

# LABORATORIO\_4\_ANGELICA\_TORRES.R

acile

2023-02-27

```
#Angelica Torres García
#22/02/2023
#2173388
```

```
# LABORATORIO 4. RESUMIR DATOS GRAFICAMENTE
```

```
# IMPORTAR DATOS CSV -----
```

```
library(readr)

esp.url <- paste0("https://raw.githubusercontent.com/mgtagle/",
                  "PrincipiosEstadistica2021/main/cuadro1.csv")
inventario <- read.csv(esp.url)

head(inventario)
```

##	Arbol	Fecha	Especie	Posicion	Vecinos	Diametros	Altura
## 1	1	12	F	C	4	15.3	14.78
## 2	2	12	F	D	3	17.8	17.07
## 3	3	9	C	D	5	18.2	18.28
## 4	4	9	H	S	4	9.7	8.79
## 5	5	7	H	I	6	10.8	10.18
## 6	6	10	C	I	3	14.1	14.90

```
#funciones para inspeccionar sus propiedades y funciones y estructura básica:
```

```
#str(inventario): mostrar la estructura general de los datos
#dim(inventario): dimensiones (i.e. tamaño) del conjunto de datos
#head(inventario, n = 5): muestra las primeras n filas
#tail(inventario, n = 5): muestra las últimas n filas
#names(inventario): nombre de las columnas
#colnames(inventario): igual names(inventario)
#summary(inventario): resumen estadístico de las variables presentes en inventario

# mostrar la estructura general de los datos

str(inventario)
```

```
## 'data.frame': 50 obs. of 7 variables:
## $ Arbol : int 1 2 3 4 5 6 7 8 9 10 ...
## $ Fecha : int 12 12 9 9 7 10 10 12 16 14 ...
## $ Especie : chr "F" "F" "C" "H" ...
## $ Posicion : chr "C" "D" "D" "S" ...
## $ Vecinos : int 4 3 5 4 6 3 2 2 4 5 ...
## $ Diametros: num 15.3 17.8 18.2 9.7 10.8 14.1 17.1 20.6 18.2 16.1 ...
## $ Altura : num 14.78 17.07 18.28 8.79 10.18 ...
```

```
# dimensiones (num filas y columnas)
```

```
dim(inventario)
```

```
## [1] 50 7
```

```
# nombre de las primeras cinco columnas
```

```
names(inventario[1:5])
```

```
## [1] "Arbol" "Fecha" "Especie" "Posicion" "Vecinos"
```

```
# Resumen estadístico básico de las columnas 3 a 5 columnas
```

```
summary(inventario[,3:5])
```

```
## Especie Posicion Vecinos
## Length:50 Length:50 Min. :0.00
## Class :character Class :character 1st Qu.:2.25
## Mode :character Mode :character Median :3.00
## Mean :3.34
## 3rd Qu.:4.00
## Max. :6.00
```

```
is.factor(inventario$Posicion)
```

```
## [1] FALSE
```

```
inventario$Posicion <- factor(inventario$Posicion)
is.factor(inventario$Posicion)
```

```
## [1] TRUE
```

```
summary(inventario[,3:5])
```

```
##   Especie      Posicion  Vecinos
## Length:50      C:14      Min.    :0.00
## Class :character D: 9      1st Qu.:2.25
## Mode  :character I:19      Median  :3.00
##                      S: 8      Mean    :3.34
##                      3rd Qu.:4.00
##                      Max.    :6.00
```

```
# TABLAS DE FRECUENCIA -----
```

*#Una tabla de frecuencias muestra los recuentos de cada categoría. En R, tenemos la función table () para obtener este tipo de tablas.*

```
freq_position <- table(inventario$Posicion)
freq_position
```

```
##
## C D I S
## 14 9 19 8
```

*#frecuencias relativas.*

```
prop_position <- freq_position / sum(freq_position)
prop_position
```

```
##
## C D I S
## 0.28 0.18 0.38 0.16
```

*#Si desea expresar las proporciones como porcentajes, multiplique prop\_position por 100:*

```
perc_position = 100 * prop_position
perc_position
```

```
##
## C D I S
## 28 18 38 16
```

```
# GRAFICAS BARPLOT Y PIE -----
```

*#Hay dos gráficos más comunes que se utilizan para visualizar frecuencias:*

*#Gráficas de barras (barplot)*

*#Gráficas de pastel (pie)*

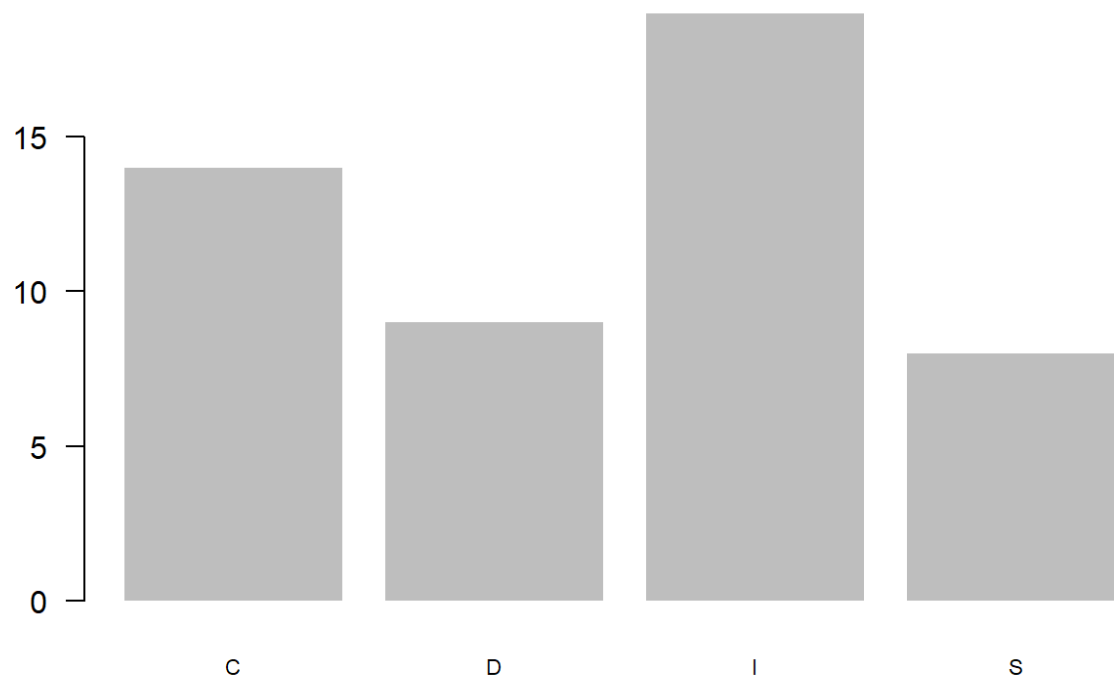
*#El uso de barplot () incluye los argumentos las,border y cex.names:*

*#las = 1: muestra las frecuencias perpendiculares al eje-y.*

*#border = NA: elimina el borde negro alrededor de las barras.*

*#cex.names = 0.7: reduce los tamaños de las etiquetas de categoría (para que todas quepan en el gráfico).*

```
barplot(freq_position, las = 1, border = NA, cex.names = 0.7)
```

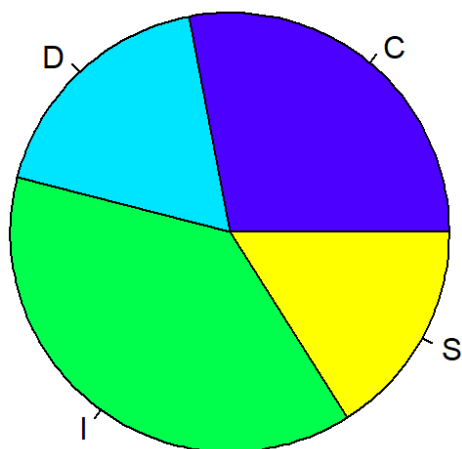


*#Gráfico circular o pie.*

*# topo.colors es una paleta de colores pre establecidas en R y*

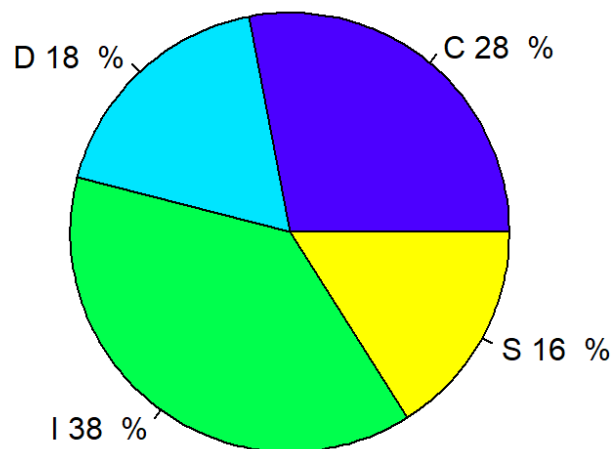
*# el paréntesis indica el # de colores a usar*

```
pie(freq_position, col=topo.colors(4))
```



*# Si se desea mostrar las frecuencias, se puede hacer algo como esto:*

```
pie(freq_position, col = topo.colors(4),  
    labels = paste(levels(inventario$Posicion), round(perc_position,)," %"))
```



```
# AUTOESTUDIO -----
```

```
#Completar una tabla de frecuencia y su representación gráfica (barplot y pie) para la variable Es  
pecie del conjunto de datos inventario
```

```
#tablas de frecuencia
```

```
freq_position <- table(inventario$Especie)  
freq_position
```

```
##  
## C F H  
## 22 14 14
```

```
prop_position <- freq_position / sum(freq_position)  
prop_position
```

```
##  
## C F H  
## 0.44 0.28 0.28
```

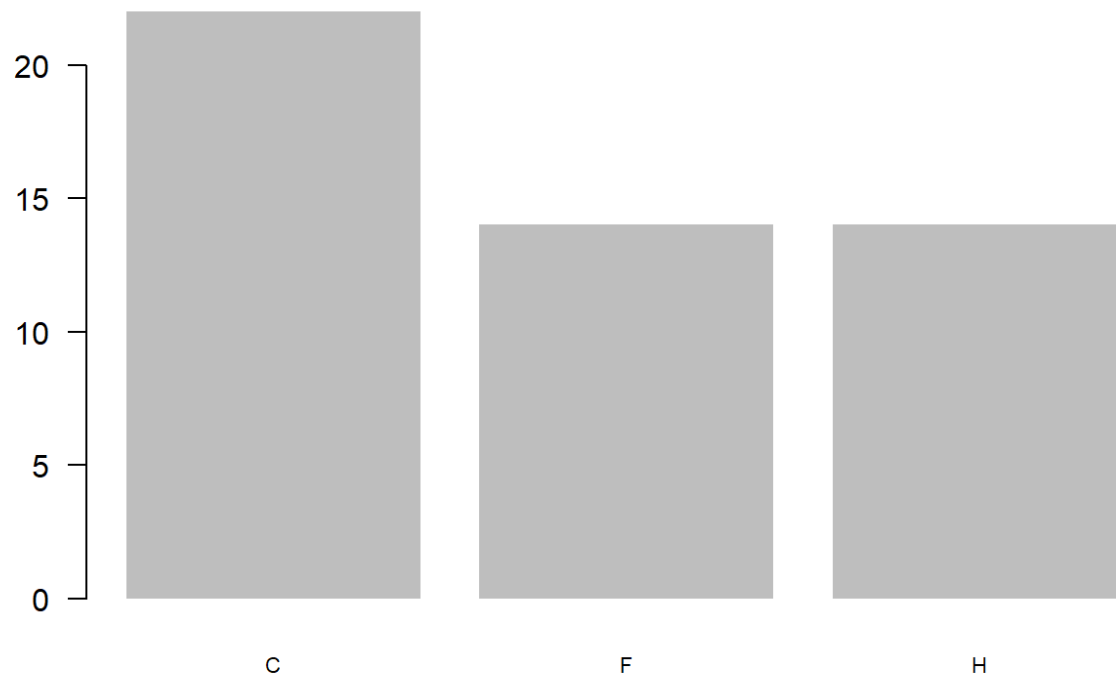
```
#proporciones como porcentajes, multiplique prop_position por 100:
```

```
perc_position = 100 * prop_position  
perc_position
```

```
##  
## C F H  
## 44 28 28
```

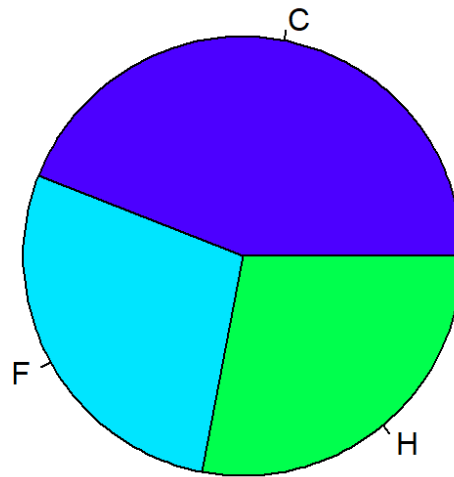
```
#Gráficas barplot y pie
```

```
barplot(freq_position, las = 1, border = NA, cex.names = 0.7)
```



```
#Gráfico circular o pie.
```

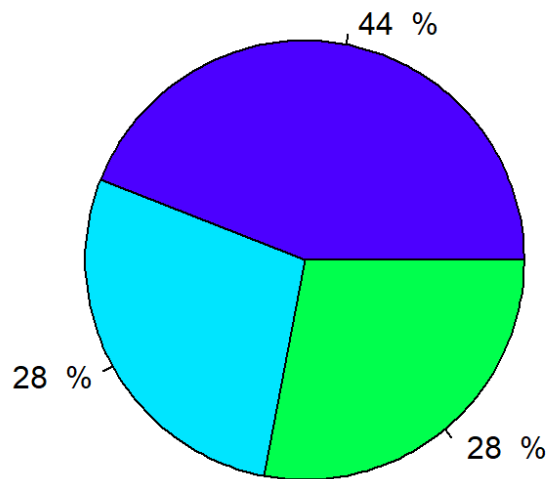
```
pie(freq_position, col=topo.colors(4))
```



```
#mostrar frecuencias
```

```
pie(freq_position, col = topo.colors(4), labels = paste(levels(inventario$Especie), round(perc_pos  
ition, 2), " %"))
```





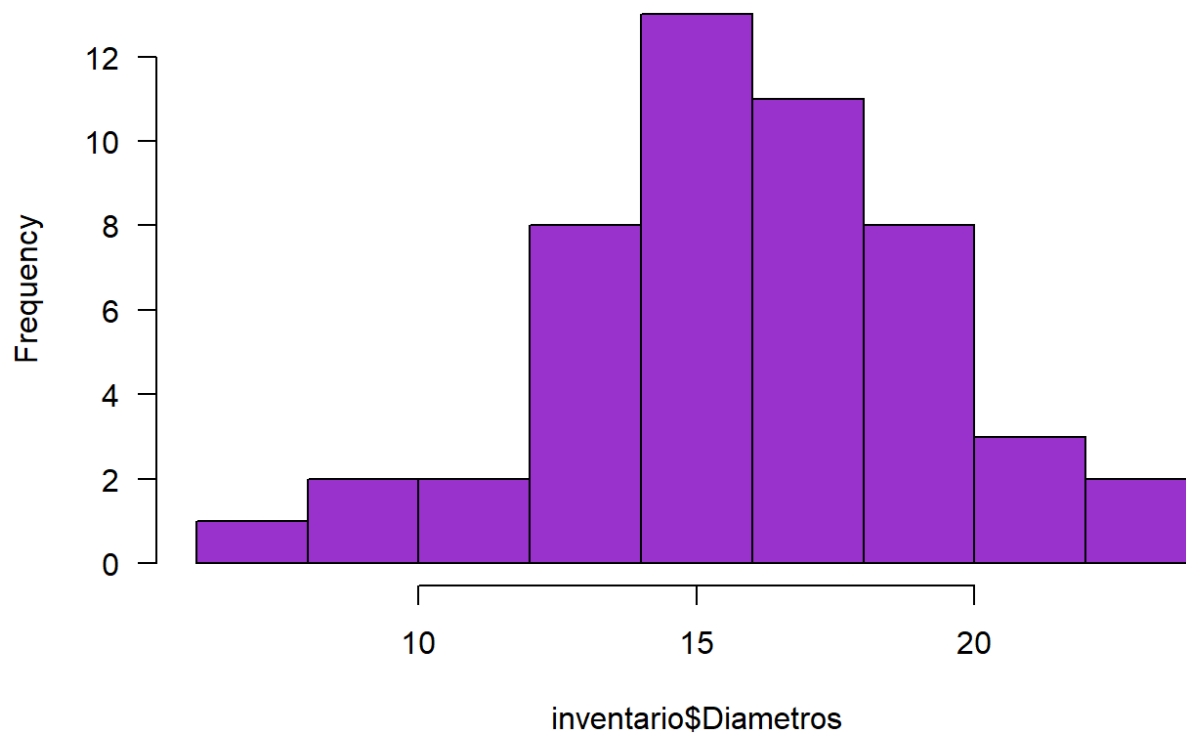
```
# REPRESENTACIÓN DE VARIABLES CUANTITATIVAS -----
```

```
#Histogramas
```

```
#Vamos a aplicar la función hist para la variable Diametros del conjunto inventario y guardar la salida en un objeto llamada diam_hist.
```

```
diam_hist <- hist(inventario$Diametros, las = 1, col = '#9932CC')
```

## Histogram of inventario\$Diametros



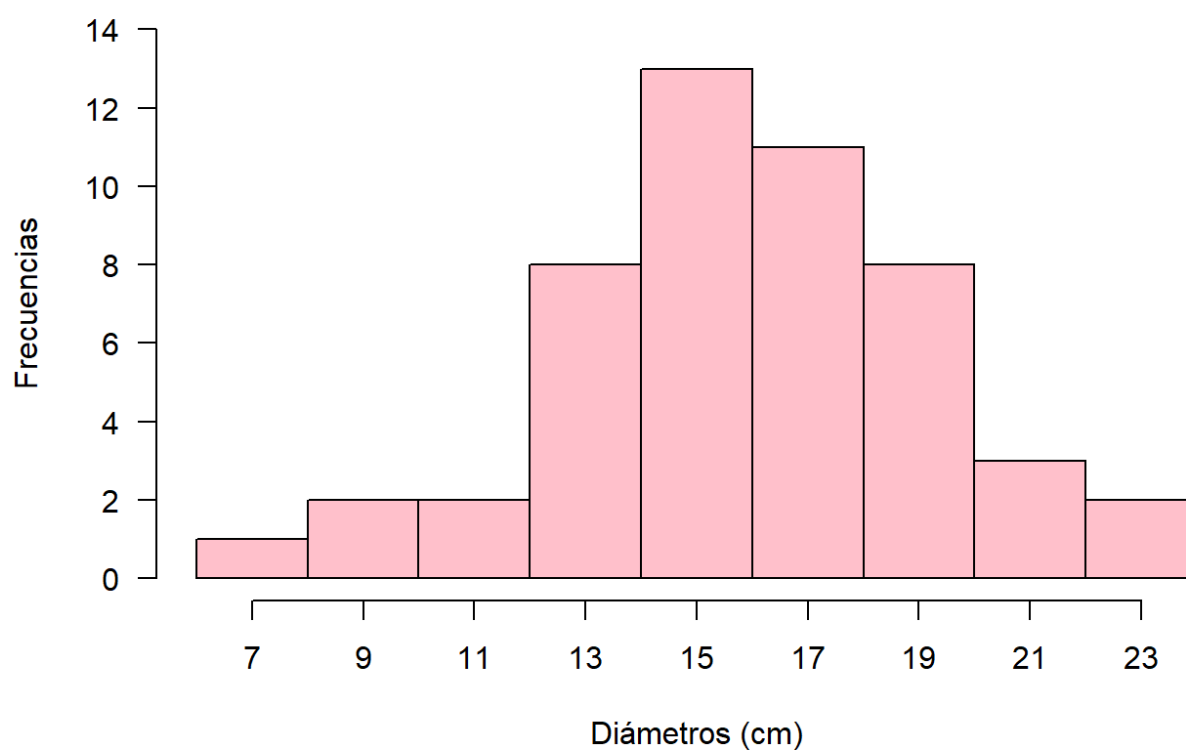
```
diam_hist
```

```
## $breaks
## [1]  6  8 10 12 14 16 18 20 22 24
##
## $counts
## [1]  1  2  2  8 13 11  8  3  2
##
## $density
## [1] 0.01 0.02 0.02 0.08 0.13 0.11 0.08 0.03 0.02
##
## $mids
## [1]  7  9 11 13 15 17 19 21 23
##
## $xname
## [1] "inventario$Diametros"
##
## $equidist
## [1] TRUE
##
## attr(,"class")
## [1] "histogram"
```

```
diam_hist$breaks
```

```
## [1] 6 8 10 12 14 16 18 20 22 24
```

```
h1 <- hist(inventario$Diametros, xaxt = "n",
           breaks = c(6, 8, 10, 12, 14, 16, 18, 20, 22, 24),
           col = "#FFC0CB", xlab="Diámetros (cm)",
           ylab= "Frecuencias",
           main = "",
           las = 1,
           ylim = c(0,14))
axis(1, h1$mids)
```



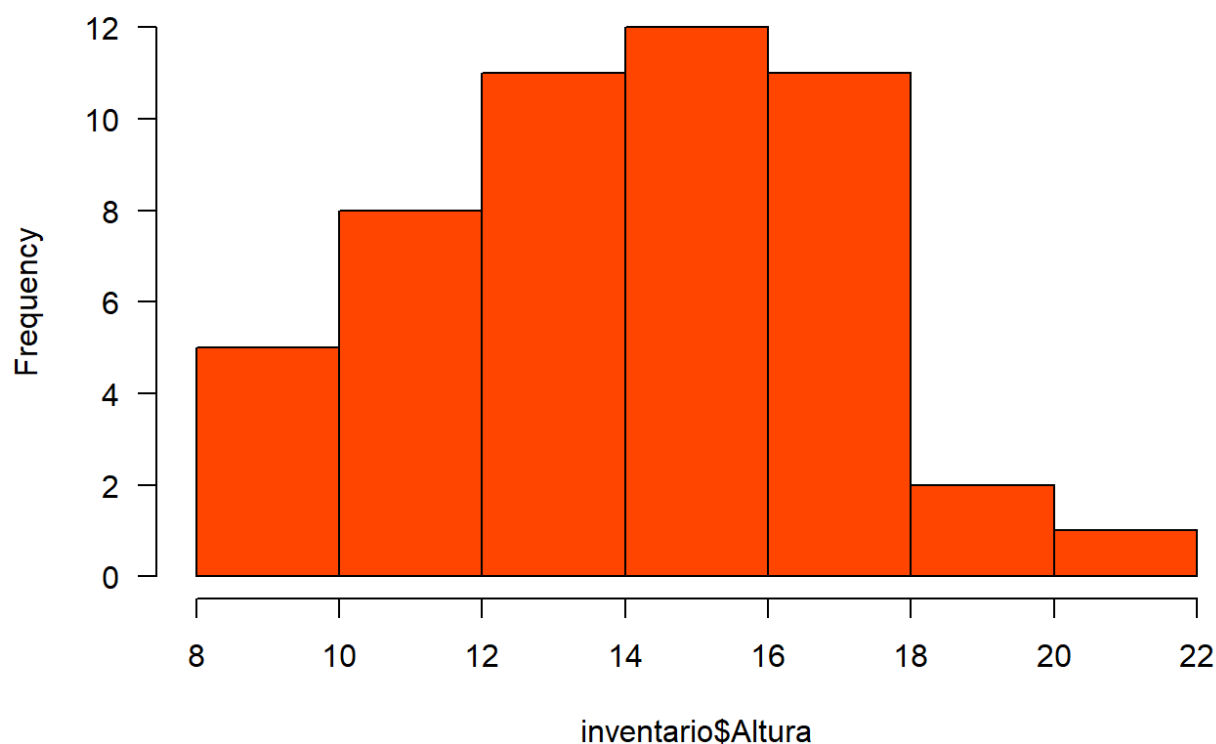
```
# AUTOESTUDIO -----
```

```
#Realizar el mismo procedimiento para la variable Altura
```

```
#HISTOGRAMA
```

```
alt_hist <- hist(inventario$Altura, las = 1, col = '#FF4500')
```

## Histogram of inventario\$Altura



```
alt_hist
```

```
## $breaks
## [1]  8 10 12 14 16 18 20 22
##
## $counts
## [1]  5  8 11 12 11  2  1
##
## $density
## [1] 0.05 0.08 0.11 0.12 0.11 0.02 0.01
##
## $mids
## [1]  9 11 13 15 17 19 21
##
## $xname
## [1] "inventario$Altura"
##
## $equidist
## [1] TRUE
##
## attr(,"class")
## [1] "histogram"
```

```
alt_hist$breaks
```

```
## [1] 8 10 12 14 16 18 20 22
```

```
#personalizar grafica
```

```
h1 <- hist(inventario$Altura, xaxt = "n",  
           breaks = c(8, 10, 12, 14, 16, 18, 20, 22),  
           col = "#87cefa", xlab="Diámetros (cm)",  
           ylab= "Frecuencias",  
           main = "",  
           las = 1,  
           ylim = c(0,14))  
axis(1, h1$mids)
```

