Clase-4.R

Usuario

2019-08-09

```
#Jesús Alberto Cuéllar
#09/Agosto/2019
#Clase 4

#mide la relacion lineal entre dos variables
#coeficiente de correlacion(r) -1,1

# Lectura de datos ----------------------------------------------------------

library(repmis)
```

```
## Registered S3 method overwritten by 'R.oo':
##   method        from
##   throw.default R.methodsS3
```

```
erupcion <- source_data("https://www.dropbox.com/s/liir6sil7hkqlxs/erupciones.csv?dl=1")
```

```
## Downloading data from: https://www.dropbox.com/s/liir6sil7hkqlxs/erupciones.csv?dl=1
```

```
## SHA-1 hash of the downloaded data file is:
## b07708389ddf62ee20d19c759c88d7dc2d0da3ac
```

```
plot(erupcion$waiting , erupcion$eruptions, pch= 10,
     xlab= "Tiempo de espera (min)",
     ylab= "Duración (min)")

library(pastecs)

stat.desc(erupcion$eruptions,basic= FALSE, norm= TRUE)
```

```
##        median          mean       SE.mean  CI.mean.0.95           var
##  4.000000e+00  3.487783e+00  6.920580e-02  1.362494e-01  1.302728e+00
##       std.dev      coef.var      skewness      skew.2SE      kurtosis
##  1.141371e+00  3.272483e-01 -4.135498e-01 -1.399854e+00 -1.511605e+00
##      kurt.2SE    normtest.W    normtest.p
## -2.567516e+00  8.459156e-01  9.036119e-16
```

```
shapiro.test(erupcion$eruptions)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  erupcion$eruptions
## W = 0.84592, p-value = 9.036e-16
```

```
shapiro.test(log(erupcion$eruptions))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  log(erupcion$eruptions)
```

```
## W = 0.81727, p-value < 2.2e-16
```

```
meanErup <- mean(erupcion$eruptions)
meanWai <- mean(erupcion$waiting)

sdErup <- sd(erupcion$eruptions)
sdWai <- sd(erupcion$waiting)


cor.test(erupcion$eruptions,erupcion$waiting, method = "pearson" )
```

```
##
##  Pearson's product-moment correlation
##
## data:  erupcion$eruptions and erupcion$waiting
## t = 34.089, df = 270, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.8756964 0.9210652
## sample estimates:
##       cor
## 0.9008112
```

```
cor.test(erupcion$eruptions,erupcion$waiting, method= "spearman")
```

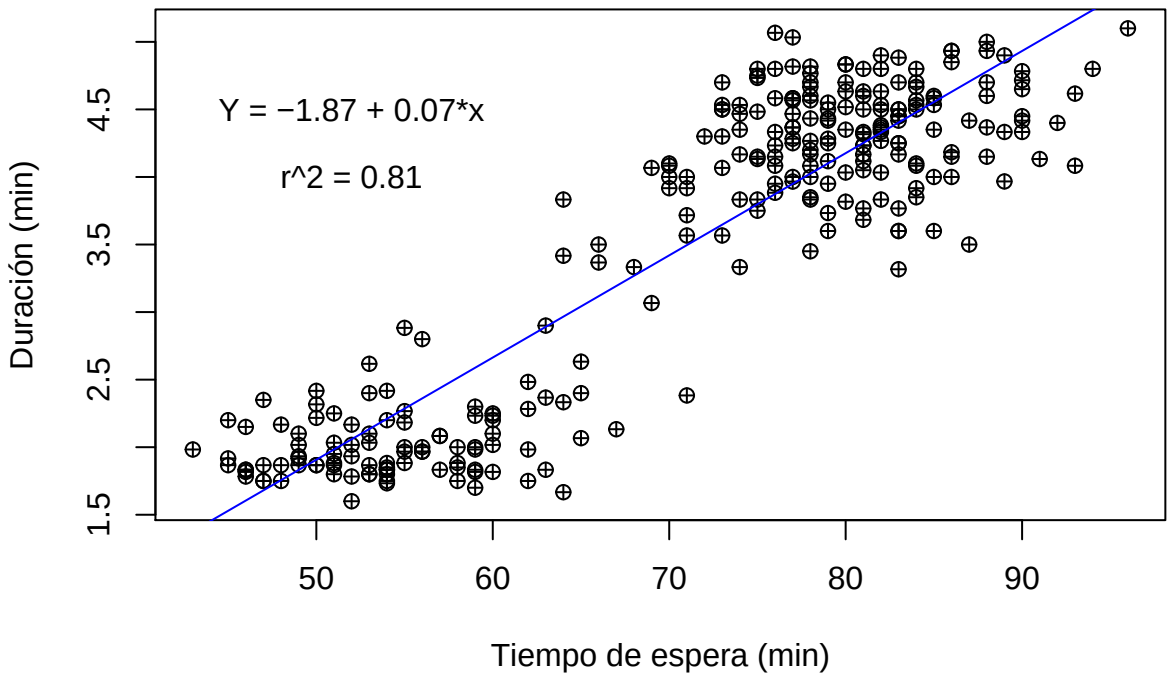
```
## Warning in cor.test.default(erupcion$eruptions, erupcion$waiting, method =
## "spearman"): Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  erupcion$eruptions and erupcion$waiting
## S = 744659, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.7779721
```

```
# Regresion lineal ------------------------------------------------------

#H0 = la predicción no es significativa
#Ha = la predicción es significativa

lm.erup <-lm(erupcion$eruptions ~ erupcion$waiting)
plot(erupcion$waiting , erupcion$eruptions, pch= 10,
     xlab= "Tiempo de espera (min)",
     ylab= "Duración (min)")
abline(lm.erup, col="blue")
text(52,4.5, "Y = -1.87 + 0.07*x")
text(52, 4, "r^2 = 0.81")
```




```
lm.erup
```

```
##
## Call:
## lm(formula = erupcion$eruptions ~ erupcion$waiting)
##
## Coefficients:
##      (Intercept)  erupcion$waiting
##         -1.87402           0.07563
```

```
summary(lm.erup)
```

```
##
## Call:
## lm(formula = erupcion$eruptions ~ erupcion$waiting)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.29917 -0.37689  0.03508  0.34909  1.19329
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -1.874016   0.160143  -11.70   <2e-16 ***
## erupcion$waiting  0.075628   0.002219   34.09   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.4965 on 270 degrees of freedom
## Multiple R-squared:  0.8115, Adjusted R-squared:  0.8108
## F-statistic:  1162 on 1 and 270 DF,  p-value: < 2.2e-16
```

```
length(erupcion$eruptions)
```

```
## [1] 272
```

```
(0.9)^2
```

```
## [1] 0.81
```

```
y.60<- -1.87+0.07*60
y.60
```

```
## [1] 2.33
```

```
espera<- erupcion$waiting
duracion<- erupcion$eruptions

res<-resid(lm.erup)
res
```

```
##           1           2           3           4           5
## -0.50059190 -0.40989320 -0.38945216 -0.53191679 -0.02135959
##           6           7           8           9          10
##  0.59747885 -0.08124343 -0.95435959 -0.03300936 -0.20435959
##          11          12          13          14          15
## -0.37689320 -0.56173164  0.17503605  0.06950243  0.29689631
##          16          17          18          19          20
##  0.10836269 -1.06491679  0.32126836 -0.45863731  0.14940810
##          21          22          23          24          25
## -0.18300936  0.06950243 -0.57496395 -0.27731242  0.81054784
##          26          27          28          29          30
## -0.80310369 -0.31852115  0.20929194 -0.17496395  0.33240810
##          31          32          33          34          35
##  0.65317579  0.51766399  0.24957142 -0.14321985  0.11054784
##          36          37          38          39          40
## -0.04163731  0.11087449  0.65678015 -0.75503294 -0.14949933
##          41          42          43          44          45
##  0.17378015 -0.62940499  0.08826836 -0.76240499  0.88617579
##          46          47          48          49          50
## -1.08610369  0.86682732 -0.03426525  0.30552425 -0.58803294
##          51          52          53          54          55
##  1.00191989 -0.21649933 -0.37689320  0.65678015 -0.47689320
##          56          57          58          59          60
##  0.47989631  0.22143168 -1.29917268  0.61766399  0.06515220
##          61          62          63          64          65
## -0.35503294  0.02126836 -0.00612552  0.47252425 -0.84666089
##          66          67          68          69          70
## -0.68375523  0.14203605  0.67503605 -0.97480063  1.05317579
##          71          72          73          74          75
## -0.29447575 -0.39414910  0.39940810  0.50443168 -0.83191679
##          76          77          78          79          80
##  1.19329194 -0.64666089  0.54203605  0.00929194 -0.80310369
##          81          82          83          84          85
##  0.33491989  0.00552425  0.68005963 -0.40880063  0.42017579
```

```
##           86           87           88           89           90
##  0.151756567  0.076291942  0.340780150  0.410874485 -0.629987537
##           91           92           93           94           95
## -0.463660891 -0.599499329 -0.040381411  0.792036046 -1.057544735
##           96           97           98           99          100
##  0.728803734  0.188268358 -0.048080110 -0.116009359  0.572524254
##          101          102          103          104          105
## -0.331916787 -0.414243433  0.268246537  0.096896306 -0.201847798
##          106          107          108          109          110
##  0.186502433  0.221268358 -0.275637307  0.220012463 -0.568847798
##          111          112          113          114          115
##  0.934919890 -0.288032943  0.043128619  0.316408098 -0.888032943
##          116          117          118          119          120
##  0.381152202  0.409618589  0.045640411 -0.771032943 -0.288615485
##          121          122          123          124          125
##  0.482734745  0.722687578  0.300663994 -0.394149099 -0.181243433
##          126          127          128          129          130
## -0.484847798  0.387758329  0.172524254 -0.018521151 -0.282499329
##          131          132          133          134          135
##  0.337758329 -0.236103694  0.438850901 -0.523871381  0.228130381
##          136          137          138          139          140
##  0.055524254 -0.100009359  0.303012463 -0.101265255 -0.367591902
##          141          142          143          144          145
## -0.018847798 -0.430660891  0.205524254  0.867663994  0.459291942
##          146          147          148          149          150
## -0.605032943  0.456780150  0.185246537 -0.286267017 -0.334265255
##          151          152          153          154          155
##  1.083663994  0.050663994 -0.641800630  0.348152202  0.071431682
##          156          157          158          159          160
##  0.580059630  0.248152202 -1.076383173 -0.334265255 -0.889871381
##          161          162          163          164          165
##  0.670758329 -0.479987537 -0.512404995 -0.191963954  0.382571422
##          166          167          168          169          170
##  0.709291942 -0.523544735  0.218756567 -0.125637307 -0.542383173
##          171          172          173          174          175
##  0.085246537 -0.353777047  0.633663994  0.064315526 -0.084847798
##          176          177          178          179          180
##  0.081152202  0.853175786  0.509618589 -0.554359589  0.444547838
##          181          182          183          184          185
## -0.402521151  0.633663994 -0.153103694 -0.636103694  0.049990641
##          186          187          188          189          190
##  0.408036046 -0.395731642  0.228130381  0.013896306 -0.102521151
##          191          192          193          194          195
##  0.548152202 -0.603777047  0.926291942 -0.378731642  0.016663994
##          196          197          198          199          200
## -0.018847798 -1.205615485  0.416663994  0.266990641  0.642036046
##          201          202          203          204          205
## -0.563660891  0.022524254 -0.875127277 -0.267265255  0.575036046
##          206          207          208          209          210
##  0.178130381  0.417663994 -0.628731642  0.101246537  0.096896306
##          211          212          213          214          215
## -1.112568318  0.523780150  0.035246537  0.034919890  0.450827317
##          216          217          218          219          220
##  0.359291942  0.265734745 -0.435011121 -0.285521151  0.276291942
```

```
##          221          222          223          224          225
## -0.040381411 -0.060475746 -0.459893203  0.684919890 -0.024963954
##          226          227          228          229          230
##  0.016408098  0.058036046  0.242036046  0.497059630  0.449408098
##          231          232          233          234          235
##  0.663059630  0.207106797 -0.446987537  0.309618589 -0.482499329
##          236          237          238          239          240
## -0.326893203 -0.359893203  0.333663994 -0.150591902 -0.633172683
##          241          242          243          244          245
##  0.351919890  0.669502433  0.303012463  0.009455265  0.028640411
##          246          247          248          249          250
## -0.494475746 -0.353777047  0.039524254 -1.060056526  0.627547838
##          251          252          253          254          255
## -0.009893203  0.046896306 -0.079824214  0.853175786 -0.631243433
##          256          257          258          259          260
## -0.359219850  0.421431682  0.046896306 -0.361149099  0.182408098
##          261          262          263          264          265
##  0.742036046  0.054268358 -0.662404995 -0.153103694  0.605014224
##          266          267          268          269          270
## -0.413660891  0.951919890 -0.134847798  0.545130381 -0.515499329
##          271          272
##  0.212130381  0.744547838
```

```
sum(res)
```

```
## [1] 6.973588e-16
```

```
pre <- fitted(lm.erup)
res.2 <- res^2
cuadro<- round(data.frame(espera, duracion,pre,
                    res, res.2), 4)

SSE<- sum((duracion - pre)^2)
SSE
```

```
## [1] 66.56178
```

```
vari<- SSE/length((erupcion$eruptions)-2)
vari
```

```
## [1] 0.2447124
```

```
# pureba hipotesis regresion ----------------------------------------------
an.erup <- anova(lm.erup)
an.erup
```

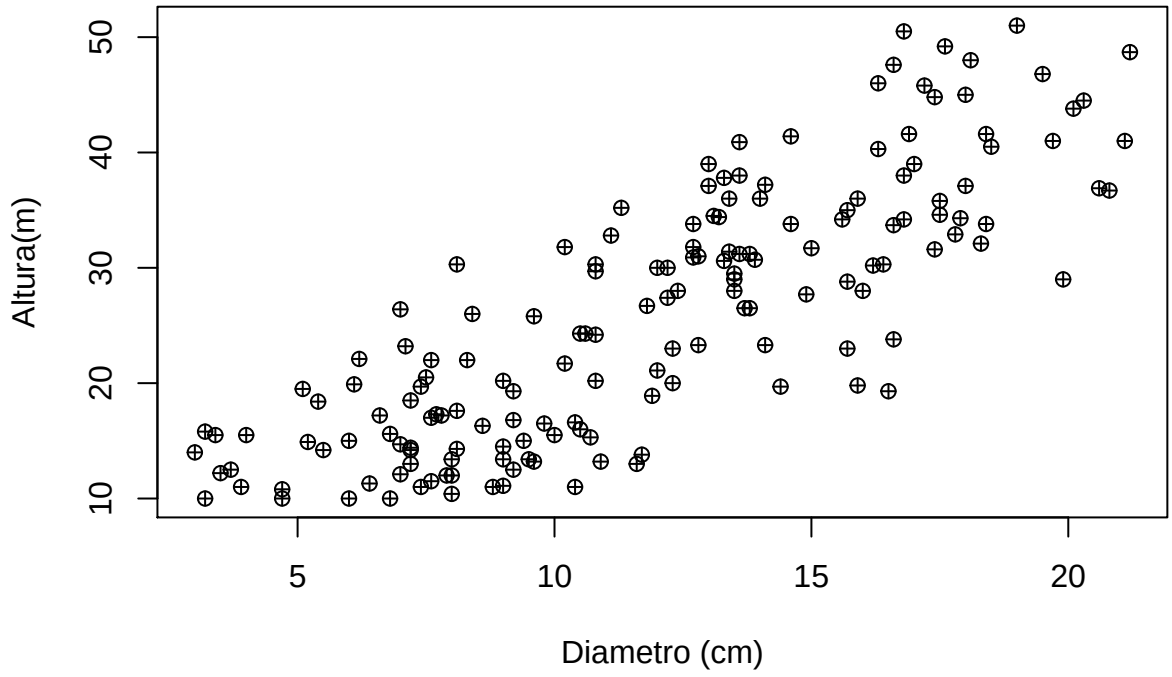
```
## Analysis of Variance Table
##
## Response: erupcion$eruptions
##                   Df  Sum Sq Mean Sq F value    Pr(>F)
## erupcion$waiting   1 286.478 286.478  1162.1 < 2.2e-16 ***
## Residuals        270  66.562   0.247
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#aceptamos la hipotesis alternativa que nos dice que
# el modelo de regresion es significativo
```

```r
# Ejercicios ebanos ------------------------------------------------------

ebanos <-read.csv("C:/MCF202-2019/Datos/ebanos.csv", header= T)

plot(ebanos$altura , ebanos$diametro, pch= 10,
     xlab= "Diametro (cm)",
     ylab= "Altura(m)")
```




```r
library(pastecs)

stat.desc(ebanos$altura,basic= FALSE, norm= TRUE)
```

```
##       median         mean      SE.mean CI.mean.0.95          var
## 12.000000000 11.885365854  0.357428221  0.705786566 20.951809068
##      std.dev     coef.var     skewness     skew.2SE     kurtosis
##  4.577314613  0.385121894  0.053516314  0.141163547 -0.932366816
##     kurt.2SE   normtest.W   normtest.p
## -1.236840496  0.977187792  0.008242431
```

```r
shapiro.test(ebanos$altura)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  ebanos$altura
## W = 0.97719, p-value = 0.008242
```

```r
shapiro.test(ebanos$diametro)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  ebanos$diametro
## W = 0.94921, p-value = 1.215e-05
```

```r
shapiro.test(sin(ebanos$altura))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  sin(ebanos$altura)
## W = 0.88458, p-value = 5.622e-10
```

```r
cor.test(ebanos$altura,ebanos$diametro, method = "pearson" )
```

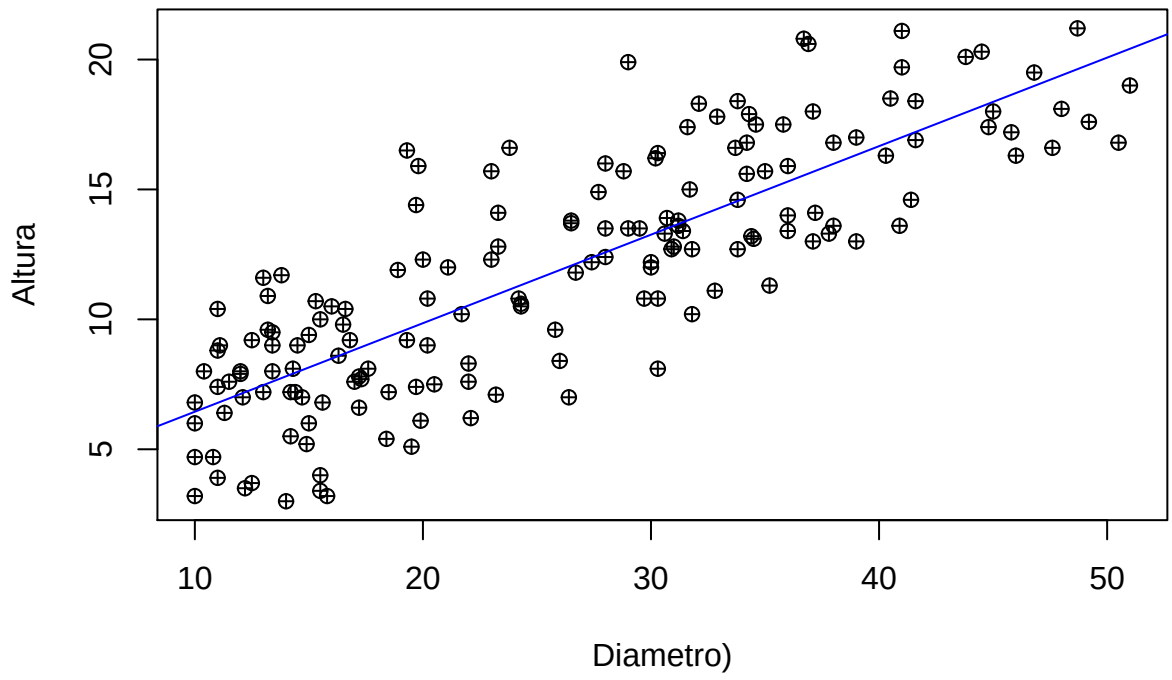
```
##
##  Pearson's product-moment correlation
##
## data:  ebanos$altura and ebanos$diametro
## t = 18.354, df = 162, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.7648115 0.8659458
## sample estimates:
##       cor
## 0.8217467
```

```r
cor.test(ebanos$altura,ebanos$diametro, method= "spearman")
```

```
## Warning in cor.test.default(ebanos$altura, ebanos$diametro, method =
## "spearman"): Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  ebanos$altura and ebanos$diametro
## S = 127826, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.8261184
```

```r
lm.eba <-lm(ebanos$altura ~ ebanos$diametro)
plot(ebanos$diametro , ebanos$altura, pch= 10,
     xlab= "Diametro)",
     ylab= "Altura")
abline(lm.eba, col="blue")
```




```
lm.eba
```

```
##
## Call:
## lm(formula = ebanos$altura ~ ebanos$diametro)
##
## Coefficients:
##     (Intercept)  ebanos$diametro
##          3.0380           0.3407
```

```
summary(lm.eba)
```

```
##
## Call:
## lm(formula = ebanos$altura ~ ebanos$diametro)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.2612 -2.0484 -0.1683  1.8420  6.9817
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.03796    0.52355   5.803 3.34e-08 ***
## ebanos$diametro  0.34070    0.01856  18.354  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 2.616 on 162 degrees of freedom
## Multiple R-squared:  0.6753, Adjusted R-squared:  0.6733
## F-statistic: 336.9 on 1 and 162 DF,  p-value: < 2.2e-16
```

```
(0.82)^2
```

```
## [1] 0.6724
```