TP1_MRRR

Zeyu CHEN

2018/9/17

Some preliminary exercices using R

1

```
options(digits=15)

vecExp2 <- 0:19

vecExp2 <- 2**vecExp2/factorial(vecExp2)

ifelse(vecExp2 > 10e-8,vecExp2,NaN)
```

```
##  [1] 1.00000000000000e+00 2.00000000000000e+00 2.00000000000000e+00
##  [4] 1.33333333333333e+00 6.66666666666667e-01 2.66666666666667e-01
##  [7] 8.88888888888889e-02 2.53968253968254e-02 6.34920634920635e-03
## [10] 1.41093474426808e-03 2.82186948853615e-04 5.13067179733846e-05
## [13] 8.55111966223077e-06 1.31555687111243e-06 1.87936695873204e-07
## [16]                  NaN                  NaN                  NaN
## [19]                  NaN                  NaN
```

```
exp2 <- sum(vecExp2)
exp2Fun <- exp(2)
exp2
```

```
## [1] 7.38905609893017
```

```
exp2Fun
```

```
## [1] 7.38905609893065
```

2

```
X <- rnorm(100,2,1)
X
```

```
##   [1]  3.696370197811979  2.770705233653624  2.167933161069842
##   [4]  3.334088995621994  3.425087352930372  0.660150501524427
##   [7]  2.426064723652296  1.881469772862444  3.830445380179677
##  [10]  1.517194335735426  2.197801781842624  2.430392246838977
##  [13]  1.406821487507959  2.337419053790716  1.763254972884713
##  [16]  3.471679915774573  2.028409829269662  3.769172955029593
##  [19]  2.148074254811381  2.129561179007547  1.685601125198869
##  [22]  2.129307178556030  1.743982851292009  1.436453666535053
##  [25]  1.792150679335178  1.794262026944054  5.123607321043321
##  [28]  2.680153027671806  0.768417597828371  1.961355626057995
##  [31]  1.635145803152051  1.824988694761876  2.292056580262457
##  [34] -0.199835055208149  3.771272203960827  2.500233556628374
```

```
## [37]  1.589084476943892  1.956879117950055  1.844052885012695
## [40]  1.482077126255446  1.315000193129441  0.855637499242235
## [43]  1.187994323758587  2.291881623028387  1.586403296354289
## [46]  1.537686290935210  1.158936232209492  2.451576952509443
## [49]  1.966823182155690  1.661408296437407  2.099052910209147
## [52]  1.247898702508364  2.453304309804838 -0.281332576944774
## [55]  0.480095450684390  0.981066941764108  2.267888168448292
## [58]  1.035159515682608  1.118616604967140  1.279515319532738
## [61]  1.631431256884319  0.851369136206415  3.831701260852801
## [64]  2.181417051023359  1.901873704947400  3.985194550394738
## [67]  2.270990647583128  3.039110807119953  2.999693350927021
## [70]  0.409668496781672  1.460731085158276  0.712650576562297
## [73]  2.570600885436683  0.371127304036526  2.359395302441494
## [76]  2.230973007244742  0.850406780064600  2.403975145473124
## [79]  1.188603768247239  1.343294166158727  1.026543653764786
## [82]  3.907293167184541  0.434897162680367  1.496046918000502
## [85]  0.995292396200524  2.200879258987288  1.802973452725702
## [88]  1.791968741479967  1.175092470586086  3.993593529066308
## [91]  2.964535657097422 -0.364794197898023  2.960936008293669
## [94]  3.340733626440563  1.550204498235391  3.781204503748137
## [97]  2.154259336969947  0.620050547166380  0.967127930308633
## [100]  3.985684768567337
```

```r
Y<- X*9.8
Y
```

```
##   [1] 36.22442793855740 27.15291128980552 21.24574497848446
##   [4] 32.67407215709554 33.56585605871765  6.46947491493939
##   [7] 23.77543429179250 18.43840377405195 37.53836472576084
##  [10] 14.86850449020718 21.53845746205772 23.81784401902198
##  [13] 13.78685057757800 22.90670672714901 17.27989873427019
##  [16] 34.02246317459083 19.87841632684269 36.93789495929002
##  [19] 21.05112769715153 20.86969955427396 16.51889102694891
##  [22] 20.86721034984910 17.09103194266169 14.07724593204352
##  [25] 17.56307665748474 17.58376786405173 50.21135174622454
##  [28] 26.26549967118370  7.53049245871804 19.22128513536835
##  [31] 16.02442887089010 17.88488920866638 22.46215448657208
##  [34] -1.95838354103986 36.95846759881611 24.50228885495807
##  [37] 15.57302787405015 19.17741535591054 18.07171827312441
##  [40] 14.52435583730337 12.88700189266853  8.38524749257390
##  [43] 11.64234437283416 22.46043990567819 15.54675230427203
##  [46] 15.06932565116506 11.35757507565302 24.02545413459254
##  [49] 19.27486718512576 16.28180130508659 20.57071852004964
##  [52] 12.22940728458197 24.04238223608742 -2.75705925405878
##  [55]  4.70493541670702  9.61445602928826 22.22530405079327
##  [58] 10.14456325368956 10.96244272867797 12.53925013142083
##  [61] 15.98802631746632  8.34341753482287 37.55067235635746
##  [64] 21.37788710002893 18.63836230848452 39.05490659386843
##  [67] 22.25570834631466 29.78328590977554 29.39699483908481
##  [70]  4.01475126846038 14.31516463455110  6.98397565031051
##  [73] 25.19188867727949  3.63704757955795 23.12207396392665
##  [76] 21.86353547099847  8.33398644463308 23.55895642563662
##  [79] 11.64831692882294 13.16428282835553 10.06012780689490
##  [82] 38.29147303840850  4.26199219426760 14.66125979640492
##  [85]  9.75386548276514 21.56861673807542 17.66913983671189
```

```
##  [88] 17.56129366650368 11.51590621174364 39.13721658484982
##  [91] 29.05244943955474 -3.57498313940062 29.01717288127796
##  [94] 32.73918953911752 15.19200408270683 37.05580413673174
##  [97] 21.11174150230548  6.07649536223052  9.47785371702460
## [100] 39.05971073195990
```

```r
Y<- Y + rnorm(100,0,1/10)
Y
```

```
##   [1] 36.26996597264141 27.15584368286268 21.29748700514993
##   [4] 32.59544933941633 33.71857506410027  6.45094894055767
##   [7] 23.57000905454849 18.45757096817719 37.63614057623924
##  [10] 14.83297333540887 21.54083219145189 23.78311416924187
##  [13] 13.76220780000121 23.03898211898244 17.40772899349749
##  [16] 33.98408399906737 19.90948574221509 36.92600703562571
##  [19] 20.95517746436380 20.73168688167556 16.53527480238708
##  [22] 20.68076039705763 17.08665417507300 14.13141930595518
##  [25] 17.45063300274658 17.60532166777738 50.09013335098771
##  [28] 26.28811509078788  7.55758809544233 19.14073537115178
##  [31] 15.91167101555200 18.02605721228631 22.52545754713207
##  [34] -1.73501716039391 36.89858407585201 24.46977378159315
##  [37] 15.77076982204956 19.21133272861112 18.27606336521176
##  [40] 14.60501233140130 12.94872876003885  8.46359694144289
##  [43] 11.59926500551942 22.56216792536139 15.51011507299322
##  [46] 15.04543468741787 11.12707795888445 24.08799218936142
##  [49] 19.24604208677216 16.42258334615351 20.54482110274946
##  [52] 12.34660676218619 24.12088420974419 -2.69289469511277
##  [55]  4.63431061478757  9.82257080332267 22.12744101036454
##  [58] 10.25076624757465 10.84148284408467 12.59426883233272
##  [61] 16.01412963100275  8.28225466796087 37.58014114191805
##  [64] 21.29130648759161 18.60828604426972 38.99200321266174
##  [67] 22.25761285153935 29.79330514286492 29.20474127775514
##  [70]  4.32007880072156 14.43307589335862  6.95890271473353
##  [73] 25.21504423309208  3.65865848080856 23.12118882000681
##  [76] 21.94857316607278  8.43136779528830 23.59663715252677
##  [79] 11.56842102520263 13.15709593685990 10.02575836083471
##  [82] 38.27723120300045  4.31423172214091 14.60080045203070
##  [85]  9.83278499653455 21.39831557043161 17.65917496750057
##  [88] 17.76390872991439 11.64502814697404 39.06250110203233
##  [91] 29.03187831443645 -3.60639630285457 29.13362898525666
##  [94] 32.65118775015357 15.24682177881943 36.91778360867844
##  [97] 21.11140942664519  5.95272594439783  9.48445166577624
## [100] 39.11446880461866
```

3

```r
dfxy <- data.frame(X,Y)
write.table(dfxy,"WXY")

rXY <- read.table("WXY")
rXY-dfxy
```

```
##                          X                     Y
## 1    8.88178419700125e-16 -7.10542735760100e-15
```

```
## 2   -3.99680288865056e-15  1.42108547152020e-14
## 3   -2.66453525910038e-15 -3.19744231092045e-14
## 4   -3.99680288865056e-15 -2.84217094304040e-14
## 5   -2.66453525910038e-15  2.84217094304040e-14
## 6   -1.11022302462516e-16  1.77635683940025e-15
## 7    3.99680288865056e-15  1.06581410364015e-14
## 8   -3.55271367880050e-15  1.42108547152020e-14
## 9    2.66453525910038e-15 -3.55271367880050e-14
## 10   3.99680288865056e-15  2.66453525910038e-14
## 11  -3.99680288865056e-15  1.42108547152020e-14
## 12   2.66453525910038e-15  3.19744231092045e-14
## 13   1.33226762955019e-15 -1.24344978758018e-14
## 14   4.44089209850063e-15 -3.90798504668055e-14
## 15  -2.88657986402541e-15  7.10542735760100e-15
## 16  -3.55271367880050e-15  2.84217094304040e-14
## 17  -2.22044604925031e-15  1.42108547152020e-14
## 18  -3.55271367880050e-15 -7.10542735760100e-15
## 19  -8.88178419700125e-16 -3.55271367880050e-15
## 20   3.10862446895044e-15  3.55271367880050e-14
## 21   1.33226762955019e-15  1.77635683940025e-14
## 22  -4.44089209850063e-16 -2.48689957516035e-14
## 23   1.33226762955019e-15  3.55271367880050e-15
## 24  -3.33066907387547e-15  2.13162820728030e-14
## 25   2.22044604925031e-15  2.13162820728030e-14
## 26  -3.99680288865056e-15  2.13162820728030e-14
## 27  -8.88178419700125e-16 -1.42108547152020e-14
## 28   4.44089209850063e-15  2.13162820728030e-14
## 29   1.11022302462516e-16 -3.55271367880050e-15
## 30  -4.66293670342566e-15  2.13162820728030e-14
## 31  -8.88178419700125e-16  3.55271367880050e-15
## 32   4.21884749357559e-15 -7.10542735760100e-15
## 33   3.10862446895044e-15  2.48689957516035e-14
## 34   3.60822483003176e-16  3.33066907387547e-15
## 35   3.55271367880050e-15 -1.42108547152020e-14
## 36  -3.99680288865056e-15  4.97379915032070e-14
## 37  -2.22044604925031e-15  3.90798504668055e-14
## 38  -4.88498130835069e-15 -2.13162820728030e-14
## 39  -4.66293670342566e-15  3.90798504668055e-14
## 40   3.77475828372553e-15 -1.77635683940025e-15
## 41  -1.33226762955019e-15  4.97379915032070e-14
## 42   3.33066907387547e-16  0.00000000000000e+00
## 43   2.44249065417534e-15 -1.42108547152020e-14
## 44   3.10862446895044e-15  7.10542735760100e-15
## 45   1.11022302462516e-15 -2.30926389122033e-14
## 46   4.44089209850063e-16  2.66453525910038e-14
## 47  -1.99840144432528e-15  4.97379915032070e-14
## 48  -2.66453525910038e-15 -1.77635683940025e-14
## 49   4.44089209850063e-16  3.90798504668055e-14
## 50   2.88657986402541e-15 -1.42108547152020e-14
## 51   3.55271367880050e-15  3.90798504668055e-14
## 52  -3.55271367880050e-15  5.32907051820075e-15
## 53   1.77635683940025e-15  1.42108547152020e-14
## 54  -3.33066907387547e-16 -4.44089209850063e-16
## 55   5.55111512312578e-17  2.66453525910038e-15
```

```
## 56   0.00000000000000e+00 -1.77635683940025e-15
## 57  -2.22044604925031e-15 -3.55271367880050e-14
## 58   2.22044604925031e-15 -4.79616346638068e-14
## 59   0.00000000000000e+00  3.01980662698043e-14
## 60   1.99840144432528e-15 -2.30926389122033e-14
## 61   1.11022302462516e-15 -4.97379915032070e-14
## 62  -3.33066907387547e-16  5.32907051820075e-15
## 63  -1.33226762955019e-15 -4.97379915032070e-14
## 64   4.44089209850063e-16 -7.10542735760100e-15
## 65  -4.44089209850063e-16 -1.77635683940025e-14
## 66   2.22044604925031e-15 -3.55271367880050e-14
## 67   2.22044604925031e-15  4.61852778244065e-14
## 68  -3.10862446895044e-15 -1.77635683940025e-14
## 69  -1.33226762955019e-15 -3.90798504668055e-14
## 70   2.77555756156289e-16 -8.88178419700125e-16
## 71   4.44089209850063e-15 -1.77635683940025e-14
## 72   0.00000000000000e+00  3.55271367880050e-15
## 73  -2.66453525910038e-15  2.13162820728030e-14
## 74   2.77555756156289e-16  2.22044604925031e-15
## 75  -3.99680288865056e-15 -1.42108547152020e-14
## 76  -1.77635683940025e-15  2.13162820728030e-14
## 77   0.00000000000000e+00 -1.77635683940025e-15
## 78  -3.55271367880050e-15  3.19744231092045e-14
## 79   1.11022302462516e-15 -3.37507799486048e-14
## 80   2.66453525910038e-15  0.00000000000000e+00
## 81   4.21884749357559e-15 -1.06581410364015e-14
## 82  -8.88178419700125e-16 -4.97379915032070e-14
## 83   1.66533453693773e-16 -3.55271367880050e-15
## 84  -1.99840144432528e-15  5.32907051820075e-15
## 85   0.00000000000000e+00 -1.77635683940025e-15
## 86   1.77635683940025e-15 -1.06581410364015e-14
## 87  -2.44249065417534e-15  3.55271367880050e-14
## 88   2.66453525910038e-15  7.10542735760100e-15
## 89   4.21884749357559e-15 -3.90798504668055e-14
## 90   2.22044604925031e-15 -2.84217094304040e-14
## 91  -1.77635683940025e-15 -4.61852778244065e-14
## 92  -2.77555756156289e-16  1.33226762955019e-15
## 93   8.88178419700125e-16  4.26325641456060e-14
## 94  -2.66453525910038e-15  3.55271367880050e-14
## 95  -1.11022302462516e-15 -2.66453525910038e-14
## 96   2.66453525910038e-15 -4.26325641456060e-14
## 97   3.10862446895044e-15  7.10542735760100e-15
## 98   0.00000000000000e+00 -3.55271367880050e-15
## 99  -1.11022302462516e-16  1.77635683940025e-15
## 100  3.10862446895044e-15  4.26325641456060e-14
```

```
#tous les valeur sont changés un peu
```

4

```
dfxy <- data.frame(X,Y)
save(dfxy,file = "xy.RData")
pdfxy <- dfxy
```

```r
rm(dfxy)
load("xy.RData")
pdfxy - dfxy
```

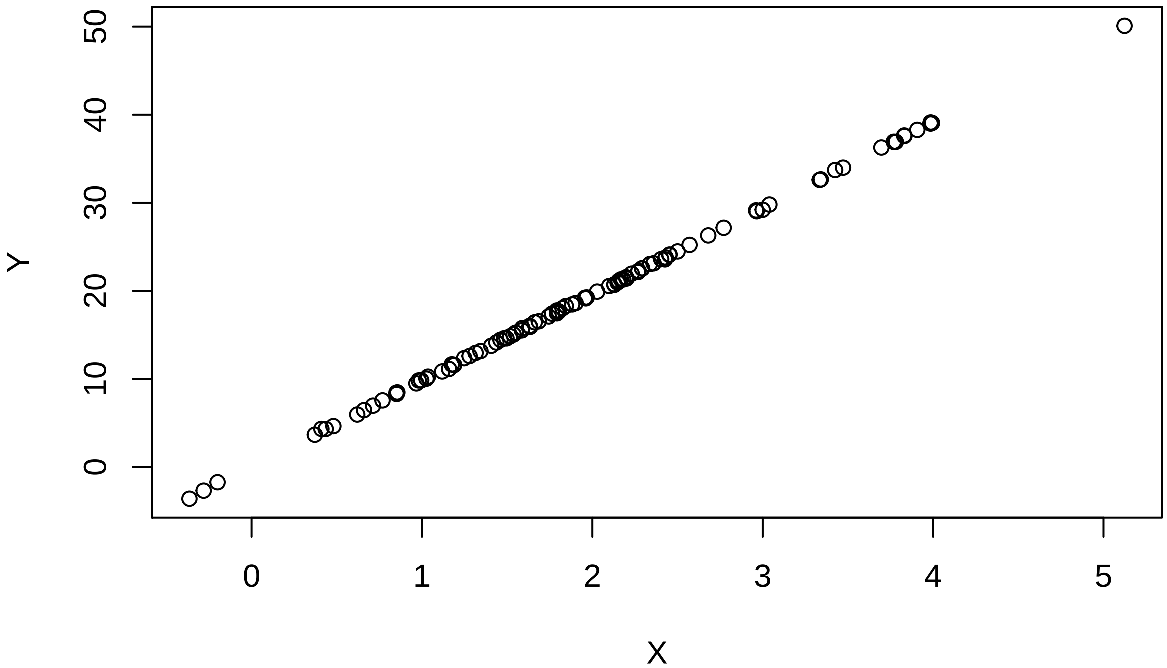
```
##     X Y
## 1   0 0
## 2   0 0
## 3   0 0
## 4   0 0
## 5   0 0
## 6   0 0
## 7   0 0
## 8   0 0
## 9   0 0
## 10  0 0
## 11  0 0
## 12  0 0
## 13  0 0
## 14  0 0
## 15  0 0
## 16  0 0
## 17  0 0
## 18  0 0
## 19  0 0
## 20  0 0
## 21  0 0
## 22  0 0
## 23  0 0
## 24  0 0
## 25  0 0
## 26  0 0
## 27  0 0
## 28  0 0
## 29  0 0
## 30  0 0
## 31  0 0
## 32  0 0
## 33  0 0
## 34  0 0
## 35  0 0
## 36  0 0
## 37  0 0
## 38  0 0
## 39  0 0
## 40  0 0
## 41  0 0
## 42  0 0
## 43  0 0
## 44  0 0
## 45  0 0
## 46  0 0
## 47  0 0
## 48  0 0
## 49  0 0
```

```
## 50  0 0
## 51  0 0
## 52  0 0
## 53  0 0
## 54  0 0
## 55  0 0
## 56  0 0
## 57  0 0
## 58  0 0
## 59  0 0
## 60  0 0
## 61  0 0
## 62  0 0
## 63  0 0
## 64  0 0
## 65  0 0
## 66  0 0
## 67  0 0
## 68  0 0
## 69  0 0
## 70  0 0
## 71  0 0
## 72  0 0
## 73  0 0
## 74  0 0
## 75  0 0
## 76  0 0
## 77  0 0
## 78  0 0
## 79  0 0
## 80  0 0
## 81  0 0
## 82  0 0
## 83  0 0
## 84  0 0
## 85  0 0
## 86  0 0
## 87  0 0
## 88  0 0
## 89  0 0
## 90  0 0
## 91  0 0
## 92  0 0
## 93  0 0
## 94  0 0
## 95  0 0
## 96  0 0
## 97  0 0
## 98  0 0
## 99  0 0
## 100 0 0
```

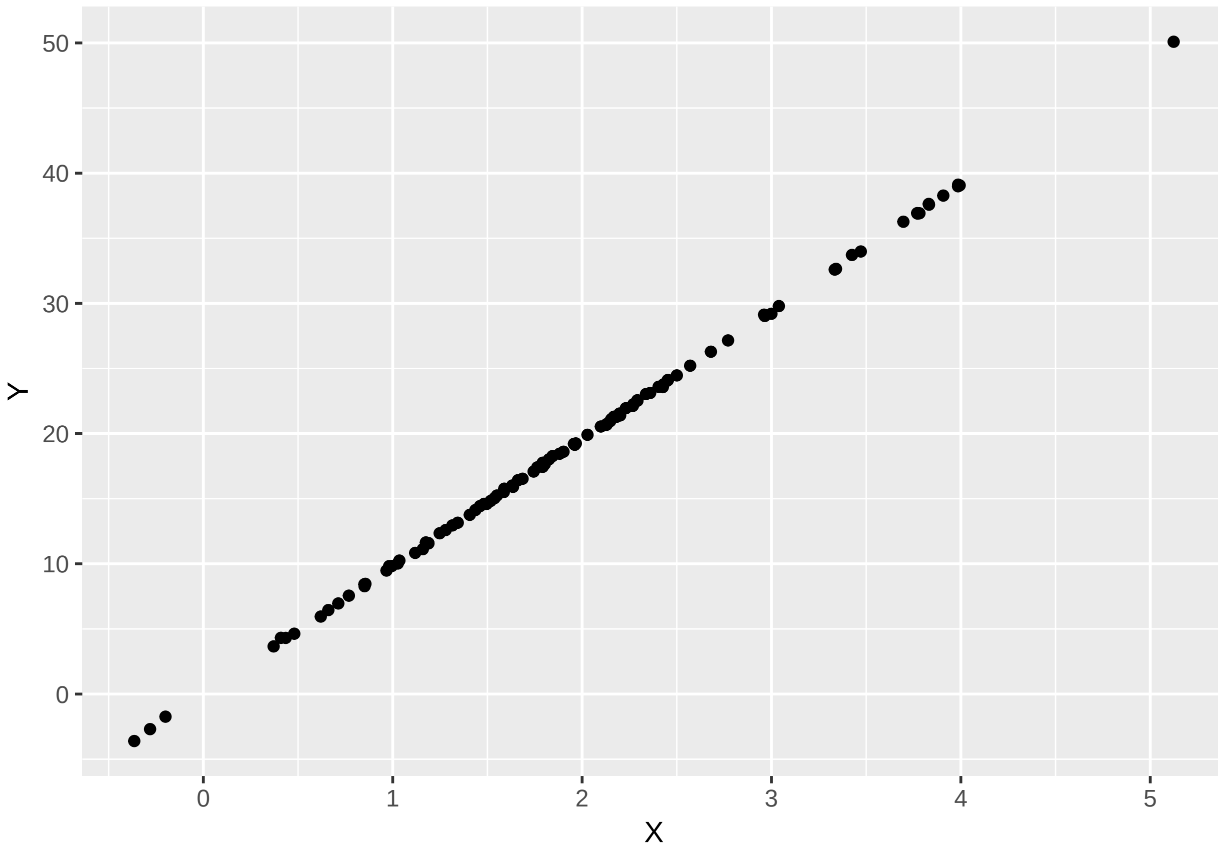
```
#tous les valeur ne sont pas changés
```

5

```r
library(ggplot2)
p <- plot(X,Y)
```

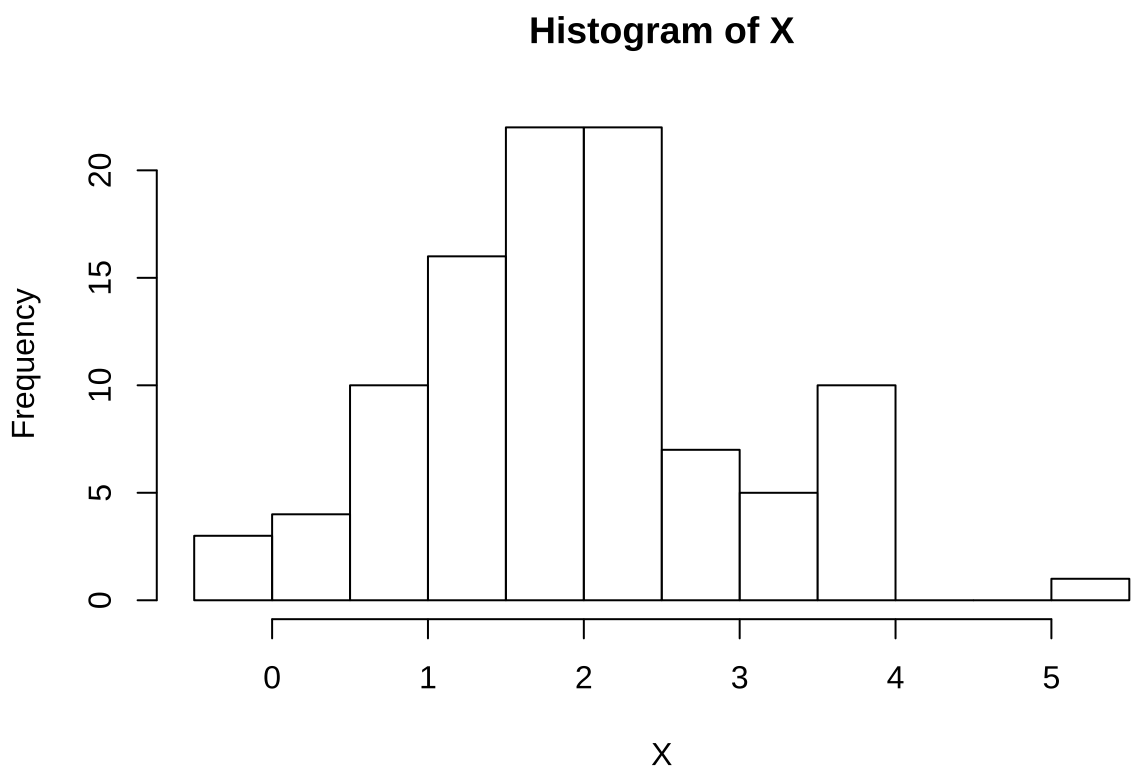



```r
ggplot(dfxy, aes(X, Y)) + geom_point()
```

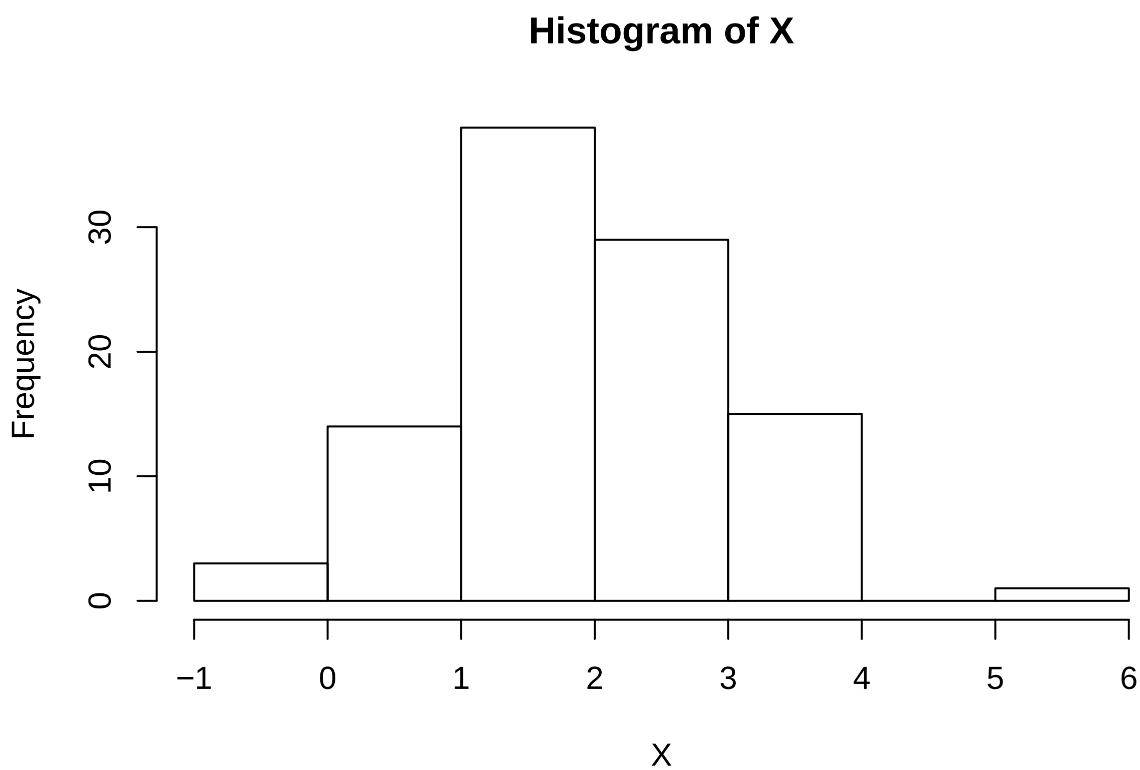


6

```r
hist(X)
```




```r
hist(X, breaks = 6)
```



1. Ordinary Least Square (OLS)

Priliminary work

```r
tab <- read.table("TP1/data/immo.txt",sep = ";",header = TRUE)
tab
```

```
##    surface valeur prix
## 1    153.1    573  748
## 2    152.0    638  740
## 3    162.5    654  729
## 4    143.3    570  700
## 5    145.7    638  749
## 6    173.3    632  760
## 7    144.8    602  720
## 8    149.1    577  735
## 9    152.5    564  745
## 10   138.9    556  735
## 11   151.8    626  715
## 12   144.4    634  710
## 13   148.7    602  789
## 14   186.3    672  865
## 15   152.0    571  680
## 16   257.6    896 1020
## 17   190.5    686  840
## 18   153.7    601  690
## 19   180.6    663  880
## 20   163.5    658  760
```

```r
head(tab)
```

```
##   surface valeur prix
## 1   153.1    573  748
## 2   152.0    638  740
## 3   162.5    654  729
## 4   143.3    570  700
## 5   145.7    638  749
## 6   173.3    632  760
```

```r
tail(tab)
```

```
##    surface valeur prix
## 15   152.0    571  680
## 16   257.6    896 1020
## 17   190.5    686  840
## 18   153.7    601  690
## 19   180.6    663  880
## 20   163.5    658  760
```

```r
#take the nom of all of colums
names(tab)
```

```
## [1] "surface" "valeur"  "prix"
```

```r
#take the first colums
#tab[,1]
```

```r
#take the  colums 'surface'
#tab$surface

#take the first and third colum
#tab[,c(1,3)]

#tab$prix

#nrow(tab)

#dim(tab)

# plot(tab)
cor(tab)
```

```
##                   surface            valeur              prix
## surface 1.000000000000000 0.925710552993891 0.913319494330805
## valeur  0.925710552993891 1.000000000000000 0.851384140800712
## prix    0.913319494330805 0.851384140800712 1.000000000000000
```

Fisrt model using Ordinary Least Square

a)

```r
c1 = rep(1,20)
tab <- read.table("TP1/data/immo.txt",sep = ";",header = TRUE)
tab=data.frame(c1,tab)

X <- as.matrix(tab[,c(1,2,3)])
Y <- as.matrix(tab[,4])


belta <- solve(t(X)%*%X)%*%t(X)%*%Y

belta
```

```
##                         [,1]
## c1      309.6656633509080621
## surface   2.6343996248646016
## valeur    0.0451838603054422
```

```r
#beta0 <- belta[1,1]
#beta1 <- belta[2,1]
#beta2 <- belta[3,1]

#beta0+144.8*beta1+602*beta2
```

b)

```r
tab <- read.table("TP1/data/immo.txt",sep = ";",header = TRUE)
  modreg = lm(prix~.,data=tab)
```

```r
#modreg

# print(modreg)

#summary(modreg)

modreg$res
```

```
##                  1                  2                  3
##   9.11740212725132   1.07829079474543 -38.30584703121945
##                  4                  5                  6
## -12.92992996816042  26.67500843139191 -34.76331805303539
##                  7                  8                  9
##   1.67258706476733   6.47426518548715   8.10469664491914
##                 10                 11                 12
##  34.29400242552018 -22.85262295661578  -8.71953661506235
##                 13                 14                 15
##  60.39842852779585  34.18213241150676 -55.89439056478701
##                 16                 17                 18
##  -8.77174554976054  -2.51492005720089 -51.72838573622116
##                 19                 20
##  64.60486501598379 -10.12098209730588
```

```r
tab
```

```
##    surface valeur prix
## 1    153.1    573  748
## 2    152.0    638  740
## 3    162.5    654  729
## 4    143.3    570  700
## 5    145.7    638  749
## 6    173.3    632  760
## 7    144.8    602  720
## 8    149.1    577  735
## 9    152.5    564  745
## 10   138.9    556  735
## 11   151.8    626  715
## 12   144.4    634  710
## 13   148.7    602  789
## 14   186.3    672  865
## 15   152.0    571  680
## 16   257.6    896 1020
## 17   190.5    686  840
## 18   153.7    601  690
## 19   180.6    663  880
## 20   163.5    658  760
```

```r
modreg$model
```

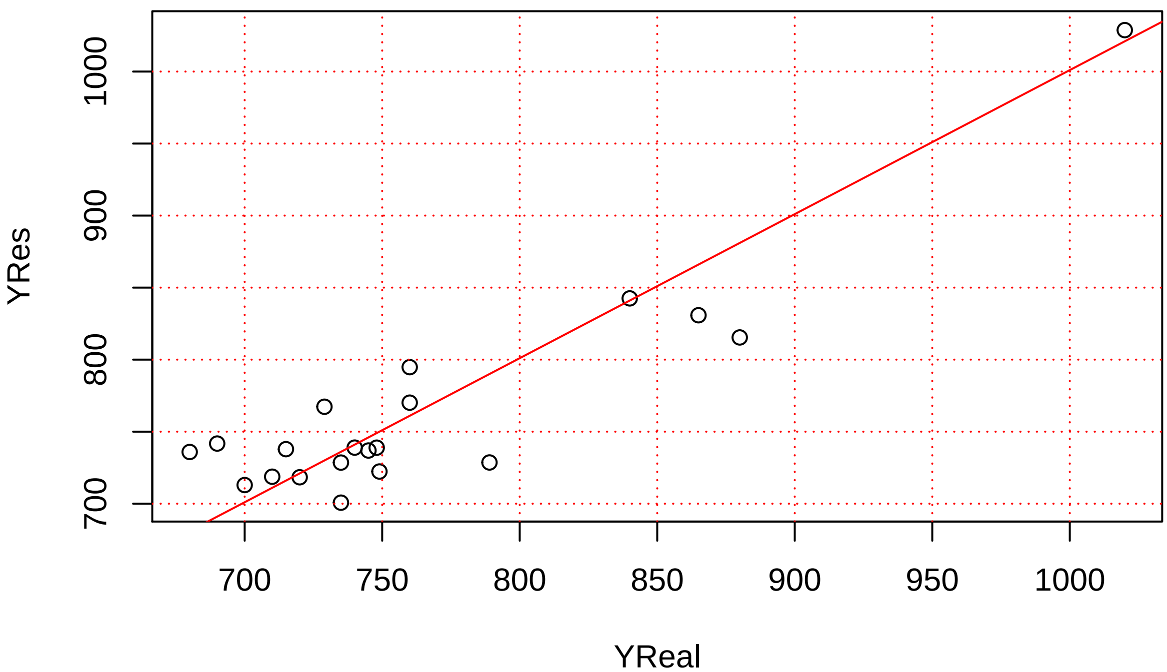
```
##    prix surface valeur
## 1   748   153.1    573
## 2   740   152.0    638
## 3   729   162.5    654
## 4   700   143.3    570
## 5   749   145.7    638
```

```
## 6    760    173.3     632
## 7    720    144.8     602
## 8    735    149.1     577
## 9    745    152.5     564
## 10   735    138.9     556
## 11   715    151.8     626
## 12   710    144.4     634
## 13   789    148.7     602
## 14   865    186.3     672
## 15   680    152.0     571
## 16  1020    257.6     896
## 17   840    190.5     686
## 18   690    153.7     601
## 19   880    180.6     663
## 20   760    163.5     658
```

```r
  help(lm)
```

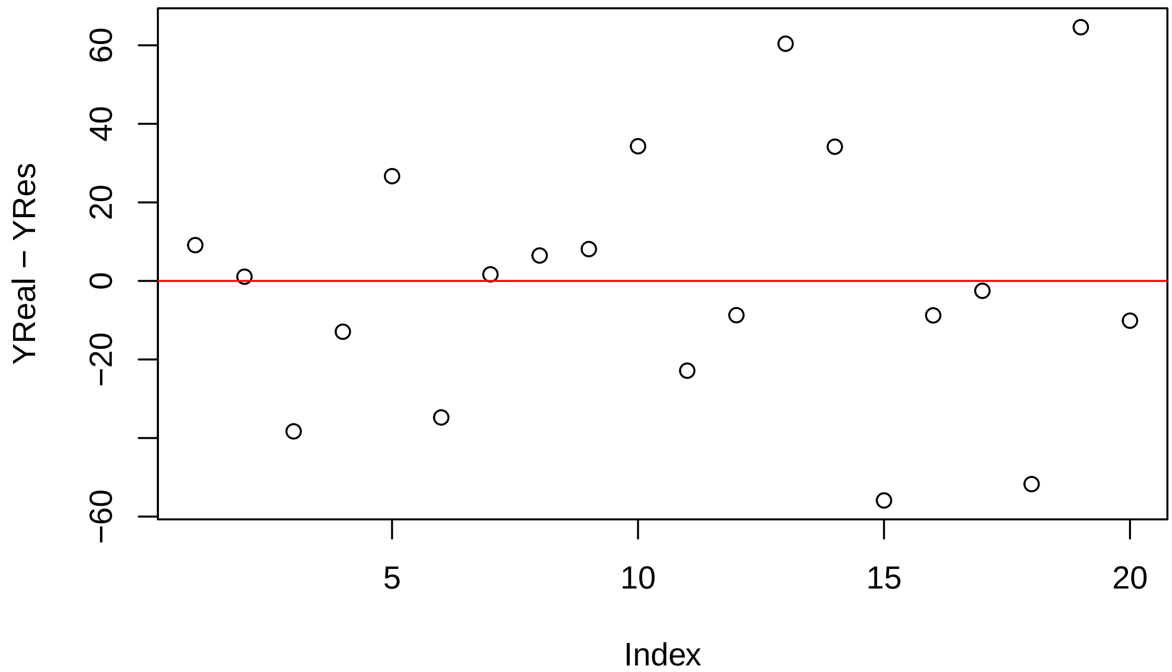
c)

```r
YReal <- tab$prix
YRes  <- modreg$fitted.values
plot(YReal,YRes)
grid(col = "red")
abline(a = 1, b = 1,col = 2)
```




```r
# The points above the red stright line represent the value being over estimated .
# The points below the red stright line represent the value being under estimated

plot(YReal-YRes)
abline(h = 0, col = 2)
```




```
# This is another way to observe values that are underestimated or overestimated.
```

d)

```
#in statistics, the coefficient of determination, denoted R2 or r2 and pronounced "R squared", is the p
YReal <- tab$prix
YRes  <- modreg$fitted.values

YMean <- sum(YReal)/length(YReal)

SSres <- sum((YRes-YReal)^2)
SSreg <- sum((YRes-YMean)^2)
SStot <- sum((YReal-YMean)^2)

R2 <- SSreg/SStot
R2
```

```
## [1] 0.834397032848454
```

```
R2_ <- 1 - SSres/SStot
R2_
```

```
## [1] 0.834397032848455
```

e)

```
#reference a)
```

2. The linear model

a)

```
tab <- read.table("TP1/data/Icecreamdata.txt", header = TRUE, sep = ";")
nrow(tab)
```

```
## [1] 30
```

```
dim(tab)
```

```
## [1] 30  4
```

$$Y = \beta_0 + \sum_{i=1}^{3} \beta_i X_i$$

b)

The consumption is proportional to the income and temps with the similar coefficients, but it is inversely proportional to the price with a large coefficient.

```
modIce <- lm(cons~.,tab)
```

output Model :

```
modIce
```

```
##
## Call:
## lm(formula = cons ~ ., data = tab)
##
## Coefficients:
##       (Intercept)             income              price
##   0.19731507194759   0.00330776043967  -1.04441399193801
##              temp
##   0.00345842973871
```

table anova :

```
anova(modIce)
```

```
## Analysis of Variance Table
##
## Response: cons
##           Df           Sum Sq          Mean Sq  F value   Pr(>F)
## income     1 0.00028842772277 0.00028842772277  0.21260 0.648569
## price      1 0.00822144362519 0.00822144362519  6.06012 0.020776 *
## temp       1 0.08174065756957 0.08174065756957 60.25195  3.1e-08 ***
## Residuals 26 0.03527283774913 0.00135664760574
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
X <-  as.matrix(cbind(rep(1,nrow(tab)),tab[,c(2,3,4)]))
Y <-  as.matrix(tab[,1])
```

d'apres le cours ,on sais $\beta = (X^t X)^{-1} X^t Y$

```r
beta <- solve(t(X)%*%X)%*%t(X)%*%Y

Sjj<- solve(t(X)%*%X)

tValue <- vector(length = 4)

for(i in 1:4)
{
  tValue[i] <- beta[i]/sqrt(0.001357 * Sjj[i,i])
}

interval95 <-  qt(1-0.05/2,26)
#interval95

pValue <- vector(length = 4)
for(i in 1:4)
{
  pValue[i] <- 2* pt(abs(tValue[i]),df = df.residual(modIce),lower.tail = FALSE)
}

tValue <- data.frame(tValue,pValue,row.names = c("b0","b1","b2","b3"))
#tValue

#summary(modIce)
confint(modIce,level = 0.95)
```

```
##                              2.5 %              97.5 %
## (Intercept) -0.358122192681126161 0.75275233657630292
## income       0.000899875229897911 0.00571564564943604
## price       -2.759460028340869897 0.67063204446484792
## temp         0.002542594983190466 0.00437426449422107
```