Practical_2

Soumarya Basak

16/04/2022

Problem 1

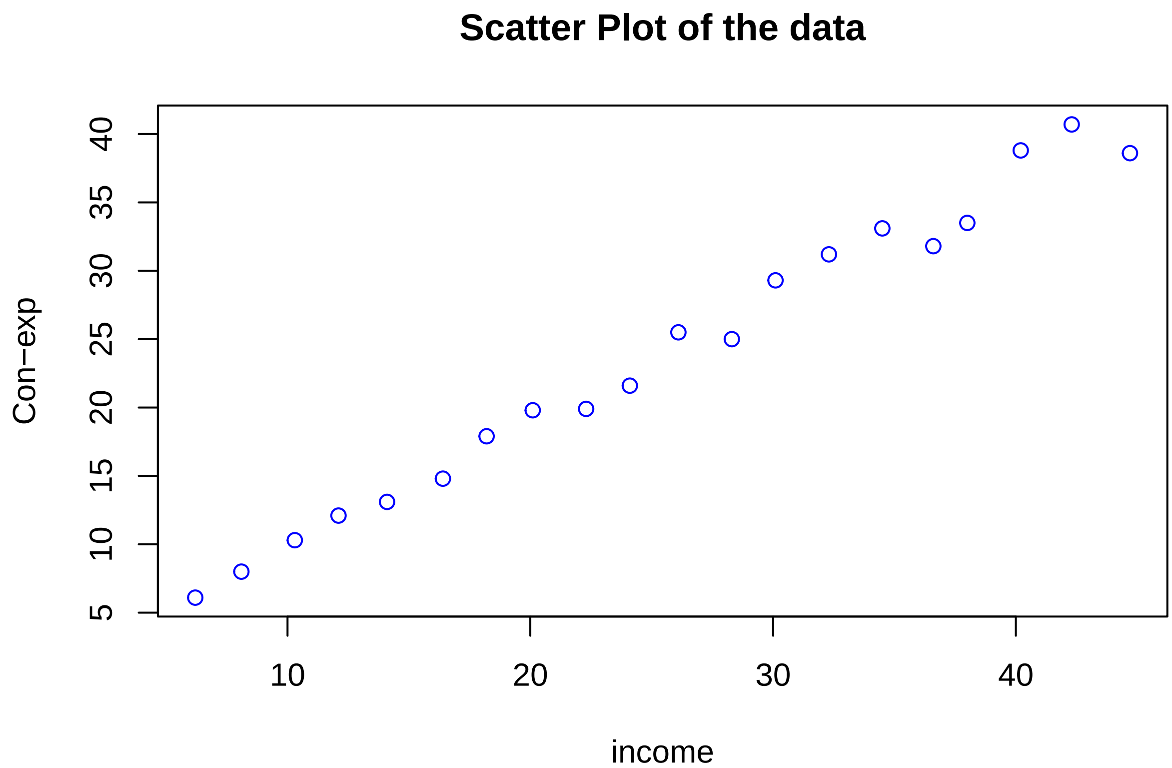
```
df<- read.csv("C:\\Users\\souma\\Dropbox\\Mstat_CU\\Sem 2\\Regression_analysis_1\\Data Sets\\con-exp_in

colnames(df)<-c('x_var','y_var')
```

```
lm_1<- lm(y_var~x_var,df)
summary(lm_1)
```

```
##
## Call:
## lm(formula = y_var ~ x_var, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.44687 -0.94108  0.02916  1.19199  1.81151
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.84705    0.70335   1.204    0.244
## x_var        0.89932    0.02531  35.534   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.314 on 18 degrees of freedom
## Multiple R-squared:  0.9859, Adjusted R-squared:  0.9852
## F-statistic:  1263 on 1 and 18 DF,  p-value: < 2.2e-16
```

Scatter Plot

```
plot(df$x_var,df$y_var,col="blue",
     main="Scatter Plot of the data", xlab="income",ylab="Con-exp")
```




```r
#abline(lm_1)
```

The model is not fitted well as the intercept term is insignificant. There may be several reason of this — the data may consist influential observations, the error variability may not be constant etc. In the following section we will try to find them.

Outlier

```r
influence.measures(lm_1)
```

```
## Influence measures of
##   lm(formula = y_var ~ x_var, data = df) :
##
##      dfb.1_ dfb.x_vr   dffit cov.r  cook.d    hat inf
## 1  -0.11552   0.0453 -0.1838 1.105 0.01727 0.0532
## 2  -0.03202   0.1450  0.2794 1.066 0.03889 0.0684
## 3   0.18229  -0.3787 -0.5417 0.925 0.13401 0.0978
## 4   0.10051  -0.0787  0.1050 1.253 0.00580 0.1142
## 5  -0.34877   0.5588  0.6760 1.018 0.21157 0.1579
## 6  -0.12527   0.1077 -0.1261 1.364 0.00839 0.1847   *
## 7   0.63252  -0.9594 -1.1173 0.804 0.50361 0.1904   *
## 8   0.07397   0.0154  0.2112 1.072 0.02250 0.0503
## 9   0.05756  -0.0467  0.0591 1.290 0.00185 0.1329
## 10 -0.27545   0.4715  0.5969 1.001 0.16594 0.1329
## 11 -0.04557   0.0382 -0.0462 1.332 0.00113 0.1591
## 12 -0.06951   0.1808  0.2902 1.096 0.04222 0.0817
```

```
## 13  0.17144  -0.3245 -0.4389 1.058 0.09344 0.1103
## 14 -0.10128   0.0754 -0.1088 1.224 0.00623 0.0961
## 15 -0.16052   0.1102 -0.1818 1.164 0.01711 0.0791
## 16 -0.08263   0.0161 -0.1635 1.114 0.01375 0.0505
## 17  0.00958   0.1050  0.2725 1.040 0.03674 0.0587
## 18 -0.04181  -0.0614 -0.2415 1.053 0.02911 0.0535
## 19  0.11892  -0.0745  0.1435 1.165 0.01073 0.0684
## 20  0.12826  -0.0693  0.1710 1.131 0.01507 0.0598
```

There are 2 influential obsn 6, 7 th obsn

SO we remove them from the model and refit the model below,

```r
df_updated<- df[-c(6,7),]

lm_1updated<- lm(y_var~x_var,df_updated)  # after removing influential observation
summary(lm_1updated)
```

```
##
## Call:
## lm(formula = y_var ~ x_var, data = df_updated)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.3578 -0.9786  0.4246  0.9694  1.3282
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.46505    0.75112   0.619    0.545
## x_var        0.92057    0.02752  33.452 3.08e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.216 on 16 degrees of freedom
## Multiple R-squared:  0.9859, Adjusted R-squared:  0.985
## F-statistic:  1119 on 1 and 16 DF,  p-value: 3.082e-16
```

But still the model is inappropriate as, the intercept is still insignificant.

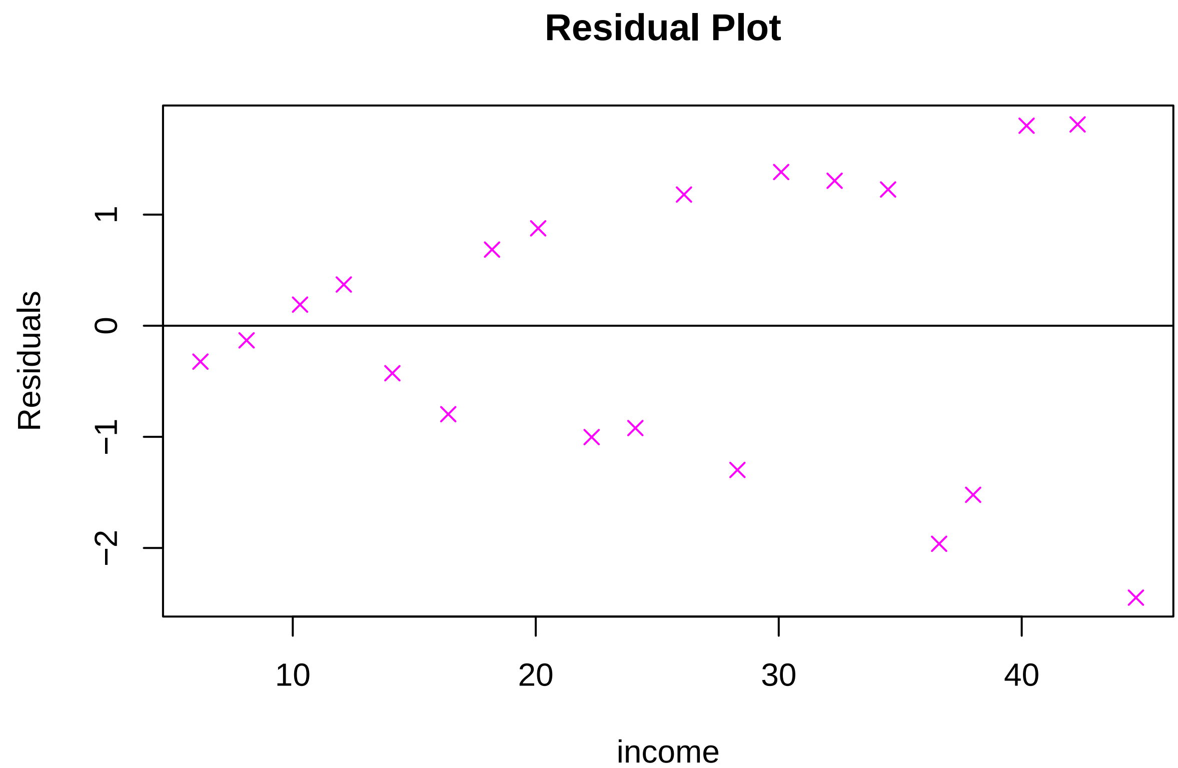
Lets check for the second reason.

As dropping the influential observations doesn't change the model so its better to work with the full model.

Residuals

Now we will find the residual form the updated fit.

```r
res1<- resid(lm_1)  # residuals

plot(df$x_var,res1,
     main="Residual Plot ",
     xlab="income",ylab="Residuals", col="magenta",pch=4) # residual plot
abline(h=0)
```



So from the residual plot we notice that there is a heteroscedasticity, but to confirm that statisticaliy we should test for this.
An famous test is **Goldfield Quant test**

Goldfield Quant Test

```
library(lmtest)
```

```
## Warning: package 'lmtest' was built under R version 4.1.3
```

```
## Loading required package: zoo
```

```
## Warning: package 'zoo' was built under R version 4.1.3
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
gqtest(lm_1,,fraction =6, order.by = ~df$x_var,data = df ) #alternative is more than
```

```
##
##  Goldfeld-Quandt test
##
## data:  lm_1
## GQ = 13.322, df1 = 5, df2 = 5, p-value = 0.006482
## alternative hypothesis: variance increases from segment 1 to 2
```

Since the p value is less than 0.05, we will reject the null hypothesis, so there is heteroscedasticity in the model.

So we need to workout with heteroscedasticity.

Obtaining model for Residuals

We will fit an regression of $residual^2$ on x_var, where $residual_i^2$ can be looked upon as a estimate of σ_i^2

```
df_res<-cbind(df,res1^2) # add the (ei)^2 values
colnames(df_res)[3]<-"res"
```

```
# the model for the sigma
res_lm1<-lm(res ~ x_var, data= df_res)
summary(res_lm1)
```

```
##
## Call:
## lm(formula = res ~ x_var, data = df_res)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.1212 -0.2607 -0.1955  0.1335  2.1811
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.37033    0.38860  -3.526  0.00241 **
## x_var        0.11580    0.01398   8.282 1.49e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7259 on 18 degrees of freedom
## Multiple R-squared:  0.7921, Adjusted R-squared:  0.7806
## F-statistic: 68.59 on 1 and 18 DF,  p-value: 1.492e-07
```

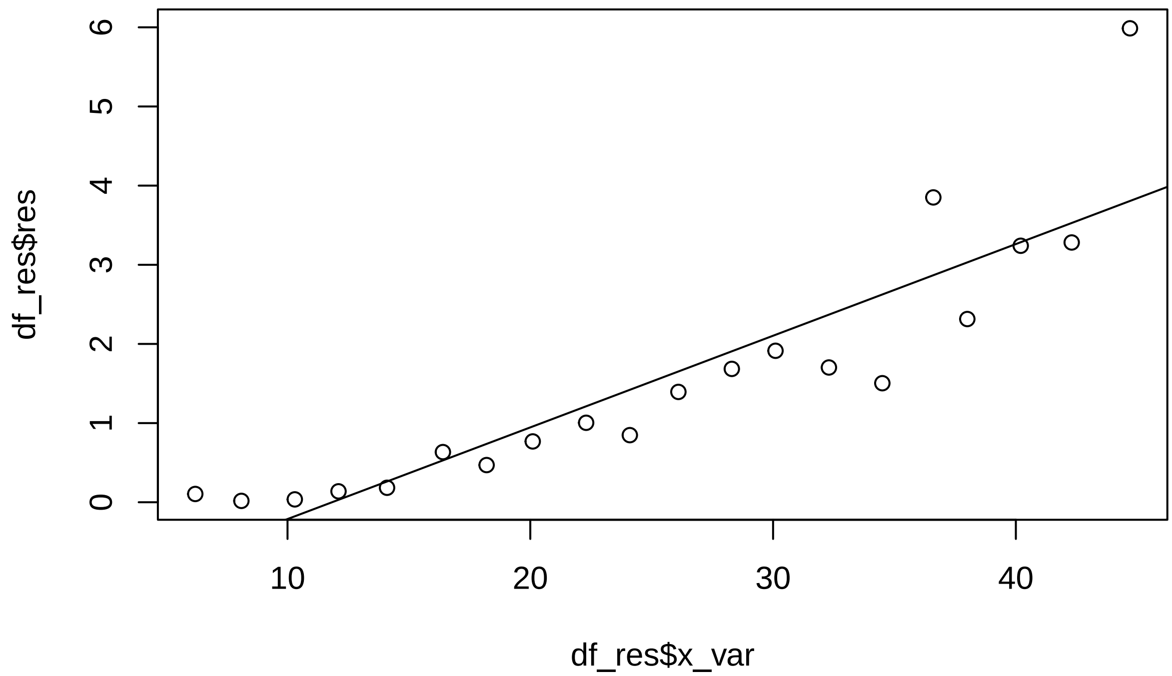
Now from the above model, we will find an estimate of σ_i^2, by fitting this model, and given by

```
sigma_estimated<- predict(res_lm1)
sigma_estimated
```

```
##           1           2           3           4           5           6
##  1.21207079  2.37009707  2.86804837  0.03088399  3.52812335 -0.65235151
##           7           8           9          10          11          12
##  3.80604966  1.65212078 -0.17756074  3.28493783 -0.43232652  2.62486285
##          13          14          15          16          17          18
##  3.03017205  0.26248925  0.52883529  1.42051552  2.11533129  1.90688656
##          19          20
##  0.73728002  0.95730501
```

The Plot of x and $fitted(\sigma_i^2)$ (Unnecessary)

```
plot(df_res$x_var,df_res$res)
abline(res_lm1)
```



Estimated Omega hat

```
omega<- diag(sigma_estimated)
```

Model's Design Matrix

```
d_matrix <- model.matrix(lm_1)
d_matrix
```

```
##    (Intercept) x_var
## 1            1  22.3
## 2            1  32.3
## 3            1  36.6
## 4            1  12.1
## 5            1  42.3
## 6            1   6.2
## 7            1  44.7
## 8            1  26.1
## 9            1  10.3
## 10           1  40.2
## 11           1   8.1
## 12           1  34.5
## 13           1  38.0
## 14           1  14.1
## 15           1  16.4
## 16           1  24.1
## 17           1  30.1
```

```
## 18           1  28.3
## 19           1  18.2
## 20           1  20.1
## attr(,"assign")
## [1] 0 1
```

```
a1<- solve( t(d_matrix) %*% solve(omega) %*% d_matrix )
a2<- t(d_matrix)  %*% solve(omega) %*% df$y_var

beta_egls<- a1 %*% a2
beta_egls
```

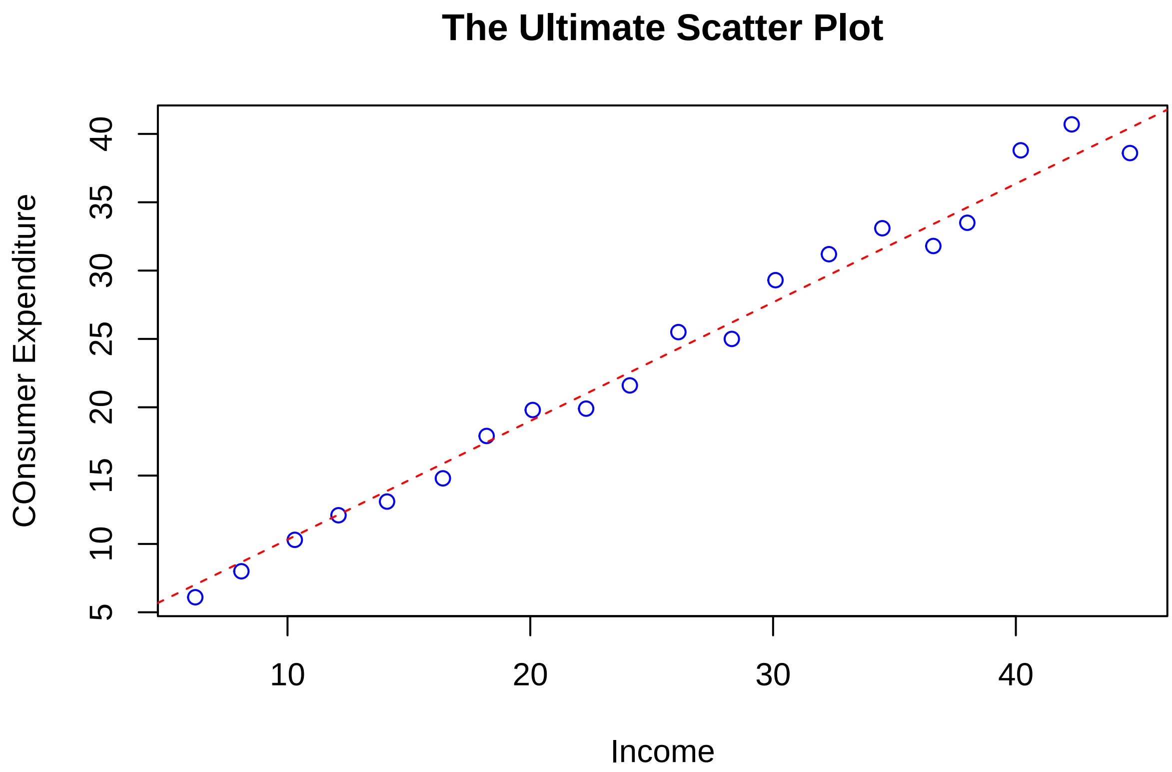
```
##                 [,1]
## (Intercept) 1.638355
## x_var       0.867986
```

The Model

$$y = 1.638355 + 0.867986 \times x$$

The Scatter Plot

```
plot(df$x_var,df$y_var,col='blue',
     xlab = "Income",ylab = "COnsumer Expenditure",
     main = "The Ultimate Scatter Plot")
abline(1.638355, 0.867986,col='red',lty=2)
```



Problem 2

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(readr)
library(readxl)
```

```
df<- read.csv("C:\\Users\\souma\\Dropbox\\Mstat_CU\\Sem 2\\Regression_analysis_1\\Data Sets\\dis_sp_data

colnames(df)<- c("y","x")   # y=dis , x= sp
head(df)
```

```
##   y x
## 1 4 4
## 2 2 5
## 3 4 5
```

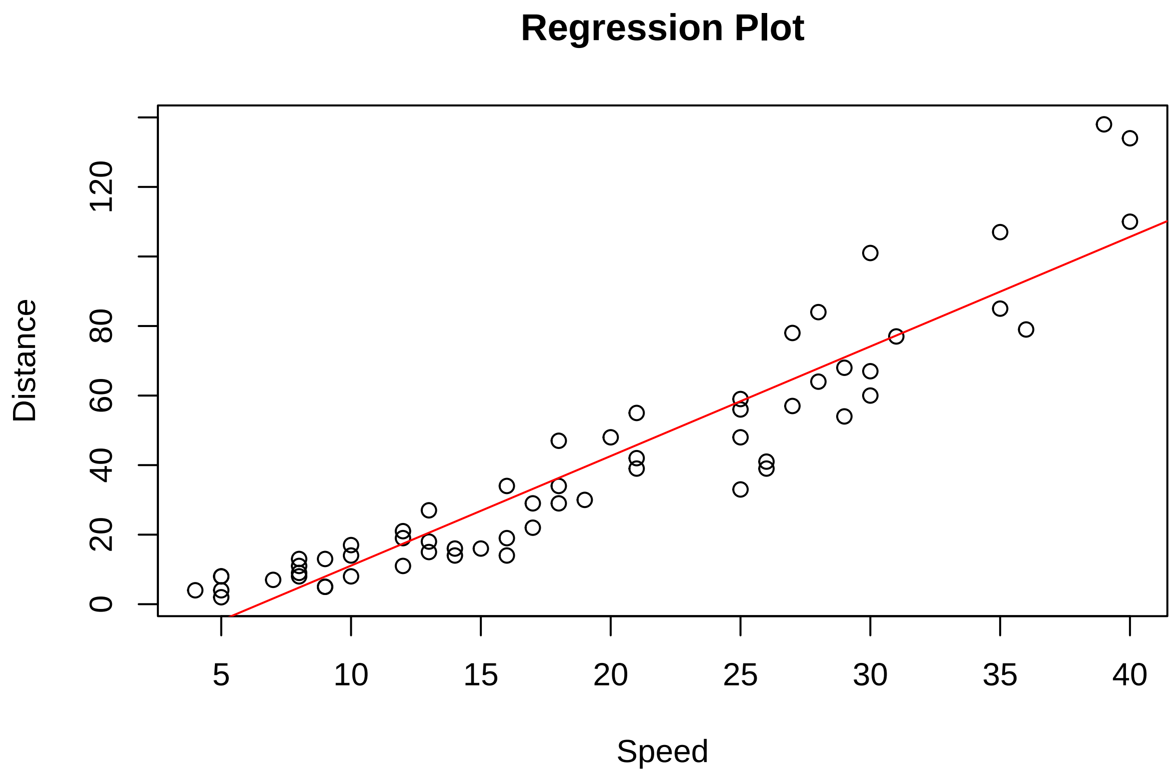
```
## 4 8 5
## 5 8 5
## 6 7 7
```

Lets fit the regression

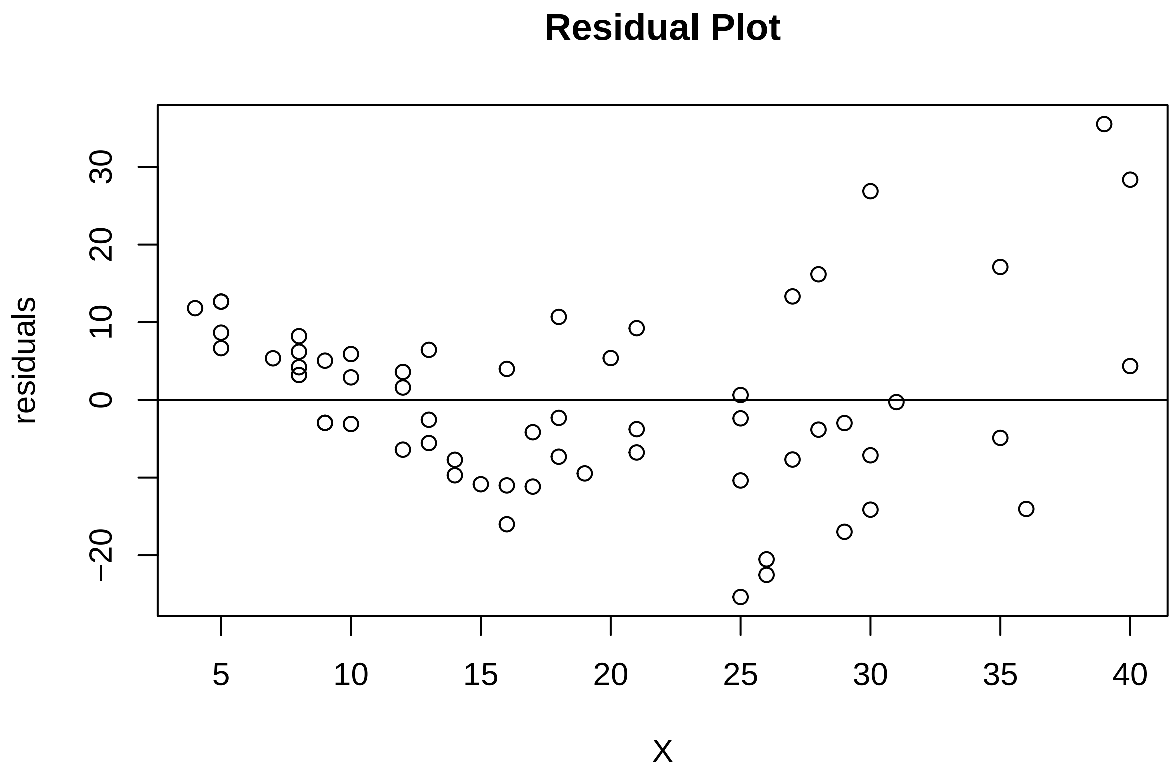
```
reg1<- lm(y~x,data=df)
summary(reg1)
```

```
##
## Call:
## lm(formula = y ~ x, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -25.371  -7.401  -2.340   6.266  35.508
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -20.4174     3.3446  -6.105 9.16e-08 ***
## x             3.1515     0.1559  20.213  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.95 on 58 degrees of freedom
## Multiple R-squared:  0.8757, Adjusted R-squared:  0.8735
## F-statistic: 408.6 on 1 and 58 DF,  p-value: < 2.2e-16
```

```
# Scatter Plot
plot(df$x,df$y,
     main="Regression Plot",
     xlab = "Speed", ylab="Distance")

abline(reg1,col="red")
```




```r
# The residuals
res1<-resid(reg1)

# Residual Plot
plot(df$x,res1, main="Residual Plot",
     xlab = "X",ylab = "residuals")
abline(h=0)
```



From the residual plot we can suspect that there is a heterosecdasticity within the data, so should test for that. So we should go for the **Glejser test**

Now lets check for Influencial Observation

```
influence.measures(reg1)
```

```
## Influence measures of
##   lm(formula = y ~ x, data = df) :
##
##        dfb.1_     dfb.x    dffit cov.r   cook.d    hat inf
## 1   0.244429 -0.205256  0.24572 1.057 3.02e-02 0.0551
## 2   0.129506 -0.106811  0.13069 1.078 8.64e-03 0.0502
## 3   0.168729 -0.139161  0.17027 1.069 1.46e-02 0.0502
## 4   0.248027 -0.204562  0.25030 1.046 3.12e-02 0.0502
## 5   0.248027 -0.204562  0.25030 1.046 3.12e-02 0.0502
## 6   0.092346 -0.072912  0.09440 1.072 4.52e-03 0.0413
## 7   0.051765 -0.039792  0.05345 1.073 1.45e-03 0.0374
## 8   0.067948 -0.052231  0.07016 1.071 2.50e-03 0.0374
## 9   0.100395 -0.077173  0.10366 1.065 5.44e-03 0.0374
## 10  0.132995 -0.102233  0.13732 1.058 9.51e-03 0.0374
## 11  0.076362 -0.056896  0.07991 1.065 3.24e-03 0.0338
## 12 -0.043535  0.031265 -0.04637 1.065 1.09e-03 0.0306
## 13  0.040782 -0.029287  0.04344 1.066 9.59e-04 0.0306
## 14  0.083076 -0.059660  0.08848 1.059 3.97e-03 0.0306
## 15 -0.076980  0.050113 -0.08649 1.051 3.79e-03 0.0251
## 16  0.019185 -0.012489  0.02156 1.061 2.36e-04 0.0251
```

```
## 17  0.043209 -0.028128  0.04855 1.059 1.20e-03 0.0251
## 18 -0.061114  0.037181 -0.07142 1.052 2.59e-03 0.0229
## 19 -0.028052  0.017066 -0.03278 1.058 5.46e-04 0.0229
## 20  0.071013 -0.043204  0.08299 1.049 3.49e-03 0.0229
## 21 -0.098289  0.048687 -0.12898 1.025 8.34e-03 0.0194
## 22 -0.130108  0.054398 -0.18559 0.989 1.70e-02 0.0182
## 23 -0.088710  0.037089 -0.12654 1.023 8.02e-03 0.0182
## 24  0.031971 -0.013367  0.04560 1.050 1.06e-03 0.0182
## 25 -0.078879  0.025187 -0.12512 1.022 7.84e-03 0.0174
## 26 -0.029202  0.009325 -0.04632 1.049 1.09e-03 0.0174
## 27 -0.044255  0.008344 -0.08033 1.039 3.26e-03 0.0168
## 28 -0.013944  0.002629 -0.02531 1.052 3.26e-04 0.0168
## 29  0.064961 -0.012248  0.11792 1.024 6.97e-03 0.0168
## 30 -0.048105  0.000349 -0.10362 1.030 5.40e-03 0.0167
## 31 -0.011699 -0.008163 -0.04187 1.050 8.90e-04 0.0173
## 32  0.028827  0.020112  0.10317 1.032 5.36e-03 0.0173
## 33  0.021391 -0.174762 -0.33842 0.898 5.36e-02 0.0227
## 34  0.008445 -0.068998 -0.13361 1.032 8.96e-03 0.0227
## 35  0.001918 -0.015673 -0.03035 1.058 4.68e-04 0.0227
## 36 -0.000509  0.004158  0.00805 1.059 3.30e-05 0.0227
## 37  0.041777 -0.179925 -0.31253 0.933 4.66e-02 0.0249
## 38  0.037853 -0.163023 -0.28317 0.954 3.87e-02 0.0249
## 39  0.021473 -0.068294 -0.10890 1.049 5.99e-03 0.0275
## 40 -0.037569  0.119488  0.19053 1.018 1.81e-02 0.0275
## 41  0.082879 -0.193023 -0.27199 0.996 3.63e-02 0.0336
## 42  0.014277 -0.033250 -0.04685 1.069 1.12e-03 0.0336
## 43  0.083110 -0.176406 -0.23759 1.022 2.80e-02 0.0371
## 44  0.041537 -0.088165 -0.11875 1.062 7.13e-03 0.0371
## 45 -0.163666  0.347392  0.46788 0.889 1.01e-01 0.0371   *
## 46  0.001913 -0.003782 -0.00491 1.080 1.22e-05 0.0410
## 47  0.054118 -0.089980 -0.10586 1.095 5.68e-03 0.0601
## 48 -0.192927  0.320774  0.37737 1.020 6.97e-02 0.0601
## 49  0.172818 -0.279509 -0.32357 1.052 5.19e-02 0.0657
## 50 -0.604751  0.918090  1.02463 0.786 4.45e-01 0.0845   *
## 51 -0.044473  0.033136 -0.04654 1.069 1.10e-03 0.0338
## 52 -0.097447  0.054319 -0.11980 1.033 7.22e-03 0.0210
## 53 -0.044473  0.033136 -0.04654 1.069 1.10e-03 0.0338
## 54 -0.077196  0.043031 -0.09490 1.042 4.55e-03 0.0210
## 55  0.022019  0.005743  0.05907 1.046 1.77e-03 0.0168
## 56 -0.021063 -0.014696 -0.07538 1.042 2.87e-03 0.0173
## 57  0.014499 -0.038326 -0.05707 1.064 1.65e-03 0.0304
## 58 -0.062272  0.164607  0.24513 0.999 2.96e-02 0.0304
## 59 -0.072962  0.108962  0.12049 1.134 7.37e-03 0.0915   *
## 60 -0.501905  0.749552  0.82882 0.909 3.12e-01 0.0915   *
```

So, we can see there are 4 influencial observation, viz 45,50,59,60 th observation.
Lets remove these

Regression fit after removing influencial

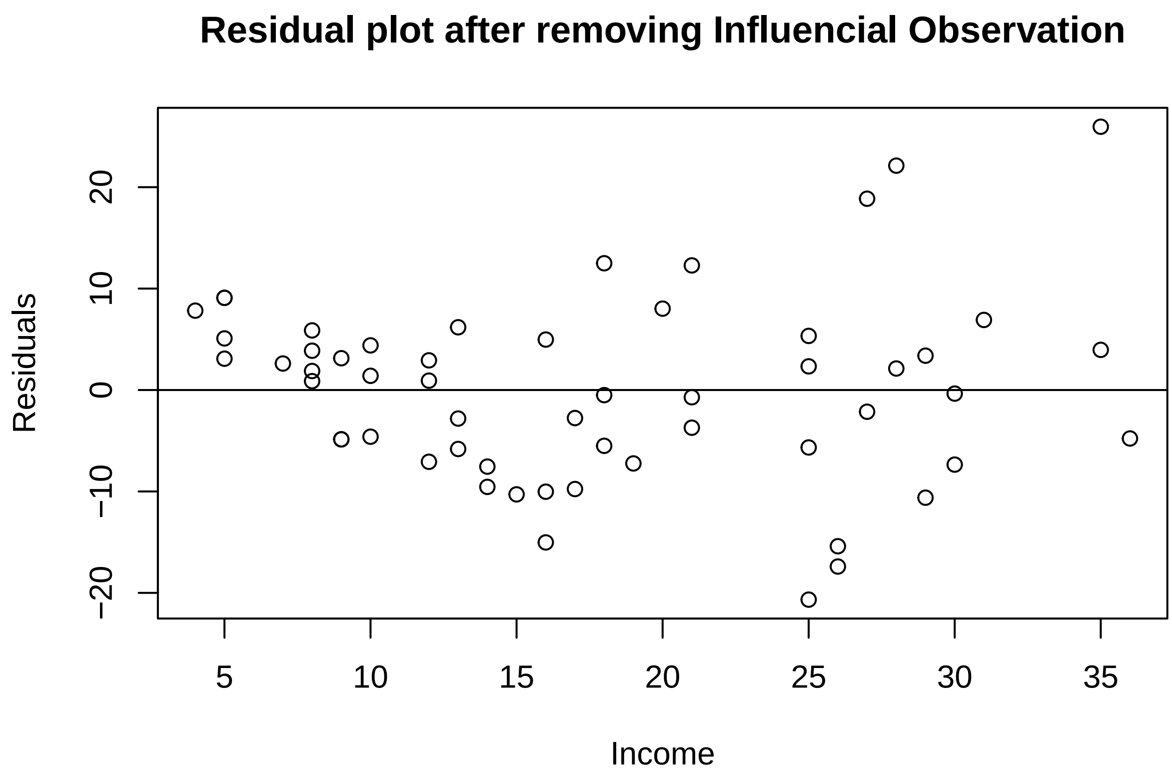
```
df1<- df[-c(45,50,59,60),]
```

```
reg2<- lm(y~x,data = df1)
summary(reg2)
```

```
##
## Call:
## lm(formula = y ~ x, data = df1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -20.6646  -5.7017   0.8999   5.0026  25.9591
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -14.7762     2.7524  -5.369 1.71e-06 ***
## x             2.7376     0.1389  19.710  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.194 on 54 degrees of freedom
## Multiple R-squared:  0.878,  Adjusted R-squared:  0.8757
## F-statistic: 388.5 on 1 and 54 DF,  p-value: < 2.2e-16
```

Lets see the residual plot

```
# Model after removing influential observation
res2<- resid(reg2)
plot(df1$x,res2,main="Residual plot after removing Influencial Observation",
     xlab = "Income",ylab = "Residuals")
abline(h=0)
```



Still we can see that, there is a heteroscedasticity in the residuals. And the adjusted R^2 for the new model doesn't change much
So we will stick in the actual data set and previous model(`reg1` model), as removing data point causes information loss from our hand.

Test For Heteroscedasticity

So we till test for heteroscedasticity by **Glejser Test**

```
library(skedastic)
```

```
## Warning: package 'skedastic' was built under R version 4.1.3
```

```
# Glejser Test
glejser(reg1)
```

```
## # A tibble: 1 x 4
##   statistic  p.value parameter alternative
##       <dbl>    <dbl>     <dbl> <chr>
## 1      12.5 0.000409         1 greater
```

It results that the p value is less than 0.05, so, the presence of Heteroscedasticity statistically significant under 5% level of significant.

So now we will remove the heteroscedasticity from the model from the data set , assuming the error variance as a linear function of speed with an intercept term.

Remedial Measure

```
res1_sq<- res1^2 # ei Square

rem_lm<- lm(res1_sq~df$x)
summary(rem_lm)
```

```
##
## Call:
## lm(formula = res1_sq ~ df$x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -352.27  -97.17  -22.72   24.55  900.68
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -73.672     56.369  -1.307    0.196
## df$x          11.123      2.628   4.233 8.33e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 201.4 on 58 degrees of freedom
## Multiple R-squared:  0.236,  Adjusted R-squared:  0.2228
## F-statistic: 17.92 on 1 and 58 DF,  p-value: 8.327e-05
```

But, the intercept term is insignificant, so we can drop this from the model. But here we are asked to work with an intercept term.

So the model is

$$e_i^2 = -73.672 + 11.123 \times x$$

So, the estiamted sigma_i sq hats are

```
sigma_estimated<- predict(rem_lm)

omega<- diag(sigma_estimated)
```

Lets find the design Matrix

```
d_matrix<- model.matrix(reg1)
d_matrix
```

```
##    (Intercept)  x
## 1            1  4
## 2            1  5
## 3            1  5
## 4            1  5
## 5            1  5
## 6            1  7
## 7            1  8
## 8            1  8
## 9            1  8
## 10           1  8
## 11           1  9
## 12           1 10
## 13           1 10
## 14           1 10
## 15           1 12
## 16           1 12
## 17           1 12
## 18           1 13
## 19           1 13
## 20           1 13
## 21           1 15
## 22           1 16
## 23           1 16
## 24           1 16
## 25           1 17
## 26           1 17
## 27           1 18
## 28           1 18
## 29           1 18
## 30           1 19
## 31           1 21
## 32           1 21
## 33           1 25
## 34           1 25
## 35           1 25
## 36           1 25
## 37           1 26
## 38           1 26
```

```
## 39           1 27
## 40           1 27
## 41           1 29
## 42           1 29
## 43           1 30
## 44           1 30
## 45           1 30
## 46           1 31
## 47           1 35
## 48           1 35
## 49           1 36
## 50           1 39
## 51           1  9
## 52           1 14
## 53           1  9
## 54           1 14
## 55           1 20
## 56           1 21
## 57           1 28
## 58           1 28
## 59           1 40
## 60           1 40
## attr(,"assign")
## [1] 0 1
```

```
a<- t(d_matrix) %*% solve(omega) %*% d_matrix
b<- t(d_matrix) %*% solve(omega) %*% df$y

beta_egls2 <- solve(a) %*% b
beta_egls2
```

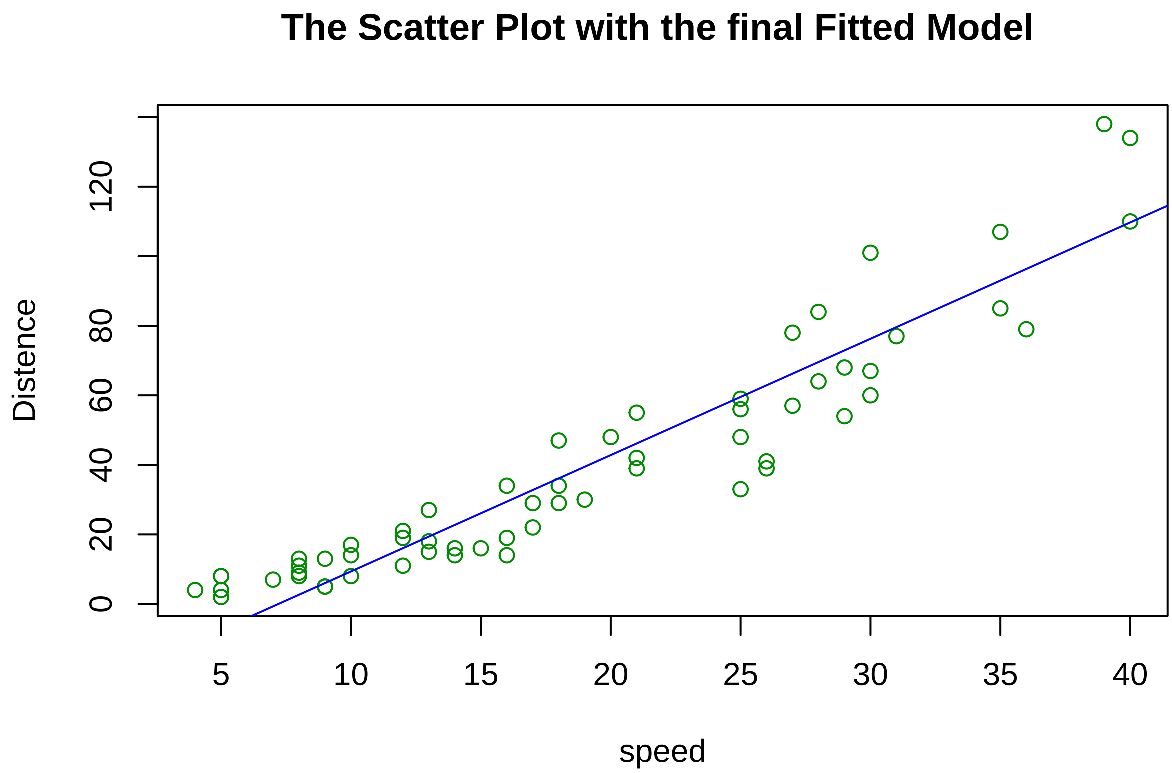
```
##                   [,1]
## (Intercept) -24.118582
## x             3.345985
```

The Final Model

So, after removing the heteroscedasticity, and find the final fitted model as,

$$y = -24.118582 + 3.345985 \times x$$

```
plot(df$x,df$y,main = "The Scatter Plot with the final Fitted Model ",
     xlab = "speed",ylab = "Distence",col="green4")
abline(-24.118582,3.345985,col='blue')
```



Problem 3

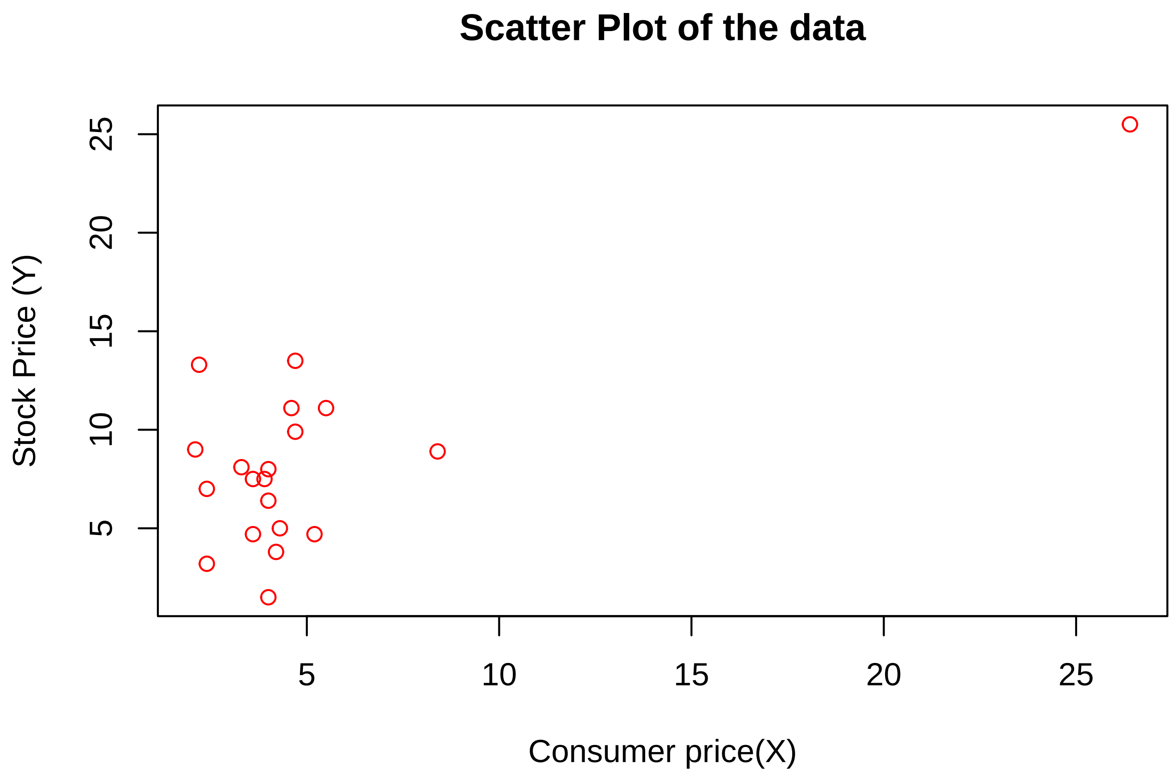
```
library(tidyverse)
library(readr)
library(readxl)
```

```
df<- read.csv("C:\\Users\\souma\\Dropbox\\Mstat_CU\\Sem 2\\Regression_analysis_1\\Data Sets\\stock_pric
head(df)
```

```
##   ï..Country    y    x
## 1  Australia  5.0  4.3
## 2    Austria 11.1  4.6
## 3    Belgium  3.2  2.4
## 4     Canada  7.0  2.4
## 5      Chile 25.5 26.4
## 6    Denmark  3.8  4.2
```

Plot the data and fit a suitable Regression

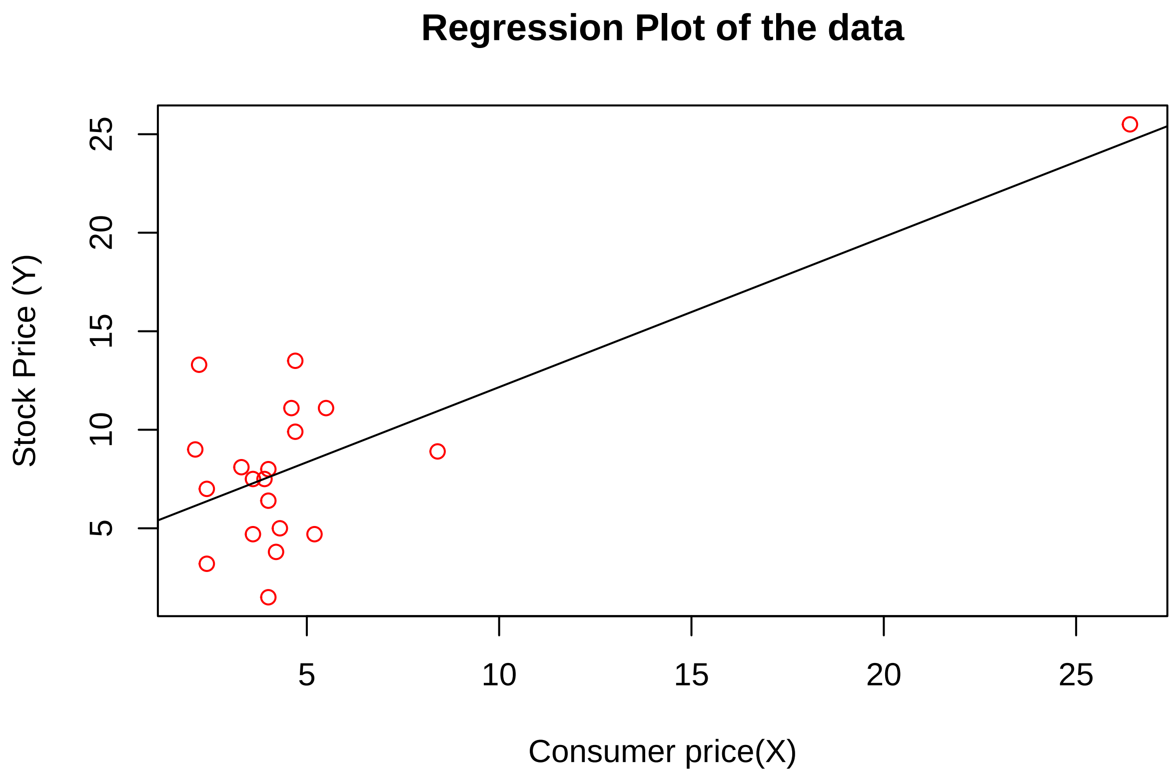
```
plot(df$x,df$y, main="Scatter Plot of the data",
     xlab="Consumer price(X)",ylab = "Stock Price (Y)", col='red')
```




```
# regission fit
reg3<- lm(y~x,data=df)
summary(reg3)
```

```
##
## Call:
## lm(formula = y ~ x, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.0893 -2.6428  0.3132  1.9246  7.0829
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.5400     1.0799   4.204 0.000533 ***
## x             0.7623     0.1493   5.108 7.36e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.375 on 18 degrees of freedom
## Multiple R-squared:  0.5917, Adjusted R-squared:  0.569
## F-statistic: 26.09 on 1 and 18 DF,  p-value: 7.361e-05
```

```
plot(df$x,df$y, main="Regression Plot of the data",
     xlab="Consumer price(X)",ylab = "Stock Price (Y)", col='red')
abline(reg3)
```



Lets check for influencial

```
influence.measures(reg3)
```

```
## Influence measures of
##   lm(formula = y ~ x, data = df) :
##
##        dfb.1_     dfb.x     dffit  cov.r   cook.d    hat inf
## 1  -0.16066  0.033792 -0.198197  1.087 1.99e-02 0.0515
## 2   0.16556 -0.024128  0.213567  1.071 2.30e-02 0.0506
## 3  -0.24469  0.123032 -0.255743  1.077 3.28e-02 0.0651
## 4   0.04743 -0.023846  0.049568  1.194 1.30e-03 0.0651
## 5  -1.83474  3.343802  3.437398 14.652 5.95e+00 0.9308   *
## 6  -0.23349  0.053797 -0.284163  1.001 3.93e-02 0.0519
## 7   0.10631  0.010464  0.163152  1.113 1.37e-02 0.0502
## 8   0.09306 -0.011410  0.122009  1.143 7.75e-03 0.0504
## 9   0.63731 -0.334860  0.660395  0.654 1.70e-01 0.0673   *
## 10 -0.39799  0.106895 -0.472310  0.775 9.56e-02 0.0527
## 11 -0.07018  0.018848 -0.083281  1.166 3.65e-03 0.0527
## 12 -0.03474 -0.091250 -0.169706  1.154 1.49e-02 0.0703
## 13  0.06892 -0.026507  0.076246  1.175 3.06e-03 0.0569
## 14  0.30265 -0.037109  0.396805  0.856 7.10e-02 0.0504
## 15 -0.18631 -0.001325 -0.267936  1.011 3.52e-02 0.0500
## 16  0.01354 -0.004575  0.015387  1.186 1.25e-04 0.0549
## 17 -0.16511  0.055796 -0.187634  1.106 1.80e-02 0.0549
## 18  0.02416 -0.006488  0.028669  1.181 4.35e-04 0.0527
## 19 -0.00078  0.000224 -0.000915  1.184 4.43e-07 0.0532
```

```
## 20  0.22899 -0.122812  0.236377  1.103 2.83e-02 0.0685
```

The 5th and 9th obsn are influential

```
df1<- df[-c(5,9),]
reg3_up<- lm(y~x,data=df1)
summary(reg3_up)
```

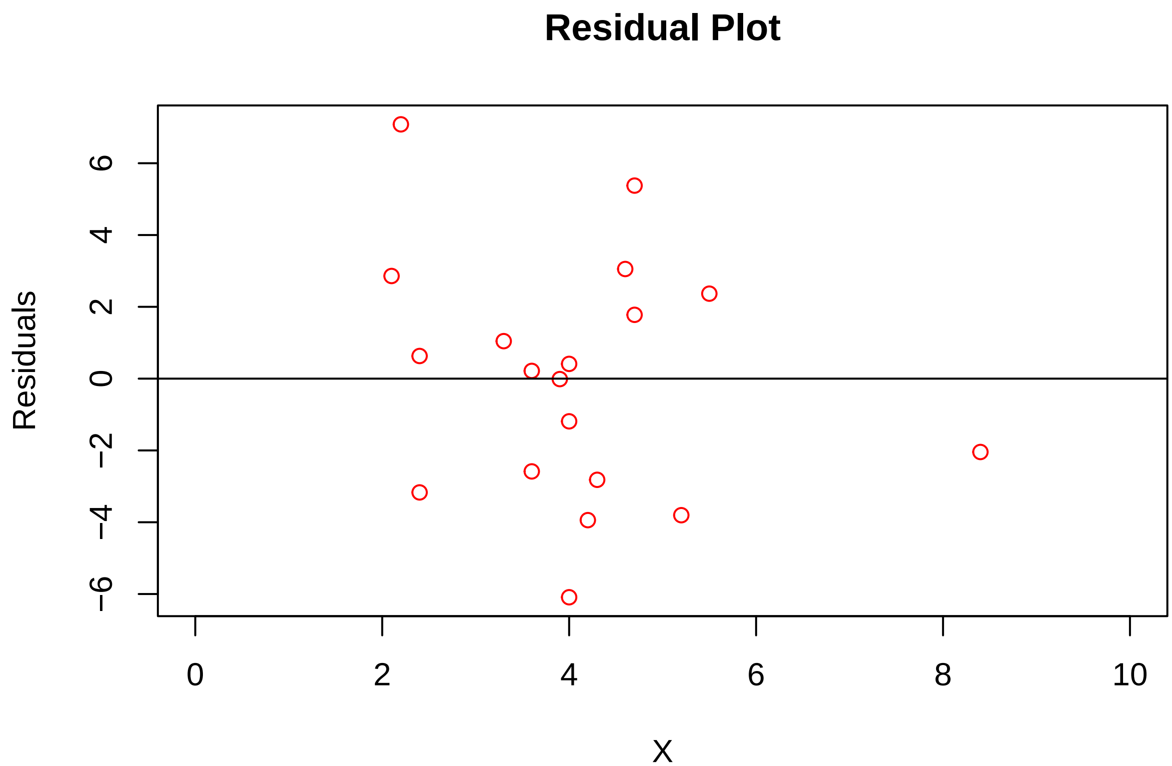
```
##
## Call:
## lm(formula = y ~ x, data = df1)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -5.672 -2.325  0.484  2.060  5.892
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.6791     2.3140   2.022   0.0602 .
## x             0.6232     0.5284   1.179   0.2554
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.063 on 16 degrees of freedom
## Multiple R-squared:  0.07999,    Adjusted R-squared:  0.02249
## F-statistic: 1.391 on 1 and 16 DF,  p-value: 0.2554
```

So its worst than the previous.

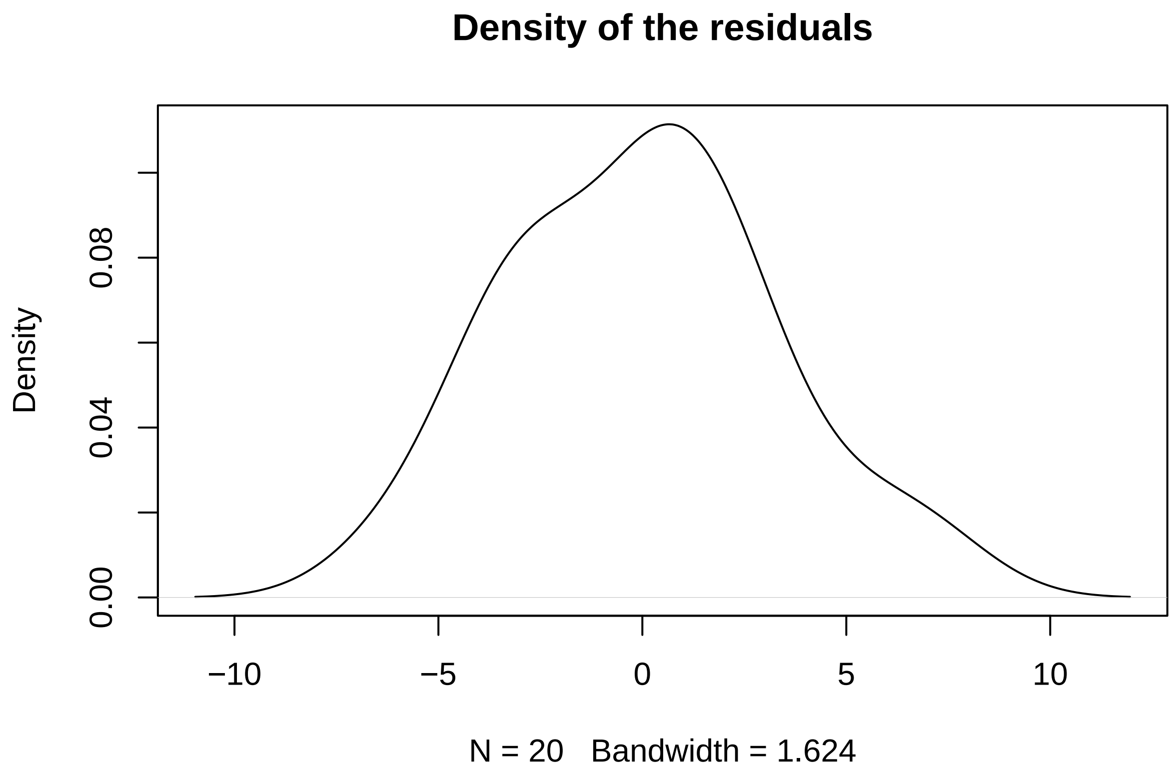
So, its better to stick with the original model

Residual Plot

```
res3<- resid(reg3)
# residual plot
plot(df$x,res3, xlim = c(0, 10),
     main="Residual Plot", xlab="X", ylab="Residuals", col='red')
abline(h=0)
```




```
plot(density(res3), main = "Density of the residuals")
```



Goldfield Quant test

```
library(lmtest)

gqtest(reg3,fraction = 6,order.by = ~x, data = df)
```

```
##
##  Goldfeld-Quandt test
##
## data:  reg3
## GQ = 1.0387, df1 = 5, df2 = 5, p-value = 0.4839
## alternative hypothesis: variance increases from segment 1 to 2
```

Since the p value is more than 0.05, So we fail to reject the null hypothesis, so the data is homoscedastic.

So OLS works good here.

Parameter of the Model

```
reg3$coefficients
```

```
## (Intercept)           x
##   4.5400120   0.7623165
```

Final Model

```
plot(df$x,df$y, main = "The Final Fit",
     xlab = "Consumer Price (X)" , ylab="Stok Price (Y)")
abline(reg3,col='red')
```