Multicollinearity 10.5 p406

Definition

Multicollinearity exists when two or more independent variables used in regression are moderately or highly correlated.

- when multicollinearity exists, regression results can be confusing and misleading.

For example in a multiple regression model all partial slopes will be significant with a significant global F-test.

Signs of the regression coefficients might not make sense.

Row	x	x-sq	x-m	(x-m)-sq
1	1	1	-4.5	20.25
2	2	4	-3.5	12.25
3	3	9	-2.5	6.25
4	4	16	-1.5	2.25
5	5	25	-0.5	0.25
6	6	36	0.5	0.25
7	7	49	1.5	2.25
8	8	64	2.5	6.25
9	9	81	3.5	12.25
10	10	100	4.5	20.25

Correlation of x and x-sq = 0.975, P-Value = 0.000

Correlation of x-m and (x-m)-sq = 0.000, P-Value = 1.000

- Variance inflation factors

Tolerance : $T_i = 1 - R_i^2$

$$VIF_i = \frac{1}{T_i} = \frac{1}{1 - R_i^2}$$

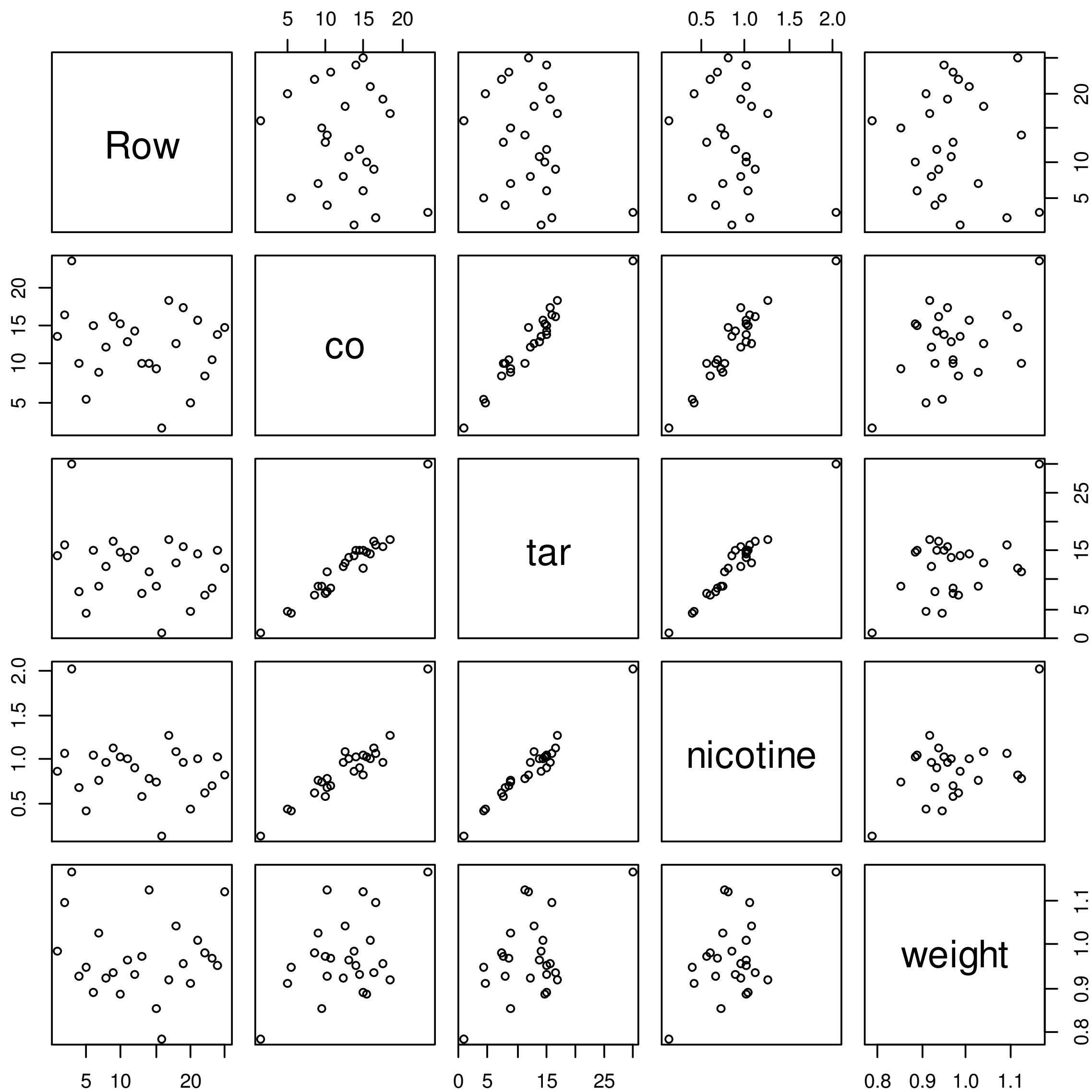
$$VIF_i = \left(\mathbf{r}_{XX}^{-1}\right)_{ii}$$

Example

R doe

```
#Multicolinearity
data=read.table("C:/Users/Mihinda/Desktop/multico.txt",
header=1) #the data file
data
plot(data)
cor(data)
fit <- lm(co ~ tar + nicotine + weight, data=data)
summary(fit)
anova(fit)
fittar <- lm(tar ~ nicotine + weight, data=data)
fitnic <- lm(nicotine ~ tar +  weight, data=data)
fitwt <- lm( weight ~nicotine + tar, data=data)
summary(fittar)
summary(fitnic)
summary(fitwt)
library(car)
vif(fit)
```

```
> #Multicolinearity
> data=read.table("C:/Users/Mihinda/Desktop/multico.txt",
header=1) #the data file
> data
   Row   co  tar nicotine weight
1    1 13.6 14.1     0.86 0.9853
2    2 16.6 16.0     1.06 1.0938
3    3 23.5 29.8     2.03 1.1650
4    4 10.2  8.0     0.67 0.9280
5    5  5.4  4.1     0.40 0.9462
6    6 15.0 15.0     1.04 0.8885
7    7  9.0  8.8     0.76 1.0267
8    8 12.3 12.4     0.95 0.9225
9    9 16.3 16.6     1.12 0.9372
10  10 15.4 14.9     1.02 0.8858
11  11 13.0 13.7     1.01 0.9643
12  12 14.4 15.1     0.90 0.9316
13  13 10.0  7.8     0.57 0.9705
14  14 10.2 11.4     0.78 1.1240
15  15  9.5  9.0     0.74 0.8517
16  16  1.5  1.0     0.13 0.7851
17  17 18.5 17.0     1.26 0.9186
18  18 12.6 12.8     1.08 1.0395
19  19 17.5 15.8     0.96 0.9573
20  20  4.9  4.5     0.42 0.9106
21  21 15.9 14.5     1.01 1.0070
22  22  8.5  7.3     0.61 0.9806
23  23 10.6  8.6     0.69 0.9693
24  24 13.9 15.2     1.02 0.9496
25  25 14.9 12.0     0.82 1.1184
> plot(data)
```




```
> cor(data)
                 Row         co        tar   nicotine      weight
Row       1.00000000 -0.1632833 -0.2516034 -0.2436872 -0.06919783
co       -0.16328334  1.0000000  0.9574853  0.9259473  0.46395920
tar      -0.25160336  0.9574853  1.0000000  0.9766076  0.49076541
nicotine -0.24368724  0.9259473  0.9766076  1.0000000  0.50018272
weight   -0.06919783  0.4639592  0.4907654  0.5001827  1.00000000
```

```
> fit <- lm(co ~ tar + nicotine + weight, data=data)
> summary(fit)

Call:
lm(formula = co ~ tar + nicotine + weight, data = data)


Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   3.2022     3.4618   0.925 0.365464
tar           0.9626     0.2422   3.974 0.000692 ***
nicotine     -2.6317     3.9006  -0.675 0.507234
weight       -0.1305     3.8853  -0.034 0.973527
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.446 on 21 degrees of freedom
Multiple R-squared: 0.9186,     Adjusted R-squared: 0.907
F-statistic: 78.98 on 3 and 21 DF,  p-value: 1.329e-11

> anova(fit)
Analysis of Variance Table

Response: co
          Df Sum Sq Mean Sq  F value    Pr(>F)
tar        1 494.28  494.28 236.4843 6.651e-13 ***
nicotine   1   0.97    0.97   0.4661    0.5023
weight     1   0.00    0.00   0.0011    0.9735
Residuals 21  43.89    2.09
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> fittar <- lm(tar ~ nicotine + weight, data=data)
> fitnic <- lm(nicotine ~ tar +  weight, data=data)
> fitwt <- lm( weight ~nicotine + tar, data=data)
> summary(fittar)

Call:
lm(formula = tar ~ nicotine + weight, data = data)


Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -1.6500     3.0263  -0.545    0.591
nicotine     15.6038     0.8472  18.419 7.39e-15 ***
weight        0.1967     3.4193   0.058    0.955
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 1.272 on 22 degrees of freedom
Multiple R-squared: 0.9538,     Adjusted R-squared: 0.9496
F-statistic: 226.9 on 2 and 22 DF,  p-value: 2.062e-15

> summary(fitnic)

Call:
lm(formula = nicotine ~ tar + weight, data = data)

Residuals:
      Min        1Q    Median        3Q       Max
-0.145674 -0.050798  0.005133  0.052035  0.160762

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.033384   0.189082   0.177    0.861
tar         0.060184   0.003268  18.419 7.39e-15 ***
weight      0.111105   0.211044   0.526    0.604
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.07902 on 22 degrees of freedom
Multiple R-squared: 0.9543,     Adjusted R-squared: 0.9502
F-statistic: 229.9 on 2 and 22 DF,  p-value: 1.799e-15

> summary(fitwt)

Call:
lm(formula = weight ~ nicotine + tar, data = data)

Residuals:
      Min        1Q    Median        3Q       Max
-0.102616 -0.056166 -0.002679  0.043905  0.165134

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.8628078  0.0473886  18.207  9.4e-15 ***
nicotine    0.1119774  0.2127000   0.526    0.604
tar         0.0007645  0.0132917   0.058    0.955
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.07933 on 22 degrees of freedom
Multiple R-squared: 0.2503,     Adjusted R-squared: 0.1821
F-statistic: 3.672 on 2 and 22 DF,  p-value: 0.04205
```

```
> library(car)
Loading required package: MASS
Loading required package: nnet
> vif(fit)
      tar  nicotine    weight
21.630706 21.899917  1.333859
```

```
>
```

$$\mathbf{r}_{XX} = \begin{bmatrix} 1.00000 & 0.97661 & 0.49077 \\ 0.97661 & 1.00000 & 0.50018 \\ 0.49077 & 0.50018 & 1.00000 \end{bmatrix}$$

$$\mathbf{r}_{XX}^{-1} = \begin{bmatrix} 21.6307 & -21.0918 & -0.06586 \\ -21.0918 & 21.8999 & -0.60285 \\ -0.0659 & -0.6028 & 1.33386 \end{bmatrix}$$

Outliers 10.2 p390

Deleted residuals $d_i = y_i - \hat{y}_{(i)}$

y_i = *i*th observed response

$\hat{y}_{(i)}$ = the predicted value of the ith repose when the data for the ith observation is deleted from the analysis.

Studentized deleted residuals p396

$$t_i = \frac{d_i}{s\{d_i\}}$$

$s^2\{d_i\} = \dfrac{MSE_i}{1-h_{ii}}$ and

$$t_i = \frac{d_i}{s\{d_i\}} = e_i \left[\frac{n-p-1}{MSE(1-h_{ii})-e_i^2} \right]^{1/2}$$

The studentized deleted residual t_i has a distribution that is approximated by a t-distribution with (*n*-1)-*p* d.f.

The appropriate Bonferroni critical value therefore is $t(1-\alpha/2n,\ n-1-p)$

(p396)

(*n*-1)-p= (24-1)-(4+1) = 18 and

$t_{(1-0.025/24,18)}$ = 3.59213

Example

```
#Outliers etc
data=read.table("C:/Users/Mihinda/Desktop/outliers.txt",
header=1) #the data file
data
fit <- lm(sales ~ city1 + city2 + city3 + traffic , data=data)
summary(fit)
anova(fit)
data$tres <- rstudent(fit)
data
X <- model.matrix(fit)
X
H = X%*%solve(t(X)%*%X)%*%t(X) # Hat matrix
l=diag(H)
I
#Another way to calculate leverages
Lv=hat(X)
Lv
plot(Lv)
cookD <- cooks.distance(fit)
sort(Lv)
cookD
```

```
> #Outliers etc
> data=read.table("C:/Users/Mihinda/Desktop/outliers.txt",
header=1) #the data file
> data
   Row city traffic sales city1 city2 city3
1    1    1    59.3   6.3     1     0     0
2    2    1    60.3   6.6     1     0     0
3    3    1    82.1   7.6     1     0     0
4    4    1    32.3   3.0     1     0     0
5    5    1    98.0   9.5     1     0     0
6    6    1    54.1   5.9     1     0     0
7    7    1    54.4   6.1     1     0     0
8    8    1    51.3   5.0     1     0     0
9    9    1    36.7   3.6     1     0     0
10  10    2    23.6   2.8     0     1     0
11  11    2    57.6   6.7     0     1     0
12  12    2    44.6   5.2     0     1     0
13  13    3    75.8  82.0     0     0     1
14  14    3    48.3   5.0     0     0     1
15  15    3    41.4   3.9     0     0     1
16  16    3    52.5   5.4     0     0     1
17  17    3    41.0   4.1     0     0     1
18  18    3    29.6   3.1     0     0     1
19  19    3    49.5   5.4     0     0     1
20  20    4    73.1   8.4     0     0     0
21  21    4    81.3   9.5     0     0     0
22  22    4    72.4   8.7     0     0     0
23  23    4    88.4  10.6     0     0     0
24  24    4    23.2   3.3     0     0     0
> fit <- lm(sales ~ city1 + city2 + city3 + traffic , data=data)
> summary(fit)

Call:
lm(formula = sales ~ city1 + city2 + city3 + traffic, data =
data)

Residuals:
    Min      1Q  Median      3Q     Max
-11.681  -7.331  -1.390   1.719  56.464

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -16.4592    13.1640  -1.250   0.2264
city1         1.1061     8.4226   0.131   0.8969
city2         6.1428    11.6800   0.526   0.6050
city3        14.4896     9.2884   1.560   0.1353
traffic       0.3629     0.1679   2.161   0.0437 *
```

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 14.86 on 19 degrees of freedom
Multiple R-squared: 0.2595,     Adjusted R-squared: 0.1036
F-statistic: 1.665 on 4 and 19 DF,  p-value: 0.1996

> anova(fit)
Analysis of Variance Table

Response: sales
          Df Sum Sq Mean Sq F value  Pr(>F)
city1      1  139.8  139.75  0.6331 0.43606
city2      1  136.8  136.81  0.6197 0.44086
city3      1  162.2  162.19  0.7347 0.40203
traffic    1 1031.0 1031.01  4.6705 0.04366 *
Residuals 19 4194.2  220.75
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> data$tres <- rstudent(fit)
> data
   Row city traffic sales city1 city2 city3          tres
1    1    1    59.3   6.3     1     0     0   0.009365649
2    2    1    60.3   6.6     1     0     0   0.004997610
3    3    1    82.1   7.6     1     0     0  -0.498420553
4    4    1    32.3   3.0     1     0     0   0.489073499
5    5    1    98.0   9.5     1     0     0  -0.860562322
6    6    1    54.1   5.9     1     0     0   0.112897652
7    7    1    54.4   6.1     1     0     0   0.119224112
8    8    1    51.3   5.0     1     0     0   0.121277869
9    9    1    36.7   3.6     1     0     0   0.407866026
10  10    2    23.6   2.8     0     1     0   0.379142283
11  11    2    57.6   6.7     0     1     0  -0.320213731
12  12    2    44.6   5.2     0     1     0  -0.053609645
13  13    3    75.8  82.0     0     0     1 179.310135591
14  14    3    48.3   5.0     0     0     1  -0.758872162
15  15    3    41.4   3.9     0     0     1  -0.657757473
16  16    3    52.5   5.4     0     0     1  -0.843855199
17  17    3    41.0   4.1     0     0     1  -0.632668636
18  18    3    29.6   3.1     0     0     1  -0.414146872
19  19    3    49.5   5.4     0     0     1  -0.761586510
20  20    4    73.1   8.4     0     0     0  -0.122417349
21  21    4    81.3   9.5     0     0     0  -0.263888726
22  22    4    72.4   8.7     0     0     0  -0.081662233
23  23    4    88.4  10.6     0     0     0  -0.382413963
24  24    4    23.2   3.3     0     0     0   1.033584706
```

Leverage values 10.3 p398

- h_{ii} = Leverage of the *i*th observation.

The leverage values are the diagonal elements of the hat matrix $H = X(X'X)^{-1}X'$

Note: $Var(e_i) = Var\left(Y_i - \hat{Y}_i\right) = \sigma^2(1 - h_{ii})$

$\Rightarrow h_{ii} \le 1$

$h_{ii} \simeq 1 \Rightarrow Var\left(Y_i - \hat{Y}_i\right) \simeq 0 \Rightarrow Y_i \simeq \hat{Y}_i$

This implies that the fitted line will pass very close to the data point $(X_{i1}, X_{i2}, \ldots, X_{i\,p-1}, Y_i)$. We say that case *i* exerts high leverage on the fitted line.

Observations with

$$h_{ii} > \frac{2p}{n}$$

are considered by this rule to indicate outlying cases with regard to their X values.

Note $\frac{p}{n}$ is the average leverage.

```
> data$hii=hat(X)
> data
   Row city traffic sales city1 city2 city3         tres       hii
1    1    1    59.3   6.3     1     0     0   0.009365649 0.1111537
2    2    1    60.3   6.6     1     0     0   0.004997610 0.1114290
3    3    1    82.1   7.6     1     0     0  -0.498420553 0.1809105
4    4    1    32.3   3.0     1     0     0   0.489073499 0.2002740
5    5    1    98.0   9.5     1     0     0  -0.860562322 0.3081442
6    6    1    54.1   5.9     1     0     0   0.112897652 0.1138398
7    7    1    54.4   6.1     1     0     0   0.119224112 0.1134971
8    8    1    51.3   5.0     1     0     0   0.121277869 0.1181469
9    9    1    36.7   3.6     1     0     0   0.407866026 0.1730506
10  10    2    23.6   2.8     0     1     0   0.379142283 0.3762601
11  11    2    57.6   6.7     0     1     0  -0.320213731 0.3646805
12  12    2    44.6   5.2     0     1     0  -0.053609645 0.3342415
13  13    3    75.8  82.0     0     0     1 179.310135591 0.2394424
14  14    3    48.3   5.0     0     0     1  -0.758872162 0.1428571
15  15    3    41.4   3.9     0     0     1  -0.657757473 0.1489377
16  16    3    52.5   5.4     0     0     1  -0.843855199 0.1451101
17  17    3    41.0   4.1     0     0     1  -0.632668636 0.1496631
18  18    3    29.6   3.1     0     0     1  -0.414146872 0.1875182
19  19    3    49.5   5.4     0     0     1  -0.761586510 0.1430411
20  20    4    73.1   8.4     0     0     0  -0.122417349 0.2037518
21  21    4    81.3   9.5     0     0     0  -0.263888726 0.2236919
22  22    4    72.4   8.7     0     0     0  -0.081662233 0.2028453
23  23    4    88.4  10.6     0     0     0  -0.382413963 0.2548308
24  24    4    23.2   3.3     0     0     0   1.033584706 0.4526824
```

Cook's distance, p402

- A measure of the overall influence of an observation on the estimated β coefficients.

Cook's distance: $D_i = \frac{(y_i - \hat{y}_i)^2}{pMSE}\left[\frac{h_{ii}}{(1-h_{ii})^2}\right]$

-Note that D_i depends on both the residual $(y_i - \hat{y}_i)$ and the leverage h_{ii}.

- A large value of D_i indicates that the ith observation has a strong influence on the estimated β coefficients.
- Values of D_i can be compared to the values of the F(p, n-p)

Usually an observation that falls above the 50^{th} percentile of the F distribution is considered to be an influential observation.

In fast-food sales example n = 24, p = 5

numerator d. f. = 5 and denominator d. f = 24 - 5 = 19 and $F_{0.50}$ = 0.9020

```
> data$cookD <- cooks.distance(fit)
> data
   Row city traffic sales city1 city2 city3          tres       hii
1    1    1    59.3   6.3     1     0     0   0.009365649 0.1111537
2    2    1    60.3   6.6     1     0     0   0.004997610 0.1114290
3    3    1    82.1   7.6     1     0     0  -0.498420553 0.1809105
4    4    1    32.3   3.0     1     0     0   0.489073499 0.2002740
5    5    1    98.0   9.5     1     0     0  -0.860562322 0.3081442
6    6    1    54.1   5.9     1     0     0   0.112897652 0.1138398
7    7    1    54.4   6.1     1     0     0   0.119224112 0.1134971
8    8    1    51.3   5.0     1     0     0   0.121277869 0.1181469
9    9    1    36.7   3.6     1     0     0   0.407866026 0.1730506
10  10    2    23.6   2.8     0     1     0   0.379142283 0.3762601
11  11    2    57.6   6.7     0     1     0  -0.320213731 0.3646805
12  12    2    44.6   5.2     0     1     0  -0.053609645 0.3342415
13  13    3    75.8  82.0     0     0     1 179.310135591 0.2394424
14  14    3    48.3   5.0     0     0     1  -0.758872162 0.1428571
15  15    3    41.4   3.9     0     0     1  -0.657757473 0.1489377
16  16    3    52.5   5.4     0     0     1  -0.843855199 0.1451101
17  17    3    41.0   4.1     0     0     1  -0.632668636 0.1496631
18  18    3    29.6   3.1     0     0     1  -0.414146872 0.1875182
19  19    3    49.5   5.4     0     0     1  -0.761586510 0.1430411
20  20    4    73.1   8.4     0     0     0  -0.122417349 0.2037518
21  21    4    81.3   9.5     0     0     0  -0.263888726 0.2236919
22  22    4    72.4   8.7     0     0     0  -0.081662233 0.2028453
23  23    4    88.4  10.6     0     0     0  -0.382413963 0.2548308
24  24    4    23.2   3.3     0     0     0   1.033584706 0.4526824
          cookD
1  2.315699e-06
2  6.612132e-07
3  1.142570e-02
4  1.247986e-02
5  6.688119e-02
6  3.454261e-04
```

```
7  3.838848e-04
8  4.156670e-04
9  7.281906e-03
10 1.816125e-02
11 1.235505e-02
12 3.045589e-04
13 1.195665e+00
14 1.963451e-02
15 1.560885e-02
16 2.454621e-02
17 1.454909e-02
18 8.278103e-03
19 1.980053e-02
20 8.088886e-04
21 4.219801e-03
22 3.581089e-04
23 1.047277e-02
24 1.760832e-01
>
```

Complete R code

```
#Outliers etc
data=read.table("C:/Users/Mihinda/Desktop/outliers.txt"
, header=1) #the data file
data
fit <- lm(sales ~ city1 + city2 + city3 + traffic ,
data=data)
summary(fit)
anova(fit)
data$tres <- rstudent(fit)
data
X <- model.matrix(fit)
X
data$hii=hat(X)
data
data$cookD <- cooks.distance(fit)
data
```