HW2

Anish mohan

September 29, 2015

1. **Q1**

Q1

Equation 10.12

$$\frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 = 2 \sum_{i \in C_k} \sum_{j=1}^{p} (x_{ij} - \bar{x}_{kj})^2$$

$$\bar{x}_{kj} = \frac{1}{|C_k|} \sum_{i \in C_k} x_{ij}$$

$$= \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j} - \bar{x}_{kj} + \bar{x}_{kj})^2$$

$$= \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} ((x_{ij} - \bar{x}_{kj}) - (x_{i'j} - \bar{x}_{kj}))^2$$

$$= \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - \bar{x}_{kj})^2 + (x_{i'j} - \bar{x}_{kj})^2 - 2(x_{ij} - \bar{x}_{kj})(x_{i'j} - \bar{x}_{kj})$$

$$= \frac{1 \cdot |C_k|}{|C_k|} \sum_{i \in C_k} \sum_{j=1}^{p} (x_{ij} - \bar{x}_{kj})^2 + \frac{|C_k|}{C_k} \sum_{i' \in C_k} \sum_{j=1}^{p} (x_{i'j} - \bar{x}_{kj})^2 - 2 \times \{0\}$$

changin i' to i for summation

don't know why this $=0$ but without it equality does not satisfy! Sorry!

$$= 2 \sum_{i \in C_k} \sum_{j=1}^{p} (x_{ij} - \bar{x}_{kj})^2$$

$= \text{R.HS}$

- 1a. Proof

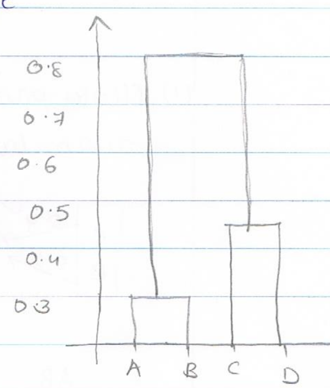
- 1b. As proved, the objective function $\sum_{i,i' \in C_k} \sum_{j=1}^{p} \{x_{i,j} - x_{i'j}\}^2$ is equivalent to finding the sum of distances of the point from the centroid of the cluster. Now, during each iteration each point is assigned to the closest centroid, hence in each iteration the cluster of points in a class are getting closer to the centroid of the class (obtained by current set of points of class). The process continues in each iteration and we continue to reduce the distance between points that belong to the same class.

2. **Q2:**

(Q2)

	A	B	C	D
A		0.3	0.4	0.7
B			0.5	0.8
C				0.45
D				

a) Dendrogram using complete linkage



→ Cluster/Pair with lowest dissimilar score = AB = 0.3

A → C = 0.4
B → C = 0.5 ✓

A → D = 0.7
B → D = 0.8 ✓

	AB	C	D
AB		0.5	0.8
C	0.5		0.45
D	0.8	0.45	

→ Cluster/Pair with lowest dissimilar score = CD = 0.45

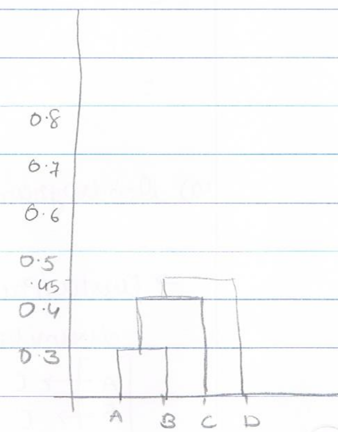
$\boxed{A, B} \xrightarrow{0.4,\ 0.5} \boxed{C, D}$ $\quad$ $\boxed{A, B} \xrightarrow{0.7,\ 0.8\ ✓} \boxed{C, D}$

	AB	CD
AB		0.8
CD	0.8	

- 2a.

(b) Dendrogram with single linkage clustering

	A	B	C	D
A		0.3	0.4	0.7
B	0.3		0.5	0.8
C	0.4			0.45
D	0.7	0.8		



(1) Cluster pair with lowest dissimilarity score = AB = 0.3

$A \xrightarrow{0.4} C$, $B \xrightarrow{0.5} C$

$A \xrightarrow{0.7} D$, $B \xrightarrow{0.8} D$

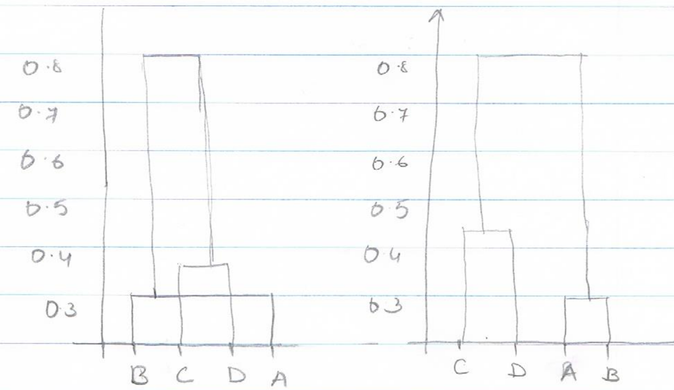
	AB	C	D
AB		0.4	0.7
C	0.4		0.45
D	0.7		

→ Cluster pair with lowest dissimilarity score = AB C = 0.4

$A \xrightarrow{0.7} D$, $B \xrightarrow{0.8} D$, $C \xrightarrow{0.45} D$

	AB	CD
AB		0.45
CD	0.45	

- 2b.
- 2c. AB and CD are the two clusters
- 2d. ABC and D are the two clusters

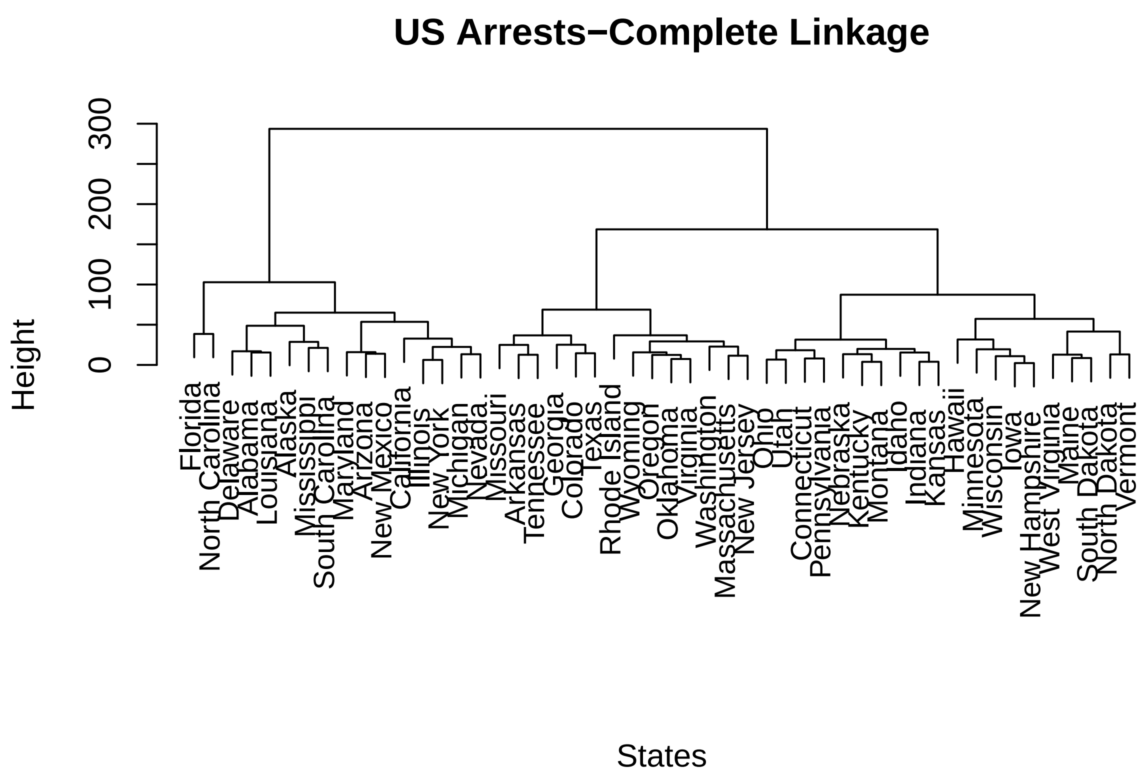


- 2e.

3. **Q3**

- 3a.

```r
Arrests_Data=USArrests
HC_Arrests_Complete=hclust(dist(Arrests_Data),method="complete")
plot(HC_Arrests_Complete, main ="US Arrests-Complete Linkage",xlab="States", sub=" ",cex=0.9)
```



- 3b.

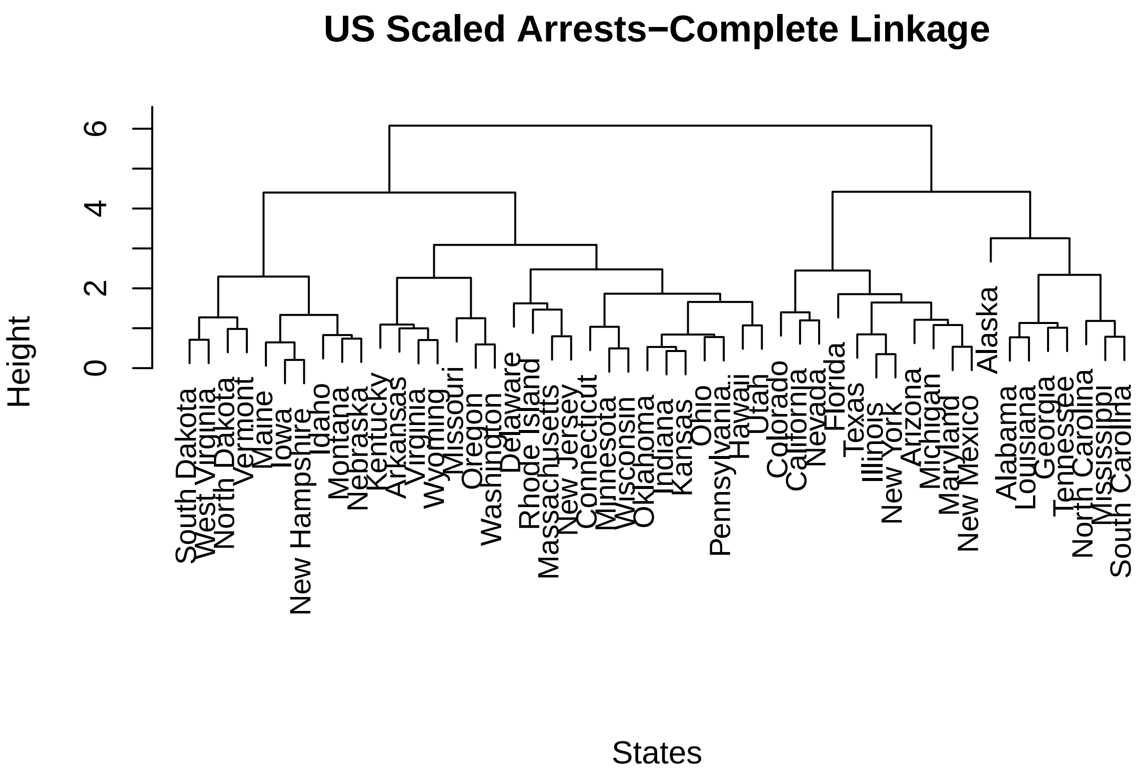
```r
cutree(HC_Arrests_Complete,3)
```

```
##        Alabama         Alaska        Arizona       Arkansas     California 
##              1              1              1              2              1 
##       Colorado    Connecticut       Delaware        Florida        Georgia 
##              2              3              1              1              2 
##         Hawaii          Idaho       Illinois        Indiana           Iowa 
##              3              3              1              3              3 
##         Kansas       Kentucky      Louisiana          Maine       Maryland 
##              3              3              1              3              1 
##  Massachusetts       Michigan      Minnesota    Mississippi       Missouri 
##              2              1              3              1              2 
##        Montana       Nebraska         Nevada  New Hampshire     New Jersey 
##              3              3              1              3              2 
##     New Mexico       New York North Carolina   North Dakota           Ohio 
##              1              1              1              3              3 
```

```
##       Oklahoma         Oregon   Pennsylvania   Rhode Island South Carolina
##              2              2              3              2              1
##   South Dakota      Tennessee          Texas           Utah        Vermont
##              3              2              2              3              3
##       Virginia     Washington  West Virginia      Wisconsin        Wyoming
##              2              2              3              3              2
```

- 3c.

```
Arrests_Data_Scaled=scale(Arrests_Data,scale=TRUE)
HC_ScaledArrests_Complete=hclust(dist(Arrests_Data_Scaled),method="complete")
plot(HC_ScaledArrests_Complete, main ="US Scaled Arrests-Complete Linkage",xlab="States", sub="",cex=0.
```



- 3d. Scaling the variables has the clustering of the states and the clustering after states is different before and after scaling. For e.g Arizon and Arkansas have moved to different clusters after scaling.

  Scaling should be done before creating the distance/dissimilarity matrix and some variables/features have higher values e.g Assault, that overwhelms the results from variables/feature with lower values/range e.g Murder in the USArrests Data.

4. **Q4**

- 4a. Theoretically, it is possible to have the linear regression and cubic regression to have the same or similar RSS if the true relationship is linear. The regression model for cubic (when the underlying model is linear) should give us β_2 and $\beta_3 == 0$. However, since the the training data would contain noise and a cubic model would be more prone to fitting the noise, the RSS value is expected to be lower than that for linear regression model.

- 4b. Test data will contain noise and the cubic model will be more prone to noise. The cubic model being more flexible will fit to the noise in the data and will have higher residual error than linear model with real datasets.
- 4c. If the true relationship is not linear then the accuracy of the model will depend upon the noise in the data and the amount of non-linearity.

  In general, a cubic regression model (flexible) would perform better than a linear regression model when the underlying function is non-linear. RSS error on training data should be lower with the cubic regression model.

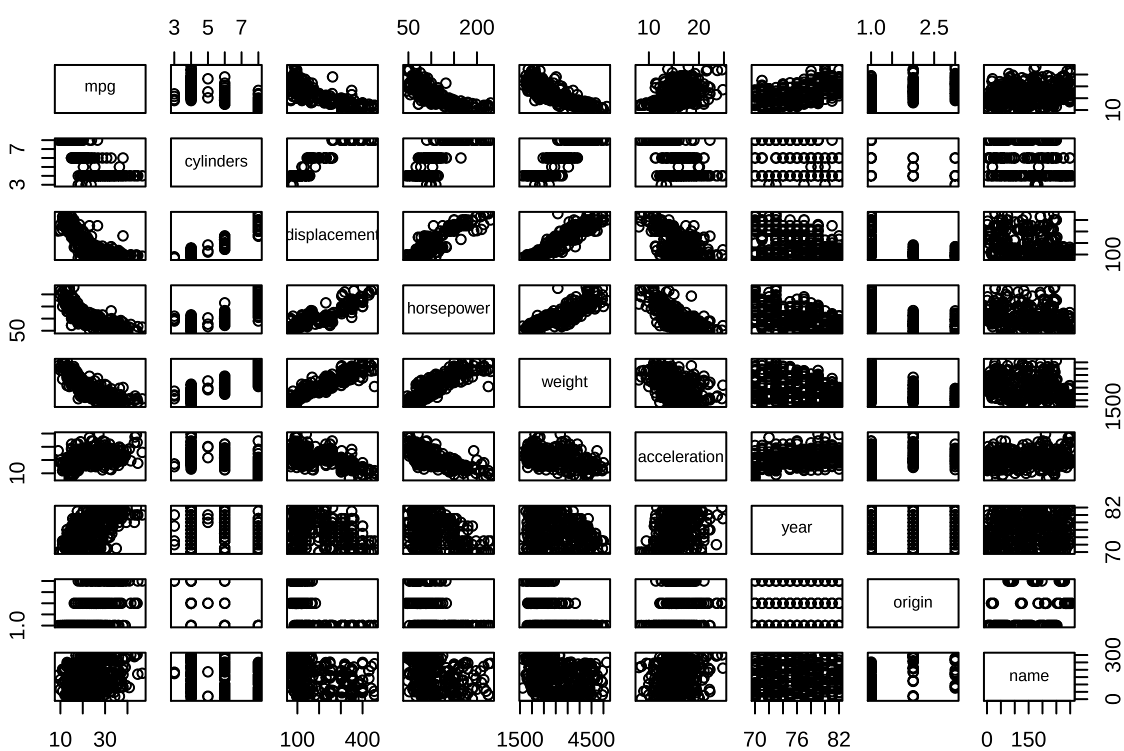
  Linear regression model introduces bias when used for non-linear true function hence can result in more errors.

  Noise in the data can impact the results we get from a cubic model. Noisy data can cause the variance to be high and impact the results from a cubic model as the model is flexible and prone to overfitting to noise.
- 4d. Same as above. In general, it is difficult to give an estimation of errors without knowing the true function, however for most scenarios (with low noise) cubic model should perform better with test data if the underlying model is non linear.

5. **Q5**

- 5a.

```
  library(MASS)
  library(ISLR)
autodat=Auto
pairs(autodat)
```



- 5b.

```r
cor(autodat[,1:8])
```

```
##                     mpg  cylinders displacement horsepower     weight
## mpg           1.0000000 -0.7776175   -0.8051269 -0.7784268 -0.8322442
## cylinders    -0.7776175  1.0000000    0.9508233  0.8429834  0.8975273
## displacement -0.8051269  0.9508233    1.0000000  0.8972570  0.9329944
## horsepower   -0.7784268  0.8429834    0.8972570  1.0000000  0.8645377
## weight       -0.8322442  0.8975273    0.9329944  0.8645377  1.0000000
## acceleration  0.4233285 -0.5046834   -0.5438005 -0.6891955 -0.4168392
## year          0.5805410 -0.3456474   -0.3698552 -0.4163615 -0.3091199
## origin        0.5652088 -0.5689316   -0.6145351 -0.4551715 -0.5850054
##              acceleration       year     origin
## mpg             0.4233285  0.5805410  0.5652088
## cylinders      -0.5046834 -0.3456474 -0.5689316
## displacement   -0.5438005 -0.3698552 -0.6145351
## horsepower     -0.6891955 -0.4163615 -0.4551715
## weight         -0.4168392 -0.3091199 -0.5850054
## acceleration    1.0000000  0.2903161  0.2127458
## year            0.2903161  1.0000000  0.1815277
## origin          0.2127458  0.1815277  1.0000000
```

- 5c.

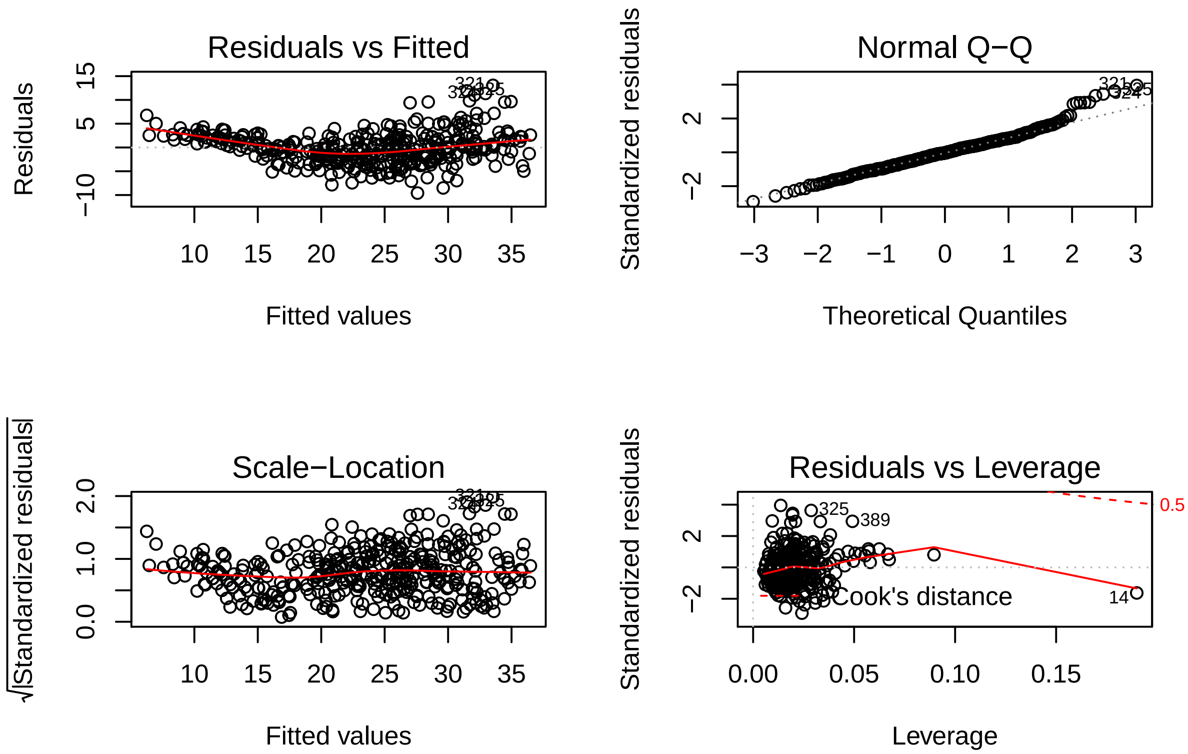
```r
attach(autodat)
autolm=lm(mpg~cylinders+displacement+horsepower+weight+acceleration+year+origin)
summary(autolm)
```

```
##
## Call:
## lm(formula = mpg ~ cylinders + displacement + horsepower + weight +
##     acceleration + year + origin)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.5903 -2.1565 -0.1169  1.8690 13.0604
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -17.218435   4.644294  -3.707  0.00024 ***
## cylinders     -0.493376   0.323282  -1.526  0.12780
## displacement   0.019896   0.007515   2.647  0.00844 **
## horsepower    -0.016951   0.013787  -1.230  0.21963
## weight        -0.006474   0.000652  -9.929  < 2e-16 ***
## acceleration   0.080576   0.098845   0.815  0.41548
## year           0.750773   0.050973  14.729  < 2e-16 ***
## origin         1.426141   0.278136   5.127 4.67e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.328 on 384 degrees of freedom
## Multiple R-squared:  0.8215, Adjusted R-squared:  0.8182
## F-statistic: 252.4 on 7 and 384 DF,  p-value: < 2.2e-16
```

- i. Yes there is relationship between some of the predictors and the response (mpg) as can be seen from the graph. Example, there is a correlation between mpg and displacement, mpg and horsepower, mpg and weight etc.
- ii. From the summary table, Year, Weight,Origin seem to have statistically significant ($p<0.001$) relationship to MPG.
- iii. Coefficient for the year variable is 0.75, hence it suggest that given specific values for other predictors,
  every year the MPG increases by 0.75 unit

- 5d.

```r
par(mfrow=c(2,2))
plot(autolm)
```



- The Residuals vs Fitted plot shows a trend line and the shape of the trend line suggests non-linearity in the data.
- Some points #321,#324 in Residuals vs Fitted graph, have higher residual values and they potentially could be the outliers.
- Point #14 in Residuals vs Leverage Graph, has high leverage.

6. Q6

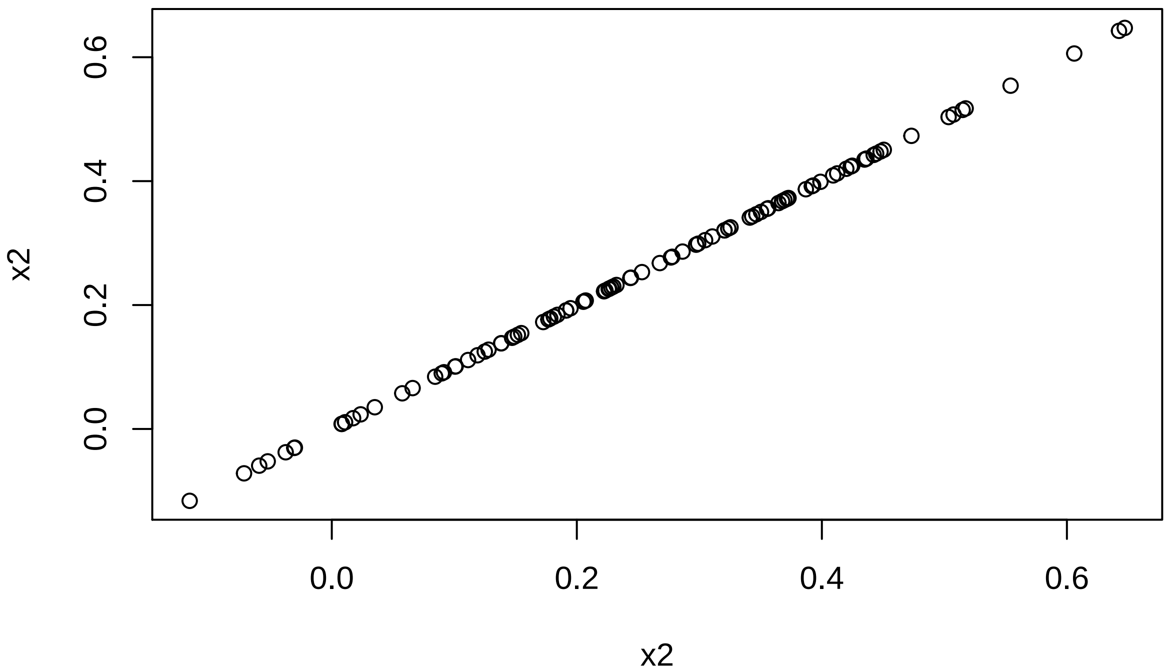
- 6a. Equation: $y = 2 + 2 * x_1 + 0.3 * x_2$
  $\beta_0 = 2$, $\beta_1 = 2$, $\beta_2 = 0.3$

- 6b.

```r
set.seed(1)
x1=runif(100)
x2=0.5*x1+rnorm(100)/10
y=2+2*x1+0.3*x2+rnorm(100)
cor(x1,x2)
```

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

```r
plot(x2,x2)
```



- 6c.

```r
ylm=lm(y~x1+x2)
summary(ylm)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8311 -0.7273 -0.0537  0.6338  2.3359
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.1305     0.2319   9.188 7.61e-15 ***
## x1            1.4396     0.7212   1.996   0.0487 *
## x2            1.0097     1.1337   0.891   0.3754
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 1.056 on 97 degrees of freedom
## Multiple R-squared:  0.2088, Adjusted R-squared:  0.1925
## F-statistic:  12.8 on 2 and 97 DF,  p-value: 1.164e-05
```

- $\hat{\beta}_0 = 2.13$ $\hat{\beta}_0$ is a good estimator of β_0 as the t-value is high and low p-value.
- $\hat{\beta}_1 = 1.4396$ $\hat{\beta}_1$ provides reasonable estimate of β_1 and gets close enough. This indicated by comparitively a t-statistic that is not very high and p-value that is near the cut-off of 0.05.
- $\hat{\beta}_2 = 1.0097$ $\hat{\beta}_1$ is a poor estimator of β_1. t-statistic is fairly low and p-value is high
- Yes, $H_0 : \beta_1 = 0$ can be rejected as p-value =0.04 is below the cut-off of 0.05 or 5%
- No, $H_0 : \beta_2 = 0$ cannot be rejected as p-value =0.375 is above the cut-off of 0.05 or 5%
- 6d.

```
y2lm=lm(y~x1)
summary(y2lm)
```

```
##
## Call:
## lm(formula = y ~ x1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.89495 -0.66874 -0.07785  0.59221  2.45560
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.1124     0.2307   9.155 8.27e-15 ***
## x1            1.9759     0.3963   4.986 2.66e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.055 on 98 degrees of freedom
## Multiple R-squared:  0.2024, Adjusted R-squared:  0.1942
## F-statistic: 24.86 on 1 and 98 DF,  p-value: 2.661e-06
```

```
+  Yes, $H_{0}: \beta_{1} =0$ can be rejected as p-value well below 0.001
```

- 6e.

```
y3lm=lm(y~x2)
summary(y3lm)
```

```
##
## Call:
## lm(formula = y ~ x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -2.62687 -0.75156 -0.03598  0.72383  2.44890
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.3899     0.1949   12.26  < 2e-16 ***
## x2            2.8996     0.6330    4.58 1.37e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.072 on 98 degrees of freedom
## Multiple R-squared:  0.1763, Adjusted R-squared:  0.1679
## F-statistic: 20.98 on 1 and 98 DF,  p-value: 1.366e-05
```

```
+  Yes, $H_{0}: \beta_{2} =0$ can be rejected as p-value well below 0.001
```

- 6f. No, the results do not contradict each other. Both x_1 and x_2 individually are good predictors of y. That is shown by 6d and 6e. However, x_1 and x_2 are highly correlated. Hence once β_1 provided appropirate weighting to x_1, adding x_2 does not introduce any additional information for better fit of y.

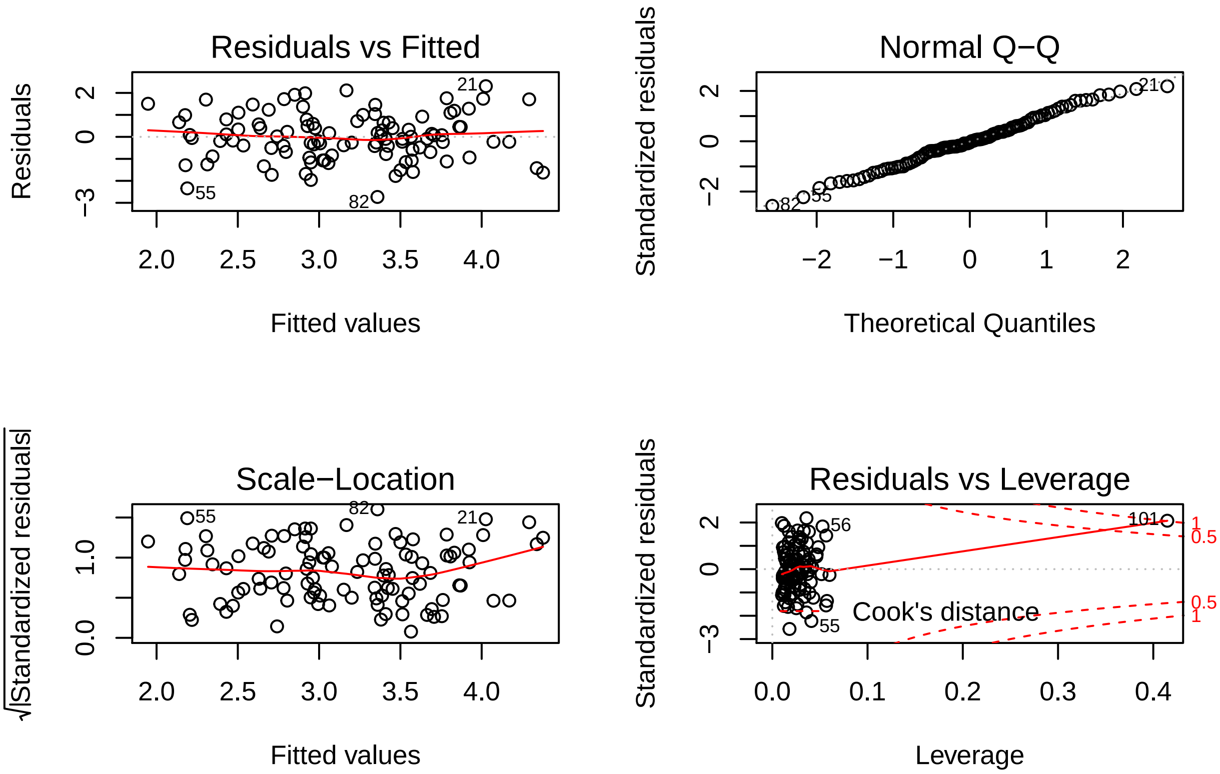
- 6g.

```
x1=c(x1, 0.1)
x2=c(x2, 0.8)
y=c(y, 6)

y5lm=lm(y~x1+x2)
summary(y5lm)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.73348 -0.69318 -0.05263  0.66385  2.30619
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.2267     0.2314   9.624 7.91e-16 ***
## x1            0.5394     0.5922   0.911  0.36458
## x2            2.5146     0.8977   2.801  0.00614 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.075 on 98 degrees of freedom
## Multiple R-squared:  0.2188, Adjusted R-squared:  0.2029
## F-statistic: 13.72 on 2 and 98 DF,  p-value: 5.564e-06
```

```
par(mfrow=c(2,2))
plot(y5lm)
```



+ $\hat{\beta_{0}}=2.23$
  $\hat{\beta_{0}}$ is a good estimator of β_{0} as the t-value is high and low p-value.

+ $\hat{\beta_{1}}=0.54$
  $\hat{\beta_{1}}$ s a poor estimator of β_{1}. t-statistic is fairly low and p-value is high

+ $\hat{\beta_{2}}=2.51$
  $\hat{\beta_{1}}$ is not a good estimator of β_{1}. t-statistic is low and p value is just abov
  of 5%

+ Yes, $H_{0}: \beta_{1} =0$ cannot be rejected as p-value is above the cut-off of 0.05 or 5%

+ No, $H_{0}: \beta_{2} =0$ cannot be rejected as p-value is above the cut-off of 0.05 or 5%

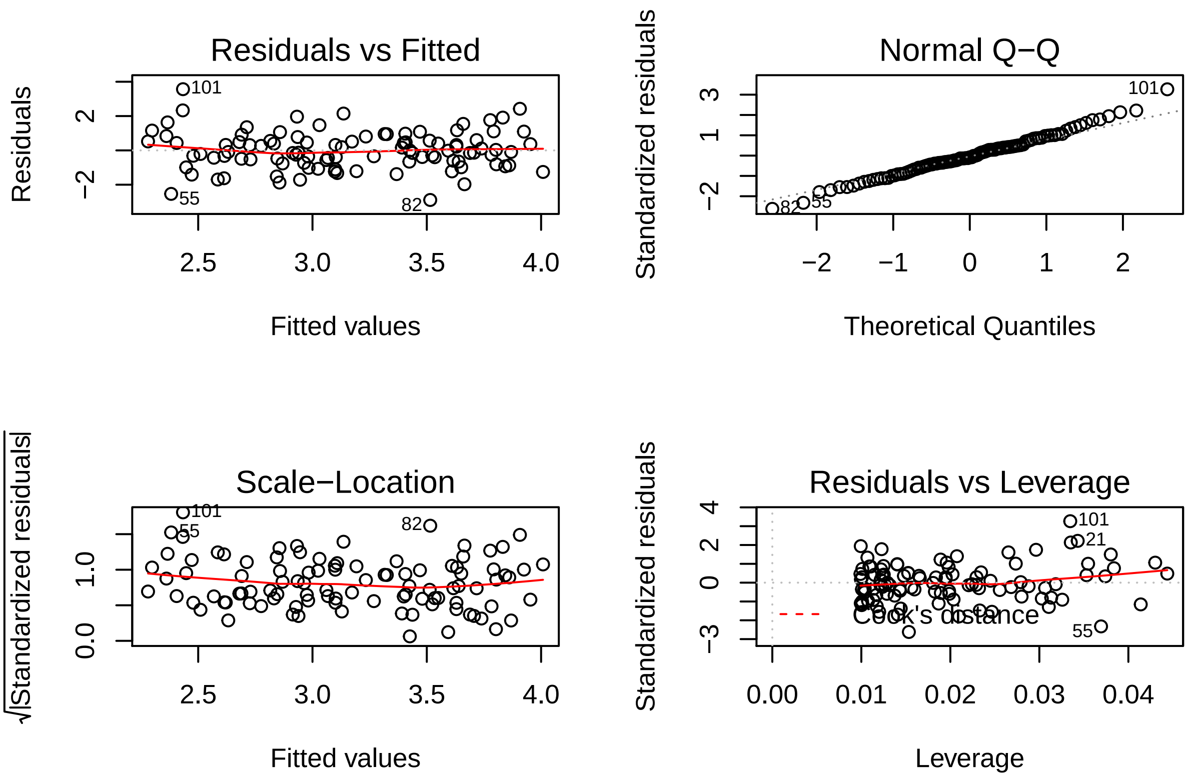
+ The new obeservation caused significant change in the estimates of β_{1} and β_{2}. This i
  because the new observation in x_{2} is a high leverage point.

```
y6lm=lm(y~x1)
summary(y6lm)
```

```
##
## Call:
## lm(formula = y ~ x1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8897 -0.6556 -0.0909  0.5682  3.5665
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.2569     0.2390   9.445 1.78e-15 ***
## x1            1.7657     0.4124   4.282 4.29e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.111 on 99 degrees of freedom
## Multiple R-squared:  0.1562, Adjusted R-squared:  0.1477
## F-statistic: 18.33 on 1 and 99 DF,  p-value: 4.295e-05
```

```r
par(mfrow=c(2,2))
plot(y6lm)
```



+ Yes, $H_{0}: \beta_{1} =0$ can be rejected as p-value well below 0.001

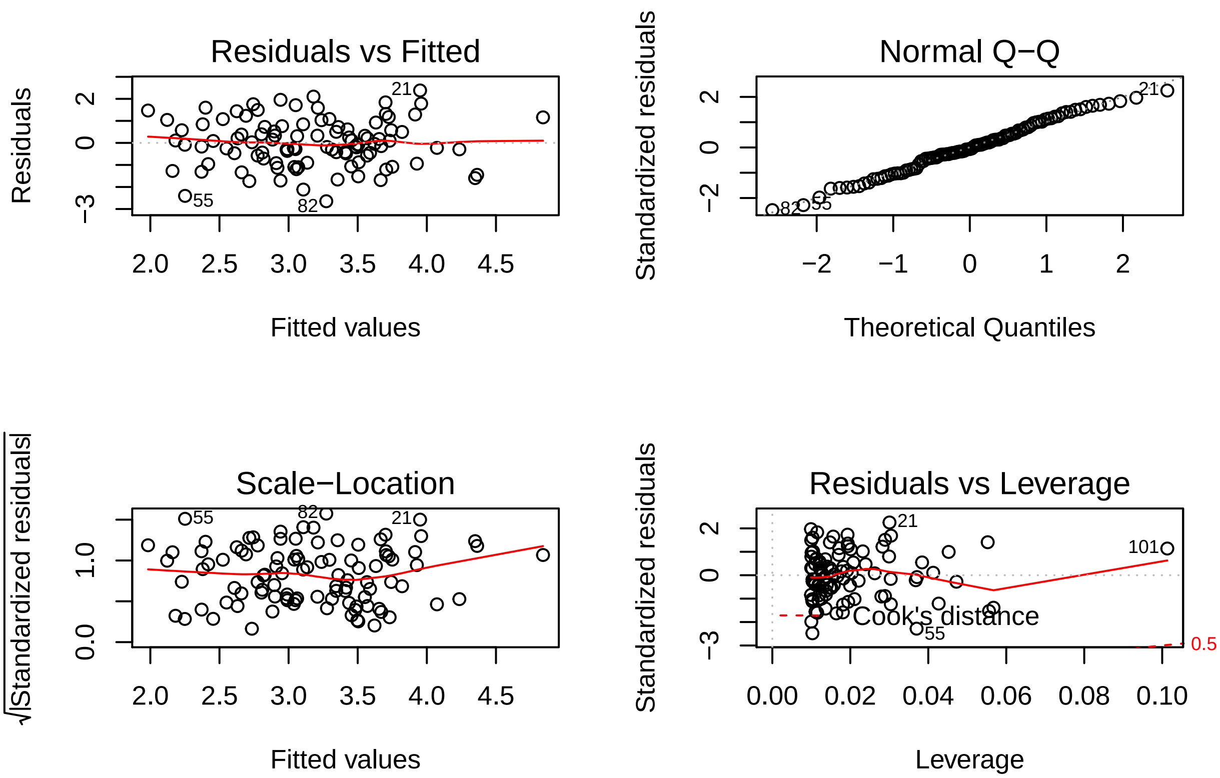
+ The new observation in x_{1} has not significantly impacted results.

```r
y7lm=lm(y~x2)
summary(y7lm)
```

```
##
## Call:
## lm(formula = y ~ x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -2.64729 -0.71021 -0.06899  0.72699  2.38074
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.3451     0.1912  12.264  < 2e-16 ***
## x2            3.1190     0.6040   5.164 1.25e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.074 on 99 degrees of freedom
## Multiple R-squared:  0.2122, Adjusted R-squared:  0.2042
## F-statistic: 26.66 on 1 and 99 DF,  p-value: 1.253e-06
```

```r
par(mfrow=c(2,2))
plot(y7lm)
```




```
+  Yes, $H_{0}: \beta_{2} =0$ can be rejected as p-value well below 0.001
+ The new observation in $x_{2}$ has had minor impact on the results but nothing very significant.


+ The new obeservation caused significant change in the estimates of $\beta_{1}$ and $\beta_{2}$ only i
  multiple variable regression. This is primarily because the new observation in $x_{2}$ is a high leve
```

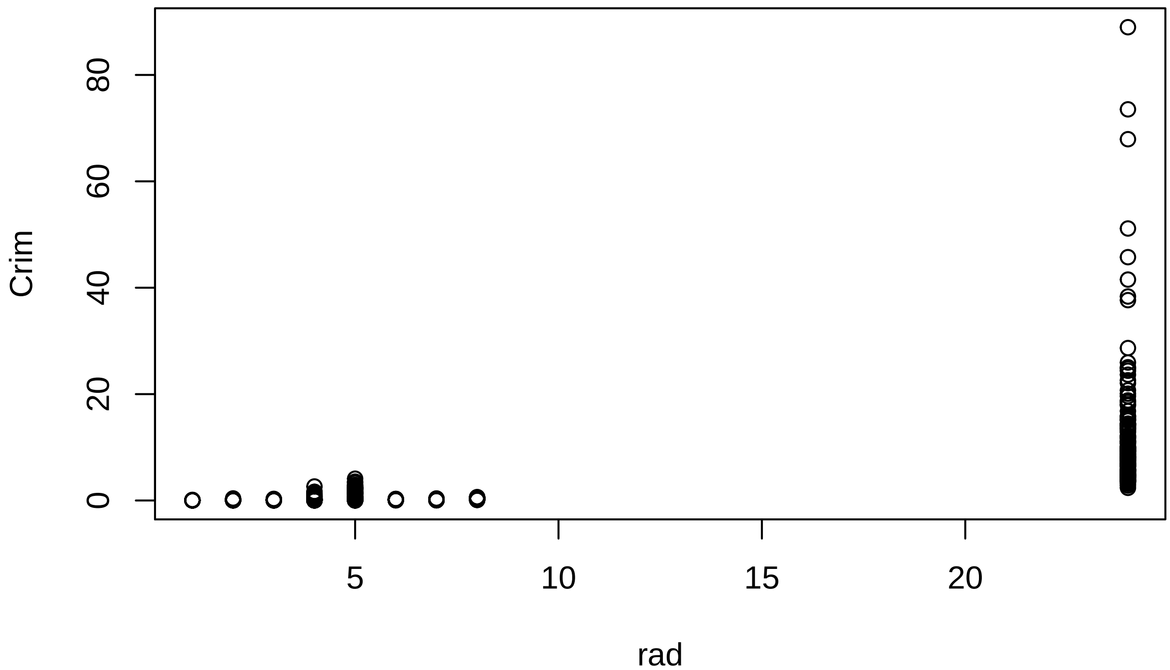
7. **Q7**

- 7a.

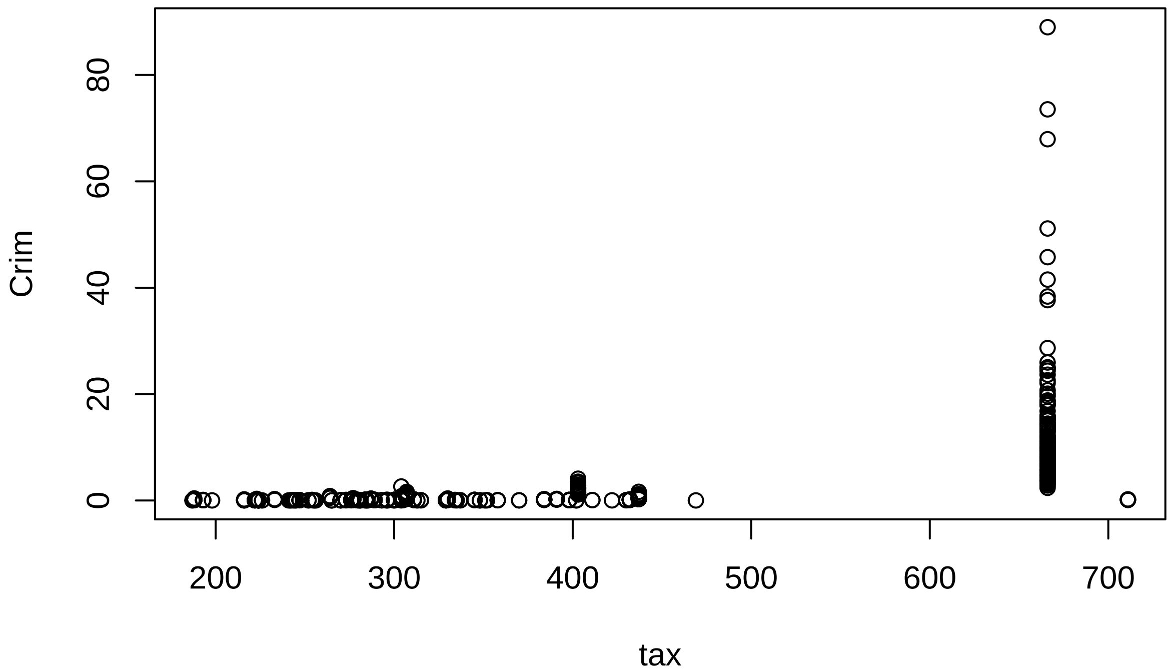
  - Plots show the predictor-response for top 3 t-values. Note that all the following variables have low p-values: Zn,Indus,Nox,Rm,Age,Dis,Rad, Tax, Ptration, Black, Lstat and Medv, but plots have been only included for the lowest 3 p-values

```
    MA=Boston
    MAName=names(MA)
    attach(MA)
    for (i in c(9,10,13)){
      y=MA$crim
      x=MA[,i]
      print(paste0("Predictor=", MAName[i]))
      print(summary(lm(y~x)))
      plot(x,y,xlab=MAName[i],ylab="Crim")
  }
```

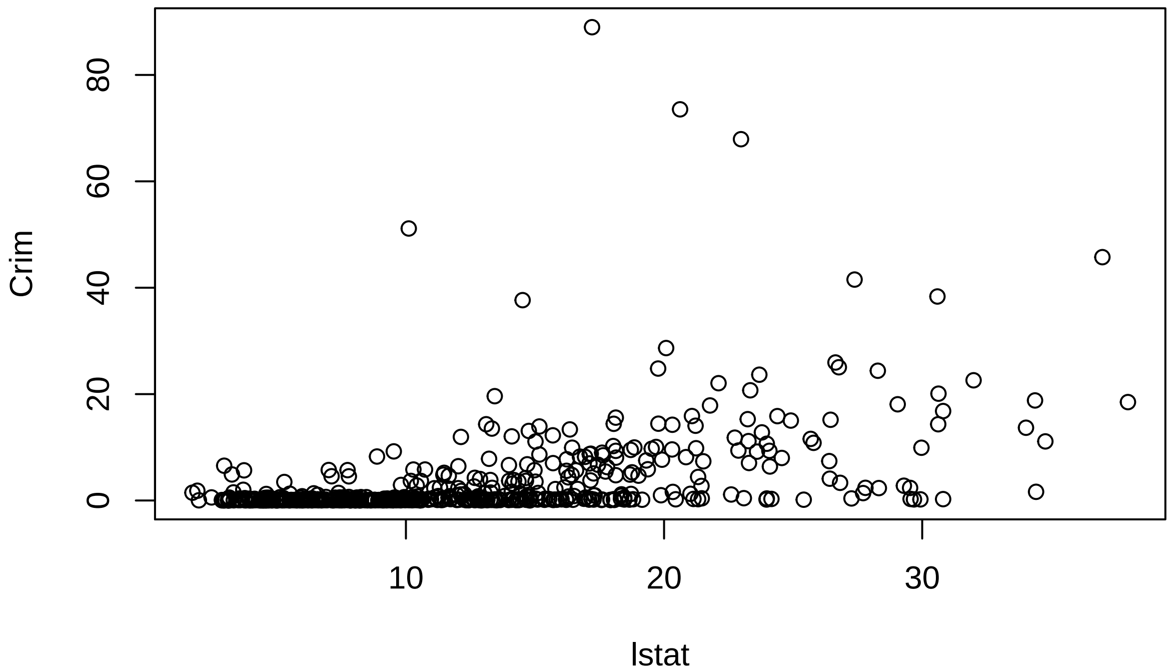
```
## [1] "Predictor=rad"
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.164  -1.381  -0.141   0.660  76.433
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.28716    0.44348  -5.157 3.61e-07 ***
## x   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




```
## [1] "Predictor=tax"
##
## Call:
## lm(formula = y ~ x)
##
## 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 ***
## x            0.029742   0.001847   16.10   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.997 on 504 degrees of freedom
## Multiple R-squared:  0.3396, Adjusted R-squared:  0.3383
## F-statistic: 259.2 on 1 and 504 DF,  p-value: < 2.2e-16
```




```
## [1] "Predictor=lstat"
##
## Call:
## lm(formula = y ~ x)
##
## 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 ***
## x            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
```



- 7b.

```r
attach(MA)
```

```
## The following objects are masked from MA (pos = 3):
##
##     age, black, chas, crim, dis, indus, lstat, medv, nox, ptratio,
##     rad, rm, tax, zn
```

```r
MAlm=lm(crim~zn+indus+chas+nox+rm+age+dis+rad+tax+ptratio+black+lstat+medv)
summary(MAlm)
```

```
##
## Call:
## lm(formula = crim ~ zn + indus + chas + nox + rm + age + dis +
##     rad + tax + ptratio + black + lstat + medv)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.924 -2.120 -0.353  1.019 75.051
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  17.033228   7.234903   2.354 0.018949 *
## zn            0.044855   0.018734   2.394 0.017025 *
## indus        -0.063855   0.083407  -0.766 0.444294
## chas         -0.749134   1.180147  -0.635 0.525867
## nox         -10.313535   5.275536  -1.955 0.051152 .
## rm            0.430131   0.612830   0.702 0.483089
## age           0.001452   0.017925   0.081 0.935488
```

```
## dis          -0.987176   0.281817  -3.503 0.000502 ***
## rad           0.588209   0.088049   6.680 6.46e-11 ***
## tax          -0.003780   0.005156  -0.733 0.463793
## ptratio      -0.271081   0.186450  -1.454 0.146611
## black        -0.007538   0.003673  -2.052 0.040702 *
## lstat         0.126211   0.075725   1.667 0.096208 .
## medv         -0.198887   0.060516  -3.287 0.001087 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.439 on 492 degrees of freedom
## Multiple R-squared:  0.454,  Adjusted R-squared:  0.4396
## F-statistic: 31.47 on 13 and 492 DF,  p-value: < 2.2e-16
```

- We can reject the Hypothesis $H_0 : \beta_j = 0$ for the following predictors because they have p-value<0.005: Dis, Rad, medv
- 7c.
  - In 7a. there were more variables with p-values<0.005: E.g Zn,Indus,Nox,Rm,Age,Dis,Rad, Tax, Ptration, Black, Lstat and Medv. However in 7b, only 3 variables (Rad, Dis, Medv) have p-values<0.005

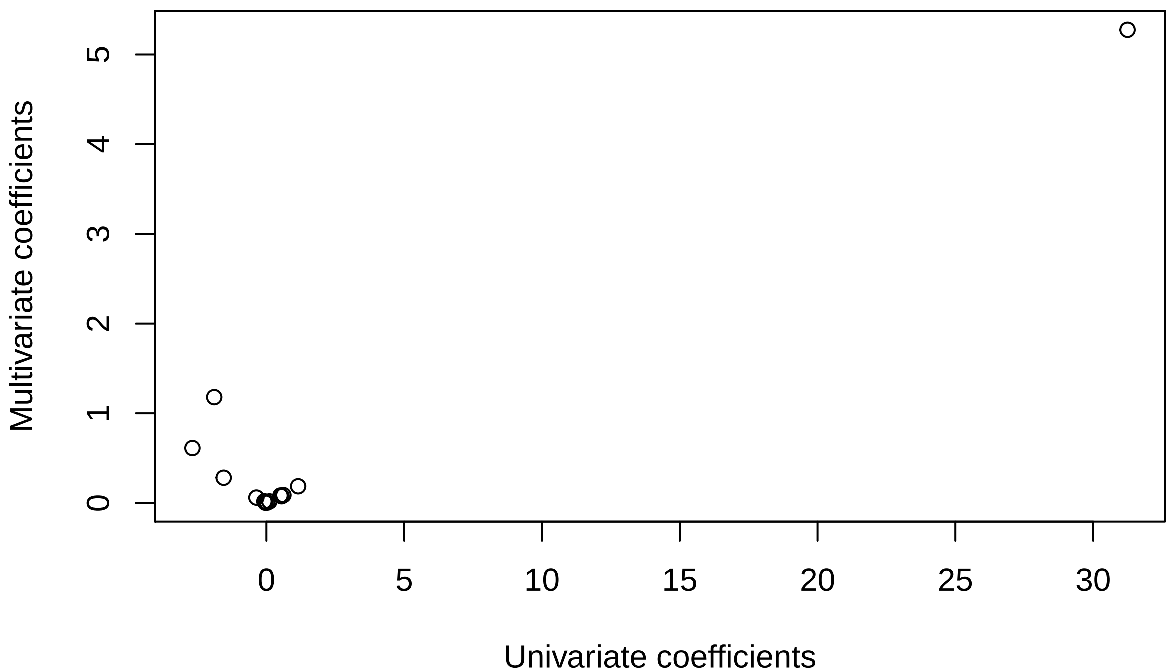
```
MA_coeff=matrix(0,14,3)
MA_coeff[,1]=names(MA)
for (i in c(2:14)){
  y=MA$crim
  x=MA[,i]
  templm=lm(y~x)
  temp=summary(templm)
  MA_coeff[i,2]=temp$coefficients[2,1]
}

S=summary(MAlm)
str(S)
```

```
## List of 11
##  $ call         : language lm(formula = crim ~ zn + indus + chas + nox + rm + age + dis + rad +
##  $ terms        :Classes 'terms', 'formula' length 3 crim ~ zn + indus + chas + nox + rm + age + dis
##   .. ..- attr(*, "variables")= language list(crim, zn, indus, chas, nox, rm, age, dis, rad, tax, ptra
##   .. ..- attr(*, "factors")= int [1:14, 1:13] 0 1 0 0 0 0 0 0 0 0 ...
##   .. .. ..- attr(*, "dimnames")=List of 2
##   .. .. .. ..$ : chr [1:14] "crim" "zn" "indus" "chas" ...
##   .. .. .. ..$ : chr [1:13] "zn" "indus" "chas" "nox" ...
##   .. ..- attr(*, "term.labels")= chr [1:13] "zn" "indus" "chas" "nox" ...
##   .. ..- attr(*, "order")= int [1:13] 1 1 1 1 1 1 1 1 1 1 ...
##   .. ..- attr(*, "intercept")= int 1
##   .. ..- attr(*, "response")= int 1
##   .. ..- attr(*, ".Environment")=<environment: R_GlobalEnv>
##   .. ..- attr(*, "predvars")= language list(crim, zn, indus, chas, nox, rm, age, dis, rad, tax, ptra
##   .. ..- attr(*, "dataClasses")= Named chr [1:14] "numeric" "numeric" "numeric" "numeric" ...
##   .. .. ..- attr(*, "names")= chr [1:14] "crim" "zn" "indus" "chas" ...
##  $ residuals    : Named num [1:506] 0.791 1.007 3.924 4.16 4.393 ...
##   ..- attr(*, "names")= chr [1:506] "1" "2" "3" "4" ...
```

```
##  $ coefficients : num [1:14, 1:4] 17.0332 0.0449 -0.0639 -0.7491 -10.3135 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:14] "(Intercept)" "zn" "indus" "chas" ...
##   .. ..$ : chr [1:4] "Estimate" "Std. Error" "t value" "Pr(>|t|)"
##  $ aliased      : Named logi [1:14] FALSE FALSE FALSE FALSE FALSE FALSE ...
##   ..- attr(*, "names")= chr [1:14] "(Intercept)" "zn" "indus" "chas" ...
##  $ sigma        : num 6.44
##  $ df           : int [1:3] 14 492 14
##  $ r.squared    : num 0.454
##  $ adj.r.squared: num 0.44
##  $ fstatistic   : Named num [1:3] 31.5 13 492
##   ..- attr(*, "names")= chr [1:3] "value" "numdf" "dendf"
##  $ cov.unscaled : num [1:14, 1:14] 1.26 4.25e-05 9.02e-04 4.16e-03 -5.22e-01 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:14] "(Intercept)" "zn" "indus" "chas" ...
##   .. ..$ : chr [1:14] "(Intercept)" "zn" "indus" "chas" ...
##  - attr(*, "class")= chr "summary.lm"
```

```
MA_coeff[2:14,3]=S$coefficients[2:14,2]
plot(MA_coeff[2:14,2],MA_coeff[2:14,3],xlab="Univariate coefficients", ylab="Multivariate coeffic
```



- 7d.

```
for (i in c(2:14)){
  y=MA$crim
  x=MA[,i]
  print(paste0("Predictor=", MAName[i]))
  print(summary(lm(y~x+I(x^2)+I(x^3))))
}
```

```
## [1] "Predictor=zn"
```

```
##
## Call:
## lm(formula = y ~ x + I(x^2) + I(x^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 ***
## x           -3.322e-01  1.098e-01  -3.025  0.00261 **
## I(x^2)       6.483e-03  3.861e-03   1.679  0.09375 .
## I(x^3)      -3.776e-05  3.139e-05  -1.203  0.22954
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.372 on 502 degrees of freedom
## Multiple R-squared:  0.05824,    Adjusted R-squared:  0.05261
## F-statistic: 10.35 on 3 and 502 DF,  p-value: 1.281e-06
##
## [1] "Predictor=indus"
##
## Call:
## lm(formula = y ~ x + I(x^2) + I(x^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 *
## x           -1.9652129  0.4819901  -4.077 5.30e-05 ***
## I(x^2)       0.2519373  0.0393221   6.407 3.42e-10 ***
## I(x^3)      -0.0069760  0.0009567  -7.292 1.20e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.423 on 502 degrees of freedom
## Multiple R-squared:  0.2597, Adjusted R-squared:  0.2552
## F-statistic: 58.69 on 3 and 502 DF,  p-value: < 2.2e-16
##
## [1] "Predictor=chas"
##
## Call:
## lm(formula = y ~ x + I(x^2) + I(x^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 ***
## x            -1.8928     1.5061  -1.257    0.209
## I(x^2)            NA         NA      NA       NA
## I(x^3)            NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.597 on 504 degrees of freedom
## Multiple R-squared:  0.003124,   Adjusted R-squared:  0.001146
## F-statistic: 1.579 on 1 and 504 DF,  p-value: 0.2094
##
## [1] "Predictor=nox"
##
## Call:
## lm(formula = y ~ x + I(x^2) + I(x^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 ***
## x           -1279.37     170.40  -7.508 2.76e-13 ***
## I(x^2)       2248.54     279.90   8.033 6.81e-15 ***
## I(x^3)      -1245.70     149.28  -8.345 6.96e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.234 on 502 degrees of freedom
## Multiple R-squared:  0.297, Adjusted R-squared:  0.2928
## F-statistic: 70.69 on 3 and 502 DF,  p-value: < 2.2e-16
##
## [1] "Predictor=rm"
##
## Call:
## lm(formula = y ~ x + I(x^2) + I(x^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 .
## x           -39.1501    31.3115  -1.250   0.2118
## I(x^2)        4.5509     5.0099   0.908   0.3641
## I(x^3)       -0.1745     0.2637  -0.662   0.5086
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.33 on 502 degrees of freedom
## Multiple R-squared:  0.06779,    Adjusted R-squared:  0.06222
## F-statistic: 12.17 on 3 and 502 DF,  p-value: 1.067e-07
```

```
##
## [1] "Predictor=age"
##
## Call:
## lm(formula = y ~ x + I(x^2) + I(x^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
## x            2.737e-01  1.864e-01   1.468  0.14266
## I(x^2)      -7.230e-03  3.637e-03  -1.988  0.04738 *
## I(x^3)       5.745e-05  2.109e-05   2.724  0.00668 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.84 on 502 degrees of freedom
## Multiple R-squared:  0.1742, Adjusted R-squared:  0.1693
## F-statistic: 35.31 on 3 and 502 DF,  p-value: < 2.2e-16
##
## [1] "Predictor=dis"
##
## Call:
## lm(formula = y ~ x + I(x^2) + I(x^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 ***
## x           -15.5543     1.7360  -8.960  < 2e-16 ***
## I(x^2)        2.4521     0.3464   7.078 4.94e-12 ***
## I(x^3)       -0.1186     0.0204  -5.814 1.09e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.331 on 502 degrees of freedom
## Multiple R-squared:  0.2778, Adjusted R-squared:  0.2735
## F-statistic: 64.37 on 3 and 502 DF,  p-value: < 2.2e-16
##
## [1] "Predictor=rad"
##
## Call:
## lm(formula = y ~ x + I(x^2) + I(x^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
## x            0.512736   1.043597   0.491    0.623
## I(x^2)      -0.075177   0.148543  -0.506    0.613
## I(x^3)       0.003209   0.004564   0.703    0.482
##
## Residual standard error: 6.682 on 502 degrees of freedom
## Multiple R-squared:    0.4,  Adjusted R-squared:  0.3965
## F-statistic: 111.6 on 3 and 502 DF,  p-value: < 2.2e-16
##
## [1] "Predictor=tax"
##
## Call:
## lm(formula = y ~ x + I(x^2) + I(x^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
## x           -1.533e-01  9.568e-02  -1.602    0.110
## I(x^2)       3.608e-04  2.425e-04   1.488    0.137
## I(x^3)      -2.204e-07  1.889e-07  -1.167    0.244
##
## Residual standard error: 6.854 on 502 degrees of freedom
## Multiple R-squared:  0.3689, Adjusted R-squared:  0.3651
## F-statistic:  97.8 on 3 and 502 DF,  p-value: < 2.2e-16
##
## [1] "Predictor=ptratio"
##
## Call:
## lm(formula = y ~ x + I(x^2) + I(x^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 **
## x           -82.36054   27.64394  -2.979  0.00303 **
## I(x^2)        4.63535    1.60832   2.882  0.00412 **
## I(x^3)       -0.08476    0.03090  -2.743  0.00630 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.122 on 502 degrees of freedom
## Multiple R-squared:  0.1138, Adjusted R-squared:  0.1085
## F-statistic: 21.48 on 3 and 502 DF,  p-value: 4.171e-13
##
## [1] "Predictor=black"
```

```
##
## Call:
## lm(formula = y ~ x + I(x^2) + I(x^3))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.096  -2.343  -2.128  -1.439  86.790
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.826e+01  2.305e+00   7.924  1.5e-14 ***
## x           -8.356e-02  5.633e-02  -1.483    0.139
## I(x^2)       2.137e-04  2.984e-04   0.716    0.474
## I(x^3)      -2.652e-07  4.364e-07  -0.608    0.544
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.955 on 502 degrees of freedom
## Multiple R-squared:  0.1498, Adjusted R-squared:  0.1448
## F-statistic: 29.49 on 3 and 502 DF,  p-value: < 2.2e-16
##
## [1] "Predictor=lstat"
##
## Call:
## lm(formula = y ~ x + I(x^2) + I(x^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
## x           -0.4490656  0.4648911  -0.966   0.3345
## I(x^2)       0.0557794  0.0301156   1.852   0.0646 .
## I(x^3)      -0.0008574  0.0005652  -1.517   0.1299
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.629 on 502 degrees of freedom
## Multiple R-squared:  0.2179, Adjusted R-squared:  0.2133
## F-statistic: 46.63 on 3 and 502 DF,  p-value: < 2.2e-16
##
## [1] "Predictor=medv"
##
## Call:
## lm(formula = y ~ x + I(x^2) + I(x^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 ***
## x           -5.0948305  0.4338321 -11.744  < 2e-16 ***
## I(x^2)       0.1554965  0.0171904   9.046  < 2e-16 ***
## I(x^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
```

```
+ Analyzing the summary tables, there seems to be non-linear relationship between "Crim" and the follow
```

8. **Q8**

- 8_1

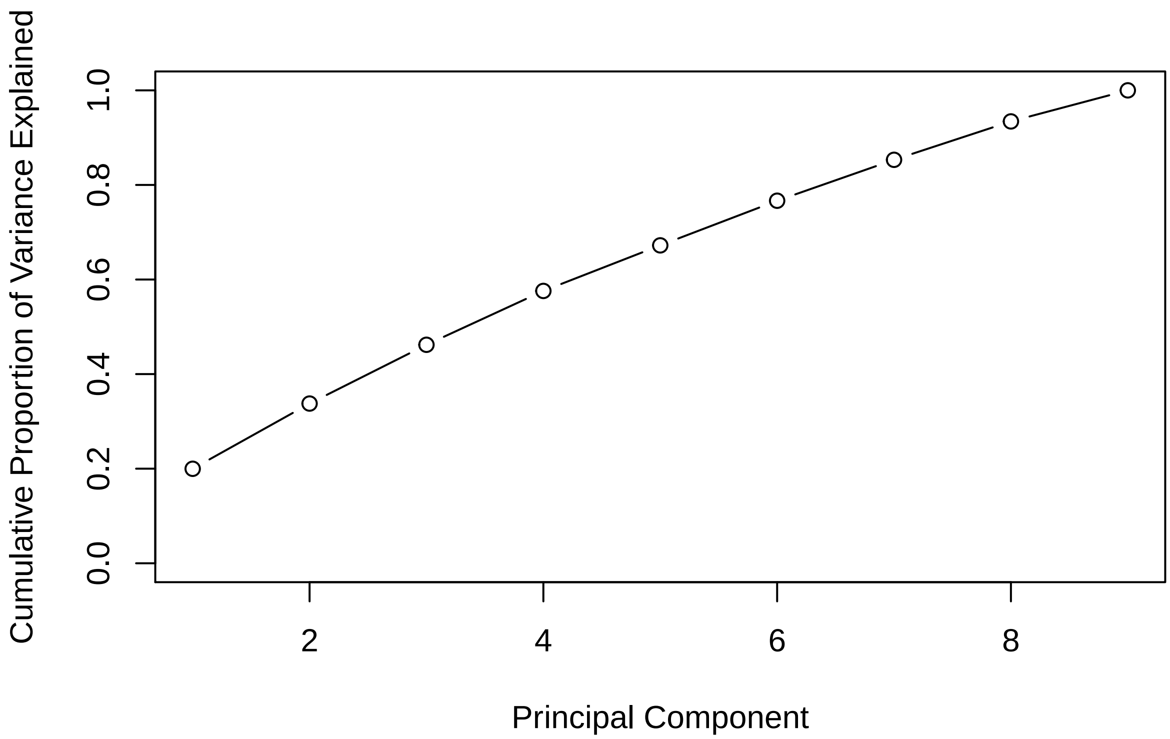
```
in_training_features=read.csv("training_features.csv")

feature.names<-names(in_training_features)
  for(feature.name in feature.names[-1]){
    dummy_name<-paste0("is.na.",feature.name)
    is.na.feature <-is.na(in_training_features[,feature.name])
    in_training_features[,dummy_name]<-as.integer(is.na.feature)
    in_training_features[is.na.feature,feature.name]<-median(in_training_features[,feature.name], na.
  }

newset=in_training_features[in_training_features$subject.id!=525450,]
newset2=newset[,c("q1_speech.slope", "q2_salivation.slope", "q3_swallowing.slope", "q4_handwriting.sl
```

- 8_2

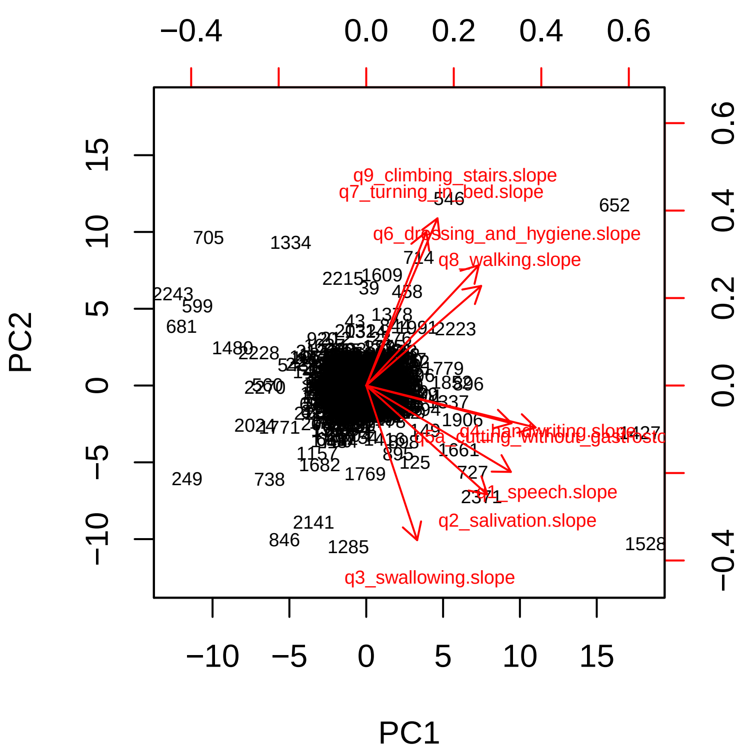
```
  pca_set2=prcomp(newset2,scale=TRUE)
  var_set2=pca_set2$sdev^2
  prop_var_set2=var_set2/sum(var_set2)
  plot(cumsum(prop_var_set2),xlab="Principal Component",ylab="Cumulative Proportion of Variance Expla
```



+ About 34% of the cumulative variance is captured by the top 2 Principal
  Components

- 8_3

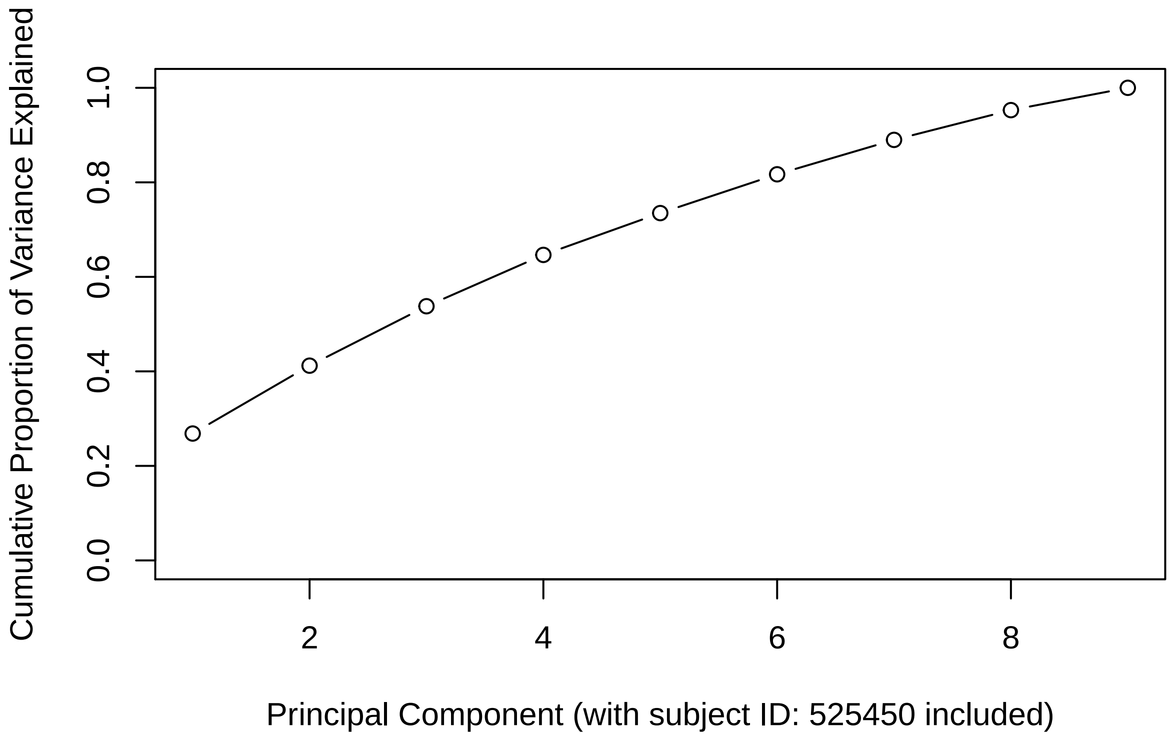
```r
biplot(pca_set2, scale=0,cex=0.6)
```



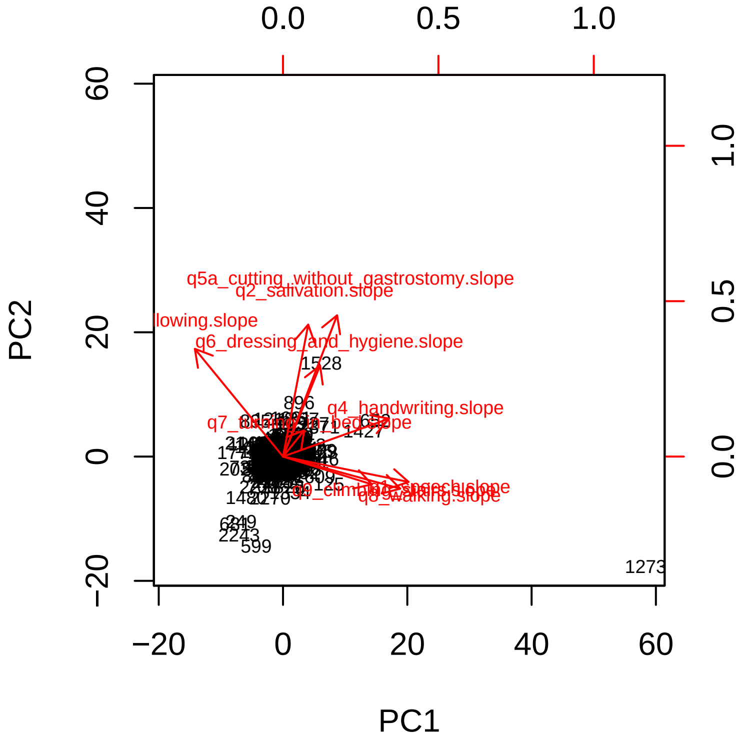
- There seems to be 2 sets of correlated vectors/dimensions:
  - Set 1: q9_climbing_stairs.slope, q7_turning_in_bed.slope, q6_dressing_and_hygiene.slope and q8_walking.slope are 1 set of correlated vectors
  - Set 2: q4_handwriting.slope,q5a_cutting_without_gastrostomy.slope,q1_speech.slope,q3_swallowing.slope and q2_salivation.slope are 2nd set of correlated vectors/dimentions.
- 8_4

```
newset3=in_training_features[,c("q1_speech.slope", "q2_salivation.slope", "q3_swallowing.slope", "q4

pca_set3=prcomp(newset3,scale=TRUE)
  var_set3=pca_set3$sdev^2
  prop_var_set3=var_set3/sum(var_set3)
  plot(cumsum(prop_var_set3),xlab="Principal Component (with subject ID: 525450 included)",ylab="Cum
```




```
biplot(pca_set3, scale=0,cex=0.6)
```




```
+ There is significant change in the directions or certain dimension vectors e.g q3_swallowing.slope and

Similarly, the q4_handwriting.slope for subject ID: 525450 is 10.14556. This is 3 times the next maximum
```

This is a good case of detecting outliers/leverage points and removing this from the data set.