

Programación y Uso de Librerías en R: Herramientas de Análisis y Visualización de Datos

Juan Luis Peñaloza Figueroa
Universidad Complutense de Madrid

Milagros Dones Tacero
Universidad Autónoma de Madrid

Carmen Gladys Vargas Pérez
Universidad Complutense de Madrid

AÑO: 2025

SCRIPT_15: CAPÍTULO XVII. VISUALIZACIÓN DE IMÁGENES

```
# Cargar la tabla de datos en R
> df<-read.csv("http://r.datametria.com/wp-content/uploads/
2020/08/ex3-2003.csv")
> names(df) [1:4]<-c("score", "negro", "ingresos", "patrimonio")
> attach(df)
> par(mfcol=c(2,3), mar=rep(2,4)) for (i in 1:length(tipo)) {
plot(ingresos, score, type = tipo[i])
}
# Argumento main
# Creamos el vector de títulos
> titu<-c("Puntos", "Líneas", "Puntos unidos con líneas",
"Puntos cubiertos", "Líneas Verticales",
"Escalera")
# Corremos el for para poder ver los títulos
> par(mfcol=c(2,3), mar=rep(2,4)) for (i in 1:length(tipo)) {
plot(ingresos, score, type = tipo[i], main = paste("Gráfico tipo:",
titu[i])) }
# Argumento sub
# Agregando los subtítulos a nuestras gráficas.
> par(mfcol=c(2,3), mar=c(5,3,2,1)) for (i in 1:length(tipo))
{plot(ingresos, score, type = tipo[i], main = paste("Gráfico tipo:",
titu[i]), sub = "Elaboración Propia") }
# Colocando las etiquetas a los ejes con los argumentos xlab e ylab.
> par(mfcol=c(2,3), mar=c(5,4,2,1)) for (i in 1:length(tipo)) {
plot(ingresos, score, type = tipo[i], main = paste("Gráfico tipo:",
titu[i]), sub = "Elaboración Propia", xlab = "INGRESOS DE LOS DE
SELECTIVIDAD", ylab = "PUNTUACIÓN OR CRÉDITO") }
## GRÁFICAS UNIDIMENSIONALES Y BIDIMENSIONALES EN R
# Cargando la tabla ex3-2003 y guardándolo en el data-frame (df)
> df<-read.csv("http://r.datametria.com/wp-content/uploads/
2020/08/ex3-2003.csv")
> names(df) [1:4]<-c("score", "negro", "ingresos", "patrimonio")
> plot(df$score)
> hist(df$ingresos)
> boxplot(x, ...)
> x <- c(8,5,14,-9,19,12,3,9,7,4,4,6,8,12,-8,2,0,-1,5,3)
> boxplot(x, horizontal = TRUE)
> stripchart(x,method="jitter",pch=19,add=TRUE, col="blue")
# Gráfico de cajas por grupo
# Datos chickwts de R base.
> head(chickwts)
> boxplot(chickwts$weight ~ chickwts$feed)
```

```

> boxplot(weight ~ feed, data = chickwts) # Equivalente
> stripchart(chickwts$weight ~ chickwts$feed, vertical = TRUE, method
= "jitter", pch = 19, add = TRUE, col = 1:length(levels(chickwts$feed)))
# Múltiples boxplots
# datos trees
> head(trees)
> stacked_df <- stack(trees)
> head(stacked_df)
> boxplot(trees, col = rainbow(ncol(trees)))
# Equivalente
> boxplot(stacked_df$values ~ stacked_df$ind, col = rainbow(ncol(trees)))
# Diagrama de caja diferente para cada columna
> par(mfrow = c(1, ncol(trees)))
> invisible(lapply(1:ncol(trees), function(i) boxplot(trees[, i])))
# Agregar la media en boxplots por grupo
> par(mfrow = c(1, 2)) mi_df <- trees
# Boxplot vertical por grupo
> boxplot(mi_df, col = rgb(0, 1, 1, alpha = 0.25))
# Añadir líneas con las medias invisible(lapply(1:ncol(mi_df),
> function(i) segments(x0 = i - 0.4, y0 = mean(mi_df[, i]), x1 = i +
0.4, y1 = mean(mi_df[, i]), col = "red", lwd = 2)))
# Añadir puntos con las medias
> medias <- apply(mi_df, 2, mean)
> medias <- colMeans(mi_df) # Equivalente (más eficiente)
> points(medias, col = "red", pch = 19)
# Boxplot horizontal por grupo
> boxplot(mi_df, col = rgb(0, 1, 1, alpha = 0.25), horizontal = TRUE)
# Añadir líneas con las medias invisible(lapply(1:ncol(mi_df)
> function(i) segments(x0 = mean(mi_df[, i]), y0 = i - 0.4, x1 =
mean(mi_df[, i]), y1 = i + 0.4, col = "red", lwd = 2)))
# Añadir puntos con las medias medias <- apply(mi_df, 2, mean)
> medias <- colMeans(mi_df), points(medias, 1:ncol(mi_df), col = "red",
pch = 19), par(mfrow = c(1, 1))
# GRÁFICAS BIDIMENSIONAL EN R
> x <- rnorm(500)
> y <- x + rnorm(500)
# plot(x,y)
# M <- cbind(x,y)
> plot(x,y)
# Diferentes tipos de gráfico con la función plot().
> par(mfrow = c(2, 3))
# Datos
> mi_ts <- ts(matrix(rnorm(500), nrow = 500, ncol = 1),
start = c(1950, 1), frequency = 12)
> mis_fechas <- seq(as.Date("2005/1/1"), by = "month", length =
50)
> mi_factor <- factor(mtcars$cyl)
> fun <- function(x) x^2
# Diagrama de dispersión
plot(x, y, main = "Gráfico de dispersión")
# Gráfico de barras
plot(mi_factor, main = "Diagrama de barras")
# Diagrama de cajas
plot(mi_factor, rnorm(32), main = "Diagrama de cajas y bigotes")
# Gráfico de una serie temporal
plot(mi_ts, main = "Serie temporal")
# Time-based plot
plot(mis_fechas, rnorm(50), main = "Gráfico basado en fechas")
# Dibujar una función
plot(fun, 0, 10, main = "Trazar una función")

```

```

# Gráfico de correlación
> plot(trees[, 1:3], main = "Matriz de correlación")
> par(mfrow = c(1, 1))
# Tipos de plot en R
# Generamos un conjunto de datos
> j <- 1:20
> k <- j
> par(mfrow = c(1, 3))
> plot(j, k, type = "l", main = "type = 'l'")
> plot(j, k, type = "s", main = "type = 's'")
> plot(j, k, type = "p", main = "type = 'p'")
> par(mfrow = c(1, 1))
# Generamos un nuevo conjunto de datos
> j <- 1:20
> k <- j
> par(mfrow = c(1, 3))
> plot(j, k, type = "o", main = "type = 'o'")
> plot(j, k, type = "b", main = "type = 'b'")
> plot(j, k, type = "h", main = "type = 'h'")
> par(mfrow = c(1, 1))
# Función LOG
> s <- 1:25
> u <- 1:25
> par(mfrow = c(1, 3))
# Log-log
> plot(log(s), log(u), pch = 19, main = "log-log")
# log(x)
> plot(log(s), u, pch = 19, main = "log(x)")
# log(y)
> plot(s, log(u), pch = 19, main = "log(y)")
> par(mfrow = c(1, 1))
> install.packages("ggplot2")
> library(ggplot2)
# Cargamos los datos en la sesión de trabajo
> install.packages("tidyverse")
> library(tidyverse)
> file.choose()
> head(Notas)
> ggplot(data = Notas) + geom_point(aes(x = nota09, y = nota10))
> ggplot(data = Notas) +
+   geom_point(aes(x = nota09, y = nota10, col = tipo))
> ggplot(data = Notas) +
+   geom_point(aes(x = nota09, y = nota10, shape = tipo, col = tipo))
# Cargamos la tabla de datos.
> df<-read.table("https://r.datametria.com/wpcontent/uploads/2020/09/p
artyid4.txt")
> head(df)
# Densidad de la edad por cuartiles
> area_quantile(df$age, titulo = "Grafico de densidad de la edad
dividido por cuartiles", xlab="Edad", ylab="Densidad",
color_area=c("darkolivegreen1","dodgerblue","gold1","orangered")
)
# Definiendo el número de elementos por cuartil.
> cuartiles<-n_percentil(df$age)
> area_quantile(df$age, titulo = "Grafico de densidad de la edad
dividido por cuartiles", xlab="Edad", ylab="",
color_area=c("darkolivegreen1","dodgerblue","gold1","orangered"),
bty="n", yaxt="n")
legend("topright", legend = c(paste("1° cuartil =", cuartiles[1]),
paste("2° cuartil =", cuartiles[2]),
paste("3° cuartil =", cuartiles[3])),

```

```
paste("4° cuartil =", cuartiles[4])),col = c("darkolivegreen1",  
"dodgerblue", "gold1", "orangered"), cex=0.7, pch = 19)
```

-----OOO-----