

**UNIVERSIDAD DE EL SALVADOR
FACULTAD MULTIDISCIPLINARIA DE OCCIDENTE
DEPARTAMENTO DE MATEMÁTICA**



Licenciatura en Estadística

Control Estadístico del Paquete R

**”UNIDAD DOS”
PRÁCTICA 08**

**Alumna:
Erika Beatrí Guillén Pineda**

**Fecha de elaboración
Santa Ana - 27 de noviembre de 2015**

1. ANÁLISIS ESTADÍSTICO DE LOS DATOS.

1. Visualiza el directorio por defecto y activa su directorio de trabajo

```
getwd()

## [1] "C:/Users/User/Documents/TODAS_PRACTICAS"

setwd("C:/Users/User/Documents/TODAS_PRACTICAS")
```

2. Crea un nuevo Script y llámale "Script08-DatosContinuos"
3. Crea el vector que contendrá los datos.

```
Notas <- c(4.47, 4.47, 3.48, 5.0, 3.42, 3.78, 3.1, 3.57, 4.2, 4.5, 3.6,
          3.75, 4.5, 2.85, 3.7, 4.2, 3.2, 4.05, 4.9, 5.1, 5.3,
          4.16, 4.56, 3.54, 3.5, 5.2, 4.71, 3.7, 4.78, 4.14, 4.14, 4.8,
          4.1, 3.83, 3.6, 2.98, 4.32, 5.1, 4.3, 3.9, 3.96, 3.54, 4.8,
          4.3, 3.39, 4.47, 3.19, 3.75, 3.1, 4.7, 3.69, 3.3, 2.85,
          5.25, 4.68, 4.04, 4.44, 5.43, 3.04, 2.95); Notas

## [1] 4.47 4.47 3.48 5.00 3.42 3.78 3.10 3.57 4.20 4.50 3.60 3.75 4.50 2.85
## [15] 3.70 4.20 3.20 4.05 4.90 5.10 5.30 4.16 4.56 3.54 3.50 5.20 4.71 3.70
## [29] 4.78 4.14 4.14 4.80 4.10 3.83 3.60 2.98 4.32 5.10 4.30 3.90 3.96 3.54
## [43] 4.80 4.30 3.39 4.47 3.19 3.75 3.10 4.70 3.69 3.30 2.85 5.25 4.68 4.04
## [57] 4.44 5.43 3.04 2.95

data.entry(Notas)
Notas

## [1] 4.47 4.47 3.48 5.00 3.42 3.78 3.10 3.57 4.20 4.50 3.60 3.75 4.50 2.85
## [15] 3.70 4.20 3.20 4.05 4.90 5.10 5.30 4.16 4.56 3.54 3.50 5.20 4.71 3.70
## [29] 4.78 4.14 4.14 4.80 4.10 3.83 3.60 2.98 4.32 5.10 4.30 3.90 3.96 3.54
## [43] 4.80 4.30 3.39 4.47 3.19 3.75 3.10 4.70 3.69 3.30 2.85 5.25 4.68 4.04
## [57] 4.44 5.43 3.04 2.95

length(Notas)

## [1] 60
```

4. Guarda el vector de datos en un archivo.

```
write(Notas, "Notas.txt")
```

5. Limpia el área de trabajo (Workspace)

```
ls()

## [1] "Notas"

rm(list=ls(all=TRUE))
ls()

## character(0)
```

6. Lee o recupera el vector de datos desde el archivo de texto.

```
X <- scan("Notas.txt", what = double(0), na.strings = "NA", flush=FALSE)
ls()

## [1] "X"

# Si el vector contiene valores reales se ocupa:

what = double(0)
```

7. Crea la tabla de frecuencias.

```
# Define el numero k de los intervalos o clases.

# Usa el Metodo de Herbert A. Sturges para determinar dicho numero.

n <- length(X); n

## [1] 60

k <- 1+3.322*logb(n, 10); k

## [1] 6.907018

k <- round(k); k

## [1] 7
```

```
# Calcula el ancho o amplitud a de cada intervalo a=rango/k
```

```
rango <- max(X)-min(X); rango
```

```
## [1] 2.58
```

```
a=rango/k; a
```

```
## [1] 0.3685714
```

```
a <- round(a, 3); a
```

```
## [1] 0.369
```

```
# Define los limites y puntos medios de cada uno de los k intervalos
```

```
limites <- c(from = min(X)-0.01/2, to = max(X)+0.01/2, by=a); limites
```

```
## from to by
```

```
## 2.845 5.435 0.369
```

```
options(digits=4)
```

```
ci <- cbind(1:k); ci
```

```
## [,1]
```

```
## [1,] 1
```

```
## [2,] 2
```

```
## [3,] 3
```

```
## [4,] 4
```

```
## [5,] 5
```

```
## [6,] 6
```

```
## [7,] 7
```

```
for(i in 2:length(limites)) ci[i-1, 1] <- (limites[i]+ limites[i-1])/2
```

```
ci
```

```
## [,1]
```

```
## [1,] 4.140
```

```
## [2,] 2.902
```

```
## [3,] 3.000
```

```
## [4,] 4.000
## [5,] 5.000
## [6,] 6.000
## [7,] 7.000
```

```
# Encuentra las frecuencias absolutas fi para cada intervalo.
```

```
options(digits=2)
fi <- cbind(table(cut(X, breaks = limites, labels=NULL, include.lowest=FALSE, right=FALSE),
```

```
##           [,1]
## [0.369,2.845)    0
## [2.845,5.435)   60
```

```
# breakses un vector o secuencia de cortes 1:6, o el numero de clases.
```

```
# labelsindica que no hay nombres para los intervalos o clases, por defecto
# las etiquetas tienen la notacion (a, b].
```

```
# include.lowestindica que si un X[i] es igual al corte inferior (0 superior,
# para right=FALSE) el valor debe ser incluido.
```

```
# rightindica que si el intervalo debe ser cerrado a la derecha y abierto a la
# izquierda, o viceversa.
```

```
# dig.labes un entero el cual es usado cuando las etiquetas no son dadas,
# determina el numero de digitos usado en el formato de numeros de cortes.
```

```
# Encuentra las frecuencias relativas o proporciones fri.
```

```
options(digits=4)
fri <- fi/n; fri
```

```
##           [,1]
## [0.369,2.845)    0
## [2.845,5.435)    1
```

```
# Encuentra las frecuencias acumuladas ascendentes Fi
```

```
options(digits=2)
Fi <- cumsum(fi); Fi
```

```
## [1] 0 60
```

```
# Encuentra las frecuencias relativas acumuladas Fri
```

```
options(digits=4)
Fri <- Fi/n; Fri
```

```
## [1] 0 1
```

```
# Completa la tabla de frecuencias.
```

```
options(digits=2)
tablaFrec <- data.frame(ci=ci, fi=fi, fri=fri, Fi=Fi, Fri=Fri)
```

```
## Error in data.frame(ci = ci, fi = fi, fri = fri, Fi = Fi, Fri = Fri): arguments
imply differing number of rows: 7, 2
```

```
names(tablaFrec) <- c("ci", "fi", "fri", "Fi", "Fri")
```

```
## Error in names(tablaFrec) <- c("ci", "fi", "fri", "Fi", "Fri"): objeto 'tablaFrec'
no encontrado
```

```
tablaFrec
```

```
## Error in eval(expr, envir, enclos): objeto 'tablaFrec' no encontrado
```

```
# Nuevamente puede usar el comando xtable para importar a código LATEX.
```

8. Crea el histograma de frecuencias

```
h <- hist(X, breaks=c(limite[1]-a, limite, limite[k+1]+a),
          freq = TRUE, probability = FALSE, include.lowest = FALSE,
          right = TRUE, main = "Histograma de frecuencias",
          col="lightyellow", lty=1, border="purple", xlab="Notas de aspirantes",
          ylab="Frecuencia (fi)", axes=TRUE, labels=FALSE)
```

```
## Warning in breaks + fuzz: longitud de objeto mayor no es múltiplo de la longitud
de uno menor
```

```
## Error in hist.default(X, breaks = c(limite[1] - a, limite, limite[k + :
some 'x' not counted; maybe 'breaks' do not span range of 'x'
```

```
text(h$mids, h$density, h$counts, adj=c(0.5, -0.5), col="red")
```

```
## Error in text(h$mids, h$density, h$counts, adj = c(0.5, -0.5), col = "red"):
objeto 'h' no encontrado
```

```
rug(jitter(X))
```

```
## Warning in rug(jitter(X)): some values will be clipped
## Error in axis(side = side, at = at, labels = labels, ...): plot.new has not
## been called yet

# adiciona marcas de los datos
# h es un objeto del tipo lista que contiene atributos del histograma

is.list(h); h

## Error in eval(expr, envir, enclos): objeto 'h' no encontrado
## Error in eval(expr, envir, enclos): objeto 'h' no encontrado
```

9. Aproxima al histograma la función de densidad normal

```
h <- hist(X, breaks=c(limites[1]-a, limites, limites[k+1]+a),
  freq = FALSE, probability = TRUE, include.lowest = FALSE,
  right = TRUE, main="Aproximacion a una Normal\n",
  col="lightyellow",lty=1,border="purple",
  xlab="Notas de aspirantes\n", ylab="Frecuencia relativa (fri)",
  axes=TRUE, labels=FALSE)

## Warning in breaks + fuzz: longitud de objeto mayor no es múltiplo de la longitud
## de uno menor
## Error in hist.default(X, breaks = c(limites[1] - a, limites, limites[k + :
## some 'x' not counted; maybe 'breaks' do not span range of 'x'

text(h$mids, h$density, h$counts, adj=c(0.5, 0.2), col="red")

## Error in text(h$mids, h$density, h$counts, adj = c(0.5, 0.2), col = "red"):
## objeto 'h' no encontrado

rug(jitter(X)) # adiciona marcas de los datos

## Warning in rug(jitter(X)): some values will be clipped
## Error in axis(side = side, at = at, labels = labels, ...): plot.new has not
## been called yet

curve(dnorm(x, mean=mean(X), sd=sd(X)), col = 2, lty = 2,lwd = 2, add = TRUE)

## Error in plot.xy(xy.coords(x, y), type = type, ...): plot.new has not been
## called yet
```

10. Crea el polígono de frecuencias

```

h<- hist(X, breaks=c(limites[1]+a, limites, limites[k+1]+a), freq =TRUE,
         probability=FALSE , include.lowest=FALSE , right=TRUE,
main = "Polígono de frecuencias",col="lightyellow",
lty=1, border = "purple" , xlab=" Notas de aspirantes" , ylab="Frecuencia (fi)",
axes=TRUE, labels=FALSE)

## Warning in breaks + fuzz: longitud de objeto mayor no es múltiplo de la longitud
de uno menor

## Error in hist.default(X, breaks = c(limites[1] + a, limites, limites[k + :
some 'x' not counted; maybe 'breaks' do not span range of 'x'

text(h$mids, h$density, h$counts, adj=c(0.5, -0.5), col="red")

## Error in text(h$mids, h$density, h$counts, adj = c(0.5, -0.5), col = "red"):
objeto 'h' no encontrado

rug(jitter(X)) # adiciona marcas de los datos

## Warning in rug(jitter(X)): some values will be clipped

## Error in axis(side = side, at = at, labels = labels, ...): plot.new has not
been called yet

vCi <- c(h$mids[1]-a, h$mids, h$mids[k+1]+a); vCi

## Error in eval(expr, envir, enclos): objeto 'h' no encontrado
## Error in eval(expr, envir, enclos): objeto 'vCi' no encontrado

vfi <- c(0, h$counts, 0); vfi

## Error in eval(expr, envir, enclos): objeto 'h' no encontrado
## Error in eval(expr, envir, enclos): objeto 'vfi' no encontrado

lines(vCi, vfi, col="blue", type="l")

## Error in lines(vCi, vfi, col = "blue", type = "l"): objeto 'vCi' no encontrado

```

11. Crea la Ojiva ascendente o polígono frecuencias acumuladas ascendentes

```

Fia <- c(0, Fi); Fia

## [1] 0 0 60

plot(limites, Fia, type = "p", pch=1, col = "blue", main="Ojiva ascendente",

```



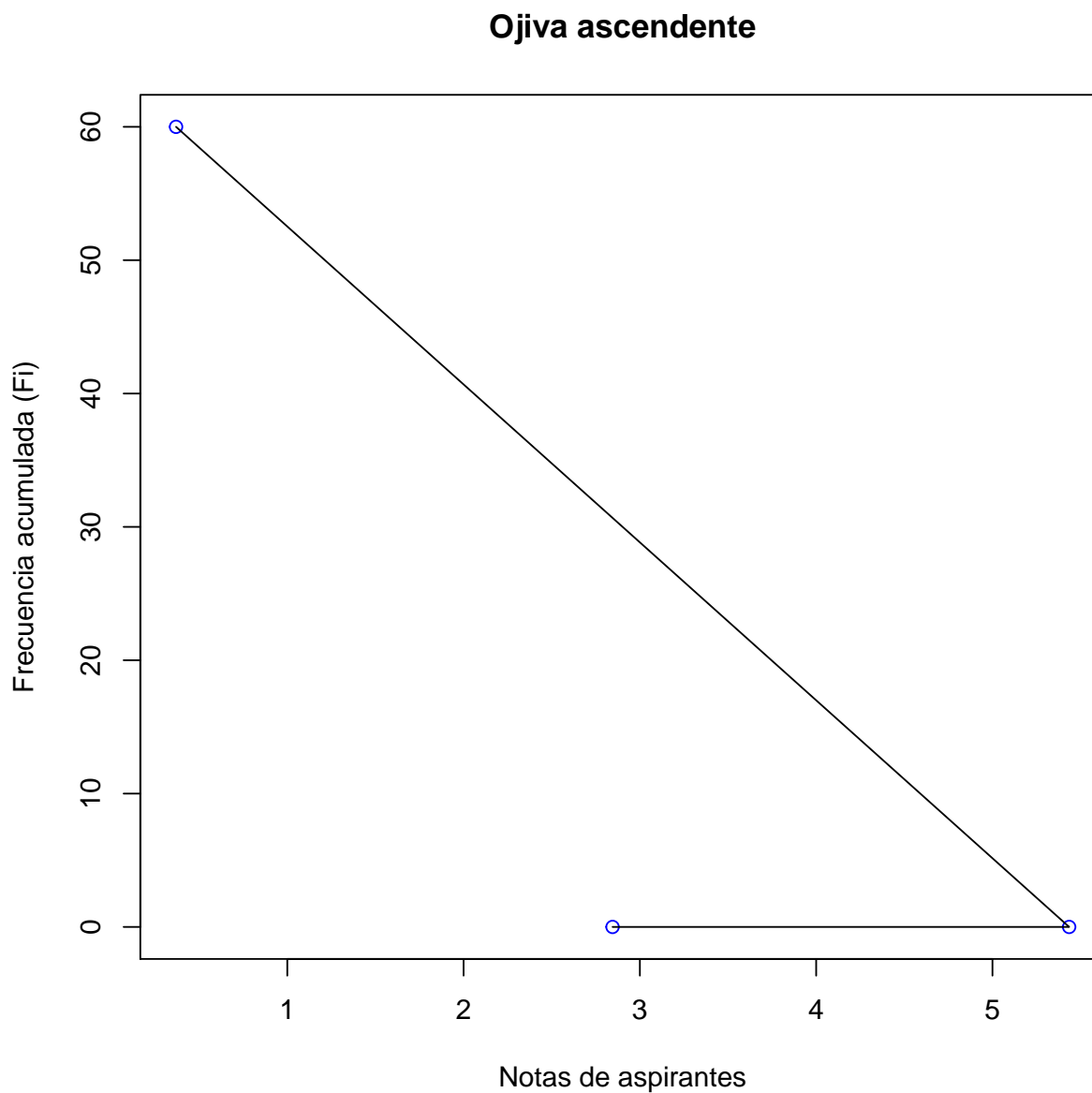
```

xlab="Notas de aspirantes",ylab="Frecuencia acumulada (Fi)")
text(limites, h$density, Fia, adj=c(0.5, -0.5), col="red")

## Error in text.default(limites, h$density, Fia, adj = c(0.5, -0.5), col = "red"):
objeto 'h' no encontrado

lines(limites, Fia, col="black", type="l")

```



12. Calcula los principales estadístico descriptivos de la variable

```

# Calcula la moda, ya que el R no proporciona una funcion para eso.
options(digits=4)
for(i in 1:k)

```

```
    if (fi[i] == max(fi)) break()
if(i > 1) moda <- limites[i]+((fi[i]-fi[i-1])/((fi[i]-fi[i-1])+
(fi[i]-fi[i+1])))*a else moda <- limites[i]+(fi[i]/(fi[i]+
(fi[i]-fi[i+1])))*a
moda

## to
## NA
```

```
# Calcula los cuartiles: Q1, Q2, Q3

Q <- 1:3
for(v in 1:3) for(i in 1:k) if (Fi[i] > (v*25*n)/100)
{
  Q[v] <- limites[i]+(((25*v*n)/100)-Fi[i-1])/fi[i])*a
  break
}
Q

## [1] 5.527 5.619 5.712
```