



# Introducción a R y RStudio

Francisco Charte



# En esta sesión nos ocuparemos de:

- R y RStudio
  - ¿Por qué aprender R?
  - Herramientas de trabajo
  - Ejecución de tareas habituales
- Introducción a R
  - Tipos de datos fundamentales en R
  - Cómo cargar conjuntos de datos
  - Fundamentos de análisis exploratorio de datos



# R y RStudio

¿Por qué aprender R?

# ¿Por qué aprender R?

- Es el lenguaje más usado para análisis de datos
- Conocimiento de R altamente valorado

Fuente KDnuggets

**R**, 46.9% share ( 38.5% in 2014)  
**RapidMiner**, 31.5% ( 44.2% in 2014)  
**SQL**, 30.9% ( 25.3% in 2014)  
**Python**, 30.3% ( 19.5% in 2014)  
**Excel**, 22.9% ( 25.8% in 2014)  
**KNIME**, 20.0% ( 15.0% in 2014)  
**Hadoop**, 18.4% ( 12.7% in 2014)  
**Tableau**, 12.4% ( 9.1% in 2014)  
**SAS**, 11.3 (10.9% in 2014)  
**Spark**, 11.3% ( 2.6% in 2014)

Fuente Dice Tech Salary Survey

AVERAGE SALARY FOR <b>High Paying Skills and Experience</b>		
SKILL	2013	YR/YR CHANGE
<b>R</b>	\$ 115,531	n/a
<b>NoSQL</b>	\$ 114,796	1.6%
<b>MapReduce</b>	\$ 114,396	n/a
<b>PMBok</b>	\$ 112,382	1.3%
<b>Cassandra</b>	\$ 112,382	n/a
<b>Omnigraffle</b>	\$ 111,039	0.3%
<b>Pig</b>	\$ 109,561	n/a
<b>SOA</b> (Service Oriented Architecture)	\$ 108,997	-0.5%
<b>Hadoop</b>	\$ 108,669	-5.6%
<b>Mongo DB</b>	\$ 107,825	-0.4%



# Además ...

- R es *Open Source* (multiplataforma, libre, abierto)
- Gran número de paquetes disponibles (CRAN/GitHub)
- Extensa comunidad de usuarios
- Ciclo completo de trabajo:
  - Implementación de algoritmos
  - Preparación de datos
  - Análisis de resultados
  - Generación de documentación



# R y RStudio

Herramientas de trabajo

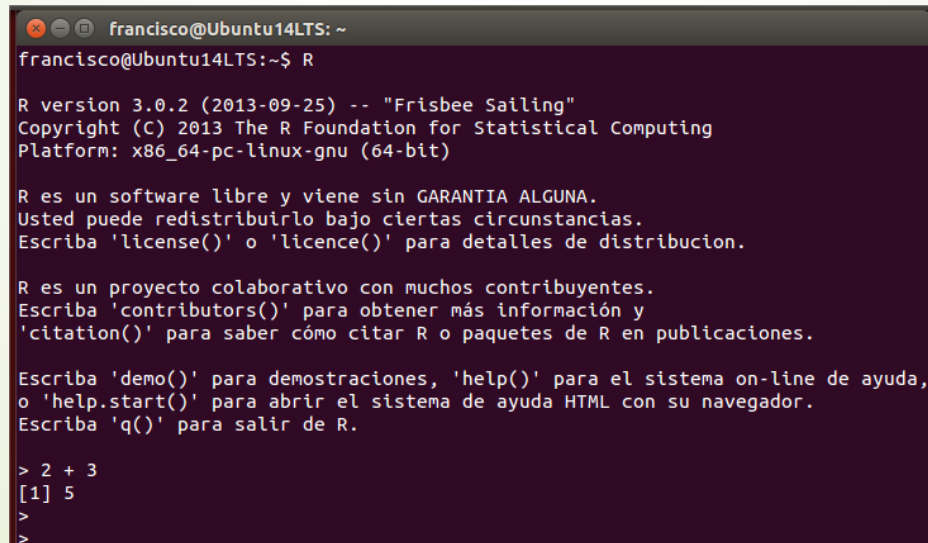
# Herramientas de trabajo - R

- Binarios disponibles para Linux, OS X y Windows
- Descarga desde <http://www.r-project.org/>
- Disponible en repositorios Linux

```
francisco@Ubuntu14LTS: ~  
francisco@Ubuntu14LTS:~$  
francisco@Ubuntu14LTS:~$ sudo apt-get install r-base r-base-dev
```

# Herramientas de trabajo - R

- Trabajo interactivo mediante línea de comandos



```
francisco@Ubuntu14LTS: ~  
francisco@Ubuntu14LTS:~$ R  
  
R version 3.0.2 (2013-09-25) -- "Frisbee Sailing"  
Copyright (C) 2013 The R Foundation for Statistical Computing  
Platform: x86_64-pc-linux-gnu (64-bit)  
  
R es un software libre y viene sin GARANTIA ALGUNA.  
Usted puede redistribuirlo bajo ciertas circunstancias.  
Escriba 'license()' o 'licence()' para detalles de distribucion.  
  
R es un proyecto colaborativo con muchos contribuyentes.  
Escriba 'contributors()' para obtener más información y  
'citation()' para saber cómo citar R o paquetes de R en publicaciones.  
  
Escriba 'demo()' para demostraciones, 'help()' para el sistema on-line de ayuda,  
o 'help.start()' para abrir el sistema de ayuda HTML con su navegador.  
Escriba 'q()' para salir de R.  
  
> 2 + 3  
[1] 5  
>  
>
```

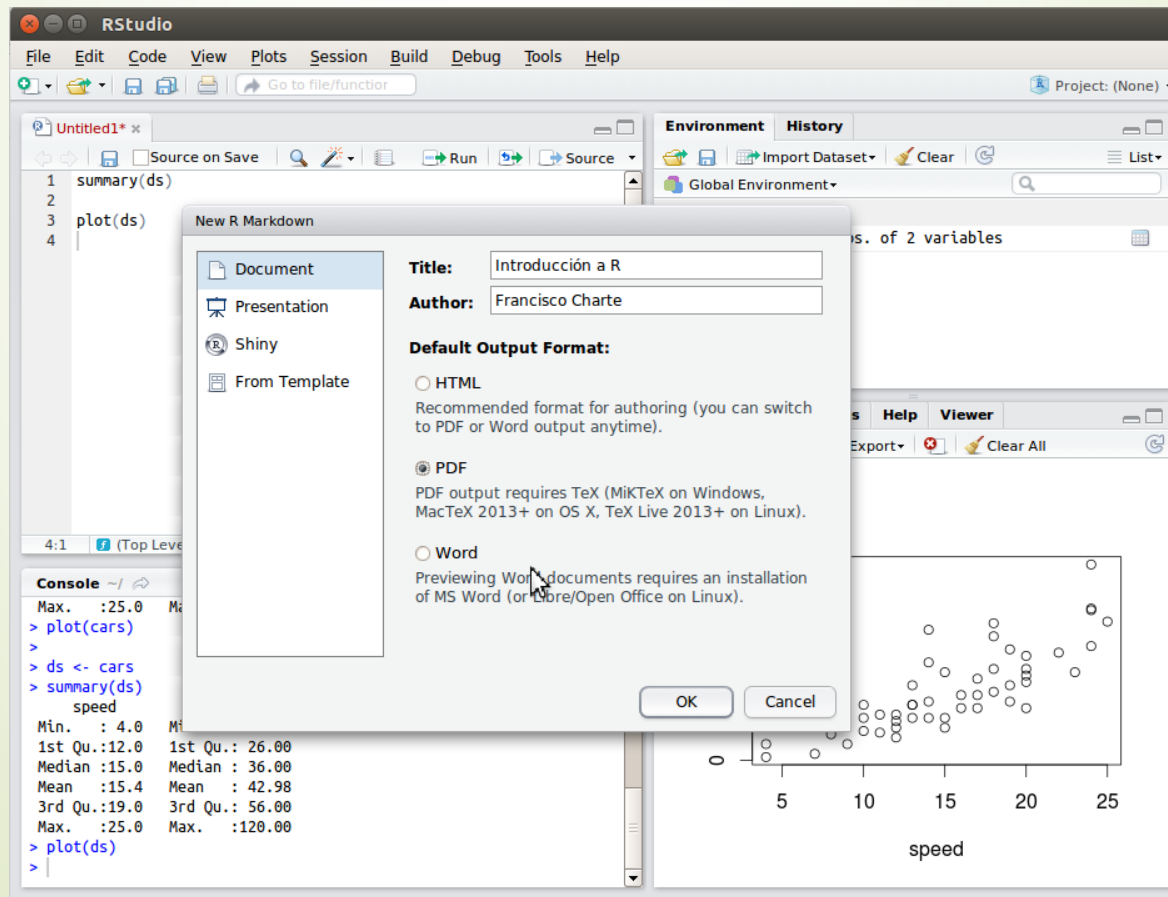


# Herramientas de trabajo - RStudio

- Binarios disponibles para Linux, OS X y Windows
- Descarga desde <http://www.rstudio.com/>
- Licencia Open Source y comercial
- IDE estándar para trabajar con R
- **Será la herramienta que usemos en el curso**

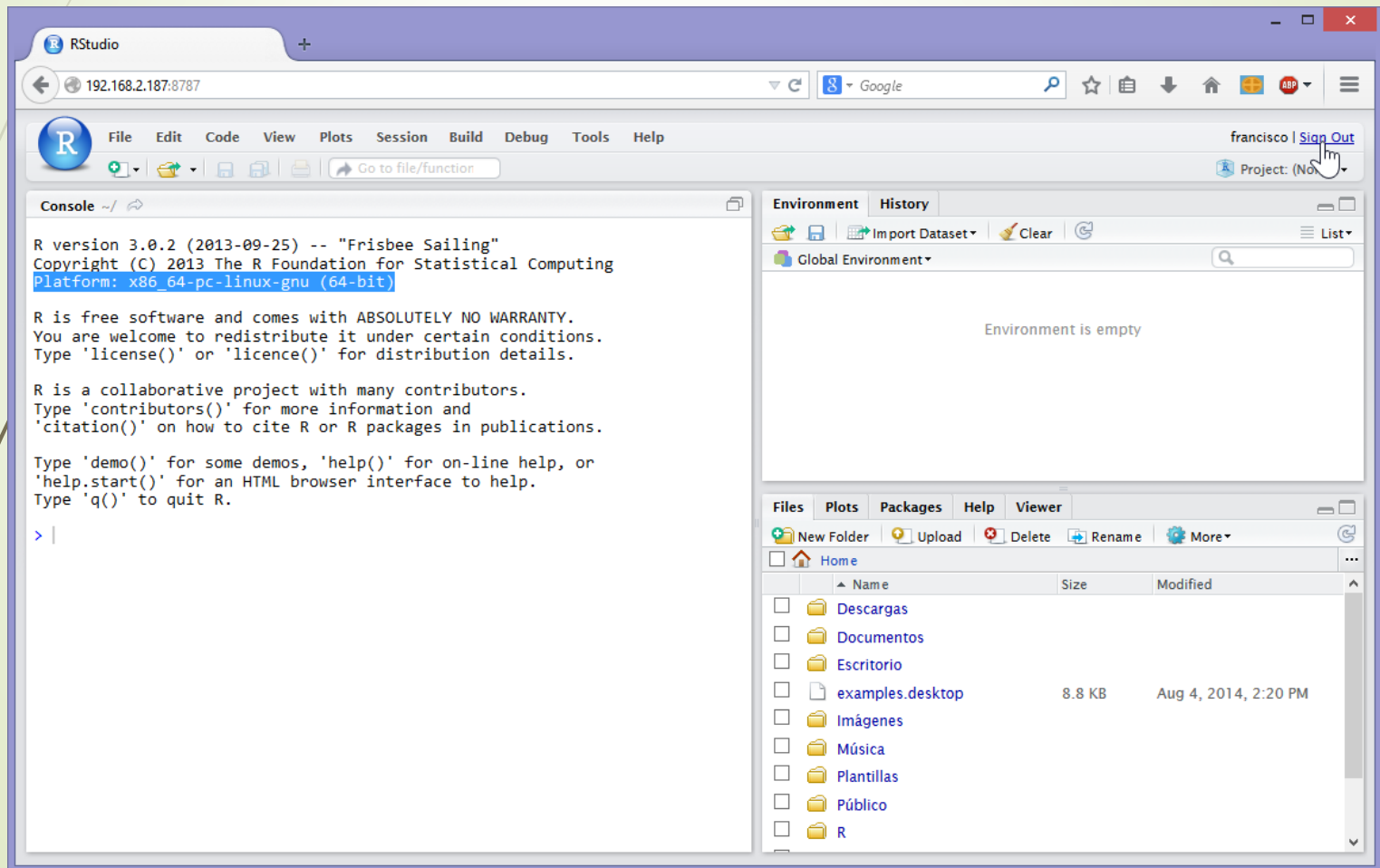
# Herramientas de trabajo - RStudio

- IDE completo para trabajar con R
- Gestión de proyectos y paquetes
- Editor de scripts
- Acceso a objetos del entorno
- Visualización de gráficas y ayuda
- Consola de R integrada



# Herramientas de trabajo - RStudio

- Instalable en un servidor web
- Accesible desde el navegador
- Multiusuario
- Idéntica interfaz de RStudio





# R y RStudio

Ejecución de tareas habituales

# Ejecución de tareas habituales

## ► tareasHabituales.R

- Acceso a la documentación
  - `help('source')`, `vignette('grid')`, `demo('image')`
- Ruta de trabajo
  - `getwd()`, `setwd()`
- Espacio de trabajo
  - `save()`, `save.image()`, `load()`
- Instalación de paquetes
  - `install.packages()`, `library()`



# Introducción a R

Tipos de datos fundamentales en R

# Tipos de datos simples

► tiposDatos.R

## ► numeric

- Enteros :: 1024, -3
- Punto flotante :: 3.1415927
- Notación exponencial :: 3.85e6
- Otros :: Inf, NaN

► integer :: as.integer(numeric)

► complex :: 1+2i

► character :: 'R', "Hola"

► logical :: TRUE, FALSE, NA

# Uso de variables

## ►tiposDatos.R

### ► Asignación

► `a = 1024` | `a <- 1024` | `1024 -> a`

### ► Obtención de clase y tipo

► `class(a)` # numeric | `typeof(a)` # double

### ► Comprobación de tipo

► `is.numeric(a)`, `is.character(a)`,  
`is.integer(a)`, `is.infinite(a)`, `is.na(a)`

### ► Objetos en el espacio de trabajo

► `ls()`, `rm(var)`, `str(var)`,  
`save(var,file = arch)`, `save.image()`, `load()`



# Vectores

## ►tiposDatos.R

### ► Definición

- `diasMes <- c(31,29,31,30,31,30,31,31,30,31,30,31)`
- `dias <- c('Lun','Mar','Mié','Jue','Vie','Sáb','Dom')`
- `quincena <- 16:30`
- `semanas <- seq(1, 365, 7)`
- `rep(T, 5)`

### ► Obtención número de elementos

- `length(dias)`

### ► Acceso a elementos

- `dias[2]`      # Segundo elemento del vector
- `dias[-2]`    # Todos los elementos menos el segundo
- `días[c(3, 7)]` # Elementos 3 y 7

# Matrices

► tiposDatos.R

## ► Definición

► `mes <- matrix(1:35, nrow = 5)`

► `mes <- matrix(1:35, ncol = 7, byrow = T)`

## ► Obtención número de elementos

► `length(mes)      |      nrow(mes)      |      ncol(mes)`

## ► Acceso a elementos

► `mes[2, ]    # 2ª fila completa`

► `mes[ ,2]    # 2ª columna completa`

► `mes[2, 5]   # 5ª columna de la segunda fila`

► `fix(mes)    # Editar elementos en la matriz`

# Factors

► tiposDatos.R

## ► Definición

- `herramientas <- factor('Consola', 'RStudio')`
- `fdias <- factor(días)`
- `tam <- ordered(c('Ligero', 'Medio', 'Pesado'))`

## ► Obtención niveles

- `nlevels(fdias)`
- `levels(días)`

## ► Relación de orden (factors ordenados)

- `tam[2] < tam[1]` `# FALSE`

# Data Frames

►tiposDatosII.R

## ► Definición

- `df <- data.frame(vect1, ..., vectN)`
- `df <- data.frame(matrix)`
- `df <- data.frame(col1 = tipo(N), ...,  
colN = tipo(N))`

## ► Ejemplo

- `df <- data.frame(Dia = fdias,  
Estimado = rep(c(T, F), 7),  
Lectura = rnorm(14, 5))`

## ► Obtención número de elementos

- `nrow(mes)`
- `ncol(mes)`

# Data Frames

## ►tiposDatosII.R

### ► Selección y proyección de datos

- `df[5, ]` # 5ª fila
- `df[, 3]` # 3ª columna
- `df[c(-3,-6), ]` # Menos 3ª y 6ª fila
- `df$Lectura` # 3ª columna
- `df$Lectura[5]` # 5ª fila de 3ª col.
- `df[, c('Dia', 'Lectura')]` # Columnas 1 y 3
- `df[df$Estimado == F, ]` # Filas condición
- # Selección de filas y columnas  
`df[df$Estimado == F & df$Lectura > 3,  
c('Dia', 'Lectura')]`



# Introducción a R

Cómo cargar conjuntos de datos

# Cargar datos CSV

► `cargaDatos.R`

- ```
datosCSV <- read.table(  
  file      = "miArchivo.csv",  
  header    = T,  
  sep       = ",",  
  dec       = ".",  
  quote     = "\"")
```
- ```
read.csv("miArchivo.csv") # sep="," , dec="."
```
- ```
read.csv2("miArchivo.csv") # sep=";" , dec=","
```
- ```
read.delim("miarchivo.txt") # sep = "\t"
```

# Cargar datos Excel

► `cargaDatos.R`

## ► Múltiples posibilidades

- Exportar desde Excel a CSV
- Copiar datos al portapapeles
- Leer archivo Excel desde R

## ► Paquetes R para trabajar con archivos Excel

### ► XLConnect

► `datos <- readworksheetFromFile('archivo.xls', sheet=n)`

### ► xlsx

► `datos <- read.xlsx('archivo.xlsx', sheetName = n, rango)`

## ► `vignette(paquete)` # Abrir el manual asociado



# Cargar datos ARFF (Weka)

► `cargaDatos.R`

## ► Paquete `foreign`

- Funciones para leer múltiples formatos de archivo
- `read.arff('dataset.arff')`

## ► Paquete `RWeka`

- Interfaz completa entre R y Weka
  - Leer y escribir archivos ARFF
  - Acceso a algoritmos de clasificación, agrupamiento, etc.
- `read.arff('dataset.arff')`

# Datasets integrados

► `cargaDatos.R`

► Lista de datasets

► `data()`

Data sets in package 'datasets':

AirPassengers  
BJsales  
BJsales.lead (BJsales)  
BOD  
CO2  
ChickWeight  
DNase  
EuStockMarkets  
Formaldehyde  
HairEyeColor  
Harman23.cor  
Harman74.cor  
Indometh  
InsectSprays  
JohnsonJohnson  
LakeHuron  
LifeCycleSavings  
Loblolly  
Nile  
Orange  
OrchardSprays  
PlantGrowth  
Puromycin  
Seatbelts  
Theoph

Monthly Airline Passenger Numbers 1949-1960  
Sales Data with Leading Indicator  
Sales Data with Leading Indicator  
Biochemical Oxygen Demand  
Carbon Dioxide Uptake in Grass Plants  
Weight versus age of chicks on different diets  
Elisa assay of DNase  
Daily Closing Prices of Major European Stock Indices,  
Determination of Formaldehyde  
Hair and Eye Color of Statistics Students  
Harman Example 2.3  
Harman Example 7.4  
Pharmacokinetics of Indomethacin  
Effectiveness of Insect Sprays  
Quarterly Earnings per Johnson & Johnson Share  
Level of Lake Huron 1875-1972  
Intercountry Life-Cycle Savings Data  
Growth of Loblolly pine trees  
Flow of the River Nile  
Growth of Orange Trees  
Potency of Orchard Sprays  
Results from an Experiment on Plant Growth  
Reaction Velocity of an Enzymatic Reaction  
Road Casualties in Great Britain 1969-84  
Pharmacokinetics of Theophylline



# Introducción a R

Fundamentos de análisis exploratorio de datos

# Contenido del dataset

► `analisisExploratorio.R`

- Estructura interna de la variable
  - `str(variable)`
- Resumen del contenido
  - `summary(variable)`
- Exploración del contenido
  - `head(variable) | tail(variable)`
  - `variable[filas, columnas]`
  - `variable$columna`
  - `variable$columna[which(condición)]`
    - `iris$Sepal.Length[which(iris$Species == 'versicolor')]`

# Estadística descriptiva

► `analisisExploratorio.R`

## ► Funciones básicas (operan sobre vectores)

- `mean`           # media
- `median`       # mediana
- `var`            # varianza
- `sd`             # desviación estándar
- `max`            # máximo valor
- `min`            # mínimo valor
- `range`         # rango de valores
- `quantile`      # cuartiles

## ► Para estructuras complejas

- `lapply(iris[,1:4], mean)` # Aplicar a cada columna

# Agrupamiento de datos

## ► analisisExploratorio.R

- Tabla de contingencia con número de combinaciones

- Longitud de sépalo según especie

```
table(iris$Sepal.Length, iris$Species)
```

- Valoración de vendedores según moneda

```
table(ebay$sellerRating, ebay$currency)
```

- Agrupamiento y selección

- Separar los casos por especie de flor

```
split(iris, iris$Species)
```

- Obtener elevación, pendiente y clase de filas que cumplan condición

```
subset(covertype, slope > 45 & soil_type == '1',  
       select = c(elevation, slope, class))
```

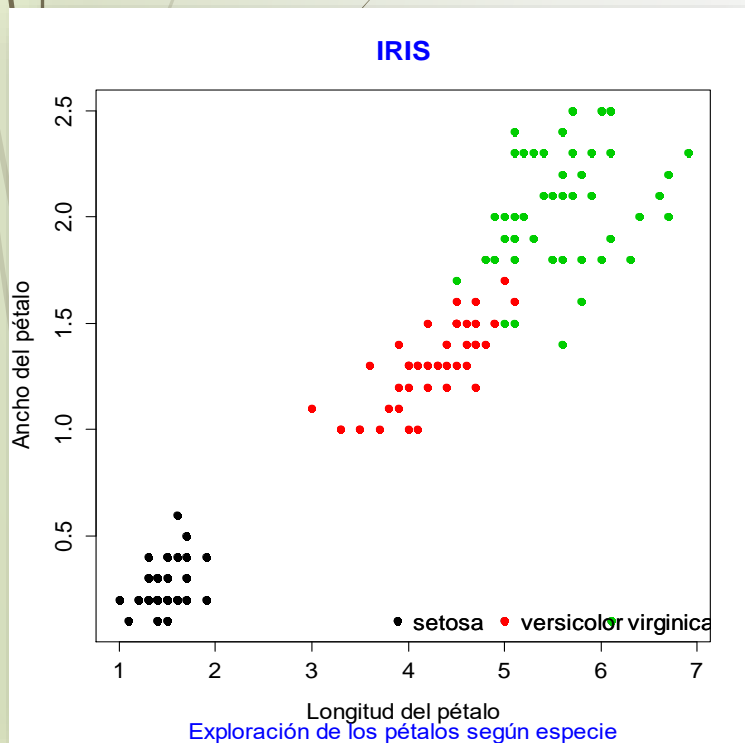
# Exploración visual - scatterplot

► análisisExploratorio.R

```
plot(iris$Petal.Length,  
     iris$Petal.Width,  
     col = iris$Species, pch = 19,  
     xlab = 'Longitud del pétalo',  
     ylab = 'Ancho del pétalo')
```

```
title(main = 'IRIS',  
      sub = 'Pétalos según especie',  
      col.main = 'blue',  
      col.sub = 'blue')
```

```
legend("bottomright",  
      legend = levels(iris$Species),  
      col = unique(iris$Species),  
      ncol = 3, pch = 19, bty = "n")
```



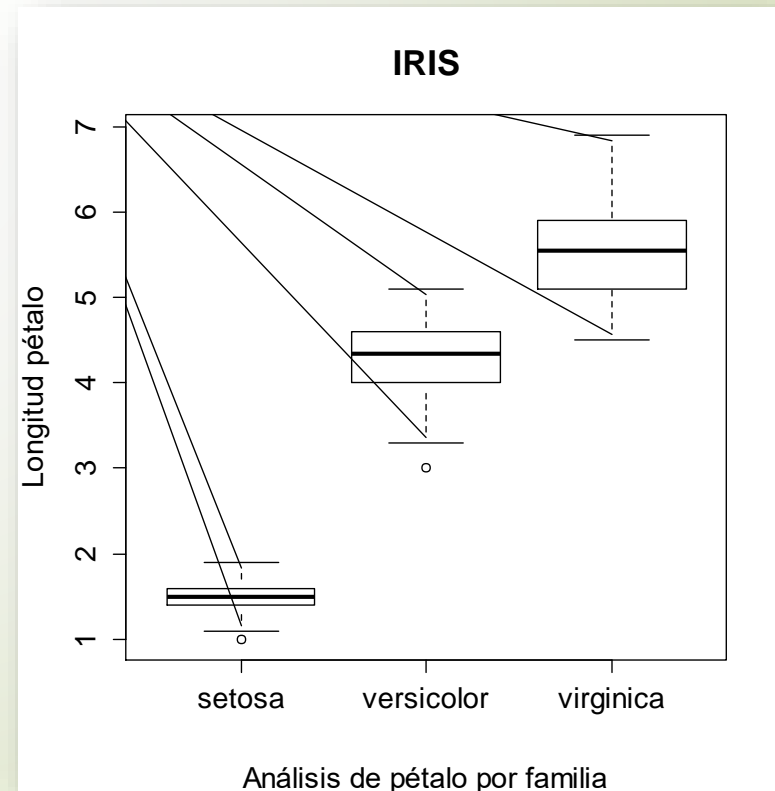


# Exploración visual - cajas y bigotes

► analisisExploratorio.R

```
boxplot(iris$Petal.Length ~ iris$Species)
```

```
title(main = 'IRIS',  
      ylab = 'Longitud pétalo',  
      sub = 'Análisis de pétalo por familia')
```

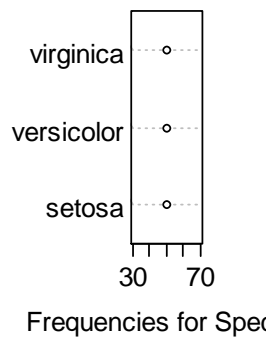
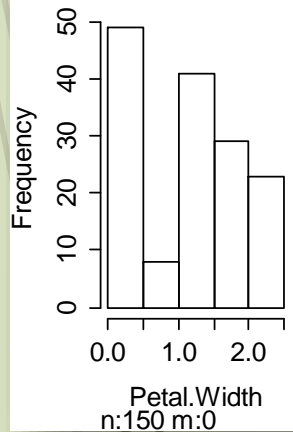
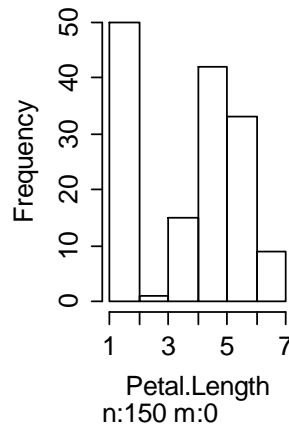
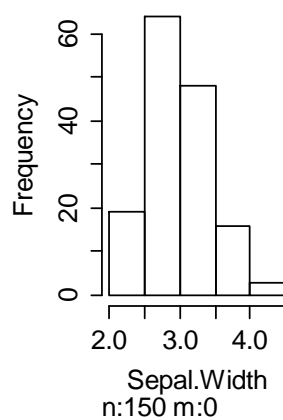




# Exploración visual - histograma

► análisisExploratorio.R

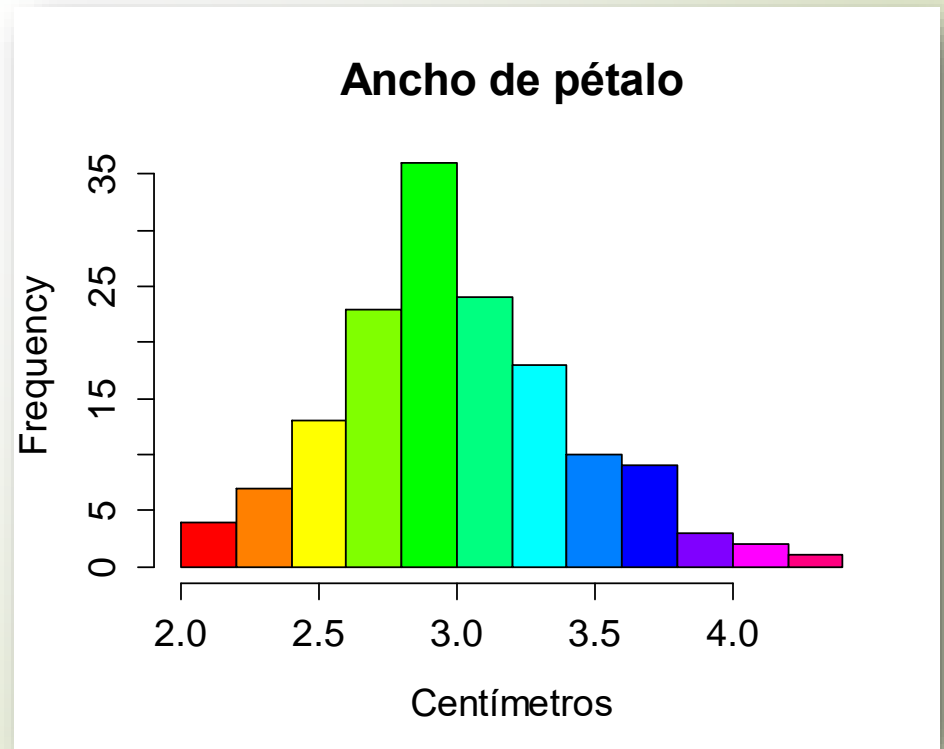
hist(iris)



# Exploración visual - histograma

► analisisExploratorio.R

```
hist(iris$Sepal.width, breaks = 12,  
     col = rainbow(12),  
     main = 'Ancho de sépalo',  
     xlab = 'Centímetros')
```





# Introducción a R y RStudio

Francisco Charte