

# **Talleres de Análisis Multivariados en R**

**Maestría en Ecología y Biodiversidad**

Javier Rodríguez-Barrios

Invalid Date

# Tabla de contenidos

<b>2</b>	<b>Resumen</b>	<b>4</b>
<b>3</b>	<b>Introducción</b>	<b>5</b>
<b>4</b>	<b>Taller 1. Introducción a R y a Tidyverse</b>	<b>6</b>
4.0.1	R como calculadora . . . . .	6
4.0.2	Asignaciones . . . . .	6
4.0.3	Algebra . . . . .	7
4.0.4	Bases de datos (data.frames - cbind) . . . . .	8
4.0.5	Enlaces de operaciones en R . . . . .	8
<b>5</b>	<b>Importar y exportar bases de datos (read y write)</b>	<b>9</b>
5.0.1	Complementos requeridos . . . . .	9
5.0.2	Base de datos del ejercicio . . . . .	9
5.0.3	Exportación . . . . .	9
5.0.4	Importación . . . . .	10
5.0.5	Importación desde internet . . . . .	10
5.0.6	Enlaces de importación y Exportación de datos . . . . .	10
<b>6</b>	<b>Introductorio a Tidyverse</b>	<b>12</b>
6.0.1	Importar la base de datos (datos) . . . . .	12
6.0.2	Manipulación de los datos enlace . . . . .	12
6.0.3	Nueva base de datos con factor . . . . .	12
6.0.4	Filtrando elementos del dataframe (filter) . . . . .	13
6.0.5	Filtrando en orden descendente o ascendente (arrange) . . . . .	13
6.0.6	Combinación de filtrado y orden (filter + arrange) . . . . .	13
6.0.7	Generación de variables derivadas (arrange) . . . . .	14
6.0.8	Combinación de filtrado, nuevas variables y orden (filter + mutate + arrange) . . . . .	14
6.0.9	Enlaces de operaciones en Tidyverse . . . . .	14
<b>7</b>	<b>Taller práctico para la casa</b>	<b>16</b>
<b>8</b>	<b>Taller 2.1 Operaciones matriciales</b>	<b>17</b>
8.0.1	1.1 producto matricial . . . . .	17
8.0.2	1.2 Determinantes . . . . .	18

8.0.3	1.3 Matriz inversa . . . . .	18
8.0.4	1.4 Matriz de varianza - covarianza* . . . . .	19
8.0.5	1.5 Covarianza generalizada (S) . . . . .	19
<b>9</b>	<b>Aplicación de matrices en un Análisis de Componentes Principales - PCA</b>	<b>20</b>
9.0.1	1. Matriz centrada (m.c) y matriz rotada (m.r) . . . . .	20
<b>10</b>	<b>Análisis de Componentes principales - PCA en librería "vegan"</b>	<b>23</b>
<b>11</b>	<b>Operaciones matriciales - base <i>Caimanes</i></b>	<b>24</b>
11.1	Procedimiento del PCA . . . . .	24
11.1.1	Análisis de Componentes principales - PCA en librería " <b>vegan</b> " . . . . .	27
<b>12</b>	<b>Taller 3. Exploración Univariada y Multivariada</b>	<b>28</b>
12.1	Procedimiento de la exploración . . . . .	28
12.1.1	1. Figuras de elipses . . . . .	30
12.1.2	3. Figuras de Dispersión por pares de variables (pairs) . . . . .	34
12.1.3	3. Histogramas . . . . .	38
12.1.4	4. Dispersión X-Y . . . . .	43
12.1.5	5. Cajas y Bigotes . . . . .	47
12.1.6	6. Coplot . . . . .	51
12.1.7	7. Figuras con estadísticos (promedios, errores, ...) . . . . .	54
12.2	Taller de entrenamiento . . . . .	60
<b>13</b>	<b>Taller 4.1 Análisis de Componentes Principales - PCA</b>	<b>62</b>
<b>14</b>	<b>Exploración Gráfica</b>	<b>64</b>
<b>15</b>	<b>1) PCA con el paquete stats (pca1)</b>	<b>68</b>
<b>16</b>	<b>2) PCA con el paquete FactoMiner</b>	<b>72</b>
<b>17</b>	<b>3) PCA con el paquete vegan</b>	<b>75</b>
<b>18</b>	<b>Taller en casa</b>	<b>78</b>
<b>19</b>	<b>4) Análisis avanzado de PCA</b>	<b>79</b>
19.0.1	Combinación de clasificación y ordenación . . . . .	79
<b>20</b>	<b>5) PCA por tipos con la función "dudi.pca" del paquete ade4</b>	<b>82</b>
<b>21</b>	<b>Taller 4.2 Análisis de Componentes Principales - PCA</b>	<b>85</b>
<b>22</b>	<b>Exploración Gráfica</b>	<b>87</b>

<b>23 1) Ajuste de las bases de datos fisiqcoquímica (amb) y biológica (tax.hel)</b>	<b>88</b>
<b>24 2) PCA con paquete factextra</b>	<b>90</b>
24.0.1 2.1) Contribución eje 1 . . . . .	90
24.0.2 2.3) Escala de contribuciones de las observaciones y las variables	92
<b>25 3) PCA con vegan</b>	<b>94</b>
25.0.1 3.1) Insumos del análisis . . . . .	94
25.0.2 3.2) Autovalores . . . . .	94
25.0.3 3.3) Figura del PCA . . . . .	94
25.0.4 3.4) PCA con vegan - biplot + orditorp . . . . .	95
25.0.5 3.5) PCA con vegan + orditorp + envfit (ajuste ambiental) . .	96
<b>26 4) PCA con paquete ggplot2</b>	<b>98</b>
26.0.1 4.1 Coordenadas de los sitios y el factor “coord.sit” . . . . .	98
26.0.2 4.2 Coordenadas de los taxones “coord.tax” . . . . .	98
26.0.3 4.3 Coordenadas de las ambientales “coord.amb” . . . . .	99
26.0.4 4.4 Figura con de elipses por concavidades - geom_mark_hull	99
26.0.5 4.5 Figura con vectores de especies y ambientales . . . . .	101
26.1 Taller de entrenamiento . . . . .	102
<b>27 Taller 5.1 Análisis de Escalamiento Multidimensional no Métrico - NMDS</b>	<b>103</b>
27.0.1 Referencias bibliográficas de apoyo. . . . .	103
27.0.2 Procedimiento de la exploración . . . . .	104
27.0.3 1) Ordenación de las localidades y las especies de malezas. . . . .	104
27.0.4 2) Figuras del nmbs con el paquete “vegan” . . . . .	105
27.0.5 3) NMDS con paquete ggplot2 . . . . .	107
27.0.6 Taller de entrenamiento . . . . .	110
<b>28 Taller 6.1 Análisis de Correspondencias Múltiples - MCA</b>	<b>111</b>
28.0.1 Procedimiento de la exploración . . . . .	111
28.0.2 1) Ajuste de la base de datos de bagres. . . . .	112
28.0.3 2) Primera ordenación de las variables cualitativas activas (mca1). . . .	113
28.0.4 3) Segunda ordenación de las variables cualitativas activas (mca2). . . .	122
28.0.5 Taller de entrenamiento . . . . .	126
<b>29 Taller 7.1 Análisis de Redundancia - RDA</b>	<b>127</b>
29.0.1 Doce pasos para el análisis de redundancia - RDA. . . . .	130
29.0.2 Taller de entrenamiento . . . . .	150
<b>30 Taller 8.1 Análisis de Clúster - CLA</b>	<b>151</b>
30.0.1 Referencias bibliográficas de apoyo. . . . .	151
30.0.2 Cuatro pasos para el análisis de clúster . . . . .	159

<b>31 Taller 9.1 Análisis Discriminante Lineal - LDA</b>	<b>179</b>
31.0.1 Referencias bibliográficas de apoyo. . . . .	180
31.0.2 Cargar las librerías requeridas . . . . .	180
31.0.3 Cargar o importar la base de datos . . . . .	180
31.0.4 Exploración de los datos . . . . .	181
31.0.5 Tres pasos para la realización del discriminante lineal - LDA . . . . .	188
<b>32 Taller 10.1 Análisis de Varianza Multivariado - MANOVA</b>	<b>199</b>
32.0.1 Referencias bibliográficas de apoyo. . . . .	200
32.0.2 Cargar las librerías requeridas . . . . .	200
32.0.3 Cargar o importar la base de datos . . . . .	200
32.0.4 Exploración de los datos . . . . .	201
32.0.5 Cuatro pasos para la realización del MANOVA . . . . .	202
<b>Referencias</b>	<b>213</b>

# 1

Talleres de Estadística Multivariada

Javier Rodríguez Barrios

Maestría en Ecología y Biodiversidad

“Actualizado: 2023-03-26”

## 2 Resumen

El siguiente manual es un compendio de temas vistos en el componente práctico del módulo de estadística multivariada de la Maestría en Ecología y Biodiversidad. Se fundamenta en cuatro grandes módulos:

- Introducción al RStudio y al álgebra lineal aplicada a multivariados.
- Exploración y visualización gráfica de datos univariados y multivariados.
- Técnicas de ordenación multivariada (PCA, NMDS, MCA y RDA).
- Técnicas de clasificación multivariada (CLA, LDA)
- Pruebas de hipótesis (MANOVAS y PERMANOVAS).

Para un mejor entendimiento de este manual, se sugiere complementar la información con las referencias bibliográficas sugeridas en cada capítulo, especialmente el libro **\*\*Análisis de datos ecológicos y ambientales - aplicaciones en el programa R\*\*** de Rodríguez-Barrios (2023) [enlace](#)

## 3 Introducción

This is a book created from markdown and executable code.

See Knuth (1984) for additional discussion of literate programming.

```
1 + 1
```

```
[1] 2
```



## 4 Taller 1. Introducción a R y a Tidyverse

### 4.0.1 R como calculadora

```
# Operaciones aritméticas básicas
5 + 7  # Suma
5 - 3  # Resta
5 * 7  # Multiplicación
5/3    # División
2^3    # Exponentes

# Logaritmos y exponenciales
x = 5/3
log2(x)  # Logaritmo en base 2 de x
log10(x) # Logaritmo en base 10 de x
exp(x)   # Exponencial de x

# Funciones trigonométricas :
cos(x)   # Coseno de x
sin(x)   # Seno de x
tan(x)   # Tangente de x
```

### 4.0.2 Asignaciones

```
# 2) Asignación de valores a objetos o a variables
sitios <- 2      # Número de sitios = 2
sitios = 2      # Otra forma
n.sitios <- "dos" # Número de sitios como un caracter
dos.sitios <- TRUE # Objeto lógico
```

### 4.0.3 Algebra

```
# Vectores
sitios <- c(2, 3, 2, 3) # Vector sitios
sitios                # Imprimir el vector

sitios <- c("dos", "tres", "dos", "dos") # Vector como caracter
sitios

abundancia <- c(TRUE, FALSE, TRUE, TRUE) # vector con elementos lógicos
abundancia

# Vectores (continuación)
sitios <- c(2, 3, 2, 3) # Vector sitios
names(sitios) <- c("dos", "tres", "dos", "dos") # Nombres de los elementos

sitios <- c(dos= 2, tres= 3, dos= 2, dos= 2) # Otra forma
sitios

# Vectores (continuación)
sitios [1:3]      # Tres primeros elementos del vector sitios
sitios[c(1,4)]    # Primer y cuarto elemento del vector
sitios [-1]       # Eliminar el primer elemento del vector

# Matrices
Matriz <- matrix(c(1:15),5,3, byrow= FALSE) # 5,3: Número de filas y columnas
Matriz

# Matrices (continuación)
t(Matriz)          # Transpuesta de la Matriz
Matriz[2,]         # Ver la segunda fila (coma a la derecha del dato)
Matriz[,2]         # Ver la segunda columna (coma a la izquierda del dato)
Matriz[2:4,]       # Filas 2 a la 4
Matriz[c(2,4),]    # Filas 2 y 4
Matriz[ ... ]      # Valores de la fila 3 y de las columnas 1:3
Matriz[ ... ]      # Excluye a la 3a fila
```

#### 4.0.4 Bases de datos (data.frames - cbind)

```
# Base de datos (datos)
datos <- data.frame(
  "n" = 1:4,                                # filas
  "indiv." = c("a", "b", "c", "d"),        # Individuos
  "sexo" = c("f", "f", "m", "m"),          # Sexo
  "variable" = c(1.2, 3.4, 4.5, 5.6))      # Valor de la variable

datos    # Ver asinación del data.frame

# Base de datos (continuación)
head(datos)    # Muestra las primeras filas
names(datos)   # Nombres de las columnas
str(datos)     # Estructura de la base de datos
t(datos)      # Transpuesta de la base de datos
```

#### 4.0.5 Enlaces de operaciones en R

[Diapositivas Intro a R](#)

[Diapositivas Operaciones en R](#)

[Trucos en R](#)

[RPubs-Intro](#)

[RPubs-Intermed](#)

[/page\(\)](#)

## 5 Importar y exportar bases de datos (read y write)

### 5.0.1 Complementos requeridos

```
# Librerías requeridas
library(tidyverse)
library(xtable)      # Importar y exportar
library(openxlsx)    # exportar "*.xlsx"
library(readxl)      # Importar y exportar
library(xlsx)        # Importar y exportar "*.xlsx"
```

### 5.0.2 Base de datos del ejercicio

```
# Base de datos (datos)
datos = data.frame (meses = c("enero", "junio", "octubre"),
                    periodos = c("sequía", "lluvias1", "lluvias2"),
                    taxón1 = c(2, 1, 3),
                    taxón2 = c(20, 25, 30),
                    taxón3 = c(4, 4, 4))

datos # Ejecutar la asignación (datos)
```

### 5.0.3 Exportación

```
# Exportar bases de datos como "datos1"

write.csv2(datos, "datos1.csv") # paquete "utils"

write_csv2(datos, "datos1.csv") # paquete "readxl"

write.xlsx(datos, "datos1.xlsx") # paquete "openxlsx" y "xlsx"
```

### 5.0.4 Importación

```
# Importar bases de datos como "datos1"

datos1 <- read.csv2("datos1.csv", row.names = 1) # paquete "utils"
datos1 <- read.csv2(file.choose(), row.names = 1) # paquete "utils"

datos1 <- read_csv2("datos1.csv") # paquete "readxl"
datos1 <- read_csv2(file.choose()) # paquete "readxl"

datos1 <- read_excel("datos1.xlsx") # paquete "readxl"
datos1 <- read_excel(file.choose()) # paquete "readxl"

datos1 <- read.xlsx("datos1.xlsx") # paquete "openxlsx"
datos1 <- read.xlsx(file.choose()) # paquete "openxlsx"
```

### 5.0.5 Importación desde internet

```
# Importar archivo *.csv desde la web
datos2 <- read.csv2("https://javier-2712.github.io/Multivariados/Insectos.csv")

datos2 <- read_csv2("https://javier-2712.github.io/Multivariados/Insectos.csv")
```

### 5.0.6 Enlaces de importación y Exportación de datos

[Importación de datos1](#)

[Importación de datos2](#)

[Diapositivas](#)

Resúmenes con psych

/page()

## 6 Introdutorio a Tidyverse

### 6.0.1 Importar la base de datos (datos)

```
datos1 <- read_excel(file.choose()) # paquete "readxl"
```

### 6.0.2 Manipulación de los datos [enlace](#)

- comando `gather` para visualizar bases de datos alargadas
- comando `spread` para visualizar bases de datos a lo ancho
- comando `%>%` tuberías o pipelines.

```
# Base de datos alargada (datos.1)
datos.1 <- datos %>%
  gather(key= Columnas, value= Valores)
datos.1
```

```
# Excluir la columna periodo en formato alargado (-periodos)
datos.1 <- datos %>%
  gather(key= columnas, value= valores, -periodos)
datos.1
```

### 6.0.3 Nueva base de datos con factor

```
# Base de datos para 4 estudiantes (con 4 replicas)
# a los que se les midieron dos variables en cuatro ocasiones.
datos <- data.frame(n= 1:16,
  Estudiante= c("a","a","a","a","b","b","b","b",
    "c","c","c","c", "d","d","d","d"),
  Sexo= c("f","f","f","f","f","f","f","f",
    "m","m","m","m","m","m","m","m"),
  Variable1= c(1.2,3.4,4.5,5.6,1.2,3.4,4.5,5.6,
```

```

                                0.8,2.4,1.8,1.5,1.6,2.1,1.2,0.8),
Variable2= c(2.4,6.8,9.0,11.2,2.4,6.8,9.0,11.2,
              1.6,4.8,3.6,3.0,3.2,4.2,2.4,1.6))

datos

```

#### 6.0.4 Filtrando elementos del dataframe (filter)

```

# Filtrado por sexos "f" y "m"
datos.f <- datos %>% filter(Sexo == "f")
datos.f # Base de datos para mujeres

datos.h <- datos %>% filter(Sexo == "m")
datos.h # Base de datos para hombres

# Filtrado por sexos y estudiantes "f" y "m"
datos.a <- datos %>% filter(Sexo == "f", Estudiante == "a")
datos.a # Datos de la estudiante a

datos.a <- datos.f %>% filter(Estudiante == "a")
datos.a # Datos de la estudiante a

```

#### 6.0.5 Filtrando en orden descendente o ascendente (arrange)

```

# Filtrando en orden descendente y ascendente
datos.des <- datos %>% arrange(desc(Variable1))
datos.des # Variable asignada

datos.asc <- datos %>% arrange(Variable1)
datos.asc # Variable asignada

```

#### 6.0.6 Combinación de filtrado y orden (filter + arrange)

```

# Filtrar mujeres en orden descendente.
datos.des.f <- datos %>%
  filter(Sexo == "f") %>%
  arrange(desc(Variable1))

```



```
datos.des.f # Asignación
```

### 6.0.7 Generación de variables derivadas (arrange)

```
# Insertar nuevas variables (mutate)
datos.3 <- datos %>%
  mutate(Variable3 = Variable1 * Variable2)
datos.3 # Asignación
```

### 6.0.8 Combinación de filtrado, nuevas variables y orden (filter + mutate + arrange)

```
# Combinación de funciones (filter, mutate, arrange)
datos.4 <- datos %>%
  filter (Sexo == "f") %>%
  mutate (Variable3 = Variable2 * 12) %>%
  arrange (desc(Variable3))
datos.4 # Asignación
```

```
# Combinación de funciones (filter, mutate, arrange)
datos.4 <- datos %>%
  filter (Sexo == "f", Estudiante == "b") %>%
  mutate (Variable3 = Variable2 * 12) %>%
  arrange (desc(Variable3))
datos.4
```

### 6.0.9 Enlaces de operaciones en Tidyverse

[Videos de Tidyverse](#)

[Introducción al Tidyverse](#)

[Introducción al Tidyverse](#)

[Introducción al Tidyverse](#)

[El Tidyverso y tidy](#)

[Curso de Tidyverse](#)

10 funciones de Tidyverse

Manipulación de datos

Estandarización de variables

Transformaciones de variables

/page()

## 7 Taller práctico para la casa

1. Realizar los ejemplos del siguiente [enlace](#), utilizando las siguientes opciones de tidyverse:

- Filter
- Arrange
- Mutate

3. Realizar los ejemplos del siguiente [enlace](#), utilizando las siguientes opciones de tidyverse:

- Pipeline
- summarize
- group\_by
- mutate
- filter
- select
- joins

2. Realizar los ejemplos del siguiente [enlace](#), utilizando las siguientes opciones de tidyverse:

- summarize
- group\_by
- mutate
- filter
- select

3. Realizar los ejemplos del siguiente [enlace](#), utilizando las siguientes opciones de tidyverse:

- Ejemplo de los censos
- spread

## 8 Taller 2.1 Operaciones matriciales

### Objetivo de la actividad:

Poner en práctica operaciones matriciales, para resolver un ejercicio de ordenación multivariada, denominado “**Análisis de Componentes Principales - PCA**“, cuyo objetivo es relacionar a las observaciones de las matrices o de las bases de datos (filas de la matriz), de acuerdo a las variables definidas (columnas de la matriz). Finalmente se realizará la misma técnica con la librería **vegan** de R.

### 8.0.1 1.1 producto matricial

```
# A (2,1,1,3)
# B (1,4,2,5,0,3)
# Calcular: (1) B'.A' (2) (A.B)' (3) Demostrar que B'.A' = (A'.B')

# R./
A = matrix(c(2,1,1,3),2,2,byrow=TRUE)      # Matriz A
B = matrix(c(1,4,2,5,0,3),2,3,byrow=TRUE)   # Matriz B

A
B
```

#### 8.0.1.1 Respuestas

```
# (1) B'.A'
t(B)%*%t(A)      # %*% representa el producto matricial,
# "t" corresponde a la transpuesta de una matriz

# (2) (A.B)'
t(A%*%B)

# (3) B'.A' = (A.B)'
(t(B)%*%t(A)) == t(A%*%B)  # Demostración
```

## 8.0.2 1.2 Determinantes

```
# A (2,3,3,2)
# B (1,4,2,5,0,3)
# Calcular: Determinante de A y de B

#R./
A = matrix (c(2,3,3,2),2,2,byrow=TRUE)
B = matrix (c(1,4,2,5,0,3),3,3,byrow=TRUE)

A
B
```

### 8.0.2.1 Respuestas

```
# Determinantes
det(A)
det(B)
```

## 8.0.3 1.3 Matriz inversa

```
# A (5,2,2,2)
# Calcular inversa de A

# R./
A = matrix(c(5,2,2,2), 2,2, byrow= T)

A
```

### 8.0.3.1 Respuestas

```
# Matriz inversa (solve)
solve(A)
```

#### 8.0.4 1.4 Matriz de varianza - covarianza\*

```
# A (2,2,4,3,6,9), 3 x 2
# B (2:10), 5 x 2

# R./
A = matrix(c(2,2,4,3,6,9),3,2, byrow= T)
B = matrix(c(2:10), 5,2, byrow=T)

A
B
```

##### 8.0.4.1 Respuestas

```
# Covarianzas de cada matriz (c/grupo)
cov.A = cov(A)
cov.B = cov(B)

cov.A
cov.B
```

#### 8.0.5 1.5 Covarianza generalizada (S)

```
cov.g = (3*(cov.A) + 5*(cov.B))/8 # cov.g corresponde a la covarianza generalizada
cov.g
```

##### 8.0.5.1 Respuestas

```
# Covarianza generalizada invertida (S-1)
cov.g.i = solve(cov.g) # cov.g.i representa a la covarianza gralizada invertida
cov.g.i

round(cov.g.i,2) # round corresponde al redondeo de decimales
```

## 9 Aplicación de matrices en un Análisis de Componentes Principales - PCA

### 9.0.1 1. Matriz centrada (m.c) y matriz rotada (m.r)

Pasos:

- Cargar librerías requeridas
- Crear una matriz (o un dataframe)
- Hallar el vector de medias (v.m)
- Hallar la matriz centrada (m.c)
- Hallar la matriz de covarianzas de m.c (s.c)
- Hallar la matriz de autovectores o vectores propios (v.p)
- Hallar la matriz rotada de A (m.r)
- Graficar (stats y ggplot2)
- Comparar con el PCA realizado en el paquete **vegan**

#### 9.0.1.1 2. Cargar las librerías requeridas

```
# Librerías requeridas
library(tidyverse)
library(xtable)      # Importar y exportar
library(openxlsx)    # exportar "*.xlsx"
library(readxl)      # Importar y exportar

library(ggplot2)      # gráfica en ggplot2
library(ggrepel)      # insertar rótulos a los puntos
library(vegan)        # para realizar el pca con vegan
```

### 9.0.1.2 3. Vector de medias (v.m) y matriz centrada (m.c)

```
A <- data.frame(Var1= c(1.2,3.4,4.5,5.6,1.2,3.4,4.5,5.6,
                        0.8,2.4,1.8,1.5,1.6,2.1,1.2,0.8),
                Var2= c(2.4,6.8,9.0,11.2,2.4,6.8,9.0,11.2,
                        1.6,4.8,3.6,3.0,3.2,4.2,2.4,1.6),
                Var3= c(4.4,10.8,19.0,21.2,12.4,16.8,19.0,21.2,
                        11.6,14.8,13.6,13.0,13.2,14.2,12.4,11.6))

A

v.m <- colMeans(A)    # Vector de medias
v.m

m.c <- t(t(A) - v.m) # Centralización de datos = Matriz centrada
round(m.c , 2)
```

### 9.0.1.3 4. Vectores propios (v.p)

```
m.c <- t(t(A) - v.m) # Matriz centrada
round(m.c , 2)

s.c <- var(m.c)      # Covarianza de la matriz centrada
round(s.c , 2)

vv.p <- eigen(s.c)   # Vectores y valores propios de m.centrada
round(vv.p, 2)

v.p <- vv.p$vectors  # Matriz de vectores propios
round(v.p, 2)
```

### 9.0.1.4 5. Matriz Rotada (m.r)

```
m.c <- as.matrix(m.c) # Matriz centrada (m.c)
round(m.c, 2)

m.r <- m.c %*% v.p     # Matriz rotada (m.r)
round(m.r, 2)
```



```
A <- data.frame (n= 1:16, A) # matriz A como dataframe
round(A, 2)
```

#### 9.0.1.5 6. figura (paquete - stats)

```
x11()
plot(m.r[,1:2],
      xlab="PC1", ylab="PC2")

text(m.r,                # Rótulos de los datos (filas)
      labels = row.names(A),
      pos=3, cex=0.7)
```

#### 9.0.1.6 7. figura

```
m.r <- data.frame(m.r) # matriz rotada como dataframe

x11()
ggplot(m.r, aes(x= X1 ,y= X2)) +
  geom_point() +
  geom_text_repel (aes (label = A$n)) +
  geom_hline(yintercept=0,linetype=2,size=1) +
  geom_vline(xintercept=0,linetype=2,size=1)
```

# 10 Análisis de Componentes principales - PCA en librería "vegan"

**Objetivo:** Comparar los resultados del PCA anterior con los generados por un paquete o librería de R

## 10.0.0.1 1. Realización del PCA

```
library(vegan)          # Librería requerida
A
head(A[,2:4])           # Variables y observaciones (columnas y filas)
pca <- rda(A[,2:4])     # Realización del pca
```

## 10.0.0.2 2. Figura del PCA

```
x11()
biplot(pca)             # Figura del pca
```

# 11 Operaciones matriciales - base *Caimanes*

## Objetivo de la actividad:

Poner en práctica operaciones matriciales, para resolver un ejercicio de ordenación multivariada, denominado “**Análisis de Componentes Principales - PCA**“, cuyo objetivo es relacionar a las observaciones de las matrices o de las bases de datos (filas de la matriz), de acuerdo a las variables definidas (columnas de la matriz). Finalmente se realizará la misma técnica con la librería **vegan** de R. La base de datos a utilizar se presenta en dos formatos: **caimanes.csv** y **caimanes.xlsx**. Esta base cuenta con 3 variables morfométricas (columnas) y 17 individuos evaluados (filas).

## 11.1 Procedimiento del PCA

- Cargar librerías requeridas
- Cargar la base **datos** (usar diferentes opciones para practicar)
- Hallar el vector de medias (v.m) y la matriz centrada (m.c)
- Hallar la matriz de covarianzas de m.c (s.c)
- Hallar la matriz de autovectores o vectores propios (v.p)
- Hallar la matriz rotada de A (m.r)
- Graficar (stats y ggplot2)
- Comparar con el PCA realizado en el paquete **vegan**

### 11.1.0.1 1. Cargar las librerías requeridas

```
# Librerías requeridas
library(tidyverse)
library(xtable)      # Importar y exportar
library(openxlsx)    # exportar "*.xlsx"
library(readxl)      # Importar y exportar
```

```
library(ggplot2)      # gráfica en ggplot2
library(ggrepel)      # insertar rótulos a los puntos
library(vegan)        # para realizar el pca con vegan
```

### 11.1.0.2 2. Cargar o importar la base de datos

```
#-----
datos <- read_excel("caimanes.xlsx")    # paquete "readxl"
head(datos)

datos <- read_csv2("caimanes.csv")      # paquete "readxl"
head(datos)

datos <- read.csv2("caimanes.csv")      # paquete "utils"
head(datos)
```

- Ajustar la base de datos

```
# Resumir los rótulos de las columnas
colnames(datos) <- c("ID", "Sexo", "LT", "CD", "CS")    # Rótulos de la base de datos
head(datos)      # Base de datos abreviada
str(datos)
```

### 11.1.0.3 3. Hallar el vector de medias (v.m) y la matriz centrada (m.c)

```
v.m <- colMeans(datos[,3:5])    # Vector de medias
v.m

m.c <- t(t(datos[,3:5]) - v.m) # Centralización de datos = Matriz centrada
round(m.c , 2)
```

### 11.1.0.4 4. Vectores propios (v.p)

```
s.c <- var(m.c)      # Covarianza de la matriz centrada
round(s.c , 2)

vv.p <- eigen(s.c)    # Vectores y valores propios de m.centrada
```

```
round(vv.p, 2)

v.p <- vv.p$vectors # Matriz de vectores propios
round(v.p , 2)
```

#### 11.1.0.5 5. Matriz Rotada (m.r)

```
m.c <- as.matrix(m.c) # Matriz centrada (m.c)
round(m.c, 2)

m.r <- m.c %*% v.p      # Matriz rotada (m.r)
round(m.r, 2)

A <- data.frame (n= 1:16, A) # matriz A como dataframe
round(A, 2)
```

#### 11.1.0.6 6. figura (paquete - stats)

```
x11()
plot(m.r[,1:2],
      xlab="PC1", ylab="PC2")

text(m.r,          # Rótulos de los datos (caimanes)
      labels = row.names(datos),
      pos=3, cex=0.7)

abline(v=0, lty=2,col= "blue")
abline(h=0, lty=2,col= "blue")
```

#### 11.1.0.7 7. figura)

```
x11()
m.r <- data.frame(m.r) # matriz rotada como data.frame

ggplot(m.r, aes(x= X1 ,y= X2)) +
  geom_point() +
  geom_text_repel (aes (label = datos$ID)) +
```

```
geom_hline(yintercept=0,linetype=2,size=1) +  
geom_vline(xintercept=0,linetype=2,size=1)
```

### 11.1.1 Análisis de Componentes principales - PCA en librería "vegan"

**Objetivo:** Comparar los resultados del PCA anterior con los generados por un paquete o librería de R

#### 11.1.1.1 1. Realización del PCA

```
# Comparar con el Análisis de Componentes Principales - pca  
head(datos[,3:5])          # Variables y observaciones (caimanes)  
pca <- rda(datos[,3:5])    # Realización del pca
```

#### 11.1.1.2 2. Figura del PCA

```
x11()  
biplot(pca)
```

# 12 Taller 3. Exploración Univariada y Multivariada

## Objetivo de la actividad:

Poner en práctica el manejo de bases de datos y la visualización de datos uni, bi, tri y multivariados, para responder principalmente a dos tipos de objetivos:

1. **Relaciones** entre variables biológicas y de estas con las ambientales (ej. figuras de elipses, pares, dispersión y coplot).
2. **Diferencias** para el caso en el que contemos con variables agrupadoras (factores o v. cualitativas), orientado a evaluar las diferencias entre variables biológicas en gradientes espaciales o temporales (ej. entre grupos de sitios).

La base de datos a utilizar se presenta en dos formatos: **Insectos.csv** e **Insectos.xlsx**. Esta base cuenta con 2 variables ambientales y 6 biológicas, así como con un factor o variable agrupadora (cuencas), todo esto distribuido en las columnas. Además cuenta con y 20 localidades o quebradas (filas).

## 12.1 Procedimiento de la exploración

- Cargar librerías requeridas
- Cargar la base **Insectos** (usar diferentes opciones para practicar)
- Explorar al **objetivo 1** (figuras de elipses, pares, dispersión y coplot).
- Explorar al **objetivo 2** (figuras de elipses, pares, dispersión y coplot).
- Realizar las opciones gráficas relacionadas a los objetivos.
- Realizar transformaciones de los datos, para mejorar la visualización de patrones.
- Practicar con leyendas y resultados de la visualización realizada.

### 12.1.0.1 Cargar las librerías requeridas

```
# Librerías requeridas
library(tidyverse)
library(xtable)      # Importar y exportar
library(openxlsx)    # exportar "*.xlsx"
library(readxl)      # Importar y exportar

library(stats)        # Para las figuras de pares
library(lattice)      # No se requiere instalar
library(ggplot2)      # gráfica en ggplot2
library(ggrepel)      # insertar rótulos a los puntos
require(SciViews)     # Fig. dispersión con coef. de pearson
library(plotrix)      # Figuras de cajas con múltiples variables
library(reshape)      # Figuras de cajas con múltiples variables
library(corrplot)     # Figuras de elipses
library(gridExtra)    # Para figuras estadísticas (varios factores)
library(grid)         # Para figuras estadísticas (varios factores)
```

**Nota:** ggcorrplot2 requiere instalarse de la siguiente manera, debido a que está en proceso de ajuste para las nuevas plataformas de R. [ver\\_enlace\\_procedimiento](#)

```
# Instalar "ggcorrplot2", solo por una vez
install.packages("remotes")
remotes::install_github("caijun/ggcorrplot2")
```

- **Nota:** gganimate requiere instalarse de la siguiente manera: [ver\\_enlace\\_procedimiento](#)

```
install.packages('gganimate')
devtools::install_github('thomasp85/gganimate')
```

### 12.1.0.2 Cargar o importar la base de datos

```
#-----
datos <- read.csv2("Insectos.csv")      # paquete "utils"
head(datos)
```

	quebrada	cuenca	pH	temp	Efem	Plec	Tric	Dipt	Cole	Ab
1		1	cuen1	6.8	17.4	26	4	9	30	3 72
2		4	cuen1	7.3	16.8	17	6	9	25	1 58



3	11	cuen1	5.6	16.0	9	3	28	24	3	67
4	13	cuen1	6.3	17.8	2	3	25	21	6	57
5	19	cuen1	5.6	18.2	6	4	24	12	13	59
6	3	cuen2	6.3	17.0	7	2	25	10	1	45

### 12.1.1 1. Figuras de elipses

El Paquete **corrplot** es el que permite realizar las opciones gráficas de elipses a color, ingresar a este enlace:

[corrplot](#)

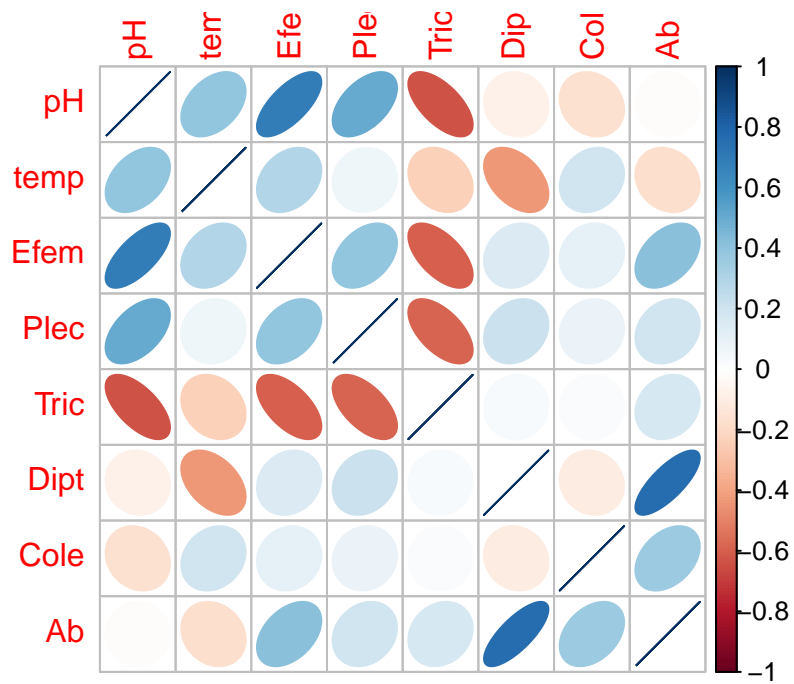
Otro enlace a **corrplot**:

[corrplot](#)

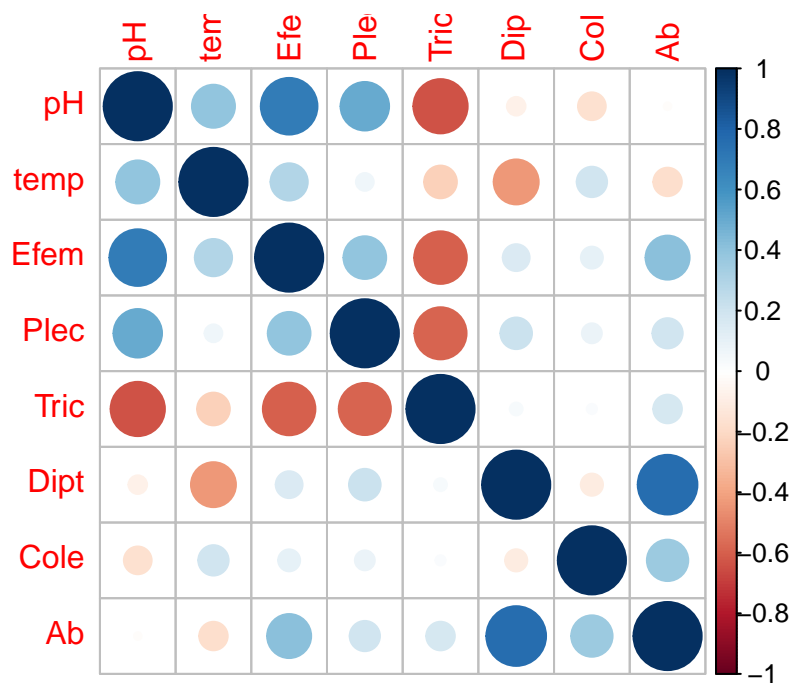
```
#---
# 1. Elipses
library(corrplot)

# Elipses con colores
M <- cor(datos[,3:10]) # Matriz de Correlación (M)

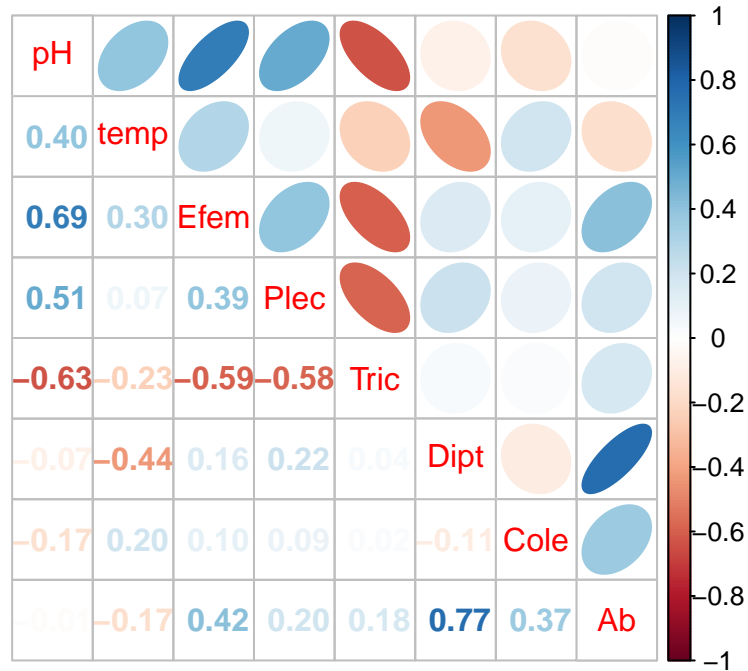
# Elipses con colores
x11() # Panel gráfico adicional
corrplot(M, method = "ellipse") # Figura de correlaciones con elipses
```



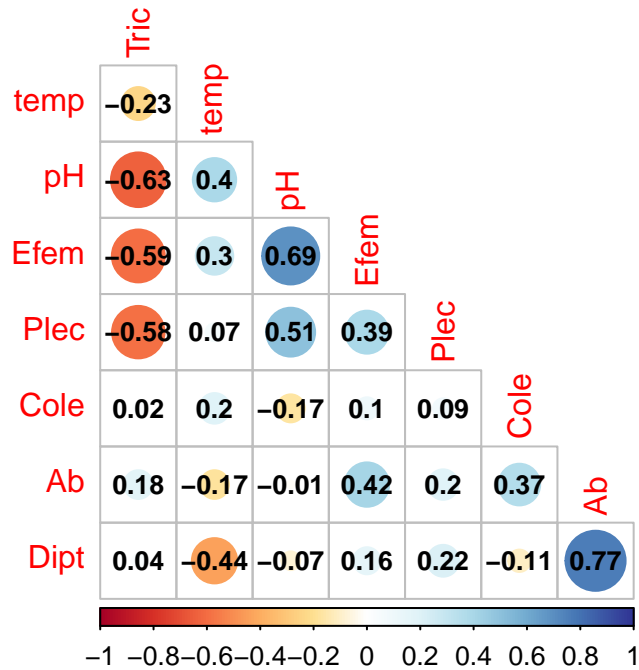
```
# Elipses con colores
X11()
corrplot(M, method = "circle") # Figura de correlaciones con circulos
```



```
# Elipses con colores
X11()
corrplot.mixed(M, upper="ellipse")
```



```
# Figura de elipses con coeficientes de correlación
x11()
corrplot(M, method = "circle",           # Correlaciones con circulos
          type = "lower", insig="blank",  # Forma del panel
          order = "AOE", diag = FALSE,    # Ordenar por nivel de correlación
          addCoef.col = "black",           # Color de los coeficientes
          number.cex = 0.8,               # Tamaño del texto
          col = COL2("RdYlBu", 200))      # Transparencia de los circulos
```



### 12.1.2 3. Figuras de Dispersión por pares de variables (pairs)

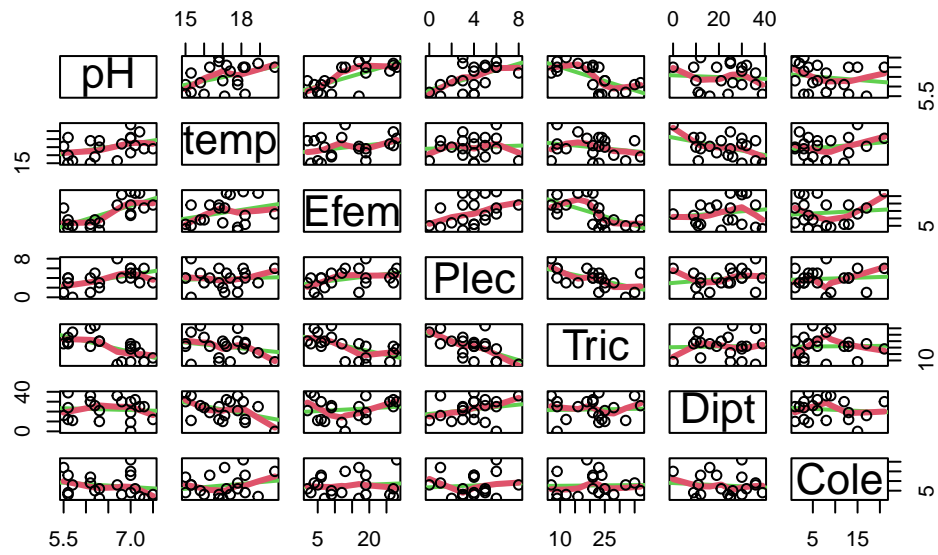
El Paquete **pairs** es el que permite realizar las opciones gráficas de dispersión por parejas de variables, ingresar a este enlace:

[pairs](#)

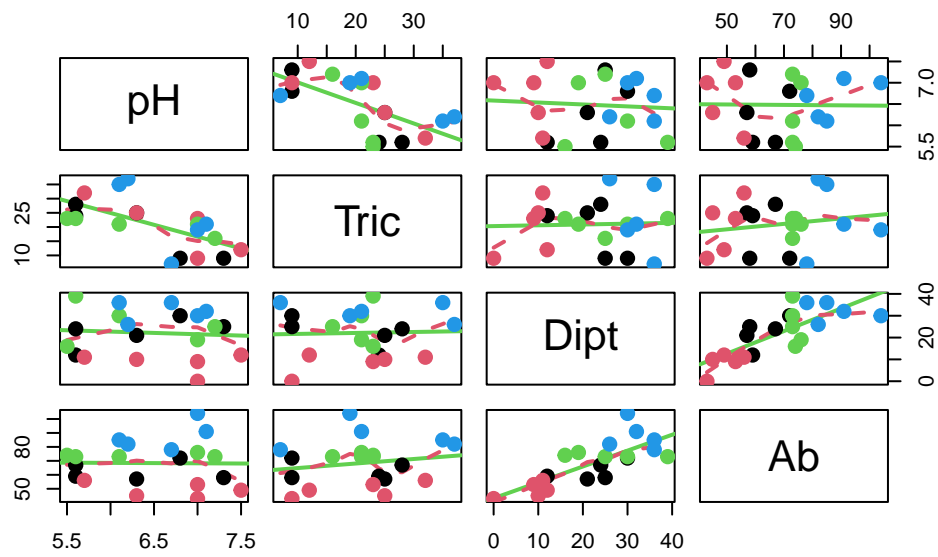
Otro enlace a **pairs**:

[pairs](#)

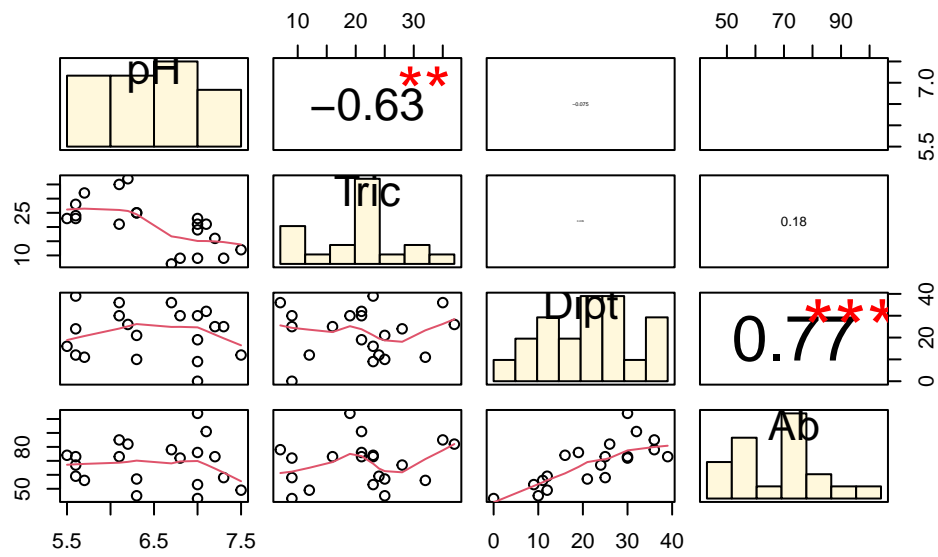
```
# Figuras de pares
x11()
pairs ((datos[,c(3:9)]),panel=function(x,y)
{abline(lsfit(x,y)$coef,lwd=2,col=3)      # lwd = Ancho de la línea
  lines(lowess(x,y),lty=1,lwd=3,col=2)    # col= Color de la línea
  points(x,y,cex=1)})
```



```
# Incluir el factor (cuenca)
# **requiere a cuenca como factor
datos$cuenca =as.factor(datos$cuenca)
pairs ((datos[,c(3,7,8,10)]),panel=function(x,y)
{abline(lsfit(x,y)$coef,lwd=2,col=3)
  lines(lowess(x,y),lty=2,lwd=2,col=2)
  points(x,y,col=datos$cuenca, cex=1.4,pch=19)}})
```

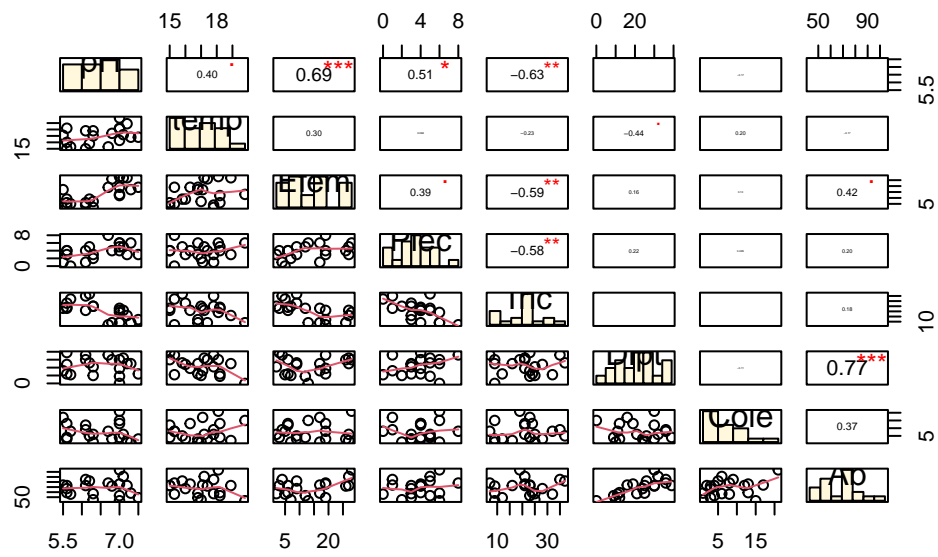


```
# Correlaciones de Pearson
library(SciViews)
x11()
pairs(datos[,c(3,7,8,10)], diag.panel = panel.hist,
      upper.panel = panel.cor, lower.panel = panel.smooth)
```



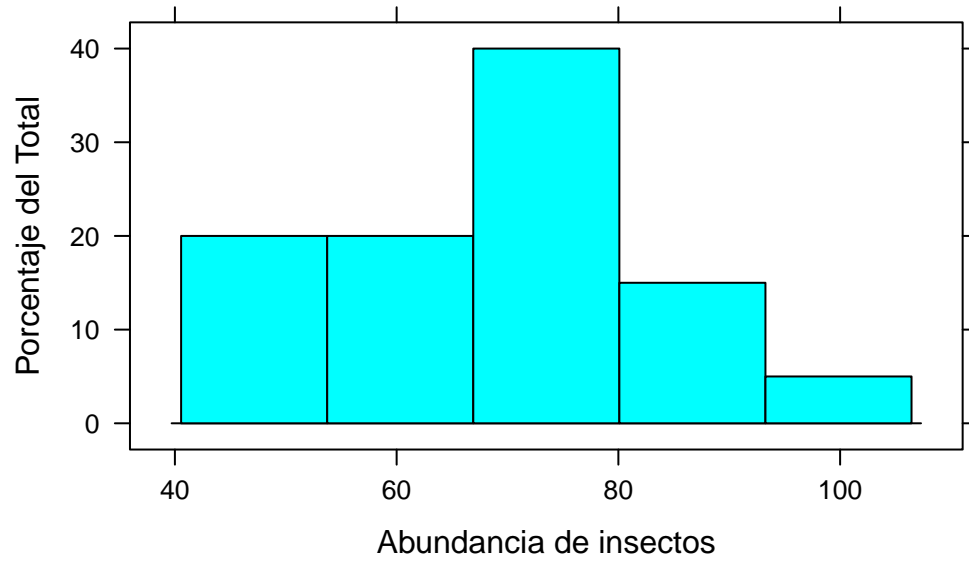
```
# Incluir histogramas
pairs(datos[, 3:10], diag.panel = panel.hist,
      upper.panel = panel.cor, lower.panel = panel.smooth)
```



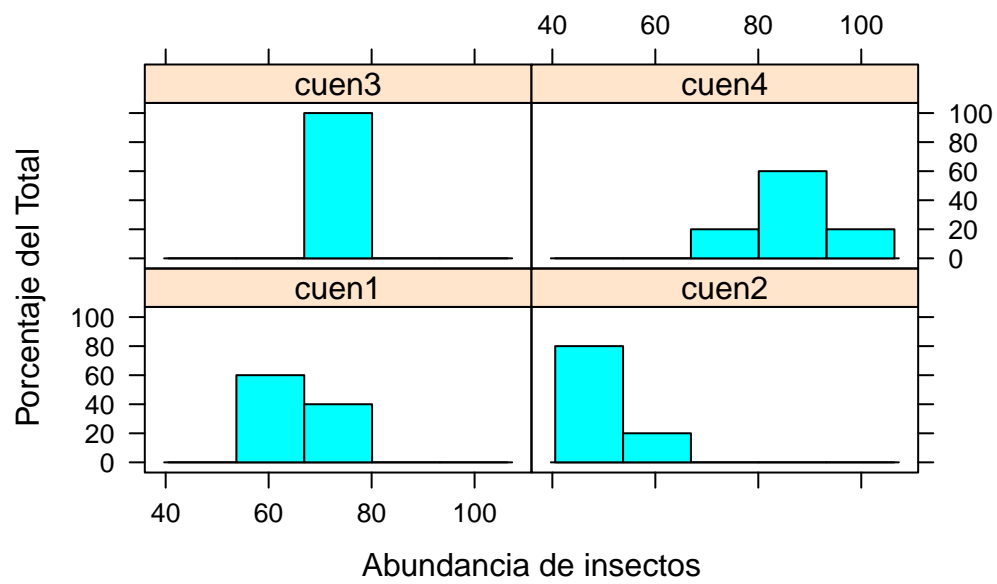


### 12.1.3 3. Histogramas

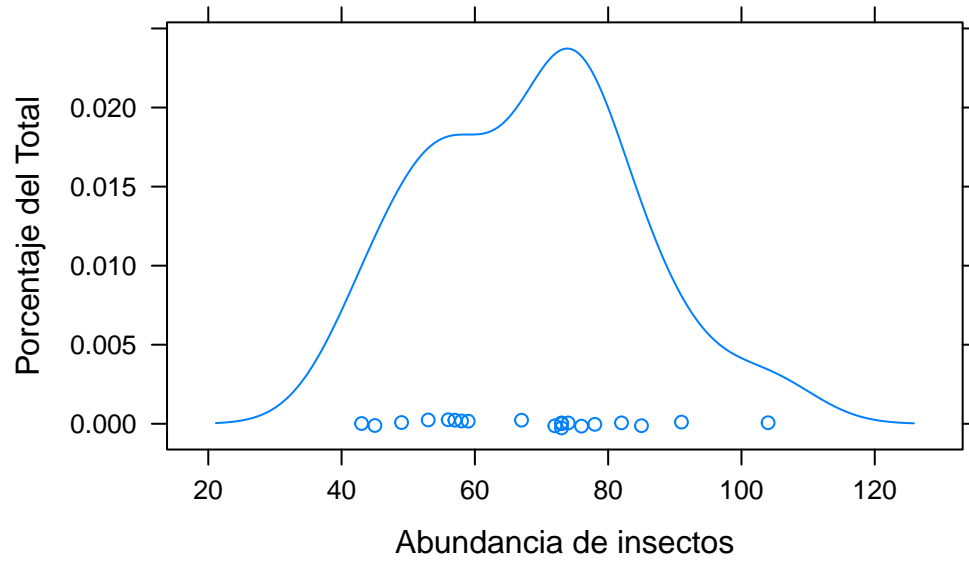
```
# Frecuencias de abundancias
histogram(~Ab, data=datos, ylab="Porcentaje del Total",
          xlab="Abundancia de insectos")
```



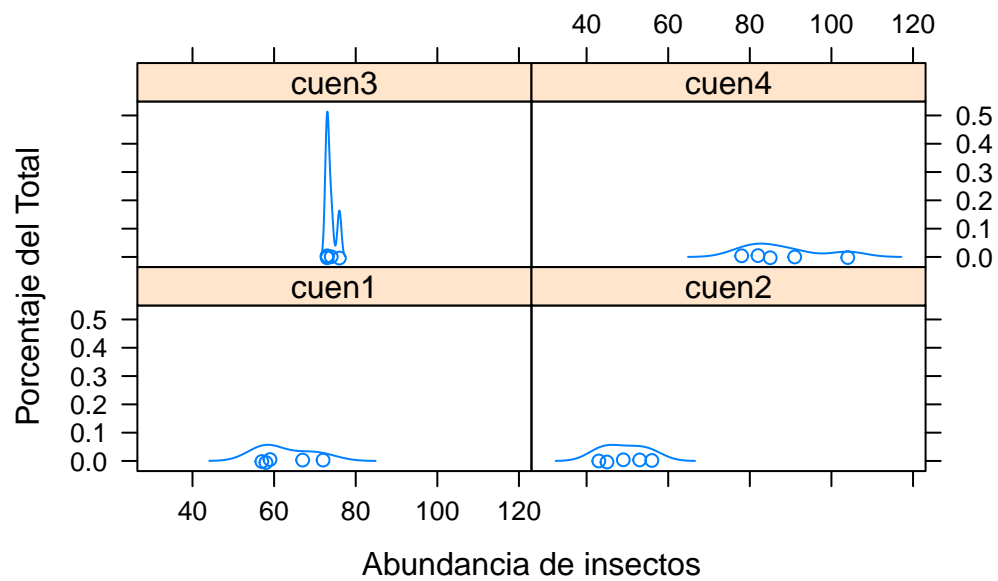
```
# Frecuencias de abundancias por cuencas  
histogram (~Ab|cuenca,data=datos, ylab="Porcentaje del Total",  
           xlab="Abundancia de insectos")
```



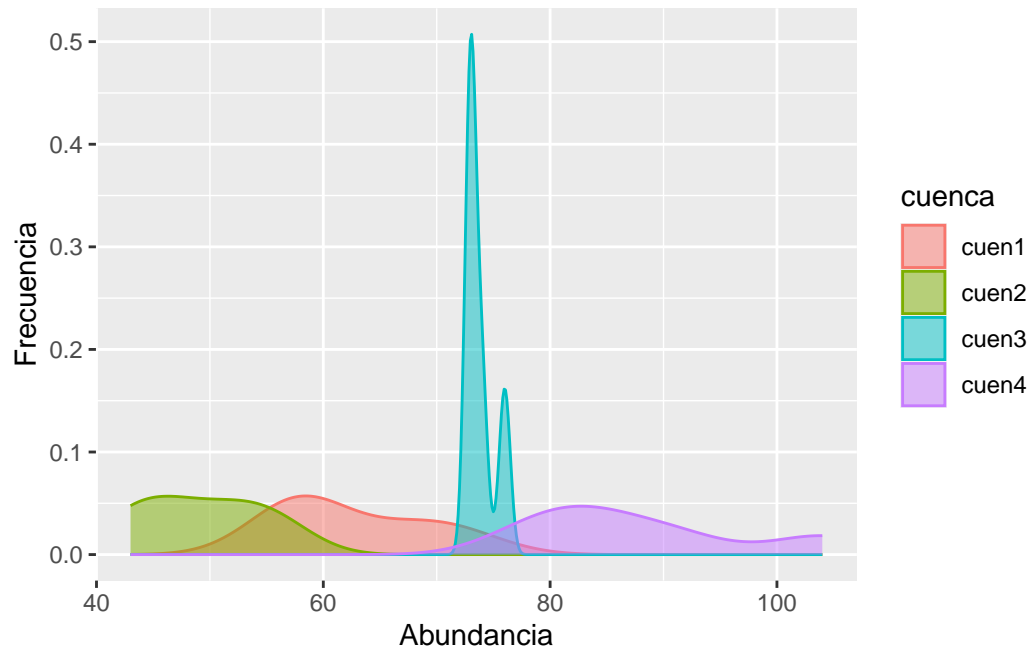
```
# Frecuencias por densidad
densityplot(~Ab,data=datos, ylab="Porcentaje del Total",
            xlab="Abundancia de insectos")
```



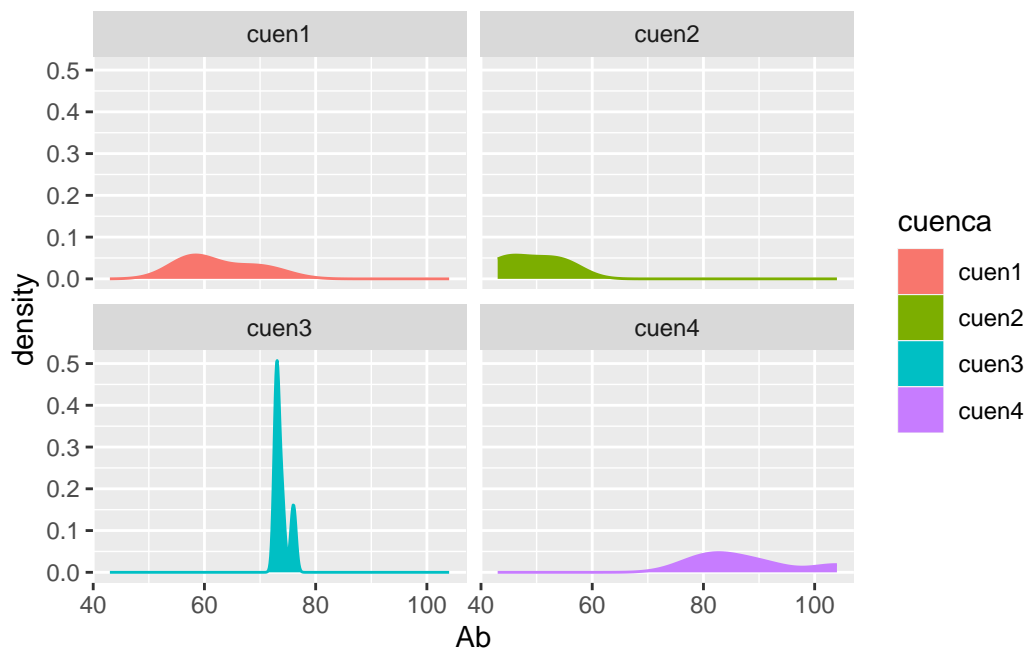
```
densityplot(~Ab|cuenca,data=datos, ylab="Porcentaje del Total",
            xlab="Abundancia de insectos")
```



```
# Frecuencias de abundancias por densidad
ggplot(data = datos, aes(x = Ab, color = cuenca)) +
  geom_density(aes(fill = cuenca), alpha = 0.5) +
  labs( y="Frecuencia", x="Abundancia")
```



```
# Otra opción
ggplot(data = datos, aes(x = Ab, color = cuenca)) +
  geom_density(aes(fill = cuenca)) +
  facet_wrap(~ cuenca)
```



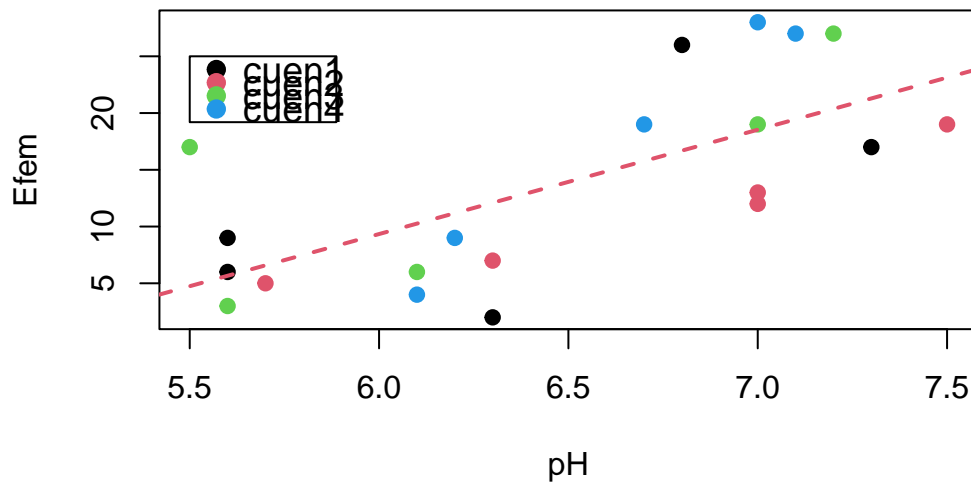
#### 12.1.4 4. Dispersión X-Y

El Paquete **lattice** es uno de los que permite realizar las opciones gráficas de dispersión, ingresar a este enlace:

**lattice**

```
# Regresión lineal (esquema básico)
library(lattice)
datos$cuenca <- as.factor(datos$cuenca) # cuenca como factor

x11()
plot(Efem~pH,                                # Relación pH vs. Efem
     col=as.integer(cuenca),                 # Colores por tipo de cuencas
     data=datos, pch=19)                     # Base de datos
legend(5.5,25,                                # Coordenadas de la leyenda
      legend=levels(datos$cuenca),           # Grupos de la leyenda
      pch=19,col=1:4,cex=1.2)
lines(abline(lm(datos$Efem~datos$pH), # regresión lineal
            lwd=2,col=2, lty=2))
```



#### 12.1.4.1 Dispersión X-Y con ggplot2

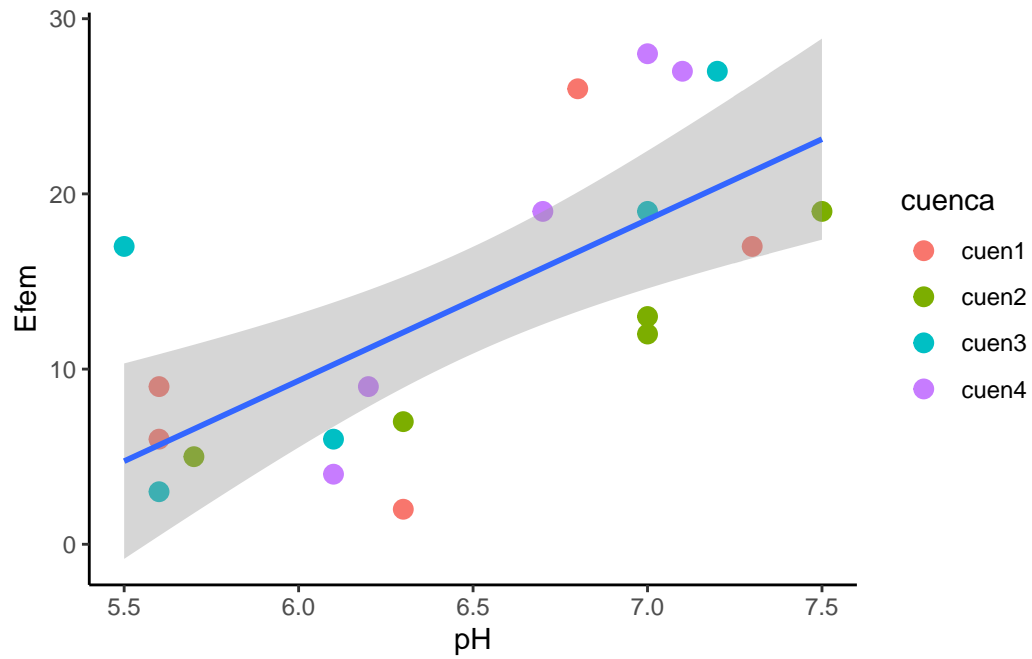
El Paquete **ggplot2** es el que permite realizar las opciones gráficas de dispersión bivariados, ingresar a este enlace:

[RPubs.](#)

Otro enlace a **ggplot2**:

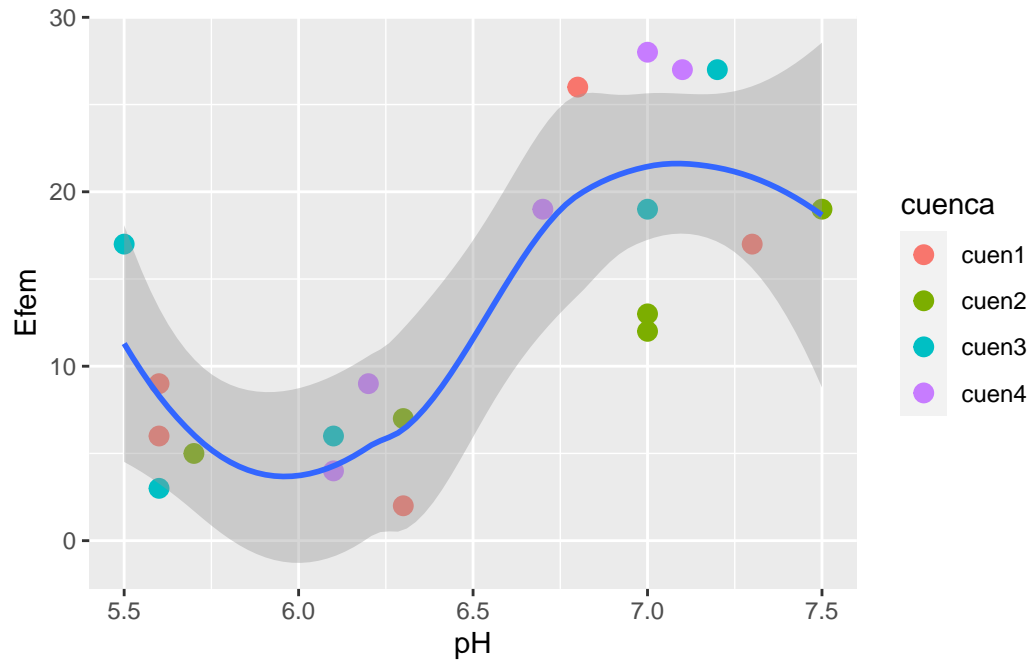
[ggplot2](#)

```
# Regresiones lineales (Esquema ggplot2)
ggplot(datos,aes(x = pH,y = Efem)) +
  geom_point(aes(color = cuenca), size = 3) +
  geom_smooth(method= "lm") +
  theme_classic()
```

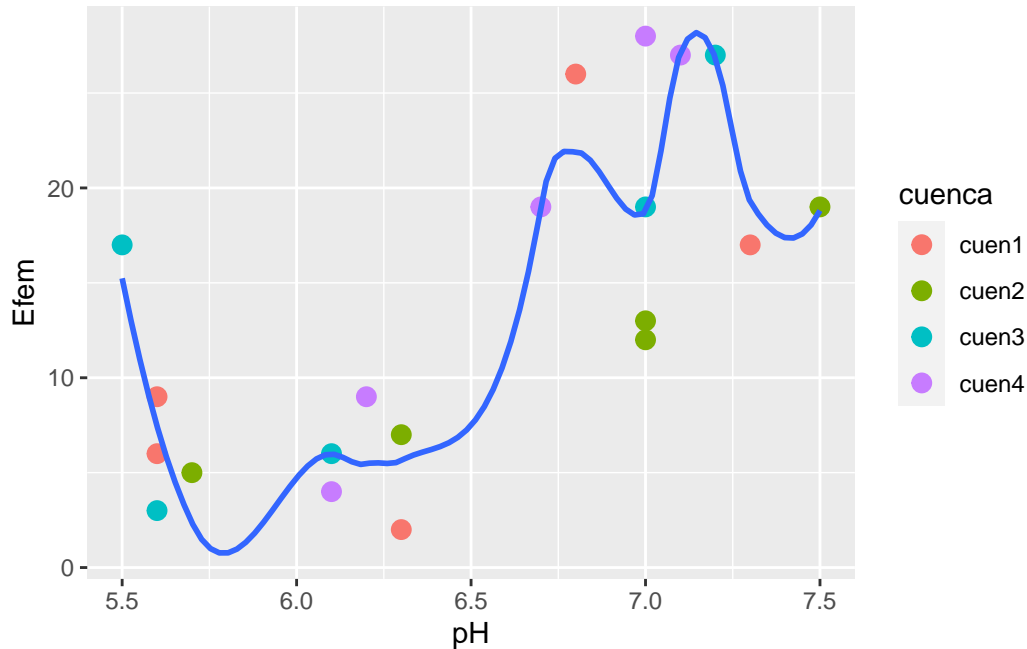


```
# Regresiones suavizadas - Loess o Lowess (Esquema ggplot2)
ggplot(datos,aes(x = pH, y = Efem)) +
  geom_point(aes(color = cuenca), size = 3) +
  geom_smooth()
```





```
# Regresiones suavizadas (Loess)
ggplot(datos,aes(x = pH, y = Efem)) +
  geom_point(aes(color = cuenca), size = 3) +
  geom_smooth(se = FALSE, span = 0.4)
```



### 12.1.5 5. Cajas y Bigotes

El Paquete **lattice** es uno de los que permite realizar las opciones gráficas de cajas, ingresar a este enlace:

[How to make a boxplot in R](#)

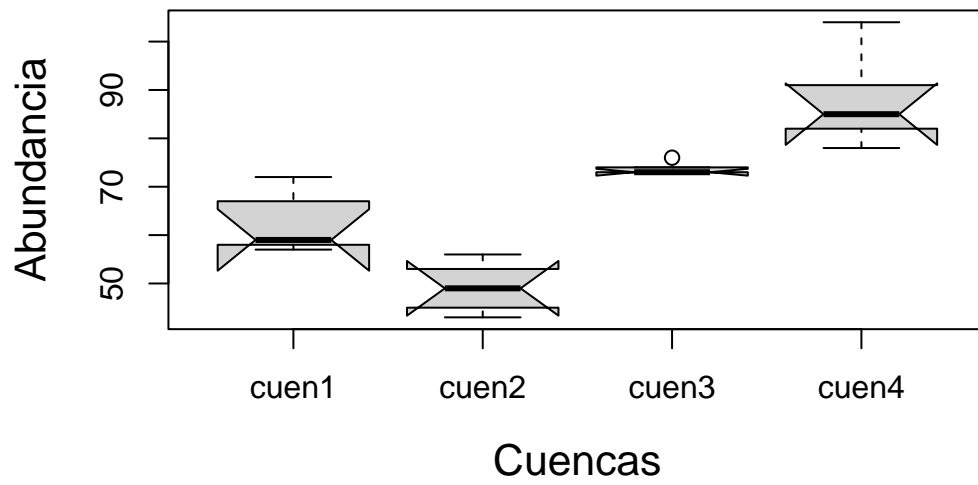
El Paquete **ggplot2** presenta opciones más estéticas y robustas para estas y muchas más figuras, ingresar a este enlace:

[Enlace1](#)

[Enlace2](#)

```
# Figuras de Cajas y bigotes
datos$cuenca<-factor(datos$cuenca,
                      levels=c("cuen1","cuen2","cuen3","cuen4"))

# Cajas y Bigotes con muescas
boxplot(Ab~cuenca,data=datos,notch=TRUE,
        xlab="Cuencas",ylab="Abundancia",
        col="lightgray", cex.lab=1.3) # Probar con otros colores
```

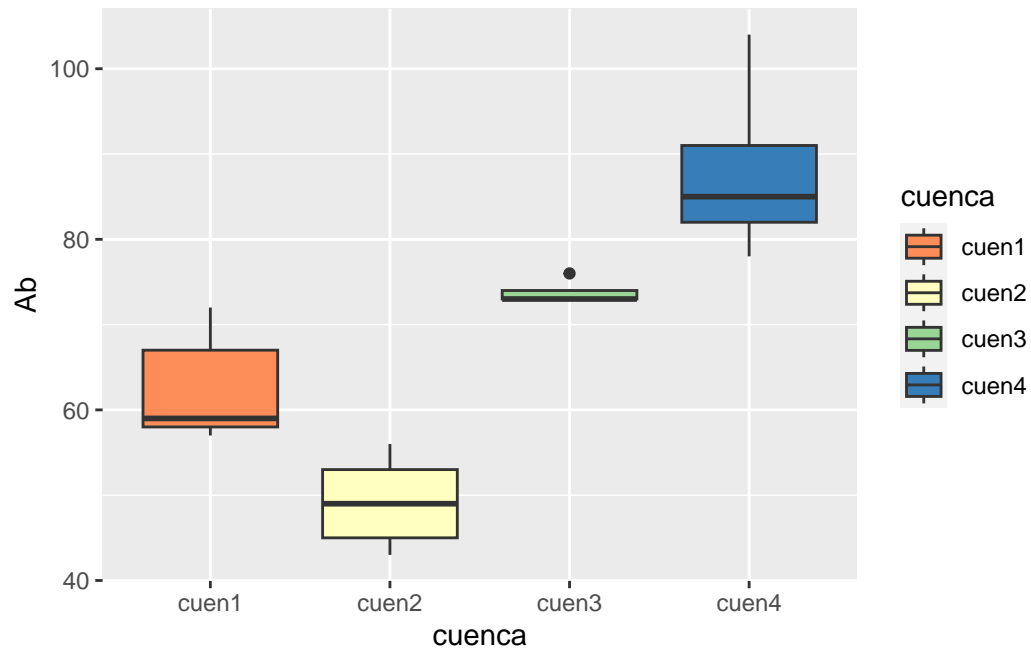


Enlaces de **paletas de colores** para la edición de las figuras:

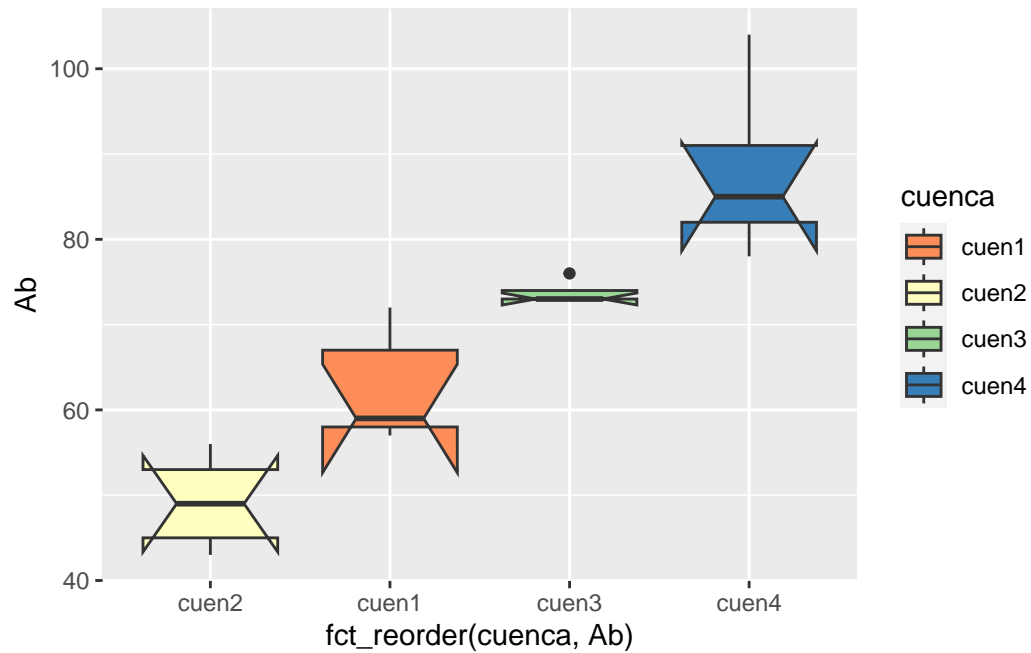
[colorbrewer](#)

[colors](#)

```
# Buscar en google: colorbrewer2
ggplot(datos, aes(x=cuenca, y=Ab)) +
  geom_boxplot(aes(fill = cuenca)) +
  scale_fill_manual(values = c('#fc8d59', '#ffffbf', '#99d594', '#377eb8'))
```

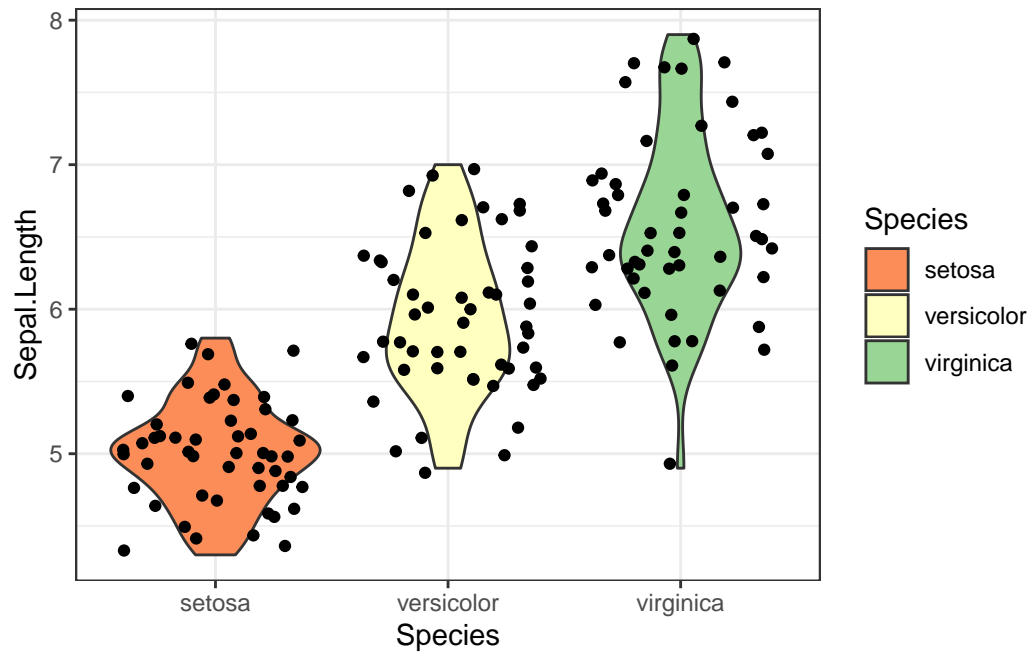


```
# Organización por nivel de magnitud
ggplot(datos, aes(x = fct_reorder(cuenca, Ab), y=Ab)) +
  geom_boxplot(notch = T, aes(fill = cuenca)) +
  scale_fill_manual(values = c('#fc8d59', '#ffffbf', '#99d594', '#377eb8'))
```



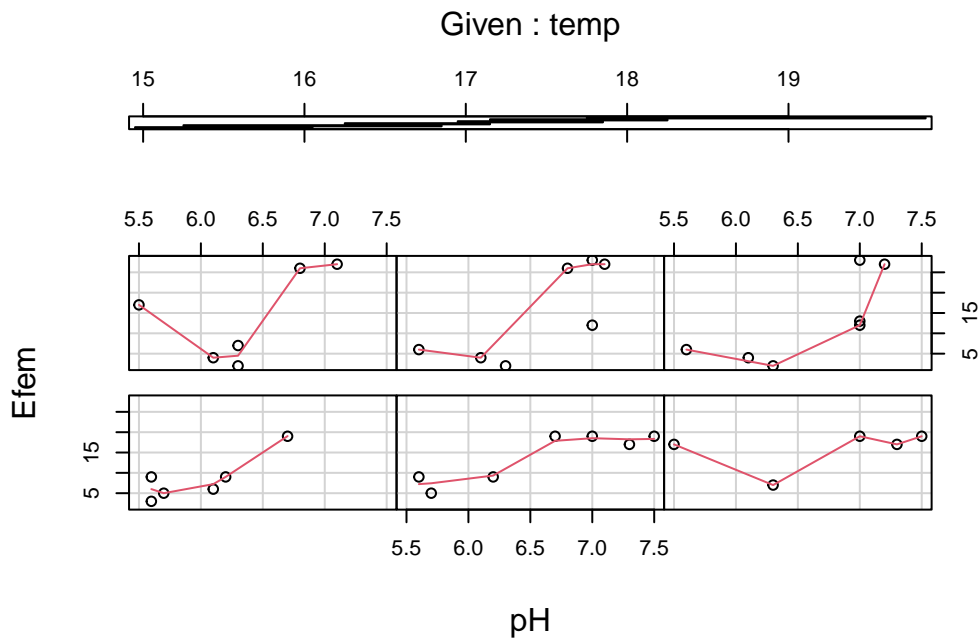
**Paréntesis** Base de datos de lirios (iris) para figura de violín

```
# violin: como histograma acostado
ggplot(iris, aes(x = Species, y = Sepal.Length)) +
  geom_violin(aes(fill = Species)) +
  geom_jitter() +
  scale_fill_manual(values = c('#fc8d59', '#ffffbf', '#99d594')) +
  theme_bw()
```



### 12.1.6 6. Coplot

```
# Coplot con líneas de ajuste suavizado (loess)
with(datos, {
  coplot(Efem~pH|temp,
    panel = panel.smooth)})
```



```
# Categorización de dos variables continuas (pH y Temp)
summary(datos[,2:8])
```

cuenca	pH	temp	Efem	Plec
cuen1:5	Min. :5.50	Min. :15.00	Min. : 2.00	Min. :0.00
cuen2:5	1st Qu.:6.00	1st Qu.:15.95	1st Qu.: 6.00	1st Qu.:3.00
cuen3:5	Median :6.50	Median :17.05	Median :12.50	Median :4.00
cuen4:5	Mean :6.48	Mean :16.99	Mean :13.75	Mean :3.85
	3rd Qu.:7.00	3rd Qu.:17.88	3rd Qu.:19.00	3rd Qu.:5.00
	Max. :7.50	Max. :19.80	Max. :28.00	Max. :8.00

Tric	Dipt
Min. : 7.00	Min. : 0.00
1st Qu.:15.00	1st Qu.:12.00
Median :22.00	Median :24.50
Mean :20.95	Mean :22.15
3rd Qu.:25.00	3rd Qu.:30.00
Max. :37.00	Max. :39.00

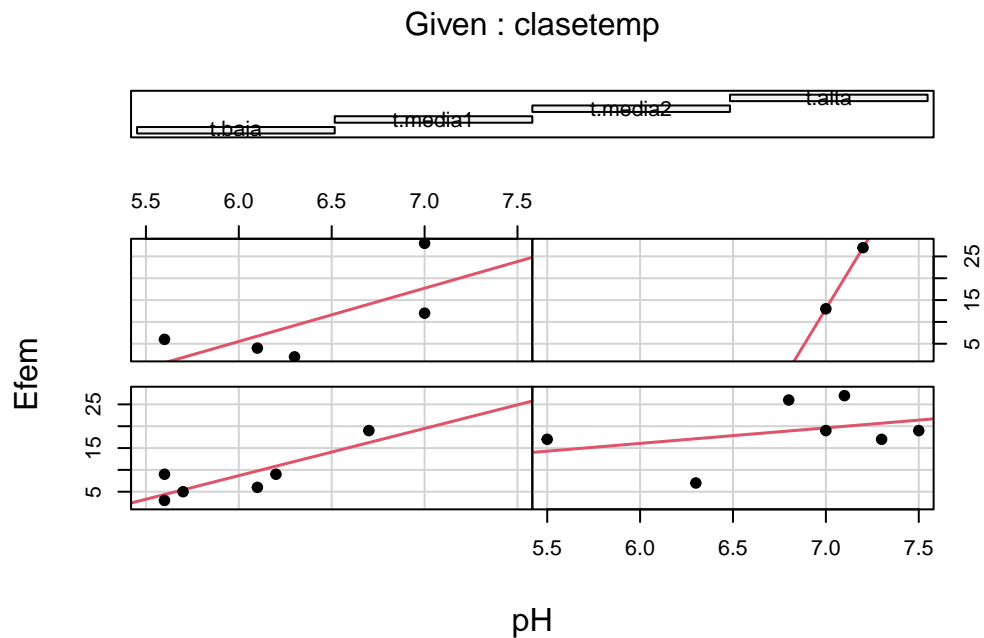
```
clasetemp <- cut(datos$temp,seq(15,20,1.2),include.lowest=T,
  labels = c("t.baja", "t.media1","t.media2", "t.alta"))
```

```
clasepH <- cut(datos$pH,seq(5,8,1),include.lowest=T,
               labels = c("pH.bajo", "pH.medio","pH.alto"))
```

```
# Función para el coplot
panel.lm = function(x, y, ...) {
  tmp<-lm(y~x,na.action=na.omit)
  abline(tmp, lwd = 1.5, col= 2)
  points(x,y, ...)}

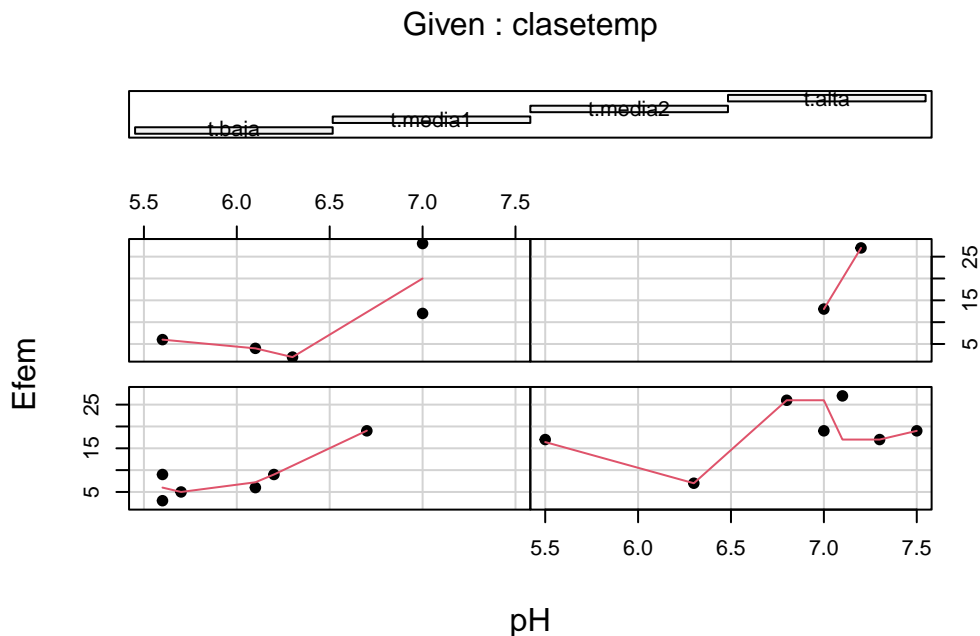
```

```
# Relación trivariada - Lineal
coplot(Efem~pH | clasetemp, pch=19,
       panel = panel.lm, data=datos)
```



```
coplot(Efem~pH | clasetemp, pch=19,
       panel = panel.smooth, data=datos)
```





### 12.1.7 7. Figuras con estadísticos (promedios, errores, ...)

El Paquete **ggplot2** es uno de los que permite realizar las opciones gráficas de barras con estadísticos, ingresar a este enlace:

#### Exploratory Data Analysis with ggplot

Existen otros enlaces en los que se puede encontrar información complementaria para figuras de barras, como los siguientes

#### ggplot2 barplots

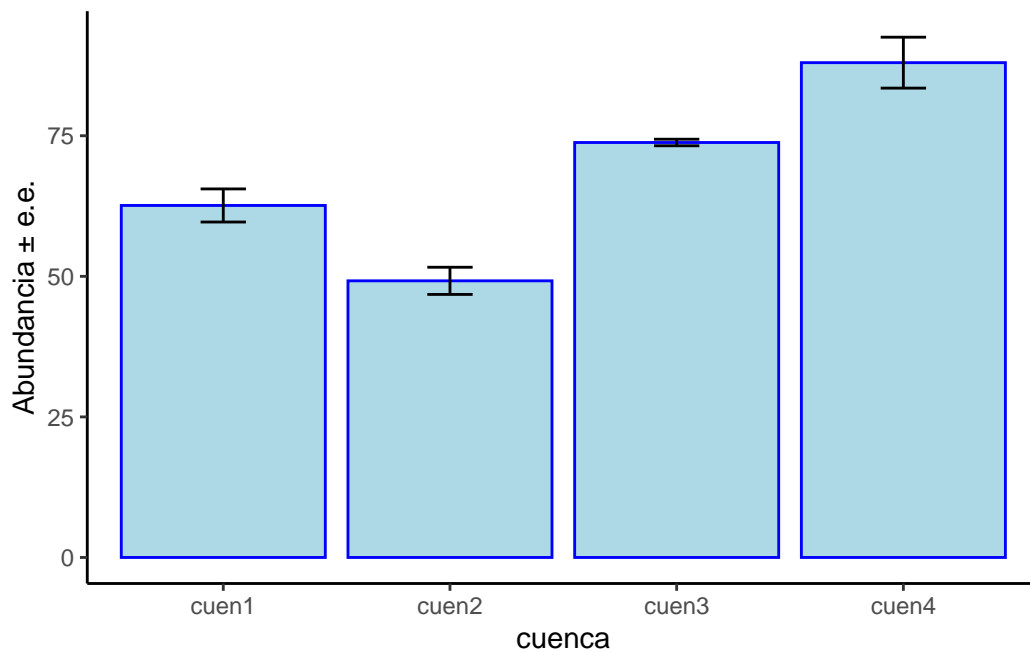
#### Stunning Bar Charts

```
# Resumen estadístico "datos_resum"
datos_resum <- datos %>%           # Base de datos resumida
  group_by(cuenca) %>%             # Factor o variable agrupadora
  summarise(datos.m = mean(Ab),    # Media de cada grupo del factor
            datos.de = sd(Ab),     # Desviaciones estándar de cada grupo
            datos.var = var(Ab),   # Varianzas de cada grupo
            n.Ab = n(),            # Tamaño de cada grupo
            datos.ee = sd(Ab)/sqrt(n())) # Error estándar de cada grupo
datos_resum
```

```
# A tibble: 4 x 6
  cuenca datos.m datos.de datos.var n.Ab datos.ee
  <fct>    <dbl>    <dbl>    <dbl> <int>    <dbl>
1 cuen1    62.6     6.58     43.3     5     2.94
2 cuen2    49.2     5.40     29.2     5     2.42
3 cuen3    73.8     1.30      1.7     5     0.583
4 cuen4    88      10.1    102.     5     4.53
```

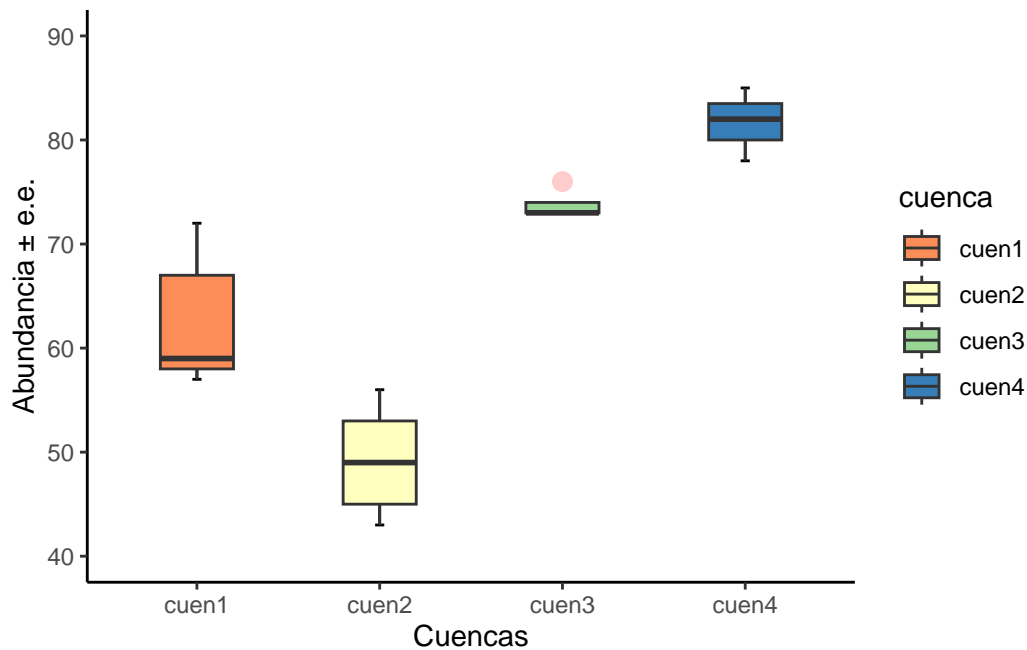
```
# Figura de promedios y errores estándar
DatosPlot<-
  ggplot(datos_resum, aes(cuenca, datos.m, dev.off())) +
  geom_bar(stat="identity", col="blue", fill="lightblue") +
  geom_errorbar(aes(ymin=datos.m-datos.ee,
                    ymax=datos.m+datos.ee),width=0.2)

# Imprimir la figura de promedios con errores estándar
print (DatosPlot +
  labs(y="Abundancia ± e.e.",
        x = "cuenca") +
  scale_fill_manual(values= c("#A1D5D5")) +
  theme_classic())
```



**Nota:** Para las figuras de **Cajas y Bigotes**, el comando `stat_boxplot(geom = "errorbar",...)` permite realizar gráficas sin necesidad de extraer algunos estadísticos previamente.

```
# Opción de cajas y bigotes con errores estándar
ggplot(datos, aes(x=cuenca, y= Ab, fill= cuenca)) +
  stat_boxplot(geom = "errorbar",width = 0.05) +
  geom_boxplot(width = 0.4,
    notchwidth = 0.9, outlier.colour="red",
    outlier.fill="red",
    outlier.size=3, outlier.alpha = 0.2) +
  theme_classic() +
  scale_fill_manual(values=c('#fc8d59','#ffffbf','#99d594','#377eb8')) +
  labs(x = "Cuencas", y = "Abundancia ± e.e.") +
  scale_y_continuous(limits = c(40,90))
```



### 12.1.7.1 Base de datos con múltiples factores

```
# Base de datos multifactorial (insectos1)
datos1<-read_csv2("Insectos2.csv")      # Formato *xlsx
head(datos1)  # Encabezado
```

```
# A tibble: 6 x 6
  No Muestreo GF   Lluvia   Ab  Biom
<dbl> <chr>    <chr> <chr> <dbl> <dbl>
1     1 M1      C-F  P1      98 56.0
2     2 M2      C-F  P1     198 52.7
3     3 M3      C-F  P2      45 11.4
4     4 M4      C-F  P2      51 25.3
5     5 M5      C-F  P2       3  0.36
6     6 M6      C-F  P2      69 23.6
```

```
# Resumen estadístico "datos_resum"
datos_resum <- datos1 %>%      # Base de datos resumida
  group_by(Lluvia,GF) %>%      # Factor o variable agrupadora
  summarise(datos.m = mean(Biom), # Media de cada grupo del factor
            datos.de = sd(Biom),  # Desviaciones estándar de cada grupo
            datos.var = var(Biom), # Varianzas de cada grupo
            n.Biom = n(),          # Tamaño de cada grupo
            datos.ee = sd(Biom)/sqrt(n())) # Error estándar de cada grupo

datos_resum
```

```
# A tibble: 10 x 7
# Groups:   Lluvia [2]
  Lluvia GF   datos.m datos.de datos.var n.Biom datos.ee
<chr> <chr>    <dbl>    <dbl>    <dbl>  <int>    <dbl>
1 P1    C-F      37.0     30.0     902.    3      17.3
2 P1    C-R      31.7     32.1    1029.    3      18.5
3 P1    D      361.     120.   14411.    3      69.3
4 P1    R      53.5     69.8    4873.    3      40.3
5 P1    T      190.     296.   87533.    3     171.
6 P2    C-F      14.4     10.2     105.    5       4.58
7 P2    C-R      88.5     115.   13273.    5      51.5
8 P2    D      176.     94.9    9010.    5      42.4
9 P2    R      21.9     12.9     165.    5       5.75
10 P2   T      151.     223.  49655.    5     99.7
```

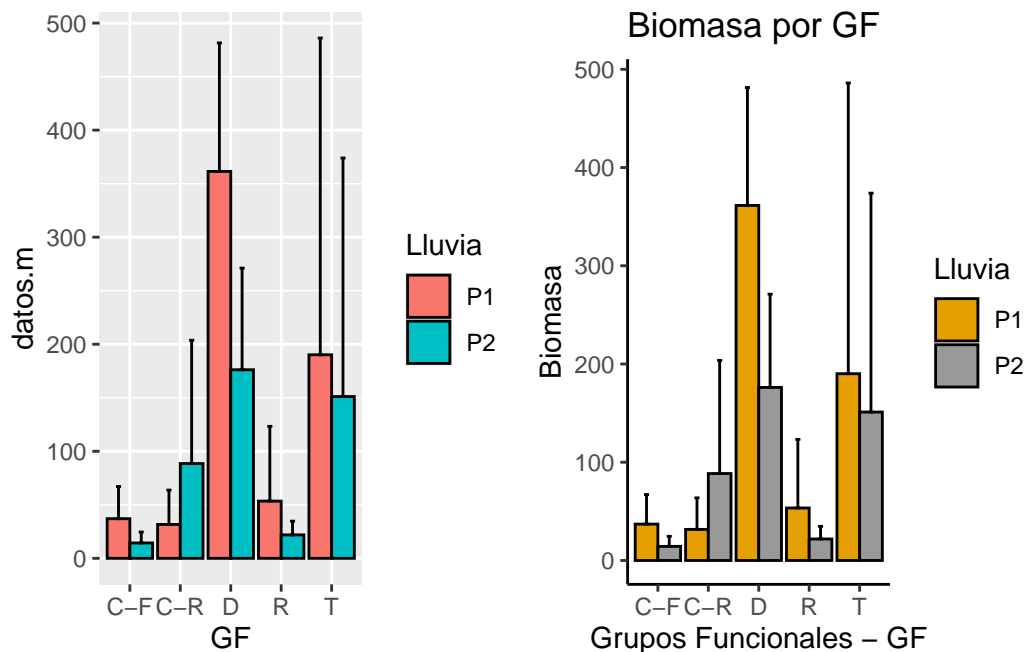
```

# Figura 1 (f1)
f1 = ggplot(datos_resum, aes(x=GF, y=datos.m, fill=Lluvia)) +
  geom_bar(stat="identity", color="black",
           position=position_dodge()) +
  geom_errorbar(aes(ymin=datos.m, ymax=datos.m+datos.de), width=.2,
               position=position_dodge(.9))

# f2: Otro formato de figura bifactorial - theme_classic
f2 = f1+labs(title="Biomasa por GF",
             x="Grupos Funcionales - GF",
             y = "Biomasa")+
  theme_classic() +
  scale_fill_manual(values=c('#E69F00', '#999999'))

# Impresión de un panel con las dos figuras (p1 y p2)
grid.arrange (f1, f2, ncol=2)

```

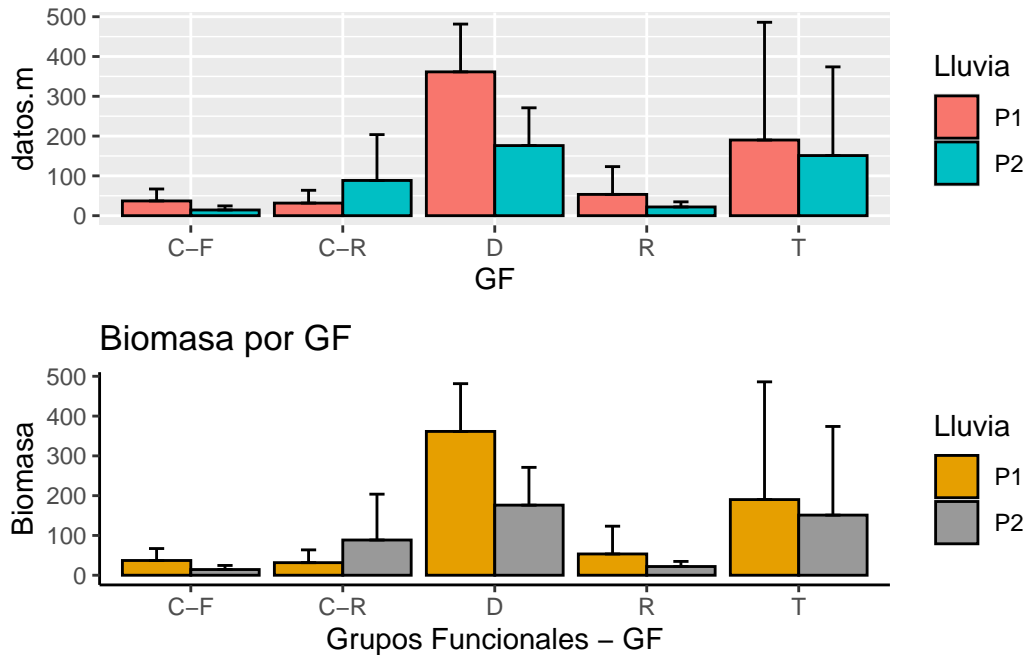


```

# Inserción de las figuras en columna (figuras p1 y p2)
f3 <- ggplotGrob(f1)
f4 <- ggplotGrob(f2)
g <- rbind(f3, f4, size="first")

```

```
g$widths <- unit.pmax(f3$widths, f4$widths)
grid.newpage()
grid.draw(g)
```



### 12.1.7.2 parentesis Figuras de dispersión animadas

[Enlace](#)

```
library(ggplot2)
library(gganimate)
theme_set(theme_bw()) # Tema o fondo de la figura por default

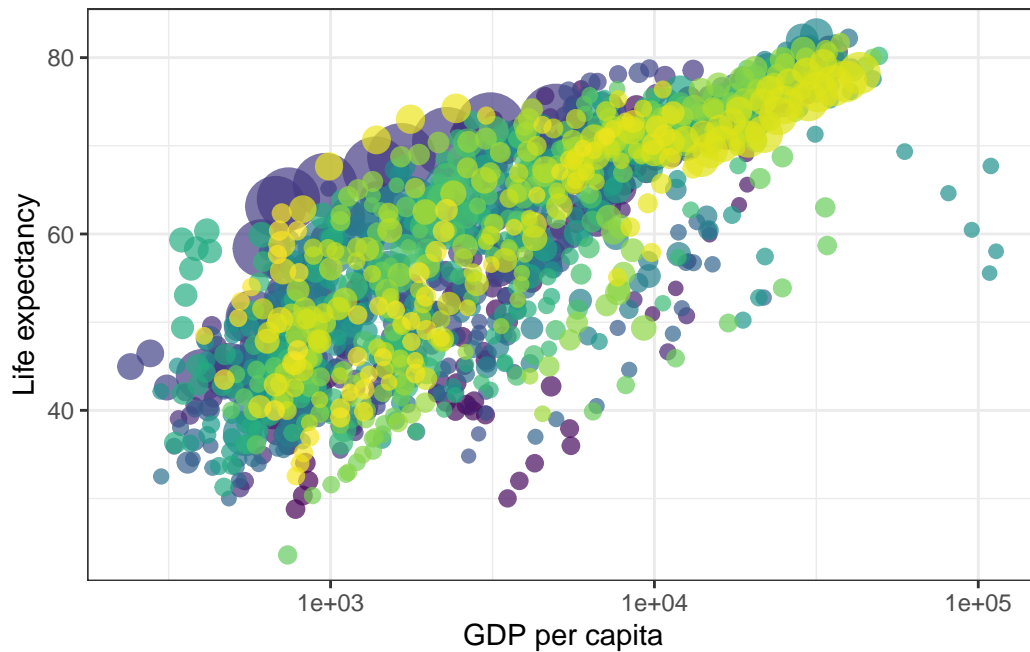
# Demo
library(gapminder)

p <- ggplot(
  gapminder,
  aes(x = gdpPercap, y=lifeExp, size = pop, colour = country)
) +
  geom_point(show.legend = FALSE, alpha = 0.7) +
```

```

scale_color_viridis_d() +
scale_size(range = c(2, 12)) +
scale_x_log10() +
labs(x = "GDP per capita", y = "Life expectancy")
p

```



```

p + transition_time(year) +
  labs(title = "Year: {frame_time}")

```

## 12.2 Taller de entrenamiento

**Objetivo:** Poner en práctica los conceptos vistos en el módulo de exploratorios multivariados, realizando las siguientes opciones gráficas en las bases de datos asignadas para los estdios de caso:

1. Figuras de elipses
2. Figuras de Dispersión por pares de variables (pairs)
3. Histogramas

4. Dispersión X-Y
5. Cajas y Bigotes
6. Coplot
7. Figuras con estadísticos (promedios, errores, ...)



# 13 Taller 4.1 Análisis de Componentes Principales - PCA

El siguiente ejemplo relaciona a 7 lugares en playas de de Santa Marta (observaciones) y en cada una de ellas se midieron 7 variables ambientales (descriptores). En este análisis de comonentes principales - PCA, se intenta saber cuál es la relación entre variables ambientales y cómo estas estructuran o caracterizan a las localidades estudiadas. La base de datos a trabajar es `FQmarino.csv`.

Ejercicio tomado de: Rodríguez-Barrios (2023) [Enlace del libro](#)

[Enlace de los archivos del libro](#)

- [Sigatoka en cultivos de banano - Aguirre et al. \(2015\)](#). Análisis de componentes principales con el paquete “dudipca” y algunas técnicas multivariadas complementarias.
- [Métodos de componentes principales en R - STHDA](#)
- [Artículos - Métodos de componentes principales - STHDA](#)
- [PCA en factoextra - datanovia](#)
- [Guía práctica sobre el PCA - datanovia](#)
- [PCA para variables categóricas - R-bloggers](#)
- [Capítulo PCA - Libro Numerical Ecology with R - Borcard et al. 2018](#)

```
# LIBRERÍAS REQUERIDAS
library(factoextra) # Para el PCA
library(rlang)      #
library(FactoMineR) # Para el PCA
library(vegan)      # Para el PCA
library(ade4)       # Para el PCA
library(corrplot)   # Figuras de elipses
library(ggplot2)    # Figuras de dispersión

# Lectura de la base de datos "FQmarino"
datos <- read.csv2("FQmarino.csv", row.names=1) # file.choose()
```

```
View(datos)
str(datos)
```

```
'data.frame':  7 obs. of  8 variables:
 $ Sitio      : chr  "S1" "S1" "S1" "S1" ...
 $ pH         : num  8.42 8.49 8.51 8.56 8.61 ...
 $ Cond       : num  38 38.1 37.8 37.3 37.3 ...
 $ Turbidez   : num  1.364 0.545 1.273 1.273 0.636 ...
 $ Temp       : num  29.5 29.5 29.6 29.3 29.3 ...
 $ Salinidad  : num  2.42 2.43 2.42 2.38 2.38 ...
 $ CapaFotica: num  19.7 22.1 22.1 10.8 9 ...
 $ Oxigeno    : num  0.097 0.147 0.331 0.17 0.098 0.098 0.098
```

## 14 Exploración Gráfica

```
# Elipses con colores
M <- cor(datos[,2:8])          # Matriz de Correlación (M)
round(head(M),2)
```

	pH	Cond	Turbidez	Temp	Salinidad	CapaFotica	Oxigeno
pH	1.00	-0.27	-0.04	-0.68	0.37	-0.77	-0.38
Cond	-0.27	1.00	0.21	0.68	-0.19	0.61	0.12
Turbidez	-0.04	0.21	1.00	0.03	-0.16	0.01	0.26
Temp	-0.68	0.68	0.03	1.00	-0.03	0.97	0.59
Salinidad	0.37	-0.19	-0.16	-0.03	1.00	0.02	-0.15
CapaFotica	-0.77	0.61	0.01	0.97	0.02	1.00	0.56

La Figura 22.1 muestra la relación entre las variables, a partir de figuras de elipses.

```
x11()                                # Panel gráfico adicional
corrplot(M, method = "ellipse")      # Figura de correlaciones con elipses
```

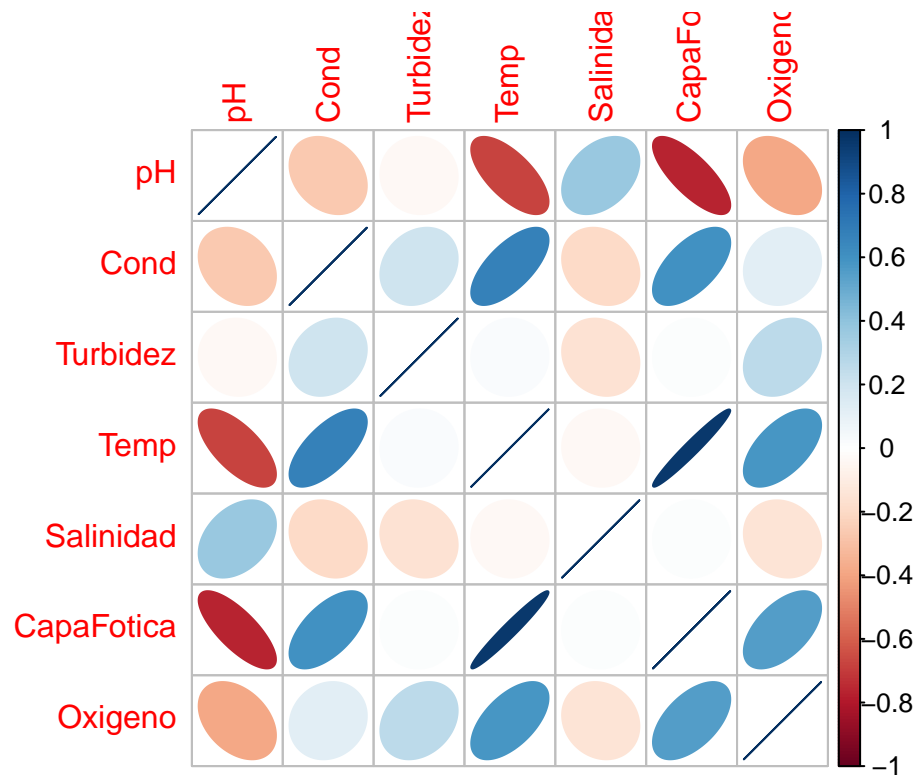


Figura 14.1: Relación de variables ambientales en las siete bahías estudiadas.

La Figura 23.1 muestra la relación entre las variables, a partir de figuras de elipses y coeficientes de correlación de Pearson.

```
X11()
corrplot.mixed(M, upper="ellipse") # Figura con coeficientes de correlación
```

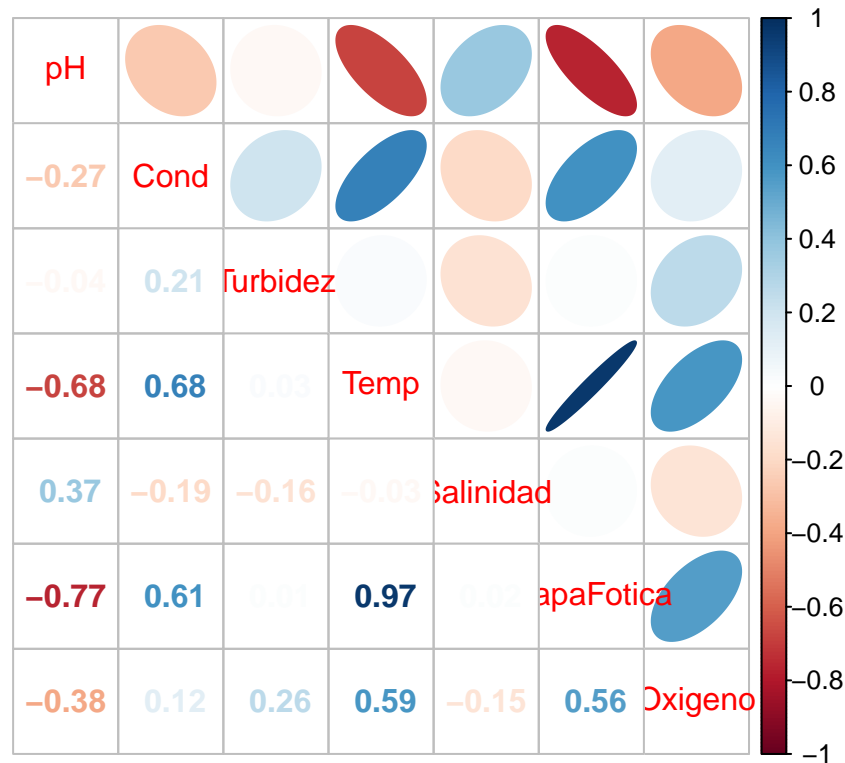


Figura 14.2: Relación de variables ambientales en las siete bahías estudiadas.

La Figura 24.1 otra forma de mostrar la relación entre las variables, a partir de figuras de elipses y coeficientes de correlación de Pearson.

```
x11()
corrplot(M, method = "circle",           # Correlaciones con círculos
          type = "lower", insig="blank",  # Forma del panel
          order = "AOE", diag = FALSE,    # Ordenar por nivel de correlación
          addCoef.col = "black",           # Color de los coeficientes
          number.cex = 0.8,                # Tamaño del texto
          col = COL2("RdYlBu", 200))      # Transparencia de los círculos
```

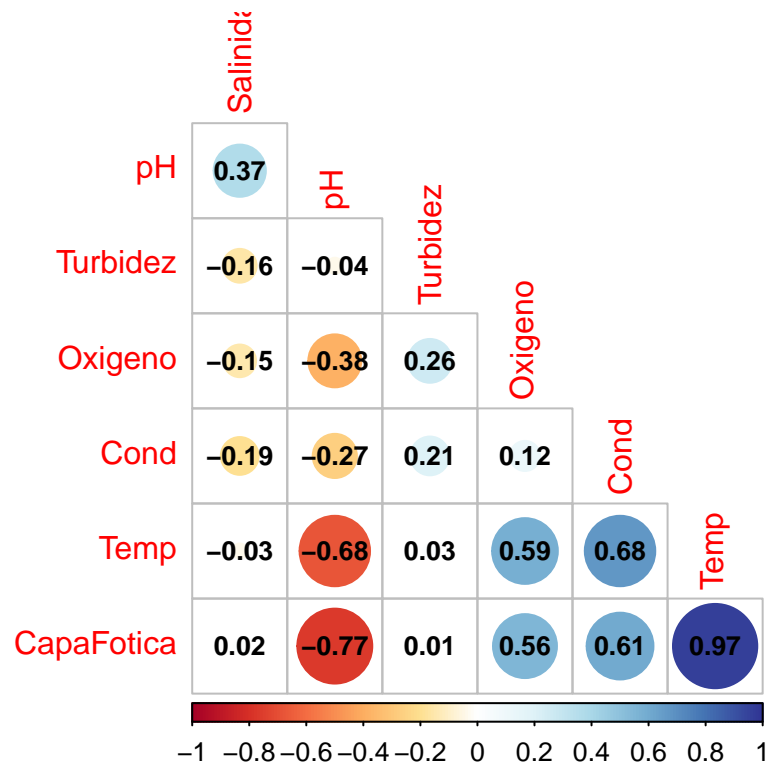


Figura 14.3: Relación de variables ambientales en las siete bahías estudiadas.

## 15 1) PCA con el paquete stats (pca1)

```
pca1 <- princomp(datos[,2:8],cor=TRUE)
```

### 1.1) Valores propios - autovalores para medir el ajuste del PCA

```
summary(pca1)
```

Importance of components:

	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5
Standard deviation	1.8454563	1.1063340	0.9919952	0.9474050	0.68078731
Proportion of Variance	0.4865298	0.1748536	0.1405792	0.1282252	0.06621019
Cumulative Proportion	0.4865298	0.6613834	0.8019626	0.9301878	0.99639798

	Comp.6	Comp.7
Standard deviation	0.158789715	0
Proportion of Variance	0.003602025	0
Cumulative Proportion	1.000000000	1

### 1.2) Insumos del pca (names)

```
names(pca1)
```

```
[1] "sdev"      "loadings" "center"    "scale"     "n.obs"     "scores"    "call"
```

### 1.3) Valores propios - autovectores y escores

```
round(pca1$loadings,2) # Autoectores (loadings)
```

Loadings:

	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7
pH	0.43		0.42	0.18	0.61	0.20	0.45
Cond	-0.36		0.16	0.74	0.27	-0.35	-0.32

Turbidez		-0.65	0.62		-0.41	0.11	
Temp	-0.52	0.21				0.81	-0.10
Salinidad	0.13	0.67	0.60	-0.11	-0.29	-0.13	-0.26
CapaFotica	-0.52	0.25			-0.13	-0.29	0.75
Oxigeno	-0.35	-0.15	0.24	-0.63	0.53	-0.25	-0.22

	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7
SS loadings	1.003	1.006	1.010	0.992	1.003	0.994	1.002
Proportion Var	0.143	0.144	0.144	0.142	0.143	0.142	0.143
Cumulative Var	0.143	0.287	0.431	0.573	0.716	0.858	1.001

```
round(pca1$scores,2)      # Coordenadas de las localidades (Scores)
```

	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7
BTag	-1.55	-0.34	0.14	0.85	-1.34	0.11	0
PBet	-1.80	1.54	-0.97	0.67	0.36	-0.19	0
Mono	-2.77	-0.36	0.84	-1.20	0.63	0.11	0
Gran	0.93	-1.34	-0.27	-0.96	-0.41	-0.27	0
PGran	1.67	0.05	-1.70	-0.52	0.16	0.22	0
Rod	1.19	-1.20	0.58	1.53	0.82	0.01	0
Aero	2.33	1.64	1.38	-0.36	-0.22	0.01	0

#### 1.4) Contribución de los ejes del pca

La Figura 24.2 muestra la manera de graficar a la varianza que captura cada componente principal.

```
x11()
screplot(pca1,ylab="Varianza",main="",
         cex.lab=1.5, col="lightblue")
```



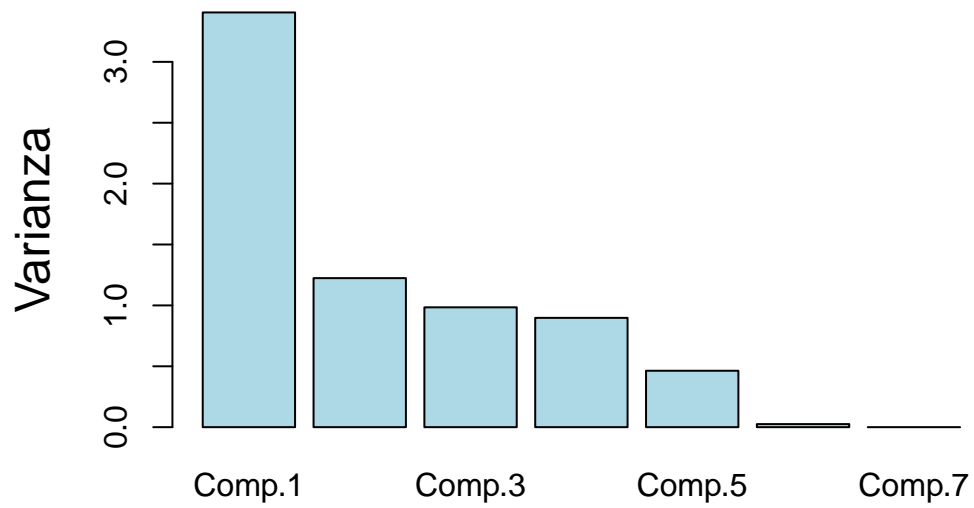


Figura 15.1: Varianza capturada o representada por cada componente principal.

### 1.5) Opciones de biplot, por combinaciones de ejes.

La Figura 24.3 muestra la ordenación de las localidades y las variables ambientales en las 7 bahías evaluadas (gráfico de biplot).

```
biplot(pca1,choices = 1:2, cex=0.9)
abline(v=0,lty=2, col=4)
abline(h=0,lty=2, col=4)
```

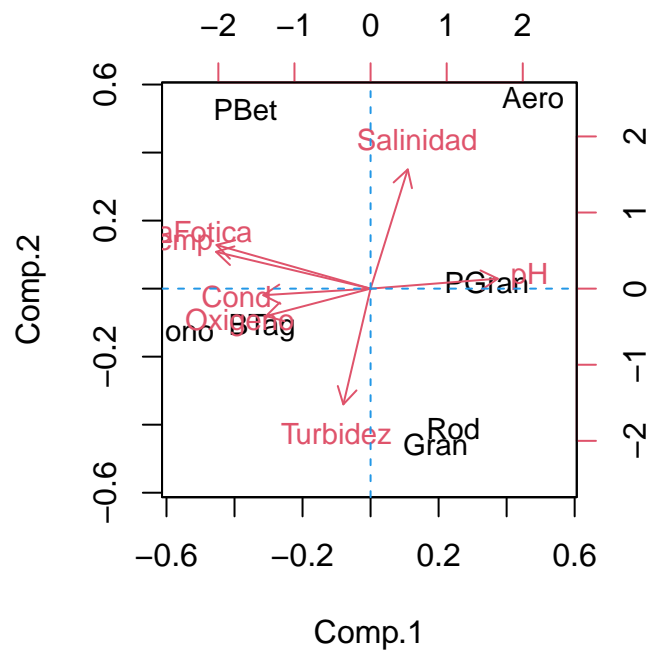


Figura 15.2: Figura del “Biplot” del análisis de componentes principales.

```
# Otras opciones de pca por combinaciones de ejes
biplot(pca1,choices = 2:3, cex=0.9)
biplot(pca1,choices = c(1,3), cex=0.9)
```

## 16 2) PCA con el paquete FactoMiner

### 2.1) Inserción de las variables al PCA

```
# Insertar las variables al PCA
names(datos)
```

```
[1] "Sitio"      "pH"         "Cond"       "Turbidez"   "Temp"
[6] "Salinidad" "CapaFotica" "Oxigeno"
```

```
datos.PCA<-datos[, c("pH", "Cond", "Turbidez", "Temp", "Salinidad",
                     "CapaFotica", "Oxigeno")]
```

### 2.1) PCA con escalamiento de las variables (similar a la matriz de correlación)

```
# Realización del pca con la librería FactoMiner
pca2<-PCA(datos.PCA , scale.unit=TRUE, ncp=5, graph = FALSE)
```

### 2.2) Figura del PCA

La Figura 25.1 muestra la ordenación de las localidades en las 7 bahías evaluadas (gráfico de biplot).

```
# Figura del pca realizado (grafica de observaciones)
plot.PCA(pca2, axes=c(1, 2), choix="ind", habillage="none", col.ind="black",
         col.ind.sup="blue", col.quali="magenta",
         label=c("ind", "ind.sup", "quali"))
```

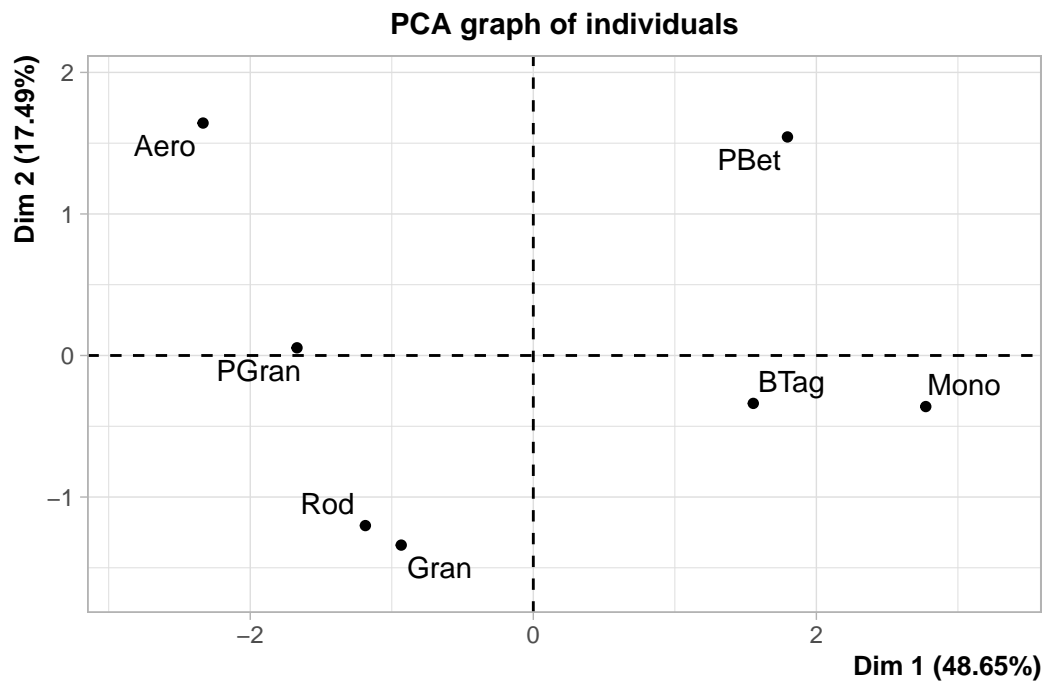


Figura 16.1: Figura del “Plot” del análisis de componentes principales.

### 2.3) Circulo de contribuciones de las variables

La Figura 25.2 muestra el circulo de contribuciones para identificar a las variables con mayor aporte por cada componente principal del análisis.

```
# Circulo de contribuciones
plot.PCA(pca2, axes=c(1, 2), choix="var", col.var="#ff0000", new.plot=T,
         col.quanti.sup="blue", label=c("var", "quanti.sup"), lim.cos2.var=0)
```

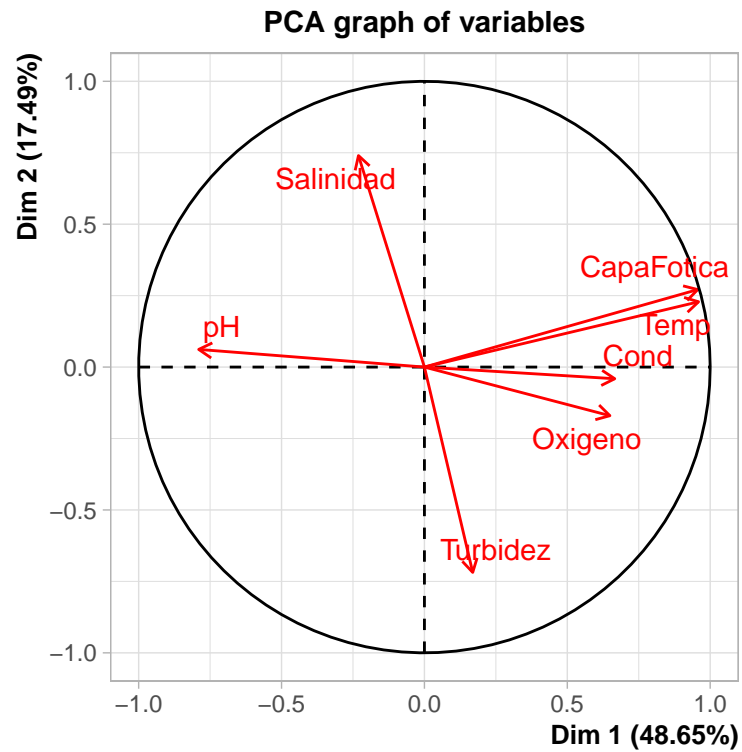


Figura 16.2: Figura del circulo de contribuciones del PCA.

#### 2.4) Selección de variables a partir del PCA

```
# Variables con mayor aporte al PC1
dimdesc=dimdesc(pca2, axes=1:2)
round(dimdesc$Dim.1$quanti,4)
```

	correlation	p.value
Temp	0.9594	0.0006
CapaFotica	0.9564	0.0007
pH	-0.7899	0.0346

## 17 3) PCA con el paquete vegan

```
# Realización del pca
pca3 <- rda(datos[,c(2:8)], scale = TRUE)
```

### 3.1) Insumos del análisis

```
# Insumos del pca
summary(pca3)
```

Call:

```
rda(X = datos[, c(2:8)], scale = TRUE)
```

Partitioning of correlations:

	Inertia	Proportion
Total	7	1
Unconstrained	7	1

Eigenvalues, and their contribution to the correlations

Importance of components:

	PC1	PC2	PC3	PC4	PC5	PC6
Eigenvalue	3.4057	1.2240	0.9841	0.8976	0.46347	0.025214
Proportion Explained	0.4865	0.1749	0.1406	0.1282	0.06621	0.003602
Cumulative Proportion	0.4865	0.6614	0.8020	0.9302	0.99640	1.000000

Scaling 2 for species and site scores

- \* Species are scaled proportional to eigenvalues
- \* Sites are unscaled: weighted dispersion equal on all dimensions
- \* General scaling constant of scores: 2.54573

Species scores

	PC1	PC2	PC3	PC4	PC5	PC6
pH	0.7600	-0.05946	-0.39634	0.16779	0.39815	-0.03017
Cond	-0.6409	0.03919	-0.15521	0.67510	0.17577	0.05298
Turbidez	-0.1626	0.69132	-0.59117	0.02172	-0.26687	-0.01640
Temp	-0.9231	-0.21981	-0.06696	0.04433	0.05922	-0.12436
Salinidad	0.2220	-0.71203	-0.56910	-0.09845	-0.18854	0.02044
CapaFotica	-0.9203	-0.26095	-0.03157	-0.01094	-0.08801	0.04456
Oxigeno	-0.6245	0.16367	-0.22590	-0.57864	0.34878	0.03860

Site scores (weighted sums of species scores)

	PC1	PC2	PC3	PC4	PC5	PC6
BTag	-0.8103	0.29431	-0.1336	0.8605	-1.8923	-0.68752
PBet	-0.9363	-1.34295	0.9378	0.6784	0.5141	1.12733
Mono	-1.4457	0.31382	-0.8173	-1.2159	0.8964	-0.64531
Gran	0.4866	1.16512	0.2621	-0.9789	-0.5863	1.60931
PGran	0.8705	-0.04719	1.6534	-0.5317	0.2280	-1.31394
Rod	0.6186	1.04517	-0.5617	1.5570	1.1561	-0.05927
Aero	1.2166	-1.42828	-1.3407	-0.3694	-0.3161	-0.03060

### 3.2) Autovalores

```
# Ajuste del pca
round((ev <- pca3$CA$eig),2)
```

```
PC1 PC2 PC3 PC4 PC5 PC6
3.41 1.22 0.98 0.90 0.46 0.03
```

### 3.3) Figura del PCA

La Figura 25.3 muestra dos opciones de visualizar los resultados del pca “scaling 1” y “scaling 2”.

```
# Panel con dos figuras del pca
x11(12,6)
par(mfrow=c(1,2))
biplot(pca3, scaling=1, main="PCA - scaling 1")
biplot(pca3, scaling=2, main="PCA - scaling 2")
```

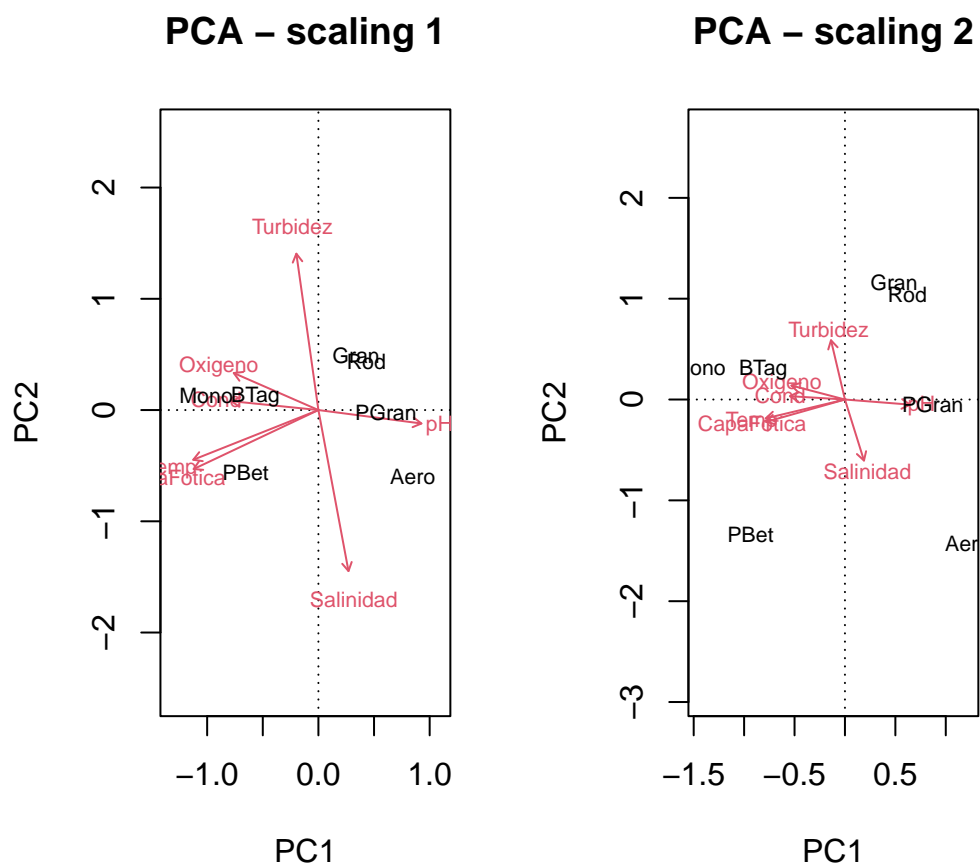


Figura 17.1: dos copciones de figuras del PCA - “scaling 1” y “scaling 2”.

#—



## 18 Taller en casa

Realizar el cálculo de los siguientes insumos de la página 126 a 127, del libro [Análisis de datos ecológicos y ambientales: Aplicaciones con el programa R](#) el cual se encuentra en la biblioteca.

- Ajuste de los componentes principales.
- Figura de autovalores
  - a. Figura del modelo de Kaiser
  - b. Figura del modelo de Varimax

#—

## 19 4) Análisis avanzado de PCA

### 19.0.1 Combinación de clasificación y ordenación

Tomado de [5.3.2.6 Combining Clustering and Ordination Results](#) del libro de Borcard et al. (2018), el cual se encuentra en la base de la Biblioteca de Unimagdalena.

#### 4.1) Generación de grupos con la distancia euclídea y el agrupamiento de Ward

```
str(datos)      # Identificación de las variables cuantitativas

'data.frame':   7 obs. of  8 variables:
 $ Sitio       : chr  "S1" "S1" "S1" "S1" ...
 $ pH          : num  8.42 8.49 8.51 8.56 8.61 ...
 $ Cond        : num  38 38.1 37.8 37.3 37.3 ...
 $ Turbidez    : num  1.364 0.545 1.273 1.273 0.636 ...
 $ Temp        : num  29.5 29.5 29.6 29.3 29.3 ...
 $ Salinidad   : num  2.42 2.43 2.42 2.38 2.38 ...
 $ CapaFotica  : num  19.7 22.1 22.1 10.8 9 ...
 $ Oxigeno     : num  0.097 0.147 0.331 0.17 0.098 0.098 0.098
```

```
# Generación de grupos
datos.w <- hclust(dist(scale(datos[,c(2:8)]))), "ward.D")
```

#### 4.2) Cortar la clasificación en 2 grupos

```
gr <- cutree(datos.w, k = 2)
gr1 <- levels(factor(gr))
```

#### 4.3) Base de datos con el factor agrupador

```
datos.gr=data.frame(gr,datos)      # Dataframe con la variable agrupadora (gr)
datos.gr$gr=as.factor(datos.gr$gr) # crear los grupos como factor
```

#### 4.4) Extraer los scores de los sitios con el pca del paquete “vegan”

```
sit.sc1 <- scores(pca3, display = "wa", scaling = 1)
```

#### 4.5) PCA con simbolos y colores por cada grupo

La Figura 26.1 muestra los dos grupos de bahías generados en el pca.

```
x11()
pc4 <- plot(pca3, display = "wa", scaling = 1, type = "n",
            main = "PCA correlation + clusters")
abline(v = 0, lty = "dotted")
abline(h = 0, lty = "dotted")

for (i in 1:length(grl)) {
  points(sit.sc1[gr == i, ],
         pch = (14 + i),
         cex = 2,
         col = i + 1)
}

# Agregar los rótulos de los sitios
text(sit.sc1, row.names(datos), cex = 0.7, pos = 3)

# Adicionar el dendograma al pca generado (uniones entre puntos)
ordicluster(pc4, datos.w, col = "dark grey")

# Adicionar la leyenda de la figura (**Nota**: hacer clic en la figura)
legend(locator(1),
       paste("Grupo", c(1:length(grl))),
       pch = 14 + c(1:length(grl)),
       col = 1 + c(1:length(grl)),
       pt.cex = 2)
```

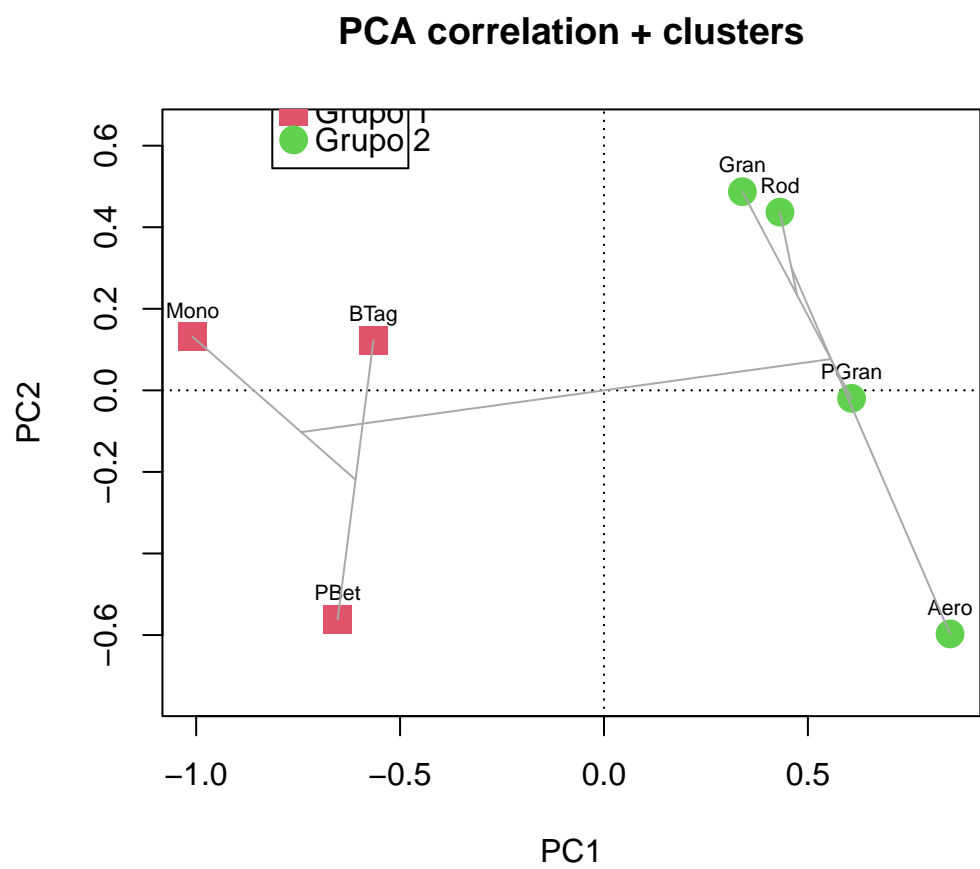


Figura 19.1: dos grupos de bahías generados en el PCA.

## 20 5) PCA por tipos con la función “dudi.pca” del paquete ade4

```
pca5 <- dudi.pca(datos[,c(2:8)],scannf=F,nf=2,scale=T)
```

### 5.1) Figuras del pca por tipo de grupo

La Figura 26.2 muestra el agrupamiento por elipses en el pca.

```
s.class(pca5$li,datos.gr$gr, cpoi = 2)
```

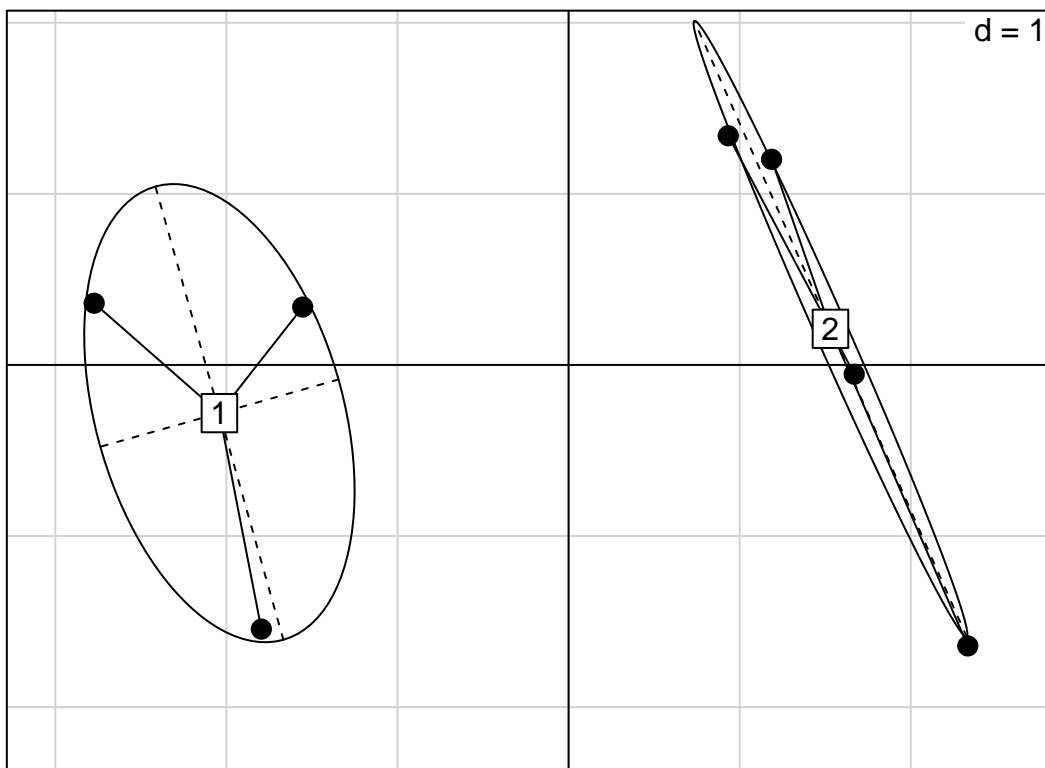


Figura 20.1: agrupamiento por elipses en el PCA.

La Figura 20.2 muestra el agrupamiento por líneas en el pca.

```
s.class(pca5$li,datos.gr$gr, cell = 0, cstar = 0.5)
```

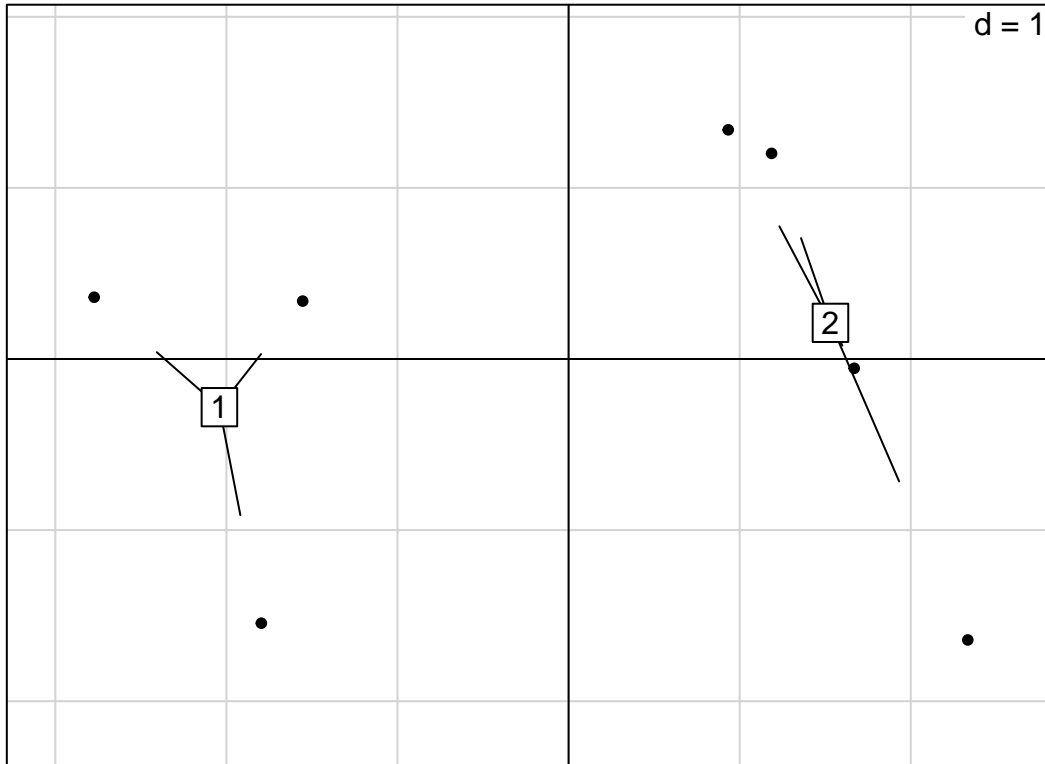


Figura 20.2: agrupamiento por líneas en el PCA.

La Figura 20.3 muestra el agrupamiento por triángulos en el pca.

```
coul <- c("red", "blue")  
s.chull(pca5$li,datos.gr$gr, cpoi = 1, col = coul)
```

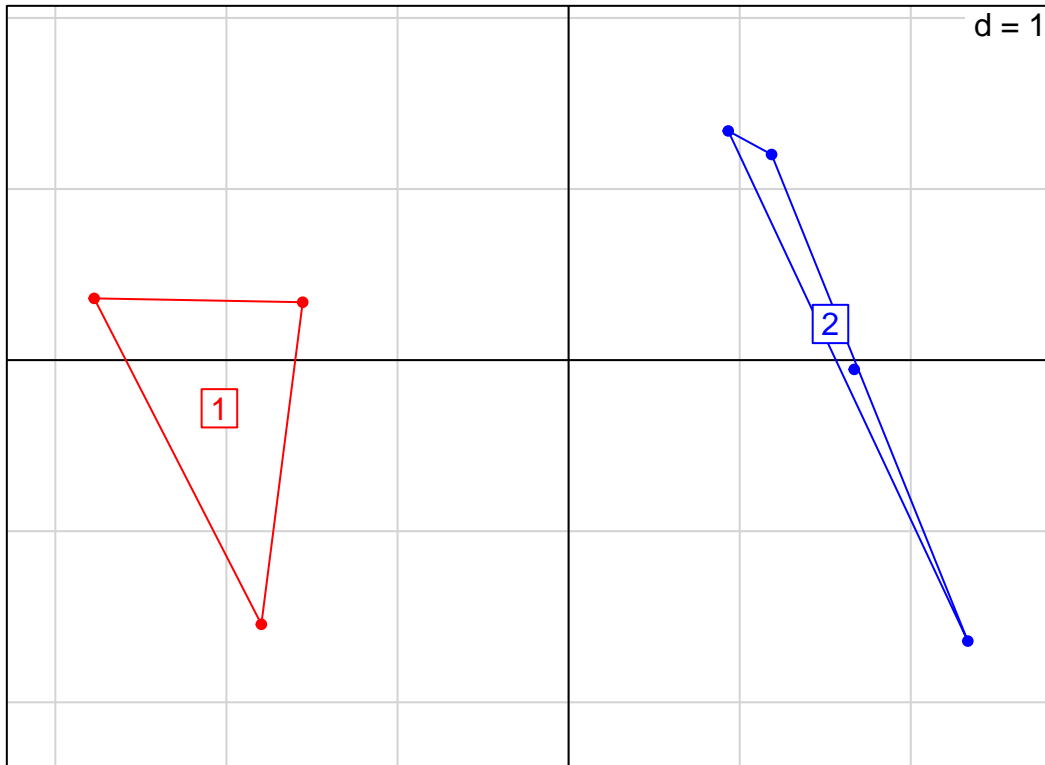


Figura 20.3: agrupamiento por triangulos en el PCA.

## 21 Taller 4.2 Análisis de Componentes Principales - PCA

El siguiente ejemplo tiene en cuenta a la propuesta de Legendre & Gallagher (2001), en el cual se realiza una linealización de datos de abundancias de taxones, mediante la transformación de Hellinger, para poderlas ordenar en un PCA. Adicionalmente se incorporan las variables ambientales, con el objeto de analizar como estas caracterizan a las biológicas en gradientes espaciales y/o temporales. La base de datos que se utilizará es Tayrona.csv y el archivo de R es Tayrona.pca.r. Estos datos corresponden a un estudio realizado en el 2015, en el Parque Nacional Natural Tayrona (PNNT), valorando la fauna de invertebrados acuáticos y variables fisicoquímicas asociadas en diferentes quebradas de ese lugar. Estos datos hacen parte del trabajo realizado por Bruges Emilio (2022).

Ejercicio tomado de: Rodríguez-Barrios (2023) [Enlace del libro](#)

[Enlace de los archivos del libro](#)

Fuentes bibliográficas sobre el análisis de componentes principales:

- [Métodos de componentes principales en R](#) - STHDA
- [Artículos - Métodos de componentes principales](#) - STHDA
- [PCA en factoextra](#) - datanovia
- [Guía práctica sobre el PCA](#) - datanovia
- [PCA para variables categóricas](#) - R-bloggers
- [Capítulo PCA](#) - Libro Numerical Ecology with R - Borcard et al. 2018

```
# Librerías requeridas
library(ggplot2)
library(reshape2)
library(ggrepel)
library(vegan)
library(factoextra)
library(ggsci)
library(ggforce)
library(concaveman)
```



```
library(corrplot)

# Lectura de la base de datos "FQmarino"
datos <- read.csv2("Tayrona.csv", row.names=1) # file.choose()
View(datos)
# str(datos)
```

## 22 Exploración Gráfica

```
# Matriz de correlaciones (M)
amb = datos[,c(2:12)]
biol = datos[,c(13:63)]
M <- cor(amb, biol)
```

La Figura 22.1 muestra la relación entre las variables, a partir de figuras de elipses.

```
x11(8, 6)
corrplot(M, method = "ellipse", type = "upper")
```

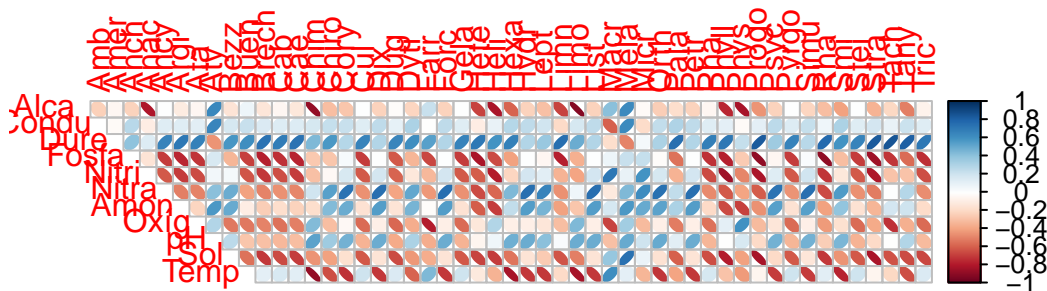


Figura 22.1: Relación de variables ambientales en las siete bahías estudiadas.

## 23 1) Ajuste de las bases de datos fisiqcoquímica (amb) y biológica (tax.hel)

```
datos$Epoca = as.factor (datos$Epoca)    # Convertir Epoca a factor
```

```
# Variables ambientales
amb= log10(datos[,c(2:12)]+1)
round(head(amb),1)
```

	Alca	Condu	Dure	Fosfa	Nitri	Nitra	Amon	Oxíg	pH	Sol	Temp
M.s	2.2	2.8	1.7	0.0	0.0	0.2	0	0.6	1.0	2.3	1.4
ST.s	2.1	2.8	1.7	0.1	0.1	0.2	0	0.8	1.0	2.4	1.4
Bo.s	2.2	3.0	1.7	0.1	0.1	0.5	0	0.8	1.0	2.6	1.4
M.1	2.2	2.9	1.7	0.1	0.1	0.5	0	0.7	1.0	2.5	1.4
ST.1	2.2	2.5	1.6	0.2	0.1	0.3	0	0.7	0.9	2.5	1.4
Bo.1	2.2	2.4	1.7	0.1	0.1	0.3	0	0.6	1.0	2.6	1.4

```
# Siete primeros Taxones transformados con Hellinger
tax.hel= decostand(datos[,c(13:63)],"hellinger")
round(head(tax.hel[,1:7]),2)
```

	Amb	Amer	Anch	Anac	Ancy	Argi	Ata
M.s	0.11	0.13	0.04	0.00	0.03	0.07	0.30
ST.s	0.07	0.07	0.00	0.08	0.00	0.04	0.14
Bo.s	0.04	0.06	0.00	0.00	0.00	0.00	0.06
M.1	0.09	0.17	0.10	0.00	0.00	0.00	0.00
ST.1	0.00	0.20	0.00	0.14	0.00	0.00	0.31
Bo.1	0.00	0.32	0.00	0.00	0.00	0.00	0.10

```
# Matriz de correlaciones con variables transformadas (M1)
M1 <- cor(amb, tax.hel)
```

La Figura 23.1 muestra la relación entre las variables, con las transformaciones realizadas.

```
# Figuras de elipses con variables transformadas
x11(8, 6)
corrplot(M1, method = "ellipse", type = "upper")
```

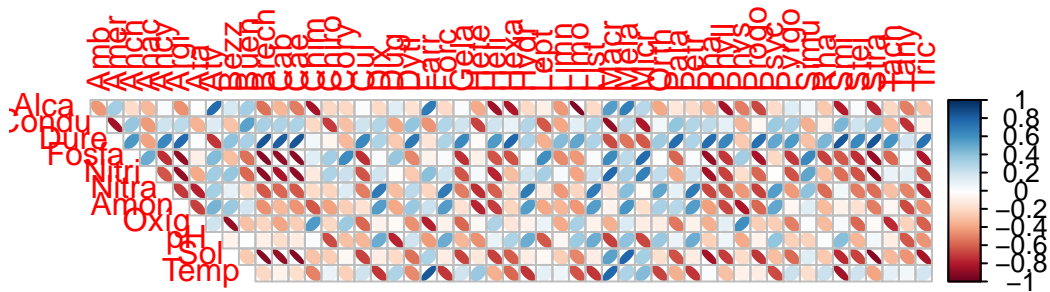


Figura 23.1: Relación de variables ambientales en las siete bahías estudiadas.

## 24 2) PCA con paquete factoextra

```
pca1 <- prcomp(amb,scale.=T)
summary(pca1)
```

Importance of components:

	PC1	PC2	PC3	PC4	PC5	PC6
Standard deviation	2.1415	1.8802	1.3296	0.90111	0.54661	1.977e-15
Proportion of Variance	0.4169	0.3214	0.1607	0.07382	0.02716	0.000e+00
Cumulative Proportion	0.4169	0.7383	0.8990	0.97284	1.00000	1.000e+00

### 24.0.1 2.1) Contribución eje 1

La Figura 24.1 muestra las contribuciones de cada variable ambiental al pca.

```
x11(5,5)
fviz_contrib(pca1,choice="var",axes=1)
```

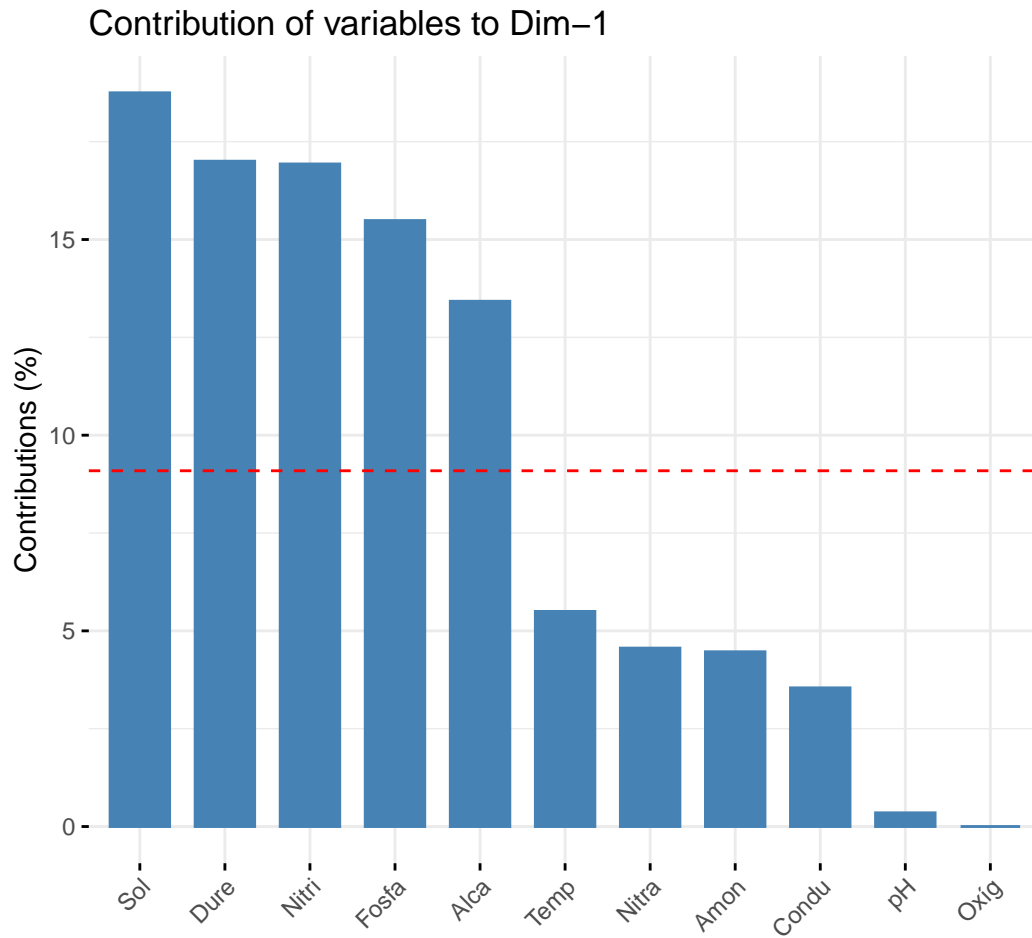


Figura 24.1: Contribuciones de las variables ambientales.

#### 24.0.1.1 2.2) Elipses por cada periodo climático

La Figura 24.2 muestra la ordenación de las localidades por cada periodo climático.

```
fviz_pca_ind(pca1, geom.ind = "point",
             col.ind = datos$Epoca, # Colores por grupo - periodo
             palette = c("#00AFBB", "#E7B800", "#FC4E07"),
             addEllipses = TRUE, ellipse.type = "confidence",
             legend.title = "Groups")
```

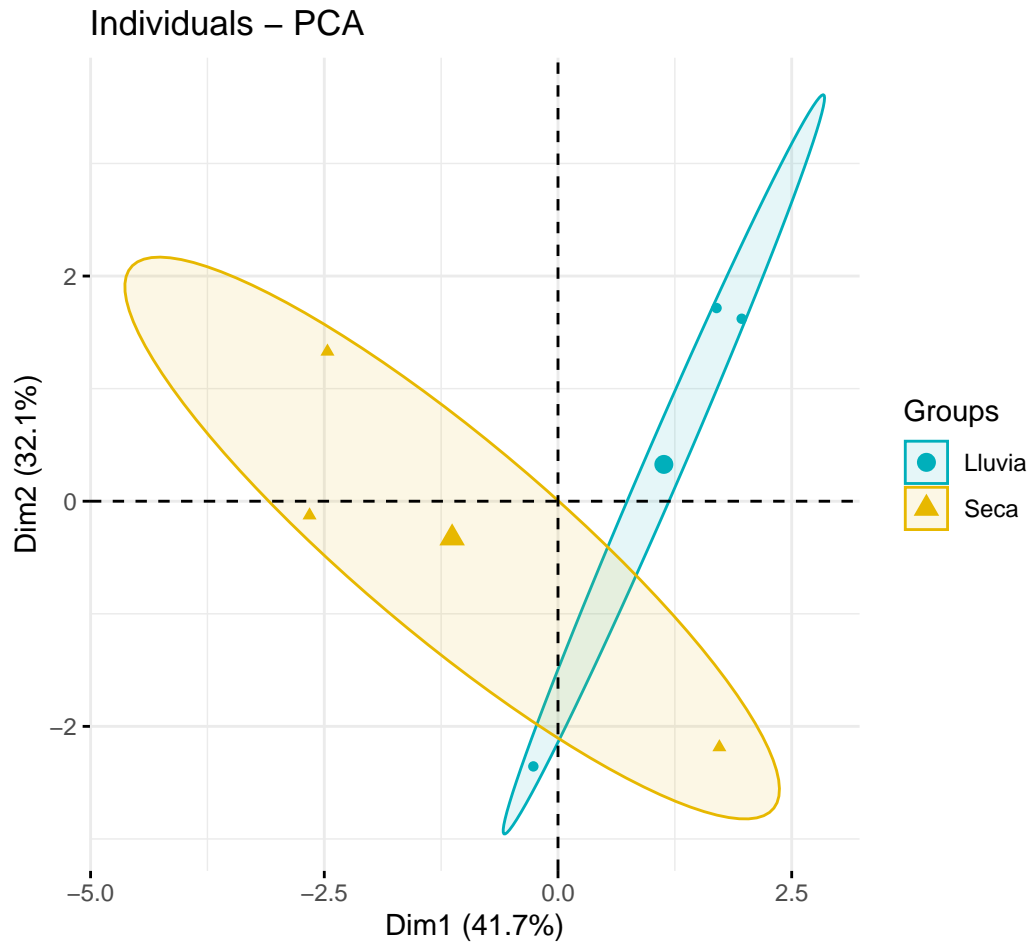


Figura 24.2: Odenación por cada periodo climático.

### 24.0.2 2.3) Escala de contribuciones de las observaciones y las variables

La Figura 24.3 muestra las contribuciones de cada variable ambiental al pca.

```
fviz_pca_biplot(pca1,
# Observaciones (Sitios)
  geom.ind = "point",
  fill.ind = datos$Epoca, col.ind = "black",
  pointshape = 21, pointsize = 2,
  palette = "jco",
  addEllipses = TRUE,
# Variables ambientales
```

```
col.var = "contrib",
gradient.cols = "RdYlBu",
legend.title = list(fill = "Epocas", color = "Contrib",
                    alpha = "Contrib"))
```

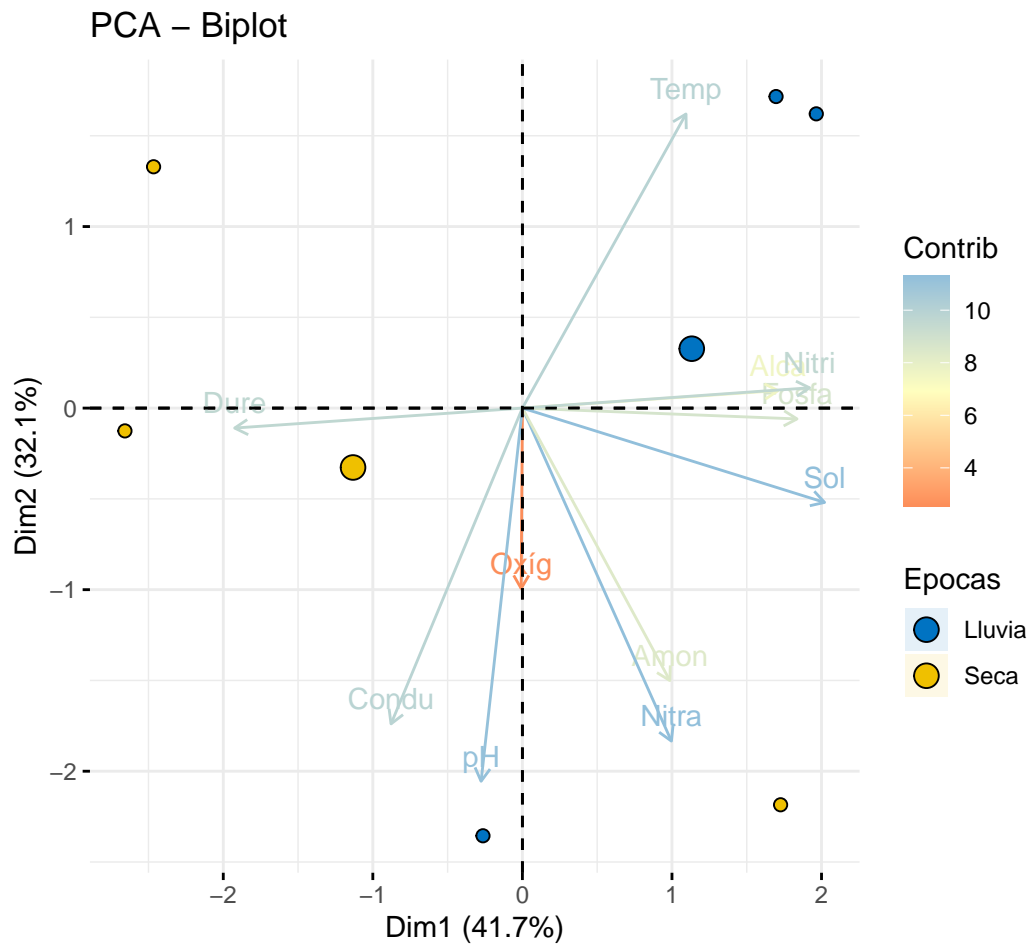


Figura 24.3: Contribuciones de las variables ambientales.



## 25 3) PCA con vegan

```
pca2 <- rda(tax.hel)
```

### 25.0.1 3.1) Insumos del análisis

**\*Nota:** No se ejecutará el siguiente comando para poder resumir los insumos obtenidos del análisis.

```
# Insumos del pca  
summary(pca2)
```

### 25.0.2 3.2) Autovalores

```
# Ajuste del pca  
round((ev <- pca2$CA$eig),2)
```

PC1	PC2	PC3	PC4	PC5
0.22	0.12	0.09	0.06	0.03

### 25.0.3 3.3) Figura del PCA

La Figura 25.1 muestra dos opciones de visualizar los resultados del pca “scaling 1” y “scaling 2”.

```
# Panel con dos figuras del pca  
x11(12,6)  
par(mfrow=c(1,2))  
biplot(pca2, scaling=1, main="PCA - scaling 1")  
biplot(pca2, scaling=2, main="PCA - scaling 2")
```

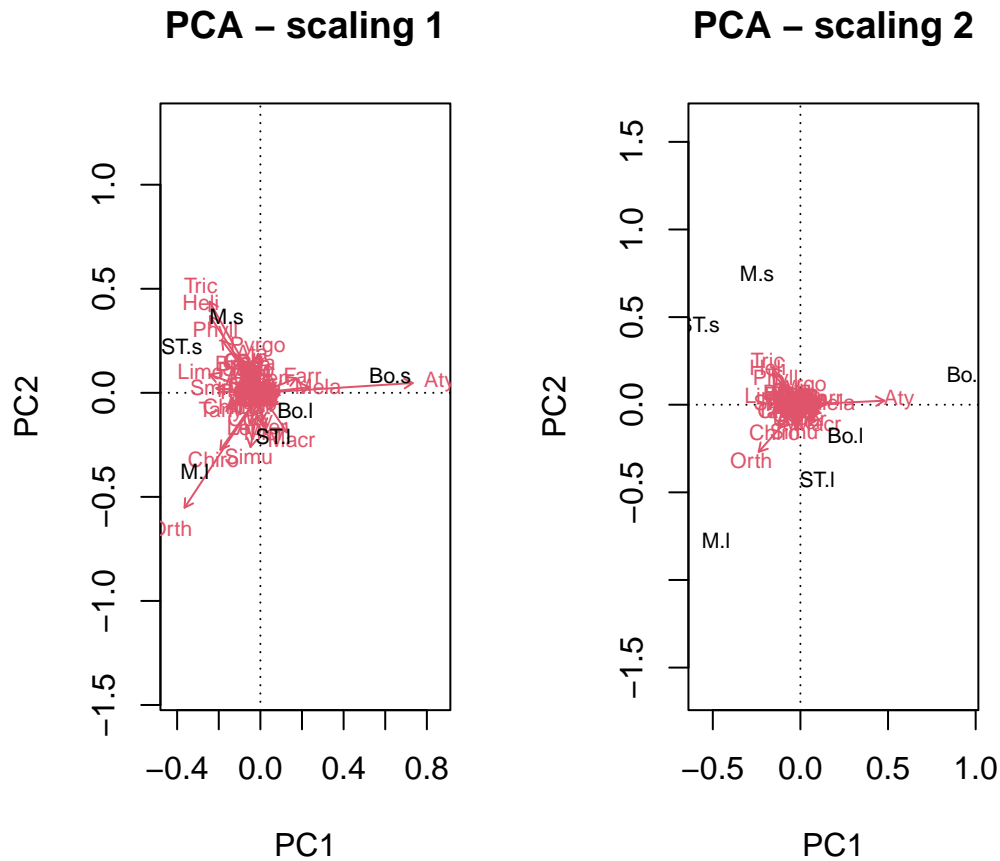


Figura 25.1: dos copciones de figuras del PCA - “scaling 1” y “scaling 2”.

#### 25.0.4 3.4) PCA con vegan - biplot + orditorp

La Figura 25.2 muestra la ordenación de las localidades y los taxones con “scaling 2”.

```
x11(8,8)
biplot(pca2, choices = c(1, 2), type = "n", scaling = 2,
       main="PCA - Scaling 2", cex=2) # Panel gráfico
text(pca2, display="sites", cex=0.8,
     col="blue", lwd=1.5, pos=3) # Figura de sitios y Épocas
ordi=orditorp(pca2, display = "species",
              shrink = FALSE, col = "red", type="n") # Taxones Filtrados
```

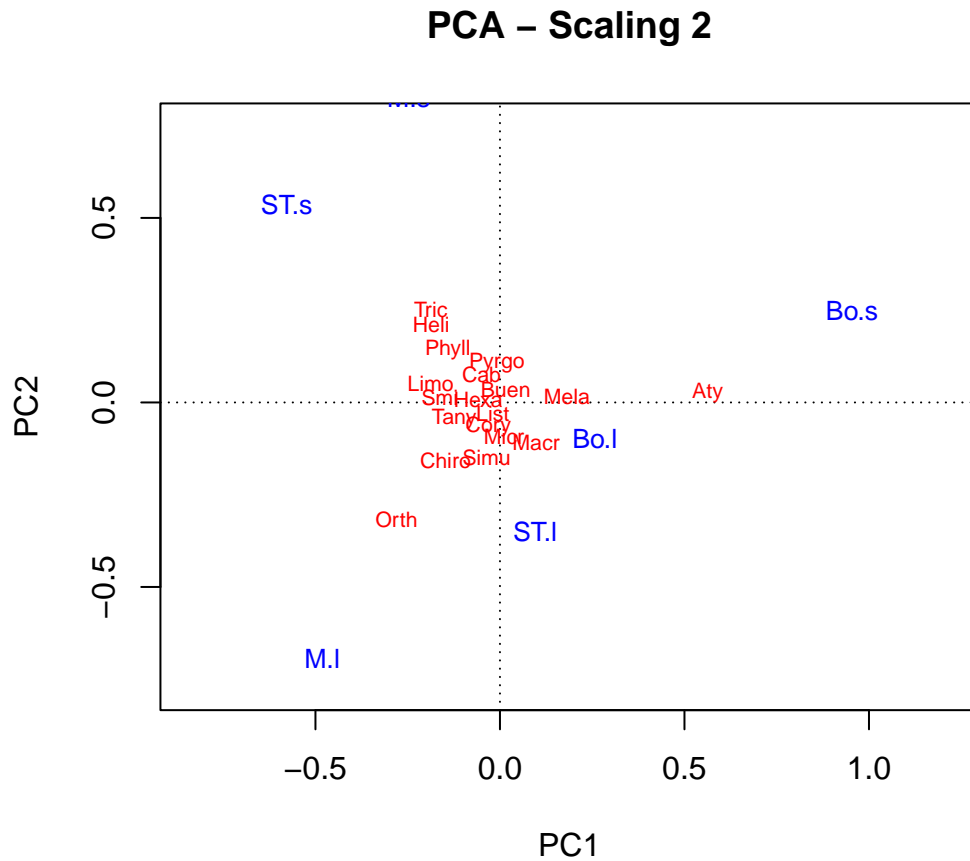


Figura 25.2: Ordenación de las localidades y los taxones, con scaling 2.

### 25.0.5 3.5) PCA con vegan + orditorp + envfit (ajuste ambiental)

La Figura 25.3 muestra la ordenación de las localidades, los taxones y las variables ambientales con “scaling 2”.

```
biplot(pca2, choices = c(1, 2), type = "n", scaling = 2,
       main = "PCA - Scaling 2", cex = 2) # Panel gráfico
text(pca2, display = "sites", cex = 0.8,
     col = "blue", lwd = 1.5, pos = 3) # Figura de sitios y Épocas
ordi = orditorp(pca2, display = "species",
               shrink = FALSE, col = "red", type = "n") # Taxones Filtrados
points(pca2, display = "sites",
```

```

    cex = 0.6, col = "lightblue", lwd=1.5) # Opcional - puntos de muestreo
amb1 = envfit(pca2,amb) # Insertar variables ambientales en el pca
plot(amb1,col=3,cex=0.7)

```

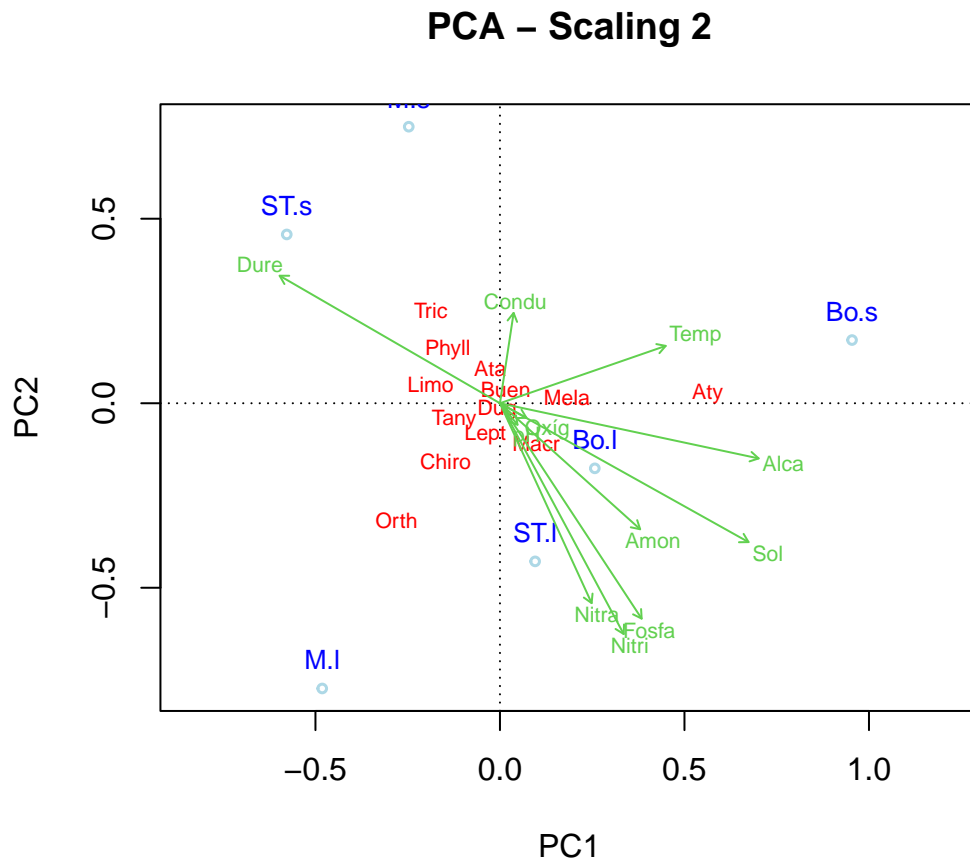


Figura 25.3: Ordenación de las localidades, los taxones y las variables ambientales

#—

## 26 4) PCA con paquete ggplot2

Realización pca de los paquetes factoextra y ggbiplot Para generar las coordenadas de los sitios y taxones

```
# Nuevamente el pca
pca3 <- prcomp(tax.hel)
```

### 26.0.1 4.1 Coordenadas de los sitios y el factor “coord.sit”

```
coord.sit <- as.data.frame(pca3$x[,1:2]) # Coordenadas de los sitios
coord.sit$sitio <- rownames(coord.sit) # Crear una columna con nombres de los sitios
coord.sit$grp <- datos$Epoca # Adicionar columna de grupos por Epoca
head(coord.sit) # vista resumida de las coordenadas de sitios
```

	PC1	PC2	sitio	grp
M.s	-0.20509987	0.4633949	M.s	Seca
ST.s	-0.47947160	0.2828349	ST.s	Seca
Bo.s	0.79183051	0.1058759	Bo.s	Seca
M.l	-0.39969319	-0.4779970	M.l	Lluvia
ST.l	0.07910077	-0.2650291	ST.l	Lluvia
Bo.l	0.21333338	-0.1090797	Bo.l	Lluvia

### 26.0.2 4.2 Coordenadas de los taxones “coord.tax”

```
coord.tax <- as.data.frame(pca3$rotation[,1:2]) # Dos primeros ejes
coord.tax$especies <- rownames(coord.tax) # Insertar columna con nombres de las es
head(coord.tax)
```

	PC1	PC2	especies
Amb	-0.050809593	0.05712214	Amb
Amer	0.005666684	-0.12845725	Amer
Anch	-0.042528327	-0.04368374	Anch

Anac	-0.024746853	-0.02282280	Anac
Ancy	-0.006287099	0.02558412	Ancy
Argi	-0.029886629	0.06956166	Argi

### 26.0.3 4.3 Coordenadas de las ambientales “coord.amb”

```
amb1 = envfit(pca3,amb)
coord.amb = as.data.frame(scores(amb1, "vectors"))
coord.amb$amb <- rownames(coord.amb)      # Insertar columna con nombres de las ambientales
head(coord.amb)
```

	PC1	PC2	amb
Alca	0.84803475	-0.2427936	Alca
Condu	0.03410165	0.3025273	Condu
Dure	-0.67034682	0.5206620	Dure
Fosfa	0.37835304	-0.7719269	Fosfa
Nitri	0.32400015	-0.8109169	Nitri
Nitra	0.23720481	-0.6939225	Nitra

### 26.0.4 4.4 Figura con de elipses por concavidades - geom\_mark\_hull

La Figura 26.1 muestra la ordenación de las localidades, los taxones y los periodos climáticos.

```
x11(6,6)
ggplot() +
  # Sitios
  geom_text_repel(data = coord.sit,aes(PC1,PC2,label=row.names(coord.sit)),
                  size=4)+ # Muestra el cuadro de la figura
  geom_point(data = coord.sit,aes(PC1,PC2,colour=grp),size=4)+
  scale_shape_manual(values = c(21:25))+
  # Taxones *valores de cero para caracteres de las flechas (arrow)
  geom_segment(data = coord.tax,aes(x = 0, y = 0, xend = PC1, yend = PC2),
              arrow = arrow(angle=0,length = unit(0,"cm"),
                            type = "closed"),linetype=0, size=0,colour = "red")+
  geom_text_repel(data = coord.tax,aes(PC1,PC2,label=especies),colour = "red")+
  # Factor
  geom_mark_hull(data=coord.sit, aes(x=PC1,y=PC2,fill=grp,group=grp,
                                     colour=grp),alpha=0.30) +

  geom_hline(yintercept=0,linetype=3,size=1) +
```

```
geom_vline(xintercept=0,linetype=3,size=1)+
guides(shape=guide_legend(title=NULL,color="black"),
       fill=guide_legend(title=NULL))+
theme_bw()+theme(panel.grid=element_blank())
```

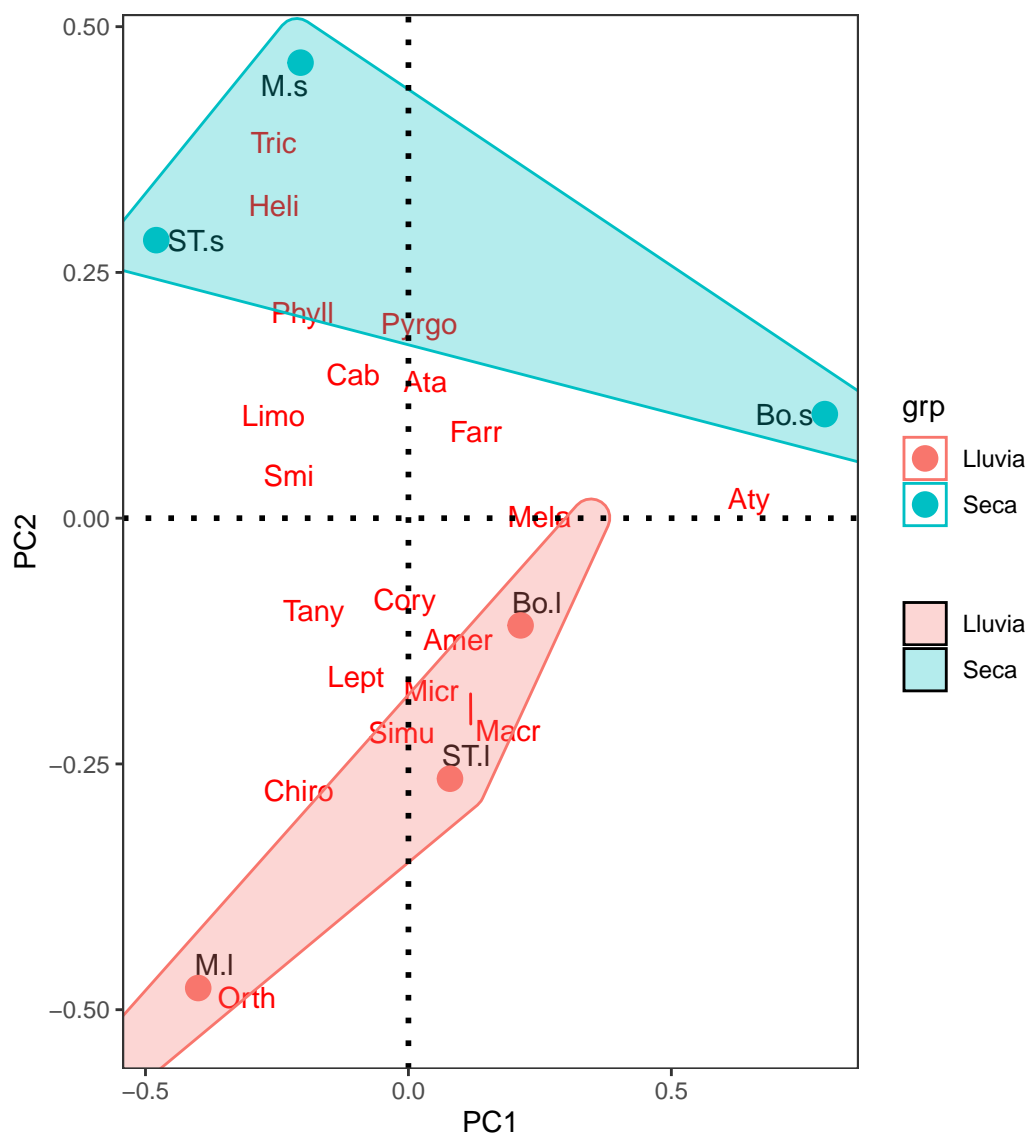


Figura 26.1: Ordenación de las localidades, los taxones y los periodos climáticos.

## 26.0.5 4.5 Figura con vectores de especies y ambientales

La Figura 26.2 muestra la ordenación de las localidades, los taxones, las variables ambientales y los periodos climáticos.

```
library(ggrepel)

x11(6,6)
ggplot() +
# Sitios
  geom_text_repel(data = coord.sit,aes(PC1,PC2,label=row.names(coord.sit)),
                  size=4)+ # Muestra el cuadro de la figura
  geom_point(data = coord.sit,aes(PC1,PC2,colour=grp),size=4)+
  scale_shape_manual(values = c(21:25))+
# especies
  geom_segment(data = coord.tax,aes(x = 0, y = 0, xend = PC1, yend = PC2),
              arrow = arrow(angle=22.5,length = unit(0.25,"cm"),
                           type = "closed"),linetype=1, size=0.6,colour = "red")+
  geom_text_repel(data = coord.tax,aes(PC1,PC2,label=especies),colour = "red")+
# Ambiental
  geom_segment(data = coord.amb,aes(x = 0, y = 0, xend = PC1, yend = PC2),
              arrow = arrow(angle=22.5,length = unit(0.25,"cm"),
                           type = "closed"),linetype=1, size=0.6,colour = "blue")+
  geom_text_repel(data = coord.amb,aes(PC1,PC2,label=row.names(coord.amb)),colour = "#00ab
# Factor
  geom_polygon(data=coord.sit,aes(x=PC1,y=PC2,fill=grp,group=grp),alpha=0.30) +

  geom_hline(yintercept=0,linetype=3,size=1) +
  geom_vline(xintercept=0,linetype=3,size=1)+
  guides(shape=guide_legend(title=NULL,color="black"),
         fill=guide_legend(title=NULL))+
  theme_bw()+theme(panel.grid=element_blank())
```





## 27 Taller 5.1 Análisis de Escalamiento Multidimensional no Métrico - NMDS

### Objetivo de la actividad:

La siguiente base de datos, corresponde a una muestra de 50 especies de malezas asociadas a cultivos de banano en cuatro localidades del Departamento del Magdalena (Regiones Alta, Norte, Media y Baja), basado en la composición y abundancia de estas especies vegetales. Estos datos fueron tomados del estudio realizado por Quintero-Pertuz et al., 2020) y solo representan a una parte de los taxones registrados (en total fueron 202 especies). Se utilizará el siguiente archivo como base de datos: **maezas.csv**

Ejercicio tomado de: Rodríguez-Barrios (2023) [Enlace del libro](#)

[Enlace de los archivos del libro](#)

### 27.0.1 Referencias bibliográficas de apoyo.

[Perifiton de un río de Montaña - Osorio et al. 2014]([https://www.scielo.sa.cr/scielo.php?script=sci\\_arttext&pid=77442011000400009#:~:text=Los%20GFA%20considerados%20en%20este,t%C3%A1xones%20neotropicales%20Valoración del proceso sucesional de microalgas perifíticas el tramo medio del río Gaira - Santa Marta.](https://www.scielo.sa.cr/scielo.php?script=sci_arttext&pid=77442011000400009#:~:text=Los%20GFA%20considerados%20en%20este,t%C3%A1xones%20neotropicales%20Valoración del proceso sucesional de microalgas perifíticas el tramo medio del río Gaira - Santa Marta.))

[Invertebrados de un río de Montaña - Rodríguez-Barrios et al. 2011](#) Estudio de diferentes atributos comunitarios en invertebrados acuáticos del río Gaira - Santa Marta.

[Descomposición de Hojarasca en Ríos - Eyes et al. 2011](#) Trabajo realizado en el bosque de ribera del río Gaira - Santa Marta.

[Nutrientes de la hojarasca - Fuentes y Rodríguez. 2011](#) Otro trabajo realizado en el bosque de ribera del río Gaira - Santa Marta.

[Análisis de Vulnerabilidad a Inundaciones - Noriega et al. 2011](#) Valoración de riesgo a inundaciones en la parte baja del río Gaira - Santa Marta.

## 27.0.2 Procedimiento de la exploración

- Cargar librerías requeridas
- Cargar la base `malezas.csv`
- Correr el NMDS con una distancia binaria (Jaccard)
- Realizar las opciones gráficas con las librerías "vegan" y "ggplot2".

### 27.0.2.1 Cargar las librerías requeridas

```
# Librerías requeridas
library(ade4)
require(vegan)
library(analogue)
library(magrittr)
library(dplyr)
library(ggpubr)
library(vegan)
library(ggplot2)
library(ggrepel)
```

### 27.0.2.2 Cargar o importar la base de datos

```
# Base de datos
datos<-read.csv2("malezas.csv",row.names=1)
```

## 27.0.3 1) Ordenación de las localidades y las especies de malezas.

Se presenta un estrés de 0.13 (13%) con la distancia binaria de Jaccard.

```
# 1. 1) Ordenación con el nmms
datos.nmms <- metaMDS(datos[,3:52],trace = FALSE,distance = "jaccard")
datos.nmms
```

Call:

```
metaMDS(comm = datos[, 3:52], distance = "jaccard", trace = FALSE)
```

global Multidimensional Scaling using monoMDS

Data: datos[, 3:52]

Distance: jaccard

Dimensions: 2

Stress: 0.1362424

Stress type 1, weak ties

Best solution was repeated 3 times in 20 tries

The best solution was from try 10 (random start)

Scaling: centring, PC rotation, halfchange scaling

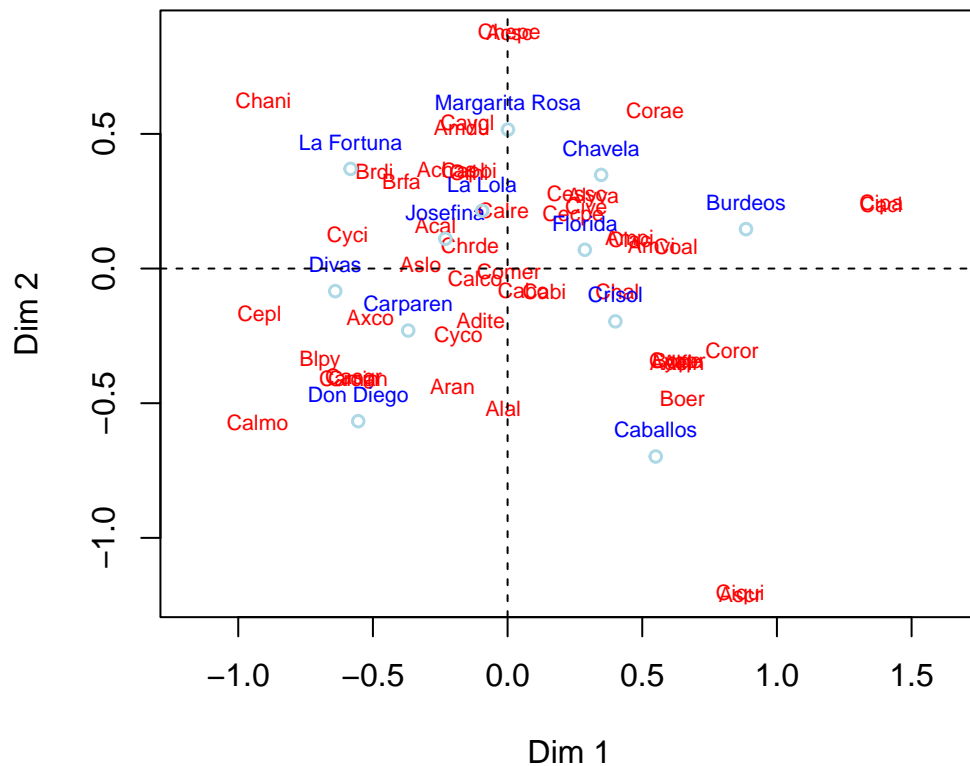
Species: expanded scores based on 'datos[, 3:52]'

## 27.0.4 2) Figuras del nmms con el paquete “vegan”

### 27.0.4.1 2.1 nMDS con solapamiento de taxones

La figura muestra la ordenación de las especies de malezas con las localidades evaluadas. Muchas de las especies quedan solapadas en la figura, en la siguiente figura se aplicará un comando para eliminar el solapamiento.

```
x11()
fig=plot(datos.nmms, type = "t",display = c("n", "species"),
        ylab="Dim 2", xlab="Dim 1", cex=0.7,shrink = FALSE)
# Texto
text(datos.nmms, display="sites", labels = as.character(datos$Finca),
      cex=0.7, col="blue", lwd=1.5, pos=3)
# Puntos *opcionales
points(datos.nmms, display = "sites",cex = 0.8,
       col = "lightblue", lwd=1.5)
# plano cartesiano
abline(h=0,lty=2)
abline(v=0,lty=2)
```

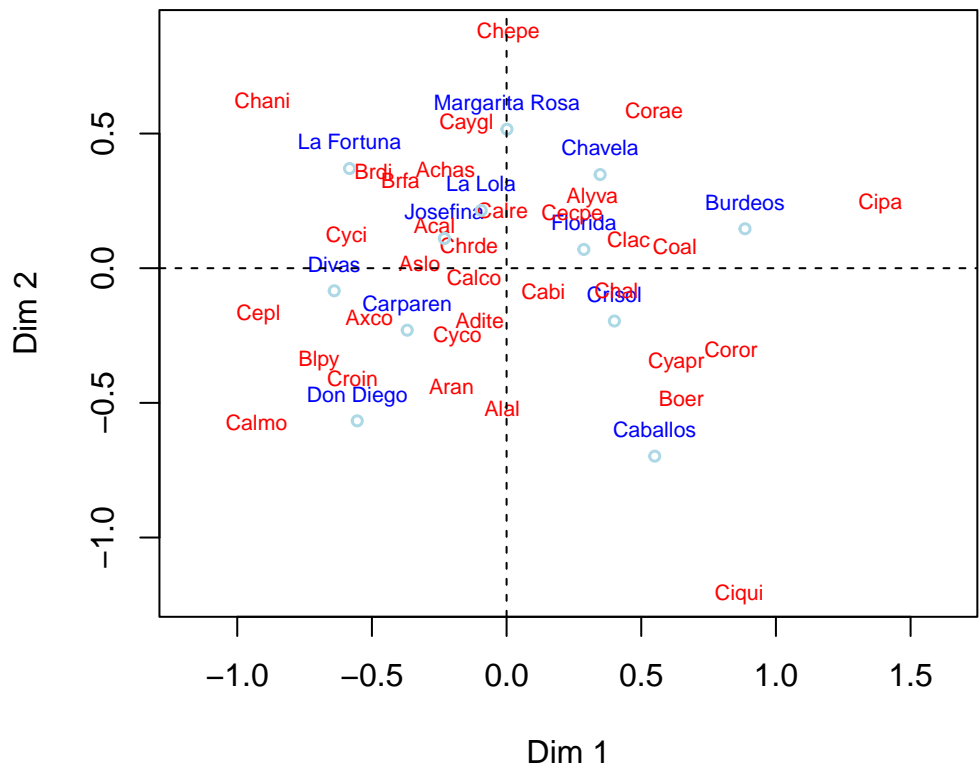


#### 27.0.4.2 2.2 Ordenación con el comando “orditorp”

La figura elimina el solapamiento de las especies con el comando `orditorp`. Las especies de malezas son graficadas en rojo y las fincas en color azul.

```
x11()
fig=plot(datos.nmds, type = "n",display = c("n", "species"),
        ylab="Dim 2", xlab="Dim 1", ex=0.7,shrink = FALSE)
#
text(datos.nmds, display="sites", labels = as.character(datos$Finca),
     cex=0.7, col="blue", lwd=1.5, pos=3)
#
ordi=orditorp(datos.nmds, display = "species",
```

```
        shrink = FALSE, col = "red", type="n")  
#  
points(datos.nmds, display = "sites", cex = 0.7,  
       col = "lightblue", lwd=1.5)  
abline(h=0,lty=2)  
abline(v=0,lty=2)
```



### 27.0.5 3) NMDS con paquete ggplot2

A continuación se utilizarán las coordenadas de las regiones, las fincas y las especies de malazas, para graficarlas con el paquete **ggplot2**, debido a que muestra unas imágenes más didácticas y compactas. El siguiente comando - **names(datos.nmds)**, permite visualizar los insumos del escalamiento multidimensional realizado.

```
# Correr nuevamente el nMDS
names(datos.nmds)
```

```
[1] "nobj"      "nfix"      "ndim"      "ndis"      "ngrp"
[6] "diss"      "iidx"      "jidx"      "xinit"     "istart"
[11] "isform"    "ities"     "iregn"     "iscal"     "maxits"
[16] "sratmx"    "strmin"    "sfgrmn"    "dist"      "dhat"
[21] "points"    "stress"    "grstress"  "iters"     "icause"
[26] "call"      "model"     "distmethod" "distcall"  "data"
[31] "distance"  "converged" "tries"     "bestry"    "engine"
[36] "species"
```

### 27.0.5.1 3.1 Coordenadas de los sitios y el factor “coord.sit”

Con el siguiente comando - `datos.nmds$points`, se extraen las coordenadas de los sitios y con el comando `datos$Región` se obtienen la columna del factor región.

```
# 1) Coordenadas de los sitios y el factor (coord.sit)
coord.sit <- as.data.frame(datos.nmds$points) # Coordenadas de los sitios
coord.sit$sitio <- rownames(coord.sit)        # Crear una columna con nombres de los sitios
coord.sit$grp <- datos$Región                  # Adicionar columna de grupos por región
head(coord.sit)                               # vista resumida de las coordenadas de sitios
```

	MDS1	MDS2	sitio	grp
ACa	-0.368863393	-0.2302086	ACa	Alta
ACr	0.400345694	-0.1960922	ACr	Alta
ALa	-0.583057215	0.3700283	ALa	Alta
BBu	0.885281859	0.1463697	BBu	Baja
BLa	-0.094239675	0.2144785	BLa	Baja
BMa	0.001413751	0.5159338	BMa	Baja

### 27.0.5.2 3.2 Coordenadas de los taxones “coord.tax”

Con el comando `datos.nmds$species`, se extraen las coordenadas de las especies.

```
# 2) Coordenadas de las especies (coord.tax)
coord.tax <- as.data.frame(datos.nmds$species) # Dos primeros ejes
coord.tax$species <- rownames(coord.tax)       # Insertar columna con nombres de las especies
head(coord.tax)
```

	MDS1	MDS2	especies
Acal	-0.267753232	0.1591921	Acal
Achas	-0.226794494	0.3661127	Achas
Acsc	0.006703529	0.8749406	Acsc
Adein	0.629901064	-0.3486008	Adein
Adite	-0.099401544	-0.1925385	Adite
Alal	-0.016474541	-0.5190483	Alal

### 27.0.5.3 3.3 Figura con de elipses

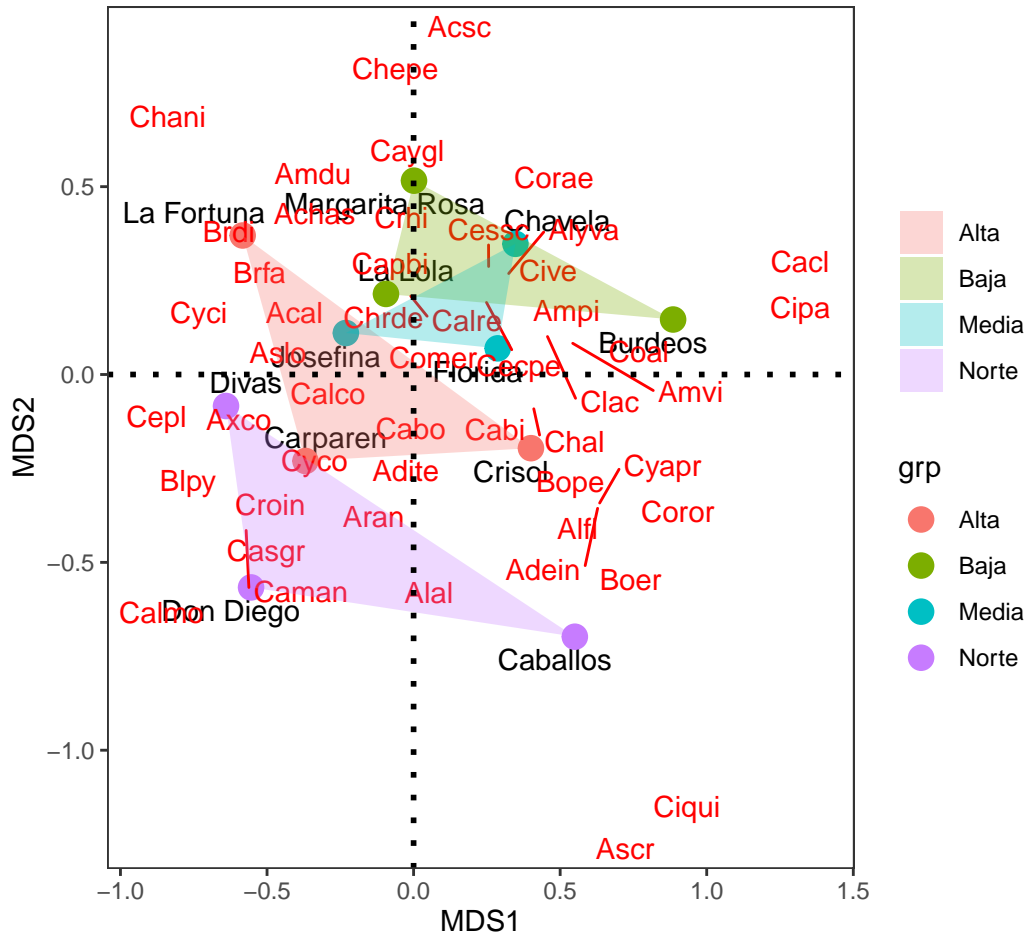
La siguiente figura presenta los comandos que se organizan por sitios, especies y el factor Región.

```
x11()
ggplot() +
# Sitios
  geom_text_repel(data = coord.sit,aes(MDS1,MDS2,label = as.character(datos$Finca)),
                  size=4)+ # Muestra el cuadro de la figura
  geom_point(data = coord.sit,aes(MDS1,MDS2,colour=grp),size=4)+
  scale_shape_manual(values = c(21:25))+

# Especies
  geom_segment(data = coord.tax,aes(x = 0, y = 0, xend = MDS1, yend = MDS2),
              arrow = arrow(angle=0,length = unit(0,"cm"),
                           type = "closed"),linetype=0, size=0,colour = "red")+
  geom_text_repel(data = coord.tax,aes(MDS1,MDS2,label=especies),colour = "red")+

#Factor
  geom_polygon(data=coord.sit,aes(x=MDS1,y=MDS2,fill=grp,group=grp),alpha=0.30) +
  geom_hline(yintercept=0,linetype=3,size=1) +
  geom_vline(xintercept=0,linetype=3,size=1)+
  guides(shape=guide_legend(title=NULL,color="black"),
         fill=guide_legend(title=NULL))+
  theme_bw()+theme(panel.grid=element_blank())
```





### 27.0.6 Taller de entrenamiento

**Objetivo:** Poner en práctica los conceptos vistos en este taller, realizando las siguientes opciones realizando un NMDS con las variables biológicas del estudio de caso (taxones). Enviar los resultados al *Teams* del profesor.

## 28 Taller 6.1 Análisis de Correspondencias Múltiples - MCA

### Objetivo de la actividad:

El siguiente ejercicio analizará los datos de un proyecto de *SEPEC (2021)*, basado en 100 registros tomados aleatoriamente de encuestas realizadas a pescadores y comercializadores de diferentes especies de bagres en Colombia.

El **objetivo** de este ejercicio consiste en valorar la relación entre las variables categóricas producto de las encuestas y la información relacionada a la comercialización de los bagres censados. Se utilizará el siguiente archivo como base de datos: **bagres.xlsx**

Ejercicio tomado de: Rodríguez-Barrios (2023) [Enlace del libro](#)

[Enlace de los archivos del libro](#)

### 28.0.1 Procedimiento de la exploración

**Análisis de Correspondencia múltiple (MCA).** A partir de la muestra de 100 registros de bagres y de variables cualitativas o categóricas obtenidas mediante las encuestas realizadas, se identifican los siguientes elementos de los datos seleccionados:

- **Individuos activos:** Filas de la base de datos (100 registros de bagres).
- **Variables activas:** Variables categóricas que se utilizarán en el primer mca, que corresponden a las categóricas que han sido encuestadas.
- **Variables cuantitativas suplementarias (quanti.sup):** Son las variables cuantitativas que presenta la base de datos de bagres (venta en kg y precio de venta de los bagres).
- **Variables cualitativas suplementarias (quali.sup):** corresponden a las que se requieran analizar por separado, en este caso serán los nombres vernaculares de los bagres. Las variables cuantitativas y cualitativas suplementarias serán evaluadas al final del ejercicio con un mca adicional.

Más detalles de este procedimiento se pueden revisar en el siguiente enlace: [MCA - Multiple Correspondence Analysis in R](#)

### 28.0.1.1 Librerías requeridas

```
# Librerías requeridas
library(tidyverse)
library(xtable)      # Importar y exportar
library(openxlsx)    # exportar "*.xlsx"
library(readxl)      # Importar y exportar

library(FactoMineR)  # Para realizar el MCA
library(factoextra)  # Para realizar el MCA
library(dplyr)       # Para pasar variables a factor
```

### 28.0.1.2 Cargar o importar la base de datos

```
# Base de datos
bagres <- read_excel("bagres.xlsx")  # paquete "readxl"
head(bagres)
str(bagres)
View(bagres)
```

### 28.0.2 1) Ajuste de la base de datos de bagres.

Para la realización del primer mca, que solo incluirá a las variables activas (excluye a las suplementarias), se escogerán solo las variables categóricas requeridas para este análisis (columnas 1, 9 a la 22).

```
# Base de variables activas
bagres <- read_excel("bagres.xlsx")  # paquete "readxl"
datos.activos = bagres[,c(1,9:22)]  # selección de columnas 1, 9 a 22
View(datos.activos)
```

Al analizar la estructura de la base `datos activos`, a excepción de las columnas 4 y 5 (variables cuantitativas o cuantitativas suplementarias) `Venta.kg` y `Precio.venta`, el resto son de tipología carácter (`chr`) y se deben pasar a factores, para que el MCA pueda ejecutarse de forma apropiada. Es importante aclarar que las columnas 1 a la 4 no corresponden a variables activas, pero serán tabuladas en el siguiente `data.frame`.

```
# Cambiar todas las variables cualitativas a factor
datos.activos <- datos.activos %>%
```

```
mutate_all(factor) # Pasar a factores excepto variables 5 y 6
print(head(datos.activos))
```

```
# A tibble: 6 x 15
  ...1 Importa~1 Origen Destino Tipo.p~2 Provee~3 Sit.c~4 Frec.~5 Trans~6 Conserv
  <fct> <fct> <fct> <fct> <fct> <fct> <fct> <fct> <fct> <fct>
1 1.Psdo Importa~ Orige~ Destin~ Tipo.p~ Provee~ Sitio.~ Frec.c~ Transf~ Conser~
2 2.Psdo Importa~ Orige~ Destin~ Tipo.p~ Provee~ Sitio.~ Frec.c~ Transf~ Conser~
3 3.Psdm Importa~ Orige~ Destin~ Tipo.p~ Provee~ Sitio.~ Frec.c~ Transf~ Conser~
4 4.Psdp Importa~ Orige~ Destin~ Tipo.p~ Provee~ Sitio.~ Frec.c~ Transf~ Conser~
5 5.Psdp Importa~ Orige~ Destin~ Tipo.p~ Provee~ Sitio.~ Frec.c~ Transf~ Conser~
6 6.Psdm Importa~ Orige~ Destin~ Tipo.p~ Provee~ Sitio.~ Frec.c~ Transf~ Conser~
# ... with 5 more variables: Empaque <fct>, Transporte <fct>,
# Cliente.prim. <fct>, Cliente.sec. <fct>, Cliente.ter. <fct>, and
# abbreviated variable names 1: Importado, 2: Tipo.prod, 3: Proveedor,
# 4: Sit.comp, 5: Frec.compra, 6: Transform
```

```
View(datos.activos)
```

### 28.0.3 2) Primera ordenación de las variables cualitativas activas (mca1).

Las variables consideradas para esta ordenación, son las cualitativas (tipo factor) que pueden ejercer un efecto en la comercialización de los bagres.

```
# 2) Ordenación de las variables acualitativas activas
# Las columnas 5 a 18 son las requeridas por el mca
str(datos.activos)
```

```
tibble [100 x 15] (S3: tbl_df/tbl/data.frame)
 $ ...1      : Factor w/ 100 levels "1.Psdo","10.Psdo",...: 1 13 24 35 46 57 68 79 90 2 ..
 $ Importado  : Factor w/ 2 levels "Importado.F",...: 1 1 1 1 2 1 1 1 1 1 ...
 $ Origen     : Factor w/ 2 levels "Origen.i","Origen.n": 2 2 2 2 1 2 2 2 2 2 ...
 $ Destino    : Factor w/ 2 levels "Destino.c","Destino.i": 2 2 2 2 2 2 2 2 2 2 ...
 $ Tipo.prod  : Factor w/ 1 level "Tipo.prod.p": 1 1 1 1 1 1 1 1 1 1 ...
 $ Proveedor  : Factor w/ 3 levels "Proveedor.i",...: 1 1 1 1 3 1 1 1 1 1 ...
 $ Sit.comp   : Factor w/ 6 levels "Sit.comp.c","Sit.comp.o",...: 3 3 3 3 6 3 3 3 4 4 ...
 $ Frec.compra : Factor w/ 3 levels "Frec.compra.d",...: 3 3 1 3 3 3 3 3 3 3 ...
 $ Transform  : Factor w/ 3 levels "Transform.c",...: 2 2 2 1 2 1 2 1 1 1 ...
 $ Conserv    : Factor w/ 2 levels "Conserv.c","Conserv.sa": 1 1 1 1 1 1 1 1 1 1 ...
```

```

$ Empaque      : Factor w/ 3 levels "Empaque.b","Empaque.cc",...: 1 1 1 1 3 1 1 1 1 1 ...
$ Transporte   : Factor w/ 3 levels "Transporte.m",...: 3 3 2 3 3 3 3 2 3 3 ...
$ Cliente.prim.: Factor w/ 3 levels "Cliente.prim.c",...: 1 3 1 1 3 1 1 1 1 1 ...
$ Cliente.sec. : Factor w/ 5 levels "Cliente.sec.c",...: 2 2 2 4 3 4 2 2 2 2 ...
$ Cliente.ter. : Factor w/ 5 levels "Cliente.ter.c",...: 3 4 3 3 3 3 4 3 3 3 ...

```

```

View(datos.activos[,c(2:15)])
mca1 <- MCA(datos.activos[,c(2:15)], graph = FALSE)
summary(mca1)

```

Call:

```
MCA(X = datos.activos[, c(2:15)], graph = FALSE)
```

#### Eigenvalues

	Dim.1	Dim.2	Dim.3	Dim.4	Dim.5	Dim.6	Dim.7
Variance	0.391	0.197	0.172	0.161	0.139	0.111	0.105
% of var.	18.874	9.525	8.307	7.768	6.700	5.348	5.073
Cumulative % of var.	18.874	28.399	36.706	44.474	51.174	56.522	61.596
	Dim.8	Dim.9	Dim.10	Dim.11	Dim.12	Dim.13	Dim.14
Variance	0.102	0.091	0.087	0.077	0.068	0.062	0.052
% of var.	4.928	4.404	4.197	3.712	3.291	3.000	2.504
Cumulative % of var.	66.523	70.928	75.125	78.837	82.128	85.128	87.632
	Dim.15	Dim.16	Dim.17	Dim.18	Dim.19	Dim.20	Dim.21
Variance	0.046	0.040	0.037	0.029	0.025	0.020	0.018
% of var.	2.243	1.937	1.774	1.381	1.223	0.958	0.850
Cumulative % of var.	89.875	91.812	93.585	94.966	96.189	97.147	97.997
	Dim.22	Dim.23	Dim.24	Dim.25	Dim.26	Dim.27	Dim.28
Variance	0.015	0.009	0.008	0.005	0.003	0.001	0.000
% of var.	0.728	0.453	0.395	0.251	0.148	0.027	0.000
Cumulative % of var.	98.726	99.179	99.574	99.825	99.973	100.000	100.000
	Dim.29						
Variance	0.000						
% of var.	0.000						
Cumulative % of var.	100.000						

#### Individuals (the 10 first)

	Dim.1	ctr	cos2	Dim.2	ctr	cos2	Dim.3	ctr
1	-0.171	0.075	0.086	-0.077	0.030	0.017	0.022	0.003
2	-0.066	0.011	0.004	-0.427	0.925	0.180	0.265	0.407

3		-0.024	0.002	0.000		0.116	0.068	0.009		0.325	0.614
4		-0.342	0.299	0.220		-0.037	0.007	0.003		-0.305	0.542
5		1.905	9.283	0.692		-0.242	0.297	0.011		-0.472	1.293
6		-0.342	0.299	0.220		-0.037	0.007	0.003		-0.305	0.542
7		-0.207	0.109	0.065		-0.157	0.124	0.037		0.200	0.233
8		-0.300	0.230	0.090		-0.046	0.011	0.002		0.053	0.016
9		-0.310	0.245	0.111		0.071	0.025	0.006		-0.250	0.363
10		-0.310	0.245	0.111		0.071	0.025	0.006		-0.250	0.363
cos2											
1		0.001									
2		0.069									
3		0.073									
4		0.175									
5		0.042									
6		0.175									
7		0.061									
8		0.003									
9		0.072									
10		0.072									

Categories (the 10 first)

	Dim.1	ctr	cos2	v.test	Dim.2	ctr	cos2	v.test
Importado.F	-0.241	0.984	0.770	-8.730	-0.013	0.005	0.002	-0.457
Importado.v	3.198	13.079	0.770	8.730	0.167	0.071	0.002	0.457
Origen.i	3.198	13.079	0.770	8.730	0.167	0.071	0.002	0.457
Origen.n	-0.241	0.984	0.770	-8.730	-0.013	0.005	0.002	-0.457
Destino.c	0.126	0.012	0.001	0.255	2.133	6.589	0.190	4.332
Destino.i	-0.005	0.000	0.001	-0.255	-0.089	0.275	0.190	-4.332
Tipo.prod.p	0.000	0.000	NaN	NaN	0.000	0.000	NaN	NaN
Proveedor.i	-0.250	1.018	0.506	-7.080	-0.112	0.404	0.102	-3.170
Proveedor.o	-0.571	0.179	0.010	-0.999	1.092	1.295	0.037	1.911
Proveedor.pa	2.997	13.129	0.781	8.794	0.837	2.027	0.061	2.455
	Dim.3	ctr	cos2	v.test				
Importado.F	0.083	0.264	0.091	2.997				
Importado.v	-1.098	3.503	0.091	-2.997				
Origen.i	-1.098	3.503	0.091	-2.997				
Origen.n	0.083	0.264	0.091	2.997				
Destino.c	1.601	4.255	0.107	3.251				
Destino.i	-0.067	0.177	0.107	-3.251				
Tipo.prod.p	0.000	0.000	Inf	-Inf				
Proveedor.i	0.015	0.008	0.002	0.412				
Proveedor.o	-0.005	0.000	0.000	-0.009				
Proveedor.pa	-0.160	0.085	0.002	-0.470				

```
Categorical variables (eta2)
      Dim.1 Dim.2 Dim.3
Importado | 0.770 0.002 0.091 |
Origen    | 0.770 0.002 0.091 |
Destino   | 0.001 0.190 0.107 |
Tipo.prod | 0.000 0.000 0.000 |
Proveedor | 0.784 0.103 0.002 |
Sit.comp  | 0.798 0.317 0.338 |
Frec.compra | 0.350 0.235 0.107 |
Transform | 0.303 0.249 0.181 |
Conserv   | 0.047 0.024 0.102 |
Empaque   | 0.494 0.321 0.398 |
```

El anterior insumo es importante, porque define el porcentaje de varianza que capturan los 29 ejes canónicos - **Eigenvalues** y selecciona a las 10 variables de mayor relevancia para el análisis de ordenación - **mca Categories (the 10 first)** en los tres primeros ejes canónicos - **Categorical variables (eta2)**.

### 28.0.3.1 2.1) Ajuste de la ordenación definida por los autovalores

A continuación se calcula el componente tabular, para identificar la varianza que captura cada eje canónico o Dim.i, en donde i es cada uno de los ejes de la ordenación.

```
# # Matriz de autovalores de los seis primeros ejes canónicos
head(mca1$eig)
```

	eigenvalue	percentage of variance	cumulative percentage of variance
dim 1	0.3909689	18.874360	18.87436
dim 2	0.1972939	9.524535	28.39889
dim 3	0.1720808	8.307348	36.70624
dim 4	0.1609039	7.767773	44.47401
dim 5	0.1387903	6.700220	51.17423
dim 6	0.1107851	5.348244	56.52248

En la Figura 32.2, se grafican los resultados de la tabla anterior.

```
# Figura de autovalores (para la escogencia de variables)
x11()
fviz_screplot(mca1, addlabels = TRUE, ylim = c(0, 20),
```

```
ylab = "% Varianza explicada", xlab = "Dimensiones",
col="steelblue")
```

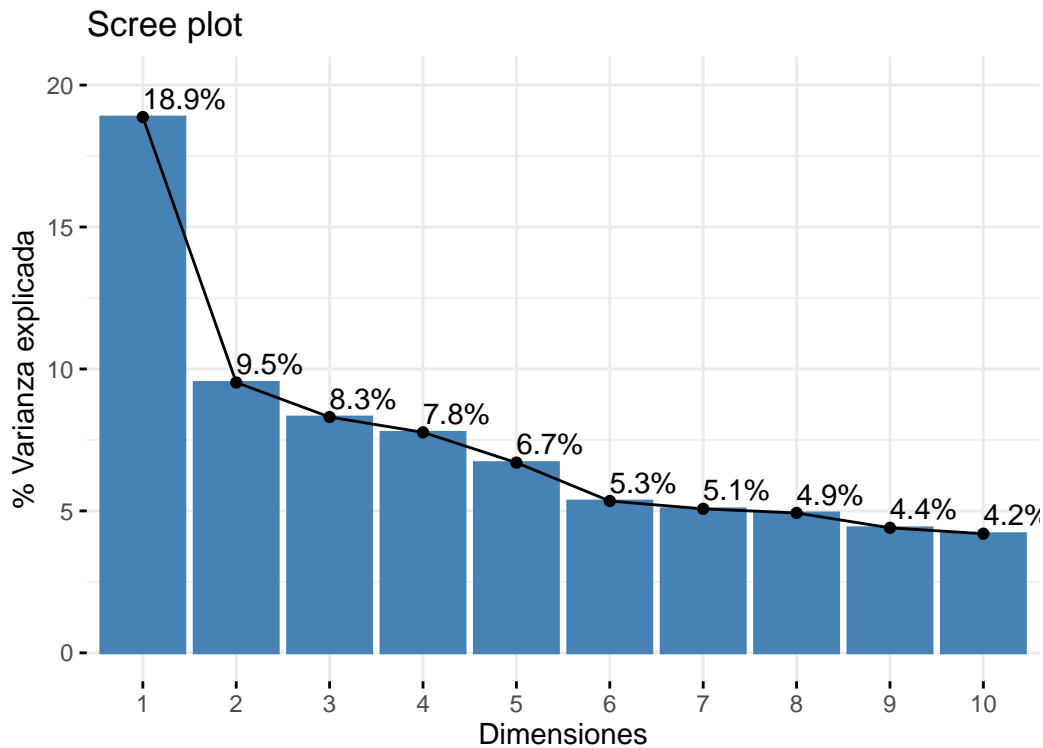


Figura 28.1: Varianza capturada por cada eje canónico

#### 28.0.3.2 2.2) Figuras generales del mca1

La Figura 32.3 muestra la relación de las 14 variables categoricas del análisis, el nivel de relación o cercanía es definido por la distancia chi cuadrado. `fviz_mca_var` representa a la grafica de variables, `mca.cor` muestra solo a las variables que más contribuyen a la ordenación, `repel = TRUE` permite visualizar los elementos sin solapamientos. `ior`.

```
# Figura de relación de las variables categóricas
x11()
fviz_mca_var(mca1, choice = "mca.cor", repel = TRUE,
ggtheme = theme_minimal())
```



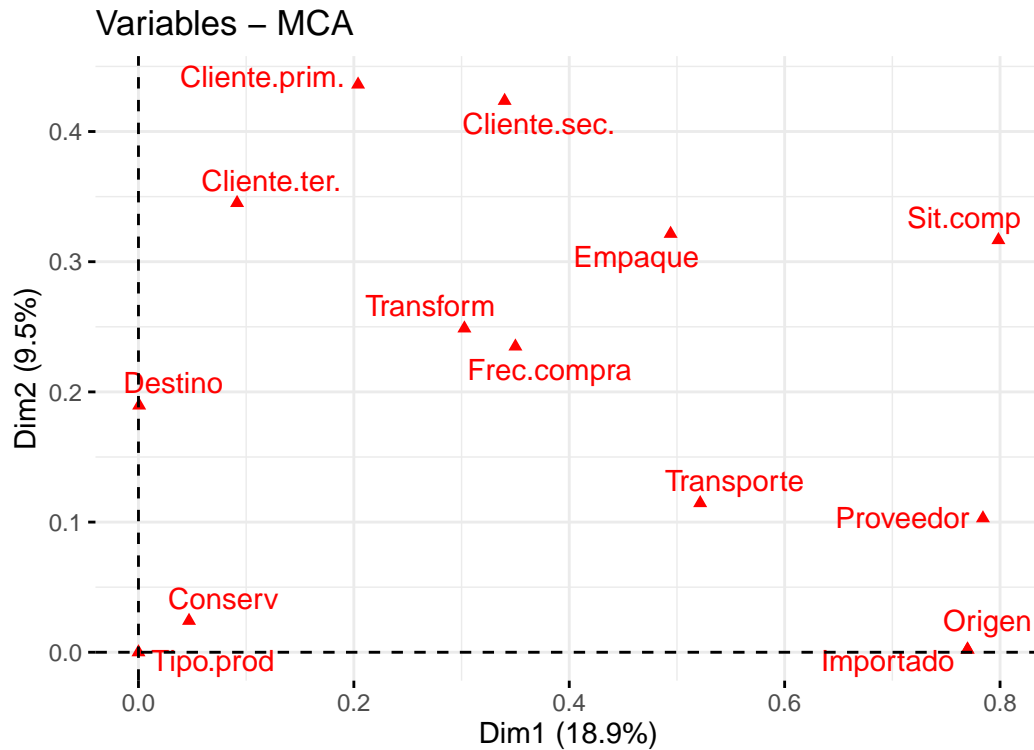


Figura 28.2: Ordenación de las variables que más contribuyen al análisis

La Figura 32.4 muestra el biplot integrado por los 100 registros de peces y las variables categóricas que los caracterizan. `fviz_mca_biplot` permite visualizar a las observaciones o filas de la base de datos (números en azul) y a las variables con mayor contribución al análisis (en rojo).

```
# Figura del Biplot de ordenación registros de peces y de variables
x11()
fviz_mca_biplot(mca1, choice = "mca.cor", repel = TRUE,
  ggtheme = theme_minimal())
```

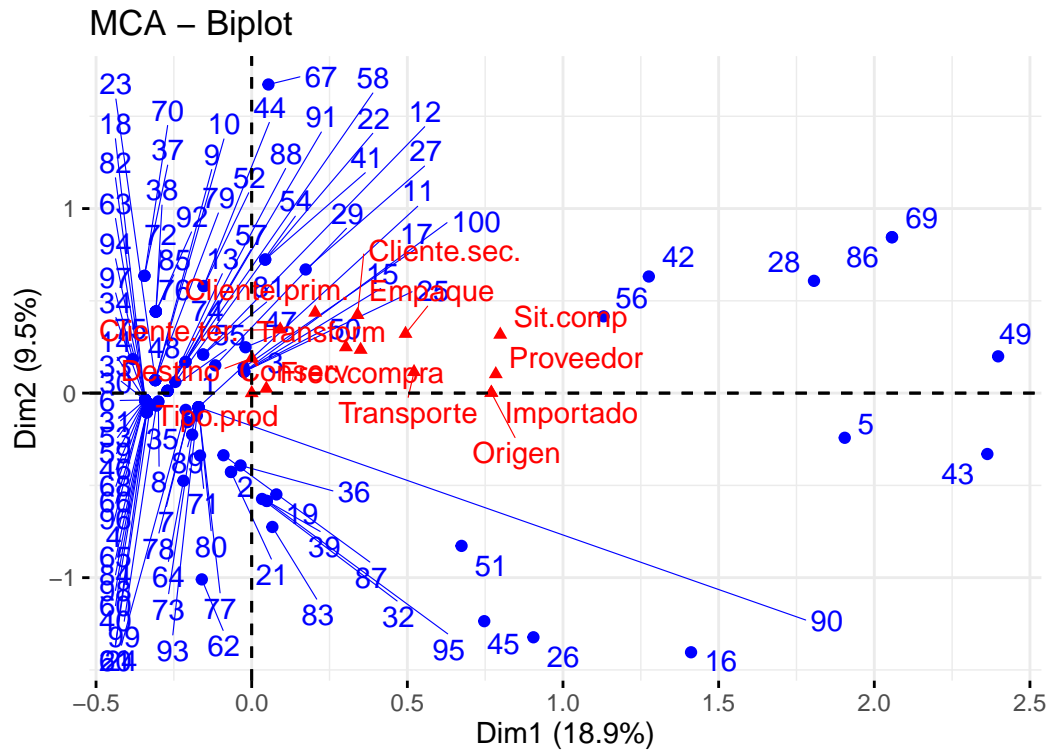


Figura 28.3: Ordenación de las variables que más contribuyen al análisis y de las observaciones (registros de peces)

La Figura 31.5 define a 100 registros de peces con todas las variables activas y sus categorías (no incluye al comando “mca.cor”).

```
# Figura del Biplot de ordenación para registros de peces y categoricas de las variables
x11()
fviz_mca_biplot(mca1, repel = TRUE,
                 ggtheme = theme_minimal())
```

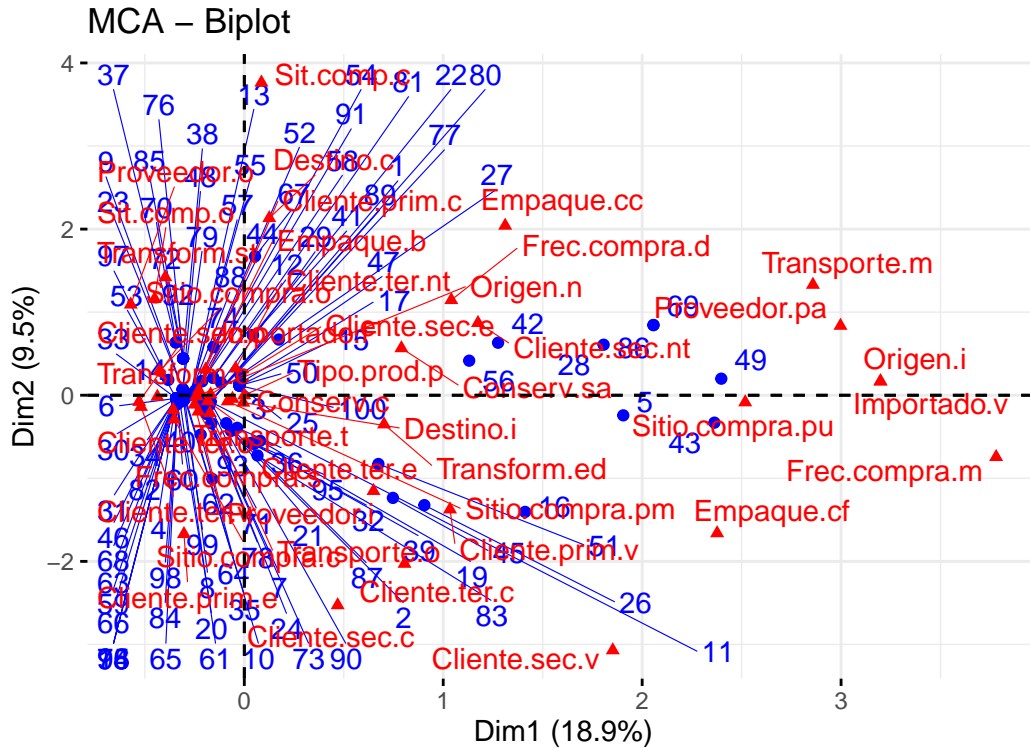


Figura 28.4: Ordenación de todas las variables activas y de las observaciones (registros de peces)

### 28.0.3.3 2.3) Figuras del mca con ponderaciones

A continuación se visualiza la ordenación de las observaciones (100 registros de peces) y su nivel de importancia (observaciones en rojo son las más importantes). `fviz_mca_ind`, permite visualizar a los individuos u observaciones (Figura 31.6).

```
# Contribuciones de las observaciones (filas de la la base de bagres)
x11()
fviz_mca_ind(mca1, col.ind = "cos2",
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             repel = TRUE,
             ggtheme = theme_minimal())
```

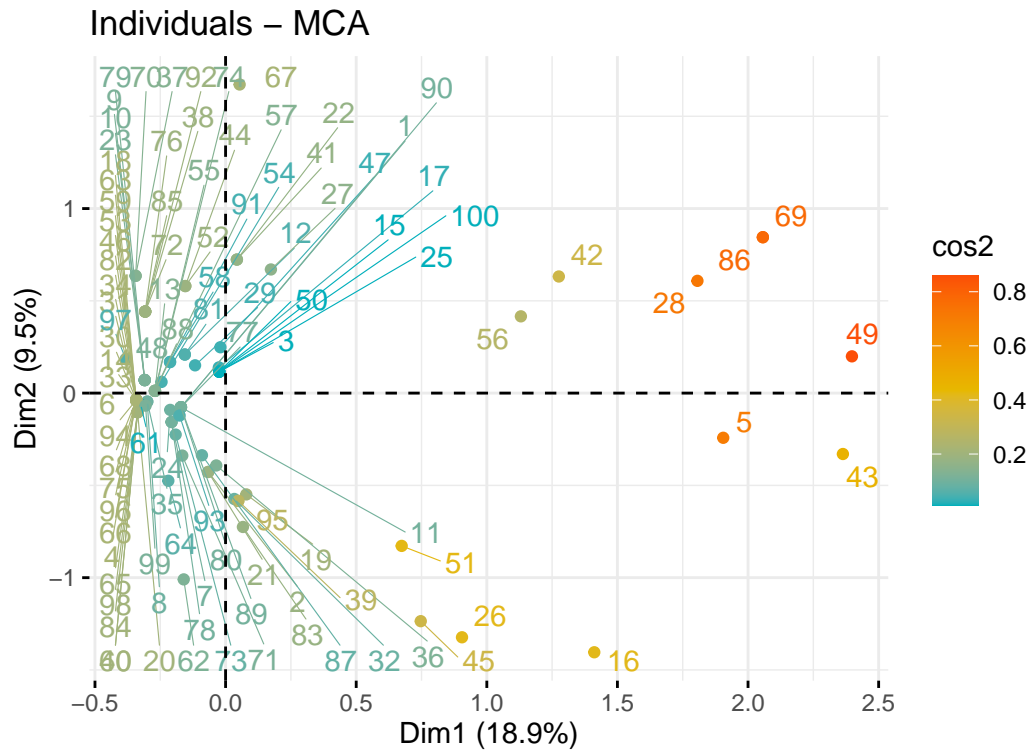


Figura 28.5: Ordenación de las observaciones (registros de peces) y su nivel de contribución al análisis.

En la Figura 31.7 se muestra la ordenación de las variables activas y su nivel de contribución (variables en rojo son las más importantes).

```
# Contribuciones de las variables categóricas (columnas de variables activas)
x11()
fviz_mca_var(mca1, col.var = "cos2",
  gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
  repel = TRUE, ggtheme = theme_minimal())
```

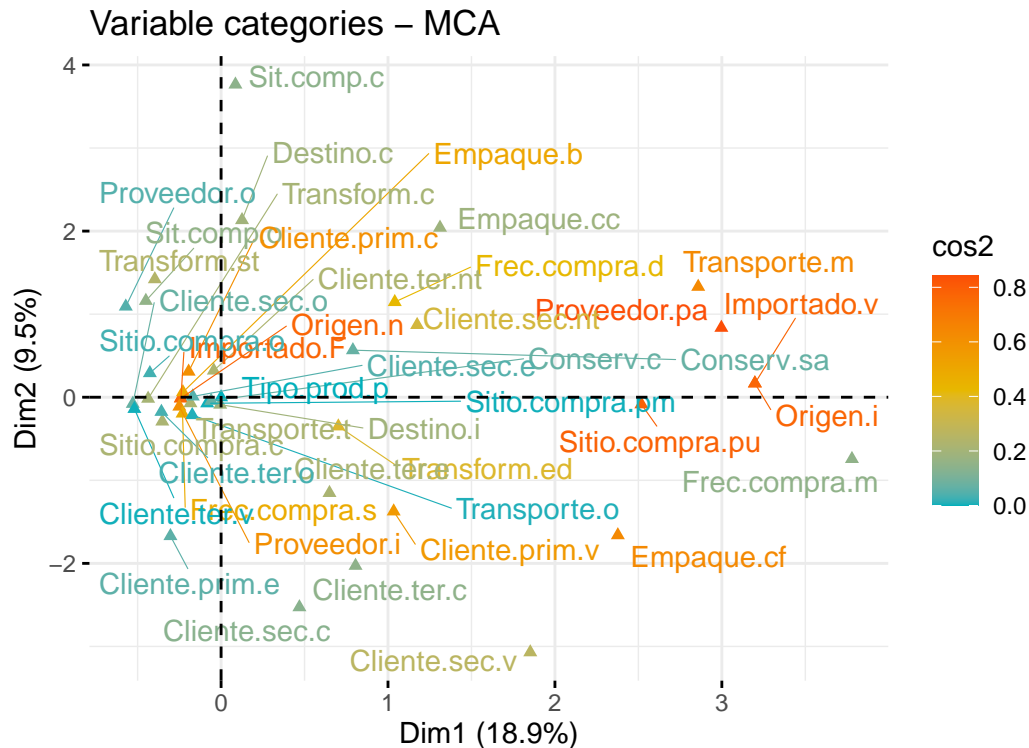


Figura 28.6: Ordenación de las variables activas y su nivel de contribución al análisis.

### 28.0.4 3) Segunda ordenación de las variables cualitativas activas (mca2).

A continuación se realiza un mca con dos elementos adicionales a a las variables activas ordenadas anteriormente:

- **Variables cuantitativas suplementarias (quanti.sup)**, correspondientes a la venta en kilogramos y el precio de venta de los bagres.
- **Variables cualitativas suplementarias (quali.sup)**, representadas por los nombres o tipos de bagres (columna: Nombre.vernacul

Esto con el objetivo de visualizar aspectos asociados a la comercialización de los bagres (variables cuantitativas y tipos de bagres), en respuesta a las variables categóricas producto de las encuestas (variables activas).

**Paso 1.** Incluir los nombres de los bagres a la base “datos.activos” (datos.activos.2)

```
# Pasar nombres de los bagres a factor
bagres <- read_excel("bagres.xlsx") # paquete "readxl"
bagres$Nombre.vernacular = as.factor(bagres$Nombre.vernacular)

# Crear data frame con los nombres de los bagres y variables cuantitativas.
datos.activos.1 <- data.frame(bagres = bagres[,6],
                              venta.kg = bagres[,7],
                              Precio.venta = bagres[,8],
                              datos.activos[,2:15])

str(datos.activos.1)
```

```
'data.frame': 100 obs. of 17 variables:
 $ Nombre.vernacular: Factor w/ 4 levels "B.Ray.o.Tigr",...: 4 4 3 2 2 2 2 4 4 4 ...
 $ Venta.kg          : num 150 1000 150 50 2200 20 30 20 50 35 ...
 $ Precio.venta      : num 18000 14000 16000 19000 13000 24000 24000 22000 22000 18000 ...
 $ Importado         : Factor w/ 2 levels "Importado.F",...: 1 1 1 1 2 1 1 1 1 1 ...
 $ Origen            : Factor w/ 2 levels "Origen.i","Origen.n": 2 2 2 2 1 2 2 2 2 2 ...
 $ Destino           : Factor w/ 2 levels "Destino.c","Destino.i": 2 2 2 2 2 2 2 2 2 2 ...
 $ Tipo.prod         : Factor w/ 1 level "Tipo.prod.p": 1 1 1 1 1 1 1 1 1 1 ...
 $ Proveedor         : Factor w/ 3 levels "Proveedor.i",...: 1 1 1 1 3 1 1 1 1 1 ...
 $ Sit.comp          : Factor w/ 6 levels "Sit.comp.c","Sit.comp.o",...: 3 3 3 3 6 3 3 3 4 4 ...
 $ Frec.compra       : Factor w/ 3 levels "Frec.compra.d",...: 3 3 1 3 3 3 3 3 3 3 ...
 $ Transform         : Factor w/ 3 levels "Transform.c",...: 2 2 2 1 2 1 2 1 1 1 ...
 $ Conserv           : Factor w/ 2 levels "Conserv.c","Conserv.sa": 1 1 1 1 1 1 1 1 1 1 ...
 $ Empaque           : Factor w/ 3 levels "Empaque.b","Empaque.cc",...: 1 1 1 1 3 1 1 1 1 1 ...
 $ Transporte        : Factor w/ 3 levels "Transporte.m",...: 3 3 2 3 3 3 3 2 3 3 ...
 $ Cliente.prim.     : Factor w/ 3 levels "Cliente.prim.c",...: 1 3 1 1 3 1 1 1 1 1 ...
 $ Cliente.sec.      : Factor w/ 5 levels "Cliente.sec.c",...: 2 2 2 4 3 4 2 2 2 2 ...
 $ Cliente.ter.      : Factor w/ 5 levels "Cliente.ter.c",...: 3 4 3 3 3 3 4 3 3 3 ...
```

**Paso 2.** Nuevo mca, que incluye a las variables suplementarias (quali.sup y quanti.sup)

```
# mca con nombres de los bagres (quali.sup) y variables cuantitativas (cuanti sub)
mca2 <- MCA (datos.activos.1,
             quali.sup = 1,           # Registros de peces (X)
             quanti.sup = 2:3,       # 2 y 3 son Variables cuantitativas
             graph=FALSE)
```

**Paso 3.** Gráfica de las variables cuantitativas suplementarias (quanti.sup) (Figura 31.8).

```
# Figura de las variables suplementarias (azul) y las activas (rojo)
x11()
fviz_mca_var(mca2,
             choice = "mca.cor",      # Principales variables cualitativas
             repel = TRUE)            # Quita el solapamiento de las variables.
```

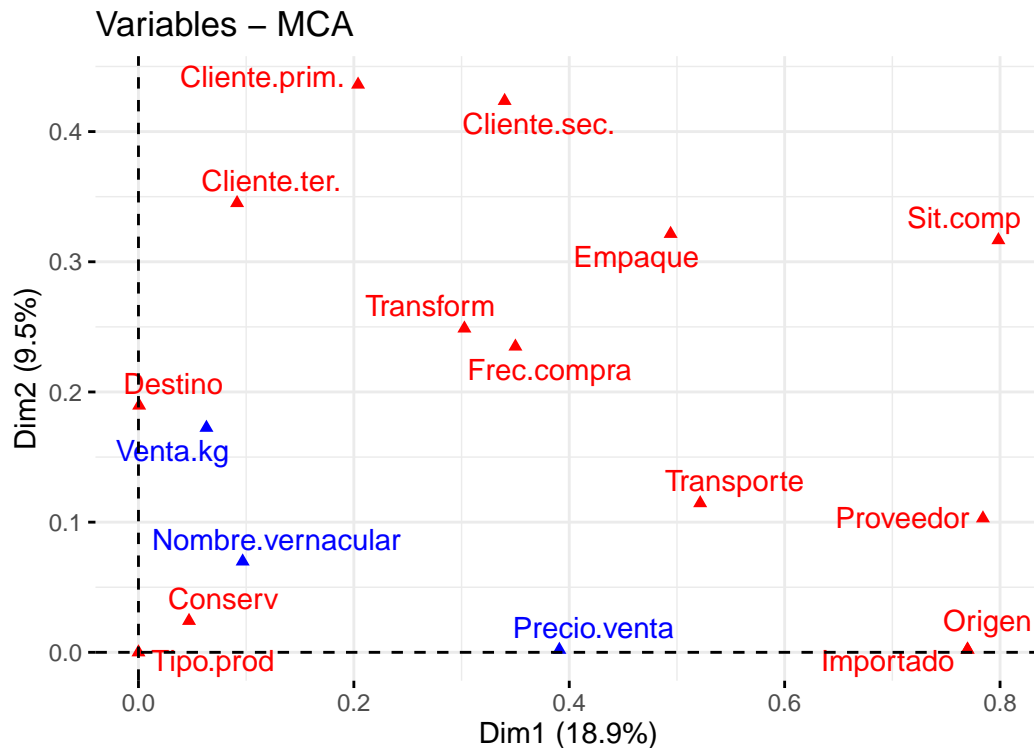


Figura 28.7: Ordenación de las variables activas y su nivel de contribución al análisis.

#### Paso 4. Grafica de las variables cualitativas suplementarias (quali.sup) (Figura 30.8)

```
# Figura de las variables suplementarias (verde), las activas (rojo) y los individuos (azul)
x11()
fviz_mca_biplot(mca2, repel = TRUE,
               geom.ind = c("n", "n"),      # No mostrar a los individuos
               ggtheme = theme_bw())        # Puede usar temas de "ggplot2"
```

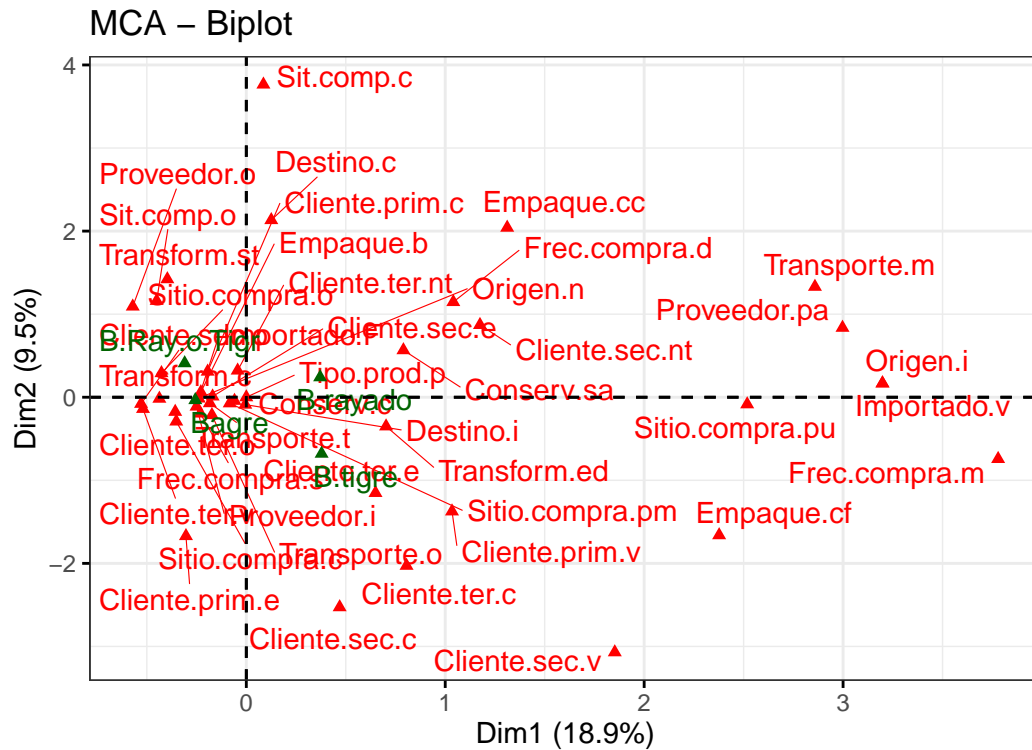


Figura 28.8: Ordenación de todas las variables activas que más contribuyen al análisis (rojo) y su relación co las variables cualitativas suplementarias (verde).

La Figura 30.9 muestra las elipses que representan a los intervalos de confianza de cada grupo de bagres, para medir el tipo de relación o de colapamiento entre estos, basado en las variables activas que los caracterizan.

```
# Elipses por cada tipo de bagre
x11()
fviz_ellipses(mca2, "Nombre.vernacular",
              geom = "text", repel = TRUE)
```



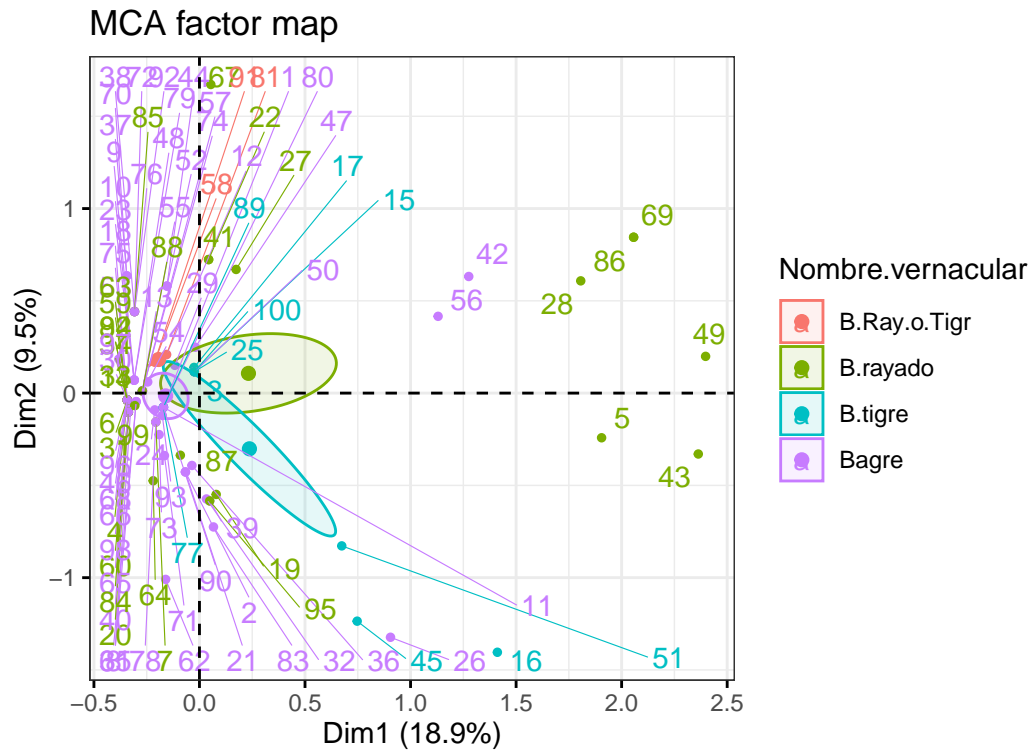


Figura 28.9: Relación entre las especies de bagres y con sus observaciones (registros de peces).

### 28.0.5 Taller de entrenamiento

**Objetivo:** Poner en práctica los conceptos vistos en este taller, realizando las siguientes opciones realizando un MCA con las variables biológicas (taxones). Enviar los resultados al *Teams* del profesor.

## 29 Taller 7.1 Análisis de Redundancia - RDA

### Objetivo de la actividad:

La siguiente base de datos es tomada del trabajo de (Osorio, 2021), relacionado a un estudio sobre la composición de microalgas de la ciénaga Sevillano en el complejo lagunar de la Ciénaga Grande de Santa Marta (Colombia). La información contiene a 21 géneros de microalgas (matriz Y) y 10 variables ambientales (matriz X) medidas en 24 observaciones (localidades y campañas de muestreo). El propósito del ejercicio consiste en determinar la relación entre la composición de las microalgas y las variables fisicoquímicas de su ambiente, aplicando un análisis de redundancia (RDA) y un Análisis de Correspondencia Canónica (ACC), para finalmente comparar la aplicación de cada técnica. Se utilizará el siguiente archivo: **Microalgas.csv**

Ejercicio tomado de: Rodríguez-Barrios (2023) [Enlace del libro](#)

[Enlace de los archivos del libro](#) Revisar el capítulo de Análisis de Redundancia - RDA

[Numerical Ecology With R - Borcard et al. 2018](#) Capítulo de Análisis de Redundancia - RDA

### 29.0.0.1 Procedimiento resumido de la ordenación con el RDA

- Cargar librerías y funciones requeridas
- Cargar la base `Microalgas.csv`
- Realizar los ajustes a las variables y factores
- Correr el RDA con todas las variables
- Correr el RDA con las variables ambientales seleccionadas
- Figuras de BILOT y TRILOT con librerías "vegan" y "ggplot2".

#### 29.0.0.2 Cargar las librerías requeridas

```
# Librerías requeridas
library(ade4)
library(adegraphics)
library(adespatial)
library(cocorresp)
library(vegan)
library(MASS)
library(ellipse)
library(FactoMineR)
library(rrcov)
library(ggplot2)
library(reshape2)
library(ggrepel)
library(ggforce)
```

#### 29.0.0.3 Funciones adicionales (Bordcard et al. 2018)

```
# Funciones a cargar
source("hcoplot.R")
source("triplot.rda.R")
source("plot.lda.R")
source("polyvars.R")
source("screestick.R")
```

#### 29.0.0.4 Cargar o importar la base de datos

Esta base de datos cuenta con una variable agrupadora o factor (Tributario), 10 variables ambientales y 21 taxones de microalgas.

```
# Base de datos
datos = read.csv2("Microalgas.csv",row.names=1)
```

#### 29.0.0.5 Ajuste de las bases de datos biológica (tax.hel) y Ambiental (amb)

A continuación se realizará un ajuste de la base de datos, primero convirtiendo a la columna Tributario como un factor, luego transformando a las variables ambientales amb con logaritmo

en base 10 y finalmente ajustando a los taxones `tax.hel` con la transformación de Hellinger. Las abreviaturas en las filas T1.1, ..., T1.6, ... Representan el número del tributario (T1) y el numero de la visita realizada al lugar de muestreo (1).

```
# Ajuste de factores
datos$Tributario = as.factor (datos$Tributario)
# str(datos)      # Nueva estructura de la base de datos

# Variables ambientales
amb=(datos[,c(2:11)]+1)
round(head(amb),2)
```

	Amonio	Nitrito	Nitrato	Oxigeno	pH	Conductividad	Caudal	Vel_Corriente
T1.1	1.30	1.84	1.90	8.68	9.10	77	1.51	1.73
T1.2	1.30	1.78	1.83	7.54	8.45	77	2.59	1.37
T1.3	2.11	2.18	4.43	6.62	8.81	77	2.48	2.15
T1.4	2.02	1.88	3.67	7.08	10.21	77	2.24	2.33
T1.5	1.19	1.15	1.52	6.10	10.24	77	2.32	2.32
T1.6	1.21	1.47	1.90	6.60	10.63	78	2.25	2.29

	Luz	Temp
T1.1	801	18.6
T1.2	401	19.3
T1.3	301	18.1
T1.4	101	19.6
T1.5	801	18.9
T1.6	201	18.8

Los datos de abundancia de los taxones están en cifras decimales, debido a la transformación logarítmica que se les aplicó.

```
# Variables biológicas linealizadas - Taxones con Hellinger
tax.hel=decostand(datos[,c(12:32)],"hellinger")
round(head(tax.hel),2)
```

	Fragillaria	Lyngbya	Chamaepinnularia	Achnantes	Amphora	Caloneis	Closterium
T1.1	0.00	0.62		0.10	0.00	0.15	0.23
T1.2	0.10	0.49		0.07	0.00	0.12	0.22
T1.3	0.34	0.16		0.20	0.00	0.00	0.18
T1.4	0.31	0.21		0.23	0.05	0.07	0.11
T1.5	0.28	0.14		0.22	0.08	0.10	0.09
T1.6	0.25	0.18		0.19	0.09	0.11	0.06

	Cocconeis	Cymbella	Eolimna	Epithemia	Eunotia	Frustulia	Girosigma
T1.1	0.22	0	0.27	0.00	0.00	0.00	0.00
T1.2	0.30	0	0.33	0.00	0.17	0.17	0.00
T1.3	0.33	0	0.32	0.00	0.09	0.11	0.00
T1.4	0.29	0	0.26	0.05	0.00	0.00	0.15
T1.5	0.44	0	0.29	0.07	0.10	0.00	0.16
T1.6	0.44	0	0.28	0.08	0.09	0.00	0.13
	Gomphonema	Melosira	Navicula	Nitzschia	Planothidium	Surirella	Pinnularia
T1.1	0.00	0.51	0.00	0.24	0.15	0.10	0.25
T1.2	0.00	0.49	0.20	0.29	0.17	0.00	0.19
T1.3	0.09	0.50	0.11	0.37	0.32	0.00	0.22
T1.4	0.18	0.49	0.20	0.33	0.33	0.12	0.20
T1.5	0.20	0.43	0.20	0.25	0.35	0.19	0.12
T1.6	0.19	0.49	0.20	0.21	0.31	0.19	0.09

## 29.0.1 Doce pasos para el análisis de redundancia - RDA.

### 29.0.1.1 Paso 1. Ordenación de los taxones y las variables ambientales.

En el siguiente análisis se relaciona a la matriz de datos biológicos (abundancia de taxones) con la matriz de datos ambientales. A continuación se determinan los insumos generales del análisis.

```
# 1. Realización del RDA
tax.rda<-rda(tax.hel ~.,amb)
tax.rda # Resultados resumidos
```

```
Call: rda(formula = tax.hel ~ Amonio + Nitrito + Nitrato + Oxigeno + pH
+ Conductividad + Caudal + Vel_Corriente + Luz + Temp, data = amb)
```

```
              Inertia Proportion Rank
Total          0.14283      1.00000
Constrained    0.07804      0.54640   10
Unconstrained  0.06479      0.45360   13
Inertia is variance
```

Eigenvalues for constrained axes:

```
   RDA1    RDA2    RDA3    RDA4    RDA5    RDA6    RDA7    RDA8
0.028864 0.019400 0.009741 0.006291 0.004905 0.003407 0.002132 0.001938
   RDA9    RDA10
0.000927 0.000440
```

Eigenvalues for unconstrained axes:

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
0.026487	0.015199	0.009607	0.004206	0.002286	0.002040	0.001779	0.001226
PC9	PC10	PC11	PC12	PC13			
0.000826	0.000485	0.000303	0.000258	0.000088			

*Matriz 1. Partición de la varianza.* La inercia restringida es la que define el ajuste (restringida) en la relación entre las dos matrices de variables. Para este caso es de 0.54 (54%). Más adelante se aplicará el R2 de Ezequiel (1930), para encontrar el ajuste sin restricción (ajuste final del RDA). A continuación se muestra el comando para presentar los resultados detallados del RDA.

*Matriz 2. Importancia de los componentes.* Muestra que se requiere de 10 ejes canónicos (RDA) para explicar el 54% de la varianza explicada por la inercia restringida. La inercia restante se explica por los ejes de los 12 componentes principales PC.

*Matriz 3. Species scores,* muestra las coordenadas de las especies en los ejes canónicos, de los cuales se graficarán los dos primeros.

*Matriz 4. Site scores,* Muestra las coordenadas de los sitios

*Matriz 5. Site constraints,* muestra a las coordenadas de los sitios en el espacio de los taxones.

*Matriz 6. Biplot scores,* muestra las coordenadas de las variables ambientales.

```
summary(tax.rda) # Resultados completos
```

Call:

```
rda(formula = tax.hel ~ Amonio + Nitrito + Nitrato + Oxigeno + pH + Conductividad + Cau
```

Partitioning of variance:

	Inertia	Proportion
Total	0.14283	1.0000
Constrained	0.07804	0.5464
Unconstrained	0.06479	0.4536

Eigenvalues, and their contribution to the variance

Importance of components:

	RDA1	RDA2	RDA3	RDA4	RDA5	RDA6
Eigenvalue	0.02886	0.0194	0.009741	0.006291	0.004905	0.003407

Proportion Explained	0.20208	0.1358	0.068195	0.044042	0.034342	0.023850
Cumulative Proportion	0.20208	0.3379	0.406096	0.450138	0.484481	0.508331
	RDA7	RDA8	RDA9	RDA10	PC1	PC2
Eigenvalue	0.002132	0.001938	0.0009267	0.0004403	0.02649	0.0152
Proportion Explained	0.014928	0.013566	0.0064878	0.0030829	0.18544	0.1064
Cumulative Proportion	0.523259	0.536825	0.5433124	0.5463953	0.73183	0.8382
	PC3	PC4	PC5	PC6	PC7	PC8
Eigenvalue	0.009607	0.004206	0.002286	0.00204	0.001779	0.001226
Proportion Explained	0.067262	0.029449	0.016001	0.01428	0.012454	0.008585
Cumulative Proportion	0.905501	0.934951	0.950952	0.96524	0.977690	0.986274
	PC9	PC10	PC11	PC12	PC13	
Eigenvalue	0.0008263	0.0004854	0.0003026	0.0002579	8.835e-05	
Proportion Explained	0.0057854	0.0033982	0.0021184	0.0018053	6.186e-04	
Cumulative Proportion	0.9920595	0.9954577	0.9975762	0.9993814	1.000e+00	

#### Accumulated constrained eigenvalues

##### Importance of components:

	RDA1	RDA2	RDA3	RDA4	RDA5	RDA6
Eigenvalue	0.02886	0.0194	0.009741	0.006291	0.004905	0.003407
Proportion Explained	0.36984	0.2486	0.124809	0.080605	0.062853	0.043650
Cumulative Proportion	0.36984	0.6184	0.743228	0.823833	0.886685	0.930335
	RDA7	RDA8	RDA9	RDA10		
Eigenvalue	0.002132	0.001938	0.0009267	0.0004403		
Proportion Explained	0.027321	0.024828	0.0118739	0.0056423		
Cumulative Proportion	0.957656	0.982484	0.9943577	1.0000000		

#### Scaling 2 for species and site scores

- \* Species are scaled proportional to eigenvalues
- \* Sites are unscaled: weighted dispersion equal on all dimensions
- \* General scaling constant of scores: 1.346293

#### Species scores

	RDA1	RDA2	RDA3	RDA4	RDA5	RDA6
Fragillaria	0.13772	-0.142031	0.176943	0.0562043	-0.063697	0.0736817
Lyngbya	-0.45345	0.075392	0.055651	0.0747107	-0.007676	-0.0088353
Chamaepinnularia	0.11821	0.067633	0.002487	-0.0394544	-0.046828	-0.0453373
Achnantes	0.12193	-0.007016	-0.005199	-0.0579162	0.007942	0.0004642
Amphora	0.07641	-0.055694	-0.073152	0.0983283	0.009532	-0.0447069
Caloneis	0.10235	0.100918	0.010924	-0.0128037	-0.110912	-0.0484938
Closterium	-0.13921	-0.049866	-0.040879	-0.1396354	0.023961	-0.0356380
Cocconeis	0.08230	0.111537	-0.116166	-0.0253457	0.078588	0.0767795

Cymbella	0.05167	-0.055582	0.016607	-0.0439576	-0.006371	0.0316069
Eolimna	-0.04478	0.040748	0.035461	0.0316799	-0.075134	-0.0244236
Epithemia	0.07696	0.053827	0.045227	0.0438611	0.041962	-0.0432302
Eunotia	-0.05858	0.073250	-0.007121	-0.0114420	-0.009144	-0.0119924
Frustulia	0.02272	-0.199618	-0.124568	0.0131044	0.025406	-0.0360106
Girosigma	0.02634	0.128337	-0.026327	-0.0387420	-0.032438	0.0447555
Gomphonema	0.07341	0.161492	0.025849	0.0627397	0.022804	0.0060812
Melosira	-0.05697	0.080419	0.113156	-0.1522371	-0.010332	-0.0211294
Navicula	0.06661	0.024720	0.170748	0.0007566	0.147542	-0.0024464
Nitzschia	-0.01762	-0.146199	-0.006291	-0.0369694	-0.062184	0.0808324
Planothidium	0.19033	0.088406	0.005299	0.0107074	-0.019909	-0.0672288
Surilella	-0.01394	0.225111	-0.076448	0.0277918	-0.025008	0.0725367
Pinnularia	-0.07301	-0.071644	-0.012235	0.0131215	-0.014800	-0.0209236

Site scores (weighted sums of species scores)

	RDA1	RDA2	RDA3	RDA4	RDA5	RDA6
T1.1	-1.07703	0.01888	-0.336999	-0.09191	-0.07613	-0.45253
T1.2	-0.84054	-0.20049	-0.141839	-0.17384	0.41711	-0.36452
T1.3	-0.21890	-0.28606	0.057019	-0.41733	-0.19777	0.18328
T1.4	-0.10950	0.13679	0.237775	-0.15358	-0.27403	0.24129
T1.5	0.03167	0.36875	-0.068462	-0.02417	0.03290	0.45435
T1.6	-0.02467	0.44011	-0.025990	-0.07247	0.04603	0.29248
T1.7	0.16166	0.39197	-0.238118	-0.11707	0.09933	-0.03318
T1.8	0.10948	0.45084	-0.180997	-0.02135	0.13711	0.57389
T2.1	-0.26911	-0.30806	-0.767805	0.87397	-0.04427	0.98553
T2.2	-0.09952	-0.13226	0.772156	0.70675	0.69534	0.25746
T2.3	-0.22223	-0.20709	0.295607	0.34609	-0.52442	0.03893
T2.4	-0.10695	0.08534	0.088760	0.27482	-0.62125	-0.25749
T2.5	-0.01470	0.16145	0.362241	0.26382	-0.47381	-0.22220
T2.6	0.16730	0.34242	0.051480	0.06839	-0.07649	-0.30779
T2.7	0.32172	0.27478	-0.097778	0.15784	0.11343	-0.39605
T2.8	0.34014	0.40401	-0.109732	0.25509	0.06223	-0.44183
T3.1	0.20708	-0.87754	-0.685639	0.93999	-0.34090	0.42622
T3.2	0.15556	-0.56176	0.588049	-0.31921	-0.28143	0.64658
T3.3	0.19044	-0.31732	0.423720	-0.29719	-0.08333	0.35854
T3.4	0.26461	-0.20676	-0.026922	-0.24747	-0.14614	-0.28402
T3.5	0.26567	-0.07627	-0.007015	-0.51250	-0.24973	-0.40467
T3.6	0.26294	-0.02789	-0.060279	-0.54987	0.29896	-0.27382
T3.7	0.27363	0.07944	-0.083931	-0.34491	0.49386	-0.54819
T3.8	0.23124	0.04670	-0.045299	-0.54391	0.99339	-0.47227



Site constraints (linear combinations of constraining variables)

	RDA1	RDA2	RDA3	RDA4	RDA5	RDA6
T1.1	-0.608094	0.069700	-0.2659805	0.04600	-0.09435	-0.06431
T1.2	-0.672062	-0.282194	-0.1487709	-0.10561	0.35054	-0.16350
T1.3	-0.356768	-0.237330	0.2273373	-0.54749	-0.12603	0.04596
T1.4	0.009029	0.394109	0.1840858	-0.36486	-0.01351	-0.26989
T1.5	-0.018557	0.506317	-0.0001257	-0.12165	-0.12484	0.38849
T1.6	-0.084608	0.400060	0.0271550	-0.29946	-0.05580	0.08539
T1.7	-0.058104	0.306643	-0.2006076	0.20594	0.15177	0.29384
T1.8	-0.160580	0.192429	-0.4894293	0.01713	0.14478	0.50417
T2.1	0.059599	-0.099508	-0.1636965	0.23651	-0.09396	0.42975
T2.2	-0.152474	-0.181670	0.7823215	0.65227	0.57180	0.22938
T2.3	-0.135133	0.072656	0.1381488	0.31328	-0.15771	-0.26812
T2.4	0.080845	0.150351	0.1744033	0.11732	-0.18811	-0.19088
T2.5	-0.031560	-0.016833	0.3379676	0.09039	-0.71895	-0.07707
T2.6	0.223424	0.286614	0.0244991	0.13025	0.03622	-0.30461
T2.7	0.162822	0.261848	-0.0774071	0.33031	-0.01031	-0.18129
T2.8	0.214704	0.144374	-0.1007849	0.12653	0.16237	-0.32115
T3.1	0.276688	-0.515338	-0.5875697	0.21567	0.03745	-0.21428
T3.2	-0.078309	-0.261042	0.0094046	0.01194	-0.09537	0.23117
T3.3	0.445486	-0.242713	0.3772510	-0.37955	0.05636	0.29527
T3.4	-0.128534	-0.394367	0.1125166	-0.31742	0.01463	-0.11643
T3.5	0.206039	-0.409549	-0.1202982	0.19481	-0.60608	0.19190
T3.6	0.512665	-0.044235	-0.0726911	-0.31755	0.36432	0.26933
T3.7	0.092435	-0.008915	-0.0763885	-0.02302	0.01115	-0.43913
T3.8	0.201048	-0.091407	-0.0913406	-0.21175	0.38361	-0.35399

Biplot scores for constraining variables

	RDA1	RDA2	RDA3	RDA4	RDA5	RDA6
Amonio	-0.05176	-0.12017	0.67080	0.2933	0.41466	0.27637
Nitrito	-0.58706	-0.22830	0.29302	-0.3920	-0.10474	-0.33554
Nitrato	0.11287	-0.04070	0.34885	-0.4042	-0.26078	-0.02453
Oxigeno	0.10513	-0.04085	-0.64672	0.1262	0.55134	-0.18159
pH	0.22497	0.26189	-0.03689	-0.5123	-0.15576	-0.46422
Conductividad	-0.61873	0.43117	-0.22737	-0.3900	0.06681	0.25390
Caudal	-0.32017	0.13247	-0.28194	-0.5026	0.08692	0.36194
Vel_Corriente	0.03368	0.27010	-0.30589	-0.4551	-0.06465	0.51914
Luz	0.30256	-0.52037	-0.35636	0.2878	-0.31339	-0.01181
Temp	-0.32332	0.77248	-0.07647	0.2202	0.25887	-0.12217

A continuación se muestra una manera de extraer algunos insumos por separado del anterior comando `summary(tax.rda)`. Las coordenadas de los taxones y de los sitios serán tenidas en cuenta más adelante, para las figuras de `ggplot2`.

```
# Matriz 3. Escores o coordenadas de los taxones
species.scores <- scores(tax.rda, display = "species")

# Escores o coordenadas de los sitios
site.scores <- scores(tax.rda, display = "sites")

# Escores de las variables restringidas
biplot.scores <- scores(tax.rda, display = "bp")
```

#### 29.0.1.2 Paso 2. Coeficientes de las variables regresoras (ambientales), en el modelo lineal.

Solo se mostrarán los tres primeros ejes canónicos [,1:3], para facilidad de su interpretación.

```
round(coef(tax.rda),2)[,1:3]
```

	RDA1	RDA2	RDA3
Amonio	0.01	-0.01	0.10
Nitrito	-0.18	-0.32	0.18
Nitrato	0.04	0.03	0.01
Oxigeno	0.03	-0.04	-0.11
pH	0.07	0.07	0.04
Conductividad	-0.02	0.01	-0.01
Caudal	0.10	-0.28	0.00
Vel_Corriente	0.37	0.21	0.02
Luz	0.00	0.00	0.00
Temp	0.06	0.14	0.01

Se puede pensar en un modelo lineal, que tiene en cuenta a los coeficientes descritos en el primer eje canónico:

Distribución de los taxones de microalgas (**Matriz Y**) = 0.01(**Amonio**) - 0.18(**Nitrito**) + ... + 0.06(**Temp**)

### 29.0.1.3 Paso 3. R2 sin ajuste vs. R2 ajustado (Ezequiel 1930)

La nueva inercia no sesgada (sin restricción) calculada con la formula de Ezequiel es de 0.19 o del 19%.

```
# R^2 sin ajuste (inercia restringida)
(R2 <- RsquareAdj(tax.rda)$r.squared)
```

```
[1] 0.5463953
```

```
# R^2 ajustado
(R2adj <- RsquareAdj(tax.rda)$adj.r.squared)
```

```
[1] 0.1974686
```

### 29.0.1.4 Paso 4. Figura de Triplot

A continuación, se realizará la gráfica del RDA (figura Triplot) (Figura 32.2), que relaciona a los tres elementos: taxones, variables ambientales y sitios de muestreo mediante dos tipos de escalamiento (Scalings 1 y 2).

```
dev.new(title = "RDA scaling 1 y 2",
         width = 16,height = 8,noRStudioGD = TRUE)
par(mfrow = c(1, 2))
# Scaling 1
plot(tax.rda,scaling=1, display = c("sp", "lc", "cn"), main="RDA - scaling 1")
# Scaling 2
plot(tax.rda, display = c("sp", "lc", "cn"), main="RDA - scaling 2")

par(mfrow = c(1, 1))
```

### 29.0.1.5 Paso 5. Prueba global del RDA

Esta prueba obtiene un valor  $p = 0.04296$  \*, por lo cual se valida que el modelo de regresión múltiple de este RDA presenta un ajuste apropiado (a pesar de la poca inercia encontrada).

```
# Prueba global del RDA (dos opciones)
# Ho= no hay relación entre las variables X y las Y
anova(tax.rda, permutations = how(nperm = 1000))
```

Permutation test for rda under reduced model

Permutation: free

Number of permutations: 1000

Model: rda(formula = tax.hel ~ Amonio + Nitrito + Nitrato + Oxigeno + pH + Conductividad + C

	Df	Variance	F	Pr(>F)
--	----	----------	---	--------

Model	10	0.078044	1.5659	0.04895 *
-------	----	----------	--------	-----------

Residual	13	0.064790		
----------	----	----------	--	--

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

A continuación se muestra que ninguno de los ejes canónicos presenta significancia para la ordenación de las variables y de las observaciones de este análisis (valor  $p > 0.05$ ), sin embargo se continuará con el procedimiento.

```
# Prueba de los ejes canónicos
```

```
anova(tax.rda, by = "axis", permutations = how(nperm = 1000))
```

Permutation test for rda under reduced model

Forward tests for axes

Permutation: free

Number of permutations: 1000

Model: rda(formula = tax.hel ~ Amonio + Nitrito + Nitrato + Oxigeno + pH + Conductividad + C

	Df	Variance	F	Pr(>F)
--	----	----------	---	--------

RDA1	1	0.028864	5.7914	0.1349
------	---	----------	--------	--------

RDA2	1	0.019400	3.8926	0.3317
------	---	----------	--------	--------

RDA3	1	0.009741	1.9544	0.9331
------	---	----------	--------	--------

RDA4	1	0.006291	1.2622	0.9990
------	---	----------	--------	--------

RDA5	1	0.004905	0.9842	0.9990
------	---	----------	--------	--------

RDA6	1	0.003407	0.6835	0.9990
------	---	----------	--------	--------

RDA7	1	0.002132	0.4278	1.0000
------	---	----------	--------	--------

RDA8	1	0.001938	0.3888	1.0000
------	---	----------	--------	--------

RDA9	1	0.000927	0.1859	1.0000
------	---	----------	--------	--------

RDA10	1	0.000440	0.0884	0.9970
-------	---	----------	--------	--------

Residual	13	0.064790		
----------	----	----------	--	--

### 29.0.1.6 Paso 6. Factor de inflación de la varianza (VIF) del RDA

```
# Factor de inflación  
round(vif.cca(tax.rda), 2)
```

Amonio	Nitrito	Nitrato	Oxigeno	pH
1.40	2.02	1.61	1.22	1.47
Conductividad	Caudal	Vel_Corriente	Luz	Temp
9.28	7.46	4.85	2.04	2.31

Los resultados están por debajo de un VIF de 10, por lo que todas las variables son importantes para el análisis.

### 29.0.1.7 Paso 7. Criterios de selección de variables ambientales (X)

#### 29.0.1.8 7.1 Forward selection usando forward.sel()

El comando `forward.sel` permitirá definir a las variables ambientales con importancia para ser relacionadas con los taxones en el RDA. Para este caso define a la *Conductividad* y a la *Velocidad del la Corriente*.

```
# Factor de inflación  
forward.sel(tax.hel, amb, adjR2thresh = R2adj)
```

Testing variable 1

Testing variable 2

Testing variable 3

Procedure stopped (adjR2thresh criteria) adjR2cum = 0.200960 with 3 variables (> 0.197469)

	variables	order	R2	R2Cum	AdjR2Cum	F	pvalue
1	Conductividad	6	0.11614724	0.1161472	0.07597211	2.891024	0.008
2	Vel_Corriente	8	0.09924816	0.2153954	0.14067115	2.656384	0.015

### 29.0.1.9 7.2 Eliminación anticipada (Backward) usando "ordistep()" de vegan

El anterior resultado es validado por esta función `ordistep`, la cual luego de varias corridas, define a las mismas variables ambientales *Conductividad* y a la *Velocidad del la Corriente*, pero incluye a la *Temperatura* como las significativas para el análisis RDA. Para continuar el ejercicio, a continuación se realizará un nuevo RDA (RDA parsimonioso) con estas tres variables.

```
# 7.2 Eliminación anticipada (Backward) usando "ordistep()" de vegan
step.backward <- ordistep(tax.rda, permutations = how(nperm = 499))
```

```
Start: tax.hel ~ Amonio + Nitrito + Nitrato + Oxigeno + pH + Conductividad +
```

```
Caudal + V
```

	Df	AIC	F	Pr(>F)
- Nitrato	1	-45.747	0.5267	0.792
- Nitrito	1	-45.375	0.7380	0.614
- Luz	1	-45.164	0.8591	0.510
- Caudal	1	-45.084	0.9053	0.468
- Amonio	1	-44.961	0.9770	0.406
- pH	1	-44.738	1.1072	0.368
- Oxigeno	1	-44.667	1.1492	0.302
- Vel_Corriente	1	-44.412	1.3001	0.238
- Temp	1	-44.020	1.5356	0.168
- Conductividad	1	-43.286	1.9875	0.096 .

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

```
Step: tax.hel ~ Amonio + Nitrito + Oxigeno + pH + Conductividad + Caudal +
```

```
Vel_Corriente
```

	Df	AIC	F	Pr(>F)
- Nitrito	1	-46.555	0.7127	0.636
- Caudal	1	-46.216	0.9222	0.450
- Amonio	1	-46.112	0.9867	0.364
- Luz	1	-45.947	1.0903	0.330
- pH	1	-45.915	1.1104	0.330
- Oxigeno	1	-45.753	1.2124	0.286
- Temp	1	-45.284	1.5132	0.178
- Vel_Corriente	1	-44.698	1.8963	0.120
- Conductividad	1	-43.786	2.5121	0.054 .

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Step: tax.hel ~ Amonio + Oxigeno + pH + Conductividad + Caudal + Vel\_Corriente + Luz + Temp

	Df	AIC	F	Pr(>F)
- Caudal	1	-47.100	0.9375	0.392
- Amonio	1	-46.967	1.0261	0.374
- Luz	1	-46.842	1.1100	0.354
- pH	1	-46.802	1.1365	0.330
- Oxigeno	1	-46.620	1.2591	0.260
- Vel_Corriente	1	-44.635	2.6618	0.046 *
- Temp	1	-45.383	2.1193	0.040 *
- Conductividad	1	-43.968	3.1590	0.018 *

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Step: tax.hel ~ Amonio + Oxigeno + pH + Conductividad + Vel\_Corriente + Luz + Temp

	Df	AIC	F	Pr(>F)
- Amonio	1	-47.616	1.0204	0.374
- Oxigeno	1	-47.387	1.1836	0.336
- Luz	1	-47.525	1.0854	0.310
- pH	1	-47.448	1.1398	0.276
- Temp	1	-45.715	2.4235	0.022 *
- Vel_Corriente	1	-44.448	3.4221	0.016 *
- Conductividad	1	-42.714	4.8773	0.002 **

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Step: tax.hel ~ Oxigeno + pH + Conductividad + Vel\_Corriente + Luz + Temp

	Df	AIC	F	Pr(>F)
- Luz	1	-48.000	1.1841	0.368
- pH	1	-47.803	1.3343	0.238
- Oxigeno	1	-47.714	1.4020	0.202
- Temp	1	-46.427	2.4162	0.038 *
- Vel_Corriente	1	-45.170	3.4595	0.004 **
- Conductividad	1	-43.486	4.9469	0.002 **

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Step: tax.hel ~ Oxigeno + pH + Conductividad + Vel\_Corriente + Temp

	Df	AIC	F	Pr(>F)
--	----	-----	---	--------

```

- pH          1 -48.202 1.4002 0.198
- Oxigeno      1 -48.138 1.4525 0.168
- Vel_Corriente 1 -45.949 3.3094 0.012 *
- Temp         1 -46.244 3.0499 0.006 **
- Conductividad 1 -44.266 4.8576 0.004 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
Step: tax.hel ~ Oxigeno + Conductividad + Vel_Corriente + Temp
```

```

          Df      AIC      F Pr(>F)
- Oxigeno    1 -48.466 1.4257 0.194
- Temp       1 -47.010 2.7027 0.016 *
- Vel_Corriente 1 -46.238 3.4121 0.004 **
- Conductividad 1 -44.954 4.6440 0.002 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
Step: tax.hel ~ Conductividad + Vel_Corriente + Temp
```

```

          Df      AIC      F Pr(>F)
- Temp       1 -47.549 2.5845 0.016 *
- Vel_Corriente 1 -46.730 3.3681 0.006 **
- Conductividad 1 -45.444 4.6545 0.004 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

### 29.0.1.10 Paso 8. R2 ajustado

Al validar el ajuste del RDA con las dos variables seleccionadas, se obtiene un valor de 0.3 o 30% de ajuste.

```

# Se define un R^2: 0.3 (30% de relación)
RsquareAdj(step.backward)

```

```

$r.squared
[1] 0.3051824

```

```

$adj.r.squared
[1] 0.2009598

```



### 29.0.1.11 Paso 9. RDA Parsimonioso (rda.par)

RDA Parsimonioso significa que se realizará un nuevo RDA con las dos variables ambientales seleccionadas.

```
# RDA resumido
(rda.pars <- rda(tax.hel ~ Temp + Vel_Corriente + Conductividad, data = amb))
```

```
Call: rda(formula = tax.hel ~ Temp + Vel_Corriente + Conductividad,
data = amb)
```

	Inertia	Proportion	Rank
Total	0.14283	1.00000	
Constrained	0.04359	0.30518	3
Unconstrained	0.09924	0.69482	20

Inertia is variance

Eigenvalues for constrained axes:

RDA1	RDA2	RDA3
0.025917	0.013517	0.004157

Eigenvalues for unconstrained axes:

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
0.03482	0.01676	0.01396	0.00973	0.00677	0.00518	0.00252	0.00220

(Showing 8 of 20 unconstrained eigenvalues)

### 29.0.1.12 Paso 10. Coeficientes del modelo lineal parsimonioso

```
# RDA resumido
round(coef(rda.pars),2)
```

	RDA1	RDA2	RDA3
Temp	0.04	0.23	0.08
Vel_Corriente	0.47	0.36	-0.36
Conductividad	-0.02	-0.01	0.00

### 29.0.1.13 Paso 11. Dos Triplots del RDA parsimonioso (Scaling 1 y Scaling 2)

```
dev.new(title = "RDA Parsimonioso scaling 1 y 2",
        width = 16,height = 8,noRStudioGD = TRUE)
par(mfrow = c(1, 2))

# Scaling 1
plot(rda.pars,scaling = 1,display = c("sp", "lc", "cn"),
     main = "Triplot RDA taxa.hel ~ amb - scaling 1")
spe.sc1 <- scores(rda.pars, choices = 1:2, scaling = 1, display = "sp")
arrows(0, 0, spe.sc1[, 1] * 0.92,spe.sc1[, 2] * 0.92,
       length = 0, lty = 1, col = "red")

# Scaling 2
plot(rda.pars,scaling = 2,display = c("sp", "lc", "cn"),
     main = "Triplot RDA taxa.hel ~ amb - scaling 2")
spe.sc1 <- scores(rda.pars, choices = 1:2, scaling = 2, display = "sp")
arrows(0, 0, spe.sc1[, 1] * 0.92,spe.sc1[, 2] * 0.92,
       length = 0, lty = 1, col = "red")
par(mfrow = c(1, 1))
```

#—

### 29.0.1.14 Paso 12. RDA con paquete ggplot2

Se realizará la figura del RDA con el paquete `ggplot2`, dada su mejor presentación, comparado a las figuras anteriores, realizadas con el paquete `vegan`. Los siguientes comandos sirven para identificar las coordenadas de los sitios (“sites”), los taxones (“sp”) y las variables ambientales (“vectors”).

```
# Insumos del RDA parsimonioso o que resume a las tres variables
(rda.pars <- rda(tax.hel ~ Temp + Vel_Corriente + Conductividad, data = amb)) # RDA resu
```

```
Call: rda(formula = tax.hel ~ Temp + Vel_Corriente + Conductividad,
data = amb)
```

	Inertia	Proportion	Rank
Total	0.14283	1.00000	
Constrained	0.04359	0.30518	3
Unconstrained	0.09924	0.69482	20

Inertia is variance

Eigenvalues for constrained axes:

RDA1	RDA2	RDA3
0.025917	0.013517	0.004157

Eigenvalues for unconstrained axes:

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
0.03482	0.01676	0.01396	0.00973	0.00677	0.00518	0.00252	0.00220

(Showing 8 of 20 unconstrained eigenvalues)

```
names(summary(rda.pars))      # Insumos del RDA parsimonioso
```

```
[1] "species"      "sites"         "constraints"   "biplot"        "call"
[6] "tot.chi"      "constr.chi"    "unconst.chi"  "cont"          "concont"
[11] "scaling"      "digits"        "inertia"      "method"
```

#### 29.0.1.14.1 12.1 Coordenadas de los sitios y el factor “coord.sit”

```
# 1) Coordenadas de los sitios y el factor (coord.sit)
coord.sit <- as.data.frame(scores(rda.pars,
                                   choices = 1:2, display = "sites"))) # Coordenadas de
coord.sit$sitio <- rownames(coord.sit) # Crear una columna con nombres de los sitios
coord.sit$grp <- datos$Tributario # Adicionar columna de grupos por Tributario
head(coord.sit) # vista resumida de las coordenadas de sitios
```

	RDA1	RDA2	sitio	grp
T1.1	-1.11971663	-0.30601405	T1.1	T1
T1.2	-0.85825274	-0.42028232	T1.2	T1
T1.3	-0.18370421	-0.36439466	T1.3	T1
T1.4	-0.12410643	0.08403049	T1.4	T1
T1.5	-0.02273675	0.46138698	T1.5	T1
T1.6	-0.10175889	0.47796564	T1.6	T1

#### 29.0.1.14.2 12.2 Coordenadas de los taxones “coord.tax”

```
# 2) Coordenadas de las especies (coord.tax)
coord.tax <- as.data.frame(scores(rda.pars,
                                   choices = 1:2, display = "sp"))) # Dos primeros ejes
coord.tax$especies <- rownames(coord.tax) # Insertar columna con nombres de las especies
```

```
head(coord.tax)
```

	RDA1	RDA2	especies
Fragillaria	0.18835404	-0.111108636	Fragillaria
Lyngbya	-0.41883579	0.006822599	Lyngbya
Chamaepinnularia	0.09146325	0.064258294	Chamaepinnularia
Achnantes	0.09957281	-0.021272988	Achnantes
Amphora	0.08594521	-0.039656846	Amphora
Caloneis	0.07482072	0.043049919	Caloneis

#### 29.0.1.14.3 12.3 Coordenadas de las ambientales “coord.amb”

```
# 3) Coordenadas de las especies (coord.tax)
amb1 <- envfit(tax.rda, amb) # Se pueden seleccionar variables con, p.max = 0.05
coord.amb = as.data.frame(scores(amb1, "vectors"))
coord.amb$amb <- rownames(coord.amb) # Insertar columna con nombres de las ambientales
coord.amb = coord.amb[c(6,8,10),] # La 3 variables seleccionadas
head(coord.amb)
```

	RDA1	RDA2	amb
Conductividad	-0.51073374	0.4101799	Conductividad
Vel_Corriente	-0.00143185	0.2248100	Vel_Corriente
Temp	-0.32050590	0.6553301	Temp

#### 29.0.1.14.4 12.4 Figura del RDA con vectores de especies

```
x11()
ggplot() +
  # Sitios
  geom_text_repel(data = coord.sit, aes(RDA1, RDA2, label = row.names(coord.sit)),
    size = 4) + # Muestra el cuadro de la figura
  geom_point(data = coord.sit, aes(RDA1, RDA2, colour = grp), size = 4) +
  scale_shape_manual(values = c(21:25)) +
  # Taxones
  geom_segment(data = coord.tax, aes(x = 0, y = 0, xend = RDA1, yend = RDA2),
    arrow = arrow(angle = 22.5, length = unit(0.25, "cm"),
      type = "closed"), linetype = 1, size = 0.6, colour = "red") +
  geom_text_repel(data = coord.tax, aes(RDA1, RDA2, label = especies), colour = "red") +
  # Factor
  geom_polygon(data = coord.sit, aes(x = RDA1, y = RDA2, fill = grp, group = grp), alpha = 0.30) +
```

```
geom_hline(yintercept=0,linetype=3,size=1) +
geom_vline(xintercept=0,linetype=3,size=1)+
guides(shape=guide_legend(title=NULL,color="black"),
       fill=guide_legend(title=NULL))+
theme_bw()+theme(panel.grid=element_blank())
```

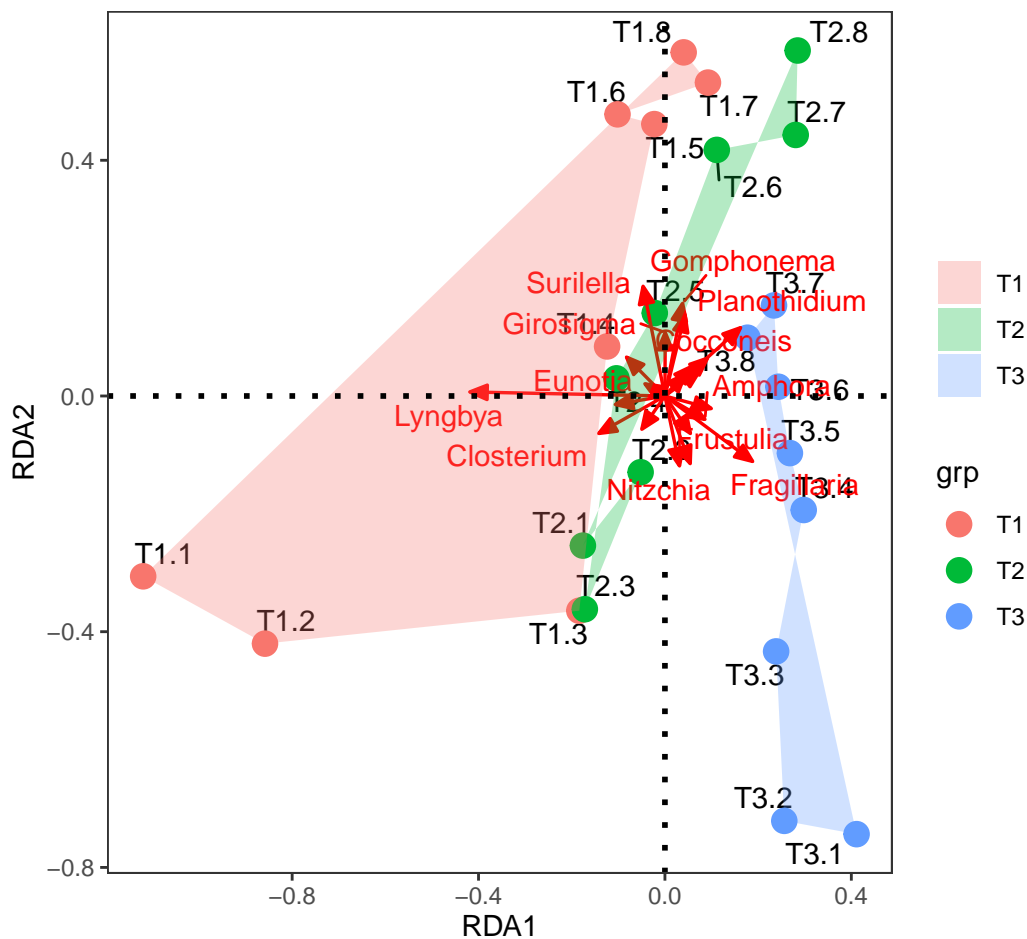


Figura 29.1: Figuras del RDA con las variables biológicas y los sitios

#### 29.0.1.14.5 12.5 Figura con vectores de especies (sin flechas)

```
x11()
ggplot() +
  # Sitios
```

```

geom_text_repel(data = coord.sit,aes(RDA1,RDA2,label=row.names(coord.sit)),
               size=4)+ # Muestra el cuadro de la figura
geom_point(data = coord.sit,aes(RDA1,RDA2,colour=grp),size=4)+
scale_shape_manual(values = c(21:25))+
# Taxones *valores de cero para caracteres de las flechas (arrow)
geom_segment(data = coord.tax,aes(x = 0, y = 0, xend = RDA1, yend = RDA2),
             arrow = arrow(angle=0,length = unit(0,"cm"),
                           type = "closed"),linetype=0, size=0,colour = "red")+
geom_text_repel(data = coord.tax,aes(RDA1,RDA2,label=especies),colour = "red")+
# Factor
geom_polygon(data=coord.sit,aes(x=RDA1,y=RDA2,fill=grp,group=grp),alpha=0.30) +

geom_hline(yintercept=0,linetype=3,size=1) +
geom_vline(xintercept=0,linetype=3,size=1)+
guides(shape=guide_legend(title=NULL,color="black"),
       fill=guide_legend(title=NULL))+
theme_bw()+theme(panel.grid=element_blank())

```

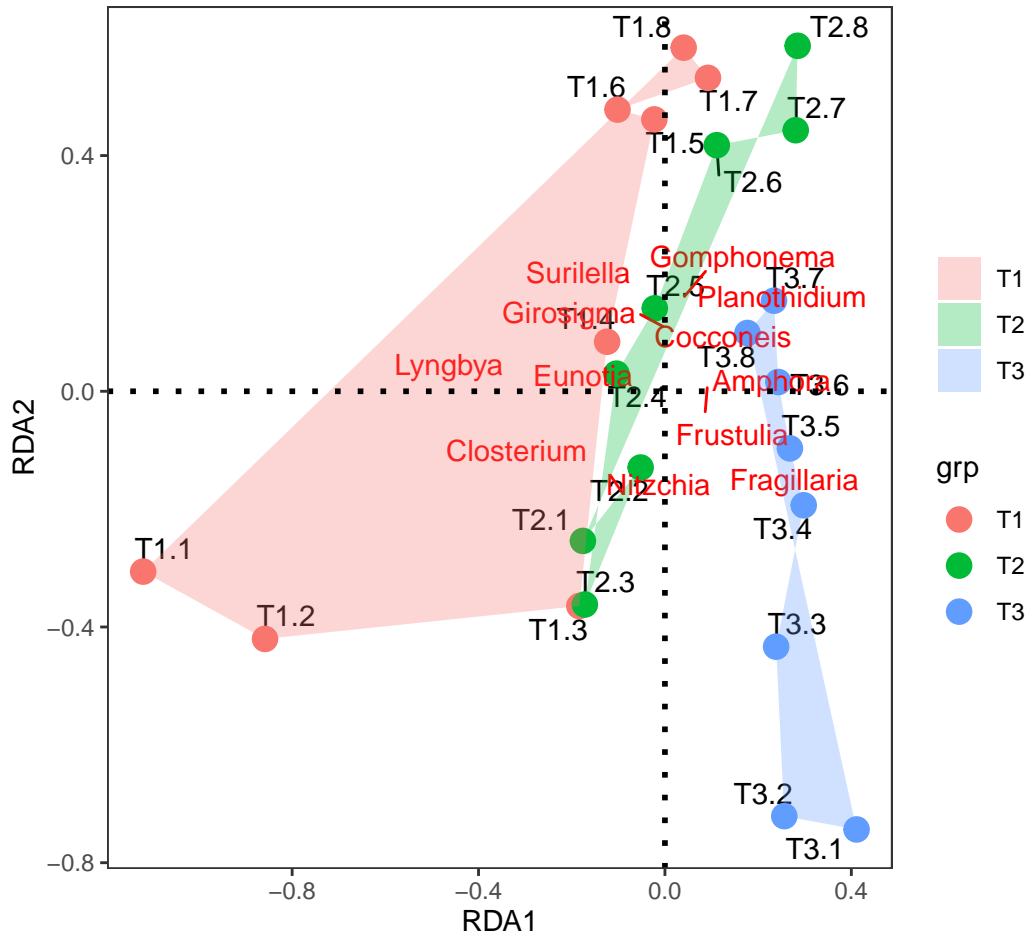


Figura 29.2: Figuras del RDA con las variables biológicas y los sitios

#### 29.0.1.14.6 12.6 Figura con vectores de especies y ambientales

```
x11()
ggplot() +
  # Sitios
  geom_text_repel(data = coord.sit, aes(RDA1, RDA2, label=row.names(coord.sit)),
                  size=4)+ # Muestra el cuadro de la figura
  geom_point(data = coord.sit, aes(RDA1, RDA2, colour=grp), size=4)+
  scale_shape_manual(values = c(21:25))+
  # especies
  geom_segment(data = coord.tax, aes(x = 0, y = 0, xend = RDA1, yend = RDA2),
              arrow = arrow(angle=0, length = unit(0, "cm"),
```

```

                                type = "closed"),linetype=0, size=0,colour = "red")+
geom_text_repel(data = coord.tax,aes(RDA1,RDA2,label=species),colour = "red")+
# Ambiental
geom_segment(data = coord.amb,aes(x = 0, y = 0, xend = RDA1, yend = RDA2),
            arrow = arrow(angle=22.5,length = unit(0.25,"cm"),
                                type = "closed"),linetype=1, size=0.6,colour = "blue")+
geom_text_repel(data = coord.amb,aes(RDA1,RDA2,label=row.names(coord.amb)),colour = "#00
# Factor
geom_mark_ellipse(data=coord.sit, aes(x=RDA1,y=RDA2,fill=grp,group=grp),alpha=0.30)  +

geom_hline(yintercept=0,linetype=3,size=1) +
geom_vline(xintercept=0,linetype=3,size=1)+
guides(shape=guide_legend(title=NULL,color="black"),
        fill=guide_legend(title=NULL))+
theme_bw()+theme(panel.grid=element_blank())

```



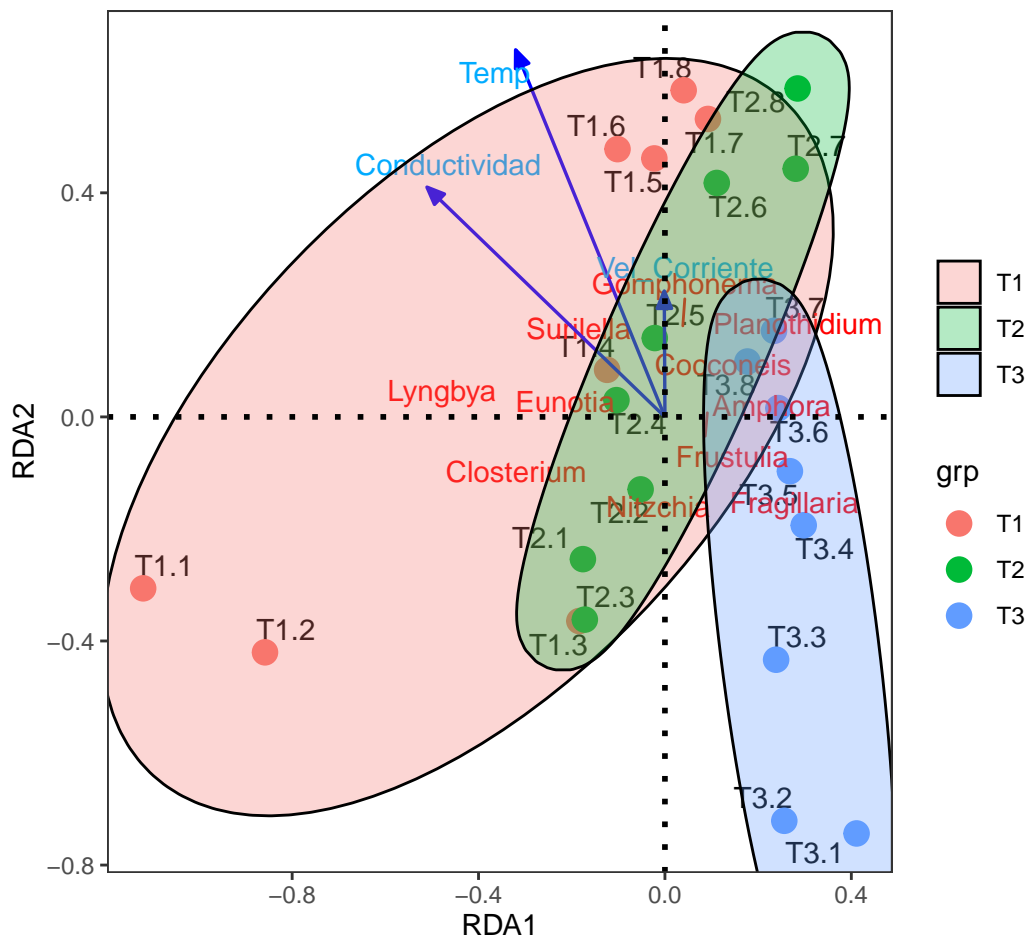


Figura 29.3: Figuras del RDA con las variables biológicas, las ambientales y los sitios

## 29.0.2 Taller de entrenamiento

**Objetivo:** Poner en práctica los conceptos vistos en este taller, realizando las siguientes opciones realizando un RDA con las variables biológicas (taxones) y variables ambientales. Enviar los resultados al *Teams* del profesor. ### P

## 30 Taller 8.1 Análisis de Clúster - CLA

### Objetivo de la actividad:

Este ejercicio se realizará con la base de datos **FQmarino.csv**, que ya fue utilizada en el *Taller 4.1 de Componentes Principales*. Esta base contiene datos de siete variables fisicoquímicas, tomadas en siete bahías de Santa Marta.

**El objetivo** de este ejercicio consiste en la realización de un análisis de clúster, basado en cuatro pasos generales (distancia, método de agrupación, número de clúster y selección de variables clasificadoras), para realizar una clasificación de las bahías, basado en las variables que las caracterizan.

### 30.0.1 Referencias bibliográficas de apoyo.

- **Clúster Jerárquicos**

[Libro: Análisis de datos ecológicos y ambientales - Rodríguez-Barrios Javier 2023](#) Se detallan todos los procedimientos descritos en el presente ejercicio.

[Microalgas de la CGSM - Vidal et al. \(2018\)](#). Implementación de un cluster no jerárquico para valorar paleoambientes con microalgas de la Ciénaga Grande de Santa Marta.

[Cluster](#) Brinda información complementaria para los diferentes pasos que requiere un análisis de clúster.

[Clustering y heatmaps](#) Similar al anterior enlace, brinda información detallada sobre el análisis de clúster.

[Análisis de conglomerados](#) Otro enlace con información general sobre los clúster.

[Clustering y heatmaps: aprendizaje no supervisado](#) Aplicación de clúster en diferentes disciplinas.

[Hierarchical Cluster Analysis](#) Enlace en el que se encuentra información sobre cluster jerárquicos y técnicas detalladas para seleccionar el número de k - clúster o grupos formados.

[Determining The Optimal Number Of Clusters](#) Información relevante para el paso 3 de este ejercicio, relacionado a la definición de los k-clúster o el número de grupos formados.

- **Clúster no Jerárquicos**

[K-means Cluster Analysis](#) Brinda información sobre la construcción de clúster no jerárquicos.

- **Otros**

[Introduction to dendextend](#) El paquete *dendextend* brinda opciones para comparar y visualizar dendogramas. Esto complementa al paso 3 del presente ejercicio, relacionado a la definición de los k-clúster formados.

[Hierarchical Clustering on Principal Components](#) Articulación de los clúster en los análisis de componentes principales.

### 30.0.1.1 Cargar las librerías requeridas

```
# Librerías requeridas
library(ellipse)
require(gclus)
require(SciViews)
require(ade4)
require(vegan)
library(corrplot)
library(ggplot2)
library(pheatmap)
library("gplots")
library(gridExtra)
library(factoextra)
```

### 30.0.1.2 Cargar o importar la base de datos

```
# Base de datos
datos = read.csv2("FQmarino.csv",row.names=1)

colnames(datos) = c("Sitio","pH","Cond","Turb","Temp","Sali","CFot","Oxig")
```

### 30.0.1.3 Exploración de los datos

Para este ejemplo se utilizarán figuras que relacionan a dos o más variables. En casos en los que se tengan diferentes grupos definidos, se pueden incluir figuras de cajas que permitan visualizar diferencias entre dichos grupos definidos por algún factor.

```
# Elipses con colores
M <- cor(datos[,2:8]) # Matriz de Correlación (M)
```

La Figura 32.2 permite visualizar las relaciones lineales entre todas las parejas de variables, incluyendo a los coeficientes de correlación de Pearson.

```
x11()
corrplot(M, method = "circle", # Correlaciones con círculos
          type = "lower", insig="blank", # Forma del panel
          order = "AOE", diag = FALSE, # Ordenar por nivel de correlación
          addCoef.col = "black", # Color de los coeficientes
          number.cex = 0.8, # Tamaño del texto
          col = COL2("RdYlBu", 200)) # Transparencia de los círculos
```

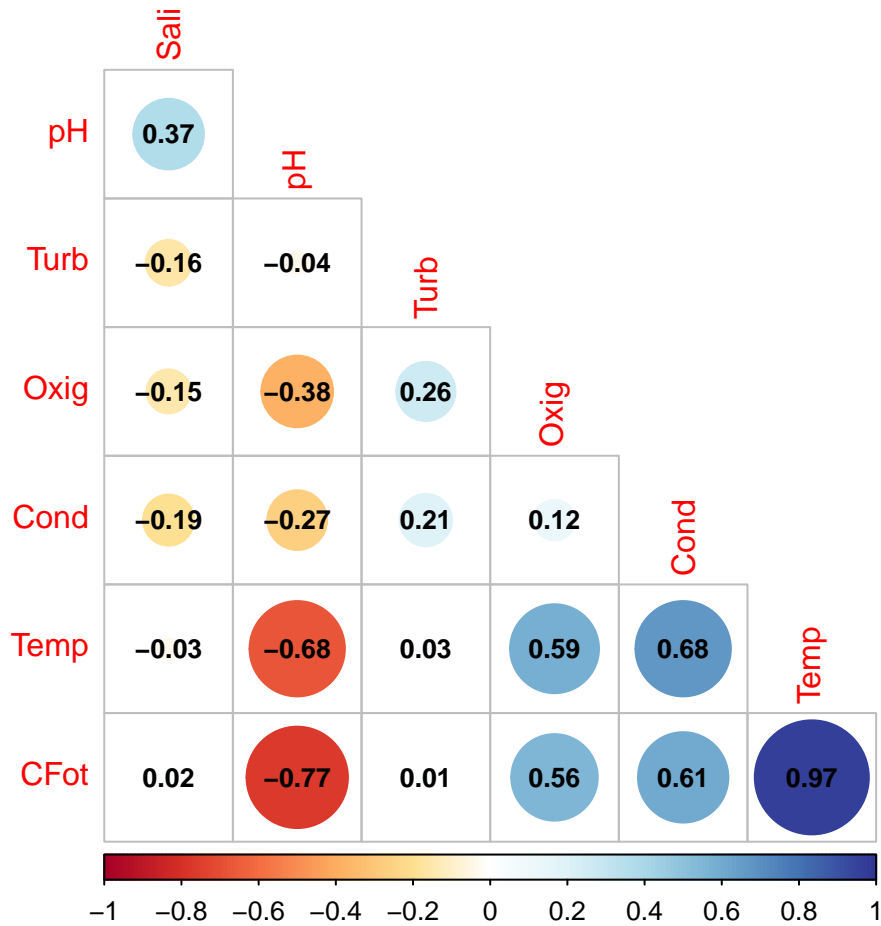


Figura 30.1: Correlaciones y coeficientes de correlación.

La Figura 32.3 es otra posibilidad para visualizar la relación entre las parejas de variables, pero además incluye paneles que visualizan la dispersión de los datos.

```
library(GGally)
x11()
ggpairs(data=datos[,c(2:8)],
        diag = list(continuous = "densityDiag"),
        upper = list(combo = "box"),
        lower = list(combo = "dot", aes(fill = Sitio))) +
scale_fill_brewer(palette = "Set1") +
theme(axis.text.x = element_text(angle = 90, vjust = 0.5))
```

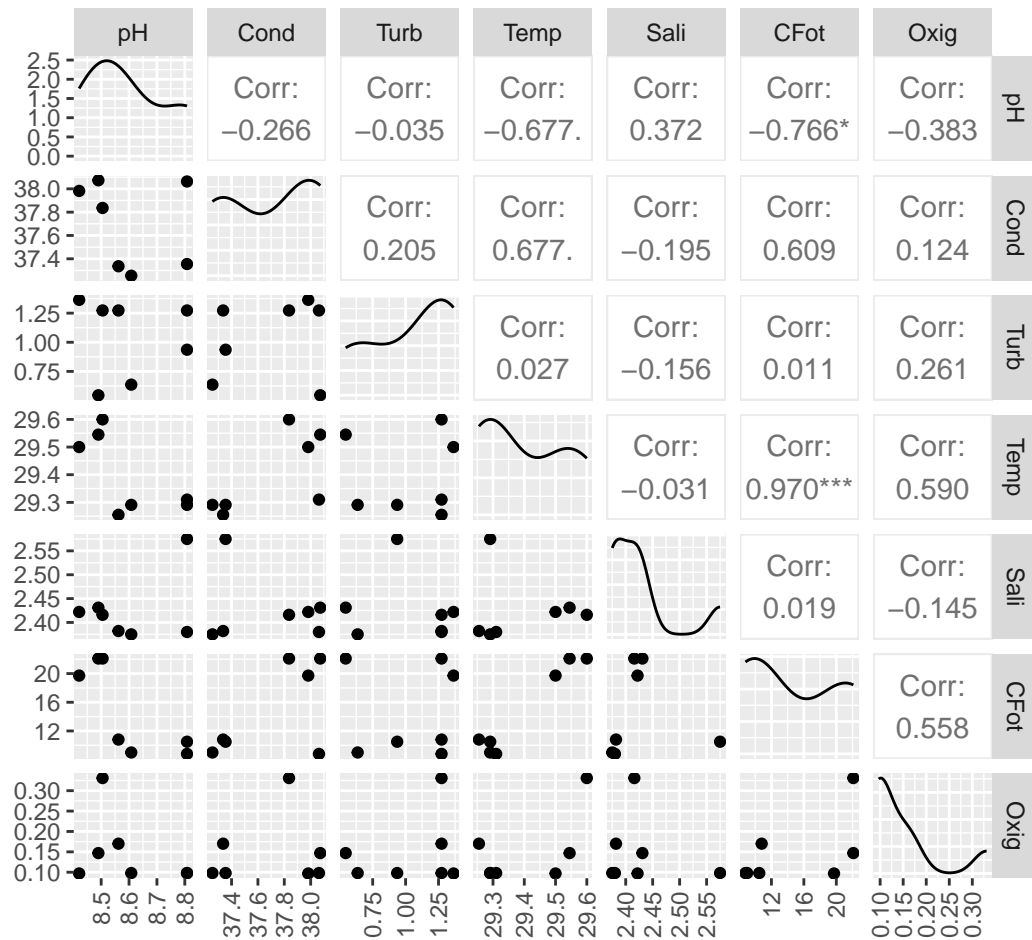


Figura 30.2: Correlaciones, dispersión y coeficientes de correlación.

La Figura 32.4 a diferencia de la anterior, clasifica a los grupos por colores y además incluye a sus coeficientes de correlación y el patrón de distribución de cada variable mediante histogramas de densidad.

```
library(GGally)
x11()
ggpairs(datos[,c(2:8)], aes(color=datos$Sitio),
  diag = list(continuous = "densityDiag"),
  upper = list(combo = "box"),
  lower = list(combo = "dot")) +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5))
```

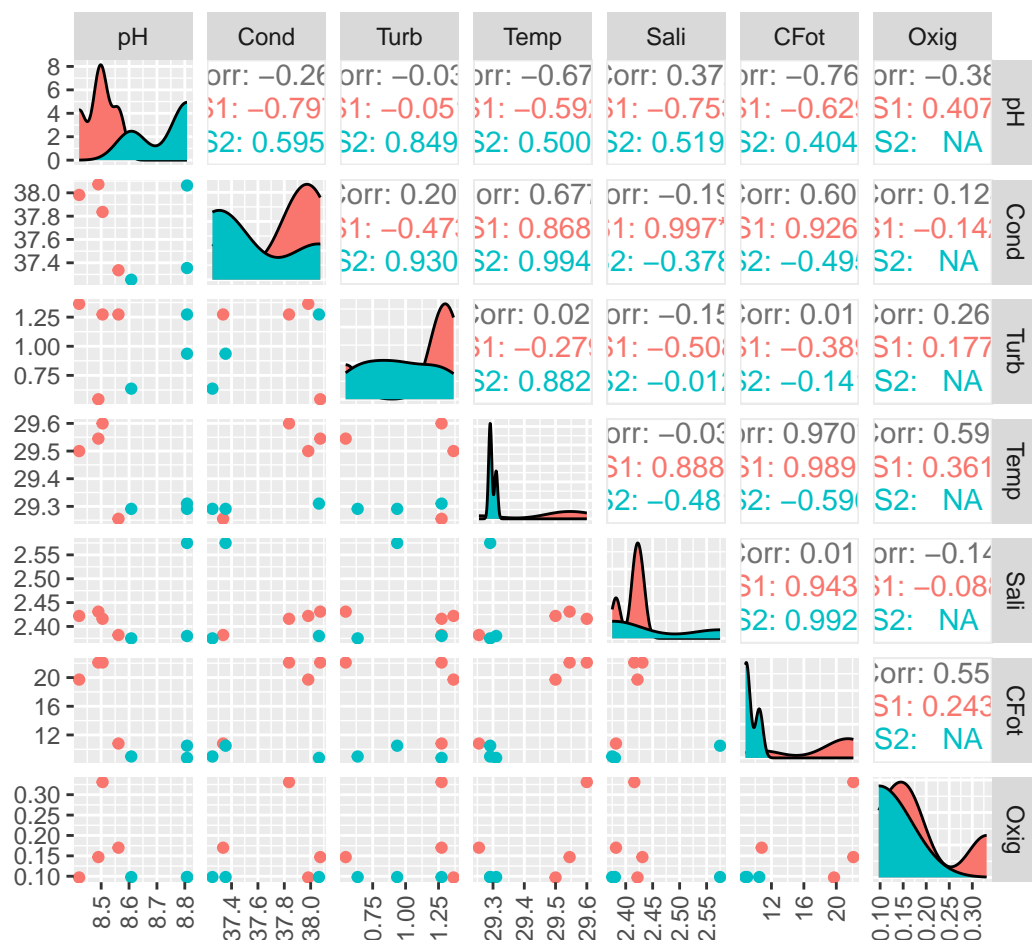


Figura 30.3: Correlaciones, dispersión y coeficientes de correlación, por cada grupo en comparación.

La Figura 31.5 permite visualizar a una de las relaciones relevantes, diferenciando por tipos de sitios (S1 y S2).

```
# Relación trivariada - Lineal
panel.lm = function(x, y, ...) {
  tmp<-lm(y~x,na.action=na.omit)
  abline(tmp, lwd = 1.5, col= 2)
  points(x,y, ...)}

coplot(Temp ~ CFot | Sitio, pch=19,
        panel = panel.smooth, data=datos)
```

Given : Sitio

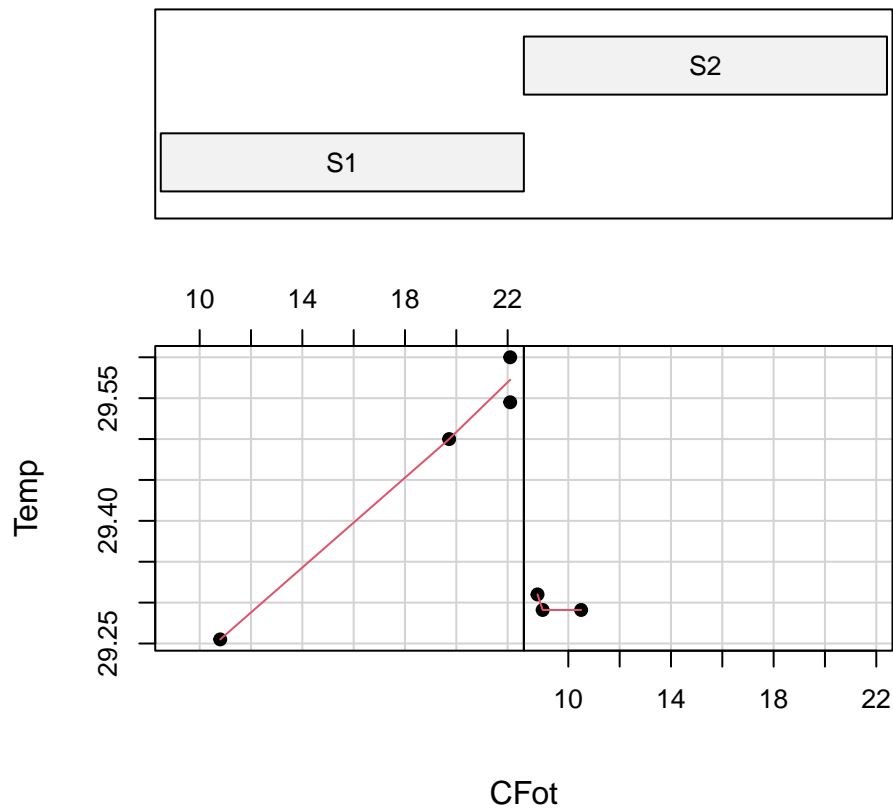


Figura 30.4: Relación bivariada en cada grupo asignado (Sitios).

La Figura 31.6 es otra forma de visualizar la relación anterior, pero con el paquete “ggplot2”

```
ggplot(datos, aes(x = CFot, y = Temp)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  facet_wrap(~Sitio) +
  theme_bw() +
  theme(panel.grid=element_blank())
```



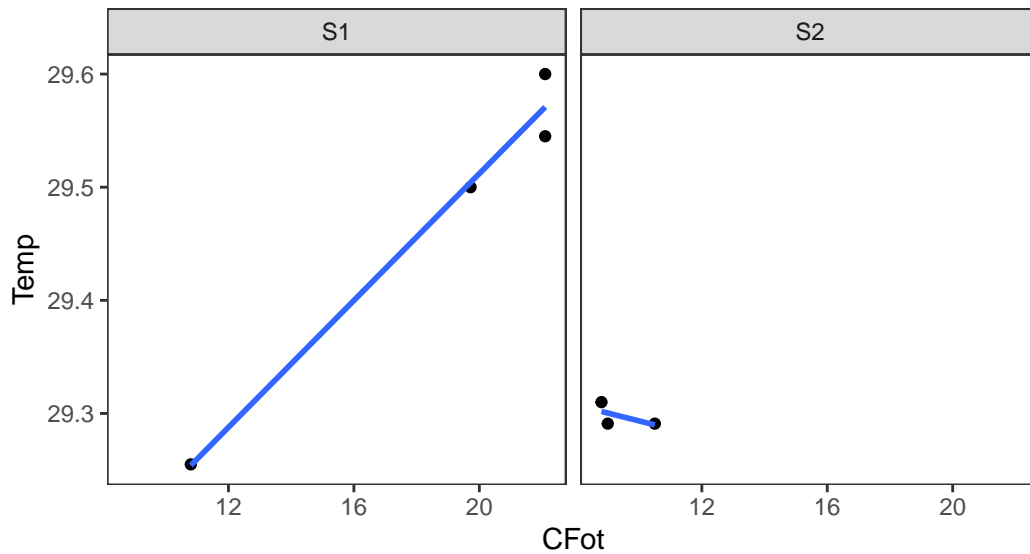


Figura 30.5: Relación bivariada en cada grupo asignado (Sitios).

Finalmente, la Figura 31.7 permite visualizar una relación más detallada entre dos variables seleccionadas y cuyos puntos caracterizan a cada uno de los sitios evaluados.

```
# Selección de una relación bivariada
names(datos)
```

```
[1] "Sitio" "pH"    "Cond"  "Turb"  "Temp"  "Sali"  "CFot"  "Oxig"
```

```
x11()
ggplot(datos, aes(x=CFot, y= Temp)) +
  geom_point(aes(color = Sitio), size=3) +
  geom_smooth(method= "lm") +
  theme_bw() +
  theme(panel.grid=element_blank())
```

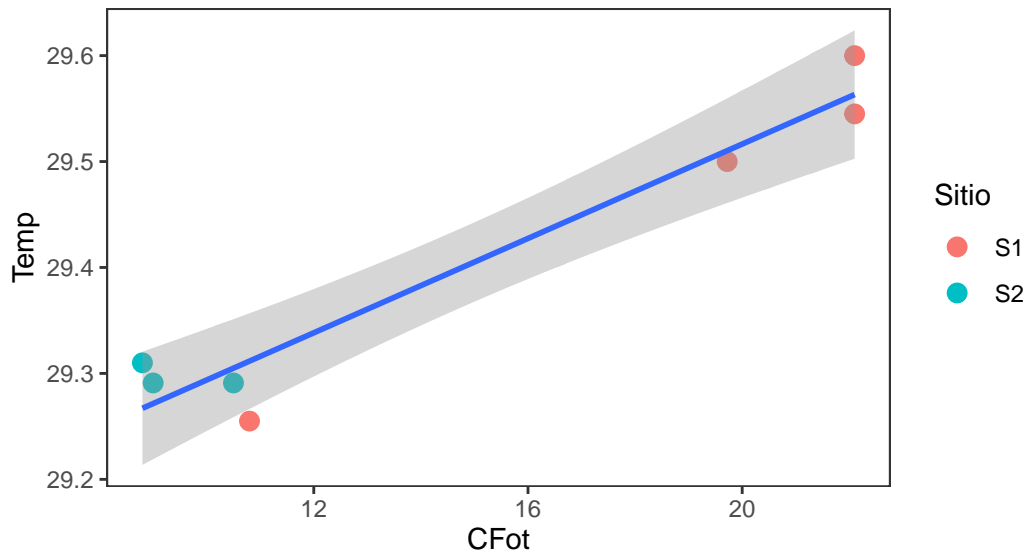


Figura 30.6: Relación bivariada en cada grupo asignado (Sitios).

### 30.0.2 Cuatro pasos para el análisis de clúster

A continuación se presenta el paso a paso requerido para un análisis de cluster - cla. Cabe mencionar que es un proceso algo dispendioso en tiempo, pero que brinda la posibilidad de contar con los códigos elaborados para ajustarlos de forma eficiente a otras bases de datos que requieran a este tipo de procedimientos.

#### 30.0.2.1 PASO 1. Distancia entre observaciones

Son muchas las distancias que pueden emplearse, pero cada una se ajusta al tipo de datos que se requieran trabajar. Para este caso se usará la **distancia euclídea**, debido a que se ajusta de manera apropiada a datos ambientales, incorporando además al comando **scale**, debido a que permite estandarizar a este tipo de variables que presentan escalas disímiles.

```
# Matriz de distancia
d.euclid <- dist(scale(datos[,c(2:8)]))
round(d.euclid,2)
```

	BTag	PBet	Mono	Gran	PGran	Rod
PBet	2.59					
Mono	2.94	3.14				
Gran	3.16	4.09	3.84			

PGran	3.94	3.75	4.82	2.13		
Rod	3.41	4.12	4.53	2.72	3.17	
Aero	4.47	4.54	5.22	3.47	3.30	3.55

### 30.0.2.2 PASO 2. Elección del método de agrupación de mayor ajuste

Son siete las opciones de dendogramas, de las cuales solo una será la que mejor se ajusta a los datos trabajados. Para ello, primero se realizarán los dendogramas y posteriormente se escogerá el de mejor ajuste con la correlación cofenética.

#### 30.0.2.3 2.1 Siete métodos de agrupamiento

```
# Método 1. Vecino más cercano "Cl.single", función "hclust" y método "single"
Cl.single <- hclust(d.euclid,method="single")

# Método 2. Vecino más lejano "Cl.complete", función "complete"
Cl.complete<-hclust(d.euclid,method="complete")

# Método 3. UPGMA función "average" Unión Promedio no Ponderado
Cl.upgma<-hclust(d.euclid,method="average")

# Método 4. UPGMC función "mcquitty" Unión Promedio Ponderado
Cl.upgmc<-hclust(d.euclid,method="mcquitty")

# Método 5. WPGMA función "centroid"
Cl.wpgma<-hclust(d.euclid,method="centroid")

# Método 6. WPGMC función "median"
Cl.wpgmc<-hclust(d.euclid,method="median")

# Método 7. WARD, función "ward"
Cl.ward<-hclust(d.euclid,method="ward.D")
```

#### 30.0.2.4 2.2 Figuras de los dendogramas con los siete métodos de agrupamiento

A continuación se realizará un panel que contenga hasta 4 figuras de dendogramas (Figura 31.8 y Figura 30.8), lo cual permite resumir al número de gráficas generadas, el comando que se

empleará para incluir a varias figuras en un mismo panel grafico es `grid.arrange()` del paquete `gridextra`.

```
x11()

f1 <- fviz_dend(Cl.single, k = 2,          # k grupos (opcionales)
  cex = 0.7,                             # tamaño del texto de las ramas
  ylab = "Distancia Euclídea", # Rotulo de la distancia
  main = "Vecino más Cercano - Single") # Rotulo de título

f2 <- fviz_dend(Cl.complete, k = 2,       # k grupos (opcionales)
  cex = 0.7,                             # tamaño del texto de las ramas
  ylab = "Distancia Euclídea", # Rotulo de la distancia
  main = "Vecino más Lejano - Complete") # Rotulo de título

f3 <- fviz_dend(Cl.upgma, k = 2,          # k grupos (opcionales)
  cex = 0.7,                             # tamaño del texto de las ramas
  ylab = "Distancia Euclídea", # Rotulo de la distancia
  main = "Unión Promedio no Ponderado - upgmc") # Rotulo de título

f4 <- fviz_dend(Cl.upgmc, k = 2,
  cex = 0.7,
  ylab = "Distancia Euclídea",
  main = "Unión Promedio Ponderado - upgmc")

grid.arrange(f1,f2,f3,f4, ncol = 2)
```

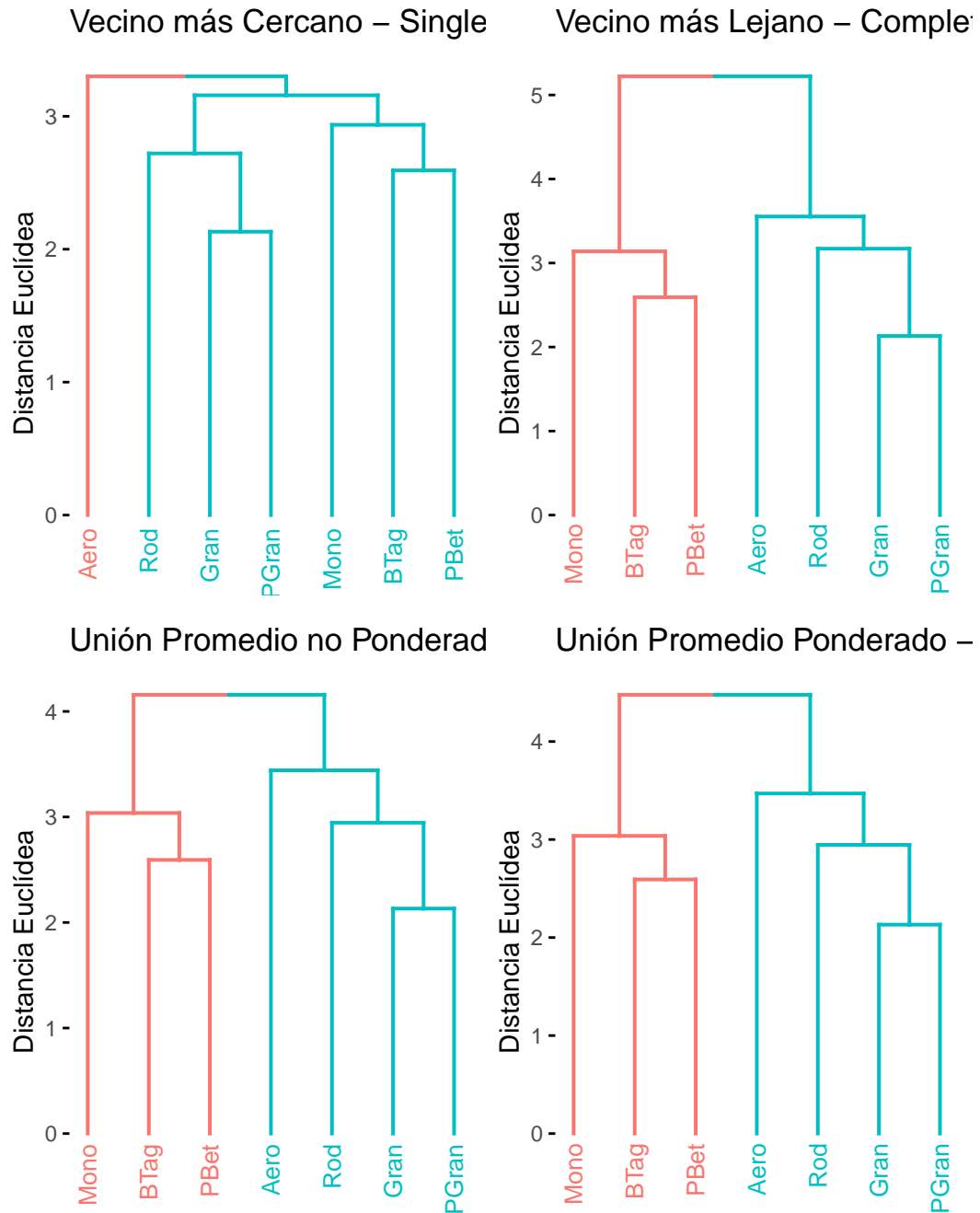


Figura 30.7: Cuatro dendrogramas jerárquicos con la distancia euclídea.

```

x11()

f5 <- fviz_dend(Cl.wpgma, k = 2,
  cex = 0.7,
  ylab = "Distancia Euclídea",
  main = "Unión Centroide no Ponderado - wpgma")

f6 <- fviz_dend(Cl.wpgmc, k = 2,
  cex = 0.7,
  ylab = "Distancia Euclídea",
  main = "Unión Centroide Ponderado - wpgmc")

f7 <- fviz_dend(Cl.ward, k = 2,
  cex = 0.7,
  ylab = "Distancia Euclídea",
  main = "Método de Ward")

grid.arrange(f5,f6,f7, ncol = 2)

```

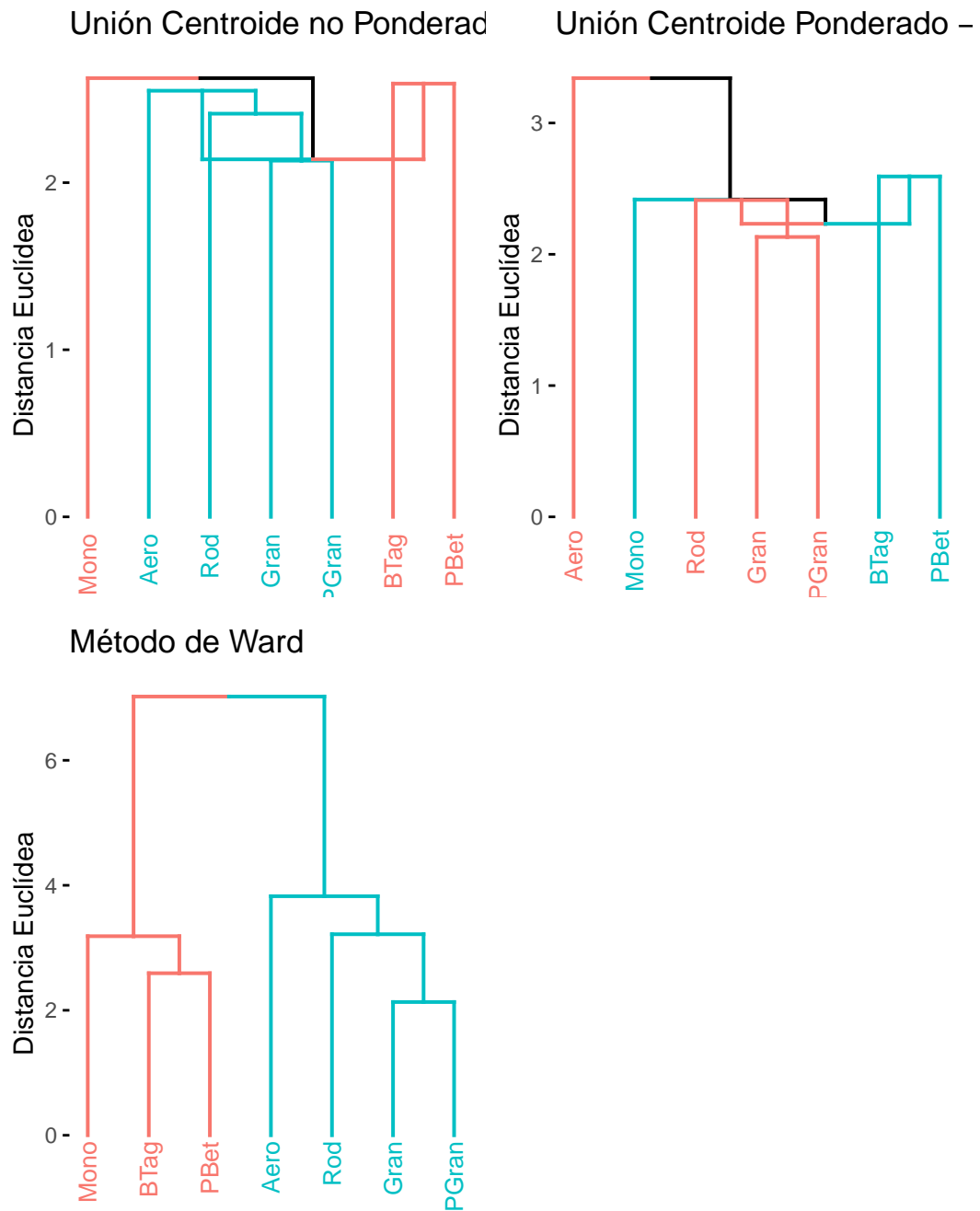


Figura 30.8: Tres dendogramas jerárquicos restantes, con la distancia euclídea.

### 30.0.2.5 2.3 Selección del mejor método de agrupamiento - Correlación Cofenética

- 2.3.1 Cálculo de las correlaciones cofenéticas

El método que presente la mayor correlación cofenética será el seleccionado. Para este caso se escogerá el dendograma con el método upgma, el cuál presenta un cofenético de 0.8234.

```
# (1) Correlación cofenpetica para "single"
cofenet1 <- cophenetic(Cl.single)
simple = cor(d.euclid,cofenet1)
simple
```

```
[1] 0.7358594
```

```
# (2) Correlación cofenética para "complete"
cofenet2<-cophenetic(Cl.complete)
compl = cor(d.euclid,cofenet2)
compl
```

```
[1] 0.8113013
```

```
# (3) Correlación cofenética para "average"
cofenet3<-cophenetic(Cl.upgma)
upgma = cor(d.euclid,cofenet3)
upgma
```

```
[1] 0.8233726
```

```
# (4) CCorrelación cofenética para "mcquitty"
cofenet4<-cophenetic(Cl.upgmc)
upgmc = cor(d.euclid,cofenet4)
upgmc
```

```
[1] 0.8209463
```

```
# (5) Correlación cofenética para "centroid"
cofenet5<-cophenetic(Cl.wpgma)
```



```
wpgma = cor(d.euclid,cofenet5)
wpgma
```

```
[1] 0.02444114
```

```
# (6) Correlación cofenética para "median"
cofenet6<-cophenetic(Cl.wpgmc)
wpgmc = cor(d.euclid,cofenet6)
wpgmc
```

```
[1] 0.3397504
```

```
# (7) Correlación cofenética para "ward"
cofenet7<-cophenetic(Cl.ward)
ward = cor(d.euclid,cofenet7)
ward
```

```
[1] 0.79712
```

- 2.3.2 Tabulación de las correlaciones cofenéticas

Los siguientes comandos permitirán organizar a los siete métodos de agrupamiento, de acuerdo a su nivel de correlación cofenética.

```
# data frame con cofenéticos
cofeneticos = data.frame(simple,compl,upgma,upgmc,
                        wpgma,wpgmc,ward)

# cofenéticos por cada métodos (Met)
cofenet=data.frame(Met = 1:7,Cofen=t(round(cofeneticos,3)))

# tabla con orden descendente de cofenéticos
cof_ordenado = cofenet[order(cofenet$Cofen, decreasing = TRUE), ]
cof_ordenado
```

	Met	Cofen
upgma	3	0.823
upgmc	4	0.821
compl	2	0.811

```
ward      7 0.797
simple     1 0.736
wpgmc     6 0.340
wpgma     5 0.024
```

Con este comando se puede exportar la tabla de cofenéticos como un archivo plano de csv.

```
# guardar tabla como csv
# write.csv2(cof_ordenado,"cofenet.csv")
```

### • 2.3.3 Figuras de algunas correlaciones cofenéticas vs. matriz de distancia

A continuación se presenta una muestra de la relación entre las matrices de distancia cofenética y de distancia euclínea, que permitió seleccionar al mejor método de agrupamiento (Figura 30.9).

```
# convertir matricesde distancia a vectores
d.euclid <- as.vector(d.euclid)
d.cofenet1 <- as.vector(cofenet1)
d.cofenet2 <- as.vector(cofenet2)
d.cofenet3 <- as.vector(cofenet3)
d.cofenet4 <- as.vector(cofenet4)

# crear un data frame con los vectores y agregar una columna de etiquetas
simple1 <- data.frame(d.euclid, d.cofenet1, d.cofenet2, d.cofenet3, d.cofenet4)
head(simple1)
```

	d.euclid	d.cofenet1	d.cofenet2	d.cofenet3	d.cofenet4
1	2.592758	2.592758	2.592758	2.592758	2.592758
2	2.935826	2.935826	3.138967	3.037397	3.037397
3	3.157893	3.157893	5.222324	4.157454	4.476721
4	3.936806	3.157893	5.222324	4.157454	4.476721
5	3.412345	3.157893	5.222324	4.157454	4.476721
6	4.469605	3.300609	5.222324	4.157454	4.476721

```
# Figuras correlaciones cofenéticas
x11()
# (1) distancia cofenética para "unión simple"
f1<-ggplot(simple1, aes(d.euclid,d.cofenet1))+
  geom_point(size=3, color="#4daf4a") +
```

```

geom_smooth(method="lm",se=FALSE,color="#377eb8") +
geom_smooth(method="loess",se=FALSE,color="#e41a1c",lty=2,size=1.3) +
labs(title= "Unión Simple",
      subtitle= paste("Correlación cofenética",
                      round(cor(d.euclid,cofenet1),4)),
      x="Distancia Euclidea",
      y="Distancia cofenética") +
theme_bw()

# (2) distancia cofenética para "unión completa"
f2<-ggplot(simple1, aes(d.euclid,d.cofenet2))+
geom_point(size=3, color="#4daf4a") +
geom_smooth(method="lm",se=FALSE,color="#377eb8") +
geom_smooth(method="loess",se=FALSE,color="#e41a1c",lty=2,size=1.3) +
labs(title= "Unión Completa",
      subtitle= paste("Correlación cofenética",
                      round(cor(d.euclid,cofenet2),4)),
      x="Distancia Euclidea",
      y="Distancia cofenética") +
theme_bw()

# (3) distancia cofenética para "unión upgma"
f3<-ggplot(simple1, aes(d.euclid,d.cofenet3))+
geom_point(size=3, color="#4daf4a") +
geom_smooth(method="lm",se=FALSE,color="#377eb8") +
geom_smooth(method="loess",se=FALSE,color="#e41a1c",lty=2,size=1.3) +
labs(title= "Unión promedio no ponderado - upgma",
      subtitle= paste("Correlación cofenética",
                      round(cor(d.euclid,cofenet3),4)),
      x="Distancia Euclidea",
      y="Distancia cofenética") +
theme_bw()

# (4) distancia cofenética para "unión upgmc"
f4<-ggplot(simple1, aes(d.euclid,d.cofenet4))+
geom_point(size=3, color="#4daf4a") +
geom_smooth(method="lm",se=FALSE,color="#377eb8") +
geom_smooth(method="loess",se=FALSE,color="#e41a1c",lty=2,size=1.3) +
labs(title= "Unión promedio ponderado - upgmc",
      subtitle= paste("Correlación cofenética",

```

```

        round(cor(d.euclid,cofenet4),4)),
        x="Distancia Euclidea",
        y="Distancia cofenética") +
        theme_bw()

grid.arrange(f1,f2,f3,f4, ncol = 2)

```

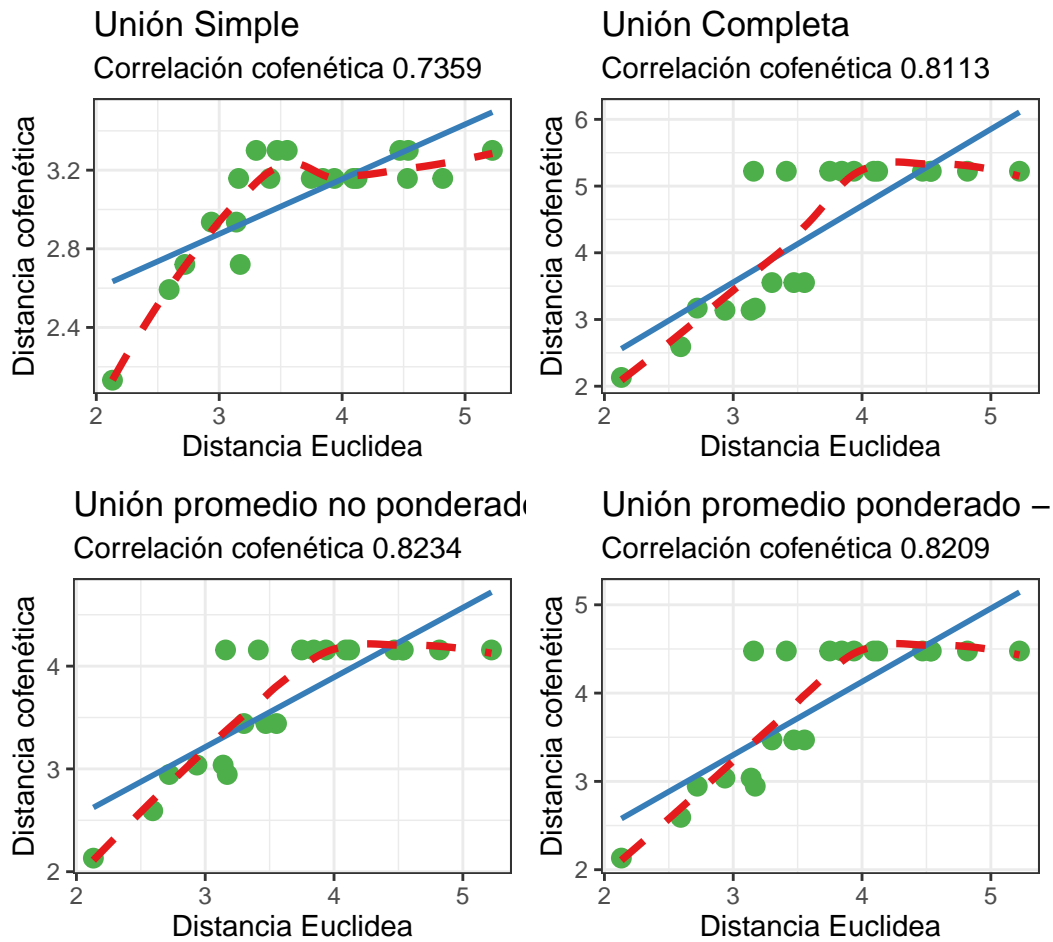


Figura 30.9: Cuatro regresiones entre la distancia euclídea empleada y las distancias cofenéticas de cada método de agrupamiento definido.

### 30.0.2.6 PASO 3. Número de grupos formados

La definición de los grupos formados, representan un insumo importante de información, debido a que permiten generar los k cluster en los que se agrupan las observaciones, basado en las

variables que las caracterizan. Este insumo es relevante además, como paso previo a otras técnicas que requieren los grupos definidos a priori, como los análisis discriminantes lineales (lda) o los análisis de varianza multivariados (manovas), de igual forma, a partir de los grupos se pueden responder hipótesis enfocadas en las variaciones que pueden presentar las variables a lo largo de gradientes discretos o en cluster.

### 30.0.2.7 Opción 1. Niveles de Fusión.

La figura de niveles de fusión es una de las más utilizadas para la generación de grupos o de cluster, debido a la sencillez del componente gráfico, en el cual se definen los cluster o grupos (eje Y), dependiendo del escalón de mayor amplitud o distancia horizontal (eje X). En la Figura 30.10 se observa que la mayor amplitud se presenta en 2 k cluster, por lo cual, el dendograma seleccionado en el paso anterior se puede clasificar en dos grupos de observaciones.

```
# Base de variables a relacionar (amb)
amb <- datos[,c(2:8)]

# Crear un data.frame con los datos de altura, k y número de cluster
f1 <- data.frame(h = Cl.upgma$height, k = nrow(amb):2, cluster = nrow(amb):2)

# Crear el gráfico de dispersión y agregar etiquetas de texto
ggplot(f1, aes(x = h, y = k, label = cluster)) +
  geom_point(color = "grey") +
  geom_text(color = "red", size = 3, vjust = -0.5) +
  geom_step(color = "grey", direction = "vh") +

# Personalizar el gráfico con títulos, etiquetas de ejes y paleta de colores
ggtitle("Niveles de Fusión - Distancia Euclídea - UPGMA") +
  ylab("k (Número de Cluster)") +
  xlab("h (Altura del Nodo)") +
  scale_color_manual(values = c("grey", "red")) +
  theme(axis.title = element_text(size = 16)) +
  theme_classic()
```

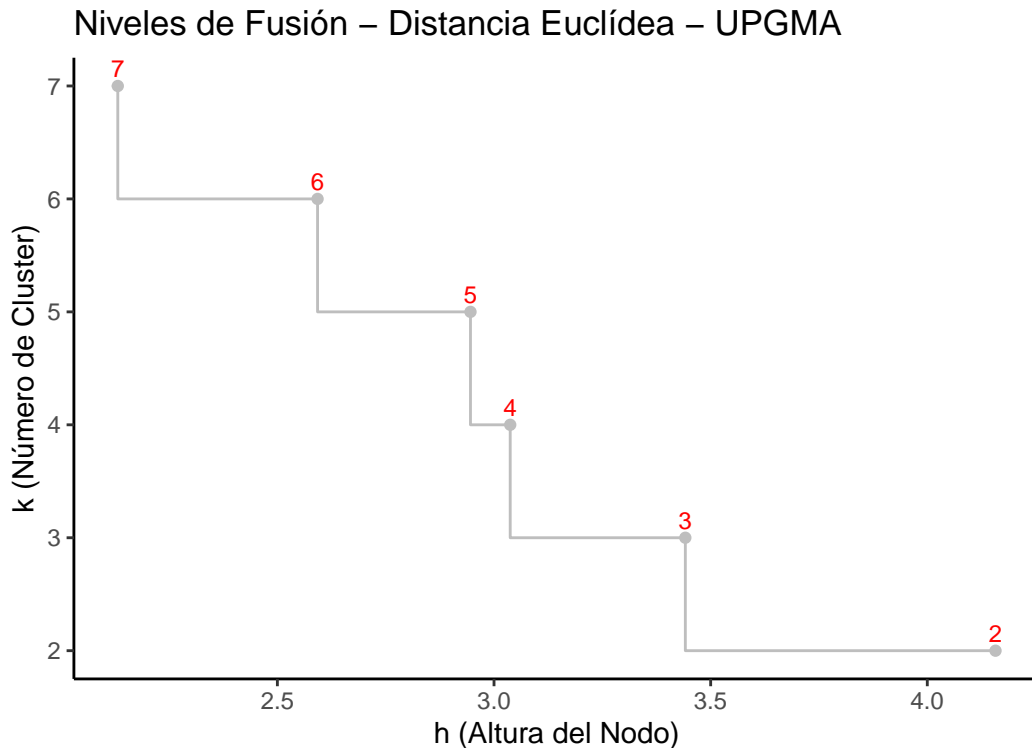


Figura 30.10: Cuatro regresiones entre la distancia euclídea empleada y las distancias cofenéticas de cada método de agrupamiento definido.

### 30.0.2.8 Opción 2. Número optimo de clusters de acuerdo al Ancho de silueta. Índice de calidad de Rousseeuw

La amplitud de silueta es de las opciones más usadas para definir al número de k cluster o grupos del dendograma realizado. En este ejercicio también se define a dos grupos. En caso que los resultados de esta técnica sean diferentes a la anterior, se suele decidir por esta, debido a su mayor grado de precisión.

```
# 1. Crear un vector vacío (amb.vacio) con asw valores
amb.vacio <- numeric(nrow(amb))

# 2. Silueta "sil"
for(k in 2: (nrow(amb)-1)){
  sil <- silhouette(cutree(Cl.upgma,k=k),d.euclid)
  amb.vacio[k]<-summary(sil)$avg.width}
```

```

# 3. Mejor o mayor amplitud de silueta (2 particiones)
k.mejor <- which.max(amb.vacio)
k.mejor

# Grafica de silueta
x11()
plot(1:nrow(amb),amb.vacio,type="h",
     main="Silueta-Número Óptimo de Clusters", xlab="(Número de grupos)",
     ylab="Amplitud promedio de silueta")

axis(1,k.mejor,paste("optimum",k.mejor,sep="\n"),col="red",
     font=2,col.axis="red")

points(k.mejor,max(amb.vacio),pch=16,col="red",cex=1.5)

cat("", "Silueta-Número óptimo de Clusters k=",k.mejor,
    "\n", "Con una amplitud promedio de silueta",max(amb.vacio),"\n")

```

### 30.0.2.9 3.1 Figura del dendograma jerárquico final

La Figura 30.11 muestra la manera en la que se organizan las observaciones en los dos grupos formados (ramas rojas y azules) debido a la naturaleza de las variables fisicoquímicas que las caracterizar.

```

# Dendograma final
x11()
fviz_dend(Cl.upgma, k = 2,          # k grupos
           cex = 0.9,              # tamaño del texto de las observaciones
           ylab = "Distancia Euclídea", # Rotulo de la distancia
           main = "Unión Promedio no Ponderada (UPGMA)", # Rotulo de título
           lower_rect = 0,          # Inicio de los rectángulos en cero
           k_colors = c("#00AFBB", "#FC4E07"),
           color_labels_by_k = TRUE, # Colores para cada grupo
           rect = TRUE)             # Rectángulos de cada grupo

```

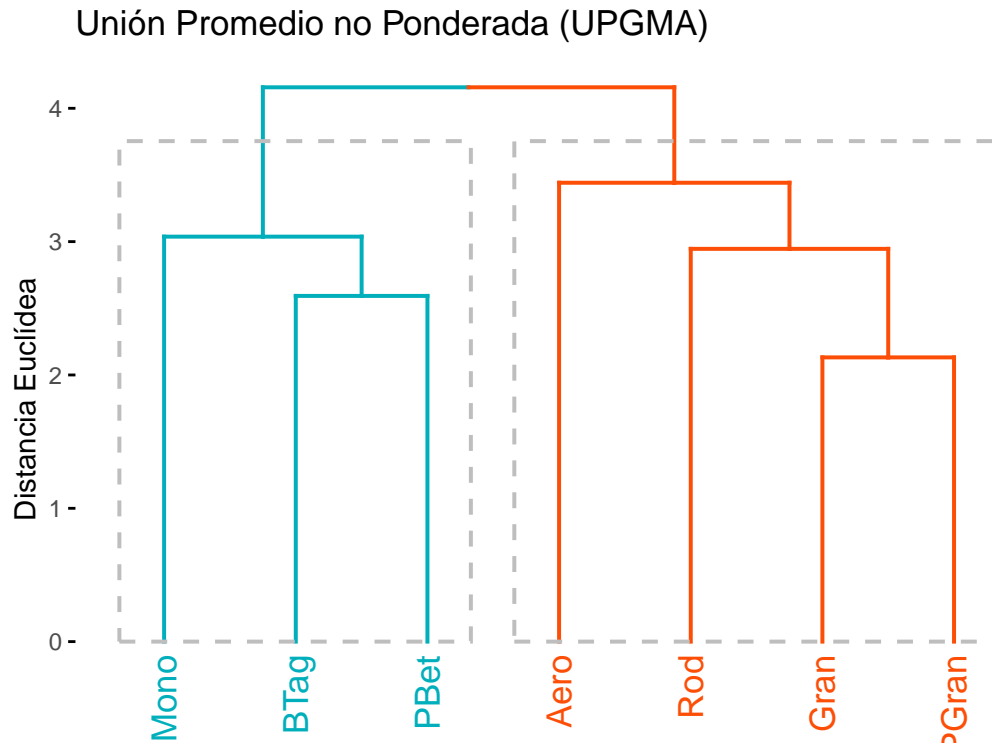


Figura 30.11: Dendrograma jerárquico final con los dos grupos asignados.

Vale la pena mencionar que estos análisis son importantes cuando se cuenta con pocas observaciones (ramas del dendrograma), en el caso contrario, es preferible utilizar dendrogramas no jerárquicos como el k-meas, el cual fue descrito en uno de los complementos del PCA y se retoma a continuación.

### 30.0.2.10 3.2 Figura del dendrograma no jerárquico final

- Agrupamiento elegido en el paso 2 (upgma)

```
# Matriz de distancia
d.euclid <- dist(scale(datos[,c(2:8)]))

# Método 3. UPGMA función "average" Unión Promedio no Ponderado
Cl.upgma<-hclust(d.euclid,method="average")
```

- Generación de la variable agrupadora (**gr**)



```
# Variable agrupadora con k=2 clúster
grp <- cutree(Cl.upgma, k = 2)      # Grupos generados "grp"
grl <- levels(factor(grp))          # Rotulos de los grupos formados
```

Este es un paso opcional en caso que se requiera insertar la nueva variable agrupadora a la base de datos en revisión.

```
# Incluir la variable agrupadora en la base de datos
datos.1=data.frame(grp,datos)      # Nuevo dataframe con la variable agrupadora (gr)
head (datos.1)
```

	grp	Sitio	pH	Cond	Turb	Temp	Sali	CFot	Oxig
BTag	1	S1	8.421	37.982	1.364	29.500	2.422	19.72	0.097
PBet	1	S1	8.490	38.073	0.545	29.545	2.431	22.10	0.147
Mono	1	S1	8.505	37.836	1.273	29.600	2.416	22.10	0.331
Gran	2	S1	8.562	37.336	1.273	29.255	2.382	10.80	0.170
PGran	2	S2	8.608	37.255	0.636	29.291	2.375	9.00	0.098
Rod	2	S2	8.808	38.063	1.273	29.310	2.380	8.80	0.098

- Generación del clúster No Jerárquico (**K-Means**)

La Figura 30.12 es la forma no jerárquica de presentar los resultados del cluster definido por el método de agrupamiento upgma.

```
x11()
fviz_cluster(list(data = amb, cluster = grp),
              palette = c("#2E9FDF", "#FC4E07"), # Colores para cada grupo
              ellipse.type = "confidence",        # Elipses
              repel = TRUE,                       # Elimina solapamiento de observaciones
              show.clust.cent = FALSE,            # Muestra a los clúster centrados
              ggtheme = theme_bw())              # Tipo de fondo tomado de ggplot2
```

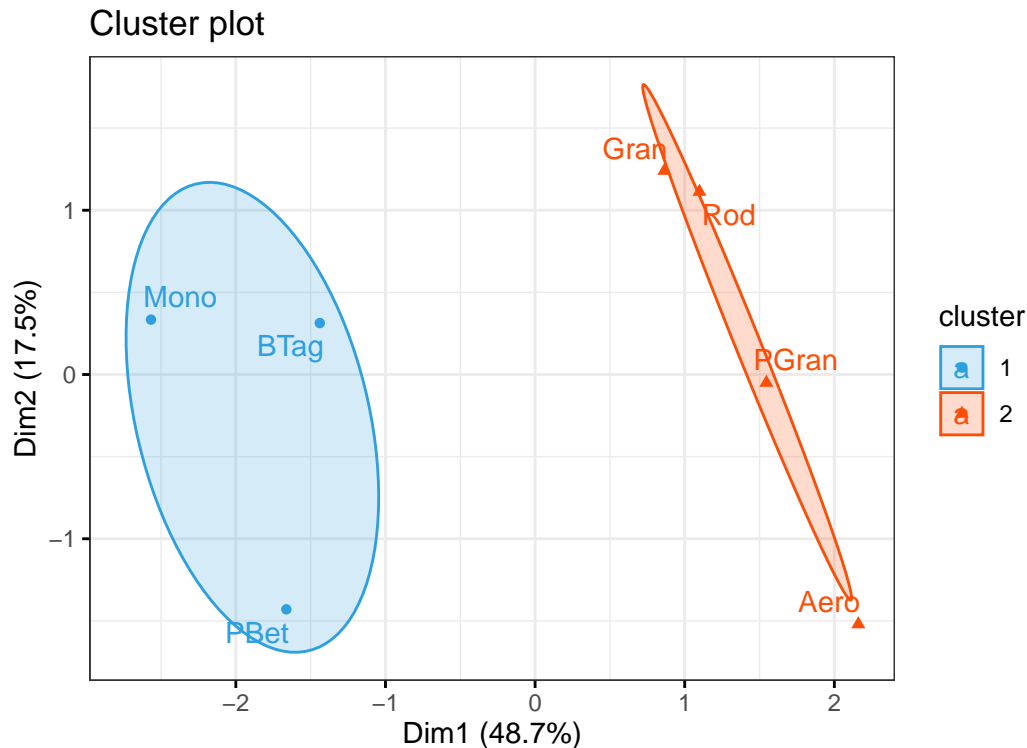


Figura 30.12: Dendrograma no jerárquico final con los dos grupos asignados.

Una pregunta interesante que se podría resolver, sería valorar si las diferencias entre los dos grupos de observaciones formadas es estadísticamente significativa, para lo cual se debe aplicar un análisis de varianza multivariado (manova).

#### 30.0.2.11 Paso 4. Variables de mayor contrinución a la clasificación

A continuación se realizan diferentes opciones de **mapas de calor** (Figura 30.13, Figura 30.14), para identificar a las variables con mayor relevancia en la clasificación realizada anteriormente en el dendrograma seleccionado. Este paso es relevante cuando se quiere ponderar o seleccionar a las variables que aportan al análisis, resumen de esta forma, la dimensionalidad del problema (número de variables).

```
amb1 <- as.matrix(amb)

# Opción 1. Mapa de calor con paquete "stats"
x11()
```

```

hv <- heatmap(amb1, margins=c(7,6),
  distfun = dist,
  xlab="Variables fisico-químicas",
  ylab= "Bahías",
  main = "Clasificación de Bahías",
  scale = "row") # Estandariza variables diferentes.

```

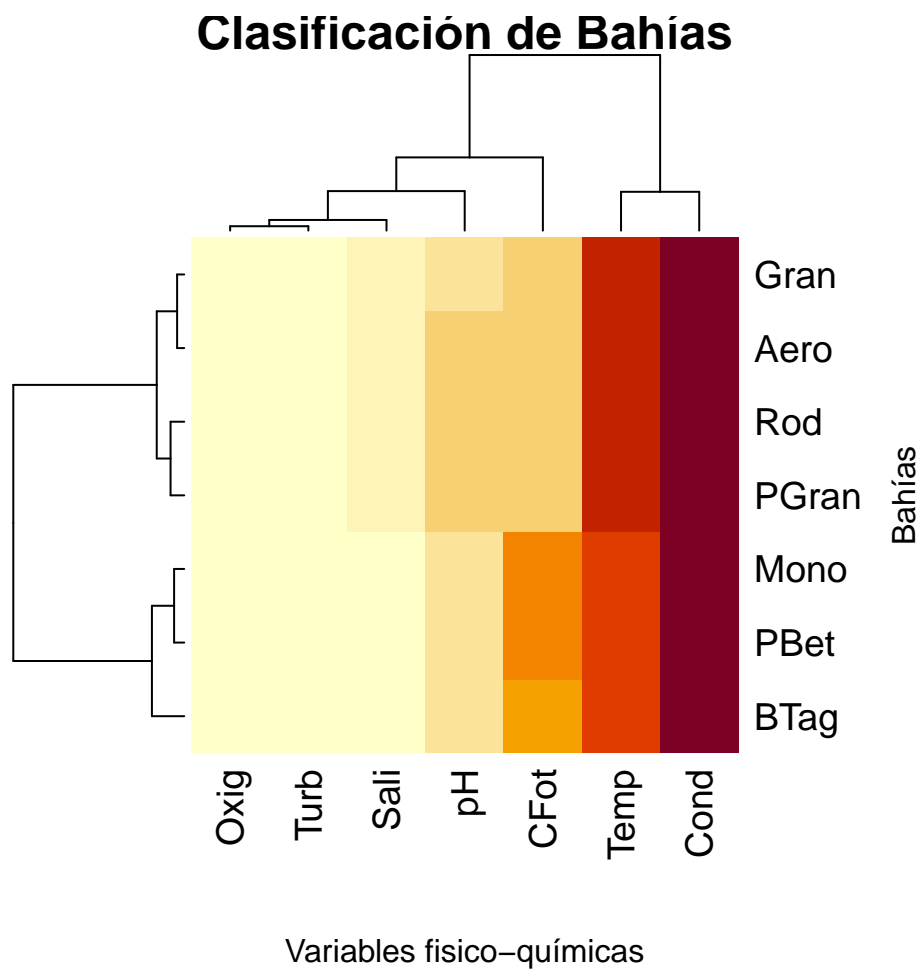


Figura 30.13: Mapa de calor que define en color rojo a las variables de mayor aporte a la clasificación realizada.

El siguiente mapa de calor (Figura 30.14) incorpora a la distancia euclídea utilizada y el método de agrupamiento seleccionado (upgma).

```

# Opción 2. Mapa de calor con paquete "stats"
hclust.fq <- function(amb1) hclust(amb1, method="average") # Inserción de agrupación UPGMA

x11()
heatmap.2(amb1, # Base de datos en formato matricial
  margins=c(7,7), # Margenes de la figura
  scale = "row", # Estandariza variables diferentes.
  col = bluered(100), # Colores del mapa de calor
  xlab ="Variables fisico-químicas",
  ylab= "Bahías",
  main = "Clasificación de Bahías",
  trace = "none",
  density.info = "none",
  distfun = dist, # Se puede usar vegdist de "vegan"
  hclustfun=hclust.fq) # Agrupamiento UPGMA

```

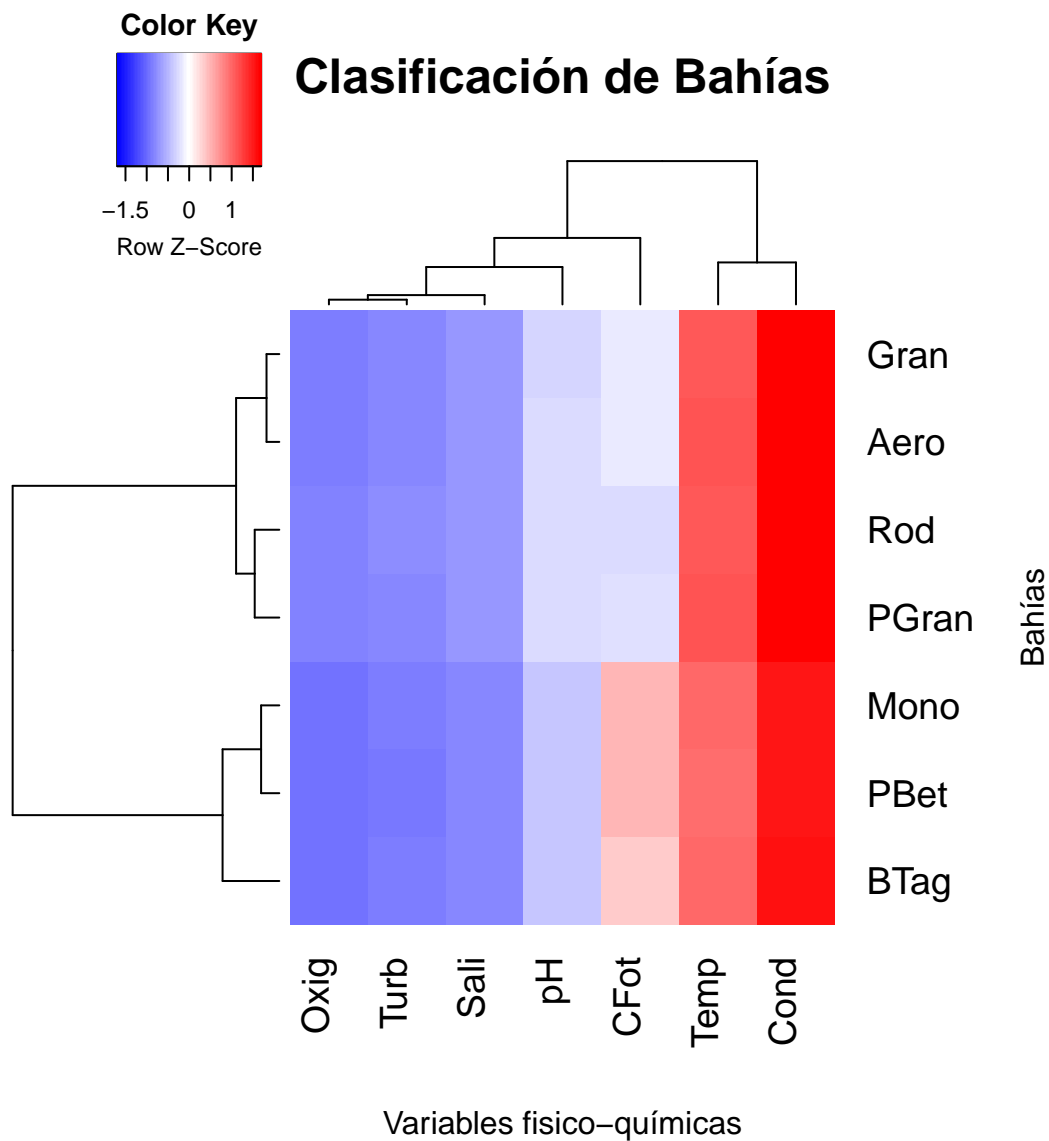


Figura 30.14: Mapa de calor que define en color rojo a las variables de mayor aporte a la clasificación realizada.

**Nota:** Es posible hacer mapas de calor cruzando a las variables con los grupos asignados. Este procedimiento se presentará en el ejercicio de análisis discriminante.

## 31 Taller 9.1 Análisis Discriminante Lineal - LDA

### Objetivo de la actividad:

La base de datos que se utilizará es la de medidas morfométricas de peces de un estudio realizado con peces de la india por **Gupta et al. (2018)** [Artículo fuente](#) en los que se validó la taxonomía de peces de la subfamilia Barbinae, utilizando 19 variables morfométricas y 19 variables merísticas, correspondientes a 5 Especie de la familia en mención.

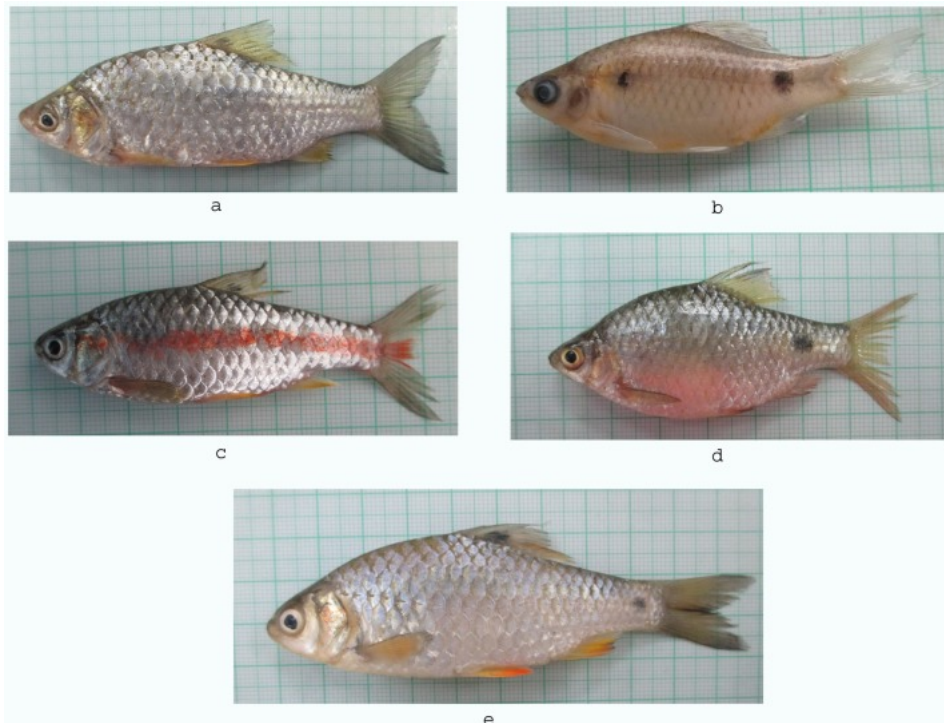


Figura 31.1: Imágen tomada de *Gupta et al. (2018)* (a) *S . Sarana* (b) *P . ticto* (c) *P . sóforo* , (d) *P . conconio* y (e) *P . chola*

El **objetivo** de este ejercicio consiste en identificar a las variables morfométricas que mejor discriminan a las Especie de peces y si dichas Especie se encuentran bien discriminadas en sus grupos taxonómicos asignados. De igual forma se construirá un modelo lineal en el que

se puedan incluir nuevos individuos tomados de la misma muestra de peces y puedan ser discriminados de manera eficiente. La base de datos que se utilizará es **peces.csv**.

### 31.0.1 Referencias bibliográficas de apoyo.

[Libro: Análisis de datos ecológicos y ambientales - Rodríguez-Barrios Javier 2023](#) Ver el capítulo de discriminante lineal (lda) en donde se detallan los procedimientos descritos en el presente ejercicio.

[Sigatoka en cultivos de banano - Aguirre et al. \(2015\)](#). Análisis de discriminante canónico y algunas técnicas multivariadas complementarias.

[Linear Discriminant Analysis in R](#) Se brinda información sobre las generalidades de los lda y su aplicación en R.

[Computing and visualizing LDA in R](#) En este documento se brinda información sobre el análisis y la visualización gráfica del lda.

### 31.0.2 Cargar las librerías requeridas

```
# Librerías requeridas
library(tidyverse)
library(lattice)
library(corrplot)
library(ggplot2)
library(ggrepel)
library(reshape2)
library(ggforce)

library(ade4)
require(vegan)
library(car)
library(MASS)
library(candisc)
library(mvnormtest)
```

### 31.0.3 Cargar o importar la base de datos

La presente base de datos se encuentra en formato plano de **csv**, presenta una columna **Especie** que agrupa a las 5 Especie de peces, otra columna **Grupo**, que asigna un número a cada especie

y posteriormente a las 19 variables morfométricas y 10 variables merísticas, de las cuales se seleccionarán las 19 morfométricas para este ejercicio M.1 a M.19.

```
# Base de datos
peces<-read.csv2("peces.csv",row.names=1)
names(peces)
```

```
[1] "Especie" "Grupo"   "M.1"     "M.2"     "M.3"     "M.4"     "M.5"
[8] "M.6"     "M.7"     "M.8"     "M.9"     "M.10"    "M.11"    "M.12"
[15] "M.13"    "M.14"    "M.15"    "M.16"    "M.17"    "M.18"    "M.19"
[22] "M.20"    "M.21"    "M.22"    "M.23"    "M.24"    "M.25"    "M.26"
[29] "M.27"    "M.28"    "M.29"
```

### 31.0.4 Exploración de los datos

Para este ejemplo se utilizarán figuras que relacionan parejas de variables y figuras de cajas que permitan visualizar diferencias entre las Especie de peces de acuerdo a su morfometría. \*Para facilidad del ejercicio se seleccionarán algunas variables morfométricas - `peces1`, debido a que son las que presentan mejores patrones lineales.

```
# Elipses con colores con variables morfométricas
peces1 <- peces[,c(3:9,15,17,20)]
M <- cor(peces1) # Matriz de Correlación (M)
```

La Figura 32.2 permite visualizar las relaciones lineales entre todas las parejas de variables, incluyendo a los coeficientes de correlación de Pearson.

```
x11()
corrplot(M, method = "circle", # Correlaciones con círculos
          type = "lower", insig="blank", # Forma del panel
          order = "AOE", diag = FALSE, # Ordenar por nivel de correlación
          addCoef.col = "black", # Color de los coeficientes
          number.cex = 0.6, # Tamaño del texto
          col = COL2("RdYlBu", 200)) # Transparencia de los círculos
```



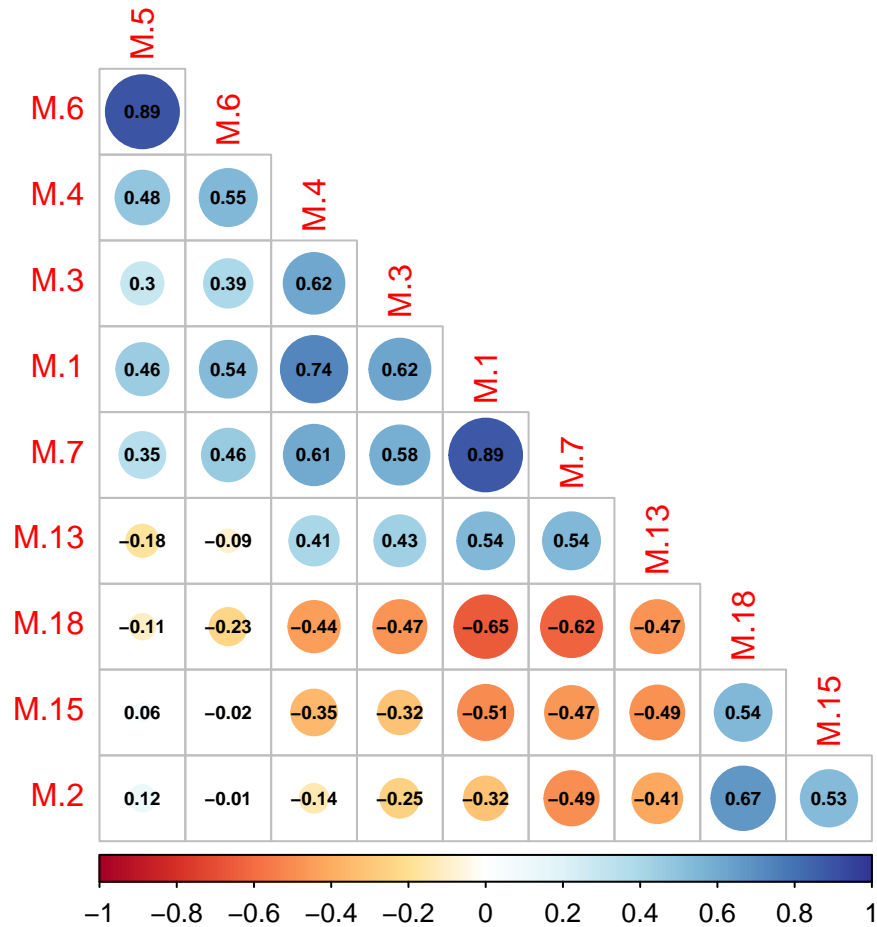


Figura 31.2: Correlaciones y coeficientes de correlación.

La Figura 32.3 a diferencia de la anterior, clasifica a los grupos por colores y además incluye a sus coeficientes de correlación y el patrón de distribución de cada variable mediante histogramas de densidad.

```

peces1 <- peces[,c(3:9,15,17,20)]
peces$Especie <- as.factor(peces$Especie)

x11()
pairs ((peces1),panel=function(x,y)
{abline(lsfit(x,y)$coef,lwd=2,col=3)
  lines(lowess(x,y),lty=2,lwd=2,col=2)
  points(x,y,col=peces$Especie, cex=1.4,pch=19,lwd=0.6)})

```

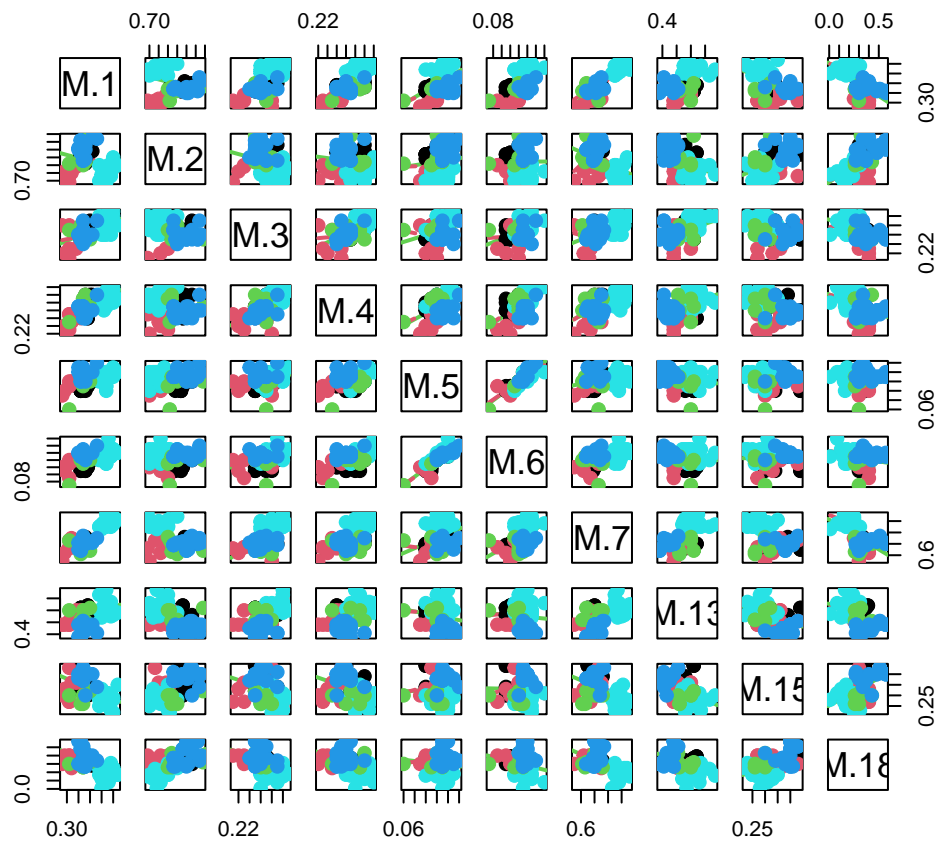


Figura 31.3: Relación entre parejas de variables, por cada grupo en comparación (colores).

La Figura 32.4 permite visualizar la resolución de cada variable para diferenciar o discriminar a las diferentes especies de peces. Esta figura sirve de insumo para descartar aquellas variables con poco potencial de discriminación de las especies.

```
# Figuras multivariadas de Cajas y bigotes
library(reshape)

x11()
ggplot(melt(peces[,c(1,3:9,15,17,20)]), aes(x=variable, y=value)) +
  geom_boxplot(aes(fill=Especie)) +
  scale_fill_manual(values = c('#fc8d59', '#ffffbf', '#99d594', '#377eb8', '#33a02c')) +
  labs(x="", y="Morfometría") +
```

```
facet_wrap(~ variable,scales="free") +
theme_bw()
```

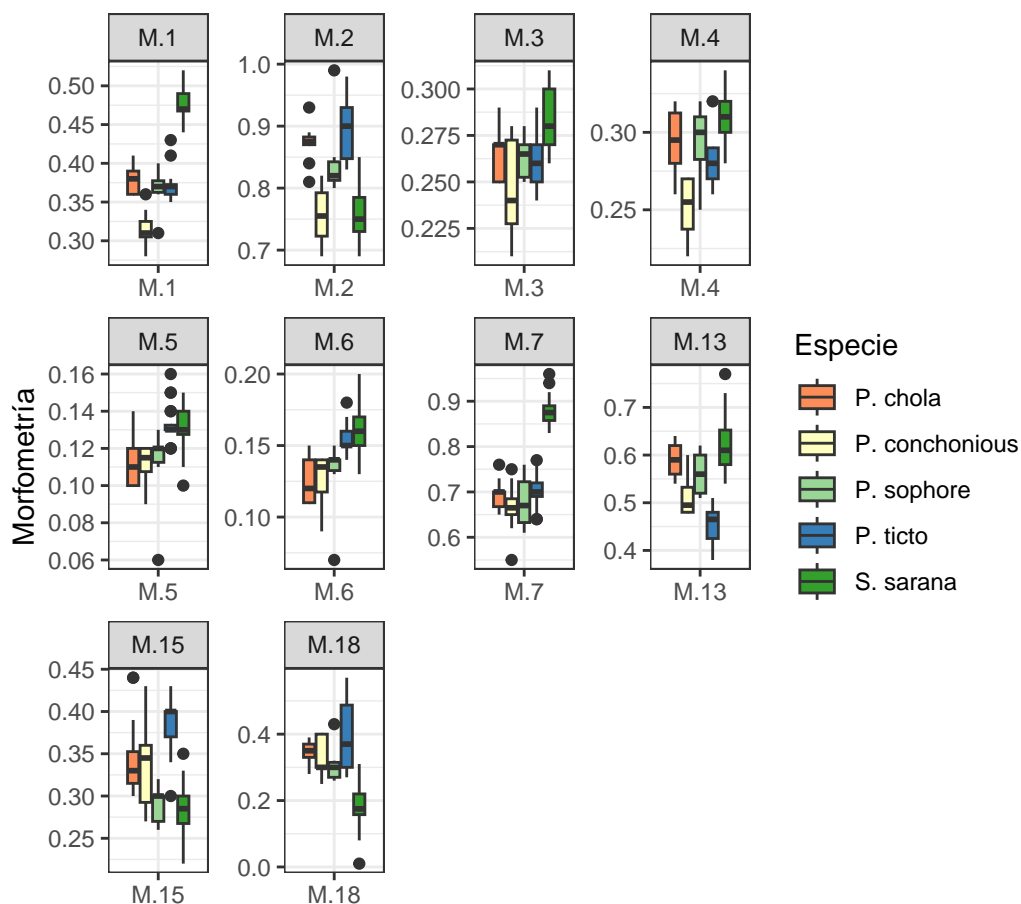


Figura 31.4: Variación en la morfometría de los peces, de acuerdo a cada una de las variables seleccionadas.

#### 31.0.4.1 Mapa de Calor

El siguiente mapa de calor también permite visualizar a la resolución de las variables morfométricas para diferenciar a las especies de peces, las cuales representan a los grupos en comparación. Con los siguientes comandos se calculará una tabla que resume a los promedios de las 10 variables morfométricas para cada especie evaluada.

```
# Extracción de los promedios de las variables para cada especie
library(tidyverse)
promedios <- peces %>%
  subset(select = c("Especie", "M.1", "M.2", "M.3", "M.4", "M.5", "M.6", "M.7", "M.13", "M.15", "M.18"),
    na.omit() %>%
    group_by(Especie) %>%
    summarize(across(everything(), mean))

promedios <- data.frame(promedios) # Guardar promedios como dataframe
# promedios
```

A continuación se convierte el dataframe a formato matricial - `promedios2`, para poder ser graficado en el mapa de calor.

```
# Seleccionar columnas de 2 a 10 del data frame peces1 y convertirlas en matriz
promedios2 <- promedios %>%
  subset(select = c(2:11)) %>%
  as.matrix()
round(promedios2, 2)
```

	M.1	M.2	M.3	M.4	M.5	M.6	M.7	M.13	M.15	M.18
1	0.38	0.87	0.27	0.30	0.11	0.12	0.69	0.59	0.35	0.35
2	0.32	0.76	0.25	0.25	0.11	0.13	0.66	0.52	0.34	0.33
3	0.36	0.85	0.26	0.29	0.11	0.13	0.68	0.56	0.29	0.31
4	0.37	0.90	0.26	0.28	0.13	0.15	0.70	0.45	0.39	0.40
5	0.48	0.76	0.28	0.31	0.13	0.16	0.88	0.62	0.28	0.18

Ahora se incluyen los nombres de las especies a la matriz `promedios2`.

```
# Asignar los valores de la primera columna de peces1 como nombres de fila en la matriz pe
rownames(promedios2) <- promedios[,1]
```

La Figura 31.5 permite visualizar a las variables que mejor discriminan a las especies de peces (variables de tonalidad rojiza).

```
# Figura del primer mapa de calor
x11()
hv <- heatmap(promedios2,
  margins=c(5,12),
  distfun = dist,
  xlab ="Variables morfométricas",
```

```
ylab= "Especie de Peces",
main = "Clasificación de Peces")
```

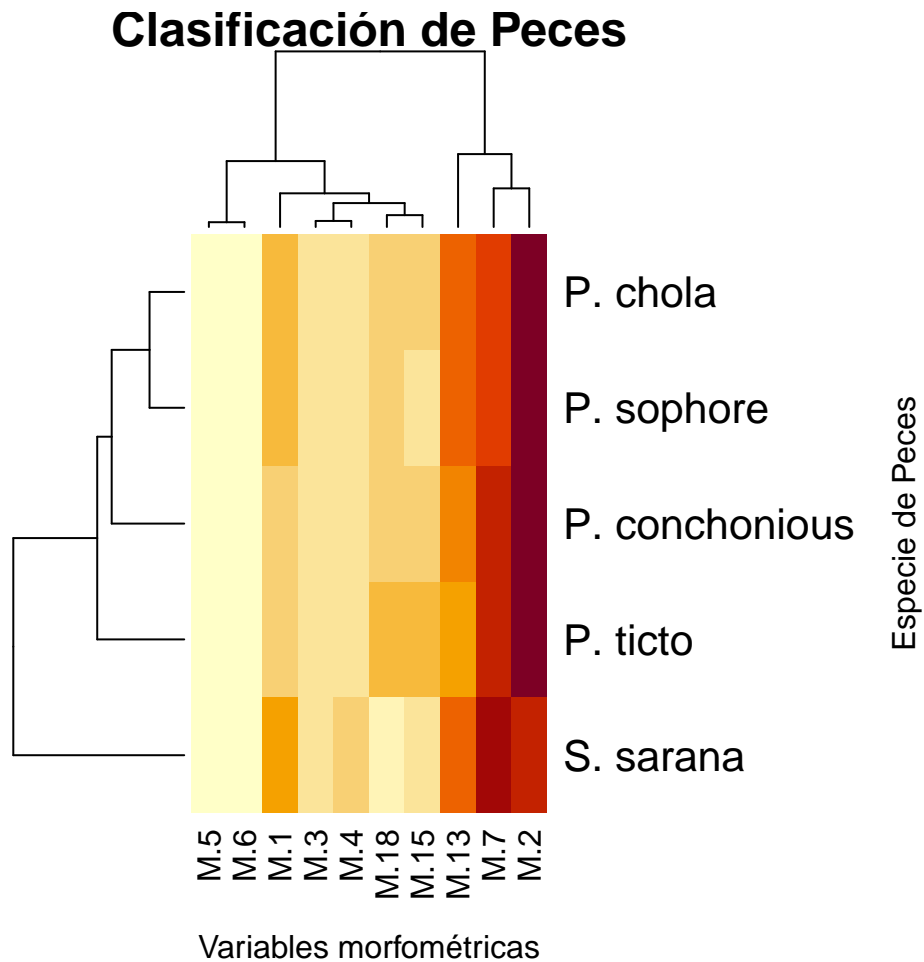


Figura 31.5: Mapa de calor que relaciona a las variables morfométricas y a las especies de peces.

La Figura 31.6 incorpora elementos adicionales como al método de agrupamiento `upgma`, asumiendo que puede ser el que mejor se ajusta a los datos de este ejercicio.

```
# Opción 2. Mapa de calor con paquete "stats"
hclust.fq <- function(promedios2)
  hclust(promedios2, method="average") # Inserción de UPGMA
```

```

library("gplots")
x11()
heatmap.2(promedios2,      # Base de datos en formato matricial
           margins=c(5,12), # Margenes de la figura
           scale = "row",   # Estandariza variables diferentes.
           col = bluered(100), # Colores del mapa de calor
           xlab="Variables morfométricas",
           ylab= "Especie de Peces",
           main = "Clasificación de Peces",
           trace = "none",
           density.info = "none",
           distfun = dist,      # Se puede usar vegdist de "vegan"
           hclustfun=hclust.fq) # Agrupamiento UPGMA

```

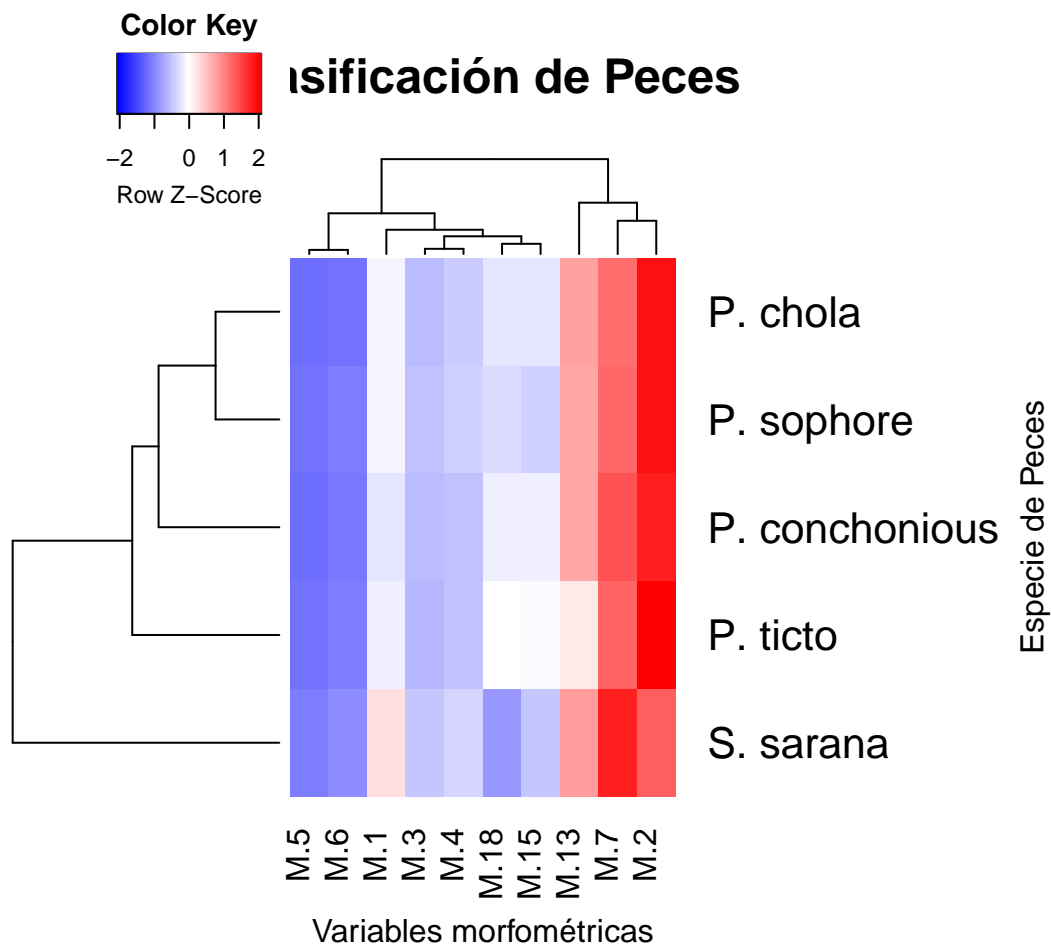


Figura 31.6: Mapa de calor que relaciona a las variables morfológicas y a las especies de peces.

### 31.0.5 Tres pasos para la realización del discriminante lineal - LDA

#### 31.0.5.1 Paso 1. Pruebas de supuestos

Para que el análisis discriminante lineal sea considerado como un modelo lineal, debe cumplir con los supuestos de normalidad multivariada y de homogeneidad de covarianzas. Para el caso del presente ejercicio, dichos supuestos no alcanzan a cumplirse con los diagnósticos utilizados (valor  $p < 0.05$ ), motivo por el cual, el lda de este ejercicio será tomado como una técnica de exploración multivariada para saber que tan bien discriminados quedan los individuos de cada especie, basado en las 10 variables morfológicas seleccionadas.

### 31.0.5.1.1 1.1 Supuesto de normalidad

El supuesto de normalidad multivariada será evaluado con el paquete `mvnrmtest`, el cual utiliza el estadístico de Shapiro Wilks Multivariado. Para ello se realizará esta prueba en cada uno de los grupos o especies en comparación.

```
# Diagnóstico de normalidad por cada tipo de Especie
library(mvnrmtest)
```

Los siguientes generan los dataframes de cada especie con las 10 variables seleccionadas, convirtiendola además en formato matricial.

```
# Dataframe por cada especie

# datos de P. chola.
P.chola  <- peces %>%
  filter(Especie == "P. chola") %>%
  subset(select = c("M.1", "M.2", "M.3", "M.4", "M.5", "M.6", "M.7", "M.13", "M.15", "M.

# datos de P. conchionious.
P.concho <- peces %>%
  filter(Especie == "P. conchionious") %>%
  subset(select = c("M.1", "M.2", "M.3", "M.4", "M.5", "M.6", "M.7", "M.13", "M.15", "M.

# datos de P. sophore.
P.sophore <- peces %>%
  filter(Especie == "P. sophore") %>%
  subset(select = c("M.1", "M.2", "M.3", "M.4", "M.5", "M.6", "M.7", "M.13", "M.15", "M.

# datos de P. ticto.
P.ticto  <- peces %>%
  filter(Especie == "P. ticto") %>%
  subset(select = c("M.1", "M.2", "M.3", "M.4", "M.5", "M.6", "M.7", "M.13", "M.15", "M.

# datos de P. ticto.
S.sarana <- peces %>%
  filter(Especie == "S. sarana") %>%
  subset(select = c("M.1", "M.2", "M.3", "M.4", "M.5", "M.6", "M.7", "M.13", "M.15", "M.
```

Vale la pena resaltar que los datos de las especies P. chola y P. sophore, son singulares, por lo cual no puede calcularse su supuesto de normalidad multivariada. Con el objeto de continuar en el ejercicio, las matrices que representan a las especies en mención, serán desactivadas con #.



```
# Prueba de normalidad para cada especie

# norm1 <- mshapiro.test(t(P.chola)) # Matriz singular
norm2 <- mshapiro.test(t(P.concho))
# norm3 <- mshapiro.test(t(P.sophore)) # Matriz singular
norm4 <- mshapiro.test(t(P.ticto))
norm5 <- mshapiro.test(t(S.sarana))
```

A continuación se resume el resultado de los tres diagnósticos de normalidad multivariada realizados. Vale la pena mencionar que ninguna especie cumple con dicho supuesto estadístico (valores  $p < 0.05$ ), aunque existe la posibilidad de probar con alguna transformación.

```
# Resumen de el diagnóstico de normalidad
(normalidad = data.frame(Norm.P.concho = norm2$p.value,
                        Norm.ticto    = norm4$p.value,
                        Norm.sarana   = norm5$p.value))
```

```
Norm.P.concho  Norm.ticto  Norm.sarana
1  0.001050723 2.629679e-08 5.169222e-07
```

### 31.0.5.2 1.2 Supuesto de homogeneidad de covarianzas

La prueba de homogeneidad de covarianza o **esfericidad**, corresponde al segundo supuesto del análisis discriminante lineal, se utilizará la función `betadisper`, la cual es complementada por dos análisis de varianza, los cuales definirán si el supuesto logra ser cumplido.

```
# Pruebas de Homogeneidad de covarianzas paquete "vegan"
peces.d <- dist(peces[,c(3:9,15,17,20)]) # Matriz de distancias
peces.homoge <- betadisper(peces.d, peces$Especie) # Permutest
```

Con la siguiente `anova` se obtiene un valor  $p$  de 0.016\*, lo cual indica que no se cumple el supuesto de homogeneidad de covarianzas (valor  $p < 0.05$ ).

```
# 1) Prueba con anova permutacional
anova(peces.homoge)
```

#### Analysis of Variance Table

```
Response: Distances
      Df    Sum Sq   Mean Sq F value    Pr(>F)
```

```
Groups      4 0.019804 0.0049509 3.281 0.01638 *
Residuals 65 0.098082 0.0015090
```

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Con el `permutest` se obtiene un valor p de 0.015\*, lo cual indica que tampoco se cumple el supuesto de homogeneidad de covarianzas (valor  $p < 0.05$ ).

```
# 2) Prueba permutacional
permutest(peces.homoge) # Se cumple el supuesto de homogeneidad
```

```
Permutation test for homogeneity of multivariate dispersions
```

```
Permutation: free
```

```
Number of permutations: 999
```

```
Response: Distances
```

```
      Df  Sum Sq  Mean Sq    F N.Perm Pr(>F)
Groups   4 0.019804 0.0049509 3.281   999 0.022 *
Residuals 65 0.098082 0.0015090
```

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

### 31.0.5.3 Paso 2. Análisis Discriminante Lineal de Fisher - LDA

A continuación, se realizará el lda, que permitirá generar definir al nivel de discriminación de cada grupo o especie de pez. Se presentan algunas opciones gráficas con el procedimiento general y con el análisis discriminante canónico (dca)

```
# Cálculo del LDA
names(peces)
```

```
[1] "Especie" "Grupo"  "M.1"    "M.2"    "M.3"    "M.4"    "M.5"
[8] "M.6"     "M.7"     "M.8"     "M.9"     "M.10"    "M.11"    "M.12"
[15] "M.13"    "M.14"    "M.15"    "M.16"    "M.17"    "M.18"    "M.19"
[22] "M.20"    "M.21"    "M.22"    "M.23"    "M.24"    "M.25"    "M.26"
[29] "M.27"    "M.28"    "M.29"
```

```
dis<-lda (Especie ~ M.1+M.2+M.3+M.4+M.5+M.6+M.7+M.13+M.15+M.18,
         data = peces)
round(dis$prior,2)
```

P. chola	P. conchionious	P. sophore	P. ticto	S. sarana
0.17	0.11	0.09	0.29	0.34

“*Prior probabilities of groups*” corresponde a la probabilidad de clasificación para cada grupo que dependerá del número de peces que lo conforman, aqueyas especies con mayor número de individuos censados, presentarán mayor probabilidad de discriminación.

Prbabilidad de clasificar indiv. de los cinco grupos: P. chola P. conchionious P. sophore P. ticto S. sarana 0.17 0.11 0.08 0.28 0.34

```
# Insumos del AD
# summary(dis)
```

A continuación se presentan los promedios de cada especie por cada variable morfométrica seleccionada para el análisis.

```
#Grupos de medias para las 4 variables
round(dis$means,2)
```

	M.1	M.2	M.3	M.4	M.5	M.6	M.7	M.13	M.15	M.18
P. chola	0.38	0.87	0.27	0.30	0.11	0.12	0.69	0.59	0.35	0.35
P. conchionious	0.32	0.76	0.25	0.25	0.11	0.13	0.66	0.52	0.34	0.33
P. sophore	0.36	0.85	0.26	0.29	0.11	0.13	0.68	0.56	0.29	0.31
P. ticto	0.37	0.90	0.26	0.28	0.13	0.15	0.70	0.45	0.39	0.40
S. sarana	0.48	0.76	0.28	0.31	0.13	0.16	0.88	0.62	0.28	0.18

Los autovalores son los que permiten definir las coordenadas de las variables para cada función canónica (eje o dimensión del lda), para la grafica del discriminante, se utilizarán las funciones LD1 y LD2, como ejes x y y respectivamente (ver Figura 31.7).

```
# Autovalores estandarizados (pesos de las variables en cada eje)
round((Cs <- dis$scaling),2)
```

	LD1	LD2	LD3	LD4
M.1	39.34	-16.59	26.35	21.80
M.2	-13.21	7.17	12.49	-11.29

M.3	-6.94	-6.34	13.47	-0.45
M.4	18.73	51.95	25.65	-9.48
M.5	-17.12	-13.39	-35.00	44.46
M.6	-13.94	-45.46	-9.77	-52.02
M.7	6.98	-9.75	-4.18	-5.66
M.13	10.55	12.26	-5.03	5.43
M.15	-9.08	-8.91	9.32	21.28
M.18	-3.50	-4.82	2.14	4.19

En el siguiente insumo se relacionan las coordenadas de las 6 primeras observaciones, las cuales también serán graficadas en la Figura 31.7.

```
# Coordenadas de las seis primeras observaciones en cada eje canónico
round(head(Fp <- predict(dis)$x),2)
```

	LD1	LD2	LD3	LD4
1	-1.19	3.63	0.21	-0.41
2	-0.69	1.72	0.03	0.62
3	-2.64	3.04	1.36	3.00
4	-2.09	1.18	0.97	0.42
5	-1.45	2.47	1.56	-0.07
6	-1.31	2.95	2.77	-0.86

La siguiente tabla es conocida como **tabla de contingencia** la cual realiza una validación cruzada entre las especies (filas) y los grupos discriminados por el lda (columnas). De acuerdo a esta tabla solo *P. sophore* no discrimina a todos sus individuos en su grupo, debido a que hay dos peces que presentan una morfometría más similar a *P. chola*.

```
# Evaluación de desempeño del AD (método 1)
attach(peces)
group<-predict(dis,method="plug-in")$class
(tabla<-table(Especie,group))
```

Especie	group				
	P. chola	P. conchonious	P. sophore	P. ticto	S. sarana
P. chola	12	0	0	0	0
P. conchonious	0	8	0	0	0
P. sophore	2	0	4	0	0
P. ticto	0	0	0	20	0
S. sarana	0	0	0	0	24

La siguiente tabla realiza una validación en términos porcentuales, definiendo que en el caso de *P. sophore* el 64% de sus individuos discriminan correctamente en su especie. El resto de especies presentan una discriminación completa en su grupo.

```
# Porcentaje de clasificación correcta
round(diag(prop.table(tabla, 1)),2)*100
```

P. chola	P. conchonious	P. sophore	P. ticto	S. sarana
100	100	67	100	100

La siguiente validación realizada por el método de Jackknife, presenta menos resolución de discriminación que la anterior, por lo cual no será tomada en cuenta en este ejercicio.

```
# Método 2: con clasificación basada en jackknife (validación cruzada dejando uno afuera)
dis.jac <- lda(Especie ~ M.1+M.2+M.3+M.4+M.5+M.6+M.7+M.13+M.15+M.18,
               data=peces, CV=TRUE)
```

```
# Número y proporciones de clasificación correcta
clases.jac <- dis.jac$class
tabla.jac <- table(Especie, clases.jac)
tabla.jac
```

	clases.jac				
Especie	P. chola	P. conchonious	P. sophore	P. ticto	S. sarana
P. chola	10	0	2	0	0
P. conchonious	0	7	0	1	0
P. sophore	4	0	2	0	0
P. ticto	0	0	0	20	0
S. sarana	0	0	0	0	24

```
# Validación cruzada
round(diag(prop.table(tabla.jac, 1)),2)*100
```

P. chola	P. conchonious	P. sophore	P. ticto	S. sarana
83	88	33	100	100

### 31.0.5.4 Paso 3. Visualización grafica del LDA

- **\*\* 3.1 Gráfico de elipses.\*\***

A continuación se realizará el componente grafico del lda, el cual inicia con una figura que definirá unas elipses, las cuales relacionan a los individuos de cada especie y cuyo solapamiento definirá el nivel de relación entre estas.

```
# Scores o coordenadas de las observaciones en cada eje can?nico
Fp <- predict(dis)$x

# Grupos asignados por el AD
group<-predict(dis,method="plug-in")$class

# Coordenadas y grupos asignados
peces.coord=data.frame(Especie=group,Fp)
```

La Figura 31.7 demuestra que si bien de presenta una buena discriminación de las especies de peces, 4 de las 5 evaluadas presentan cierta relación, definida por el solapamiento de sus elipses.

```
# Figura del LDA
attach(peces)
x11()
scatterplot(LD2~LD1 | Especie, data=peces.coord,reg.line=FALSE,
            smooth=F, spread=F,span= 1,grid=F,
            legend=list(coords="bottom"),
            ellipse=T,font.lab=2, pch=c(15,16,17,18,19),
            col=c('#fc8d59','#e41a1c','#984ea3','#377eb8','#33a02c'),
            main="Análisis discriminante",
            font.main=2,cex.main=2,cex.lab=1.5,
            xlab="Eje1", ylab="Eje2")
```

# Análisis discriminante

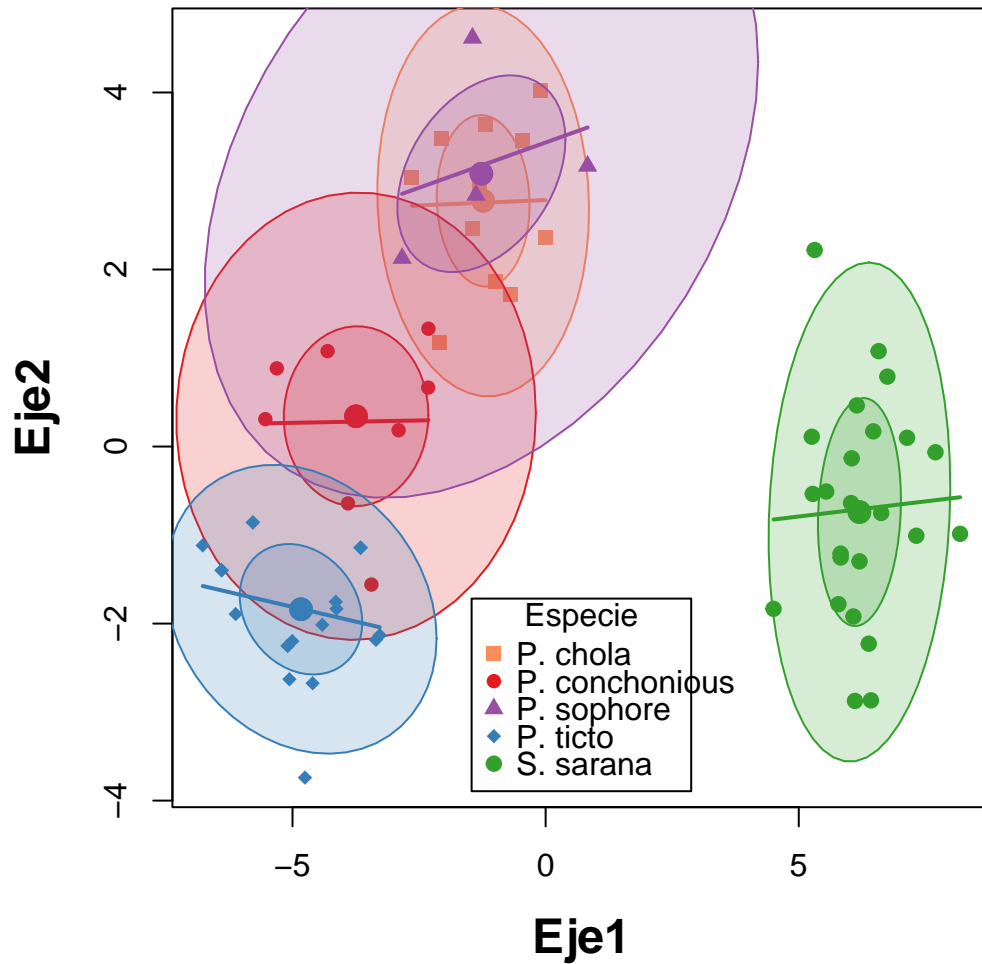


Figura 31.7: Mapa de calor que relaciona a las variables morfométricas y a las especies de peces.

## • \*\* 3.2 Gráfico del discriminante canónico - cda\*\*

El siguiente análisis permite visualizar la influencia de cada variable morfométrica en la separación de los grupos de especies y selecciona a las variables que presentan más influencia en la discriminación de los peces.

```
attach(peces)
names(peces)
```

```
[1] "Especie" "Grupo"   "M.1"     "M.2"     "M.3"     "M.4"     "M.5"
[8] "M.6"     "M.7"     "M.8"     "M.9"     "M.10"    "M.11"    "M.12"
[15] "M.13"    "M.14"    "M.15"    "M.16"    "M.17"    "M.18"    "M.19"
[22] "M.20"    "M.21"    "M.22"    "M.23"    "M.24"    "M.25"    "M.26"
[29] "M.27"    "M.28"    "M.29"
```

Lo primero que se realiza es el modelo lineal `mod` indicando las variables y los grupos a discriminar.

```
# Modelo Lineal multivariado con las variables morfométricas de peces
mod <- lm(cbind(M.1,M.2,M.3,M.4,M.5,M.6,M.7,M.13,M.15,M.18) ~ Especie,peces)

# Resumen del modelo multivariado
# summary(mod)
```

Posteriormente realiza el discriminante canónico `can` que permite realizar la discriminación de los grupos en los ejes canónicos.

```
# Análisis discriminante canónico - ADC
can <- candisc(mod, term="Especie",data=peces,ndim=1)
```

A continuación se presenta la Figura 31.8 que define a la discriminación de las especies con un solo eje canónico el cual explica el 81.8% de la variación de los datos. La orientación de los vectores (variables morfométricas), en relación a las cajas, indica su importancia para discriminar a cada especie o grupo en comparación.

```
x11()
plot(can,titles.1d = c("Puntuación canónica", "Estructura"))
```



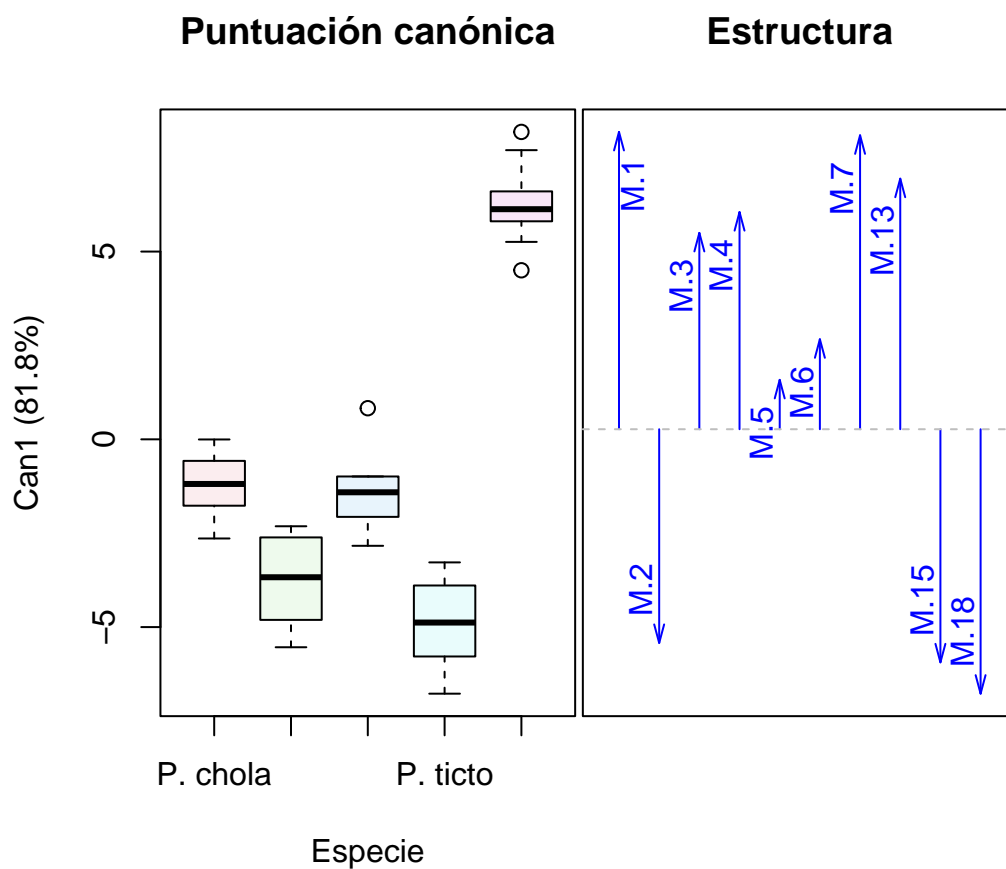


Figura 31.8: Mapa de calor que relaciona a las variables morfométricas y a las especies de peces.

En el siguiente enlace se puede obtener un caso aplicado con este tipo de análisis multivariados: [Aguirre et al. \(2015\)](#). Ver la figura 2 del manuscrito en mención.

## 32 Taller 10.1 Análisis de Varianza Multivariado - MANOVA

### Objetivo de la actividad:

La base de datos que se utilizará es la de medidas morfométricas de peces de un estudio realizado con peces de la india por **Gupta et al. (2018)** [Artículo fuente](#) en los que se validó la taxonomía de peces de la subfamilia Barbinae, utilizando 19 variables morfométricas y 19 variables merísticas, correspondientes a 5 Especie de la familia en mención.

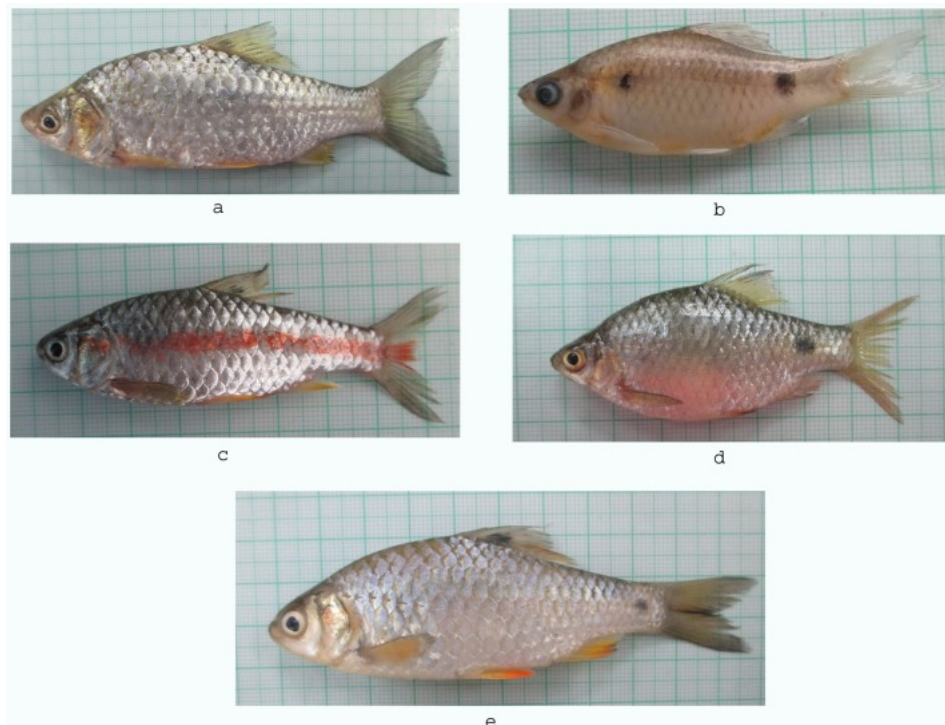


Figura 32.1: Imágen tomada de *Gupta et al. (2018)* (a) *S . Sarana* (b) *P . ticto* (c) *P . sóforo* , (d) *P . conconio* y (e) *P . chola*

El **objetivo** de este ejercicio consiste en comparar los promedios multivariados de las variables morfométricas que caracterizan a los peces de cada especie (grupos en comparación), para

conocer si los atributos morfométricos generan diferencias en cada grupo evaluado. La base de datos que se utilizará es **peces.csv**.

### 32.0.1 Referencias bibliográficas de apoyo.

[Libro: Análisis de datos ecológicos y ambientales - Rodríguez-Barrios Javier 2023](#) Ver el capítulo de MANOVA, en donde se detallan los procedimientos descritos en el presente ejercicio.

[MANOVA\(Multivariate Analysis of Variance\)](#) Este documento presenta información relevante sobre la fundamentación de los MANOVAs y su análisis en R

[How to perform the MANOVA test in R](#) Este documento brinda información sobre los supuestos del MANOVA y la forma de realizar esta prueba multivariada en R.

### 32.0.2 Cargar las librerías requeridas

```
# Librerías requeridas
library(tidyverse)
library(ggplot2)
library(reshape2)
library(ggforce)

library(vegan)           # Para el permutes en homogeneidad de covarianzas
library(mvnormtest)      # Prueba de normalidad "mshapiro.test"
source("funciones.r")    # Figuras de normalidad multivariada
library(ade4)

library(car)             # Para ejecutar el diagnóstico de independencia
library(MASS)
```

### 32.0.3 Cargar o importar la base de datos

La presente base de datos se encuentra en formato plano de **csv**, presenta una columna **Especie** que agrupa a las 5 Especie de peces, otra columna **Grupo**, que asigna un número a cada especie y posteriormente a las 19 variables morfométricas y 10 variables merísticas, de las cuales se seleccionarán las 19 morfométricas para este ejercicio M.1 a M.19.

```
# Base de datos
peces<-read.csv2("peces.csv",row.names=1)
names(peces)
```

```
[1] "Especie" "Grupo"  "M.1"    "M.2"    "M.3"    "M.4"    "M.5"
[8] "M.6"     "M.7"     "M.8"     "M.9"     "M.10"    "M.11"    "M.12"
[15] "M.13"    "M.14"    "M.15"    "M.16"    "M.17"    "M.18"    "M.19"
[22] "M.20"    "M.21"    "M.22"    "M.23"    "M.24"    "M.25"    "M.26"
[29] "M.27"    "M.28"    "M.29"
```

### 32.0.4 Exploración de los datos

Para este ejemplo se utilizarán figuras que relacionan parejas de variables y figuras de cajas que permitan visualizar diferencias entre las Especie de peces de acuerdo a su morfometría.

\*Para facilidad del ejercicio se seleccionarán algunas variables morfométricas - `peces1`, debido a que son las que presentan mejores patrones lineales.

```
# Elipses con colores con variables morfométricas
peces1 <- peces[,c(3:9,15,17,20)]
M <- cor(peces1) # Matriz de Correlación (M)
```

La Figura 32.2 permite visualizar la resolución de cada variable para diferenciar o discriminar a las diferentes especies de peces. Esta figura sirve de insumo para descartar aquellas variables con poco potencial de discriminación de las especies.

```
# Figuras multivariadas de Cajas y bigotes
library(reshape)

x11()
ggplot(melt(peces[,c(1,3:9,15,17,20)]), aes(x=variable, y=value)) +
  geom_boxplot(aes(fill=Especie)) +
  scale_fill_manual(values = c('#fc8d59','#ffffbf','#99d594','#377eb8','#33a02c')) +
  labs(x="",y="Morfometría") +
  facet_wrap(~ variable,scales="free") +
  theme_bw()
```

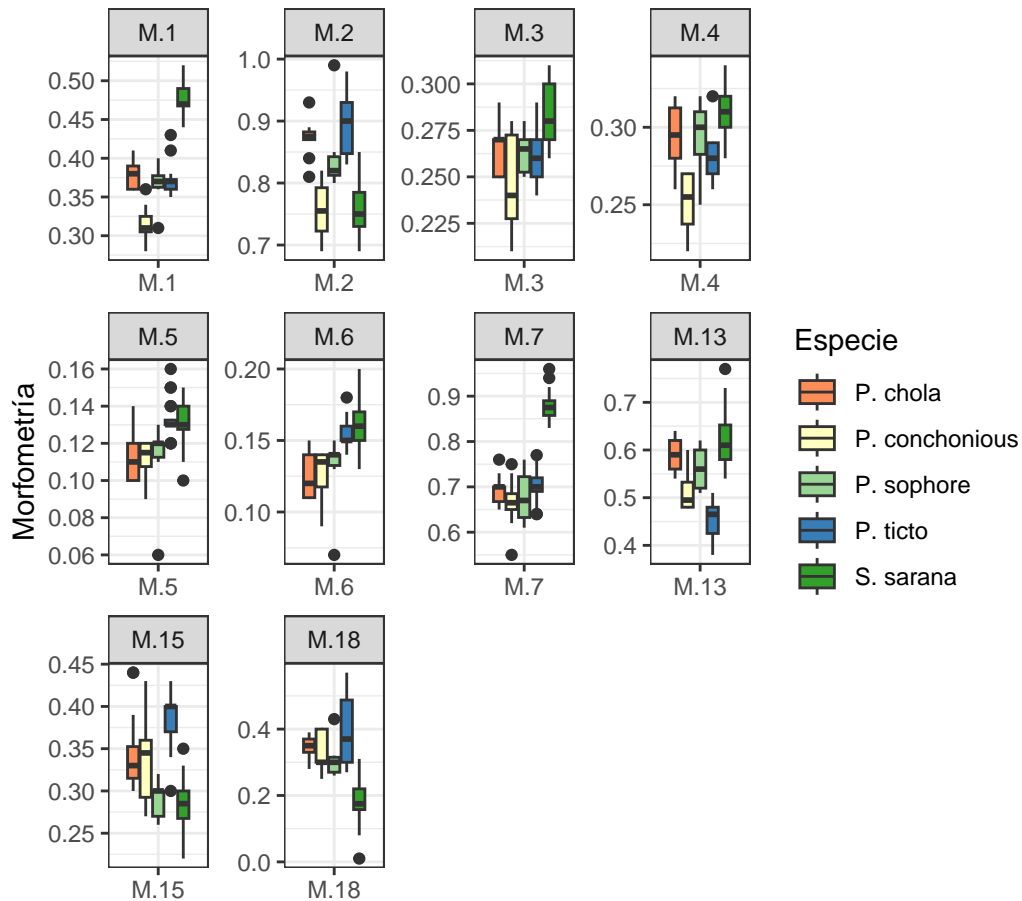


Figura 32.2: Variación en la morfometría de los peces, de acuerdo a cada una de las variables seleccionadas.

### 32.0.5 Cuatro pasos para la realización del MANOVA

#### 32.0.5.1 Paso 1. Pruebas de supuestos

Para que el análisis de varianza multivariado - manova sea considerado como un modelo lineal, debe cumplir con los supuestos de normalidad multivariada, de homogeneidad de covarianzas y de independencia. Para el caso del presente ejercicio, los dos primeros supuestos no alcanzan a cumplirse con los diagnósticos utilizados (valor  $p < 0.05$ ), motivo por el cual, el **manova** de este ejercicio será tomado como una técnica de exploración multivariada para evaluar las diferencias en las 5 especies, basado en las 10 variables morfométricas seleccionadas. En el siguiente ejercicio se realizarán análisis de varianza no paramétricos **pemanovas**, que permiten

probar hipótesis sin el cumplimiento de los dos primeros supuestos, por lo cual serán los diseños multivariados más apropiados para esta base de datos.

#### 32.0.5.1.1 1.1 Supuesto de normalidad

El supuesto de normalidad multivariada será evaluado con el paquete `mvnrmtest`, el cual utiliza el estadístico de Shapiro Wilks Multivariado. Para ello se realizará esta prueba en cada uno de los grupos o especies en comparación. **NOTA:** Este supuesto también será evaluado con los **residuales del manova**, posterior a su ejecución.

```
# Diagnóstico de normalidad por cada tipo de Especie
library(mvnrmtest)
```

Los siguientes generan los dataframes de cada especie con las 10 variables seleccionadas, convirtiéndola además en formato matricial.

```
# Dataframe por cada especie

# datos de P. chola.
P.chola <- peces %>%
  filter(Especie == "P. chola") %>%
  subset(select = c("M.1", "M.2", "M.3", "M.4", "M.5", "M.6", "M.7", "M.13", "M.15", "M.16", "M.17", "M.18", "M.19", "M.20", "M.21", "M.22", "M.23", "M.24", "M.25", "M.26", "M.27", "M.28", "M.29", "M.30", "M.31", "M.32", "M.33", "M.34", "M.35", "M.36", "M.37", "M.38", "M.39", "M.40", "M.41", "M.42", "M.43", "M.44", "M.45", "M.46", "M.47", "M.48", "M.49", "M.50", "M.51", "M.52", "M.53", "M.54", "M.55", "M.56", "M.57", "M.58", "M.59", "M.60", "M.61", "M.62", "M.63", "M.64", "M.65", "M.66", "M.67", "M.68", "M.69", "M.70", "M.71", "M.72", "M.73", "M.74", "M.75", "M.76", "M.77", "M.78", "M.79", "M.80", "M.81", "M.82", "M.83", "M.84", "M.85", "M.86", "M.87", "M.88", "M.89", "M.90", "M.91", "M.92", "M.93", "M.94", "M.95", "M.96", "M.97", "M.98", "M.99", "M.100"))

# datos de P. conchionious.
P.concho <- peces %>%
  filter(Especie == "P. conchionious") %>%
  subset(select = c("M.1", "M.2", "M.3", "M.4", "M.5", "M.6", "M.7", "M.13", "M.15", "M.16", "M.17", "M.18", "M.19", "M.20", "M.21", "M.22", "M.23", "M.24", "M.25", "M.26", "M.27", "M.28", "M.29", "M.30", "M.31", "M.32", "M.33", "M.34", "M.35", "M.36", "M.37", "M.38", "M.39", "M.40", "M.41", "M.42", "M.43", "M.44", "M.45", "M.46", "M.47", "M.48", "M.49", "M.50", "M.51", "M.52", "M.53", "M.54", "M.55", "M.56", "M.57", "M.58", "M.59", "M.60", "M.61", "M.62", "M.63", "M.64", "M.65", "M.66", "M.67", "M.68", "M.69", "M.70", "M.71", "M.72", "M.73", "M.74", "M.75", "M.76", "M.77", "M.78", "M.79", "M.80", "M.81", "M.82", "M.83", "M.84", "M.85", "M.86", "M.87", "M.88", "M.89", "M.90", "M.91", "M.92", "M.93", "M.94", "M.95", "M.96", "M.97", "M.98", "M.99", "M.100"))

# datos de P. sophore.
P.sophore <- peces %>%
  filter(Especie == "P. sophore") %>%
  subset(select = c("M.1", "M.2", "M.3", "M.4", "M.5", "M.6", "M.7", "M.13", "M.15", "M.16", "M.17", "M.18", "M.19", "M.20", "M.21", "M.22", "M.23", "M.24", "M.25", "M.26", "M.27", "M.28", "M.29", "M.30", "M.31", "M.32", "M.33", "M.34", "M.35", "M.36", "M.37", "M.38", "M.39", "M.40", "M.41", "M.42", "M.43", "M.44", "M.45", "M.46", "M.47", "M.48", "M.49", "M.50", "M.51", "M.52", "M.53", "M.54", "M.55", "M.56", "M.57", "M.58", "M.59", "M.60", "M.61", "M.62", "M.63", "M.64", "M.65", "M.66", "M.67", "M.68", "M.69", "M.70", "M.71", "M.72", "M.73", "M.74", "M.75", "M.76", "M.77", "M.78", "M.79", "M.80", "M.81", "M.82", "M.83", "M.84", "M.85", "M.86", "M.87", "M.88", "M.89", "M.90", "M.91", "M.92", "M.93", "M.94", "M.95", "M.96", "M.97", "M.98", "M.99", "M.100"))

# datos de P. ticto.
P.ticto <- peces %>%
  filter(Especie == "P. ticto") %>%
  subset(select = c("M.1", "M.2", "M.3", "M.4", "M.5", "M.6", "M.7", "M.13", "M.15", "M.16", "M.17", "M.18", "M.19", "M.20", "M.21", "M.22", "M.23", "M.24", "M.25", "M.26", "M.27", "M.28", "M.29", "M.30", "M.31", "M.32", "M.33", "M.34", "M.35", "M.36", "M.37", "M.38", "M.39", "M.40", "M.41", "M.42", "M.43", "M.44", "M.45", "M.46", "M.47", "M.48", "M.49", "M.50", "M.51", "M.52", "M.53", "M.54", "M.55", "M.56", "M.57", "M.58", "M.59", "M.60", "M.61", "M.62", "M.63", "M.64", "M.65", "M.66", "M.67", "M.68", "M.69", "M.70", "M.71", "M.72", "M.73", "M.74", "M.75", "M.76", "M.77", "M.78", "M.79", "M.80", "M.81", "M.82", "M.83", "M.84", "M.85", "M.86", "M.87", "M.88", "M.89", "M.90", "M.91", "M.92", "M.93", "M.94", "M.95", "M.96", "M.97", "M.98", "M.99", "M.100"))

# datos de P. ticto.
S.sarana <- peces %>%
  filter(Especie == "S. sarana") %>%
  subset(select = c("M.1", "M.2", "M.3", "M.4", "M.5", "M.6", "M.7", "M.13", "M.15", "M.16", "M.17", "M.18", "M.19", "M.20", "M.21", "M.22", "M.23", "M.24", "M.25", "M.26", "M.27", "M.28", "M.29", "M.30", "M.31", "M.32", "M.33", "M.34", "M.35", "M.36", "M.37", "M.38", "M.39", "M.40", "M.41", "M.42", "M.43", "M.44", "M.45", "M.46", "M.47", "M.48", "M.49", "M.50", "M.51", "M.52", "M.53", "M.54", "M.55", "M.56", "M.57", "M.58", "M.59", "M.60", "M.61", "M.62", "M.63", "M.64", "M.65", "M.66", "M.67", "M.68", "M.69", "M.70", "M.71", "M.72", "M.73", "M.74", "M.75", "M.76", "M.77", "M.78", "M.79", "M.80", "M.81", "M.82", "M.83", "M.84", "M.85", "M.86", "M.87", "M.88", "M.89", "M.90", "M.91", "M.92", "M.93", "M.94", "M.95", "M.96", "M.97", "M.98", "M.99", "M.100"))
```

Vale la pena resaltar que los datos de las especies *P. chola* y *P. sophore*, son singulares, por lo cual no puede calcularse su supuesto de normalidad multivariada. Con el objeto de continuar en el ejercicio, las matrices que representan a las especies en mención, serán desactivadas con #.

```
# Prueba de normalidad para cada especie
library(mvnormtest)

# norm1 <- mshapiro.test(t(P.chola)) # Matriz singular
norm2 <- mshapiro.test(t(P.concho))
# norm3 <- mshapiro.test(t(P.sophore)) # Matriz singular
norm4 <- mshapiro.test(t(P.ticto))
norm5 <- mshapiro.test(t(S.sarana))
```

A continuación se resume el resultado de los tres diagnósticos de normalidad multivariada realizados. Vale la pena mencionar que ninguna especie cumple con dicho supuesto estadístico (valores  $p < 0.05$ ), aunque existe la posibilidad de probar con alguna transformación.

```
# Resumen de el diagnóstico de normalidad
(normalidad = data.frame(Norm.P.concho = norm2$p.value,
                        Norm.ticto    = norm4$p.value,
                        Norm.sarana   = norm5$p.value))
```

```
Norm.P.concho  Norm.ticto  Norm.sarana
1  0.001050723  2.629679e-08  5.169222e-07
```

### 32.0.5.1.2 1.2 Supuesto de homogeneidad de covarianzas

La prueba de homogeneidad de covarianza o **esfericidad**, corresponde al segundo supuesto del análisis discriminante lineal, se utilizará la función **betadisper**, la cual es complementada por dos análisis de varianza, los cuales definirán si el supuesto logra ser cumplido.

```
# Pruebas de Homogeneidad de covarianzas paquete "vegan"
library(vegan)

peces.d <- dist(peces[,c(3:9,15,17,20)]) # Matriz de distancias
peces.homoge <- betadisper(peces.d, peces$Especie) # Permutest
```

Con la siguiente **anova** se obtiene un valor  $p$  de 0.016\*, lo cual indica que no se cumple el supuesto de homogeneidad de covarianzas (valor  $p < 0.05$ ).

```
# 1) Prueba con anova permutacional
anova(peces.homoge)
```

Analysis of Variance Table

Response: Distances

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Groups	4	0.019804	0.0049509	3.281	0.01638 *
Residuals	65	0.098082	0.0015090		

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Con el `permutest` se obtiene un valor p de 0.015\*, lo cual indica que tampoco se cumple el supuesto de homogeneidad de covarianzas (valor  $p < 0.05$ ).

```
# 2) Prueba permutacional
permutest(peces.homoge) # Se cumple el supuesto de homogeneidad
```

Permutation test for homogeneity of multivariate dispersions

Permutation: free

Number of permutations: 999

Response: Distances

	Df	Sum Sq	Mean Sq	F	N.Perm	Pr(>F)
Groups	4	0.019804	0.0049509	3.281	999	0.022 *
Residuals	65	0.098082	0.0015090			

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

### 32.0.5.2 Paso 2. Análisis de Varianza Multivariado - MANOVA

El manova, por ser un modelo lineal especial, requiere que se indiquen las variables continuas (variables  $X_i$ ) y la variable cualitativa o categórica (variable  $Y_i$ ), que para este caso es la Especie.

```
# Manova (variables efecto: 10 morfológicas y la respuesta: Especie)
attach(peces)
peces.manova<-manova(cbind(M.1,M.2,M.3,M.4,M.5,M.6,M.7,M.13,M.15,M.18)~Especie)
```



A continuación se presentará la tabla del manova para las tres primeras variables morfométricas. Para visualizar todos los resultados de este insumo, es necesario colocar solo: `summary.aov(peces.manova)`. Para este caso se observa que todas las variables efecto o morfológicas, tienen un efecto muy significativo (valor p  $\ll 0.01$ ) en la diferenciación de los 5 grupos o especies de peces.

```
# respuesta de la variable M.1
summary.aov(peces.manova)$" Response M.1"
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Especie	4	0.221052	0.055263	127.36	< 2.2e-16 ***
Residuals	65	0.028205	0.000434		

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

```
# respuesta de la variable M.2
summary.aov(peces.manova)$" Response M.2"
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Especie	4	0.28343	0.070857	34.76	1.654e-15 ***
Residuals	65	0.13250	0.002038		

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

```
# respuesta de la variable M.3
summary.aov(peces.manova)$" Response M.3"
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Especie	4	0.010738	0.00268443	10.486	1.29e-06 ***
Residuals	65	0.016641	0.00025601		

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

A continuación se presentan los cuatro tipos de MANOVAS, con la diferencia de que Hotelling puede utilizarse sin el cumplimiento del supuesto de normalidad multivariada (pero deben cumplirse los otros supuestos como el de la homogeneidad y el de la independencia). Para este caso los 4 estadísticos muestran altas diferencias entre algunos de los grupos representados por las cinco especies, posiblemente por el efecto de *P. sarana*, que al ser de un género diferente, presenta marcadas diferencias morfológicas (ver Figura 32.4).

```
# Tipos de MANOVA para evaluar si hay diferencias en los promedios de cada Especie
summary(peces.manova,test="Pillai")
```

```

      Df Pillai approx F num Df den Df    Pr(>F)
Especie  4  2.513   9.9711     40   236 < 2.2e-16 ***
Residuals 65
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(peces.manova,test="Wilks")
```

```

      Df      Wilks approx F num Df den Df    Pr(>F)
Especie  4 0.0028912   19.667     40  214.2 < 2.2e-16 ***
Residuals 65
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(peces.manova,test="Hotelling")
```

```

      Df Hotelling-Lawley approx F num Df den Df    Pr(>F)
Especie  4          29.052   39.584     40   218 < 2.2e-16 ***
Residuals 65
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(peces.manova,test="Roy")
```

```

      Df      Roy approx F num Df den Df    Pr(>F)
Especie  4 23.757   140.17     10    59 < 2.2e-16 ***
Residuals 65
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

### 32.0.5.3 Paso 3. Supuestos del MANOVA

SON tres los supuestos que a continuación se probarán: (1) Normalidad en los residuales, (2) homogeneidad de las covarianzas y (3) Independencia en los datos. Vale la pena resaltar la importancia del cumplimiento de la independencia, debido a que presenta un efecto sobre diseños que requieran ser aleatorios. La homogeneidad fue diagnosticada al inicio de este ejercicio no se cumple.

### 32.0.5.4 3.1. Supuesto de normalidad de los residuales del MANOVA

A continuación se probará el supuesto de (1) normalidad en los residuales del manova, de forma numérica y gráfica. El estadístico de Shapiro Wilks Multivariado es el que se utiliza, demostrando que los residuales están muy alejados del patrón normal (valor  $p \ll 0.01$  o  $p = 3.043e-07$ ).

```
# 1) Prueba de multinormalidad de los residuales del manova (mshapiro.test)
library(mvnormtest)
x <- as.matrix(t(residuals(peces.manova)))
mshapiro.test(x)
```

Shapiro-Wilk normality test

```
data: Z
W = 0.83893, p-value = 3.043e-07
```

```
# No se cumple este supuesto
```

A continuación se hará uso del código fuente “funciones.r” el cual presenta los comandos requeridos para la figura que diagnostica la normalidad multivariada (Figura 32.3 qqplot).

```
# Figura de multinormalidad
# Funciones para la figura
source("funciones.r")
```

En la Figura 32.3 se observa que algunos residuales (puntos circulares) se alejan considerablemente del patrón de normalidad, definido por la recta roja.

```
# Grafica QQ-PLot para visualizar la normalidad
x <- as.matrix(residuals(peces.manova))
# centroide
```

```

center <- colMeans(x)
n <- nrow(x); p <- ncol(x); cov <- cov(x);
d <- mahalanobis(x,center,cov)

x11()
qqplot(qchisq(ppoints(n),df=p),d,
       main="Normalidad multivariada",
       ylab="Cuantil Chi-Cuadrado", xlab= "Distancia Mahalanobish")
abline(a=0,b=1,col=2)

```

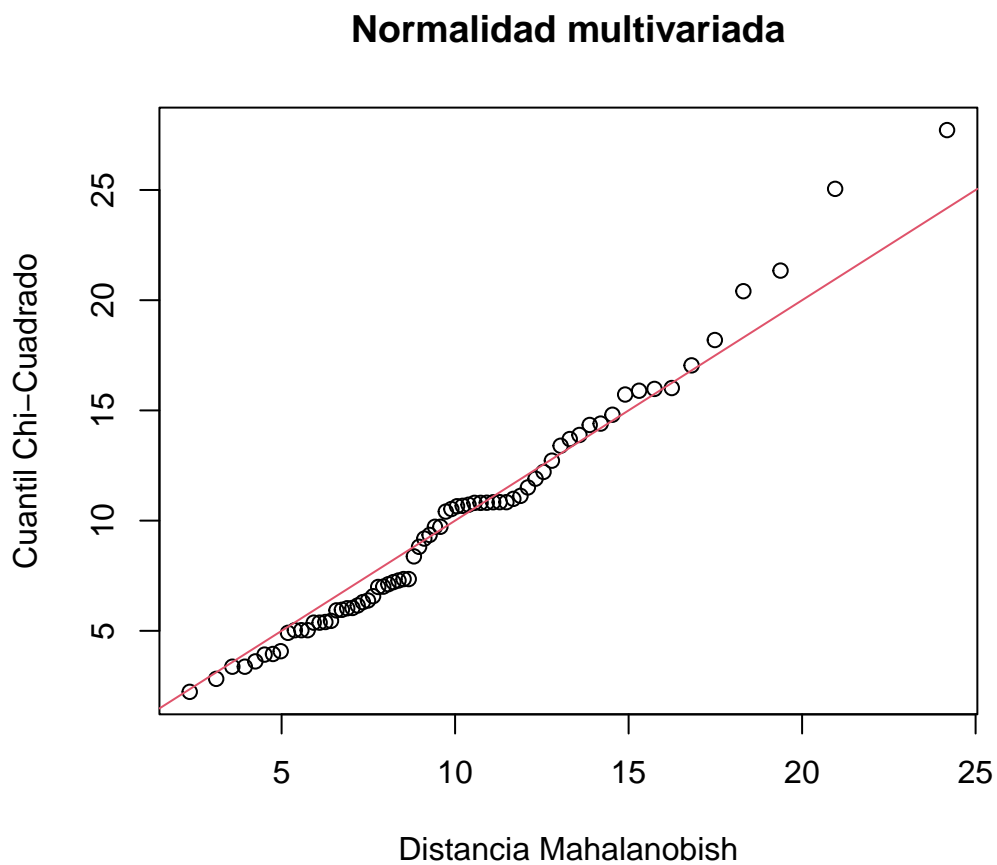


Figura 32.3: Figura qqplot entre los residuales observados (Distancia Mahalanobish) y los cuantiles chi cuadrado (estimados).

### 32.0.5.5 3.2. Supuesto de independencia

Se utilizará el estadístico Durbin Watson (DW) el cual demuestra que se cumple la independencia (valor  $p > 0.05$ ).

```
# Prueba de Independencia - Estadístico Durbin Watson
attach(peces)
modelo<-lm(M.1+M.2+M.3+M.4+M.5+M.6+M.7+M.13+M.15+M.18~Especie)
durbinWatsonTest(modelo)
```

```
lag Autocorrelation D-W Statistic p-value
1      0.08444408      1.80425    0.21
Alternative hypothesis: rho != 0
```

### 32.0.5.6 Paso 4. Prueba a posteriori del MANOVA

A continuación, se realizará una figura del análisis discriminante - *lda*, que permitirá generar definir al nivel de discriminación de cada grupo o especie de pez. Se presentan algunas opciones gráficas con el procedimiento general y con el análisis discriminante canónico (dca)

```
# Cálculo del LDA
names(peces)
```

```
[1] "Especie" "Grupo"   "M.1"     "M.2"     "M.3"     "M.4"     "M.5"
[8] "M.6"     "M.7"     "M.8"     "M.9"     "M.10"    "M.11"    "M.12"
[15] "M.13"    "M.14"    "M.15"    "M.16"    "M.17"    "M.18"    "M.19"
[22] "M.20"    "M.21"    "M.22"    "M.23"    "M.24"    "M.25"    "M.26"
[29] "M.27"    "M.28"    "M.29"
```

```
dis<-lda (Especie ~ M.1+M.2+M.3+M.4+M.5+M.6+M.7+M.13+M.15+M.18,
          data = peces)
```

A continuación se realizará el componente gráfico del *lda*, el cual inicia con una figura que definirá unas elipses, las cuales relacionan a los individuos de cada especie y cuyo solapamiento definirá el nivel de relación entre estas.

```
# Scores o coordenadas de las observaciones en cada eje canónico
Fp <- predict(dis)$x
```

```
# Grupos asignados por el AD
group<-predict(dis,method="plug-in")$class
```

```
# Coordinadas y grupos asignados
peces.coord=data.frame(Especie=group,Fp)
```

La Figura 32.4 demuestra que si bien de presenta una buena discriminación de las especies de peces, 4 de las 5 evaluadas presentan cierta relación, definida por el solapamiento de sus elipses.

```
# Figura del LDA
attach(peces)
x11()
scatterplot(LD2~LD1 | Especie, data=peces.coord,reg.line=FALSE,
            smooth=F, spread=F,span= 1,grid=F,
            legend=list(coords="bottom"),
            ellipse=T,font.lab=2, pch=c(15,16,17,18,19),
            col=c('#fc8d59','#e41a1c','#984ea3','#377eb8','#33a02c'),
            main="Análisis discriminante",
            font.main=2,cex.main=2,cex.lab=1.5,
            xlab="Eje1", ylab="Eje2")
```

# Análisis discriminante

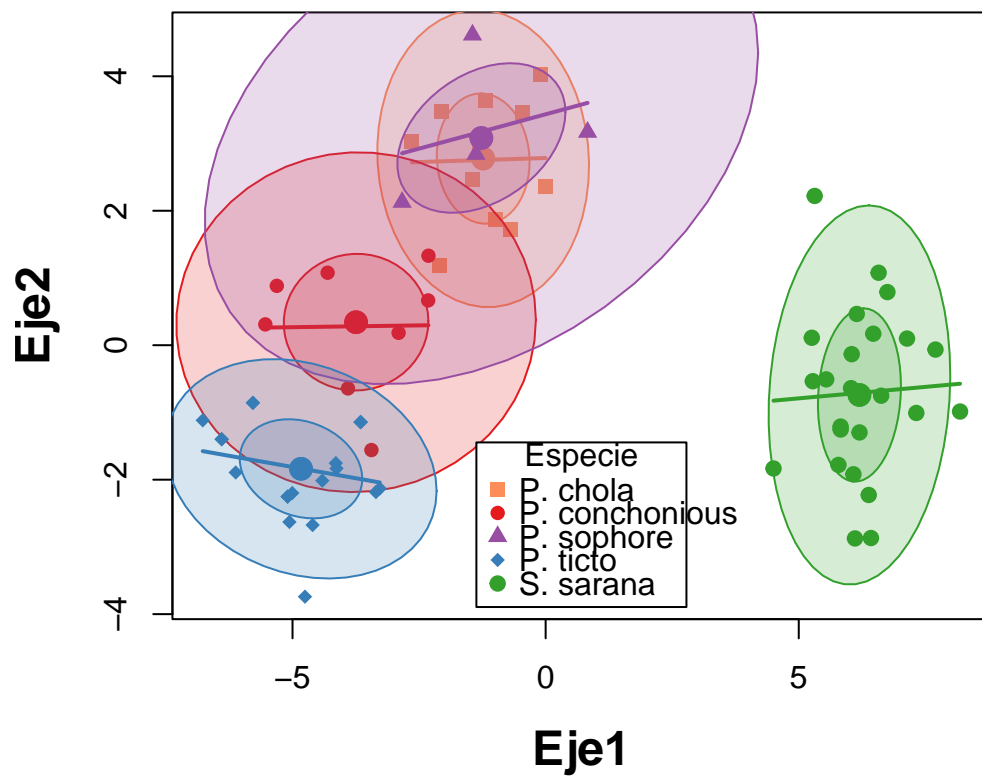


Figura 32.4: Mapa de calor que relaciona a las variables morfométricas y a las especies de peces.

## Referencias

Knuth, Donald E. 1984. «Literate Programming». *Comput. J.* 27 (2): 97-111. <https://doi.org/10.1093/comjnl/27.2.97>.