Intro to ML Examination | Book - An Introduction to Statstical Learning - Second Edition

Aashi Aashi | aa92533

07/30/2022

Final Examination - Aashi Aashi (aa92533)

Chapter 2: Question 10

Part A

```
dim(Boston)
```

```
## [1] 506  13
```

Results

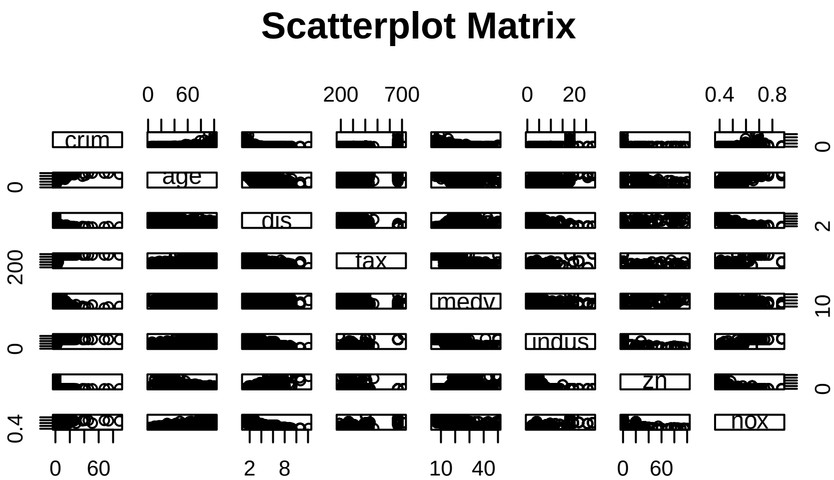
There are 506 rows in the Boston data set and 13 columns.

Response Variable: crim -per capita crime rate by town.

Predictors: a) zn proportion of residential land zoned for lots over 25,000 sq.ft. b) indus proportion of non-retail business acres per town. c) chas Charles River dummy variable (= 1 if tract bounds river; 0 otherwise). d) nox nitrogen oxides concentration (parts per 10 million). e) age proportion of owner-occupied units built prior to 1940. f) dis weighted mean of distances to five Boston employment centres. g) rad index of accessibility to radial highways. h) tax full-value property-tax rate per $10,000. i) ptratio pupil-teacher ratio by town. j) lstat lower status of the population (percent). k) medv median value of owner-occupied homes in $1000s.

Part B

```
pairs(~crim+age+dis+tax+medv+indus+zn+nox, data=Boston, main = "Scatterplot Matrix")
```



Variable ‘crim’ is statistically positive correlated with variable ‘age’ i.e. as per capita crime rate by town increases, the proportion of owner-occupied units built prior to 1940 increases

Variable ‘crim’ is statistically negative correlated with variable ‘dis’ i.e. as per capita crime rate by town increases, the weighted mean of distances to five Boston employment centres decreases

Variable ‘zn’ is statistically negative correlated with variable ‘indus’ i.e. as proportion of residential land zoned for lots over 25,000 sq.ft increases, the proportion of non-retail business acres per town decreases as residential areas are usually built far off from the industries.

Variable ‘zn’ is statistically negative correlated with variable ‘nox’ i.e. as proportion of residential land zoned for lots over 25,000 sq.ft increases, the nitrogen oxides concentration decreases

Variable ‘zn’ is statistically negative correlated with variable ‘lstat’ i.e. as proportion of residential land zoned for lots over 25,000 sq.ft increases, as the percentage of lower status of population decreases

Variable ‘indus’ is statistically negative correlated with variable ‘dis’ i.e. as proportion of non-retail business acres per town increases, the weighted mean of distances to five Boston employment centres decreases

Variable ‘nox’ is statistically negative correlated with variable ‘dis’ i.e. as nitrogen oxides concentration increases, the weighted mean of distances to five Boston employment centres decreases

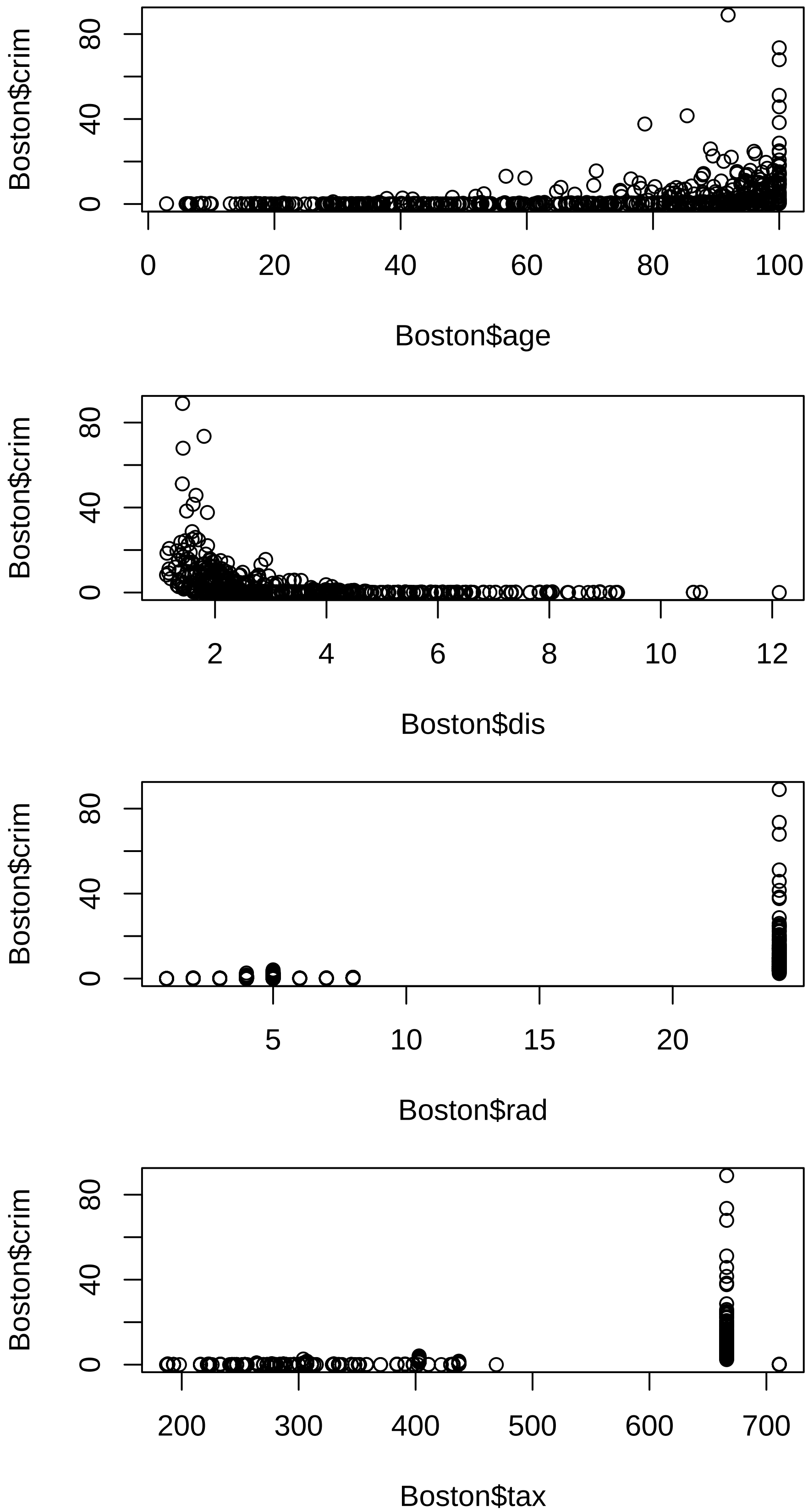
Variable ‘nox’ is statistically positive correlated with variable ‘age’ i.e. as nitrogen oxides concentration increases, the proportion of owner-occupied units built prior to 1940 increases

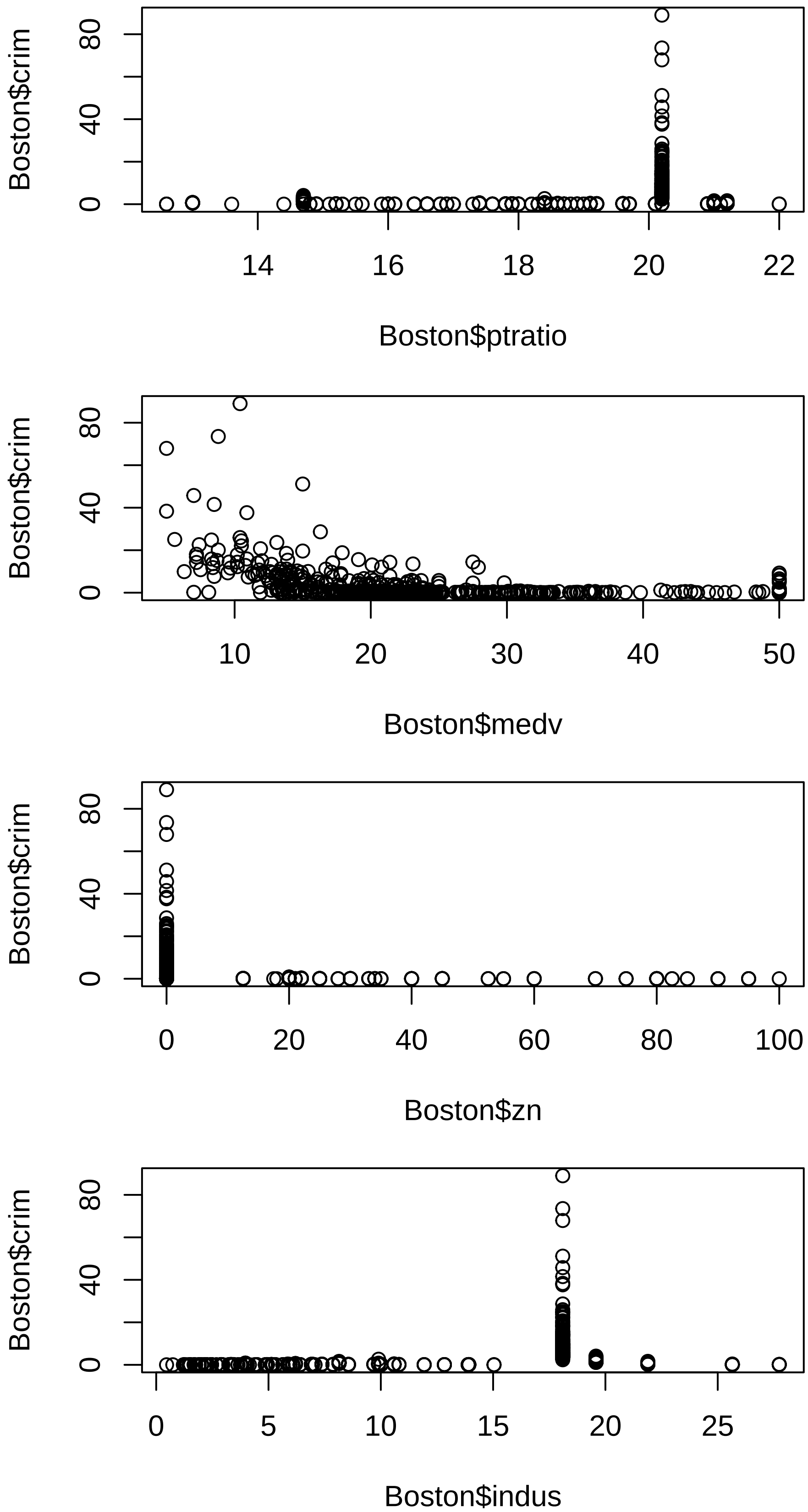
Variable ‘dis’ is statistically negative correlated with variable ‘lstat’ i.e. as weighted mean of distances to five Boston employment centres increases, the percentage of lower status of population decreases

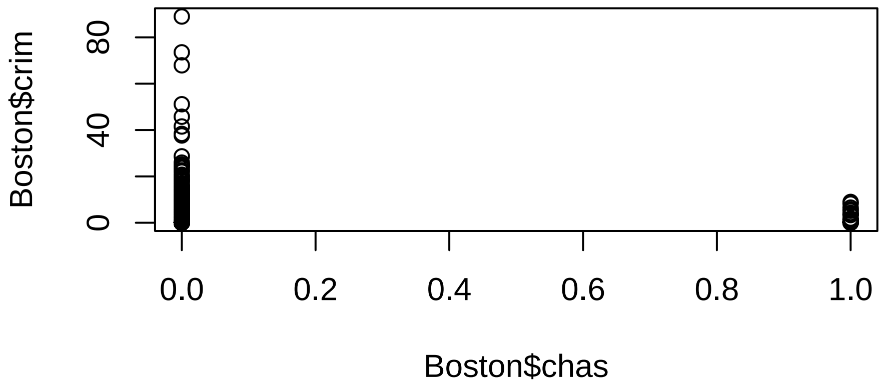
Variable ‘medv’ is statistically negative correlated with variable ‘crim’ i.e. as the per capita crime rate increases the median value of the owner-occupied homes decreases.

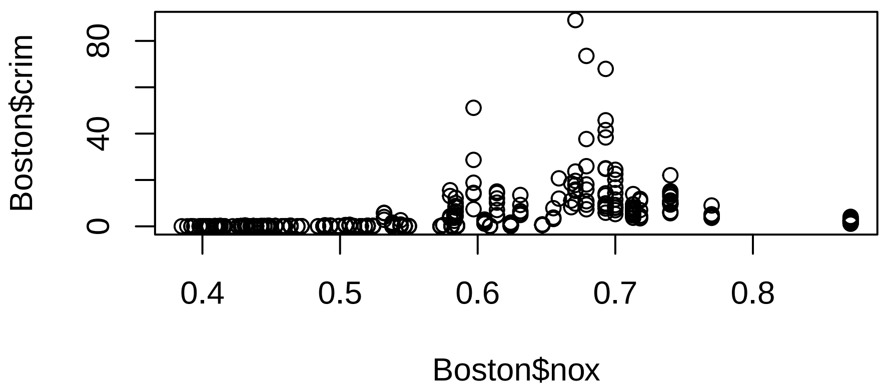
Variable ‘medv’ is statistically positive correlated with variable ‘rm’

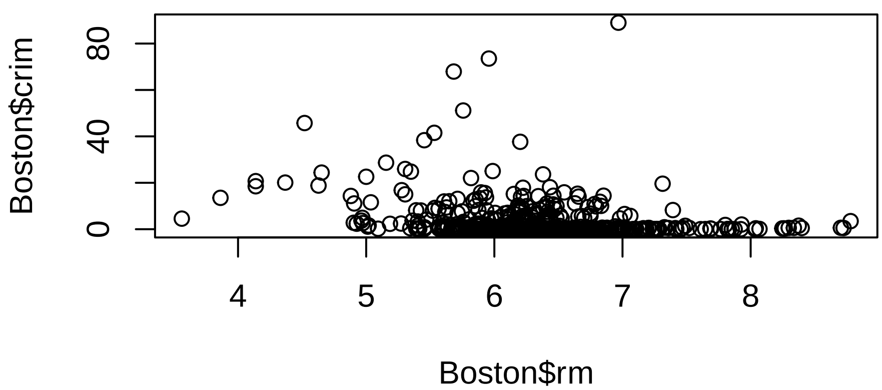
Part C

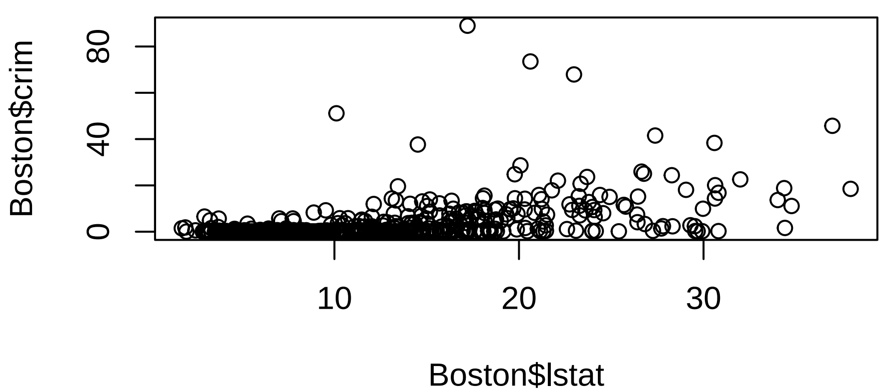

Boston$crim
Boston$age
Boston$crim
Boston$dis
Boston$crim
Boston$rad
Boston$crim
Boston$tax


Boston$crim
0
40
80
14
16
18
20
22
Boston$ptratio
Boston$crim
0
40
80
10
20
30
40
50
Boston$medv
Boston$crim
0
40
80
0
20
40
60
80
100
Boston$zn
Boston$crim
0
40
80
0
5
10
15
20
25
Boston$indus









Result-

Few Observations:

a) The crime rate dramatically increases as the nitrogen oxide content exceeds a threshold of 0.6, indicating that regions with high crime rates typically have high nox levels. This might be because there isn't much regulation in these places, which leads to high nox and crime rates.

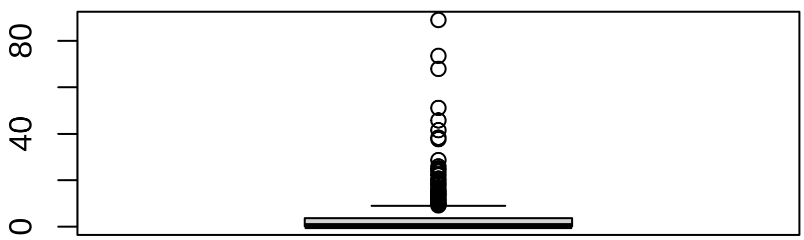
b) The crime rate rises in tandem with the percentage of older housing units. This may be because locations with a higher percentage of older structures have lower building prices, which draw people with poor incomes, who may also include criminals.

c) Plot suggests that the crime rate is high for the areas within 3 weighted mean distance to five Boston employment centers.

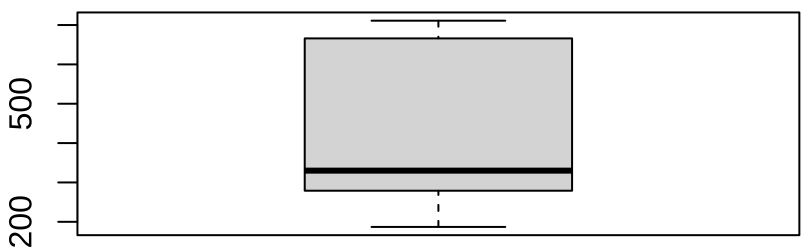
d) Plot suggests a non-linear trend between criminal rate and lstat. As the lstat increases, crime rate increases. This may be due to the fact that if the lower status of population comprises of more criminals and when lstat increases, number of criminals in that locality also increase leading to more crime rate.

e) The median home value and the crime rate have a non-linear inverse connection. Criminality declines sharply until medv = 25, at which point it plateaus. This could be because locations with higher medv have better local communities, higher-income residents, and fewer criminals as a result. Additionally, increased security in areas with high medv could cut crime rates

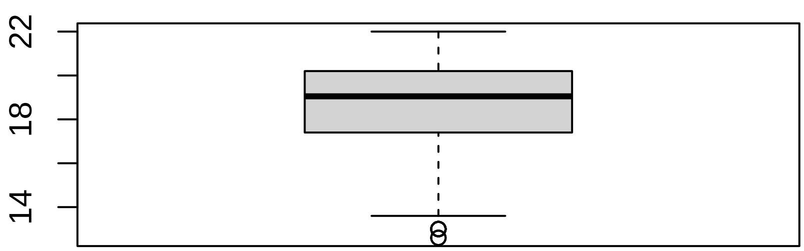
Part D



Result - The box plot reveals that there are numerous suburbs with crime rates higher than Q3 + 1.5 IQR, indicating that these locations are outliers.



Result - There are no data points in this set that are either above or below Q3 + 1.5 IQR, indicating that there are no outliers.



Result - From the boxplot, we can see that there are 2 suburbs which have pupil-teacher ratio less than Q1-1.5IQR, suggesting these suburbs are outliers.

Part E

```
## [1] "There are 35 suburbs that bound the Charles river."
```

Part F

```
## [1] "The median pupil-teacher ratio among the towns in this data set: 19.05"
```

Part G

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

Result - Both the houses are in Suburb with crim, indus, nox ,age, rad, tax, ptratio lying on or beyond 75th percentile

Part H

Number of records with average number of rooms per dwelling >7

```
## [1] "There are 64 suburbs average more than 7 rooms per dwelling"
```

Number of records with average number of rooms per dwelling >8

```
## [1] "There are 13 suburbs average more than 8 rooms per dwelling"
```

Number of records with average number of rooms per dwelling <=8

```
## [1] "There are 493 suburbs average less than 8 rooms per dwelling"

## [1] "The mean median value of owner-occupied homes in suburbs with average number of rooms per dwelli

## [1] "The mean median value of owner-occupied homes in suburbs with average number of rooms per dwelli
```

Result - The mean median value of owner-occupied homes in suburbs with average number of rooms per dwelling >8 (44.2) is more than twice than that of homes in suburbs with rooms <=8 (21.96)

```
## [1] "The mean of number of lower stat people in suburbs with average number of rooms per dwelling gr

## [1] "The mean ofmnumber of lower stat people in suburbs with average number of rooms per dwelling le
```

Result - Suburbs with > 8 average number of rooms per dwelling have ~4% (on average) proportion of lower status people, which is almost 3 times higher than that of homes in suburbs with rooms <=8. ~12.87%. This makes sense as the medv value for such suburbs are quite high as discussed in previous point.

```
## [1] "The mean age of houses in suburbs with average number of rooms per dwelling greater than 8 is 7

## [1] "The mean age of houses in suburbs with average number of rooms per dwelling less than or equal 8
```

Result - There is no significant difference in the age of houses in suburbs with average number of rooms per dwelling >8 and that of homes in suburbs with rooms <=8

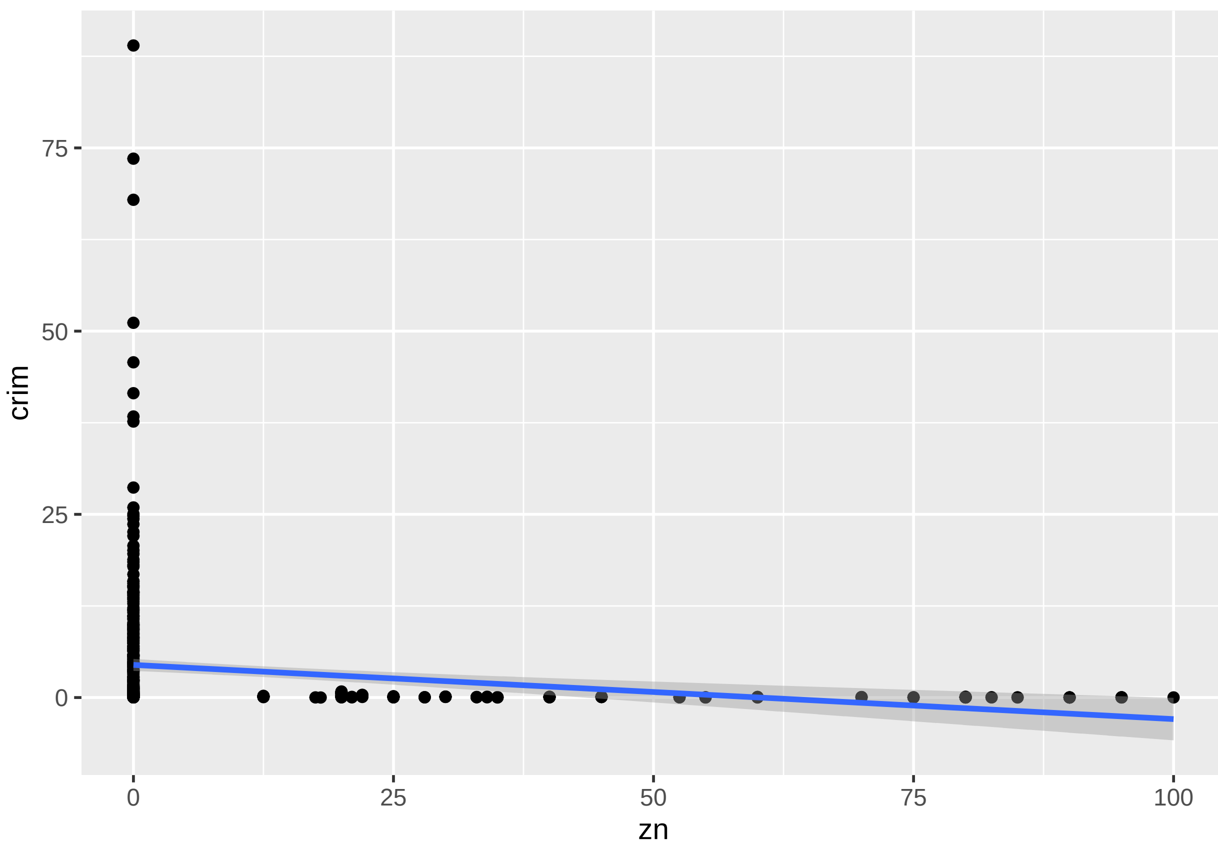
Chapter 3: Question 15

Part A

Crime vs 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
```

```
## `geom_smooth()` using formula 'y ~ x'
```

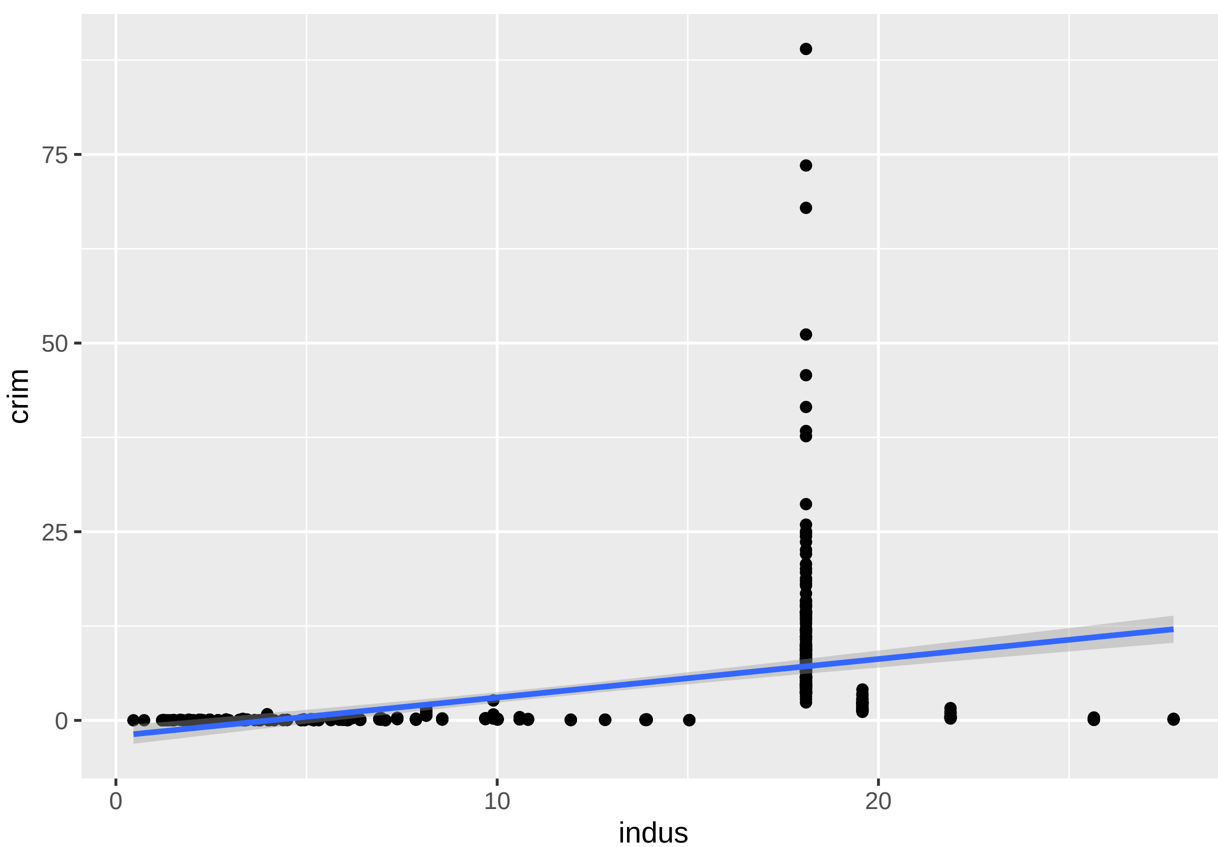


Result - zn explain less than 5% variance in crim (evident from the graph), 'zn' seems to have a statistically significant coefficient in predicting 'crim'.

Crime vs 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
```

```
## `geom_smooth()` using formula 'y ~ x'
```

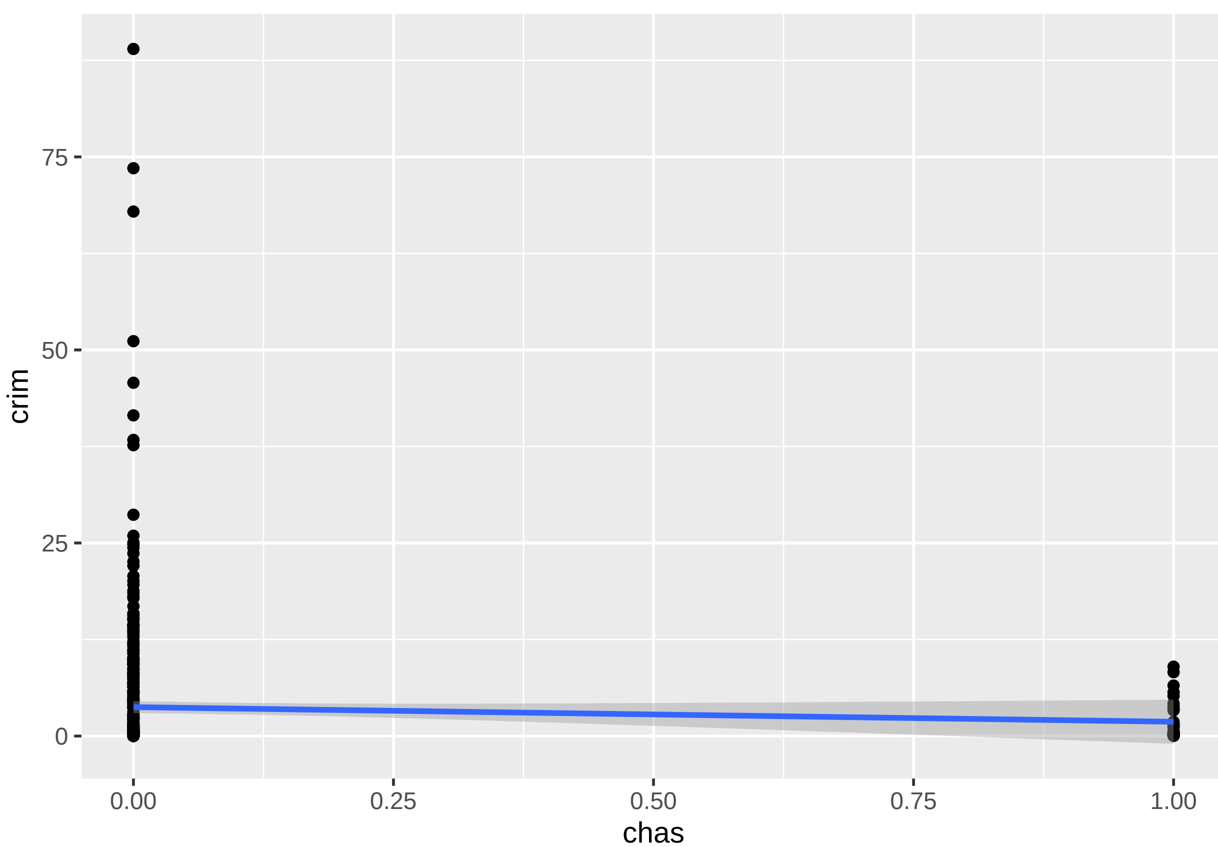


Result - According to both the graph and the regression results, "indus" appears to have a statistically significant positive coefficient in predicting "crim," explaining less than 15% of the variance in crim.

Crime vs 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 ***
## 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
```

```
## `geom_smooth()` using formula 'y ~ x'
```

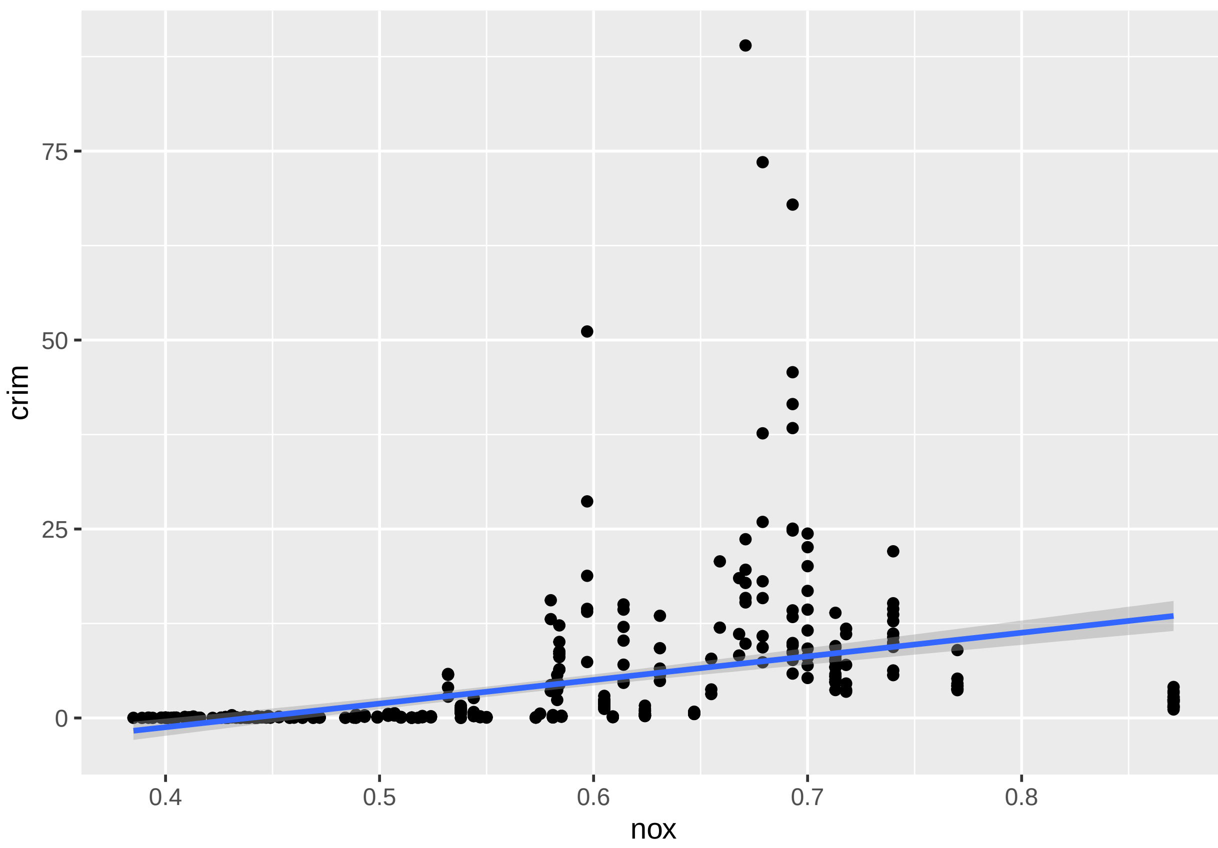


Result - Chas does not have a statistically significant coefficient and only accounts for a small portion of the variance in the crim variable. Additionally, we can observe from the graph that chas only accepts discrete values between 0 and 1, and the graph does not appear to show any relationship.

Crime vs 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

## `geom_smooth()` using formula 'y ~ x'
```

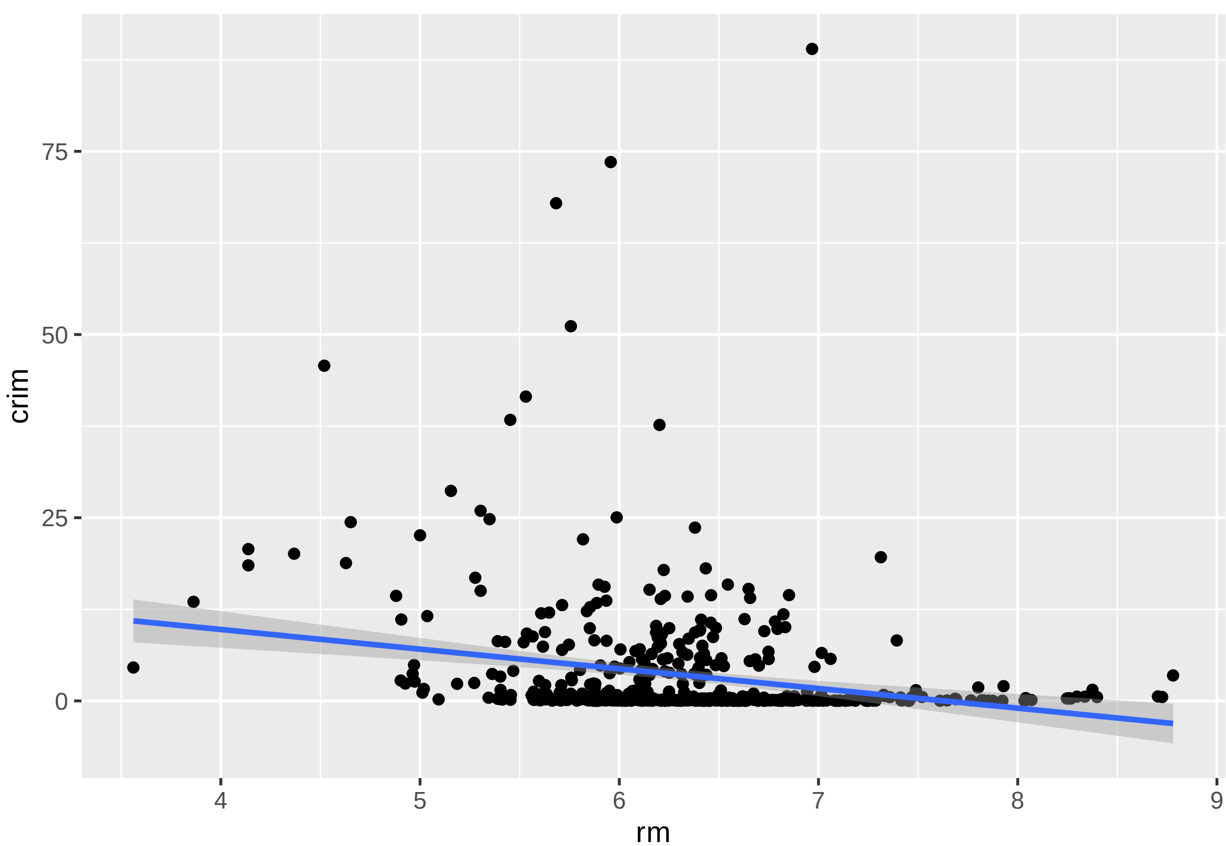


Result - Nox account for 17% of the variation in "crim." Additionally, the graph shows that they are positively associated with a statistically significant positive coefficient.

Crime vs 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
```

```
## `geom_smooth()` using formula 'y ~ x'
```

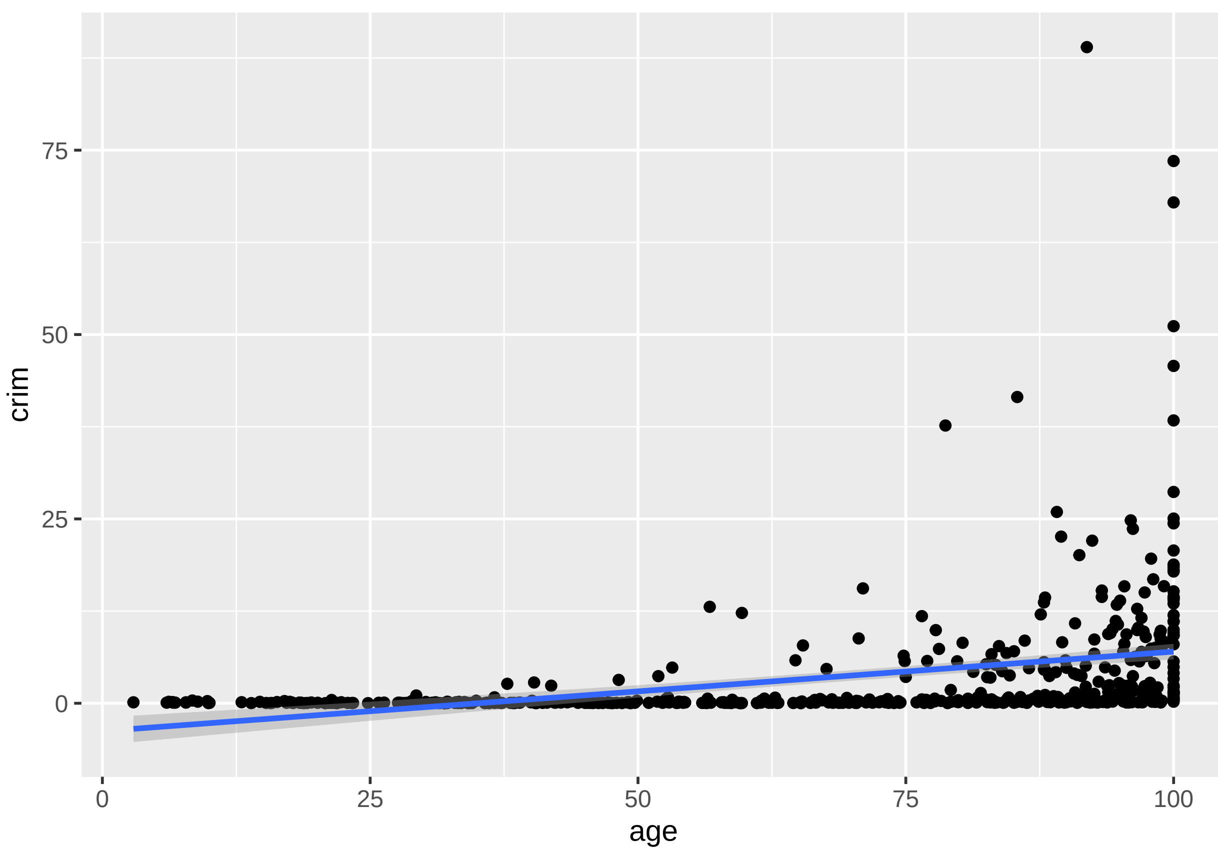


Result - ‘rm’ explain less than 5% variance in ‘crim’. ‘rm’ has a negative correlation with ‘crim’ with a statistically significant coefficient. From the graph we can see that as rm increases, crim rate decreases, but again, it explains very less variance in crim.

Crime vs 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
```

```
## `geom_smooth()` using formula 'y ~ x'
```

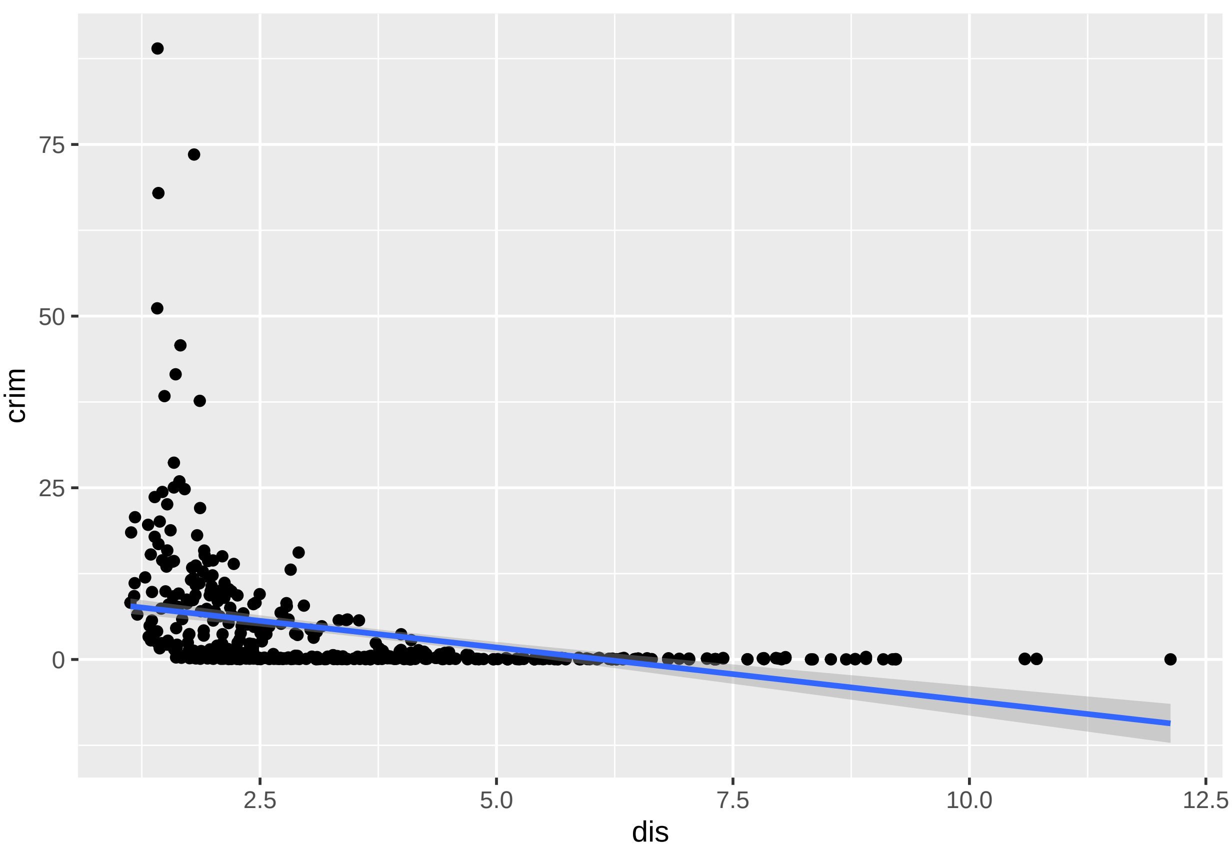


Result- Age appears to have a statistically significant positive coefficient in predicting "crim," explaining around 12% of the variance in that variable. We can observe from the graph that as age grows, crime also rises.

Crime vs 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

## `geom_smooth()` using formula 'y ~ x'
```

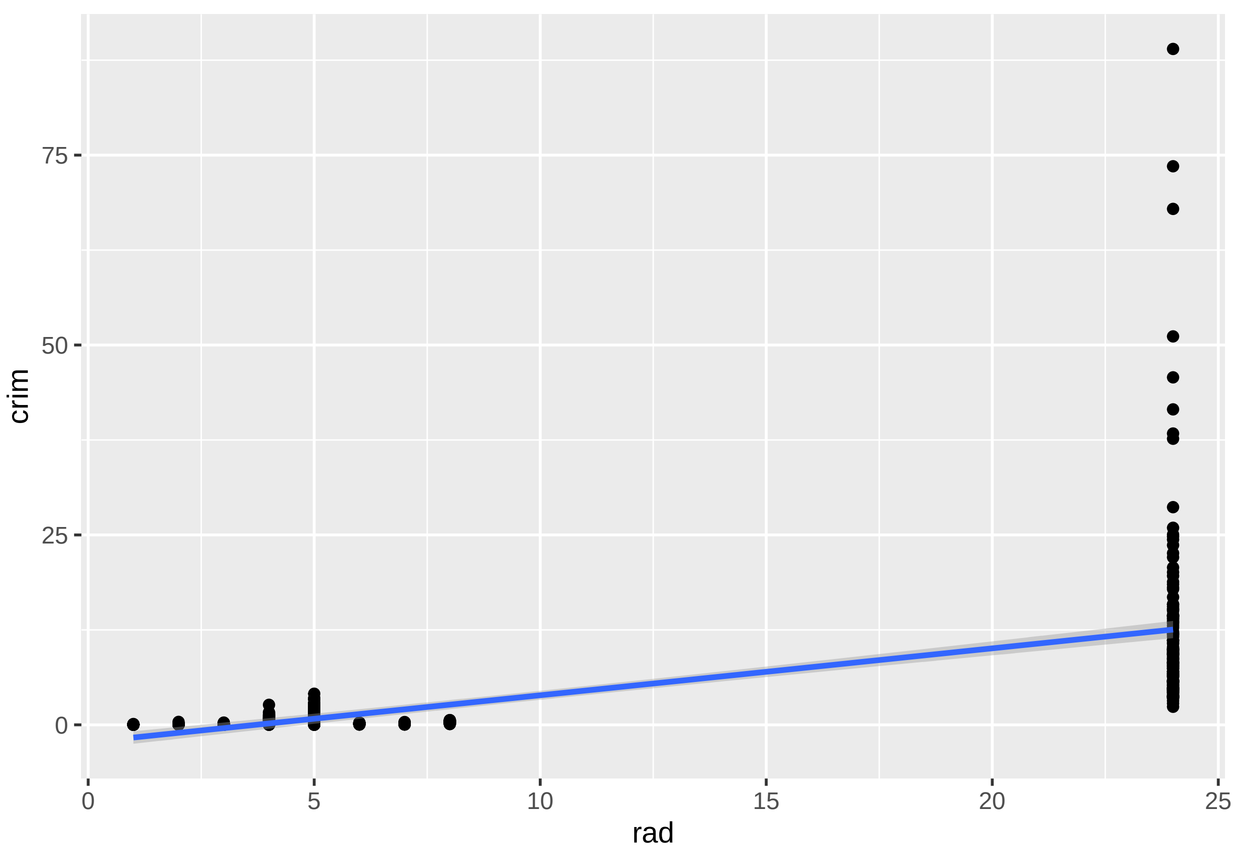


Result - "Dis" explains around 15% of the variation in "crim." Dis and Crim have a statistically significant negative coefficient negative association. This negative tendency is shown in the graph.

Crime vs 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

## `geom_smooth()` using formula 'y ~ x'
```

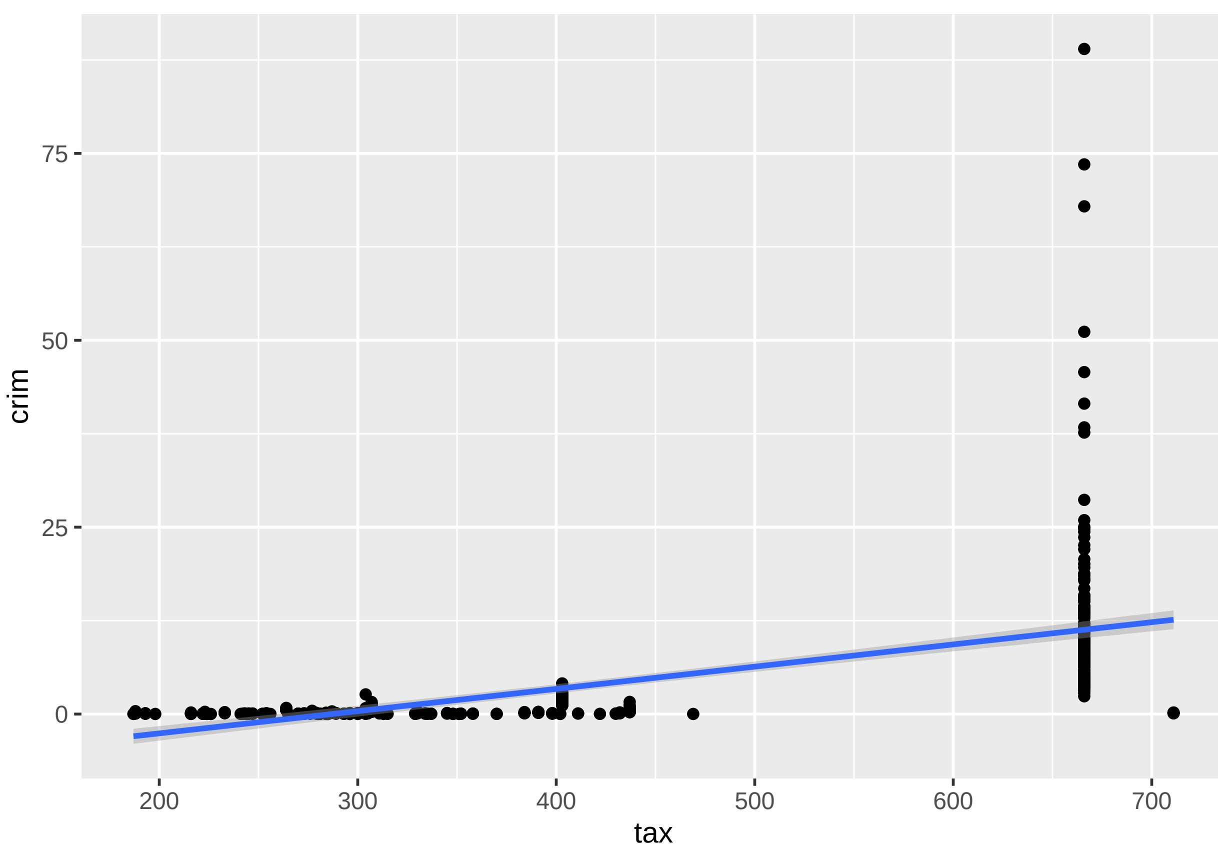


Result - 'rad' explain ~40% variance in 'crim'. 'rad' has a positive correlation with 'crim' with a statistically significant positive coefficient.

Crime vs 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
```

```
## `geom_smooth()` using formula 'y ~ x'
```

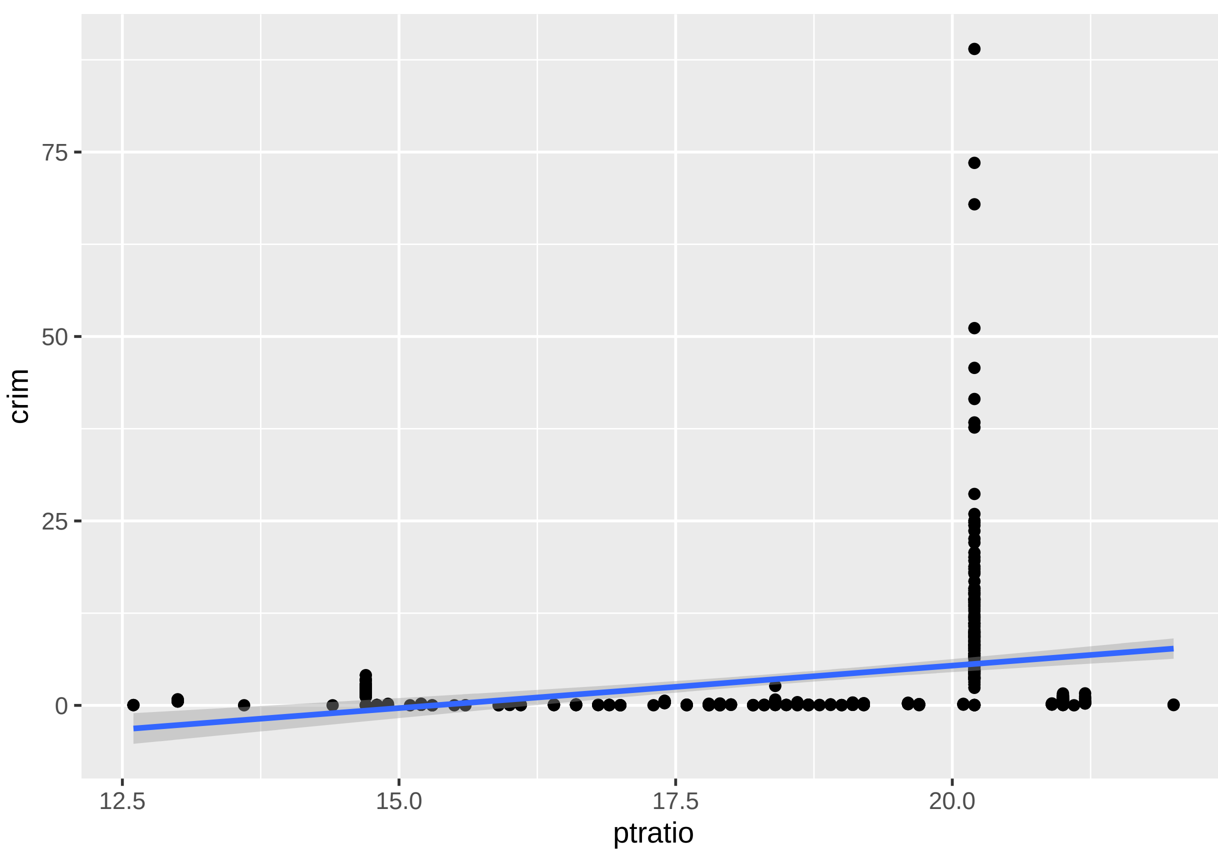


Result - A positive association between tax rate and crim explains 33% of the variation. It has a positive coefficient that is statistically significant.

Crime vs 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
```

```
## `geom_smooth()` using formula 'y ~ x'
```

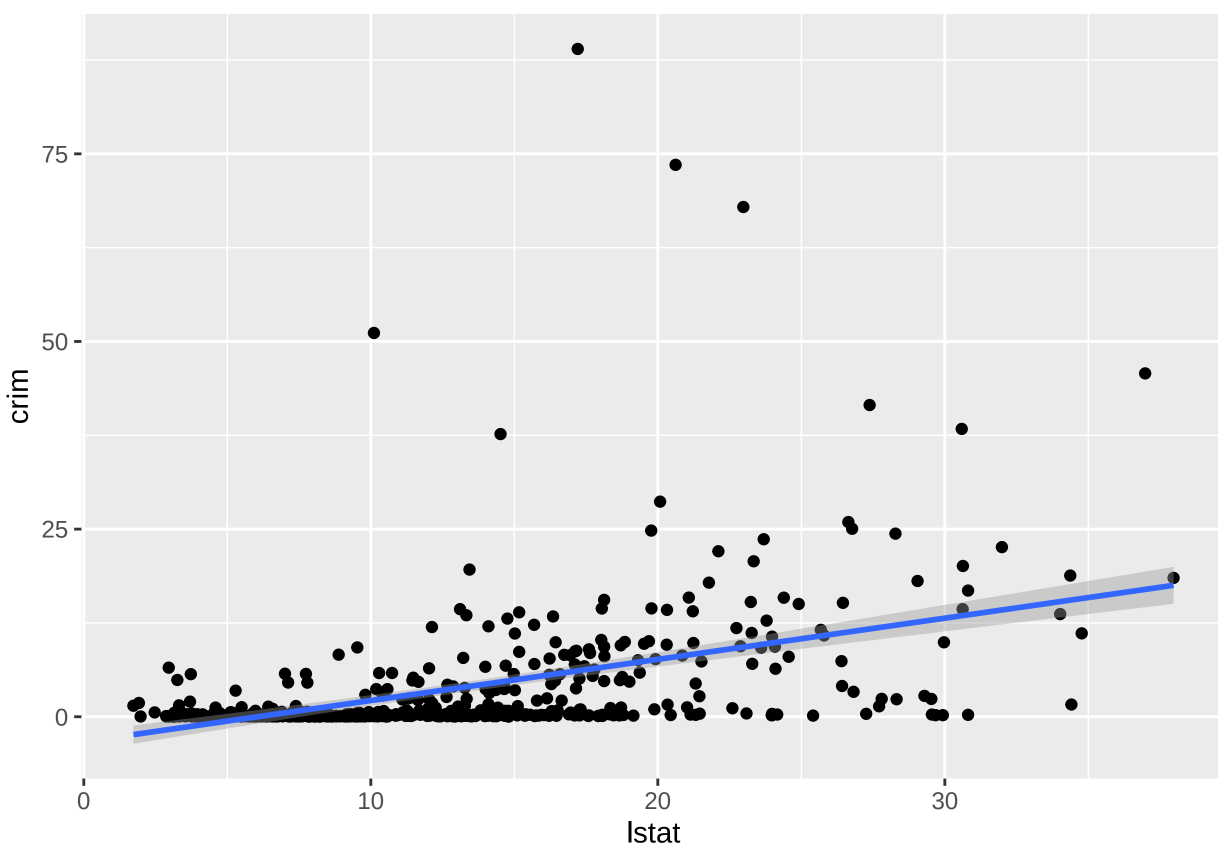


Result - ptratio explains <10% variation in crim with a positive correlation. It has a statistically significant positive coefficient.

Crime vs 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
```

```
## `geom_smooth()` using formula 'y ~ x'
```

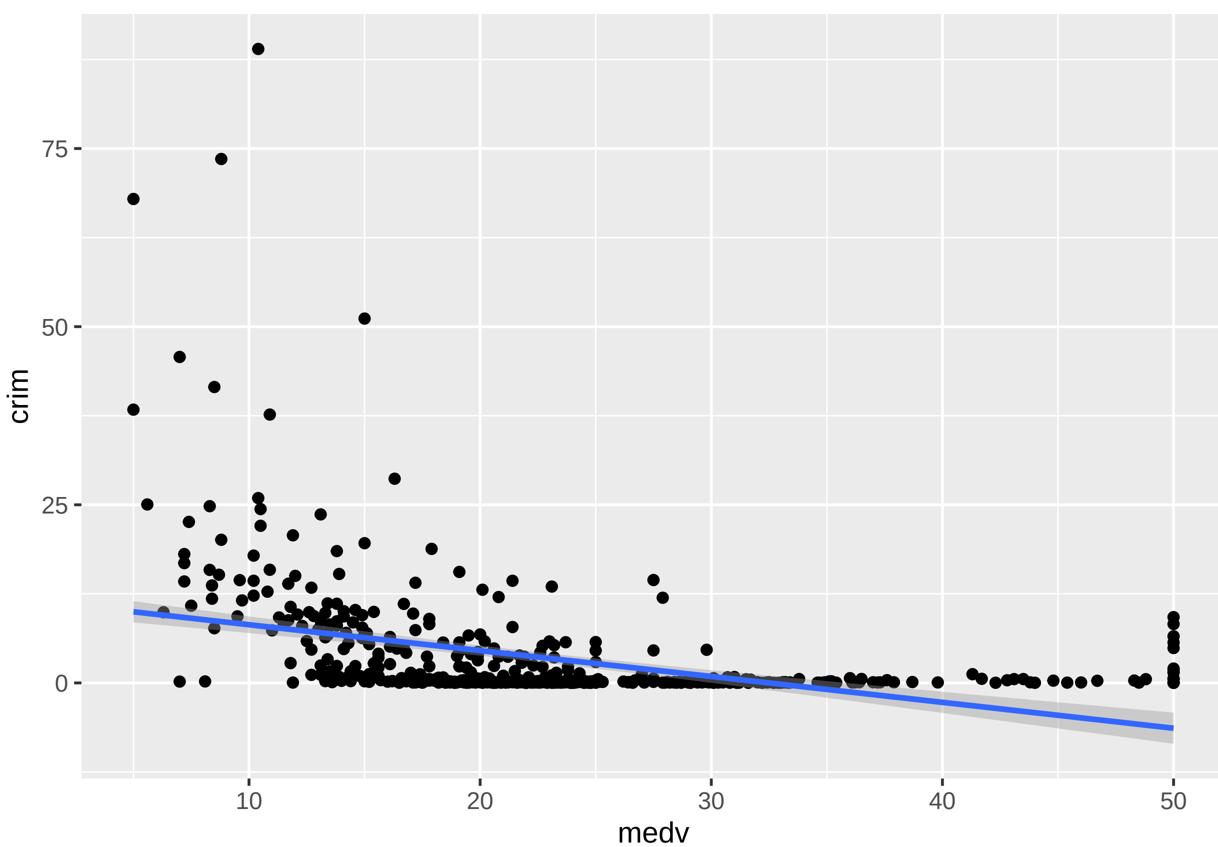


Result - lstat has a positive correlation with crim with a statistically significant coefficient and explains ~20% variance in crim.

Crime vs 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
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Result - medv has a negative correlation with crim with a statistically significant coefficient and explains ~15% variance in crim.

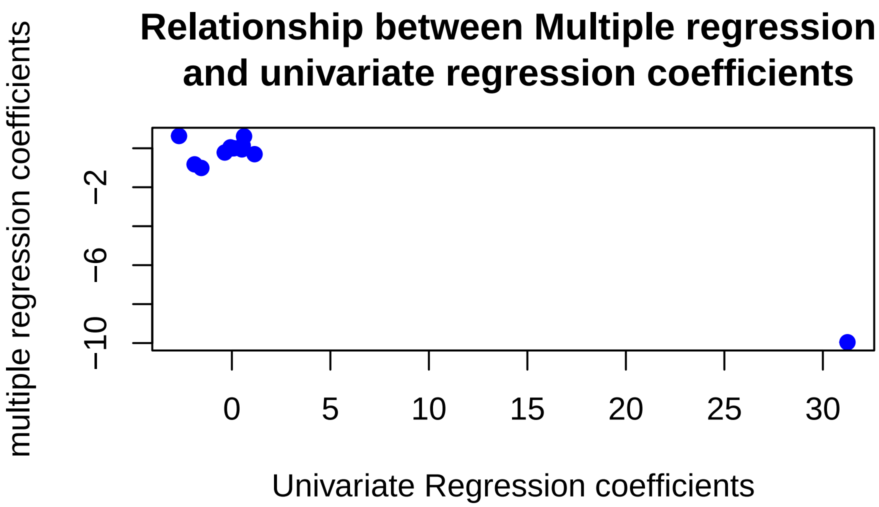
Part B

```
##
## Call:
## lm(formula = crim ~ ., data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -8.534 -2.248 -0.348  1.087 73.923
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 13.7783938  7.0818258   1.946 0.052271 .
## zn           0.0457100  0.0187903   2.433 0.015344 *
## indus       -0.0583501  0.0836351  -0.698 0.485709
## chas        -0.8253776  1.1833963  -0.697 0.485841
## nox         -9.9575865  5.2898242  -1.882 0.060370 .
## rm           0.6289107  0.6070924   1.036 0.300738
## age         -0.0008483  0.0179482  -0.047 0.962323
## dis         -1.0122467  0.2824676  -3.584 0.000373 ***
## rad          0.6124653  0.0875358   6.997 8.59e-12 ***
## tax         -0.0037756  0.0051723  -0.730 0.465757
## ptratio     -0.3040728  0.1863598  -1.632 0.103393
## lstat        0.1388006  0.0757213   1.833 0.067398 .
## medv        -0.2200564  0.0598240  -3.678 0.000261 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.46 on 493 degrees of freedom
## Multiple R-squared:  0.4493, Adjusted R-squared:  0.4359
## F-statistic: 33.52 on 12 and 493 DF,  p-value: < 2.2e-16
```

Result - These set of features explain ~44% variance in 'crim', and a residual standard error of 6.439 (model run on complete dataset) with 4 features having statistically significant coefficients at significance level (alpha) = 0.05

For variables- zn, dis, rad, medv , we can reject the null hypothesis H0 at significance level (alpha) = 0.05

Part C



Results -

Nearly all of the features had statistically significant coefficients for predicting "crim" in the results of univariate regression. However, when all of the variables are taken into account, only a small number of them have statistically significant coefficients, suggesting that even when all of the variables are taken into account, only a small number can accurately predict "crime". The only features with statistically significant coefficients were zn, dis, rad, black, and medv.

Additionally, we can observe that some coefficients that had favorable effects in univariate regression are now having adverse effects in multivariate analysis, and vice versa. However, it's noteworthy to observe that the multivariate results do not show a statistically significant coefficient for these coefficients where we see such a dramatic change.

As depicted from the plot, the coefficient value for predictor nox has significantly changed from linear (univariate) model to the multiple regression model. The value was positive (~31) in the linear model and has now reduced to very high negative value (-10). The change in coefficient value of nox is very high

Only 'zn' changes its impact from negative in univariate to positive in multivariate, but the overall deviation (-0.07 to +0.04) is still very low.

Part D

Crime vs Zn (Polynomial Fit)

```
##
## Call:
## lm(formula = crim ~ zn + I(zn^2) + I(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)  4.846e+00  4.330e-01  11.192  < 2e-16 ***
## zn          -3.322e-01  1.098e-01  -3.025  0.00261 **
```

```
## I(zn^2)      6.483e-03  3.861e-03   1.679  0.09375 .
## I(zn^3)     -3.776e-05  3.139e-05  -1.203  0.22954
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.372 on 502 degrees of freedom
## Multiple R-squared:  0.05824,    Adjusted R-squared:  0.05261
## F-statistic: 10.35 on 3 and 502 DF,  p-value: 1.281e-06
```

Crime vs Indus (Polynomial Fit)

```
##
## Call:
## lm(formula = crim ~ indus + I(indus^2) + I(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)  3.6625683  1.5739833   2.327   0.0204 *
## indus       -1.9652129  0.4819901  -4.077 5.30e-05 ***
## I(indus^2)   0.2519373  0.0393221   6.407 3.42e-10 ***
## I(indus^3)  -0.0069760  0.0009567  -7.292 1.20e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.423 on 502 degrees of freedom
## Multiple R-squared:  0.2597, Adjusted R-squared:  0.2552
## F-statistic: 58.69 on 3 and 502 DF,  p-value: < 2.2e-16
```

Crime vs Chas (Polynomial Fit)

```
##
## Call:
## lm(formula = crim ~ chas + I(chas^2) + I(chas^3))
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.738 -3.661 -3.435  0.018 85.232
##
## Coefficients: (2 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.7444     0.3961   9.453   <2e-16 ***
## chas         -1.8928     1.5061  -1.257    0.209
## I(chas^2)         NA         NA      NA       NA
## I(chas^3)         NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.597 on 504 degrees of freedom
## Multiple R-squared:  0.003124,   Adjusted R-squared:  0.001146
```

```
## F-statistic: 1.579 on 1 and 504 DF,  p-value: 0.2094
```

Crime vs Nox (Polynomial Fit)

```
##
## Call:
## lm(formula = crim ~ nox + I(nox^2) + I(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)   233.09      33.64   6.928 1.31e-11 ***
## nox         -1279.37     170.40  -7.508 2.76e-13 ***
## I(nox^2)     2248.54     279.90   8.033 6.81e-15 ***
## I(nox^3)    -1245.70     149.28  -8.345 6.96e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.234 on 502 degrees of freedom
## Multiple R-squared:  0.297,  Adjusted R-squared:  0.2928
## F-statistic: 70.69 on 3 and 502 DF,  p-value: < 2.2e-16
```

Crime vs RM (Polynomial Fit)

```
##
## Call:
## lm(formula = crim ~ rm + I(rm^2) + I(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) 112.6246    64.5172   1.746   0.0815 .
## rm          -39.1501    31.3115  -1.250   0.2118
## I(rm^2)       4.5509     5.0099   0.908   0.3641
## I(rm^3)      -0.1745     0.2637  -0.662   0.5086
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.33 on 502 degrees of freedom
## Multiple R-squared:  0.06779,    Adjusted R-squared:  0.06222
## F-statistic: 12.17 on 3 and 502 DF,  p-value: 1.067e-07
```

Crime vs Age (Polynomial Fit)

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

```
## lm(formula = crim ~ age + I(age^2) + I(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) -2.549e+00  2.769e+00  -0.920  0.35780
## age          2.737e-01  1.864e-01   1.468  0.14266
## I(age^2)    -7.230e-03  3.637e-03  -1.988  0.04738 *
## I(age^3)     5.745e-05  2.109e-05   2.724  0.00668 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.84 on 502 degrees of freedom
## Multiple R-squared:  0.1742, Adjusted R-squared:  0.1693
## F-statistic: 35.31 on 3 and 502 DF,  p-value: < 2.2e-16
```

Crime vs Dis (Polynomial Fit)

```
##
## Call:
## lm(formula = crim ~ dis + I(dis^2) + I(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)  30.0476     2.4459  12.285  < 2e-16 ***
## dis         -15.5543     1.7360  -8.960  < 2e-16 ***
## I(dis^2)      2.4521     0.3464   7.078 4.94e-12 ***
## I(dis^3)     -0.1186     0.0204  -5.814 1.09e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.331 on 502 degrees of freedom
## Multiple R-squared:  0.2778, Adjusted R-squared:  0.2735
## F-statistic: 64.37 on 3 and 502 DF,  p-value: < 2.2e-16
```

Crime vs Rad (Polynomial Fit)

```
##
## Call:
## lm(formula = crim ~ rad + I(rad^2) + I(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) -0.605545   2.050108  -0.295    0.768
## rad          0.512736   1.043597   0.491    0.623
## I(rad^2)    -0.075177   0.148543  -0.506    0.613
## I(rad^3)     0.003209   0.004564   0.703    0.482
##
## Residual standard error: 6.682 on 502 degrees of freedom
## Multiple R-squared:    0.4,  Adjusted R-squared:  0.3965
## F-statistic: 111.6 on 3 and 502 DF,  p-value: < 2.2e-16
```

Crime vs Tax (Polynomial Fit)

```
##
## Call:
## lm(formula = crim ~ tax + I(tax^2) + I(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)  1.918e+01  1.180e+01   1.626    0.105
## tax         -1.533e-01  9.568e-02  -1.602    0.110
## I(tax^2)     3.608e-04  2.425e-04   1.488    0.137
## I(tax^3)    -2.204e-07  1.889e-07  -1.167    0.244
##
## Residual standard error: 6.854 on 502 degrees of freedom
## Multiple R-squared:  0.3689, Adjusted R-squared:  0.3651
## F-statistic:  97.8 on 3 and 502 DF,  p-value: < 2.2e-16
```

Crime vs PTRATIO (Polynomial Fit)

```
##
## Call:
## lm(formula = crim ~ ptratio + I(ptratio^2) + I(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)  477.18405  156.79498   3.043  0.00246 **
## ptratio      -82.36054   27.64394  -2.979  0.00303 **
## I(ptratio^2)   4.63535    1.60832   2.882  0.00412 **
## I(ptratio^3)  -0.08476    0.03090  -2.743  0.00630 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.122 on 502 degrees of freedom
## Multiple R-squared:  0.1138, Adjusted R-squared:  0.1085
## F-statistic: 21.48 on 3 and 502 DF,  p-value: 4.171e-13
```

Crime vs LSTAT (Polynomial Fit)

```
##
## Call:
## lm(formula = crim ~ lstat + I(lstat^2) + I(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)  1.2009656  2.0286452   0.592   0.5541
## lstat       -0.4490656  0.4648911  -0.966   0.3345
## I(lstat^2)   0.0557794  0.0301156   1.852   0.0646 .
## I(lstat^3)  -0.0008574  0.0005652  -1.517   0.1299
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.629 on 502 degrees of freedom
## Multiple R-squared:  0.2179, Adjusted R-squared:  0.2133
## F-statistic: 46.63 on 3 and 502 DF,  p-value: < 2.2e-16
```

Crime vs Medv (Polynomial Fit)

```
##
## Call:
## lm(formula = crim ~ medv + I(medv^2) + I(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) 53.1655381  3.3563105  15.840  < 2e-16 ***
## medv        -5.0948305  0.4338321 -11.744  < 2e-16 ***
## I(medv^2)    0.1554965  0.0171904   9.046  < 2e-16 ***
## I(medv^3)   -0.0014901  0.0002038  -7.312 1.05e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.569 on 502 degrees of freedom
## Multiple R-squared:  0.4202, Adjusted R-squared:  0.4167
## F-statistic: 121.3 on 3 and 502 DF,  p-value: < 2.2e-16
```

```
##    Features Linear(Error) Polynomial(Error) Linear(Adj Rsq) Polynomial(Adj Rsq)
## 1        zn      8.435290          8.372207      0.03828352          0.05261394
## 2     indus      7.866281          7.423121      0.16365394          0.25523350
## 3      chas      8.596615          8.596615      0.00114594          0.00114594
## 4       nox      7.809972          7.233605      0.17558468          0.29277657
## 5        rm      8.400586          8.329676      0.04618036          0.06221506
## 6       age      8.056649          7.839703      0.12268419          0.16929612
```

```
## 7       dis      7.965369          7.331479      0.14245126          0.27350898
## 8       rad      6.717752          6.682402      0.39004886          0.39645144
## 9       tax      6.996901          6.853707      0.33830395          0.36511046
## 10  ptratio      8.240212          8.121583      0.08225111          0.10848545
## 11    lstat      7.664461          7.629436      0.20601869          0.21325872
## 12     medv      7.934451          6.569152      0.14909551          0.41673532
```

Result - As seen from above table, Polynomial has higher Adjusted R2 error for Nox,Indus, Dis,Medv, which means it fits the data-set much better

Chapter 6 : Question 9

Part A

```
## [1] "Few records of Train"
```

```
##                              Private Apps Accept Enroll Top10perc Top25perc
## Abilene Christian University     Yes 1660   1232    721        23        52
## Adelphi University               Yes 2186   1924    512        16        29
## Adrian College                   Yes 1428   1097    336        22        50
## Alaska Pacific University        Yes  193    146     55        16        44
## Alderson-Broaddus College        Yes  582    498    172        21        44
## Alfred University                Yes 1732   1425    472        37        75
##                              F.Undergrad P.Undergrad Outstate Room.Board Books
## Abilene Christian University        2885         537     7440       3300   450
## Adelphi University                  2683        1227    12280       6450   750
## Adrian College                      1036          99    11250       3750   400
## Alaska Pacific University            249         869     7560       4120   800
## Alderson-Broaddus College            799          78    10468       3380   660
## Alfred University                   1830         110    16548       5406   500
##                              Personal PhD Terminal S.F.Ratio perc.alumni Expend
## Abilene Christian University     2200  70       78      18.1          12   7041
## Adelphi University               1500  29       30      12.2          16  10527
## Adrian College                   1165  53       66      12.9          30   8735
## Alaska Pacific University        1500  76       72      11.9           2  10922
## Alderson-Broaddus College        1800  40       41      11.5          15   8991
## Alfred University                 600  82       88      11.3          31  10932
##                              Grad.Rate
## Abilene Christian University        60
## Adelphi University                  56
## Adrian College                      54
## Alaska Pacific University           15
## Alderson-Broaddus College           52
## Alfred University                   73
```

```
## [1] "Few records of Test"
```

```
##                                         Private Apps Accept Enroll Top10perc
## Agnes Scott College                         Yes  417    349    137        60
## Albertson College                           Yes  587    479    158        38
## Albertus Magnus College                     Yes  353    340    103        17
```

```
## Albion College                              Yes 1899   1720    489        37
## Albright College                            Yes 1038    839    227        30
## Allentown Coll. of St. Francis de Sales     Yes 1179    780    290        38
##                                         Top25perc F.Undergrad P.Undergrad
## Agnes Scott College                            89         510          63
## Albertson College                              62         678          41
## Albertus Magnus College                        45         416         230
## Albion College                                 68        1594          32
## Albright College                               63         973         306
## Allentown Coll. of St. Francis de Sales        64        1130         638
##                                         Outstate Room.Board Books Personal PhD
## Agnes Scott College                        12960       5450   450      875  92
## Albertson College                          13500       3335   500      675  67
## Albertus Magnus College                    13290       5720   500     1500  90
## Albion College                             13868       4826   450      850  89
## Albright College                           15595       4400   300      500  79
## Allentown Coll. of St. Francis de Sales     9690       4785   600     1000  60
##                                         Terminal S.F.Ratio perc.alumni Expend
## Agnes Scott College                           97       7.7          37  19016
## Albertson College                             73       9.4          11   9727
## Albertus Magnus College                       93      11.5          26   8861
## Albion College                               100      13.7          37  11487
## Albright College                              84      11.3          23  11644
## Allentown Coll. of St. Francis de Sales       84      13.3          21   7940
##                                         Grad.Rate
## Agnes Scott College                            59
## Albertson College                              55
## Albertus Magnus College                        63
## Albion College                                 73
## Albright College                               80
## Allentown Coll. of St. Francis de Sales        74
```

Result - Dataset has been split into Train and Test in the ration of 80-20

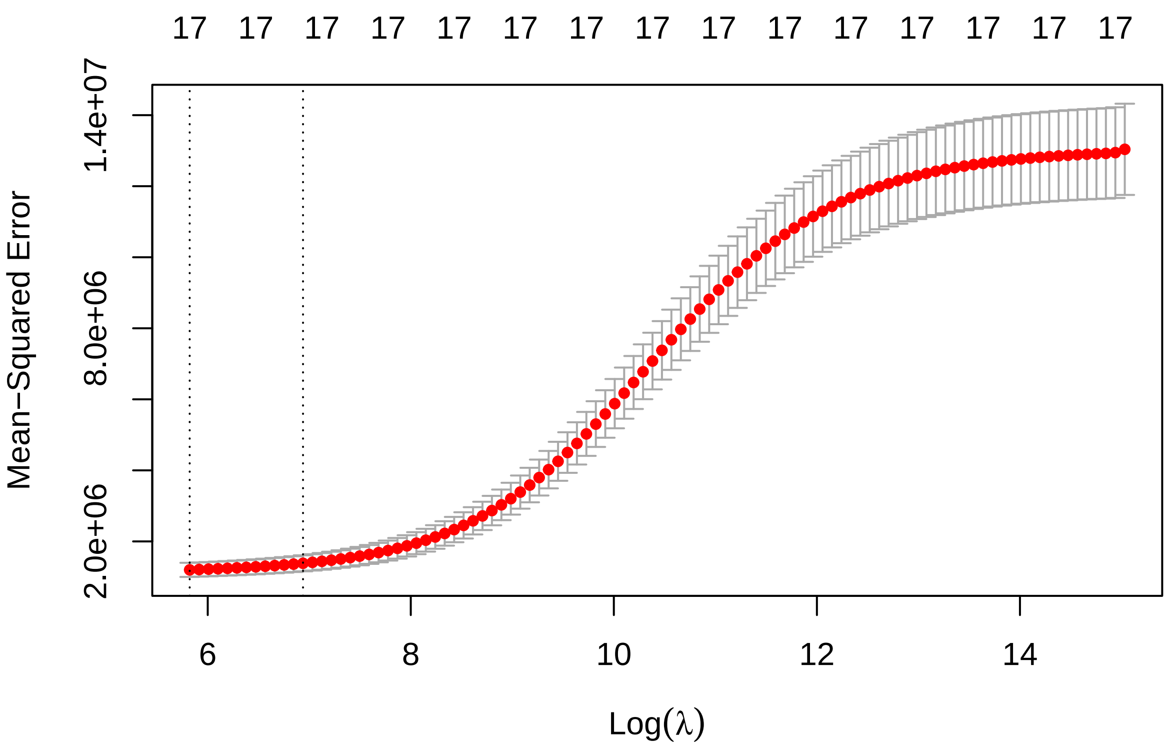
Part B - Linear Regression

```
##
## Call:
## lm(formula = Apps ~ ., data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2573.3  -423.5   -51.1   307.6  6762.0
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.471e+02  4.923e+02  -0.908 0.364325
## PrivateYes  -6.778e+02  1.762e+02  -3.847 0.000136 ***
## Accept       1.191e+00  6.572e-02  18.116  < 2e-16 ***
## Enroll      -1.281e-01  2.272e-01  -0.564 0.573054
## Top10perc    5.197e+01  7.163e+00   7.255 1.75e-12 ***
## Top25perc   -1.713e+01  5.675e+00  -3.019 0.002683 **
## F.Undergrad  7.921e-02  3.890e-02   2.036 0.042314 *
```

```
## P.Undergrad -5.039e-03  4.706e-02  -0.107 0.914772
## Outstate    -9.835e-03  2.430e-02  -0.405 0.685853
## Room.Board   1.073e-01  6.230e-02   1.722 0.085775 .
## Books        1.339e-01  3.250e-01   0.412 0.680506
## Personal     3.000e-02  7.844e-02   0.383 0.702247
## PhD         -1.036e+01  6.303e+00  -1.644 0.100868
## Terminal     1.455e+00  6.830e+00   0.213 0.831348
## S.F.Ratio    3.405e+00  1.538e+01   0.221 0.824845
## perc.alumni -7.875e+00  5.093e+00  -1.546 0.122735
## Expend       5.961e-02  1.465e-02   4.069 5.57e-05 ***
## Grad.Rate    8.848e+00  3.598e+00   2.459 0.014291 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1005 on 454 degrees of freedom
## Multiple R-squared:  0.9253, Adjusted R-squared:  0.9225
## F-statistic: 330.7 on 17 and 454 DF,  p-value: < 2.2e-16
```

```
## [1] "RMSE for Linear Regression is 1273.9406248427"
```

Part C - Ridge Regression

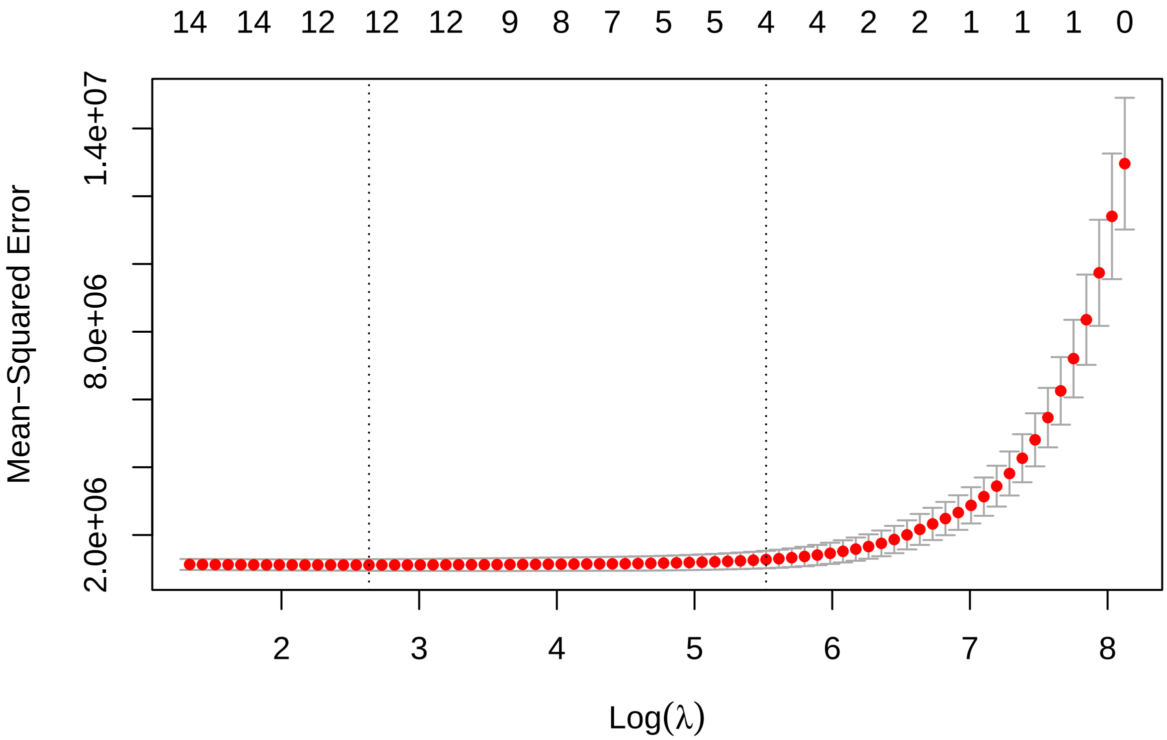



```
## [1] "Best Lambda selected by CV is 337.692874949187"
```

```
sprintf("RMSE for Ridge Regression is %s", RMSE_Ridge)
```

```
## [1] "RMSE for Ridge Regression is 1667.0344169475"
```

Part D - LASSO Regression



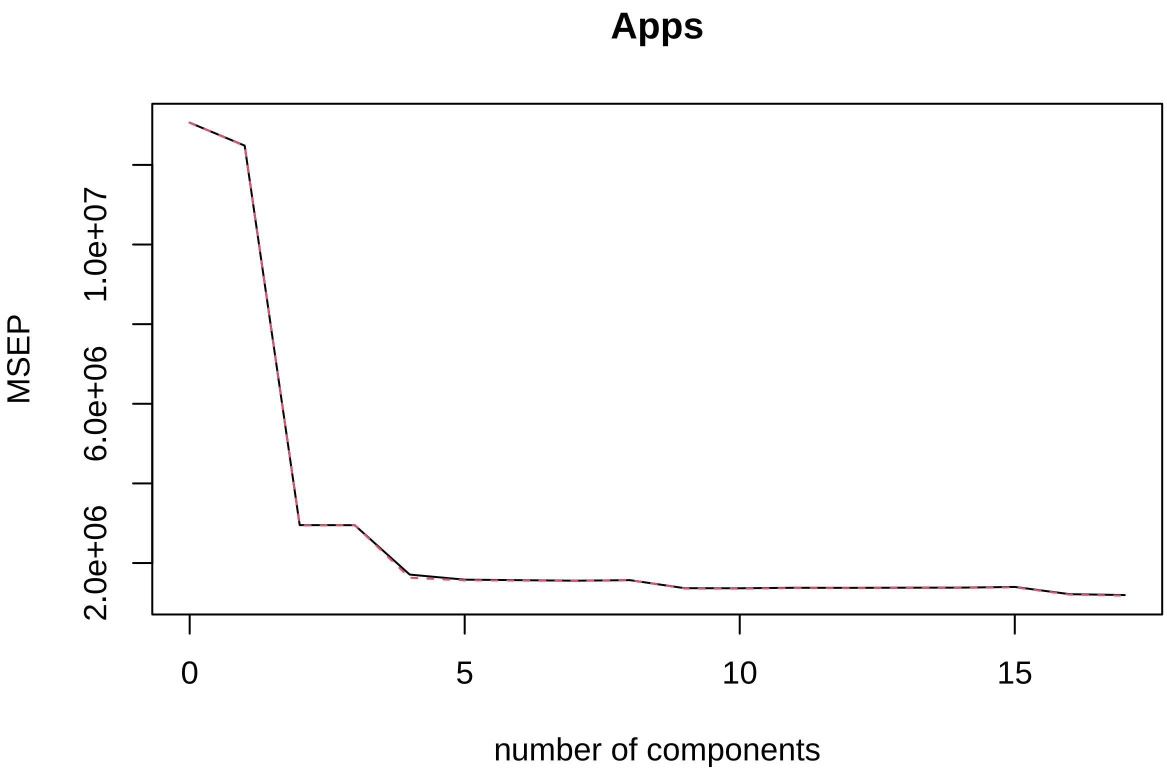

```
## [1] "Best Lambda selected by CV is 13.9535114672064"
```

```
## [1] "RMSE for LASSO Regression is 1297.04346282574 "
```

```
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                       s0
## (Intercept) -540.45364069
## PrivateYes  -625.98590071
## Accept         1.16186035
## Enroll         .
## Top10perc     42.37225836
## Top25perc     -9.51569116
## F.Undergrad    0.06679842
## P.Undergrad    .
## Outstate       .
## Room.Board     0.08326044
## Books          0.07760267
## Personal       0.01526032
## PhD           -7.21015065
## Terminal       .
## S.F.Ratio      .
## perc.alumni   -7.29632308
## Expend         0.05835291
## Grad.Rate      7.02154777
```

Result - There are 5 predictors for which coefficient is coming as 0 - ***Enroll, Outstate, Terminal, SF Ratio*** *and* ***P.Undergrad***

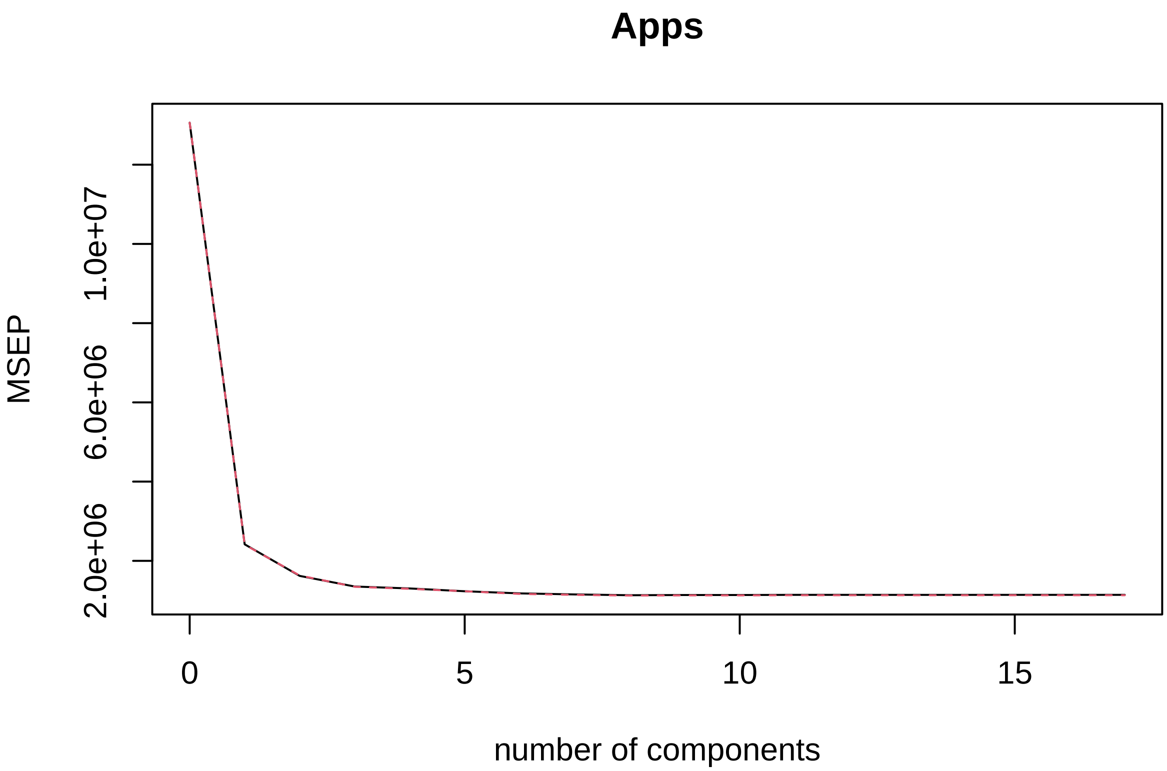
Part E - PCR



Result - As seen in the graph above, MSEP value decrease sharply till number of components=5 and post that it kind of remains constant

```
## [1] "RMSE for PCR is 1941.73579787895 "
```

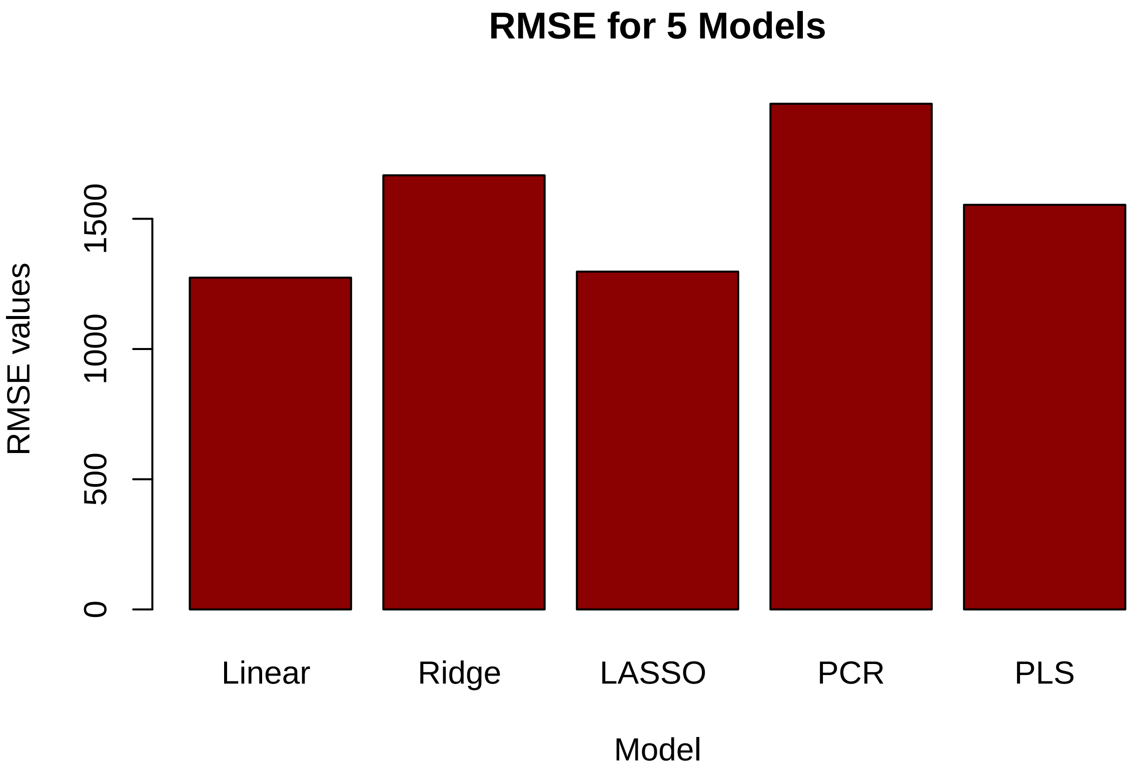
Part F - PLS



Result - As seen in the graph above, MSEP value decrease sharply till number of components=2 and post that it decreases at a slow pace till number of components=4, post that it remains kind of constant

```
## [1] "RMSE for PLS is 1553.73959417835 "
```

Part G



Result - Linear Regression, LASSO and PLS have comparable MSE value among the 5 models. PCR has the worst MSE value among all.

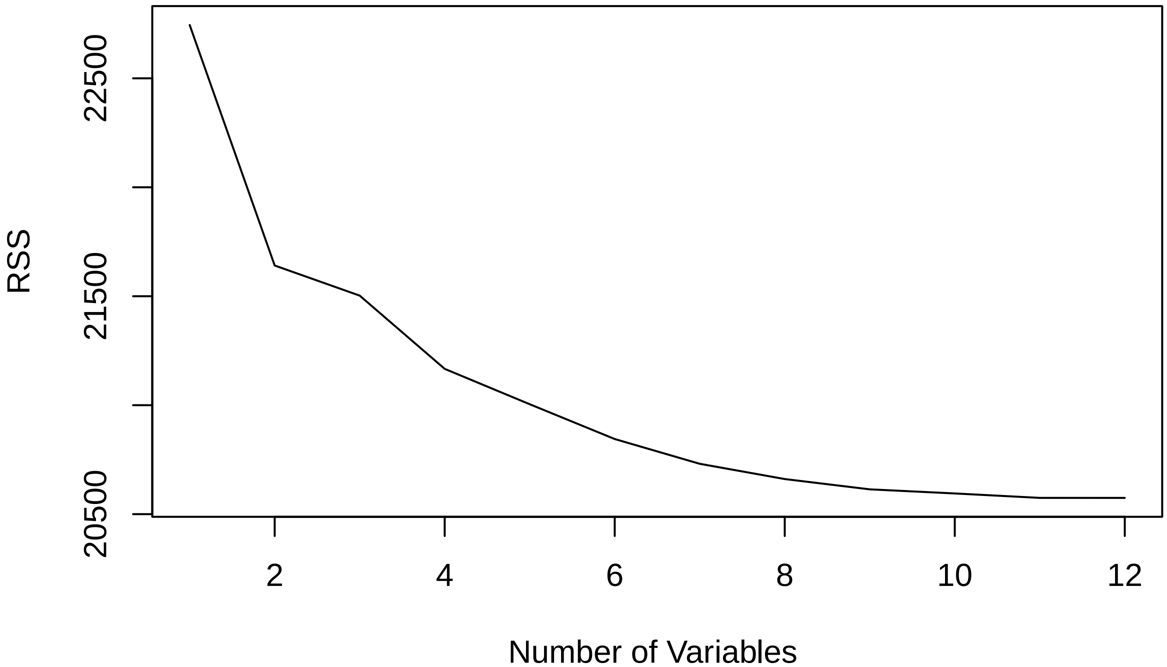
Chapter 6 : Question 11

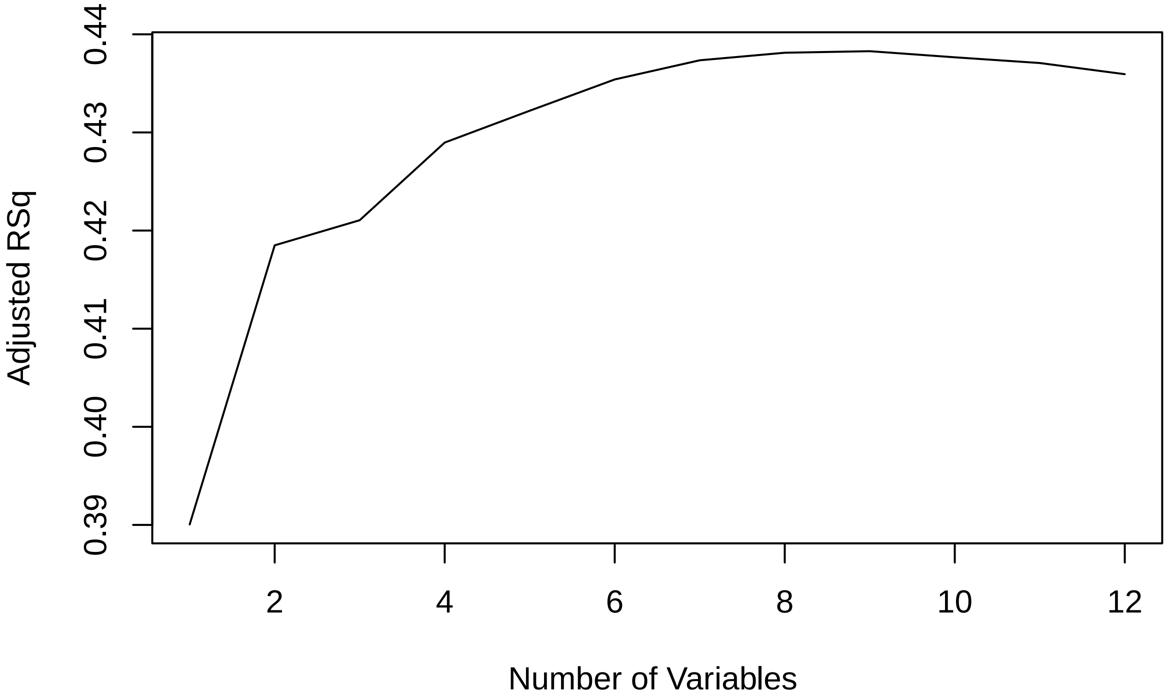
Part A

Linear Model

```
## [1] "Test RMSE obtained is: 8.15307721599405"
```

Best-Subset Selection and Linear Model

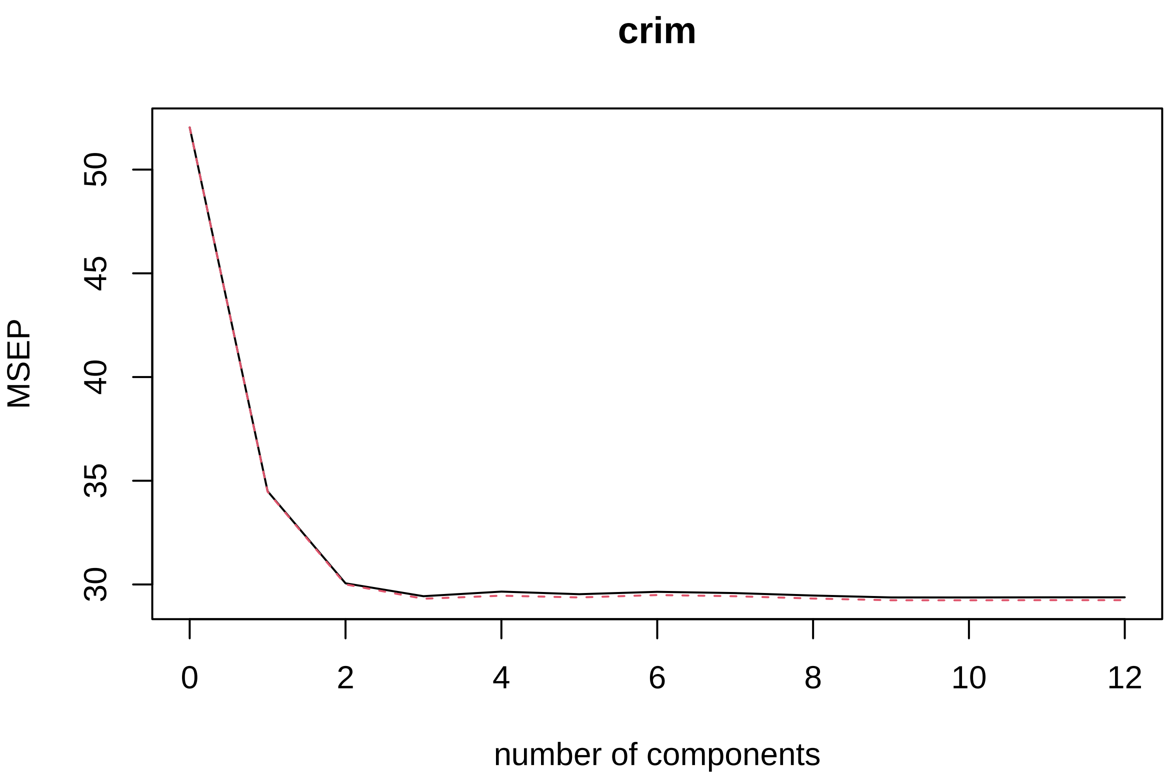





```
## [1] "RMSE obtained for Linear Regression is 8.1456518214199: "
```

Using PLS

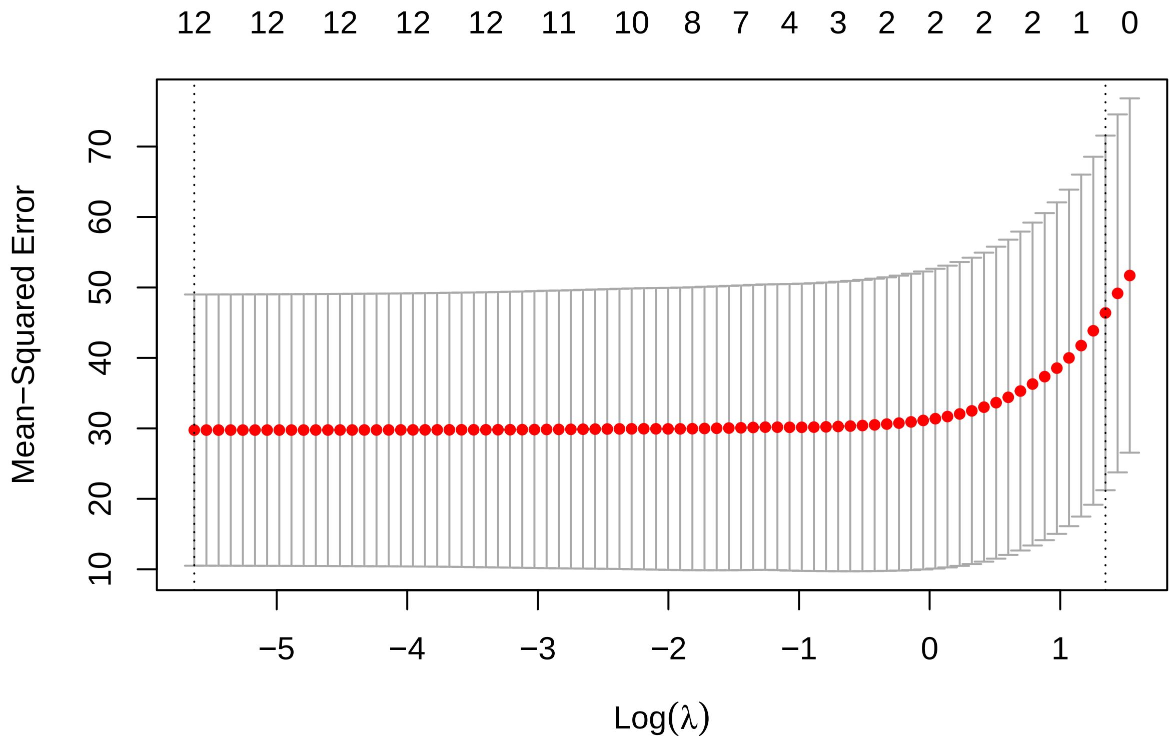
```r
validationplot(pls.fit, val.type = "MSEP")
```




```
## [1] "RMSE obtained for PLS is 8.33442546742444: "
```

Using Lasso

```
plot(boston_model_lasso)
```



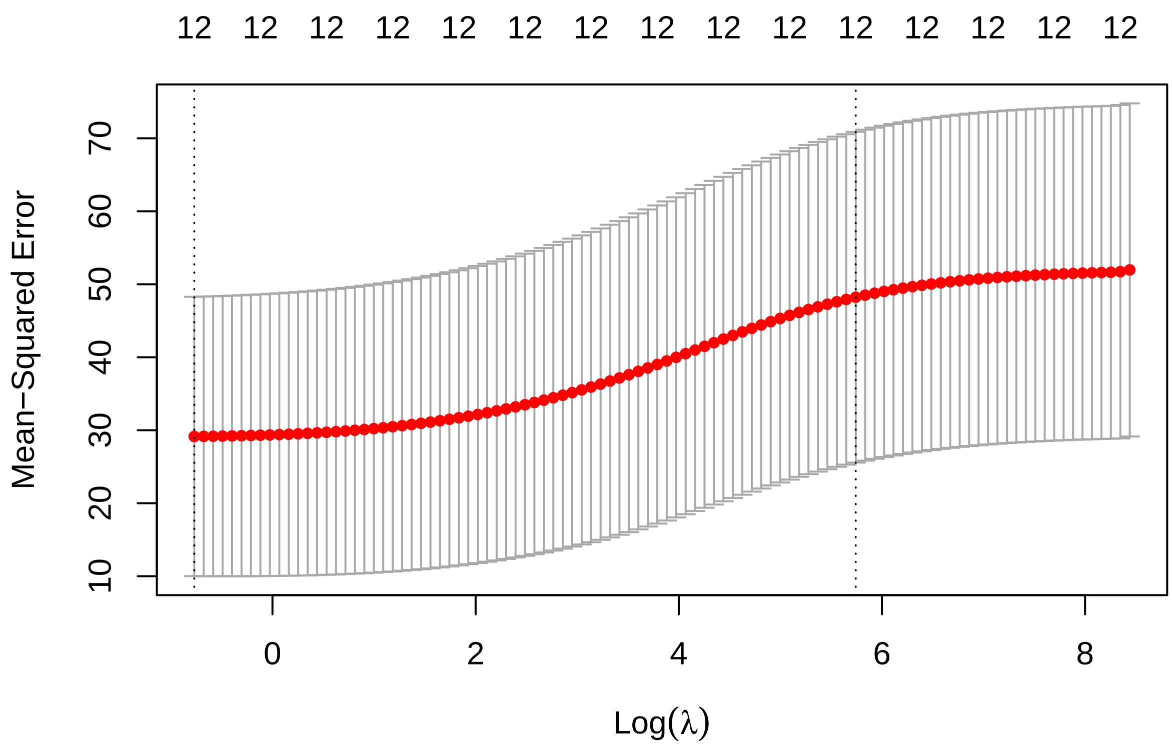

```
coef(best_model_lasso)
```

```
## 13 x 1 sparse Matrix of class "dgCMatrix"
##                        s0
## (Intercept)  3.142255121
## zn           0.035996330
## indus       -0.023003766
## chas        -0.729085243
## nox         -5.904866948
## rm           1.681578947
## age         -0.011642181
## dis         -0.794734130
## rad          0.486534802
## tax         -0.002825719
## ptratio     -0.256074757
## lstat        0.186977037
## medv        -0.210835758
```

```
## [1] "RMSE obtained for Lasso is 8.15462985261151: "
```

Using Ridge

```
plot(boston_model_ridge)
```



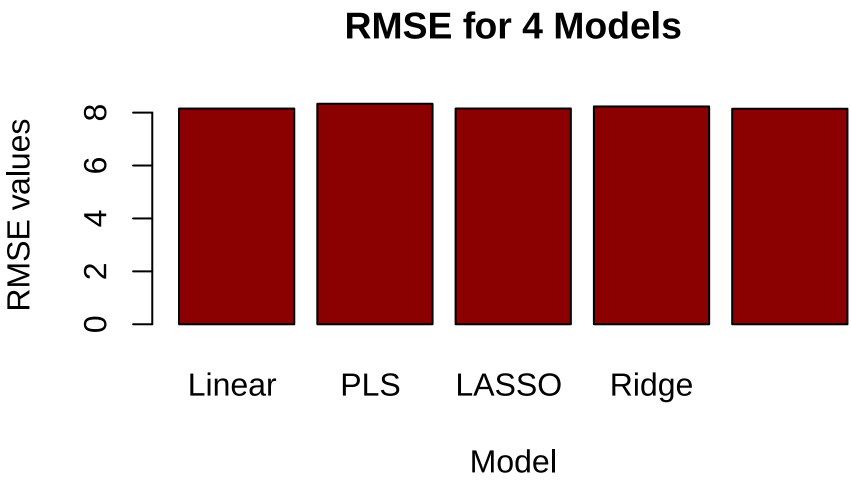

```r
coef(boston_model_ridge)
```

```
## 13 x 1 sparse Matrix of class "dgCMatrix"
##                       s1
## (Intercept)  2.2525509770
## zn          -0.0011709195
## indus        0.0086523930
## chas        -0.0361339197
## nox          0.5131566033
## rm          -0.0292287196
## age          0.0018230371
## dis         -0.0258605341
## rad          0.0112876851
## tax          0.0005381042
## ptratio      0.0202342289
## lstat        0.0100357407
## medv        -0.0060788077
```

```r
sprintf('RMSE obtained for Ridge is %s: ',rms_error_ridge)
```

```
## [1] "RMSE obtained for Ridge is 8.23203441189903: "
```

Part B



Part C

Linear Regression has the lowest RMSE value among the 4 models, Also I have tried Running Linear Regression with Best Subset Selection. Even though RMSE is pretty relatable with and without Best Subset Selection, selecting lesser number of variables makes the model easier to fit and less complex.

Linear Regression with Best Subset Selection is the best fitted model

Chapter 8 : Question 8

Part A - Splitting the Dataset

```
## [1] "Two Dimension of Train is 287" "Two Dimension of Train is 11"
```

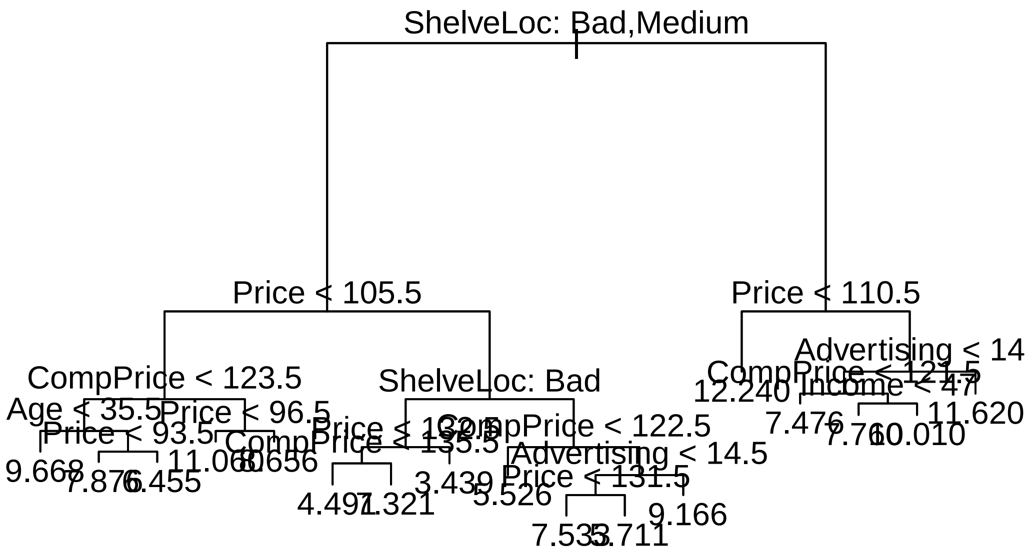
```
## [1] "Two Dimension of Test is 113" "Two Dimension of Test is 11"
```

Part B - Trees

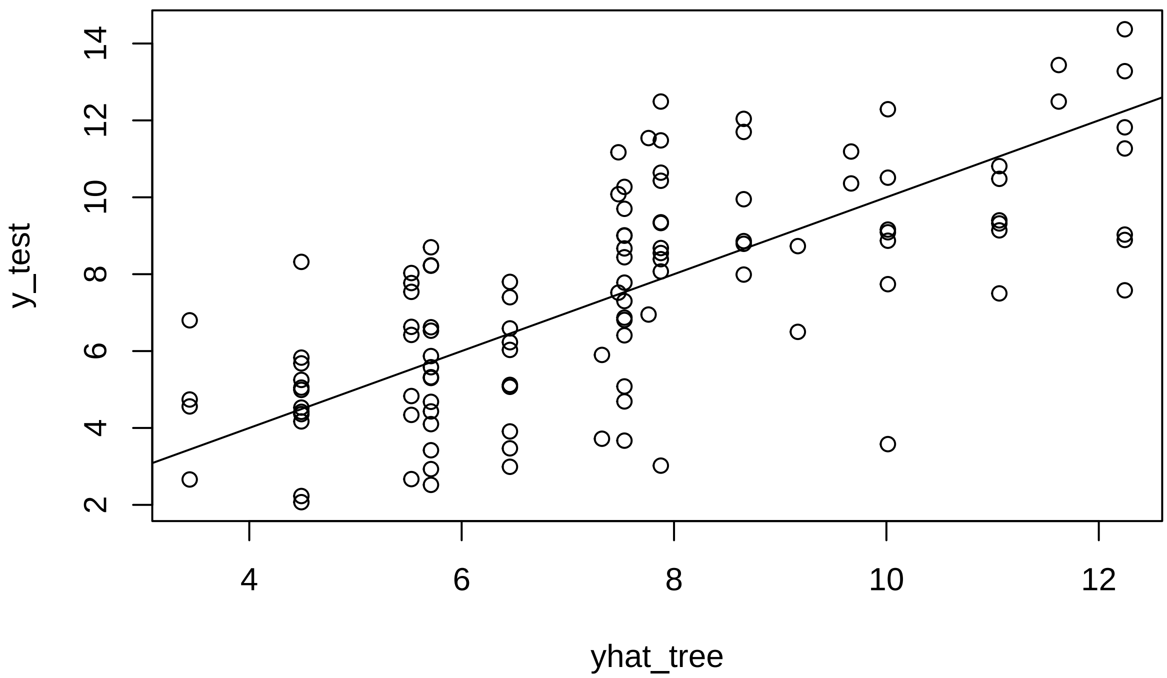
```
summary(tree.Carseats)
```

```
##
## Regression tree:
## tree(formula = Sales ~ ., data = train)
## Variables actually used in tree construction:
## [1] "ShelveLoc"   "Price"       "CompPrice"   "Age"         "Advertising"
## [6] "Income"
## Number of terminal nodes:  17
## Residual mean deviance:  2.533 = 683.8 / 270
## Distribution of residuals:
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -5.711000 -1.013000  0.006667  0.000000  1.099000  4.025000
```

```
plot(tree.Carseats)
text(tree.Carseats , pretty = 0)
```



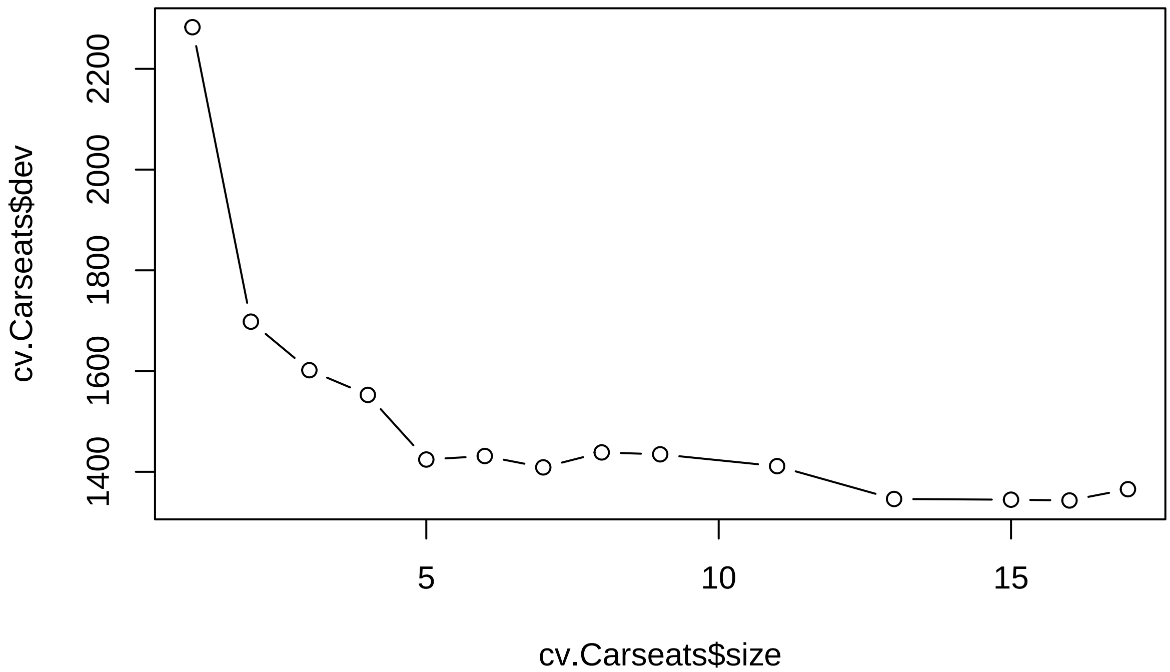
Result - As visible from the plot, shelve location is the most important predictor, followed by price.




```
## [1] "MSE value for the tree is : 4.43361158699893"
```

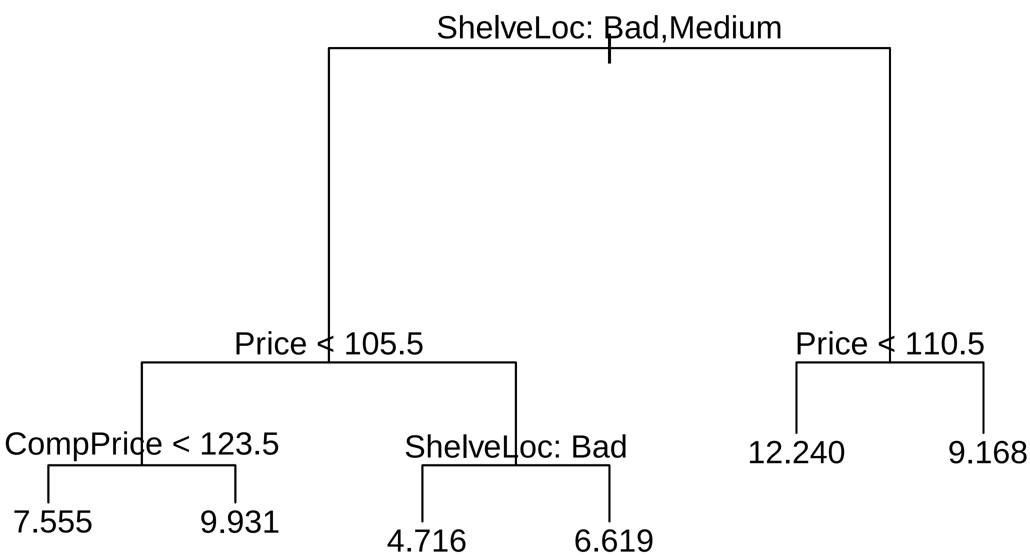
Part C - Pruned Trees

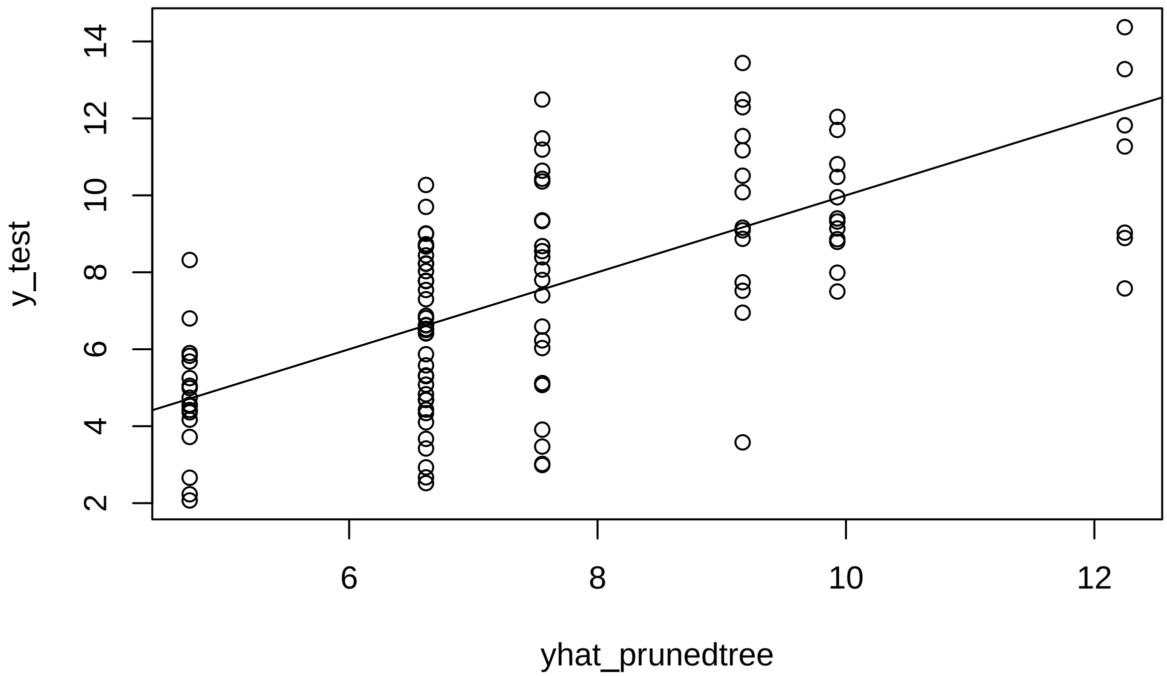
```
plot(cv.Carseats$size, cv.Carseats$dev, type = "b")
```



Result - As per Cross Validation, the optimal level of tree complexity is: 6

```
plot(prune.Carseats)
text(prune.Carseats , pretty = 0)
```



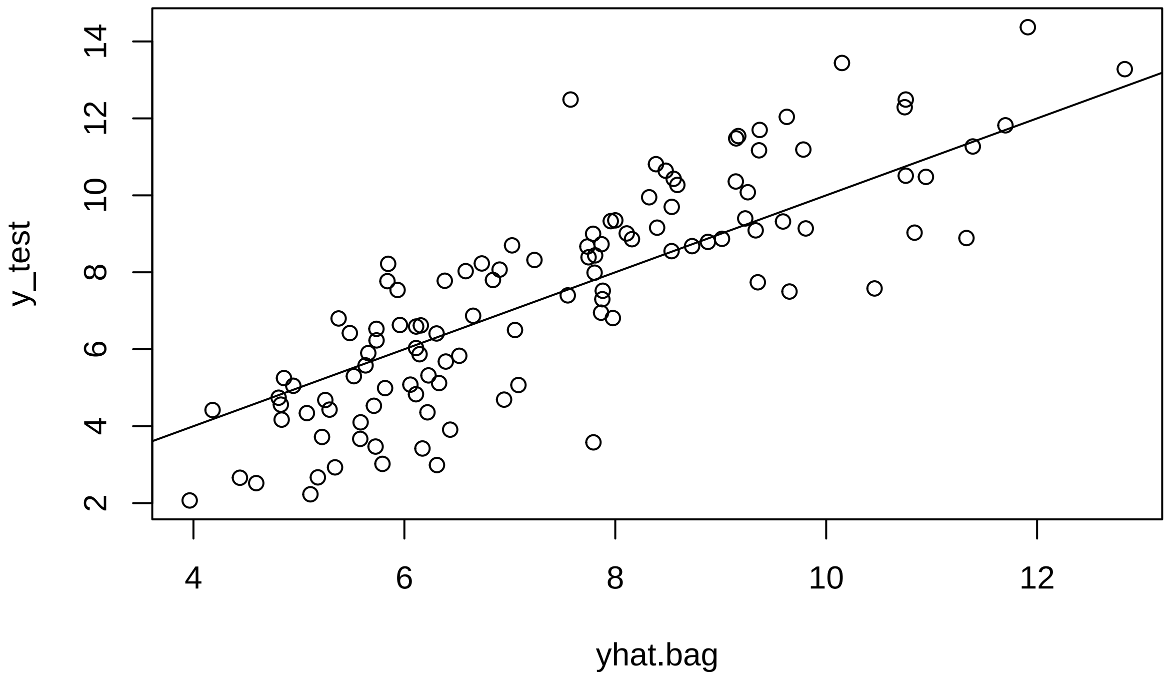



```
## [1] "MSE value for the pruned tree is : 4.80354782996407"
```

Result - No, Pruning the tree doesn't help in improving MSE

Part D - Bagging

```
##
## Call:
##  randomForest(formula = Sales ~ ., data = train, mtry = 10, importance = TRUE)
##                Type of random forest: regression
##                      Number of trees: 500
## No. of variables tried at each split: 10
##
##           Mean of squared residuals: 2.683038
##                     % Var explained: 66.14
```

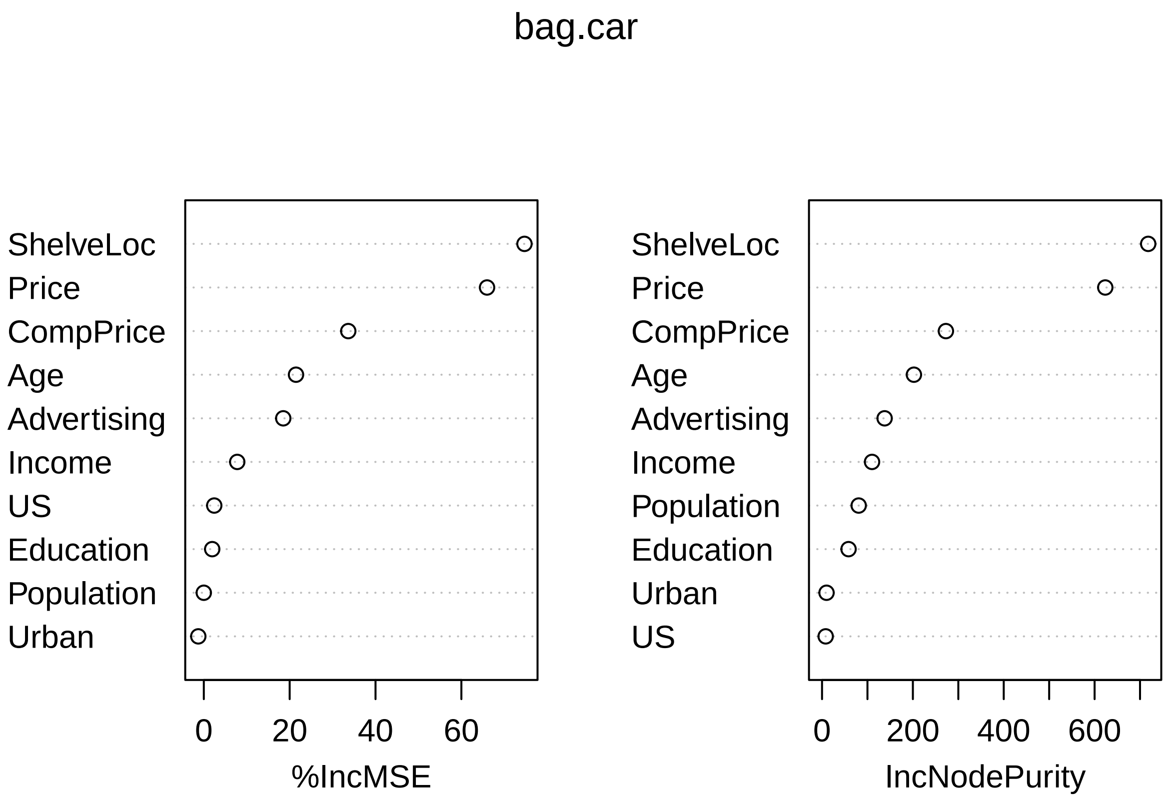



```
## [1] "MSE value for the Bagging is : 2.50293807640954"
```

```
importance(bag.car)
```

```
##                   %IncMSE IncNodePurity
## CompPrice   33.649077313    272.650228
## Income       7.785862891    110.066219
## Advertising 18.504590806    137.790166
## Population  -0.006556997     80.739802
## Price       65.977970064    623.462554
## ShelveLoc   74.701306945    718.042390
## Age         21.474976984    202.289725
## Education    1.955852867     58.387446
## Urban       -1.290220168      9.927368
## US           2.421884025      8.056529
```

```
varImpPlot(bag.car)
```



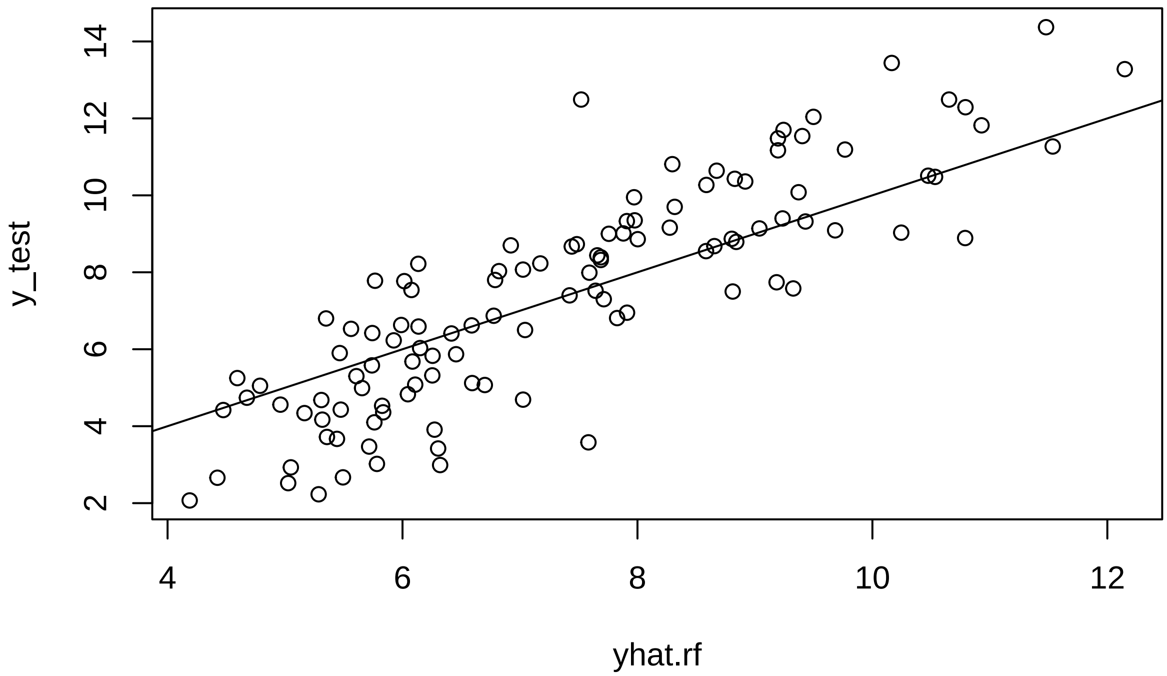

```
varImp(bag.car)
```

```
##                  Overall
## CompPrice   33.649077313
## Income       7.785862891
## Advertising 18.504590806
## Population  -0.006556997
## Price       65.977970064
## ShelveLoc   74.701306945
## Age         21.474976984
## Education    1.955852867
## Urban       -1.290220168
## US           2.421884025
```

Result - ShelveLoc and Price are two most important predictors for Sales

Part E - Random Forest

```
##
## Call:
##  randomForest(formula = Sales ~ ., data = train, mtry = 5, importance = TRUE)
##                Type of random forest: regression
##                      Number of trees: 500
## No. of variables tried at each split: 5
##
##           Mean of squared residuals: 2.678598
##                     % Var explained: 66.19
```

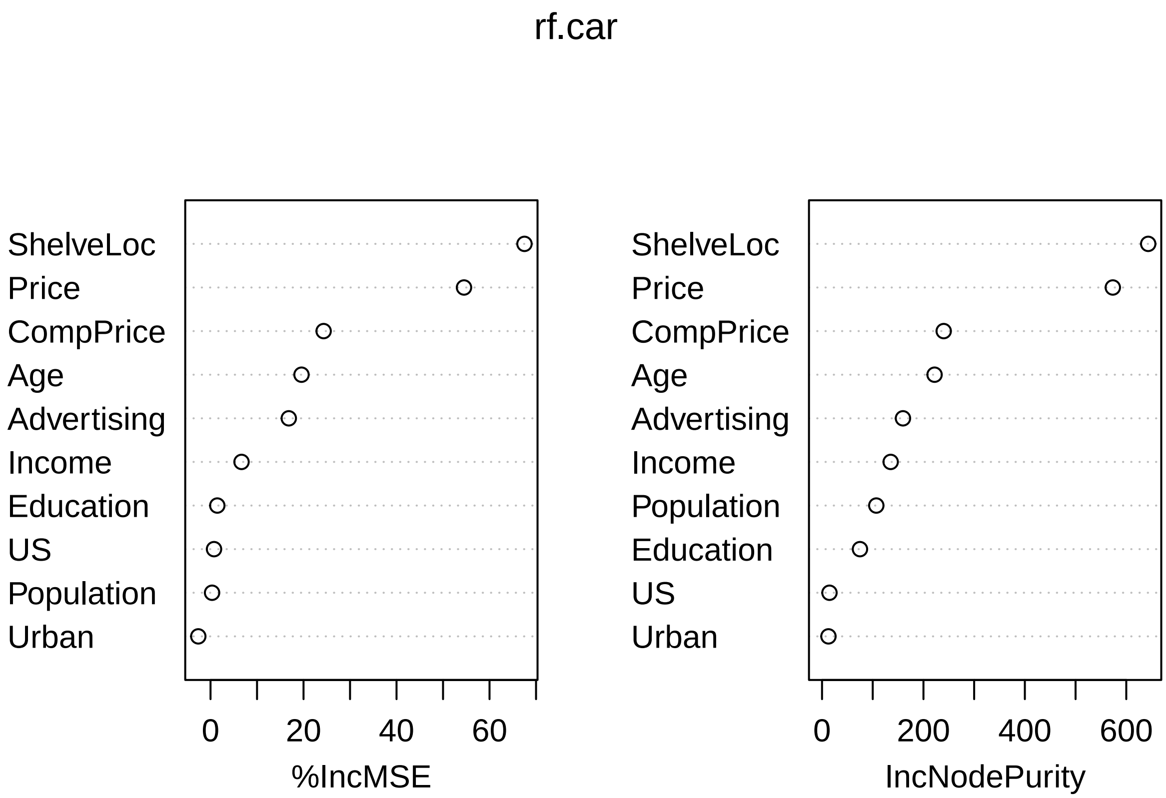



```
## [1] "MSE value for the Random Forest is : 2.46285896081955"
```

```
importance(rf.car)
```

```
##                %IncMSE IncNodePurity
## CompPrice   24.3199357     239.99870
## Income       6.6658448     135.37866
## Advertising 16.8282026     159.62205
## Population   0.3304106     107.04200
## Price       54.5074226     573.42174
## ShelveLoc   67.5158408     643.27739
## Age         19.5615233     221.95767
## Education    1.4521616      74.83801
## Urban       -2.6348456      12.70497
## US           0.7494543      14.68773
```

```
varImpPlot(rf.car)
```

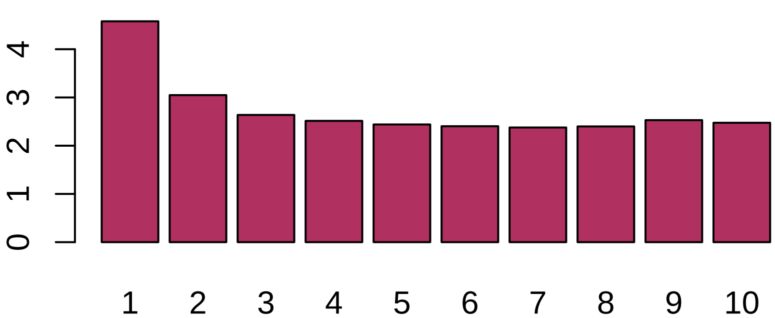



```
varImp(rf.car)
```

```
##                Overall
## CompPrice   24.3199357
## Income       6.6658448
## Advertising 16.8282026
## Population   0.3304106
## Price       54.5074226
## ShelveLoc   67.5158408
## Age         19.5615233
## Education    1.4521616
## Urban       -2.6348456
## US           0.7494543
```

Result - ShelveLoc and Price are two most import predictors for Sales

Relationship of M in random Forest



Part F - BART

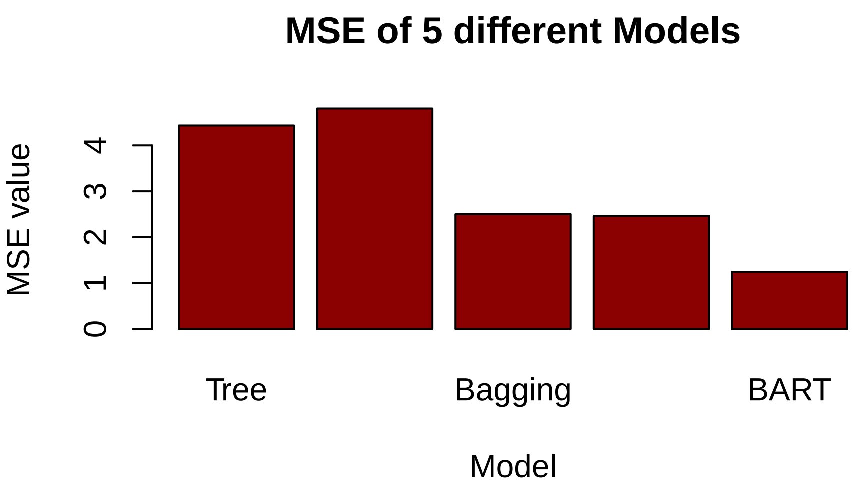
```
## [1] "MSE value for the BART is : 1.24592486738236"
```

```
ord <- order(bartfit$varcount.mean, decreasing = T)
bartfit$varcount.mean[ord]
```

```
##          Price       CompPrice ShelveLocMedium   ShelveLocGood        UrbanYes
##         27.393          24.004          23.896          22.644          21.217
##      Education             Age          Income           USYes      Population
##         21.164          20.884          19.984          19.681          19.307
##    Advertising
##         17.746
```

Result -In the above output, we can check how many times each variable appeared in the collection of trees. Price and ShelveLoc are the predictors which have occurred most number of times

Plotting MSE values for each Model



Result - BART has lowest MSE among the 5 models we used in this problem. Pruned Tree has the highest with ~5 MSE where as BART has the least

Chapter 8 : Question 11

Part A

```
dim(Caravan_train)
```

```
## [1] 1000   86
```

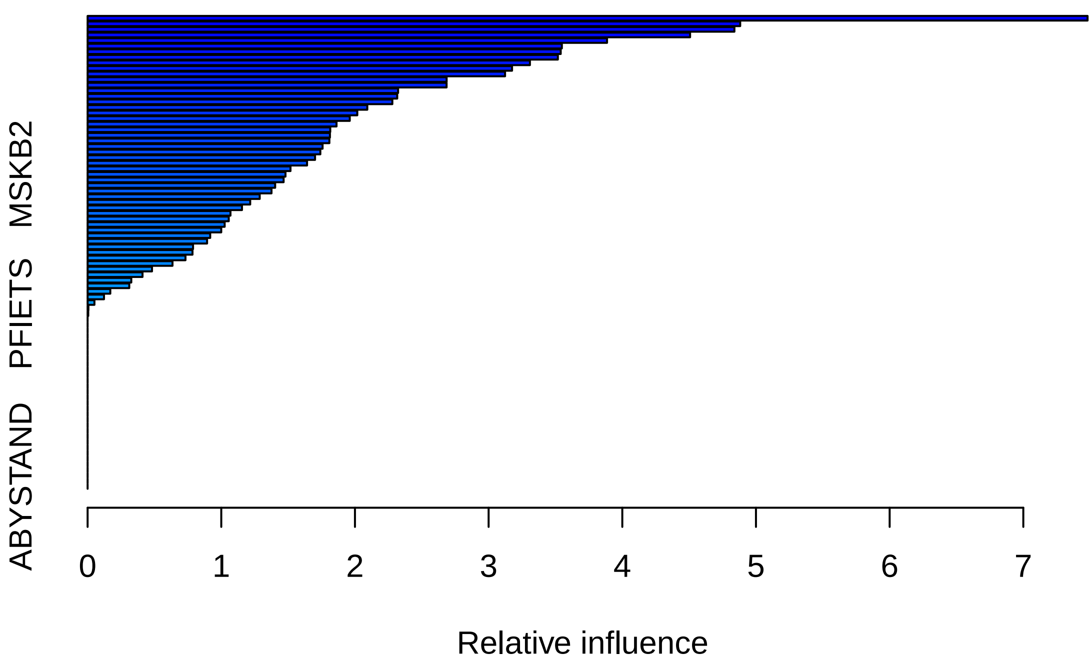
```
dim(Caravan_test)
```

```
## [1] 4822   86
```

Result - There are 1000 records in Caravan_train and remaining 4822 records in Caravan_test

Part B

```
boost.Caravan <- gbm(Purchase ~ ., data = Caravan_train,distribution = "bernoulli", n.trees = 1000, inte
summary(boost.Caravan)
```




```
##                 var     rel.inf
## PPERSAUT PPERSAUT 7.480819014
## MOPLHOOG MOPLHOOG 4.882054338
## MGODGE     MGODGE 4.838869962
## MKOOPKLA MKOOPKLA 4.507280400
## MOSTYPE   MOSTYPE 3.886043943
## MGODPR     MGODPR 3.547892360
## PBRAND     PBRAND 3.539487907
## MBERMIDD MBERMIDD 3.518082698
## MBERARBG MBERARBG 3.309004843
## MINK3045 MINK3045 3.175313873
## MSKC         MSKC 3.123008472
## MSKA         MSKA 2.685844523
## MAUT2       MAUT2 2.685548007
## MAUT1       MAUT1 2.322786246
## PWAPART   PWAPART 2.316252267
## MSKB1       MSKB1 2.279820190
## MRELOV     MRELOV 2.092410309
## MFWEKIND MFWEKIND 2.017651081
## MBERHOOG MBERHOOG 1.961378700
## MBERARBO MBERARBO 1.862074416
## MRELGE     MRELGE 1.815276446
## MINK7512 MINK7512 1.812894054
## MINKM30   MINKM30 1.808781053
## MOPLMIDD MOPLMIDD 1.757784665
## MFGEKIND MFGEKIND 1.741172971
```

```
## MGODOV     MGODOV 1.701539077
## MZFONDS   MZFONDS 1.641658796
## MFALLEEN MFALLEEN 1.517763739
## MSKB2       MSKB2 1.480397941
## MINK4575 MINK4575 1.466410983
## MAUT0       MAUT0 1.403097259
## ABRAND     ABRAND 1.375696683
## MHHUUR     MHHUUR 1.287672857
## MINKGEM   MINKGEM 1.216351643
## MHKOOP     MHKOOP 1.154970948
## MGEMLEEF MGEMLEEF 1.068800262
## MGODRK     MGODRK 1.056066524
## MRELSA     MRELSA 1.025383382
## MZPART     MZPART 0.999705745
## MSKD         MSKD 0.917077921
## MGEMOMV   MGEMOMV 0.893757812
## MBERZELF MBERZELF 0.788935429
## APERSAUT APERSAUT 0.784652995
## MOPLLAAG MOPLLAAG 0.732210597
## MOSHOOFD MOSHOOFD 0.634998065
## PMOTSCO   PMOTSCO 0.481824116
## PLEVEN     PLEVEN 0.410808274
## PBYSTAND PBYSTAND 0.326851643
## MBERBOER MBERBOER 0.311571820
## MINK123M MINK123M 0.169710044
## MAANTHUI MAANTHUI 0.122660387
## ALEVEN     ALEVEN 0.051158218
## PAANHANG PAANHANG 0.006040057
## PFIETS     PFIETS 0.004694048
## PWABEDR   PWABEDR 0.000000000
## PWALAND   PWALAND 0.000000000
## PBESAUT   PBESAUT 0.000000000
## PVRAAUT   PVRAAUT 0.000000000
## PTRACTOR PTRACTOR 0.000000000
## PWERKT     PWERKT 0.000000000
## PBROM       PBROM 0.000000000
## PPERSONG PPERSONG 0.000000000
## PGEZONG   PGEZONG 0.000000000
## PWAOREG   PWAOREG 0.000000000
## PZEILPL   PZEILPL 0.000000000
## PPLEZIER PPLEZIER 0.000000000
## PINBOED   PINBOED 0.000000000
## AWAPART   AWAPART 0.000000000
## AWABEDR   AWABEDR 0.000000000
## AWALAND   AWALAND 0.000000000
## ABESAUT   ABESAUT 0.000000000
## AMOTSCO   AMOTSCO 0.000000000
## AVRAAUT   AVRAAUT 0.000000000
## AAANHANG AAANHANG 0.000000000
## ATRACTOR ATRACTOR 0.000000000
## AWERKT     AWERKT 0.000000000
## ABROM       ABROM 0.000000000
## APERSONG APERSONG 0.000000000
## AGEZONG   AGEZONG 0.000000000
```

```
## AWAOREG   AWAOREG 0.000000000
## AZEILPL   AZEILPL 0.000000000
## APLEZIER APLEZIER 0.000000000
## AFIETS     AFIETS 0.000000000
## AINBOED   AINBOED 0.000000000
## ABYSTAND ABYSTAND 0.000000000
```

Result - *PPERSAUT* seems to be the most importance feature with relative inference of 7.782, followed by ***MGODGE*** and ***PBRAND***

Part C

```
yhat.boost <- predict(boost.Caravan ,newdata = Caravan_test, n.trees = 1000, type = 'response')
yhat.boost_prediction<- ifelse(yhat.boost>0.2,1,0)
Cm_boost<-table(yhat.boost_prediction,y_test)
Cm_boost
```

Apply Boosting Modelling Technique to the train dataset

```
##                       y_test
## yhat.boost_prediction    0    1
##                      0 4336  258
##                      1  197   31
```

```
Precision_boost=(Cm_boost[2,2]/(Cm_boost[2,1]+Cm_boost[2,2]))*100
sprintf('Percentage of the people predicted by Boosting to make a purchase do in fact make one: %s', Pr
```

```
## [1] "Percentage of the people predicted by Boosting to make a purchase do in fact make one: 13.59649
```

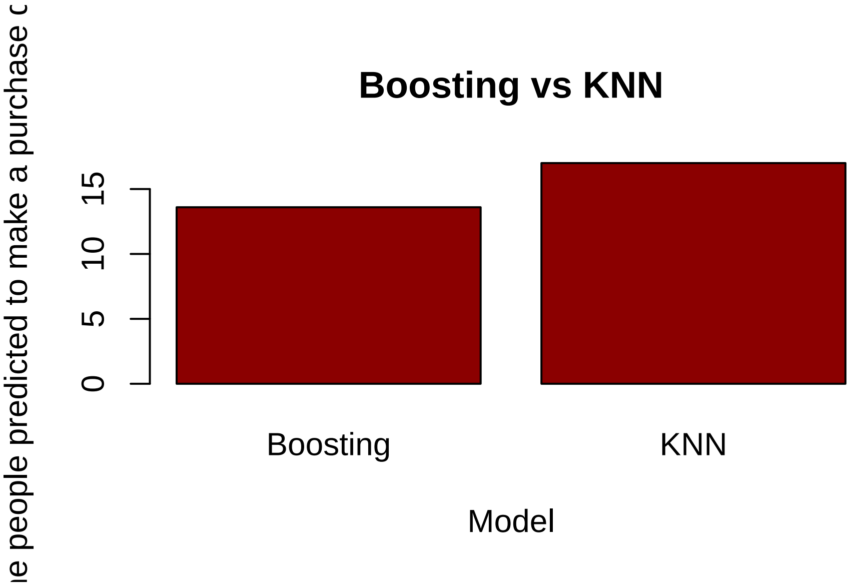
```
Cm_KNN=table(knn.pred, test.Y)
Cm_KNN
```

Applying KNN Modelling Technique to the dataset

```
##          test.Y
## knn.pred   No  Yes
##      No  4450  272
##      Yes   83   17
```

```
Precision_KNN=(Cm_KNN[2,2]/(Cm_KNN[2,1]+Cm_KNN[2,2]))*100
sprintf('Percentage of the people predicted by KNN to make a purchase do in fact make one: %s', Precisi
```

```
## [1] "Percentage of the people predicted by KNN to make a purchase do in fact make one: 17"
```

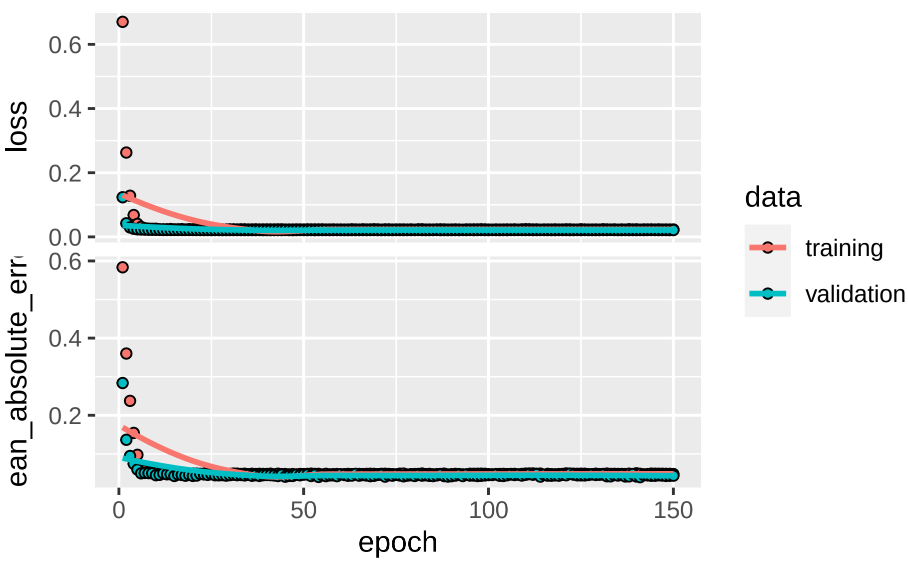


Result - Percentage of the people predicted by Boosting to make a purchase do in fact make one is ~15 % where as Percentage of the people predicted by KNN to make a purchase do in fact make one is ~17 %

Chapter 10 : Question 7

```
plot(predict_default)
```

Applying Neural Net of 1 hidden Layer with 10 Neurons




```
npred <- predict(modnn , x_test)
mean(abs(y_test - npred))
```

```
## [1] 0.0435226
```

```
confusionMatrix(y_test, npred)
```

```
##      0  1
## 0 1449 33
## 1    5 13
```

Result - Applied the NN with one hidden layer and 10 Neurons with 150 Epochs and Batch Size = 150. As seen, RMSE value is coming as 0.046, i.e 96% of the data is being correctly classified. One point to note here is class is highly imbalanced , which is leading the model to predict mostly 0s, which can be seen in the confusion matrix above. As seen, Model is predicting correct 0 - 1449 times and predicting correct 1 - only 6 times

Applying Logistic Regression

```
##
## Call:
## glm(formula = default ~ ., family = "binomial", data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4524  -0.1442  -0.0564  -0.0208   3.7392
##
## Coefficients:
##                  Estimate    Std. Error z value            Pr(>|z|)
## (Intercept) -10.811193172   0.529082055 -20.434 <0.0000000000000002 ***
## student1     -0.628448738   0.251855031  -2.495              0.0126 *
## balance       0.005716091   0.000248191  23.031 <0.0000000000000002 ***
## income        0.000001996   0.000008821   0.226              0.8210
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2526.0  on 8550  degrees of freedom
## Residual deviance: 1363.3  on 8547  degrees of freedom
## AIC: 1371.3
##
## Number of Fisher Scoring iterations: 8
```

```
pscl::pR2(mylogit)["McFadden"]
```

Accessing Model fit for Logestic Regression

```
## fitting null model for pseudo-r2
```

```
##  McFadden
## 0.4602969
```

Result - Value of **0.4728807** is quite high for McFadden's R^2, which indicates that our model fits the data very well and has high predictive power.

```
caret::varImp(mylogit)
```

```
##             Overall
## student1  2.4952797
## balance  23.0310013
## income    0.2262322
```

Result - Higher values indicate more importance. These results match up nicely with the p-values from the model.

```
#find optimal cutoff probability to use to maximize accuracy
optimalCutoff(y_test, y_predicted)[1]
```

```
## [1] 0.4376094
```

Result - Any individual with a probability of defaulting of 0.437 or higher will be predicted to default, while any individual with a probability less than this number will be predicted to not default.

```
confusionMatrix(y_test, y_predicted)
```

```
##      0   1
## 0 9627 228
## 1   39 105
```

Extra Questions

Problem 1 : Beauty Pays

Part A

```
data <- read.csv("BeautyData.csv")
lm.fit <- lm(formula = data$CourseEvals~., data=data)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = data$CourseEvals ~ ., data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.31385 -0.30202  0.01011  0.29815  1.04929
##
## Coefficients:
##             Estimate Std. Error t value             Pr(>|t|)
## (Intercept)  4.06542    0.05145  79.020 < 0.0000000000000002 ***
```

```
## BeautyScore  0.30415    0.02543  11.959 < 0.0000000000000002 ***
## female      -0.33199    0.04075  -8.146  0.00000000000000362 ***
## lower       -0.34255    0.04282  -7.999  0.00000000000001038 ***
## nonenglish  -0.25808    0.08478  -3.044              0.00247 **
## tenuretrack -0.09945    0.04888  -2.035              0.04245 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4273 on 457 degrees of freedom
## Multiple R-squared:  0.3471, Adjusted R-squared:  0.3399
## F-statistic: 48.58 on 5 and 457 DF,  p-value: < 0.00000000000000022
```

```
lm.fit <- lm(formula = data$CourseEvals~data$BeautyScore, data=data)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = data$CourseEvals ~ data$BeautyScore, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.5936 -0.3346  0.0097  0.3702  1.2321
##
## Coefficients:
##                  Estimate Std. Error t value            Pr(>|t|)
## (Intercept)       3.71340    0.02249 165.119 <0.0000000000000002 ***
## data$BeautyScore  0.27148    0.02837   9.569 <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4809 on 461 degrees of freedom
## Multiple R-squared:  0.1657, Adjusted R-squared:  0.1639
## F-statistic: 91.57 on 1 and 461 DF,  p-value: < 0.00000000000000022
```

Result - According to the stated linear regression results, BeautyScore and CourseEvals have a positive connection with a statistically significant coefficient when we attempt to predict course ratings using all the characteristics. This indicates that beauty has a direct favorable effect on CourseEvals while holding the other factors, or in this case "other determinants," constant.

Part B

Dr. Hamermesh is pointing out, in my opinion, that it is very difficult or impossible to ***isolate the impact of beauty*** on students' perceptions of teachers. In other words, as a student will always be exposed to see a teacher's appearance when being taught in person, ***it is challenging to control for the unconscious bias linked with the same***. Perhaps we can control for other variables if we run an experiment in which pupils aren't shown a teacher's face. But once more, we are unable to separate the influence of voice quality from any potential unconscious bias, making it "probably impossible" to resolve this problem, in Dr. Hamermesh's words.

Problem 2 : Housing Price Structure

Part A and B

```
lm.fit <- lm(formula = MidCity$Price~., data=MidCity)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = MidCity$Price ~ ., data = MidCity)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -27897.8  -6074.8    -48.7   5551.8  27536.4
##
## Coefficients:
##                Estimate Std. Error t value           Pr(>|t|)
## (Intercept)     308.114   9605.692   0.032           0.974465
## Home            -11.456     25.387  -0.451           0.652616
## Offers        -8350.128   1103.693  -7.566 0.0000000000089599 ***
## SqFt             53.634      5.926   9.051 0.0000000000000033 ***
## Bedrooms       4136.461   1621.775   2.551           0.012023 *
## Bathrooms      7975.157   2133.831   3.737           0.000287 ***
## Neigborhood1   1729.613   2433.756   0.711           0.478675
## Neigborhood3  22264.319   2540.699   8.763 0.0000000000000156 ***
## Brick_Param   17313.540   1988.548   8.707 0.0000000000000212 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10050 on 119 degrees of freedom
## Multiple R-squared:  0.8688, Adjusted R-squared:   0.86
## F-statistic: 98.54 on 8 and 119 DF,  p-value: < 0.00000000000000022
```

Part A| Result- As we can see from the results of the linear regression, "BrickParam" has a positive association with the price of the house and a statistically significant positive coefficient. This implies that if the house is a brick house, its price would be higher even if all other attributes remained the same.

Part B| Result - Fitted the linear regression model after one-hot encoding the Nbhd column. According to the model summary, a house has a positive association with a house if it is located in neighborhood 3 and the correlation is statistically significant. This suggests that, assuming all other features were the same, a home in neighborhood 3 would cost, on average, $22,264 more than a home in a neighborhood without neighborhood 3.

Part C

```
MidCity$Neigborhood3_Brick<-MidCity$Neigborhood3*MidCity$Brick_Param
MidCity<-MidCity[,-8]
lm.fit <- lm(formula = MidCity$Price~., data=MidCity)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = MidCity$Price ~ ., data = MidCity)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -27036.5  -8206.1    677.8   6394.4  29212.7
##
## Coefficients:
##                      Estimate Std. Error t value            Pr(>|t|)
## (Intercept)         -2089.243  10651.442  -0.196            0.844830
## Home                   -9.927     28.148  -0.353            0.724962
## Offers             -10500.101   1162.181  -9.035 0.00000000000000360 ***
## SqFt                   58.686      6.546   8.966 0.00000000000000523 ***
## Bedrooms             8025.337   1686.824   4.758 0.00000555387937520 ***
## Bathrooms            5513.238   2447.619   2.252            0.026124 *
## Neigborhood1        -2496.563   2652.232  -0.941            0.348455
## Brick_Param          8714.012   2545.952   3.423            0.000851 ***
## Neigborhood3_Brick  24424.734   3849.969   6.344 0.00000000420250101 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11150 on 119 degrees of freedom
## Multiple R-squared:  0.8387, Adjusted R-squared:  0.8279
## F-statistic: 77.37 on 8 and 119 DF,  p-value: < 0.00000000000000022
```

Result - Fitted the linear regression model after creating interaction term Neighborhood3_Brick which is a multiplication of Neighborhood3 and Brick_Param, essentially a flag for houses which are brick and are in neighborhood 3. According to the model summary, a house has a positive association with a Brick house if it is located in neighborhood 3 and the correlation is statistically significant. This suggests that, assuming all other features were the same, a home in neighborhood 3 would cost, on average, $24,424.734 more than a Non-Brick home in a neighborhood without neighborhood 3.

Part D

```r
MidCity_Actual$Neigborhood1<- ifelse(MidCity_Actual$Nbhd==1,1,0)
MidCity_Actual$Neigborhood2<- ifelse(MidCity_Actual$Nbhd==2,1,0)
MidCity_Actual$Neigborhood3<- ifelse(MidCity_Actual$Nbhd==3,1,0)
MidCity_Actual$Brick_Param<- ifelse(MidCity_Actual$Brick=='Yes',1,0)
MidCity_Actual<-MidCity_Actual[,-2]
MidCity_Actual<-MidCity_Actual[,-4]

lm.fit <- lm(formula = MidCity_Actual$Price~., data=MidCity_Actual)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = MidCity_Actual$Price ~ ., data = MidCity_Actual)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -27897.8  -6074.8    -48.7   5551.8  27536.4
##
## Coefficients: (1 not defined because of singularities)
##                Estimate Std. Error t value          Pr(>|t|)
## (Intercept)   22572.432  10287.606   2.194          0.030168 *
## Home            -11.456     25.387  -0.451          0.652616
## Offers        -8350.128   1103.693  -7.566 0.0000000000089599 ***
## SqFt             53.634      5.926   9.051 0.0000000000000033 ***
## Bedrooms       4136.461   1621.775   2.551          0.012023 *
## Bathrooms      7975.157   2133.831   3.737          0.000287 ***
## Neigborhood1 -20534.706   3176.051  -6.465 0.0000000023267089 ***
## Neigborhood2 -22264.319   2540.699  -8.763 0.0000000000000156 ***
## Neigborhood3         NA         NA      NA                NA
## Brick_Param   17313.540   1988.548   8.707 0.0000000000000212 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10050 on 119 degrees of freedom
## Multiple R-squared:  0.8688, Adjusted R-squared:   0.86
## F-statistic: 98.54 on 8 and 119 DF,  p-value: < 0.00000000000000022
```

Result - Fitted the linear regression model after one-hot encoding the Nbhd column. According to the model summary, co-relation between Neighborhood 1 with Price and Neighborhood 2 with Price is almost same. This suggests that, assuming all other features were the same, a home in neighborhood 2 would cost, same as a home in neighborhood 2 .

Problem 3: What causes what?

Part A

The facts for this can be extremely confusing since while a city with a high crime rate will recruit more police officers, it's also possible that having more officers will result in a city with a lower crime rate. Therefore, we are unable to simply add more data and perform regression.

Part B

The researchers picked the high alert days at random, so they weren't always the days with the highest crime rates. And today was chosen to study how increasing the number of police officers affects crime rates. As a result, the "natural experiment" was successful.

Also, from table 2 - we can see that when metro ridership is constant in Model 2 of table 2, the beta value for high alert days is still low. On high alert days the number of cops are higher thus we can conclude that higher cops can lead to lower crime rate in this case.

Part C

Controlling metro use is necessary because we don't want less people on the streets during the trial, which would result in reduced crime—which wouldn't be caused by more officers, but by fewer possible victims—rather than the opposite.

Part D

The interactive impact of High Alert on various districts is seen in this table. According to the analysis's findings, High Alert X District 1 and High Alert X Other Districts both have negative coefficients, however High Alert X District 1's coefficient is only statistically significant. Furthermore, the coefficient of High Alert X District 1 is significantly larger than that of High Alert X Other Districts, indicating that the influence of High Alert on District 1 has on lowering crime is significantly greater than that of other districts. On a related note, the previous regression results showed that Log(midday ridership) has a statistically significant positive coefficient, which means that as midday ridership rises, crime rates rise. This could be because there are more people on the streets, which leads to an increase in the number of victims and crimes.

PROJECT REPORT

Topic - Credit Score Classification | Group 8

My contribution in the project involves selecting the selecting the problem statement from Kaggle along with the dataset. Our problem statement was to predict the category of Credit Score into Good, Poor and Standard. We needed to select a modeling technique that outputs a higher recall value irrespective of the class imbalance. Since random forest and logistic regression were not giving us the expected results, we decided to use KNN as its Acuracy was the highest on the dataset.

All of the members of team were involved in the data cleaning, along with the initial data understanding. Together, we performed data cleaning, ie dealing with Nulls and NaNs. We replaced the Missing values in Categorical columns by "Unknown" and Missing Values in Numerical by finding the median of the data per Occupation.

I have performed Exploratory Data Analysis and Outliers Treatment on the cleaned data set. EDA was divided into 2 parts - Univariate and Bi-Variate Analysis. In the Univariate Analysis, I looked for how is the spread of numerical data set and the frequency of each categorical columns. In the Bi- variate Analysis, I looked again the same but with respect to our Target Variable,i.e Credit_Score. After EDA, I performed Outlier Treatment. For this, we capped and floored the Outliers at 90th and 10th Percentile, so that we do not have to lose any data ; also not creating any bias

Our dataset contains ~100K rows and ~22 columns (after performing PCA, dropping the not required columns).I have divided the dataset into 3 parts : Train Set, Validation Set and Test Set. I have performed Random Forest Classifier on the Dataset by using Stratified K Fold Cross Validation n_splits=10. Stratified Cross validation validation was performed to make sure that distribution of Target Class in dataset is balanced. Using the Trained Model, I predicted Credit_Score for the Test Set and to check if the trained Model is working fine on Unknown dataset , i.e Checking to see if the model is not overfitted or underfitted. Random Forest gave a decent accuracy on the test data , equals to 70.67%.

Apart from KNN, Random Forest and Gradient Boosting performed equally on the Dataset.