clase_4.R

52618

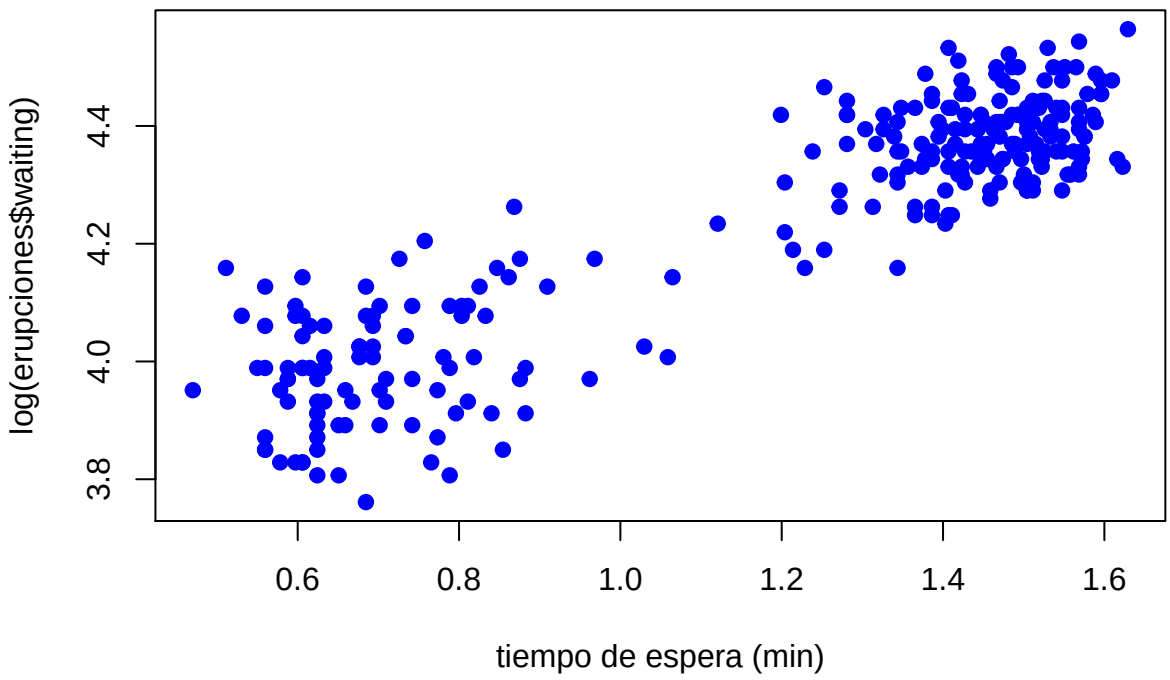
2019-08-09

```
#Cipriano Guerrero Cabrera
#09/08/
#clase_4

erupciones <- read.csv("C:/MCF 202-2019/MCF202/Datos/erupciones.csv")
head(erupciones)
```

```
##   eruptions waiting
## 1     3.600      79
## 2     1.800      54
## 3     3.333      74
## 4     2.283      62
## 5     4.533      85
## 6     2.883      55
```

```
plot(log(erupciones$eruptions), log(erupciones$waiting), pch=19, col="blue",
     xlab= "tiempo de espera (min)")
```




```
library(pastecs)
stat.desc(erupciones$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(erupciones$eruptions)
```

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

```
shapiro.test(erupciones$waiting)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  erupciones$waiting
## W = 0.92215, p-value = 1.015e-10
```

```
cor.test(erupciones$eruptions, erupciones$waiting)
```

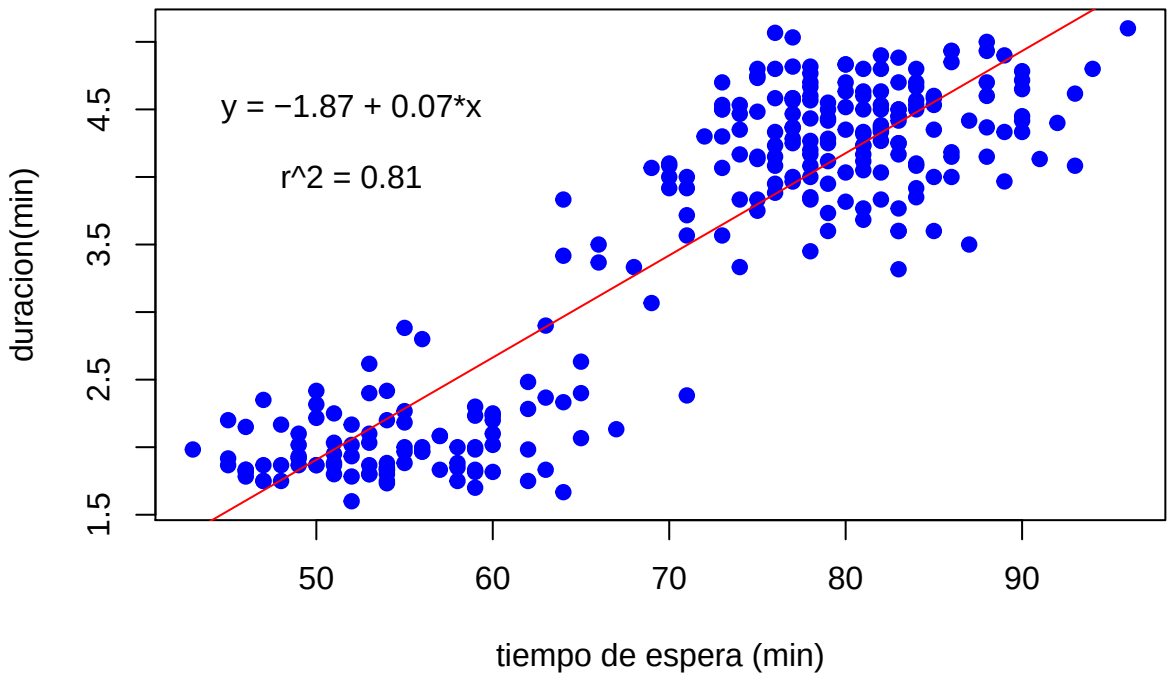
```
##
##  Pearson's product-moment correlation
##
## data:  erupciones$eruptions and erupciones$waiting
## t = 34.089, df = 270, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.8756964 0.9210652
## sample estimates:
##       cor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

```
# regrecion lineal ------------------------------------------------------

# h0. no es sinifivaticva la prediccion de las erupciones
# h1. es significativa kla predioccion de las erupciones.
#el comando (lm) para realizar la regrecion lineal

lm.erup<-lm(erupciones$eruptions~erupciones$waiting)

plot(erupciones$waiting, erupciones$eruptions, pch=19, col="blue",
     xlab= "tiempo de espera (min)",
     ylab= "duracion(min)")
#en la grafica se pone la la variable dependiente  "x" y de la independiente "y"
abline(lm.erup, col="red")
text(52, 4.5, "y = -1.87 + 0.07*x")
#este comando es para poner liea en la grafica plot
text(52, 4, "r^2 = 0.81")
```




```
lm.erup
```

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

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

```
##
## Call:
## lm(formula = erupciones$eruptions ~ erupciones$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 ***
## erupciones$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
```

```
#los valores residuales es la diferencia entre el valor observado y el valor predicho
length(erupciones$eruptions)
```

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

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

y.60
```

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

```
# datos de regrecipon -----------------------------------------------------
espera<-erupciones$waiting
duracion<-erupciones$eruptions
res<-resid(lm.erup)
pre<- fitted(lm.erup)
res.2<-res^2

cuadro<- round(data.frame(espera, duracion, pre, res,res.2)^2)

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

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

```
vari<- sse/(length(erupciones$waiting)-2)
vari
```

```
## [1] 0.2465251
```

```
#esta varianza es del modelo que estamos estimando


# prueba de hipotesis de la regrecion -------------------------------------

an.erup<-anova(lm.erup)
an.erup
```

```
## Analysis of Variance Table
##
## Response: erupciones$eruptions
##                     Df  Sum Sq Mean Sq F value    Pr(>F)
## erupciones$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
```

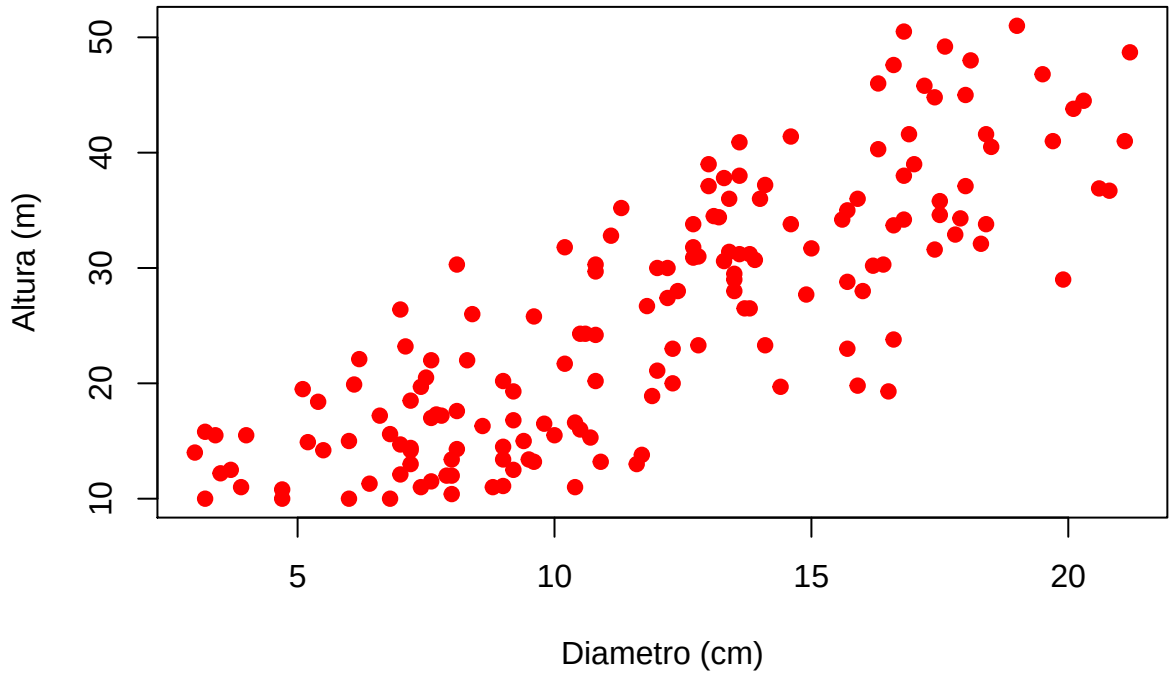
```
#se hacepta la hipotesis alternativa en el modelo lineal esto quiere decir que
#las erupciones son


# ejercicio 2 ebanos ------------------------------------------------------

ebano <- read.csv("C:/MCF 202-2019/MCF202/Datos/ebanos.csv")
head(ebano)
```

```
##   diametro altura
## 1     31.2   13.8
## 2     35.2   11.3
## 3     15.5    4.0
## 4     30.6   13.3
## 5     32.9   17.8
## 6     17.2    7.8
```

```
plot(ebano$altura, ebano$diametro, pch=19, col="red",
     xlab = "Diametro (cm)",
     ylab = "Altura (m)")
```




```
library(pastecs)
stat.desc(ebano$diametro, basic = FALSE, norm = TRUE)
```

```
##        median          mean       SE.mean  CI.mean.0.95           var
##  2.590000e+01  2.596829e+01  8.620934e-01  1.702311e+00  1.218856e+02
##       std.dev      coef.var      skewness      skew.2SE      kurtosis
##  1.104018e+01  4.251408e-01  3.253519e-01  8.582025e-01 -9.701612e-01
##      kurt.2SE    normtest.W    normtest.p
## -1.286977e+00  9.492059e-01  1.215075e-05
```

```
stat.desc(ebano$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
```

```
shapiro.test(ebano$diametro)
```

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

```
shapiro.test(ebano$altura)
```

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