Regression Analysis lab 2

1 Simple linear regression

1.1 Import data

```
rocket<-read.csv(file="D:/chilo/Regression 2/rocket.csv", header=T)
rocket

   observation strength   age
1            1     2159 15.50
2            2     1678 23.75
3            3     2316  8.00
4            4     2061 17.00
5            5     2208  5.50
6            6     1708 19.00
7            7     1785 24.00
8            8     2575  2.50
9            9     2358  7.50
10          10     2257 11.00
11          11     2165 13.00
12          12     2400  3.75
13          13     1780 25.00
14          14     2337  9.75
15          15     1765 22.00
16          16     2054 18.00
17          17     2414  6.00
18          18     2200 12.50
19          19     2654  2.00
20          20     1754 21.50
```

```
names(rocket)  # list all variable names in rocket data frame

[1] "observation" "strength"    "age"

dim(rocket)  # there are 3 variables and 20 observations in rocket data frame

[1] 20  3
```

1.2 Enter data and create a data fram

```r
y<-c(2158.70, 1678.15, 2316.00, 2061.00, 2207.50, 1708.30,
1784.70, 2575.00, 2357.90, 2256.70, 2165.20, 2399.55, 1779.80,
2336.75, 1765.30, 2053.50, 2414.40, 2200.50, 2654.20, 1753.70)
x<-c(15.50, 23.75, 8.00, 17.00, 5.50, 19.00, 24.00, 2.50, 7.50,
11.00, 13.00, 3.75, 25.00, 9.75, 22.00, 18.00, 6.00, 12.50,
2.00, 21.50)
rocket1<-data.frame(y,x)
rocket1
```

```
      y     x
1  2159 15.50
2  1678 23.75
3  2316  8.00
4  2061 17.00
5  2208  5.50
6  1708 19.00
7  1785 24.00
8  2575  2.50
9  2358  7.50
10 2257 11.00
11 2165 13.00
12 2400  3.75
13 1780 25.00
14 2337  9.75
15 1765 22.00
16 2054 18.00
17 2414  6.00
18 2200 12.50
19 2654  2.00
20 1754 21.50
```

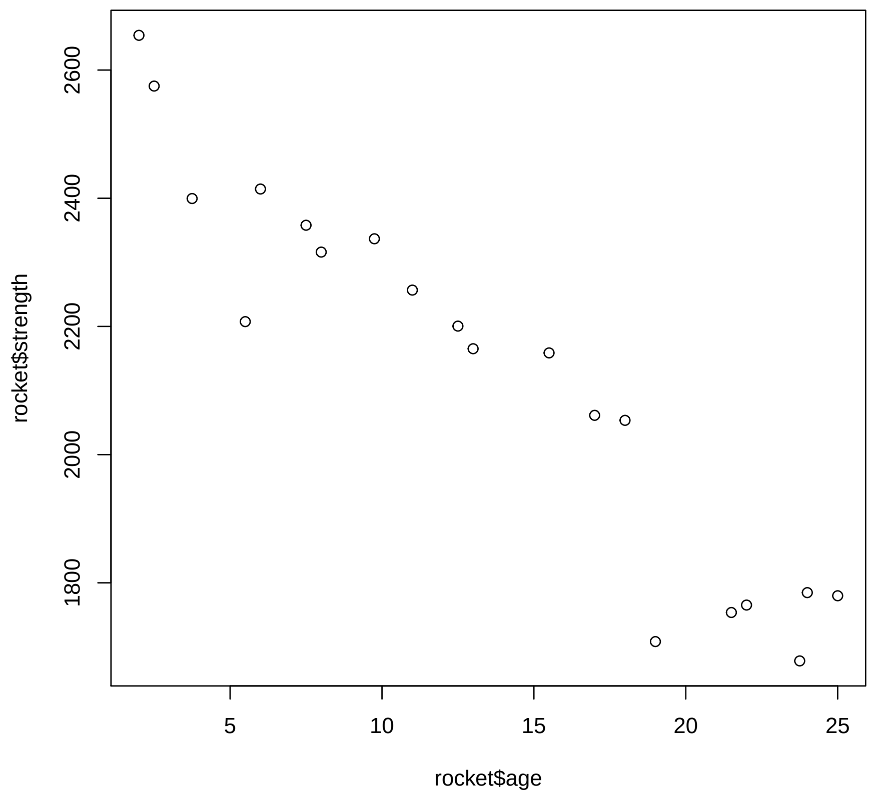
```r
names(rocket1)<-c("strength", "age")
head(rocket1)
```

```
  strength   age
1     2159 15.50
2     1678 23.75
3     2316  8.00
4     2061 17.00
5     2208  5.50
6     1708 19.00
```

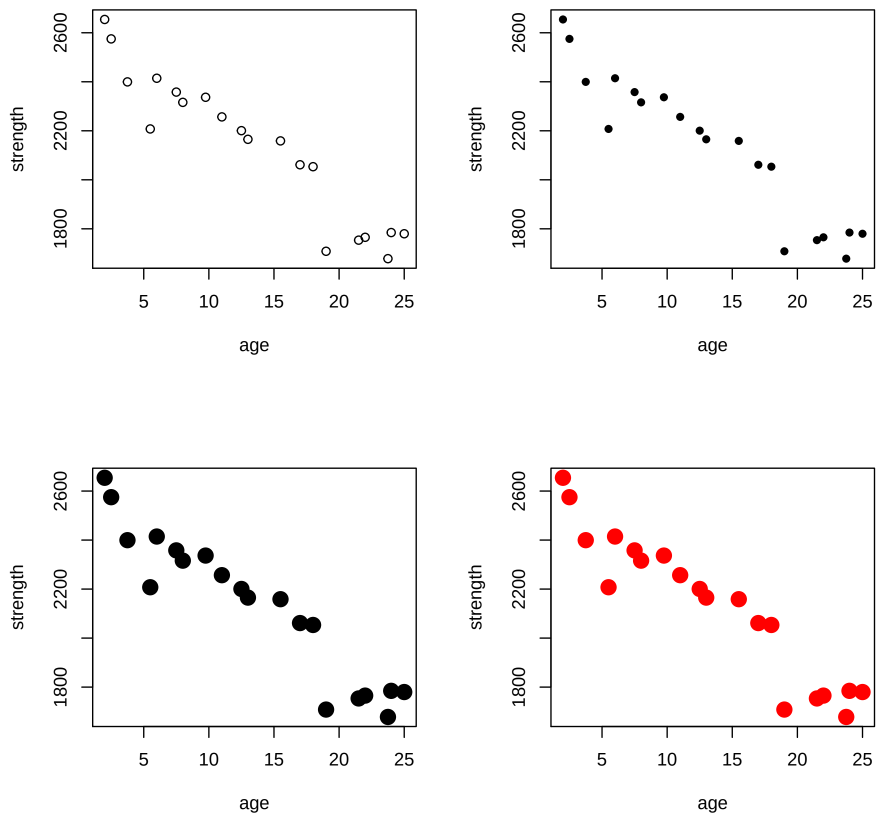
```r
rocket1<-edit(rocket1) # can enter and/or edit data in an edit windows
```

1.3 Scatter plot

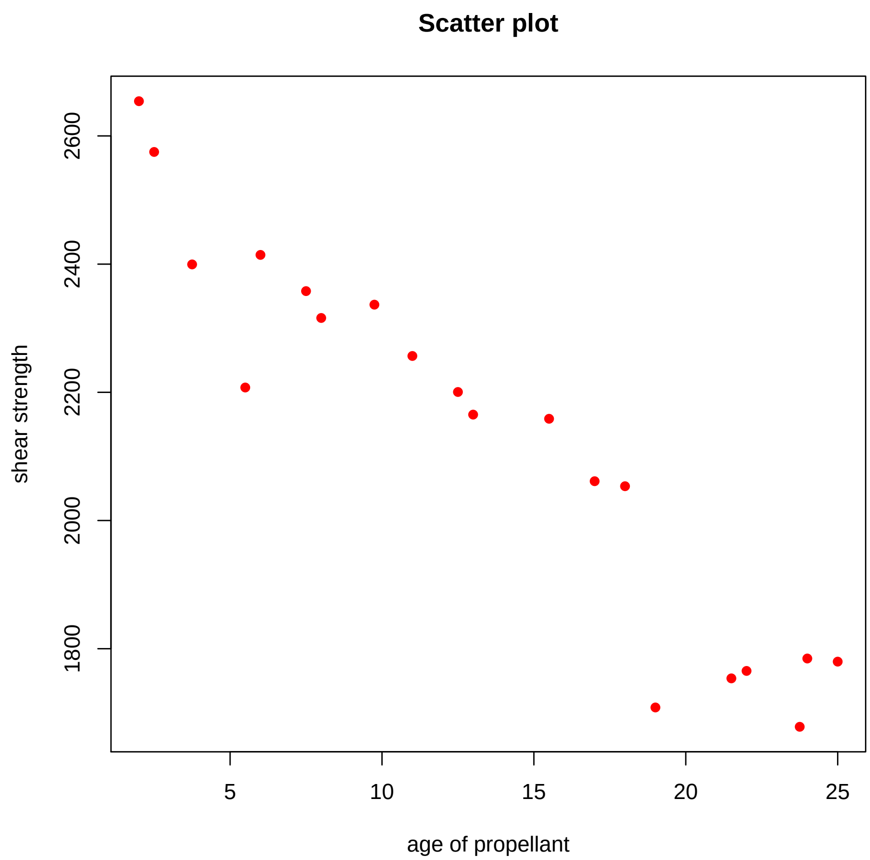
```
plot(rocket$age, rocket$strength) # make a scatter plot
```



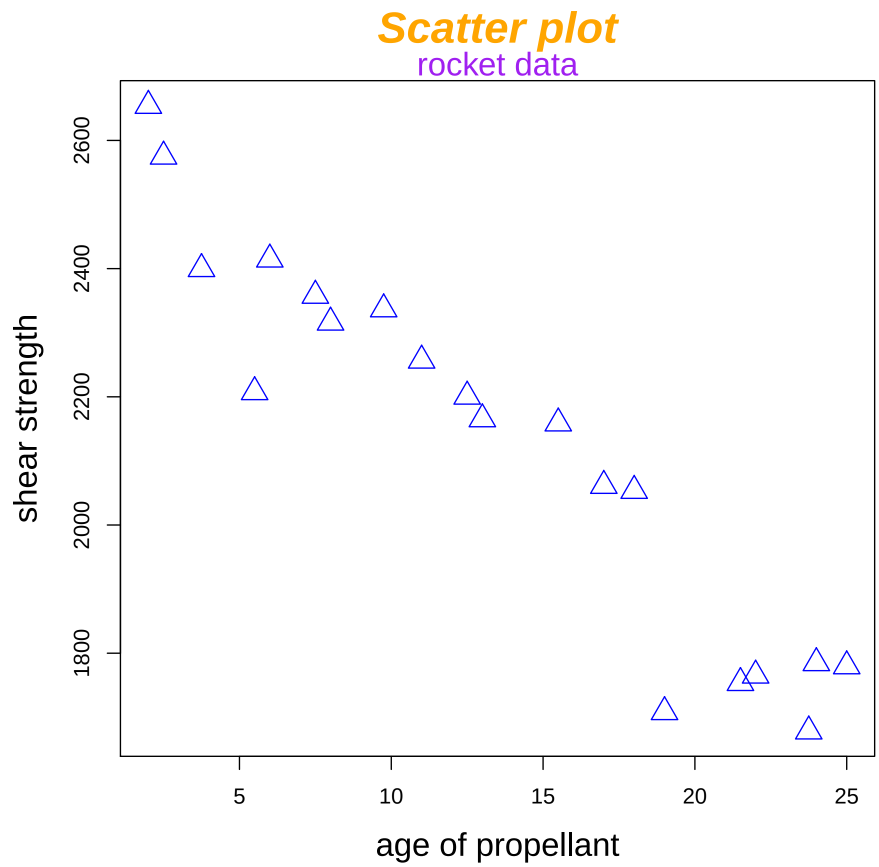

```
attach(rocket)
par(mfrow=c(2,2))
plot(age, strength)  # points are circles
plot(age, strength, pch=16)  # points are solid circles
plot(age, strength, pch=16, cex=2)  # cex control size
plot(age, strength, pch=16, cex=2, col=2) # col control colour
```




```
par(mfrow=c(1,1))
plot(age, strength, pch=16, cex=1, col=2, xlab="age of propellant",
     ylab="shear strength", main="Scatter plot")  # add title
```

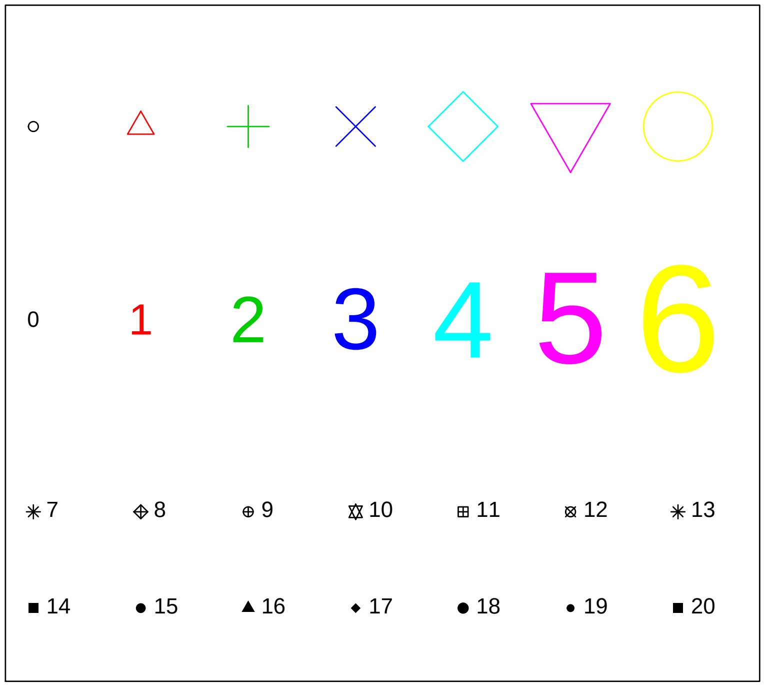



```
plot(age, strength, pch=2, cex=2, col=4, xlab="age of propellant",
    ylab="shear strength", cex.lab=1.5)
title(main="Scatter plot", cex.main=2, font.main=4,
      col.main="orange", cex.sub=1, font.sub=3, col.sub="purple")
# adding text in the margin
mtext(side=3, line=0, text="rocket data", cex=1.5, col="purple")
```



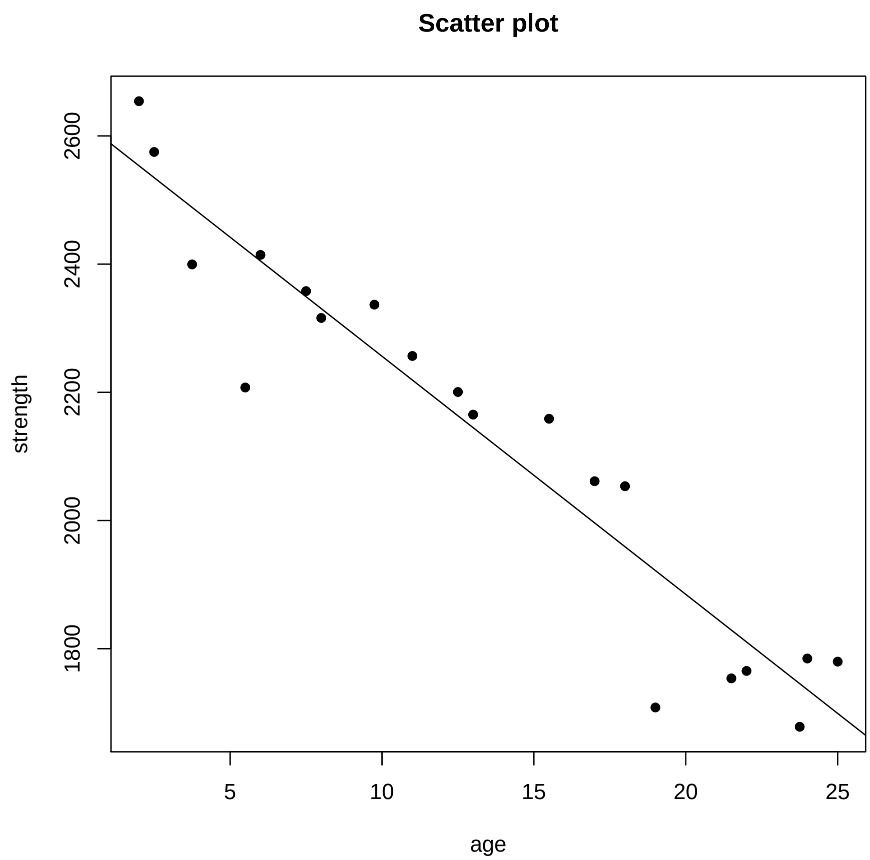
1.4 Size, colour, and choice of plotting symbol

```
plot(1,1, xlim=c(1,7.5), ylim=c(1.75, 5), type="n", axes=F,
     xlab="", ylab="")  # do not plot pints
box()
points(1:7, rep(4.5,7), cex=1:7, col=1:7, pch=1:6)
text(1:7,rep(3.5,7), labels=paste(0:6), cex=1:7, col=1:7)
points(1:7, rep(2.5,7), pch=(1:6)+7)
text((1:7),rep(2.5,7), paste((0:6)+7), pos=4)
points(1:7, rep(2,7), pch=(1:6)+14)
text((1:7),rep(2,7), paste((0:6)+14), pos=4)
```

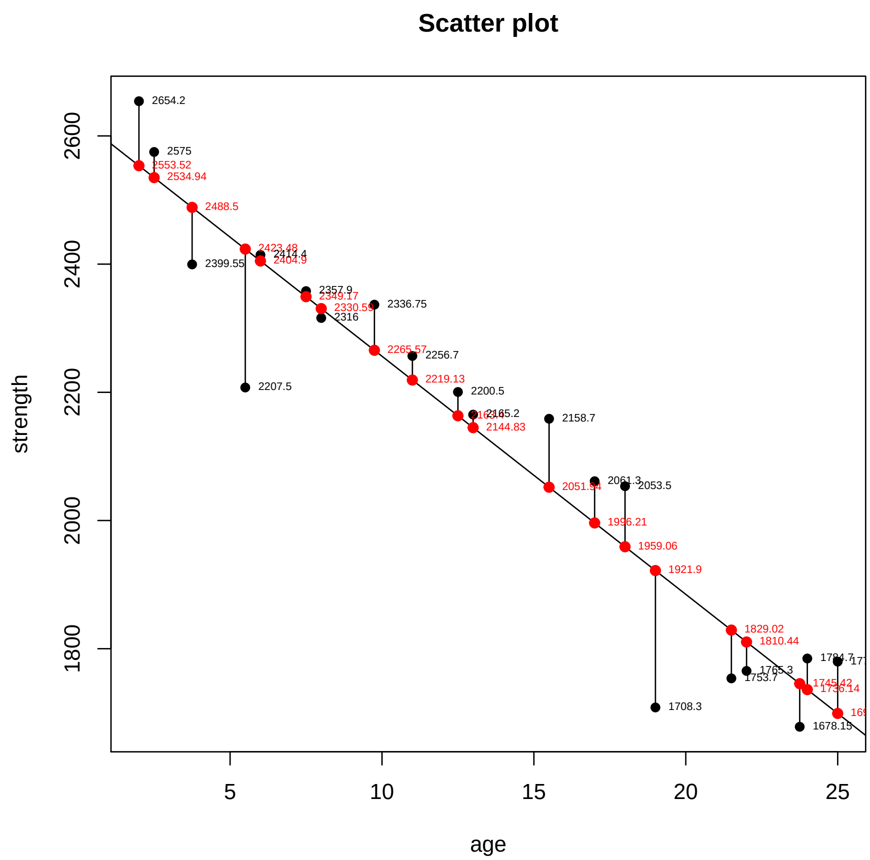


1.5 Plot a regression line on scatter plot

```
rocket.lm<-lm(strength~age, data=rocket)
# fit a simple linear regression model
plot(strength ~ age, data=rocket, pch=16, main="Scatter plot")
abline(reg=rocket.lm)  # plot the regression line on scatter plot
```




```
plot(strength ~ age, data=rocket, pch=16, main="Scatter plot")
abline(reg=rocket.lm)
fitted<-fitted.values(rocket.lm)
for(i in 1:length(age)){
lines(c(age[i],age[i]),c(strength[i], fitted[i]))
}
points(age, fitted, col="red", pch=19)
text(age, fitted, labels=round(fitted, 2), pos=4, col="red", cex=0.5)
text(age, strength, labels=round(strength, 2), 3, pos=4, cex=0.5)
```



1.6 Fit a simple linear regression model

```
rocket.lm<-lm(strength~age, data=rocket)
rocket.lm


Call:
lm(formula = strength ~ age, data = rocket)

Coefficients:
(Intercept)          age
     2627.8        -37.2

summary(rocket.lm) # estimated coefficients, tests
```

```
Call:
lm(formula = strength ~ age, data = rocket)

Residuals:
   Min     1Q Median     3Q    Max
-216.0  -50.7   28.7   66.6  106.8

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2627.82      44.18    59.5  < 2e-16 ***
age           -37.15       2.89   -12.9  1.6e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 96.1 on 18 degrees of freedom
Multiple R-squared:  0.902,Adjusted R-squared:  0.896
F-statistic:  165 on 1 and 18 DF,  p-value: 1.64e-10
```

```
names(rocket.lm)
```

```
 [1] "coefficients"  "residuals"     "effects"       "rank"
 [5] "fitted.values" "assign"        "qr"            "df.residual"
 [9] "xlevels"       "call"          "terms"         "model"
```

```
coef(rocket.lm)
```

```
(Intercept)         age
    2627.82      -37.15
```

```
fitted.values(rocket.lm)
```

```
   1    2    3    4    5    6    7    8    9   10   11   12   13   14   15
2052 1745 2331 1996 2423 1922 1736 2535 2349 2219 2145 2488 1699 2266 1810
  16   17   18   19   20
1959 2405 2163 2554 1829
```

```
resid(rocket.lm)
```

```
       1        2        3        4        5        6        7        8
 106.758  -67.275  -14.594   65.089 -215.978 -213.604   48.564   40.062
       9       10       11       12       13       14       15       16
   8.730   37.567   20.374  -88.946   80.817   71.175  -45.143   94.442
      17       18       19       20
   9.499   37.098  100.685  -75.320
```

```
fitted<-fitted.values(rocket.lm)
residual<-resid(rocket.lm)
cbind(strength, fitted, residual)
```

```
   strength fitted residual
1      2159   2052  106.758
2      1678   1745  -67.275
3      2316   2331  -14.594
4      2061   1996   65.089
5      2208   2423 -215.978
6      1708   1922 -213.604
7      1785   1736   48.564
8      2575   2535   40.062
9      2358   2349    8.730
10     2257   2219   37.567
11     2165   2145   20.374
12     2400   2488  -88.946
13     1780   1699   80.817
14     2337   2266   71.175
15     1765   1810  -45.143
16     2054   1959   94.442
17     2414   2405    9.499
18     2200   2163   37.098
19     2654   2554  100.685
20     1754   1829  -75.320
```

```
round(cbind(strength, fitted, residual),2)
```

```
   strength fitted residual
1      2159   2052   106.76
2      1678   1745   -67.27
3      2316   2331   -14.59
4      2061   1996    65.09
5      2208   2423  -215.98
6      1708   1922  -213.60
7      1785   1736    48.56
8      2575   2535    40.06
9      2358   2349     8.73
10     2257   2219    37.57
11     2165   2145    20.37
12     2400   2488   -88.95
13     1780   1699    80.82
14     2337   2266    71.18
15     1765   1810   -45.14
16     2054   1959    94.44
17     2414   2405     9.50
18     2200   2163    37.10
19     2654   2554   100.68
20     1754   1829   -75.32
```

1.7 Compute estimated coefficients, fitted values and residuals by formula

```
attach(rocket)

The following objects are masked from rocket (position 3):

    age, observation, strength

n<-length(age)
n

[1] 20

sumx<-sum(age)
sumx

[1] 267.2

sumx2<-sum(age^2)
sumx2

[1] 4678

sumy<-sum(strength)
sumy

[1] 42627

sumy2<-sum(strength^2)
sumy2

[1] 92547433

sumxy<-sum(age*strength)
sumxy

[1] 528493

xbar<-mean(age)
xbar

[1] 13.36

ybar<-mean(strength)
ybar

[1] 2131
```

```
Sxx<-sumx2-(sumx)^2/n
Sxx

[1] 1107

Sxy<-sumxy-(sumx)*(sumy)/n
Sxy

[1] -41113

beta1<-Sxy/Sxx
beta1

[1] -37.15

beta0<-ybar-beta1*xbar
beta0

[1] 2628

yhat<-beta0+beta1*age
yhat

 [1] 2052 1745 2331 1996 2423 1922 1736 2535 2349 2219 2145 2488 1699 2266
[15] 1810 1959 2405 2163 2554 1829

e<-strength-yhat
e

 [1]  106.758  -67.275  -14.594   65.089 -215.978 -213.604   48.564
 [8]   40.062    8.730   37.567   20.374  -88.946   80.817   71.175
[15]  -45.143   94.442    9.499   37.098  100.685  -75.320

sigmahat2<-sum(e^2)/(n-2)
sigmahat2

[1] 9236
```

1.8 Compute ANOVA by formula

```
SST<-sumy2-(sumy)^2/n
SST

[1] 1693738

SSR<-beta1*Sxy
SSR
```

```
[1] 1527483

SSE<-SST-SSR
SSE

[1] 166255

MSR<-SSR
MSR

[1] 1527483

MSE<-SSE/(n-2)
MSE

[1] 9236

Fratio<-MSR/MSE
Fratio

[1] 165.4

pvalue<-1-pf(Fratio,1,n-2)
pvalue

[1] 1.643e-10
```

1.9 Compute ANOVA table

```
rocket.lm<-lm(strength~age, data=rocket)
anova(rocket.lm)

Analysis of Variance Table

Response: strength
          Df  Sum Sq Mean Sq F value  Pr(>F)
age        1 1527483 1527483     165 1.6e-10 ***
Residuals 18  166255    9236
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

1.10 Compute t tests by formula

```
# Hypothesis testing for beta1
#
SE.beta1<-sqrt(MSE/Sxx)
SE.beta1

[1] 2.889

t1<-beta1/SE.beta1
t1

[1] -12.86

pvalue1<-2*(1-pt(abs(t1),n-2))
pvalue1

[1] 1.643e-10
```

```
# Hypothesis testing for beta0
#
SE.beta0<-sqrt(MSE*(1/n+xbar^2/Sxx))
SE.beta0

[1] 44.18

t0<-beta0/SE.beta0
t0

[1] 59.47

pvalue0<-2*(1-pt(abs(t0),n-2))
pvalue0

[1] 0
```

1.11 Compute confidence intervals by formula

```
qt(.975, n-2)

[1] 2.101

# 95% C.I. for beta1
#
c(beta1-qt(.975, n-2)*SE.beta1, beta1+qt(.975, n-2)*SE.beta1)

[1] -43.22 -31.08
```

```
# 95% C.I. for beta0
#
c(beta0-qt(.975, n-2)*SE.beta0, beta0+qt(.975, n-2)*SE.beta0)

[1] 2535 2721
```

```
# 95% C.I. for sigma^2
#
qchisq(.975, n-2)

[1] 31.53

qchisq(.025, n-2)

[1] 8.231

c((n-2)*MSE/qchisq(.975, n-2), (n-2)*MSE/qchisq(.025, n-2))

[1]  5274 20199
```

1.12 Compute confidence interval for the mean value of y when x=x0

```
x0<-xbar
muhat<-beta0+beta1*x0
muhat

[1] 2131

qt(.975, n-2)

[1] 2.101

bm<-qt(.975, n-2)*sqrt(MSE*(1/n+(x0-xbar)^2/Sxx))
bm

[1] 45.15

c(muhat-bm, muhat+bm)

[1] 2086 2177
```

1.13 Compute confidence interval for individual value of y when x=x0

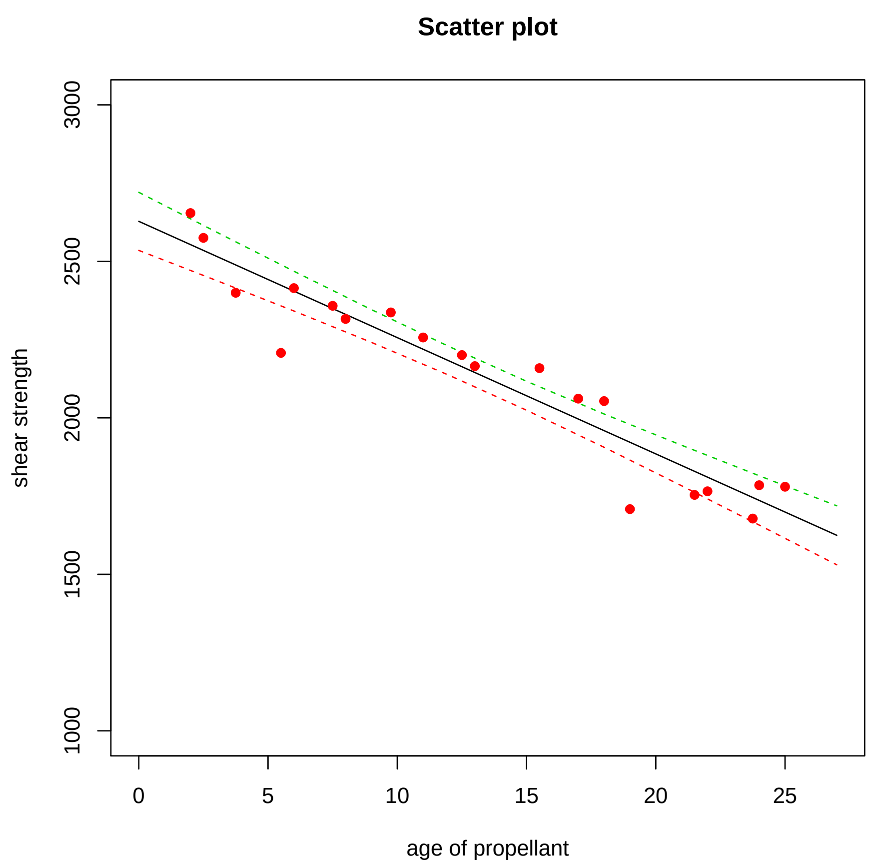
```
x0<-10
y0<-beta0+beta1*x0
y0

[1] 2256

qt(.975, n-2)

[1] 2.101

bm<-qt(.975, n-2)*sqrt(MSE*(1+1/n+(x0-xbar)^2/Sxx))
bm

[1] 207.9

c(y0-bm, y0+bm)

[1] 2048 2464
```

1.14 Compute and plot confidence limits and prediction intervals

```
x1<-c(0,3,6,9,12,15,18,21,24,27)
muhat1<-beta0+beta1*x1
muhat1

 [1] 2628 2516 2405 2293 2182 2071 1959 1848 1736 1625

bm1<-qt(.975, n-2)*sqrt(MSE*(1/n+(x1-xbar)^2/Sxx))
bm1

 [1] 92.83 77.42 63.53 52.34 45.90 46.23 53.20 64.71 78.79 94.29

lcl<-muhat1-bm1
lcl

 [1] 2535 2439 2341 2241 2136 2024 1906 1783 1657 1530

ucl<-muhat1+bm1
ucl

 [1] 2721 2594 2468 2346 2228 2117 2012 1912 1815 1719

y1<-beta0+beta1*x1
bm2<-qt(.975, n-2)*sqrt(MSE*(1+1/n+(x1-xbar)^2/Sxx))
lpi<-y1+bm2
lpi
```

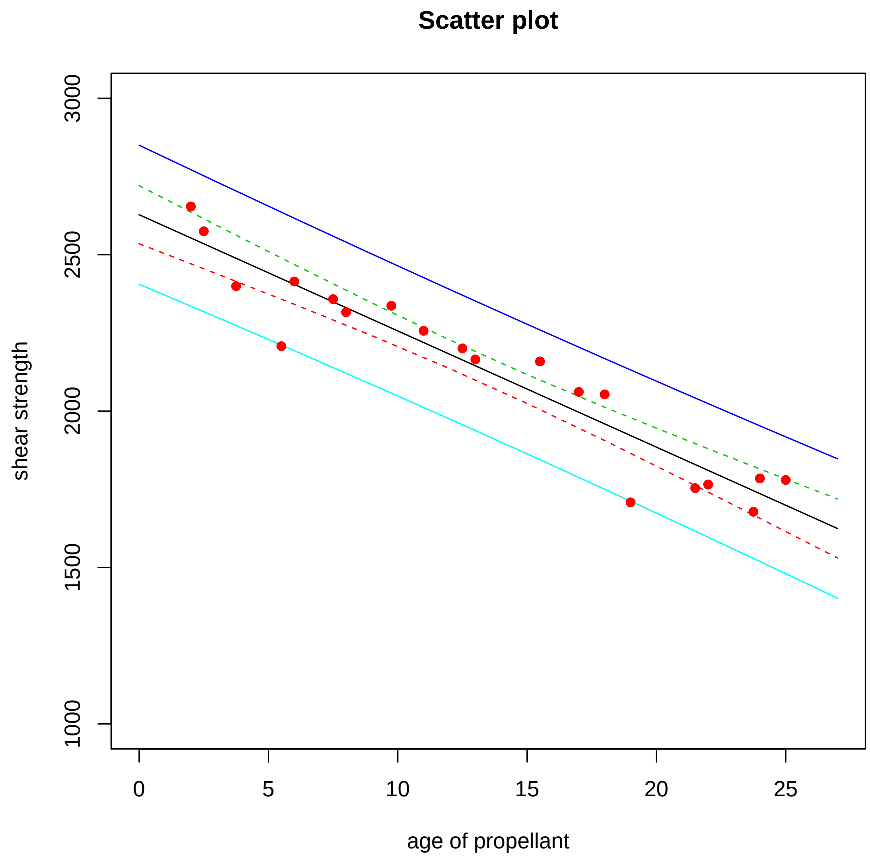
```
 [1] 2850 2733 2617 2502 2389 2278 2168 2060 1953 1848

upi<-y1-bm2
upi

 [1] 2406 2300 2193 2085 1975 1863 1750 1636 1519 1402
```

```
matplot(x1, cbind(muhat1, lcl, ucl), ylim=c(1000,3000),
        type="l", lty=c(1,2,2), xlab="age of propellant",
        ylab="shear strength", main="Scatter plot")
points(age, strength, ylim=c(1000,3000), pch=16, cex=1, col=2)
```




```
matplot(x1, cbind(muhat1, lcl, ucl, lpi, upi), ylim=c(1000,3000),
        type="l", lty=c(1,2,2,1,1), xlab="age of propellant",
        ylab="shear strength", main="Scatter plot")
```

```
points(age, strength, ylim=c(1000,3000), pch=16, cex=1, col=2)
```



1.15 Compute coefficient of determination R2 and adjusted R2

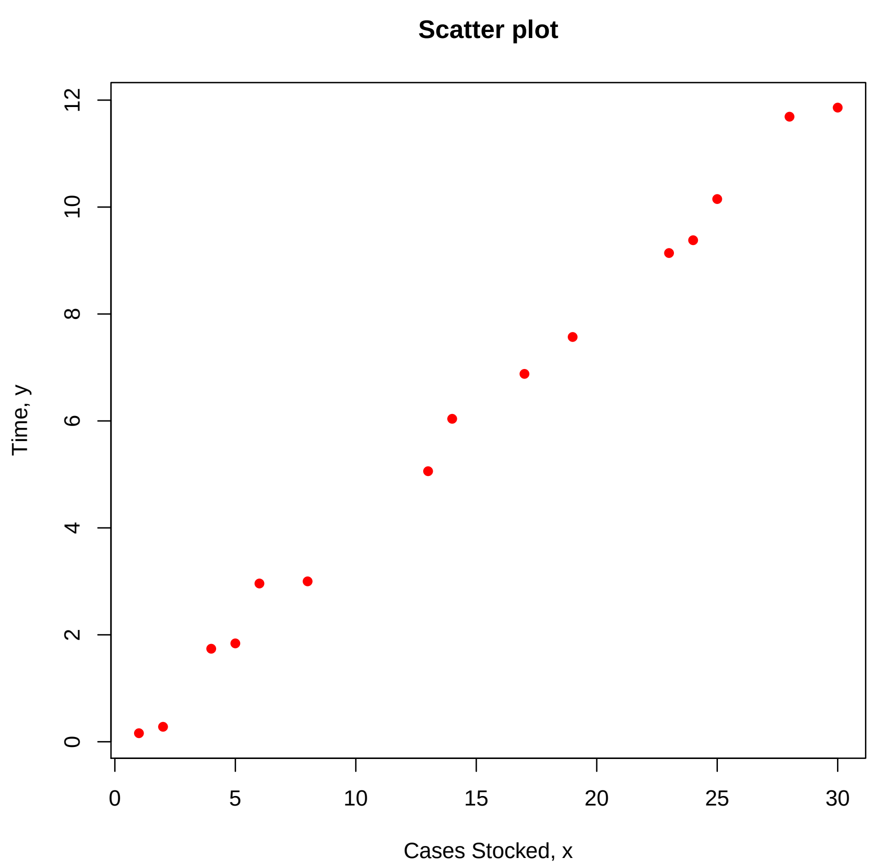
```
rocket.lm<-lm(strength~age, data=rocket)
rocket.lm


Call:
lm(formula = strength ~ age, data = rocket)

Coefficients:
(Intercept)          age
     2627.8        -37.2
```

```
summary(rocket.lm)


Call:
lm(formula = strength ~ age, data = rocket)

Residuals:
   Min     1Q Median     3Q    Max 
-216.0  -50.7   28.7   66.6  106.8 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2627.82      44.18    59.5  < 2e-16 ***
age           -37.15       2.89   -12.9  1.6e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 96.1 on 18 degrees of freedom
Multiple R-squared:  0.902,Adjusted R-squared:  0.896 
F-statistic:  165 on 1 and 18 DF,  p-value: 1.64e-10

R2<-SSR/SST
R2

[1] 0.9018

adjR2<-1-(n-1)/(n-2)*(1-R2)
adjR2

[1] 0.8964
```

2 Regression through the origin

```
shelfstock<-read.csv(file="D:/chilo/Regression 2/Shelf-Stocking.csv",
                     header=T)
shelfstock

    Time Cases
1  10.15    25
2   2.96     6
3   3.00     8
4   6.88    17
5   0.28     2
6   5.06    13
```

```
7   9.14    23
8  11.86    30
9  11.69    28
10  6.04    14
11  7.57    19
12  1.74     4
13  9.38    24
14  0.16     1
15  1.84     5

attach(shelfstock)
plot(Cases, Time, pch=16, cex=1, col=2, xlab="Cases Stocked, x",
     ylab="Time, y", main="Scatter plot")  # add title
```



2.1 Fit a no-intercept regression model

```
model0<-lm(Time~Cases-1, data=shelfstock) # omitting intercept
model0


Call:
lm(formula = Time ~ Cases - 1, data = shelfstock)

Coefficients:
Cases
0.403

summary(model0) # estimated coefficients, tests


Call:
lm(formula = Time ~ Cases - 1, data = shelfstock)

Residuals:
   Min     1Q Median     3Q    Max
-0.525 -0.220 -0.120  0.107  0.544

Coefficients:
      Estimate Std. Error t value Pr(>|t|)
Cases  0.40262    0.00442    91.1   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.299 on 14 degrees of freedom
Multiple R-squared:  0.998,Adjusted R-squared:  0.998
F-statistic: 8.31e+03 on 1 and 14 DF,  p-value: <2e-16

MSE0<-sum(model0$residuals^2)/(length(Time)-1)
MSE0

[1] 0.08929
```

2.2 Fit a regression model with intercept

```
model1<-lm(Time~Cases, data=shelfstock) # omitting intercept
model1


Call:
lm(formula = Time ~ Cases, data = shelfstock)
```

```
Coefficients:
(Intercept)        Cases
    -0.0938       0.4071

summary(model1) # estimated coefficients, tests


Call:
lm(formula = Time ~ Cases, data = shelfstock)

Residuals:
   Min     1Q Median     3Q    Max
-0.440 -0.158 -0.102  0.136  0.611

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.09376    0.14358   -0.65     0.53
Cases        0.40711    0.00822   49.52  3.4e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.305 on 13 degrees of freedom
Multiple R-squared:  0.995,Adjusted R-squared:  0.994
F-statistic: 2.45e+03 on 1 and 13 DF,  p-value: 3.4e-16

MSE1<-sum(model1$residuals^2)/(length(Time)-2)
MSE1

[1] 0.09311
```

2.3 Compute coefficients, fitted values and residuals for no-intercept model by formula

```
attach(shelfstock)

The following objects are masked from shelfstock (position 3):

    Cases, Time

n<-length(Time)
n

[1] 15

sumx<-sum(Cases)
sumx
```

```
[1] 219

sumx2<-sum(Cases^2)
sumx2

[1] 4575

sumy<-sum(Time)
sumy

[1] 87.75

sumy2<-sum(Time^2)
sumy2

[1] 742.9

sumxy<-sum(Cases*Time)
sumxy

[1] 1842

beta1<-sumxy/sumx2
beta1

[1] 0.4026

yhat<-beta1*Cases
yhat

 [1] 10.0655  2.4157  3.2209  6.8445  0.8052  5.2340  9.2602 12.0786
 [9] 11.2733  5.6367  7.6498  1.6105  9.6628  0.4026  2.0131

e<-Time-yhat
e

 [1]  0.08454  0.54429 -0.22095  0.03548 -0.52524 -0.17404 -0.12023
 [8] -0.21856  0.41668  0.40334 -0.07975  0.12953 -0.28285 -0.24262
[15] -0.17309

sigmahat2<-sum(e^2)/(n-1)
sigmahat2

[1] 0.08929
```

2.4 Compute ANOVA for no-intercept model by formula

```
SST<-sumy2
SST

[1] 742.9

SSR<-beta1*sumxy
SSR

[1] 741.6

SSE<-SST-SSR
SSE

[1] 1.25

MSR<-SSR
MSR

[1] 741.6

MSE<-SSE/(n-1)
MSE

[1] 0.08929

Fratio<-MSR/MSE
Fratio

[1] 8305

pvalue<-1-pf(Fratio,1,n-1)
pvalue

[1] 0
```

2.5 Compute t tests for no-intercept model by formula

```
SE.beta1<-sqrt(MSE/sumx2)
SE.beta1

[1] 0.004418

t1<-beta1/SE.beta1
t1

[1] 91.13
```

```
pvalue1<-2*(1-pt(abs(t1),n-1))
pvalue1

[1] 0
```

2.6 Compute confidence intervals for no-intercept model by formula

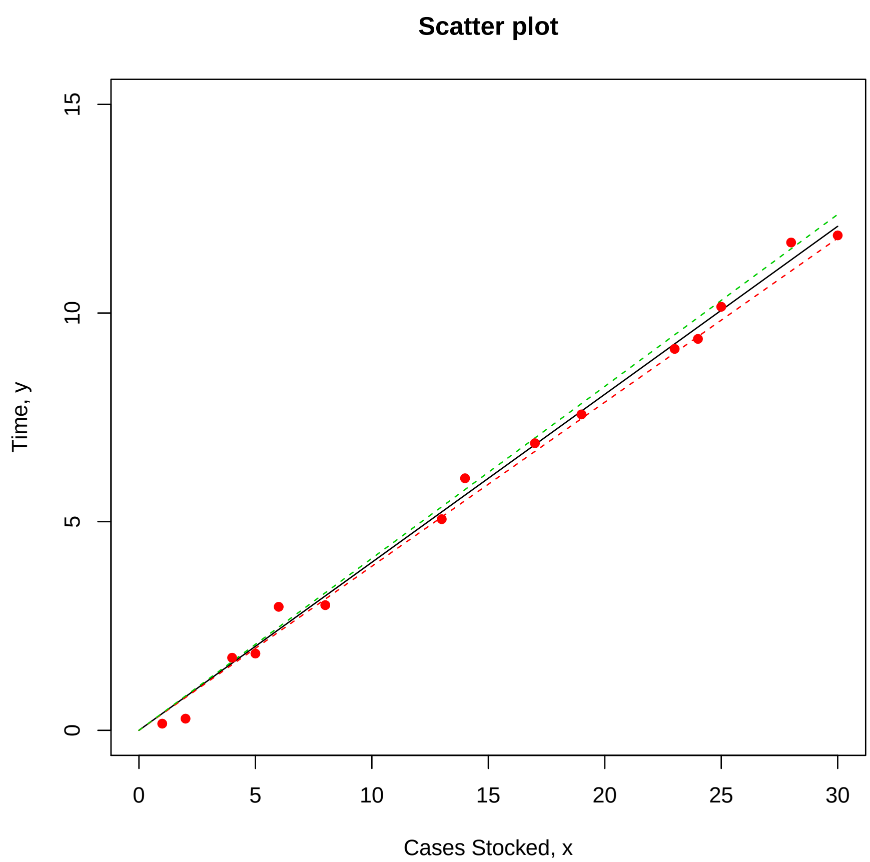
```
qt(.975, n-1)

[1] 2.145

# 95% C.I. for beta1
#
c(beta1-qt(.975, n-1)*SE.beta1, beta1+qt(.975, n-1)*SE.beta1)

[1] 0.3931 0.4121
```

2.7 Compute and plot confidence limits and prediction intervals for no-intercept model

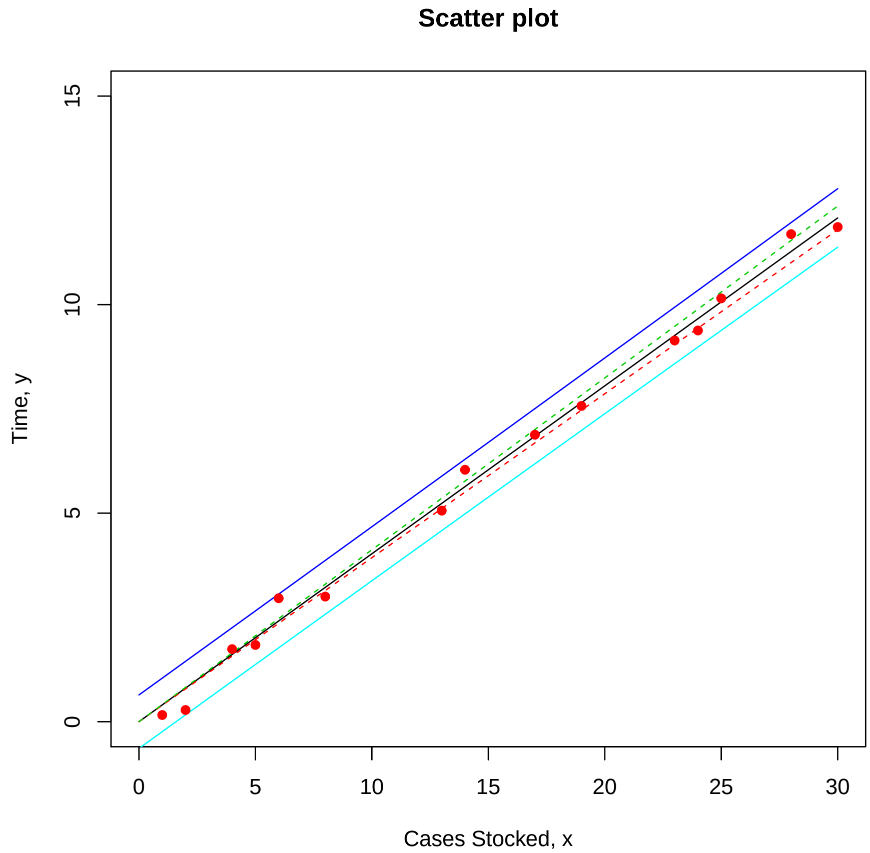
```
x0<-c(0,5,10,15,20,25,30)
muhat<-beta1*x0
muhat

[1]  0.000  2.013  4.026  6.039  8.052 10.065 12.079

bm<-qt(.975, n-1)*sqrt(MSE*(x0^2)/sumx2)
bm

[1] 0.00000 0.04738 0.09475 0.14213 0.18951 0.23689 0.28426

lcl<-muhat-bm
lcl

[1]  0.000  1.966  3.931  5.897  7.863  9.829 11.794

ucl<-muhat+bm
ucl

[1]  0.000  2.060  4.121  6.181  8.242 10.302 12.363
```

```
y0<-beta1*x0
bm2<-qt(.975, n-1)*sqrt(MSE*(1+(x0^2)/sumx2))
lpi<-y0+bm2
lpi

[1]  0.6409  2.6558  4.6741  6.6958  8.7207 10.7488 12.7797

upi<-y0-bm2
upi

[1] -0.6409  1.3704  3.3783  5.3828  7.3840  9.3822 11.3774
```

```
matplot(x0, cbind(muhat, lcl, ucl), ylim=c(0,15), xlim=c(0,30),
        type="l", lty=c(1,2,2),
        xlab="Cases Stocked, x", ylab="Time, y", main="Scatter plot")
points(Cases, Time, ylim=c(0,15), xlim=c(0,30), pch=16, col=2)
```



```r
matplot(x0, cbind(muhat, lcl, ucl, lpi, upi), ylim=c(0,15), xlim=c(0,30),
        type="l", lty=c(1,2,2,1,1),
        xlab="Cases Stocked, x", ylab="Time, y", main="Scatter plot")
points(Cases, Time, ylim=c(0,15), xlim=c(0,30), pch=16, col=2)
```



2.8 Compute coefficient of determination $R^2 for no-intercept model$

```r
model0<-lm(Time~Cases-1, data=shelfstock)
model0


Call:
lm(formula = Time ~ Cases - 1, data = shelfstock)

Coefficients:
```

```
Cases
0.403

summary(model0)


Call:
lm(formula = Time ~ Cases - 1, data = shelfstock)

Residuals:
   Min     1Q Median     3Q    Max
-0.525 -0.220 -0.120  0.107  0.544

Coefficients:
      Estimate Std. Error t value Pr(>|t|)
Cases  0.40262    0.00442    91.1   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.299 on 14 degrees of freedom
Multiple R-squared:  0.998,Adjusted R-squared:  0.998
F-statistic: 8.31e+03 on 1 and 14 DF,  p-value: <2e-16

R2<-SSR/SST
R2

[1] 0.9983

adjR2<-1-(n-1)/(n-2)*(1-R2)
adjR2

[1] 0.9982
```

3 Correlation analysis

```
delivery<-read.csv(file="D:/chilo/Regression 2/Delivery.csv",
                   header=T)
delivery

   Observation  Time Cases
1            1 16.68     7
2            2 11.50     3
3            3 12.03     3
4            4 14.88     4
```

```
5            5 13.75     6
6            6 18.11     7
7            7  8.00     2
8            8 17.83     7
9            9 79.24    30
10          10 21.50     5
11          11 40.33    16
12          12 21.00    10
13          13 13.50     4
14          14 19.75     6
15          15 24.00     9
16          16 29.00    10
17          17 15.35     6
18          18 19.00     7
19          19  9.50     3
20          20 35.10    17
21          21 17.90    10
22          22 52.32    26
23          23 18.75     9
24          24 19.83     8
25          25 10.75     4

attach(delivery)

The following objects are masked from shelfstock (position 3):

    Cases, Time
The following objects are masked from shelfstock (position 4):

    Cases, Time
```

```
r<-cor(Time, Cases) # compute the correlation coefficeint
r

[1] 0.9646
```

3.1 Test for pupulation correlation coefficient

```
n<-length(Time)
t<-r/sqrt((1-r^2)/(n-2))  # test for rao=0
t

[1] 17.55
```

```r
qt(.975, n-2)
```

```
[1] 2.069
```

```r
pvalue<-2*(1-pt(abs(t),n-2))
pvalue
```

```
[1] 8.216e-15
```

3.2 Test for pupulation correlation coefficient

```r
# test for rao=0.9
w<-atanh(r)
w
```

```
[1] 2.008
```

```r
muw<-atanh(0.9)
muw
```

```
[1] 1.472
```

```r
z<-(w-muw)/sqrt(1/(n-3))
z
```

```
[1] 2.515
```

```r
qnorm(.975,0,1)
```

```
[1] 1.96
```

```r
pvalue<-2*(1-pnorm(abs(z),0,1))
pvalue
```

```
[1] 0.01191
```

3.3 Approximate 95 C.I. for arc hyp tan rao

```r
atanh(r)
```

```
[1] 2.008
```

```r
qnorm(0.975,0,1)
```

```
[1] 1.96
```

```r
c(atanh(r)-qnorm(0.975,0,1)/sqrt(n-3), atanh(r)+qnorm(0.975,0,1)/sqrt(n-3))
```

```
[1] 1.591 2.426
```

3.4 Approximate 95 C.I. for rao

```r
tanh(c(atanh(r)-qnorm(0.975,0,1)/sqrt(n-3), atanh(r)+qnorm(0.975,0,1)/sqrt(n-3)))
```

```
[1] 0.9202 0.9845
```