

# w1\_descriptiveStats\_partI

2024-07-16

## Tabla de frecuencias según leyes de porte de armas en 1999

Guns es una estructura de **datos de tipo panel** (Cada estado tiene su serie de tiempo propia). Para nuestro análisis inicial vamos a generar un subconjunto de un solo año.

En la línea anterior, generamos una nueva tabla de datos llamada **Guns\_1999**.

Para eso aplicamos el operador de extracción de fila y columnas [**condición de filas, condición de columnas**] a la tabla original Guns.

En este caso **solo condicionamos nuestra consulta a las filas**, específicamente tomamos los datos de todas las **filas en donde la variable ‘year’ tuviera el valor 1999**. La condición de columnas está vacía debido a que queremos tener todas las variables de la tabla: **no tenemos condiciones sobre estas**.

```
Guns_1999 <- Guns[Guns$year == 1999, ]  
  
# La función head nos permite ver las primeras observaciones de la tabla de datos Guns_1999.  
  
#head(Guns_1999)  
table(Guns_1999$law)
```

```
##  
##    no yes  
##    22  29
```

En este caso la variable ‘law’ cuenta con dos clases, ‘yes’ y ‘No’, siendo yes la más frecuente con un total de 29 de los 51 Estados.

## Tabla de frecuencias relativas

Para calcular la frecuencia relativa de la presencia de leyes de portes de armas por estado, debemos dividir la frecuencia observada en cada clase por el total de observaciones disponibles.

```
table(Guns_1999$law) / nrow(Guns_1999) # Frecuencias relativas
```

```
##  
##          no          yes  
## 0.4313725 0.5686275
```

```
table(Guns_1999$law) / nrow(Guns_1999) * 100 # Frecuencias relativas porcentuales
```

```
##  
##          no          yes  
## 43.13725 56.86275
```

Podemos consolidar estos resultados, e incluso construirlos, dentro de un objeto tipo *data frame*

```
RF <- data.frame(table(Guns_1999$law)) # Tabla de frecuencia convertida en una tabla de datos
n <- nrow(Guns_1999) # Numero de estados

RF$FreqR <- RF$Freq/n      # Frecuencia relativa
RF$FreqPR <- RF$FreqR * 100 # Frecuencia porcentual relativa

nombres <- c('Ley_porte', 'Frecuencia', 'Frecuencia_relativa', 'Frecuencia_p_relativa')
names(RF) <- nombres
row.names(RF) <- as.character(RF$Ley_porte)

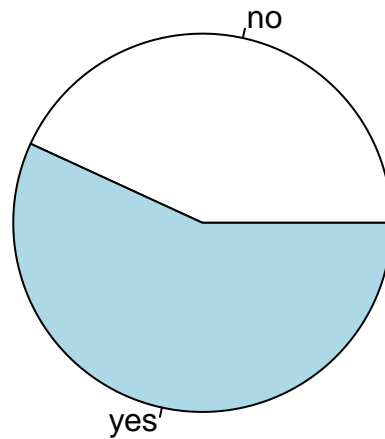
RF
```

```
##      Ley_porte Frecuencia Frecuencia_relativa Frecuencia_p_relativa
## no          no         22           0.4313725           43.13725
## yes         yes         29           0.5686275           56.86275
```

## Gráfico de torta

```
pie(
  RF$Frecuencia, # Cuales datos vamos a graficar?
  labels = RF$Ley_porte, # Cuales son las etiquetas de esos datos?
  main = "Gráfico de torta con la frecuencia de Leyes de porte en 51 estados (1999)" # Cual es el titulo?
)
```

## Gráfico de torta con la frecuencia de Leyes de porte en 51 estados (1999)



```
# Tambien podemos construir la torta desde la función table
# pie(
#   table(Guns_1999$law),
#   main = "Gráfico de torta con la frecuencia de Leyes de porte en 51 estados (1999)" # Cual es el título
# )
#
```

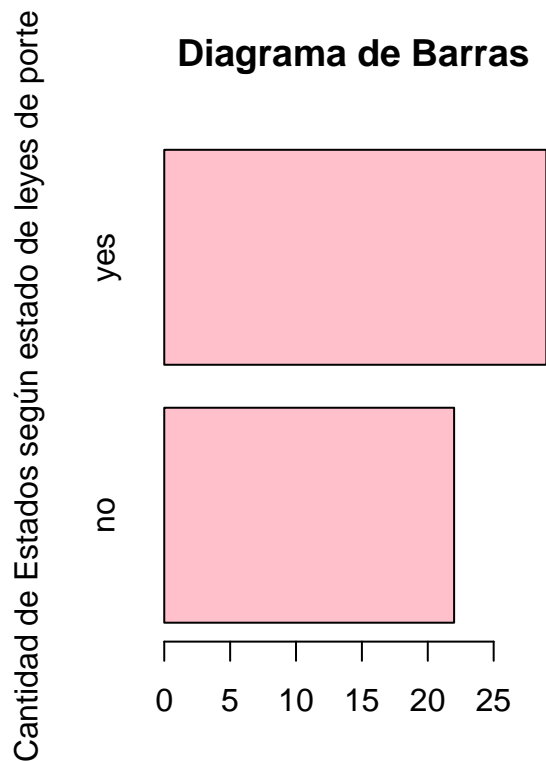
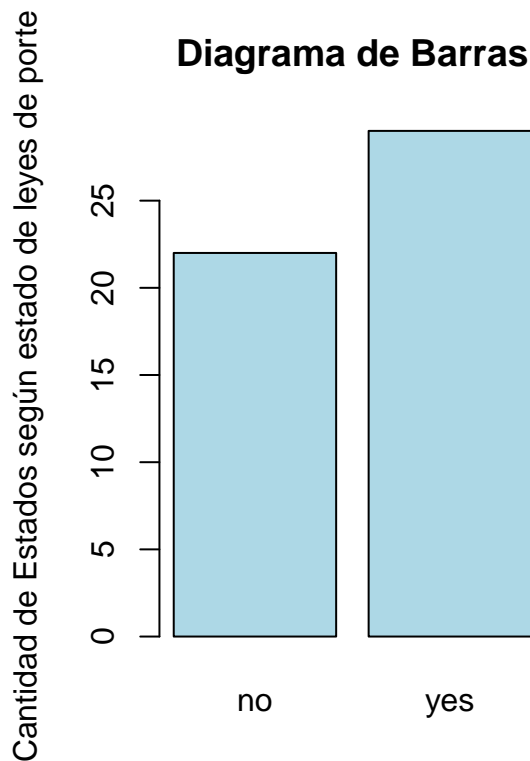
## Gráfico de barras

```
par(mfrow = c(1,2)) # argumento para cambiar nuestro ambiente gráfico. Lo definimos ahora con una fila y una columna

barplot(
  RF$Frecuencia,
  names.arg = RF$Ley_porte,
  main = "Diagrama de Barras",
  ylab = "Cantidad de Estados según estado de leyes de porte",
  col = "lightblue")

barplot(
  RF$Frecuencia,
  names.arg = RF$Ley_porte,
```

```
main = "Diagrama de Barras",
ylab = "Cantidad de Estados según estado de leyes de porte",
col = "pink",
horiz = TRUE)
```



## Gráfico de pareto

El Diagrama de Pareto es un caso particular del gráfico de barras, en el que las barras que representan los factores correspondientes a una magnitud cualquiera están ordenados de mayor a menor (en orden descendente) y de izquierda a derecha.

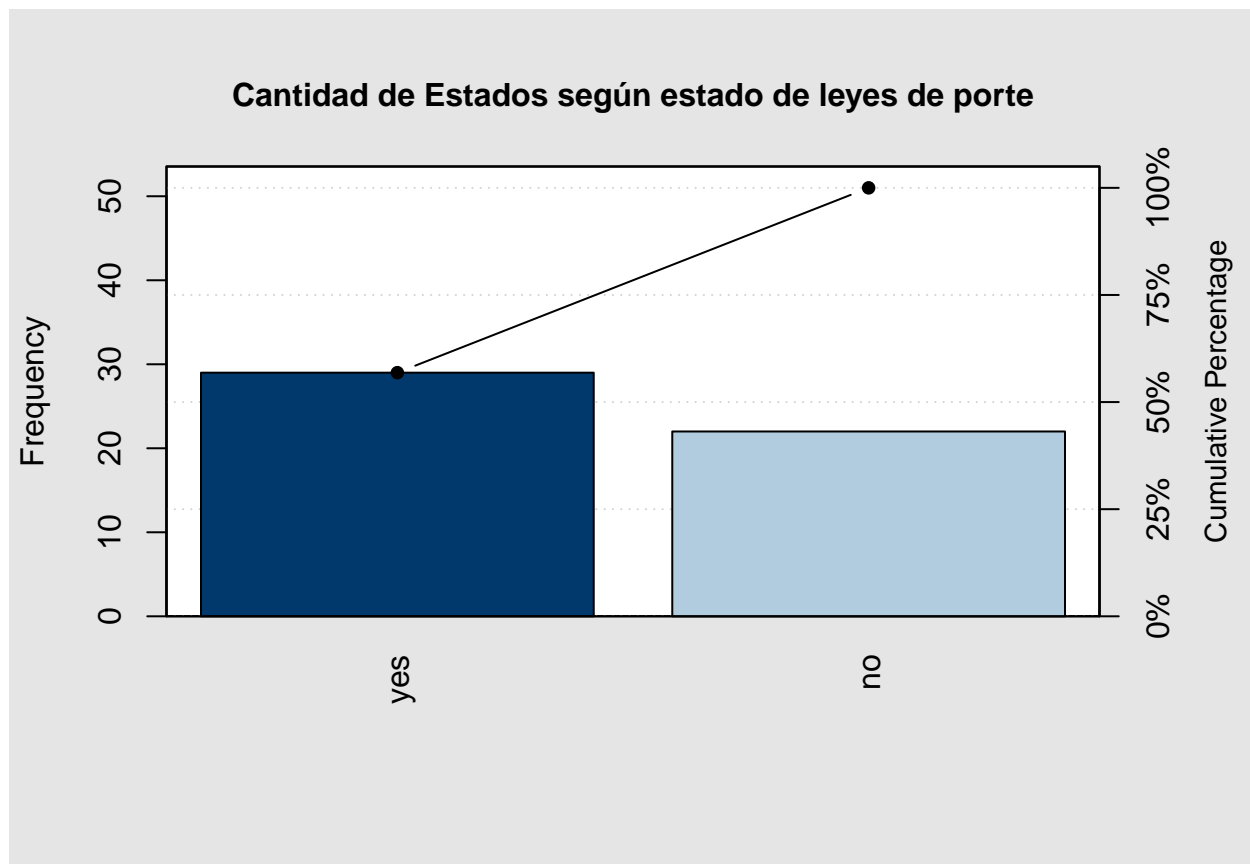
```
#install.packages("qcc")
library(qcc)
```

```
## Warning: package 'qcc' was built under R version 4.2.3
```

```
## Package 'qcc' version 2.7
```

```
## Type 'citation("qcc")' for citing this R package in publications.
```

```
qcc::pareto.chart(table(Guns_1999$law),
  main = "Cantidad de Estados según estado de leyes de porte")
```



```
##
## Pareto chart analysis for table(Guns_1999$law)
##      Frequency Cum.Freq. Percentage Cum.Percent.
##   yes  29.00000   29.00000    56.86275     56.86275
##   no   22.00000   51.00000    43.13725    100.00000
```

## Histograma

Un histograma se construye colocando la variable de interés en el eje horizontal y la frecuencia (o frecuencia relativa o frecuencia porcentual) en el eje vertical.

Uno de los usos más importantes de un histograma es proveer información acerca de la forma de la distribución de la variable.

```
# Clases
#
struge <- function(n){
  round(3.3 * (log(n)) + 1, 0)
} # Define una función, parametrizada en el número de observaciones, para definir el total de clases.

C <- struge(nrow(Guns)) # aplica la función del número de clases para todo el panel de datos Guns
print(C)
```

```
## [1] 24
```

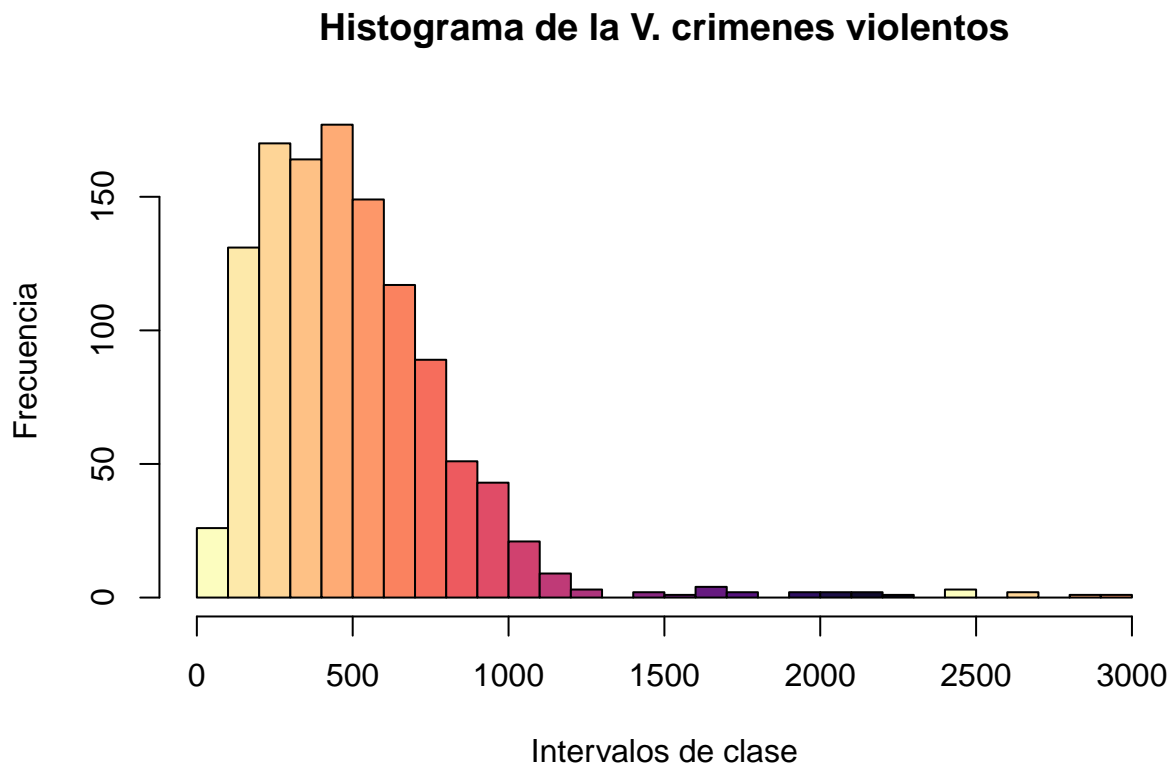
```
# Rango
#
R <- max(Guns$violent) - min(Guns$violent) # Con las funciones max y min, calcula el rango de la variable
print(R)
```

```
## [1] 2874.8
```

```
# Ancho
#
w <- R/C
print(w)
```

```
## [1] 119.7833
```

```
hist(Guns$violent,breaks = C,
     xlab = "Intervalos de clase",
     ylab = "Frecuencia",
     freq = TRUE,
     main = "Histograma de la V. crímenes violentos",
     col = viridis::magma(C, direction = -1))
```



## Tablas cruzadas

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.2.3
```

```
##
```

```
## Attaching package: 'dplyr'
```

```
## The following object is masked from 'package:car':
```

```
##
```

```
##      recode
```

```
## The following objects are masked from 'package:stats':
```

```
##
```

```
##      filter, lag
```

```
## The following objects are masked from 'package:base':
```

```
##
```

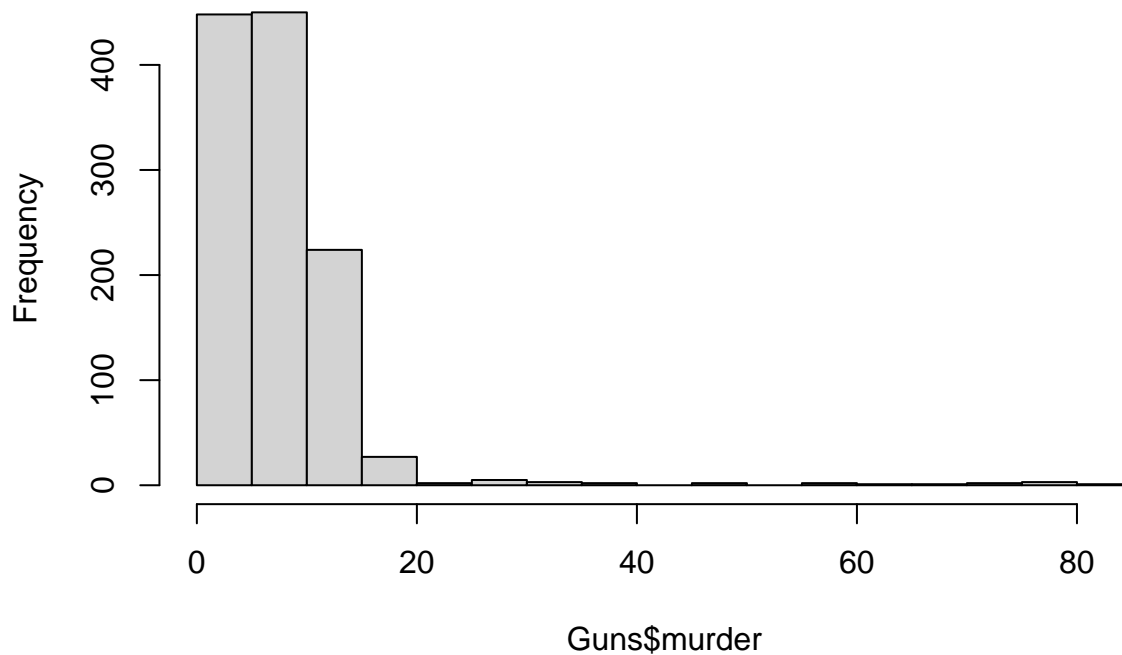
```
##      intersect, setdiff, setequal, union
```

```
library(tidyr)
```

```
## Warning: package 'tidyr' was built under R version 4.2.3
```

```
hist_murder <- hist(Guns$murder,breaks = C )
```

## Histogram of Guns\$murder



```
Guns$class_murder <- cut(Guns$murder,
                        hist_murder$breaks)
```

```
Guns[Guns$year == 1999,] %>%
  group_by(class_murder, law) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n)) %>%
  select(-n) %>%
  spread(class_murder, freq)
```

## 'summarise()' has grouped output by 'class\_murder'. You can override using the  
## '.groups' argument.

```
## # A tibble: 2 x 5
##   law   '(0,5]' '(5,10]' '(10,15]' '(45,50]'
##   <fct>   <dbl>   <dbl>   <dbl>   <dbl>
## 1 no      0.5    0.348     NA      1
## 2 yes     0.5    0.652     1      NA
```

```
Guns[Guns$year == 1999,] %>%
  group_by(class_murder, law) %>%
  summarise(n = n()) %>%
  spread(class_murder, n)
```

## 'summarise()' has grouped output by 'class\_murder'. You can override using the



```
## '.groups' argument.
```

```
## # A tibble: 2 x 5
##   law      '(0,5]' '(5,10]' '(10,15]' '(45,50]'
##   <fct>   <int>   <int>   <int>   <int>
## 1 no      13      8      NA      1
## 2 yes     13     15      1     NA
```

```
Guns %>%
  group_by(year, law) %>%
  summarise(n = n()) %>%
  spread(law, n)
```

```
## 'summarise()' has grouped output by 'year'. You can override using the
## '.groups' argument.
```

```
## # A tibble: 23 x 3
## # Groups:   year [23]
##   year    no  yes
##   <fct> <int> <int>
## 1 1977    47    4
## 2 1978    47    4
## 3 1979    47    4
## 4 1980    47    4
## 5 1981    47    4
## 6 1982    46    5
## 7 1983    46    5
## 8 1984    46    5
## 9 1985    46    5
## 10 1986    44    7
## # i 13 more rows
```

## Diagramas de dispersión

```
plot(Guns_1999$violent,
     Guns_1999$male,
     col = "lightblue",
     main = "Scatterplot",
     xlab = "Acciones violentas",
     ylab = "Proporción de hombres jóvenes")

abline(lm(Guns_1999$male ~ Guns_1999$violent), col = "red")
```

**Scatterplot**

