

LABORATORIO_04_JAIRO_LEAL.R

jairo

2023-02-25

```
# Laboratorio 4: Resumir datos gráficamente
```

```
# Maestria en Ciencias Forestales UANL
```

```
# Alumno: Jairo Alberto Leal Gómez
```

```
# Matricula: 1723093
```

```
# Objetivos de la práctica: Visualización de variables cuantitativas y Conozca la función hist
().
```

```
# Importar datos csv -----
```

```
# Para este laboratorio vamos a considerar los datos del inventario con diferentes especies y po
sición de los individuos dentro de un rodal.
```

```
esp.url <- paste0("https://raw.githubusercontent.com/mgtagle/", "PrincipiosEstadistica2021/main/c
uadro1.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
```

```
# El objeto inventario es un conjunto de datos. Por lo general, después de importar un grupo de dat
os en R, es posible que desee utilizar algunas funciones para inspeccionar sus propiedades y fun
ciones y estructura básica:
```

```
# 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
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 -----

Como mencionamos antes, una consideración importante tiene que ver con identificar el tipo de variables:cuantitativas vs cualitativas.

Un ejemplo de variable cualitativa es "posición". Esta variable contiene la posición de cada árbol dentro del rodal.

Cuando se inspecciona una variable cualitativa, normalmente se inicia calculando una tabla 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
```

A menudo, es conveniente expresar las frecuencias como proporciones o porcentajes, también conocido como 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
```

```
# Gráficas barplot y pie -----
```

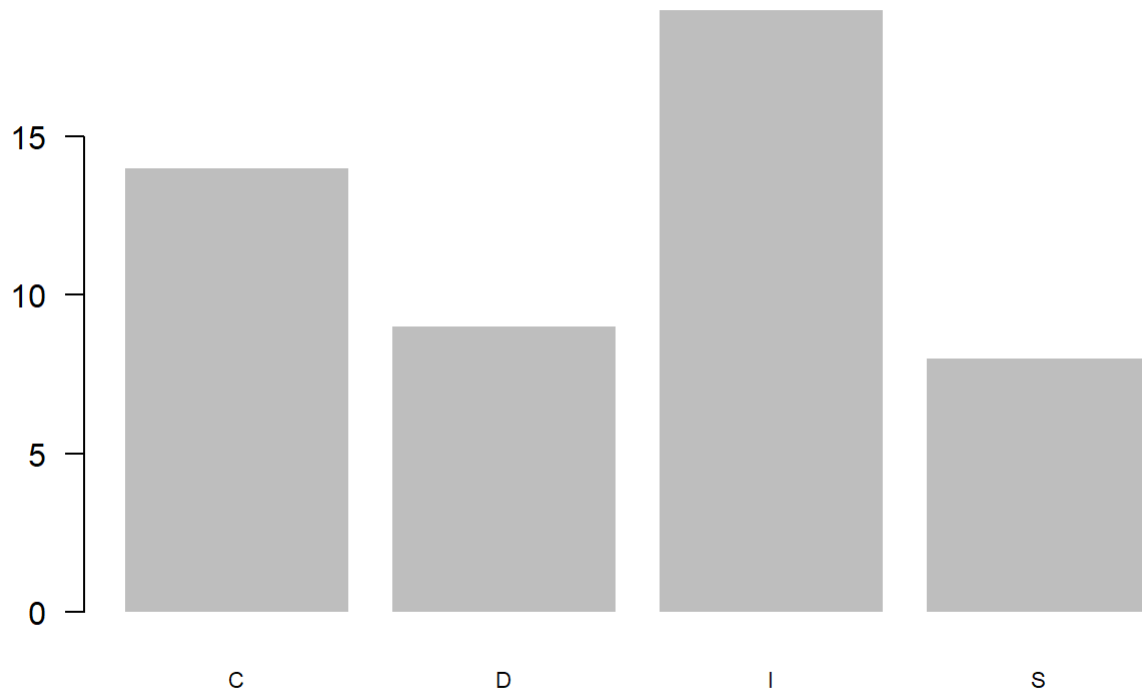
#Habiendo obtenido las frecuencias y / o proporciones de las categorías de un variable cualitativa, podemos continuar nuestra exploración con algunas representaciones visuales. Hay dos gráficos más comunes que se utilizan para visualizar frecuencias:

```
# Gráficas de barras (barplot)
```

```
# Gráficas de pastel (pie)
```

Para crear un gráfico de barras en R puede usar la función `barplot()`. Esta función requiere un vector numérico o una tabla de frecuencias:

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



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

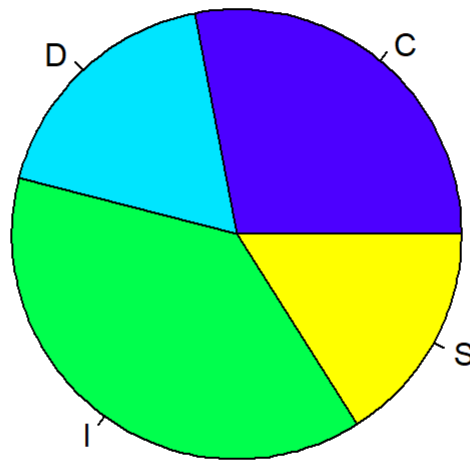
`las = 1`: muestra las frecuencias perpendiculares al eje-y.

`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).

Gráfico circular o `pie`. El otro tipo común de gráfico para ver frecuencias es un gráfico circular. R proporciona la función `pie()` para producir estos gráficos:

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



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

el paréntesis indica el # de colores a usar

Auto estudio -----

EJERCICIO: Completar una tabla de frecuencia y su representación gráfica (`barplot` y `pie`) para la variable `Especie` del conjunto de datos `inventario`

```
summary(inventario[3])
```

```
##   Especie
## Length:50
## Class :character
## Mode  :character
```

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

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

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

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

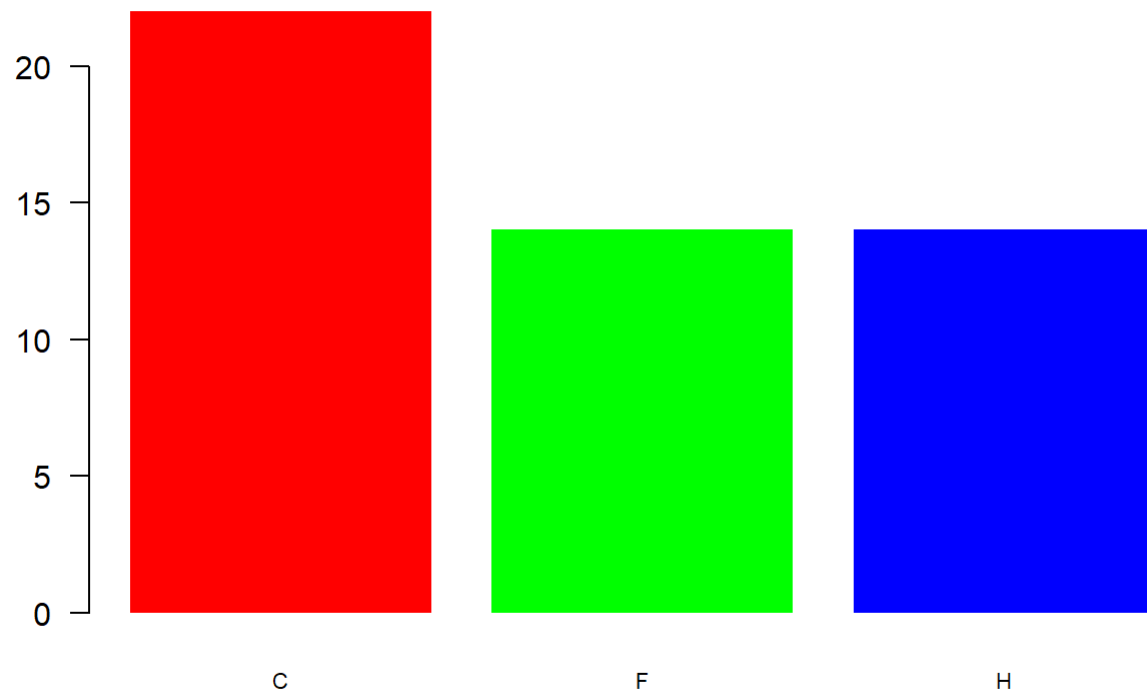
```
summary(inventario[3])
```

```
##   Especie
##   C:22
##   F:14
##   H:14
```

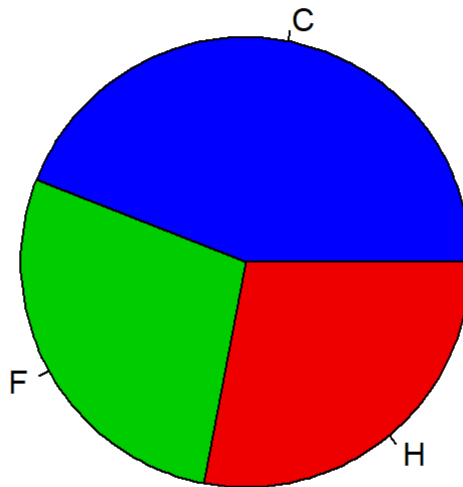
```
freq_especie <- table(inventario$Especie)
freq_especie
```

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

```
barplot(freq_especie, las = 1, border = NA, cex.names = 0.7, col=rainbow(3))
```



```
pie(freq_especie, col = c("blue", "green3", "red2"))
```



Representación de variables cuantitativas La mayoría de las variables del conjunto de datos Inventario son de naturaleza cuantitativa. Una posibilidad de inspeccionar visualmente esas variables es categorizarlas y luego usar un gráfico de barras o un gráfico circular. Otra posibilidad es utilizar gráficos específicamente para variables cuantitativas:

histogramas

boxplots o gráfica de cajas

Histogramas -----

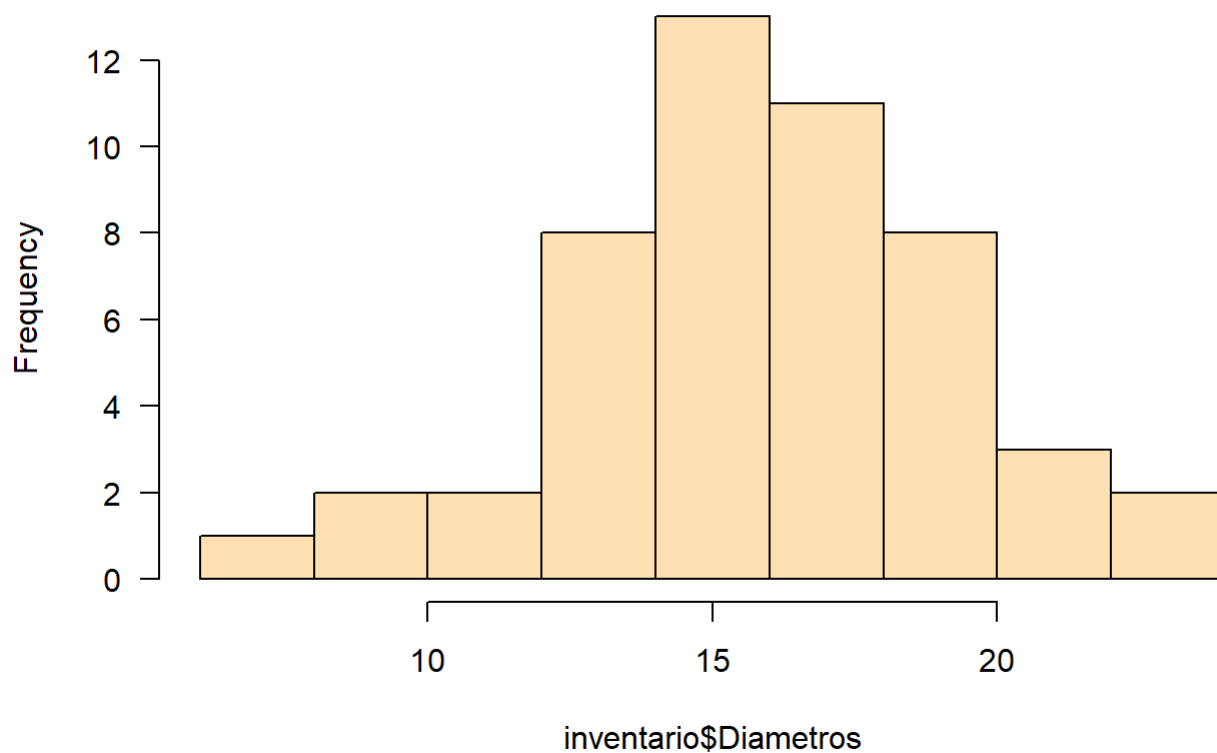
Un histograma es un tipo de gráfico que muestra la distribución de datos numéricos.

Para producir histogramas, R proporciona la función hist(). La acción predeterminada de hist() es para trazar un histograma, pero también puede almacenar su salida en un objeto R. Inspeccionar tal objeto le permitirá ver las diferentes componentes utilizadas para trazar el histograma.

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

```
diam_hist <- hist(inventario$Diametros, las = 1, col = '#ffe0b3')
```


Histogram of inventario\$Diametros



En un histograma, las barras son adyacentes (sin espacios entre barras). Es más, las barras no se pueden reorganizar en un orden diferente. A diferencia de los gráficos de barras, lo que importa en un histograma no es la longitud de las barras sino sus áreas. El área de una barra en un histograma debe ser igual a la proporción del intervalo de clase.

Debido a que almacenamos la salida producida por `hist()` en el objeto `diam_hist`, podemos escribir este objeto para ver qué salida contiene:

```
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"
```

```
# breaks: puntos de ruptura (corte) de los intervalos de clase
# counts: número de observaciones en cada categoría
# density: densidad
# mids: punto central del intervalo
# xname: nombre del objeto (variable) que se está graficando
# equidist: ¿Los categorías tienen el mismo ancho?
# attr: Tipo de clase
```

La función hist() produce un histograma usando configuraciones predefinidas. Por defecto, la función podrá determinar el número de bins o intervalos de clase automáticamente. Como la mayoría de los histogramas producidos por un software estadístico, los intervalos predeterminados son de igual tamaño. Además, los intervalos de clase están cerrados a la derecha (es decir, se incluye el punto final derecho). En el ejemplo anterior, esto significa que los intervalos son:

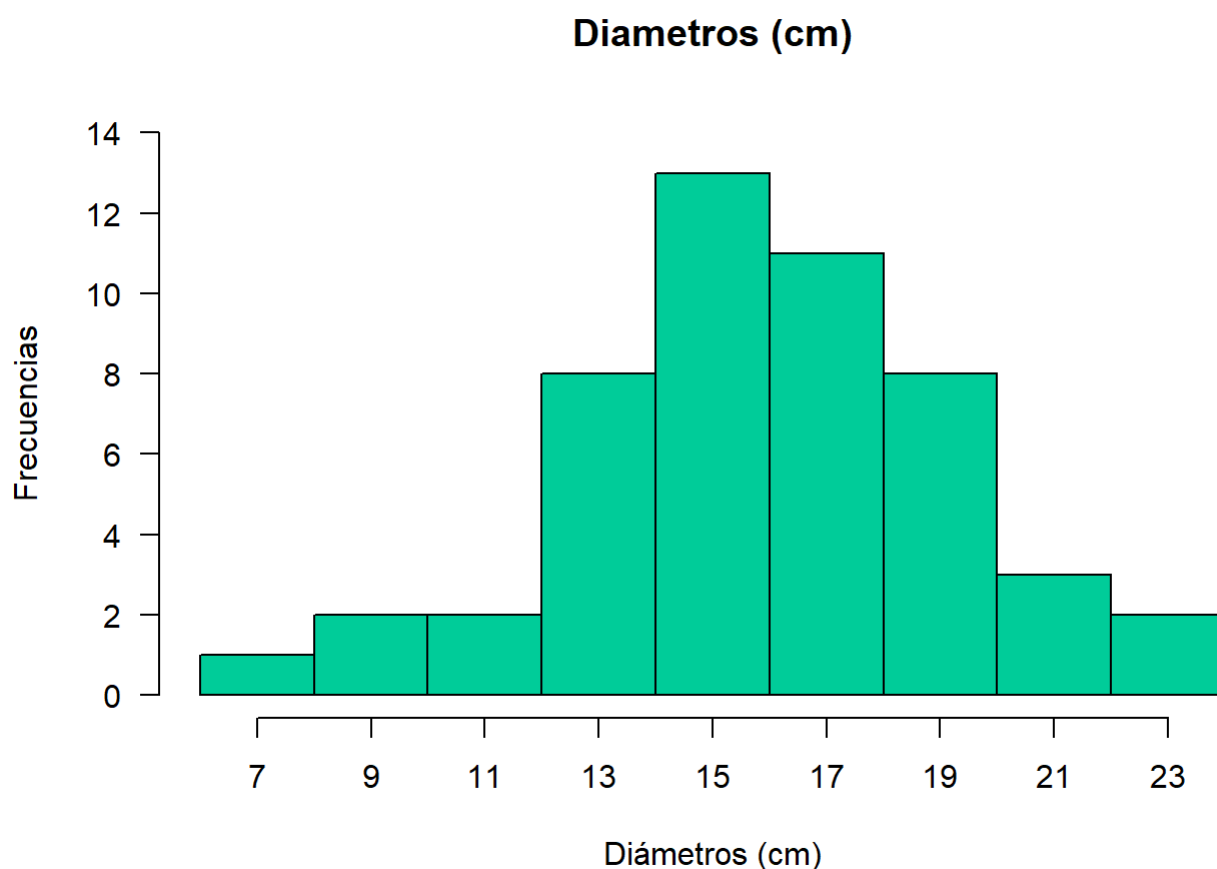
```
diam_hist$breaks
```

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

Como siguiente ejemplo vamos a personalizar la gráfica, definiendo los valores centrales de cada categoría : 7, 9, 11, 13, 15, 17, 19, 21, 23. Estos valores se encuentran en los atributos de l objeto que guardamos anteriormente `diam_hist$mids`, o para el ejemplo de abajo es `h1$mids`

```
h1 <- hist(inventario$Diametros, xaxt = "n",
           breaks = c(6, 8, 10, 12, 14, 16, 18, 20, 22, 24),
           col = "#00cc99",
           xlab="Diámetros (cm)", ylab= "Frecuencias",
           main = "Diámetros (cm)",
           las = 1,
           ylim = c(0,14))

axis(1, h1$mids)
```

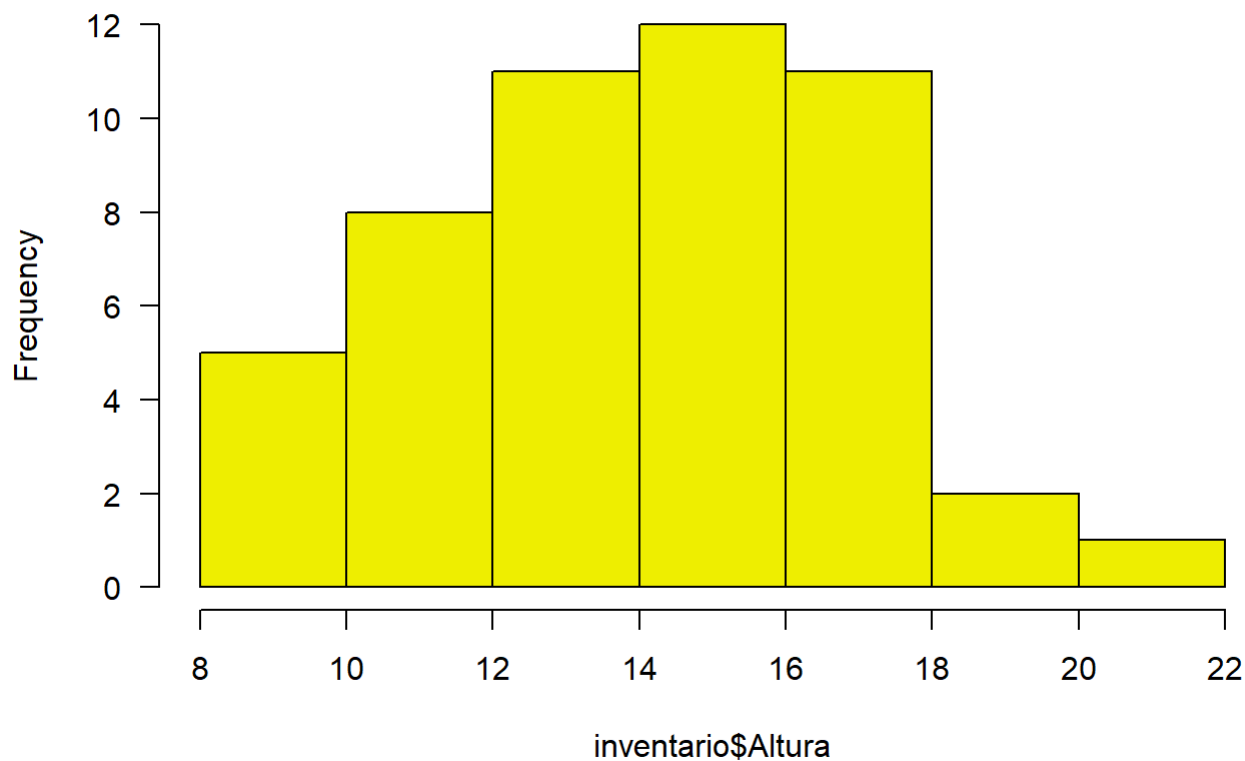


```
# Autoestudio 2 -----

# Realizar el mismo procedimiento para la variable Altura.

alt_hist <- hist(inventario$Altura, las = 1, col = "yellow2")
```

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
```

```
alt1 <- hist(inventario$Altura, xaxt = "n",  
             breaks = c(6, 8, 10, 12, 14, 16, 18, 20, 22, 24),  
             col = "blue",  
             xlab="Alturas (m)", ylab= "Frecuencias",  
             main = "Alturas (m)",  
             las = 1,  
             ylim = c(0,12))  
  
axis(1, alt1$mids)
```

