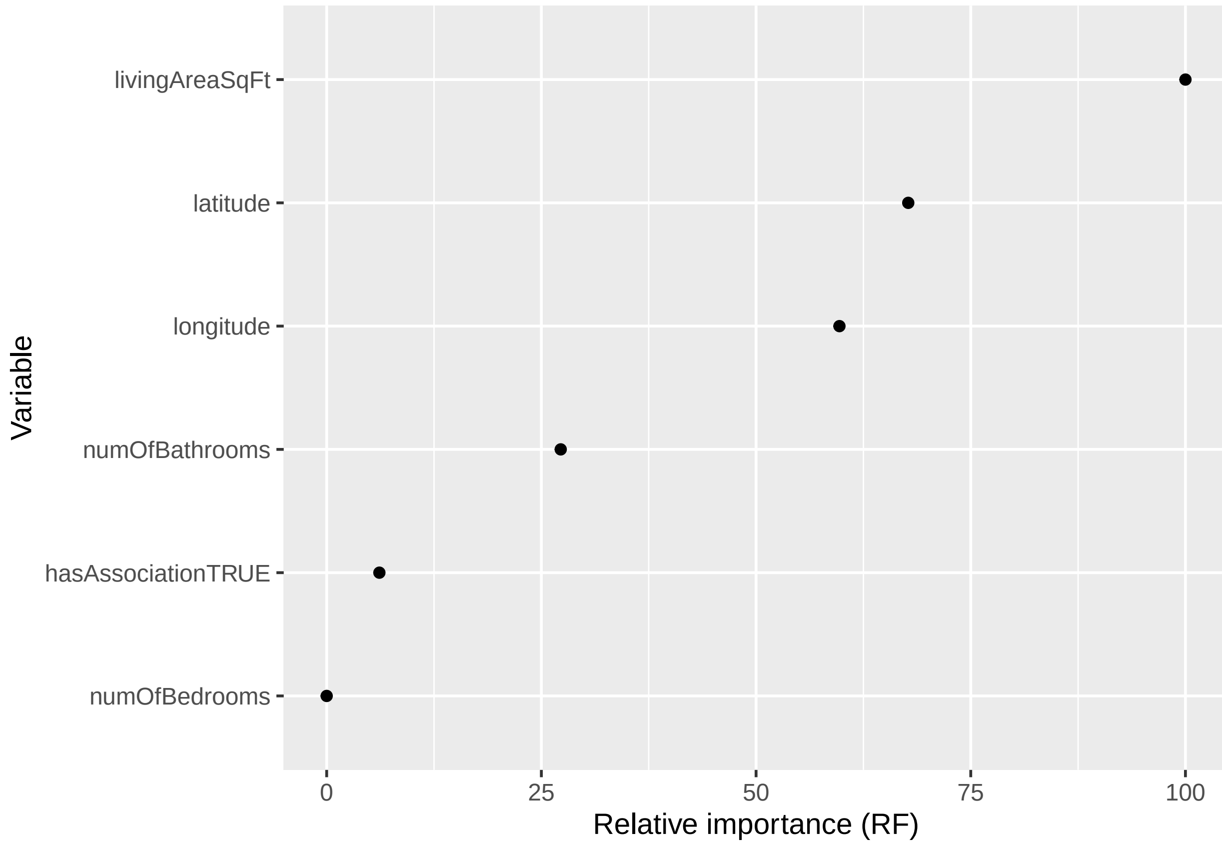
e.) Use random forests to analyze this data. What test MSE do you obtain? Use the importance() function to determine which variables are most important. Describe the effect of m, the number of variables considered at



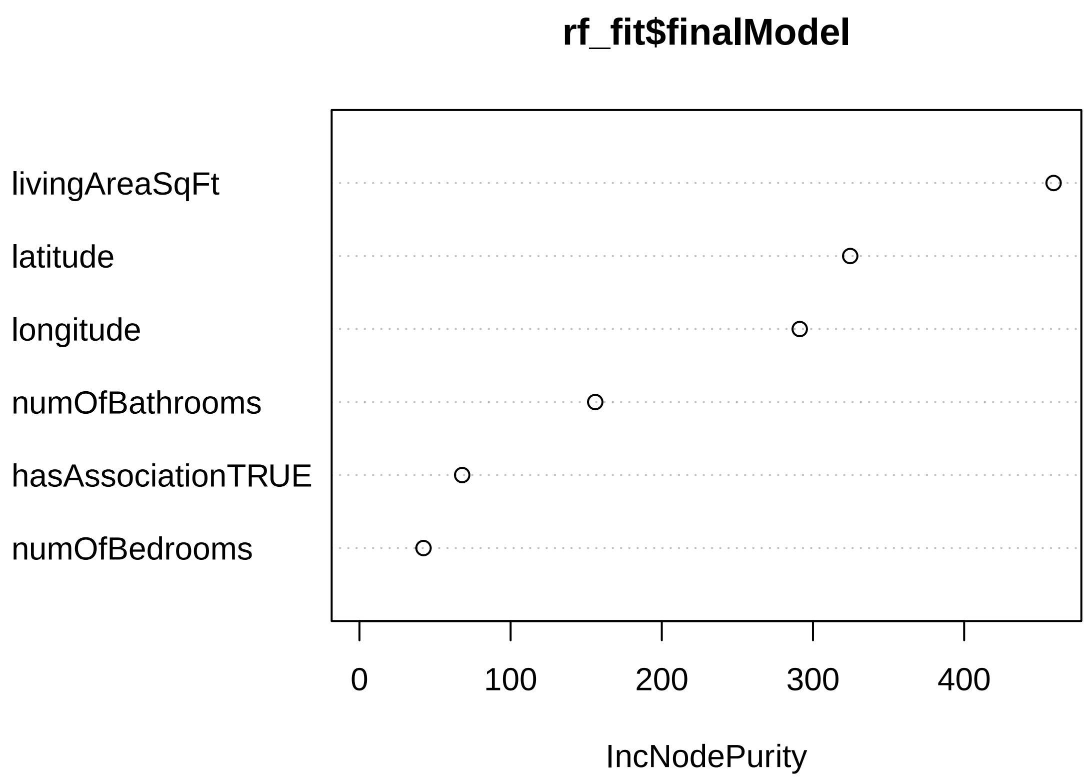
each split, on the error rate obtained.



RMSE (Cross−Validation)
0.25
0.20
0.15
2
4
6
#Randomly Selected Predictors




```
##                       Overall
## latitude           324.62735
## longitude          291.23186
## hasAssociationTRUE  67.95376
## livingAreaSqFt     459.15280
## numOfBathrooms     155.97131
## numOfBedrooms       42.37727
```




```
## [1] "Test MSE with the best RRandom Forest model is "
```

```
## [1] 72478.74
```

The OOB error rate is least at mtry =6 and therefore that is our ideal mtry value. If we take the one SE rule into consideration then the mtry=3 will be picked.

(f) Now analyze the data using BART, and report your results.

```
## Loading required package: nlme
```

```
##
## Attaching package: 'nlme'
```

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

```
## Loading required package: survival
```

```
##
## Attaching package: 'survival'
```

```
## The following object is masked from 'package:caret':
##
##     cluster
```

```
## *****Calling gbart: type=1
## *****Data:
## data:n,p,np: 5429, 7, 1355
## y1,yn: -0.062941, 0.719818
## x1,x[n*p]: 30.495638, 3.000000
## xp1,xp[np*p]: 30.426456, 2.000000
## *****Number of Trees: 200
## *****Number of Cut Points: 100 ... 7
## *****burn,nd,thin: 100,1000,1
## *****Prior:beta,alpha,tau,nu,lambda,offset: 2,0.95,0.115489,3,0.0274104,6.05441
## *****sigma: 0.375122
## *****w (weights): 1.000000 ... 1.000000
## *****Dirichlet:sparse,theta,omega,a,b,rho,augment: 0,0,1,0.5,1,7,0
## *****printevery: 100
##
## MCMC
## done 0 (out of 1100)
## done 100 (out of 1100)
## done 200 (out of 1100)
## done 300 (out of 1100)
## done 400 (out of 1100)
## done 500 (out of 1100)
## done 600 (out of 1100)
## done 700 (out of 1100)
## done 800 (out of 1100)
## done 900 (out of 1100)
## done 1000 (out of 1100)
## time: 30s
## trcnt,tecnt: 1000,1000
```

```
## Test MSE (BART): 67619.29
```

```
##                 Length  Class     Mode
## sigma              1100 -none-    numeric
## yhat.train      5429000 -none-    numeric
## yhat.test       1355000 -none-    numeric
## varcount           7000 -none-    numeric
## varprob            7000 -none-    numeric
## treedraws             2 -none-    list
## proc.time             5 proc_time numeric
## hostname              1 -none-    logical
## yhat.train.mean    5429 -none-    numeric
## sigma.mean            1 -none-    numeric
## LPML                  1 -none-    numeric
## yhat.test.mean     1355 -none-    numeric
## ndpost                1 -none-    numeric
## offset                1 -none-    numeric
## varcount.mean         7 -none-    numeric
## varprob.mean          7 -none-    numeric
## rm.const              7 -none-    numeric
```

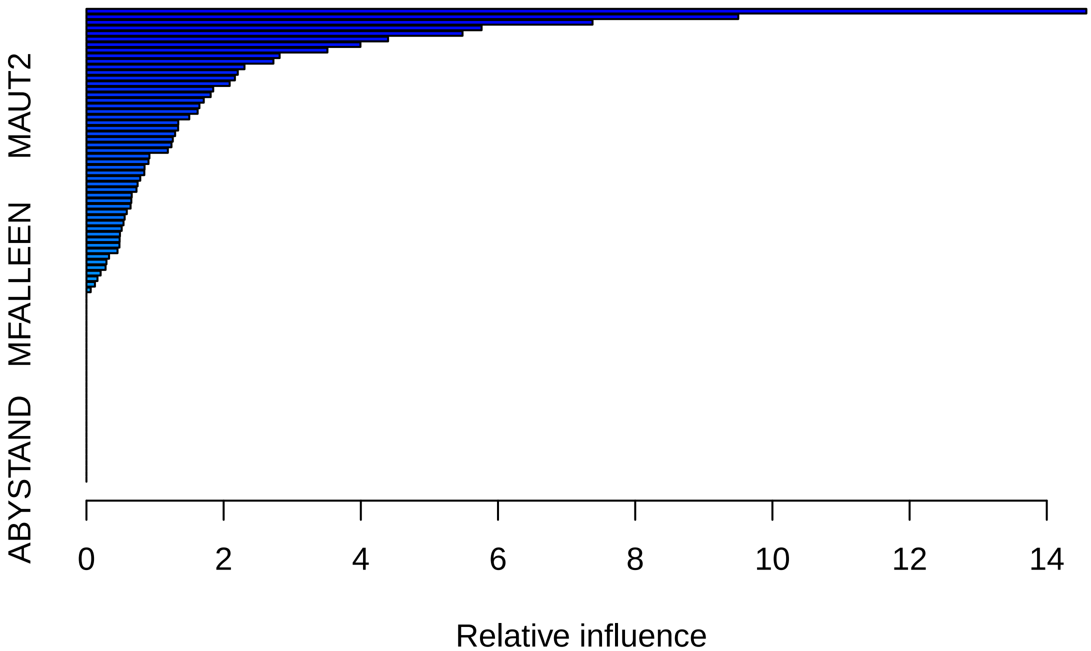
The least MSE is obtained using the BART model!

5. Classification (Trees and Logistic regression): Chapter 8: #11; in part c) use logistic regression. This question uses the Caravan data set.

(a) Create a training set consisting of the first 1,000 observations, and a test set consisting of the remaining observations

(b)Fit a boosting model to the training set with Purchase as the response and the other variables as predictors. Use 1,000 trees, and a shrinkage value of 0.01. Which predictors appear to be the most important?

```
## Loaded gbm 2.2.2

## This version of gbm is no longer under development. Consider transitioning to gbm3, https://github.co

## [1] "The dimensions of training and test set are as follows:"

## [1] 1000   86

## [1] 4822   86

## Warning in gbm.fit(x = x, y = y, offset = offset, distribution = distribution,
## : variable 50: PVRAAUT has no variation.

## Warning in gbm.fit(x = x, y = y, offset = offset, distribution = distribution,
## : variable 71: AVRAAUT has no variation.

## gbm(formula = Purchase ~ ., distribution = "bernoulli", data = train_set,
##     n.trees = 1000, shrinkage = 0.01)
## A gradient boosted model with bernoulli loss function.
## 1000 iterations were performed.
## There were 85 predictors of which 51 had non-zero influence.
```




```
##               var    rel.inf
## PPERSAUT PPERSAUT 14.57675709
## MKOOPKLA MKOOPKLA  9.50265357
## MOPLHOOG MOPLHOOG  7.37799005
## MBERMIDD MBERMIDD  5.76076392
## PBRAND     PBRAND  5.48277194
## ABRAND     ABRAND  4.39700295
## MGODGE     MGODGE  3.99434675
## MINK3045 MINK3045  3.51290076
## MAUT1       MAUT1  2.81629454
## MOSTYPE   MOSTYPE  2.72570844
## MGODPR     MGODPR  2.30408805
## MBERARBG MBERARBG  2.20649222
## MSKA         MSKA  2.16442017
## PWAPART   PWAPART  2.08836336
## MGODOV     MGODOV  1.84827998
## MBERHOOG MBERHOOG  1.81102959
## MSKC         MSKC  1.71054855
## MAUT2       MAUT2  1.64700459
## MINKGEM   MINKGEM  1.62159662
## PBYSTAND PBYSTAND  1.49907725
## MRELGE     MRELGE  1.33901832
## MAUT0       MAUT0  1.33676679
## MHHUUR     MHHUUR  1.29290755
## MFWEKIND MFWEKIND  1.25905508
## MSKB1       MSKB1  1.23987090
```

```
## MINK7512 MINK7512  1.18733937
## MRELOV     MRELOV  0.91827155
## MFGEKIND MFGEKIND  0.90489988
## MINK4575 MINK4575  0.84760079
## MOSHOOFD MOSHOOFD  0.84361448
## MSKD         MSKD  0.78456685
## MOPLMIDD MOPLMIDD  0.74653323
## MGODRK     MGODRK  0.72951628
## APERSAUT APERSAUT  0.66039728
## MHKOOP     MHKOOP  0.65438724
## MGEMOMV   MGEMOMV  0.64372664
## PMOTSCO   PMOTSCO  0.58857228
## MINK123M MINK123M  0.55765109
## MBERARBO MBERARBO  0.54069629
## MSKB2       MSKB2  0.51415079
## MINKM30   MINKM30  0.48863988
## PLEVEN     PLEVEN  0.48404853
## MZFONDS   MZFONDS  0.48117584
## MGEMLEEF MGEMLEEF  0.45320595
## MBERBOER MBERBOER  0.32956600
## MBERZELF MBERZELF  0.29202110
## MRELSA     MRELSA  0.27929478
## MZPART     MZPART  0.20701606
## MOPLLAAG MOPLLAAG  0.16082307
## MFALLEEN MFALLEEN  0.12402699
## MAANTHUI MAANTHUI  0.06254873
## PWABEDR   PWABEDR  0.00000000
## PWALAND   PWALAND  0.00000000
## PBESAUT   PBESAUT  0.00000000
## PVRAAUT   PVRAAUT  0.00000000
## PAANHANG PAANHANG  0.00000000
## PTRACTOR PTRACTOR  0.00000000
## PWERKT     PWERKT  0.00000000
## PBROM       PBROM  0.00000000
## PPERSONG PPERSONG  0.00000000
## PGEZONG   PGEZONG  0.00000000
## PWAOREG   PWAOREG  0.00000000
## PZEILPL   PZEILPL  0.00000000
## PPLEZIER PPLEZIER  0.00000000
## PFIETS     PFIETS  0.00000000
## PINBOED   PINBOED  0.00000000
## AWAPART   AWAPART  0.00000000
## AWABEDR   AWABEDR  0.00000000
## AWALAND   AWALAND  0.00000000
## ABESAUT   ABESAUT  0.00000000
## AMOTSCO   AMOTSCO  0.00000000
## AVRAAUT   AVRAAUT  0.00000000
## AAANHANG AAANHANG  0.00000000
## ATRACTOR ATRACTOR  0.00000000
## AWERKT     AWERKT  0.00000000
## ABROM       ABROM  0.00000000
## ALEVEN     ALEVEN  0.00000000
## APERSONG APERSONG  0.00000000
## AGEZONG   AGEZONG  0.00000000
```

```
## AWAOREG   AWAOREG  0.00000000
## AZEILPL   AZEILPL  0.00000000
## APLEZIER APLEZIER  0.00000000
## AFIETS     AFIETS  0.00000000
## AINBOED   AINBOED  0.00000000
## ABYSTAND ABYSTAND  0.00000000
```

PPERSAUT, MKOOPKLA, MOPLHOOG, and MBERMIDD are the most important predictors.

c.)Use the logistic regression model to predict the response on the test data. Predict that a person will make a purchase if the estimated probability of purchase is greater than 20%. Form a confusion matrix. What fraction of the people predicted to make a purchase do in fact make one?

```
##          Actual
## Predicted    0    1
##         0 1073   62
##         1   20    9
```

```
## Fraction of people predicted to make a purchase who actually make one: 0.3103448
```