STATS 202: Data Mining and Analysis

Homework 1

Problem 1:

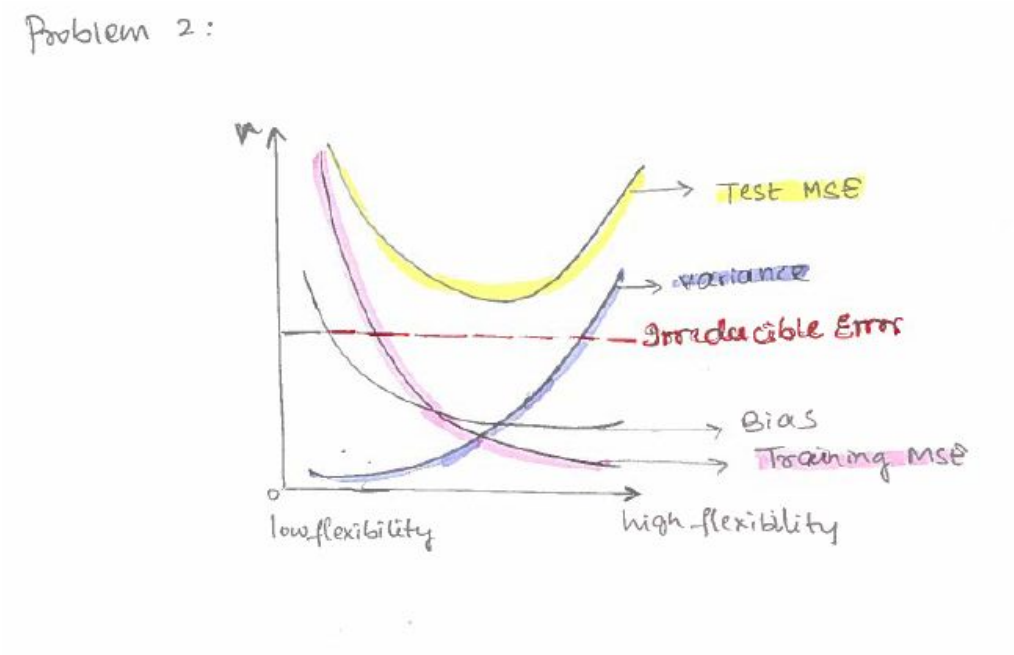
a) Responses in this scenario is CEO salary which is a quantitative data hence this is a **Regression problem.** Since the problems requires us to understand which factors affect CEO salary we are interested in **inference**.
   - **n=500**
   - **p= profit, number of employs, industry**

b) Responses in this scenario are success or failure which is categorical data hence this is a **Classification problem**. Since the problems requires to predict if the new product launch will be a success or a failure we are interested in **prediction.**
   - **n=20**
   - **p=price charged for the product, marketing budget, competition price and 10 other variables not mentioned in the scenario description.**

c) Responses in this scenario are % changes in the US dollar which is quantitative data hence this is a **Regression problem.** Since the problem requires to predict % change in the dollar in relation to the changes in the different world markets we are interested in **prediction.**
   - **n= 52**
   - **p=% change in the US market, %change in the British market, % change in the German market.**

Problem 2:

Problem 2:



Irreducible error: It remains constant as it is not affected by the flexibility of the statistical method

Bias: Bias starts of high and monotonically reduces as the flexibility of the statistical method increases because as the flexibility of the model increase, it fits the data better reducing the bias

Variance: Variance is pretty low for restrictive statistical method and increases monotonically as the flexibility increases as the model will follow the data closely resulting in overfitting.

Training MSE: Is typically high for restrictive models and starts decreasing as the model flexibility increases because as the flexibility of the model increases, it follows the data closely and generates better fit may be even perfect fit.

Test MSE: Based on bias variance decomposition, we know Test MSE = Variance + bias ^2 + variance of error terms. Based on this, test MSE is typically high for less flexible statistical methods. Because for restrictive methods though variance is low, Bias component is much higher. It then monotonically decreases as the flexibility increases to a certain level and then gradually increases with the high flexibility. Because as the flexibility of the statistical method increases, even though bias is low, variance component is pretty high.

Problem 3:

a) In Cartesian coordinates, if p = (p1, p2,..., pn) and q = (q1, q2,..., qn) are two points in Euclidean n-space, then the distance (d) from p to q, or from q to p is given by:

$$\mathrm{d}(\mathbf{p},\mathbf{q}) = \mathrm{d}(\mathbf{q},\mathbf{p}) = \sqrt{(q_1-p_1)^2 + (q_2-p_2)^2 + \cdots + (q_n-p_n)^2} = \sqrt{\sum_{i=1}^{n}(q_i-p_i)^2}.$$

With this equation, the Euclidean distance of X1=X2=X3=0 with the training data set is

Obs	Euclidean distance from X1=X2=X3=0		Y
1	sqrt((0-0)^2 + (0-3)^2+(0-0)^2)	= 3	Red
2	sqrt((0-2)^2+(0-0)^2+(0-0)^2)	= 2	Red
3	sqrt((0-0)^2+(0-1)^2+(0-3)^2)	= 3.16	Red
4	sqrt((0-0)^2+(0-1)^2+(0-2)^2)	= 2.24	Green
5	sqrt((0+1)^2+(0-0)^2+(0-1)^2)	=1.41	Green
6	sqrt((0-1)^2+(0-1)^2+(0-1)^2)	=1.73	Red

b) The prediction for the observation at X1=X2=X3=0 with K=1 is **Green**.

- **Justification:** KNN method tries to find the K nearest elements and assigns the test data point to the class with largest probability. With K=1, the test data at X1=X2=X3=0 is nearest to observation 5 with Euclidean distance equal to 1.41. Data point at 5th observation is Green and has probability=1 for K=1 hence the prediction is Green.

c) The prediction for the observation at X1=X2=X3=0 with K=3 is **Red**.

- **Justification:** KNN method tries to find the K nearest elements and assigns the test data point to the class with largest probability. With K=3, the test data at X1=X2=X3=0 is nearest to observations 2,5 and 6 with Euclidean distances as shown in the table:

Obs	Euclidean distance from X1=X2=X3=0		Y
2	sqrt((0-2)^2+(0-0)^2+(0-0)^2)	= 2	Red
5	sqrt((0+1)^2+(0-0)^2+(0-1)^2)	=1.41	Green
6	sqrt((0-1)^2+(0-1)^2+(0-1)^2)	=1.73	Red

The probability of class Red is 2/3 and for Green is 1/3 for K=3. Since Red has largest probability, the test data at X1=X2=X3=0 will be predicted to belong to class Red.

d) **The best value of K will be small.** Since the decision boundary is highly non-linear, i.e the model is very flexible hence closely follows the data. To produce a highly non-linear decision boundary using KNN, we need to choose a smaller value for K.

Problem 4:

To prove:

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

Proof :-

Lets add and subtract the centeroid to the LHS of (A)

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

Expanding the above equation :-

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

Since i & i' are just indexing notations we can rewrite (a) as

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

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

Since, $\sum$ constant = Constant,

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

$$\sum_{i' \in C_k} 1 = |C_k| \in \sum_{i' \in C_k} (\bar{x}_{kj} - x_{i'j}) = 0$$

$$= \frac{2 |C_k|}{|C_k|} \sum_{i \in C_k} \sum_{j=1}^{p} (x_{ij} - \bar{x}_{kj}) + \frac{2}{|C_k|} \sum_{i \in C_k} \sum_{j=1}^{p} (x_{ij} - \bar{x}_k) (0) \quad \text{— (b)}$$

From (b), we proved the identity in A.

i.e

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

b) From a) we know that sum of squared deviations between observations and the centroid is less than average sum of squared deviation between the observations with in a cluster. In K-means during each step we assign the observations to the nearest centroid thus reducing the sum of squared deviations each time, which means each time we assign observations to the nearest centroid we are minimizing the dissimilarity(WSS) between observations within the cluster i.e. minimize the objective of K-means algorithm.

Problem 5:

a) **First iteration:** Distance between observations using complete linkage:

(1,2)	0.3
(1,3)	0.4
(1,4)	0.7
(2,3)	0.5
(2,4)	0.8
(3,4)	0.45

The shortest distance is between (1,2). They will be clustered to A(1,2) first at height 0.3

Second iteration: Distance between cluster A(1,2), 3 and 4 using complete linkage

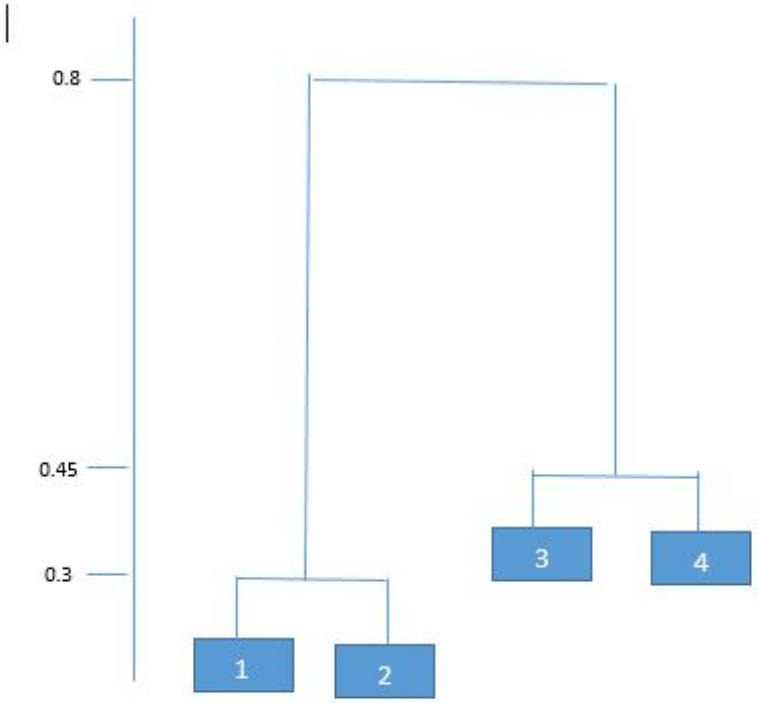
(A,3)= max[(1,3),(2,3)]	0.5
(A,4)= max[(1,4),(2,4)]	0.8
(3,4)	0.45

Shortest distance is between (3,4) and they will be clustered B(3,4) next at height 0.45

Third iteration: Distance between cluster A(1,2), B(3,4) using complete linkage

(A,B)=max[(1,2),(1,4),(2,3),(2,4)]	0.8

Hence clusters A(1,2) and B(3,4) are fused at height 0.8



b) First iteration: Distance between observations using single linkage:

(1,2)	0.3
(1,3)	0.4
(1,4)	0.7
(2,3)	0.5
(2,4)	0.8
(3,4)	0.45

The shortest distance is between (1,2). They will be clustered to A(1,2) first at height 0.3

Second iteration: Distance between cluster A(1,2),3 and 4 using single linkage

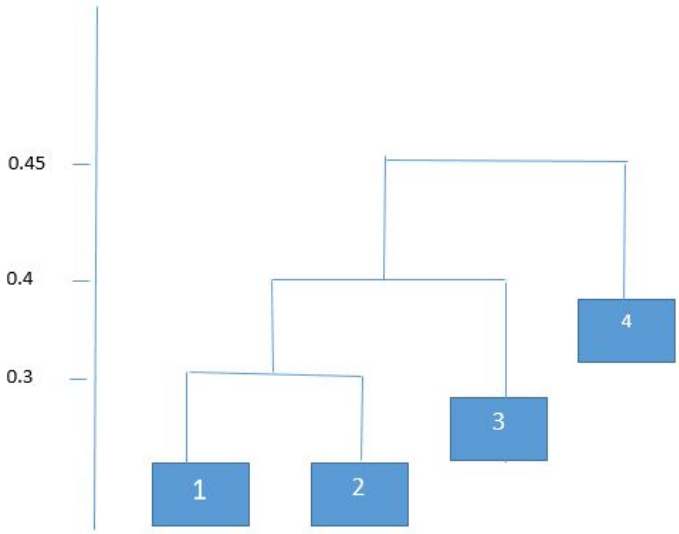
(A,3)= min[(1,3),(2,3)]	0.4
(A,4)= min[(1,4),(2,4)]	0.7
(3,4)	0.45

The shortest distance is between cluster A(1,2) and 3 to B(1,2,3). They are fused next at height 0.4

Third iteration: Distance between cluster B(1,2,3) and 4 using single linkage:

(B,4)=min[(1,4),(2,4),(3,4)]	0.45

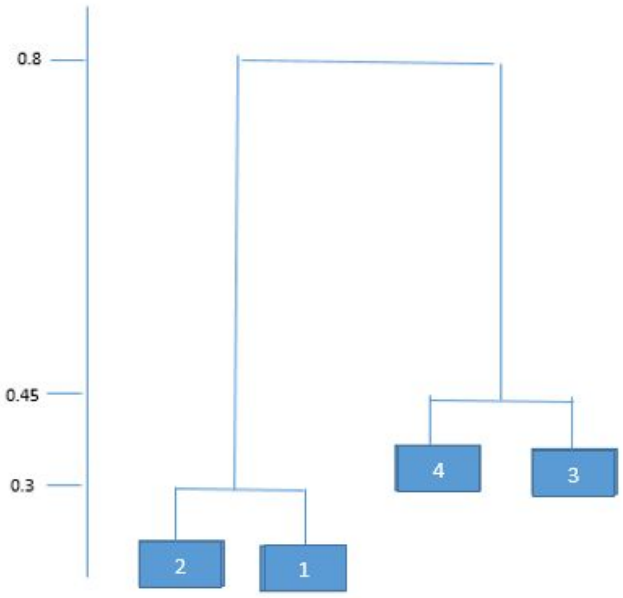
Cluster B(1,2,3) will be fused with 4 and height = 0.45.



c) Cluster 1 = {1, 2}; Cluster 2 = {3, 4}

d) Cluster 1 = {1, 2, 3}; Cluster 2 = {4}

e)



Problem 6:

a) **There is not enough information to tell**. If we consider all the points within the 2 clusters are equi distant to each other then they fuse at the same height, if that is not the case then for complete linkage fusion will occur higher on the tree.

b) Since the distance between the {5} and {6} doesn't change irrespective of the method used to cluster using hierarchical clustering. **{5}, {6} fuse at the same height.**

Problem 7:

```
#Load the data
data(USArrests)
names(USArrests) # Features
```

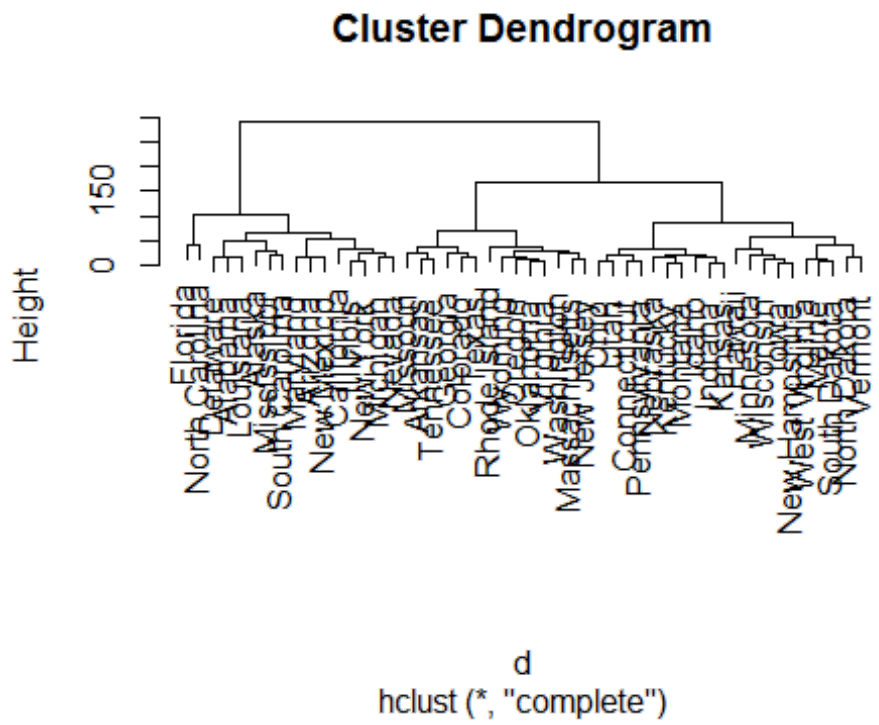
```
## [1] "Murder"   "Assault"  "UrbanPop" "Rape"
```

```
rownames(USArrests) # Observations = States
```

```
##  [1] "Alabama"        "Alaska"         "Arizona"        "Arkansas"
##  [5] "California"     "Colorado"       "Connecticut"    "Delaware"
##  [9] "Florida"        "Georgia"        "Hawaii"         "Idaho"
## [13] "Illinois"       "Indiana"        "Iowa"           "Kansas"
## [17] "Kentucky"       "Louisiana"      "Maine"          "Maryland"
## [21] "Massachusetts"  "Michigan"       "Minnesota"      "Mississippi"
## [25] "Missouri"       "Montana"        "Nebraska"       "Nevada"
## [29] "New Hampshire"  "New Jersey"     "New Mexico"     "New York"
## [33] "North Carolina" "North Dakota"   "Ohio"           "Oklahoma"
## [37] "Oregon"         "Pennsylvania"   "Rhode Island"   "South Carolina"
## [41] "South Dakota"   "Tennessee"      "Texas"          "Utah"
## [45] "Vermont"        "Virginia"       "Washington"     "West Virginia"
## [49] "Wisconsin"      "Wyoming"
```

(a)

```
d<- dist(USArrests, method = "euclidean") # Compute euclidean distant
matrix
h <- hclust(d, method = "complete") # Hierarchical clustering using
complete linkage
plot(h)
```



(b)

```
clusters<-cutree(h,3)

#States in cluster 1
print("States in cluster 1")

## [1] "States in cluster 1"

names(clusters[clusters==1])

##  [1] "Alabama"        "Alaska"         "Arizona"        "California"
##  [5] "Delaware"       "Florida"        "Illinois"       "Louisiana"
##  [9] "Maryland"       "Michigan"       "Mississippi"    "Nevada"
## [13] "New Mexico"     "New York"       "North Carolina" "South Carolina"

#States in cluster 2
print("States in cluster 2")

## [1] "States in cluster 2"

names(clusters[clusters==2])

##  [1] "Arkansas"      "Colorado"      "Georgia"       "Massachusetts"
##  [5] "Missouri"      "New Jersey"    "Oklahoma"      "Oregon"
##  [9] "Rhode Island"  "Tennessee"     "Texas"         "Virginia"
## [13] "Washington"    "Wyoming"
```

```
#States in cluster 3
print("States in cluster 3")
```

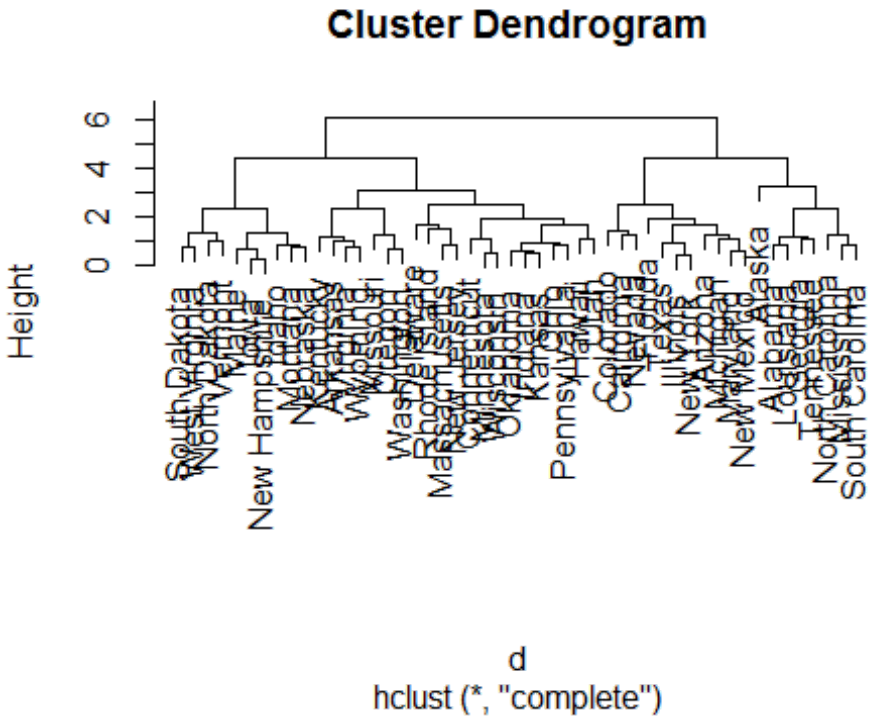
```
## [1] "States in cluster 3"
```

```
names(clusters[clusters==3])
```

```
##  [1] "Connecticut"   "Hawaii"        "Idaho"          "Indiana"
##  [5] "Iowa"          "Kansas"        "Kentucky"       "Maine"
##  [9] "Minnesota"     "Montana"       "Nebraska"       "New Hampshire"
## [13] "North Dakota"  "Ohio"          "Pennsylvania"   "South Dakota"
## [17] "Utah"          "Vermont"       "West Virginia"  "Wisconsin"
```

(c)

```
d<-dist(scale(USArrests), method = "euclidean") #Computing distance matrix after scaling the variables
h<-hclust(d, method = "complete") # Hierarchical clustering using complete linkage
plot(h)
```




```
clusters<-cutree(h,3)
```

```
#States in cluster 1
print("States in cluster 1")
```

```
## [1] "States in cluster 1"
```

```
names(clusters[clusters==1])
```

```
## [1] "Alabama"        "Alaska"         "Georgia"        "Louisiana"
## [5] "Mississippi"    "North Carolina" "South Carolina" "Tennessee"
```

```
#States in cluster 2
print("States in cluster 2")
```

```
## [1] "States in cluster 2"
```

```
names(clusters[clusters==2])
```

```
##  [1] "Arizona"    "California" "Colorado"   "Florida"    "Illinois"
##  [6] "Maryland"   "Michigan"   "Nevada"     "New Mexico" "New York"
## [11] "Texas"
```

```
#States in cluster 3
print("States in cluster 3")
```

```
## [1] "States in cluster 3"
```

```
names(clusters[clusters==3])
```

```
##  [1] "Arkansas"      "Connecticut"   "Delaware"      "Hawaii"
##  [5] "Idaho"         "Indiana"       "Iowa"          "Kansas"
##  [9] "Kentucky"      "Maine"         "Massachusetts" "Minnesota"
## [13] "Missouri"      "Montana"       "Nebraska"      "New Hampshire"
## [17] "New Jersey"    "North Dakota"  "Ohio"          "Oklahoma"
## [21] "Oregon"        "Pennsylvania"  "Rhode Island"  "South Dakota"
## [25] "Utah"          "Vermont"       "Virginia"      "Washington"
## [29] "West Virginia" "Wisconsin"     "Wyoming"
```

(d)

Scaling the variables before preforming hierarchical clustering results in 3 clusters with different states in each one of them compared to hierarchical clusters without scaling.

In this case it makes sense to scale the variables for 2 reasons:

1. Unit of measurement of UrbanPop is different than that of Rape, Assault and Murder.
2. Variance of Assault and UrbanPop is much higher hence has larger effect on computing dissimilarities between states.

```
?USArrests
```

```
## starting httpd help server ... done
```

```
var(USArrests$Murder)
```

```
## [1] 18.97
```

```
var(USArrests$Assault)
```

```
## [1] 6945
```

```
var(USArrests$UrbanPop)
```

```
## [1] 209.5
```

```
var(USArrests$Rape)
```

```
## [1] 87.73
```

Problem 8:

a) We expect training RSS for cubic regression to be lower than that of linear regression as the cubic regression over fits the data.

b) Since the true relationship between X and Y is linear, linear regression provides a better fit than the fit provided by cubic regression. Hence Test RSS will be lower for linear regression than for cubic regression.

c) We expect training RSS for cubic regression to be lower than that of linear regression as the cubic regression provides better fit to the non-linearity in the data.

d) Since the true relationship between X and Y is not linear and we don't know how far it is from being linear it is difficult to answer if Test RSS for linear model will be small or for cubic regression. If true relationship is close to being linear then Test RSS for linear regression would be lower. If true relationship is far from linear then Test RSS for cubic regression would be lower.

Problem 9:

Load the data

```
library(ISLR)
data(Auto)
?Auto
```

```
## starting httpd help server ... done
```

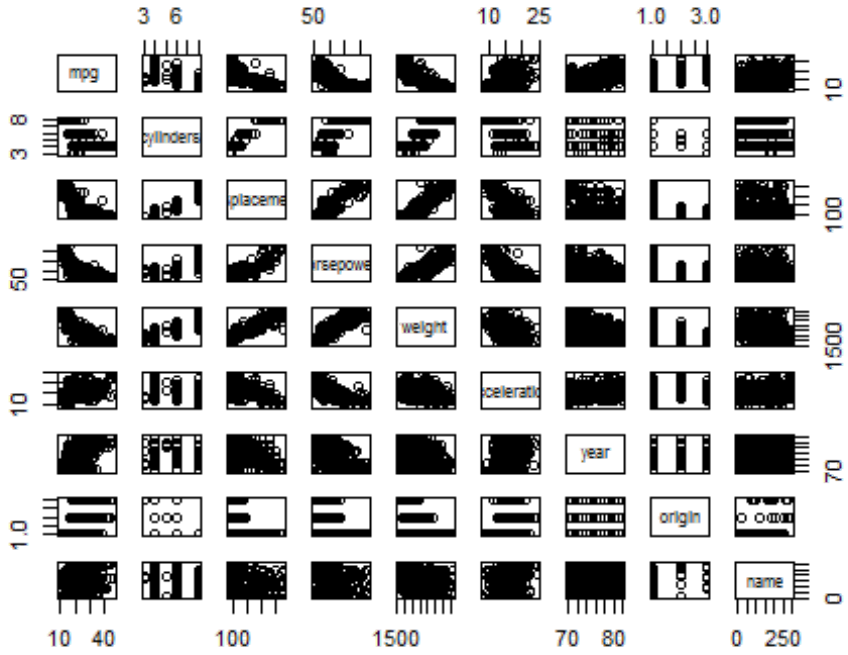
```
str(Auto)
```

```
## 'data.frame':    392 obs. of  9 variables:
##  $ mpg         : num  18 15 18 16 17 15 14 14 14 15 ...
##  $ cylinders   : num  8 8 8 8 8 8 8 8 8 8 ...
##  $ displacement: num  307 350 318 304 302 429 454 440 455 390 ...
##  $ horsepower  : num  130 165 150 150 140 198 220 215 225 190 ...
##  $ weight      : num  3504 3693 3436 3433 3449 ...
##  $ acceleration: num  12 11.5 11 12 10.5 10 9 8.5 10 8.5 ...
##  $ year        : num  70 70 70 70 70 70 70 70 70 70 ...
##  $ origin      : num  1 1 1 1 1 1 1 1 1 1 ...
```

```
##  $ name         : Factor w/ 304 levels "amc ambassador brougham",..: 49
36 231 14 161 141 54 223 241 2 ...
```

(a) Scatter plot matrix

```
pairs(Auto)
```




```
#It appears that Origin data is qualitative. But coded as quantitative. We
need to change its type.
```

```
Auto$origin<-as.factor(Auto$origin)
```

(b) correlation matrix

```
cor(Auto[,-c(8:9)])
```

```
##                     mpg cylinders displacement horsepower  weight
## mpg           1.0000   -0.7776      -0.8051    -0.7784 -0.8322
## cylinders    -0.7776    1.0000       0.9508     0.8430  0.8975
## displacement -0.8051    0.9508       1.0000     0.8973  0.9330
## horsepower   -0.7784    0.8430       0.8973     1.0000  0.8645
## weight       -0.8322    0.8975       0.9330     0.8645  1.0000
## acceleration  0.4233   -0.5047      -0.5438    -0.6892 -0.4168
## year          0.5805   -0.3456      -0.3699    -0.4164 -0.3091
##              acceleration    year
## mpg                0.4233  0.5805
## cylinders         -0.5047 -0.3456
```

```
## displacement      -0.5438 -0.3699
## horsepower        -0.6892 -0.4164
## weight            -0.4168 -0.3091
## acceleration       1.0000  0.2903
## year               0.2903  1.0000
```

(c)

i. F-statistic is much greater than 1 and p-value associated with it smaller than 0.05. So at 0.05 significance level, we can say at least one of the predictors is related in estimating MPG.

ii. Displacement, weight, year, origin2 and origin3 have statistically significant relationship to the response mpg.

iii. On an average mpg increases by 7.770e-01 units for one unit increase in the year, holding all other predictors fixed.

```
fit <-lm(mpg ~ . - name, data = Auto)
summary(fit)
```

```
##
## Call:
## lm(formula = mpg ~ . - name, data = Auto)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.009 -2.078 -0.098  1.986 13.361
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.80e+01   4.68e+00   -3.84  0.00014 ***
## cylinders    -4.90e-01   3.21e-01   -1.52  0.12821
## displacement  2.40e-02   7.65e-03    3.13  0.00186 **
## horsepower   -1.82e-02   1.37e-02   -1.33  0.18549
## weight       -6.71e-03   6.55e-04  -10.24  < 2e-16 ***
## acceleration  7.91e-02   9.82e-02    0.81  0.42110
## year          7.77e-01   5.18e-02   15.01  < 2e-16 ***
## origin2       2.63e+00   5.66e-01    4.64  4.7e-06 ***
## origin3       2.85e+00   5.53e-01    5.16  3.9e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.31 on 383 degrees of freedom
## Multiple R-squared:  0.824,  Adjusted R-squared:  0.821
## F-statistic:  224 on 8 and 383 DF,  p-value: <2e-16
```

(d)

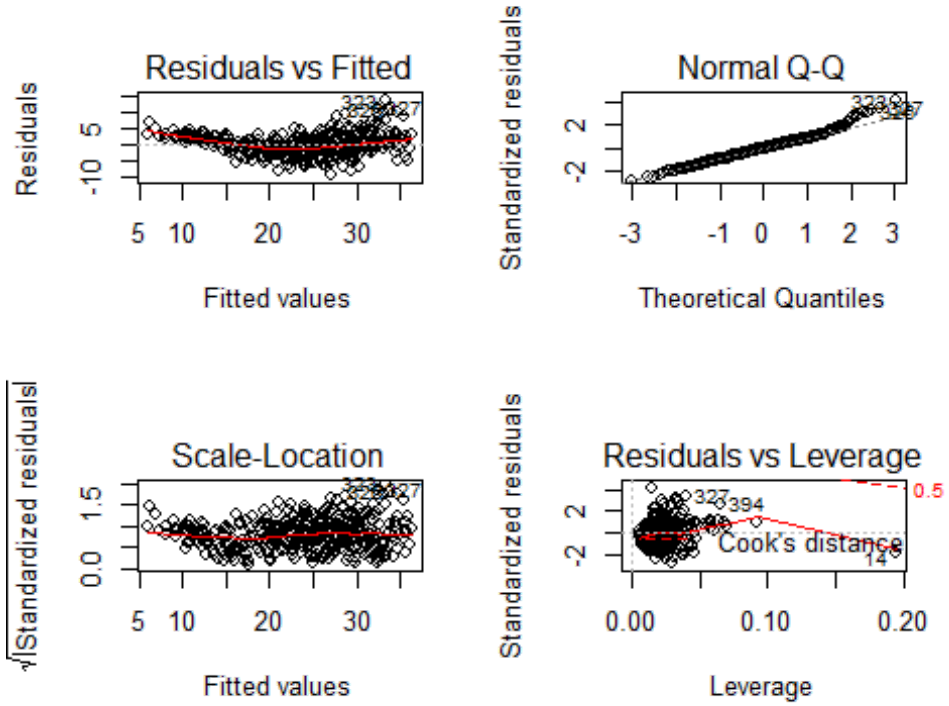
i. Residual vs fitted plots exhibits a clear U-Shape which provides strong indication of non-linearity in the data.

ii. There is a clear funnel shape in the residual plot indicating heteroscedasticity

iii. There are no observation greater than absolute value of 3 in Studentized residual vs fitted plot. Hence there are no outliers

iv. Observation 14 is a high leverage point based on the studentized residual vs leverage plot.

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




```
which.max(hatvalues(fit))
```

```
## 14
## 14
```

(e)

I have included interaction terms between, weight and acceleration, cylinder and year and finally between cylinder and horespower. From the summary below, interaction between cylinder:year and cylinder:horspower is statistically significant.

```
summary(lm(mpg~ . -name +weight*acceleration + cylinders:year +
cylinders:horsepower, data=Auto))
```

```
##
## Call:
## lm(formula = mpg ~ . - name + weight * acceleration + cylinders:year +
##     cylinders:horsepower, data = Auto)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -7.697 -1.613 -0.059  1.257 12.207
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -4.31e+01   1.58e+01   -2.73  0.00668 **
## cylinders             4.81e+00   2.62e+00    1.84  0.06658 .
## displacement         -1.33e-03   7.14e-03   -0.19  0.85197
## horsepower           -2.48e-01   3.69e-02   -6.74  6.0e-11 ***
## weight               -2.04e-03   1.68e-03   -1.21  0.22561
## acceleration          2.97e-01   2.91e-01    1.02  0.30832
## year                  1.29e+00   1.63e-01    7.90  3.0e-14 ***
## origin2               1.74e+00   5.15e-01    3.39  0.00078 ***
## origin3               1.84e+00   4.96e-01    3.71  0.00024 ***
## weight:acceleration  -1.43e-04   9.72e-05   -1.47  0.14218
## cylinders:year       -1.03e-01   3.05e-02   -3.38  0.00081 ***
## cylinders:horsepower  2.95e-02   5.08e-03    5.80  1.4e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.88 on 380 degrees of freedom
## Multiple R-squared:  0.867, Adjusted R-squared:  0.863
## F-statistic:  226 on 11 and 380 DF, p-value: <2e-16
```

(F)

I have added 2 trasformations, square of displacement and log of the year variables in addition to the rest. This increases the proportion of variance explained from 82.42% to 86.81% which is a huge improvement from the previous model with no transformations. Further the summary indicates that these trasformations are statistically significant.

```
summary(lm(mpg ~.-name, data = Auto))
```

```
##
## Call:
## lm(formula = mpg ~ . - name, data = Auto)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.009 -2.078 -0.098  1.986 13.361
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  -1.80e+01   4.68e+00   -3.84  0.00014 ***
## cylinders    -4.90e-01   3.21e-01   -1.52  0.12821
## displacement  2.40e-02   7.65e-03    3.13  0.00186 **
## horsepower   -1.82e-02   1.37e-02   -1.33  0.18549
## weight       -6.71e-03   6.55e-04  -10.24  < 2e-16 ***
## acceleration  7.91e-02   9.82e-02    0.81  0.42110
## year          7.77e-01   5.18e-02   15.01  < 2e-16 ***
## origin2       2.63e+00   5.66e-01    4.64  4.7e-06 ***
## origin3       2.85e+00   5.53e-01    5.16  3.9e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.31 on 383 degrees of freedom
## Multiple R-squared:  0.824, Adjusted R-squared:  0.821
## F-statistic:  224 on 8 and 383 DF, p-value: <2e-16
```

```
summary(lm(mpg~ . -name + I(displacement ^ 2)  + I(log(year)), data =
Auto))
```

```
##
## Call:
## lm(formula = mpg ~ . - name + I(displacement^2) + I(log(year)),
##     data = Auto)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.615 -1.540  0.103  1.471 11.863
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         2.95e+03   4.77e+02    6.18  1.7e-09 ***
## cylinders           8.77e-01   3.08e-01    2.85   0.0046 **
## displacement       -1.06e-01   1.49e-02   -7.12  5.5e-12 ***
## horsepower         -7.22e-02   1.28e-02   -5.62  3.7e-08 ***
## weight             -3.43e-03   6.40e-04   -5.36  1.5e-07 ***
## acceleration       -5.87e-02   8.62e-02   -0.68   0.4965
## year                1.25e+01   1.89e+00    6.60  1.3e-10 ***
## origin2             8.65e-01   5.35e-01    1.62   0.1068
## origin3             9.10e-01   5.26e-01    1.73   0.0844 .
## I(displacement^2)   2.04e-04   2.18e-05    9.36  < 2e-16 ***
## I(log(year))       -8.88e+02   1.43e+02   -6.20  1.5e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.87 on 381 degrees of freedom
## Multiple R-squared:  0.868,  Adjusted R-squared:  0.865
## F-statistic:  251 on 10 and 381 DF,  p-value: <2e-16
```

Problem 10:

(a) Form of the linear model is: y = b0+ b1 * x1 + b2 * x2 + e Regression coefficients are: b0 = 2 ; b1 = 2; b2 = 0.3

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

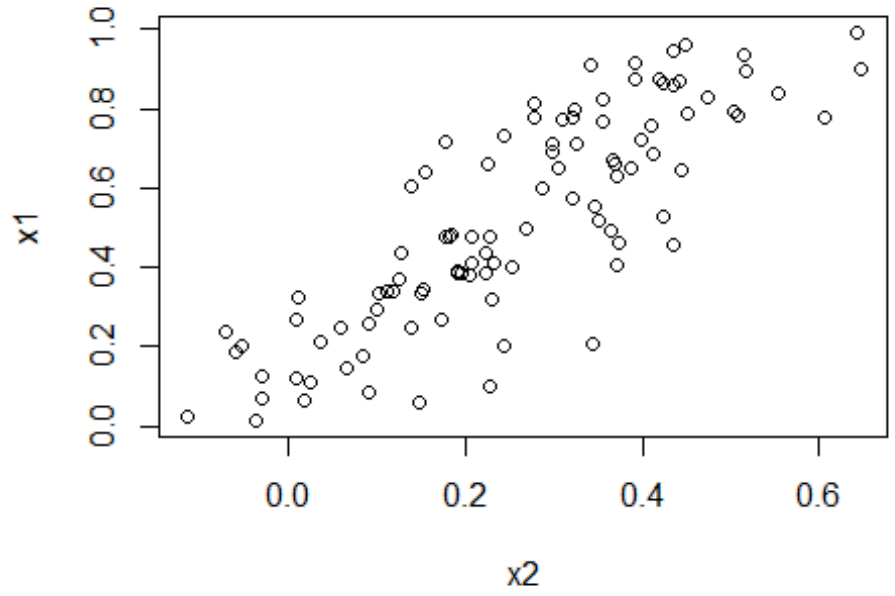
(b)

Correlation between x1 and x2 is 0.835.

```
cor(x1,x2)
```

```
## [1] 0.8351
```

```
#Scatter plot matrix displaying relationship between the variables x1 and x2
plot(x1~x2)
```



(C)

x1 and x2 together explain 20.24% of the variability in y. F-statistic is greater than 1 and p-value associated with it is significant indication at least one predictor is related to the response y.

b0_hat = 2.1305. Is slightly greater than b0 b1_hat = 1.4396. Is less than b1 b2_hat = 1.0097. Is significantly greater than b2

Can reject the H0: b1 = 0 since p-value associated with b1_hat is significant.

Can't reject the H0 : b2 = 0 since p-value associated with b2_hat is not significant.

```
fit1 = lm(y~ x1+x2)
summary(fit1)
```

```
##
## 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)    2.130      0.232    9.19  7.6e-15 ***
## x1             1.440      0.721    2.00    0.049 *
## x2             1.010      1.134    0.89    0.375
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.06 on 97 degrees of freedom
## Multiple R-squared:  0.209,  Adjusted R-squared:  0.193
## F-statistic: 12.8 on 2 and 97 DF,  p-value: 1.16e-05
```

(d)

p-value associated with x1 is significant and this model explains 20.24% of the variability y. Both b0_hat and b1_hat are similar to b0 and b1.

Can reject H0: b1 = 0

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

```
##
## Call:
## lm(formula = y ~ x1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8950 -0.6687 -0.0779  0.5922  2.4556
##
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.112      0.231    9.15  8.3e-15 ***
## x1             1.976      0.396    4.99  2.7e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.06 on 98 degrees of freedom
## Multiple R-squared:  0.202,  Adjusted R-squared:  0.194
## F-statistic: 24.9 on 1 and 98 DF,  p-value: 2.66e-06
```

(e) p-value associated with x1 is significant and this model explains 17.63% variability in y. Both b0_hat and b2_hat are greater than b0 and b2.

can reject H0 : b2 =0

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

```
##
## Call:
## lm(formula = y ~ x2)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -2.627 -0.752 -0.036  0.724  2.449
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.390      0.195   12.26  < 2e-16 ***
## x2             2.900      0.633    4.58  1.4e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.07 on 98 degrees of freedom
## Multiple R-squared:  0.176,  Adjusted R-squared:  0.168
## F-statistic:   21 on 1 and 98 DF,  p-value: 1.37e-05
```

(f)

Yes the results in (c) - (e) contradict each other. In (c) we said we can't reject the H0: b2 = 0 and in (e) we said we can reject H0:b2 = 0. This is because, x1 and x2 are highly correlated variables with cor > 0.8. Because of this the presence of x1 is masking the presence of x2 and hence in (c) we saw that b1_hat is significant and b2_hat not being significant. When we removed x1 from the model in (e) since the highly correlated variable x1is removed, we saw that b2 is significant too.

(g)

In model (C) adding the new observation improves the R2 statistic and now, we can't reject H0: b1 = 0 but we can reject H0: b2 = 0. Observation 101 is a high leverage point in this model and not an outlier.

In model (d) adding the new observation reduced the R2 statistic quite significantly. The observation 101 is neither an outlier nor a leverage point.

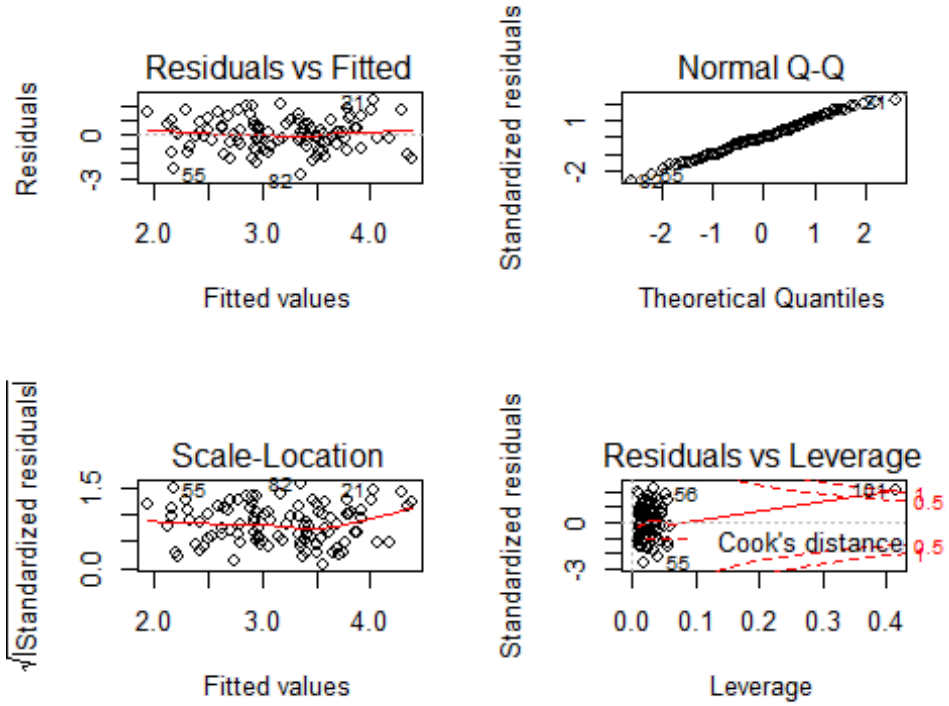
In model (e) adding the new observation significantly improves R2 statistic and the observation 101 is a high leverage point.

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

fit1<- lm(y~x1+x2)
summary(fit1)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.7335 -0.6932 -0.0526  0.6638  2.3062
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.227      0.231    9.62  7.9e-16 ***
## x1             0.539      0.592    0.91   0.3646
## x2             2.515      0.898    2.80   0.0061 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.07 on 98 degrees of freedom
## Multiple R-squared:  0.219,  Adjusted R-squared:  0.203
## F-statistic: 13.7 on 2 and 98 DF,  p-value: 5.56e-06
```

```
par(mfrow=c(2,2))
plot(fit1)
```




```
which.max(hatvalues(fit1))
```

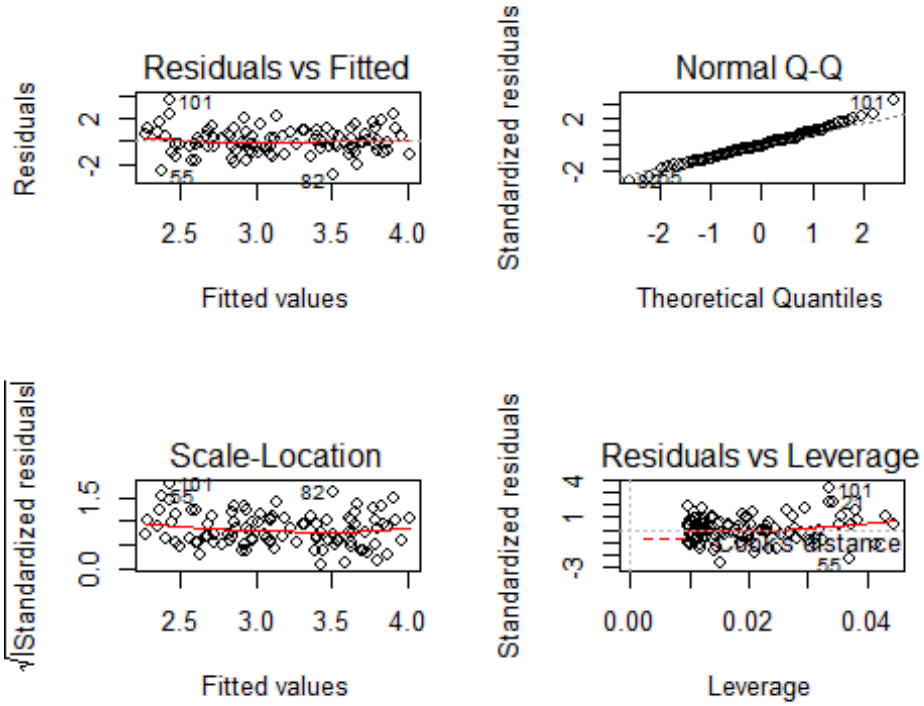
```
## 101
## 101
```

```
fit2<- lm(y~x1)
summary(fit2)
```

```
##
## Call:
## lm(formula = y ~ x1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.890 -0.656 -0.091  0.568  3.567
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.257      0.239    9.44  1.8e-15 ***
## x1             1.766      0.412    4.28  4.3e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.11 on 99 degrees of freedom
```

```
## Multiple R-squared:  0.156,  Adjusted R-squared:  0.148
## F-statistic: 18.3 on 1 and 99 DF,  p-value: 4.29e-05
```

```
par(mfrow=c(2,2))
plot(fit2)
```




```
which.max(hatvalues(fit2))
```

```
## 27
## 27
```

```
fit3<- lm(y~x2)
summary(fit3)
```

```
##
## Call:
## lm(formula = y ~ x2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.647 -0.710 -0.069  0.727  2.381
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.345      0.191   12.26  < 2e-16 ***
## x2             3.119      0.604    5.16  1.3e-06 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.07 on 99 degrees of freedom
## Multiple R-squared:  0.212,  Adjusted R-squared:  0.204
## F-statistic: 26.7 on 1 and 99 DF,  p-value: 1.25e-06
```

```
par(mfrow=c(2,2))
which.max(hatvalues(fit3))
```

```
## 101
## 101
```

```
plot(fit3)
```