



CURSOS DE VERANO 2014

APROXIMACIÓN PRÁCTICA A LA CIENCIA DE DATOS Y BIG DATA: HERRAMIENTAS KNIME, R, HADOOP Y MAHOUT

Visualización de datos con R

Francisco Charte Ojeda

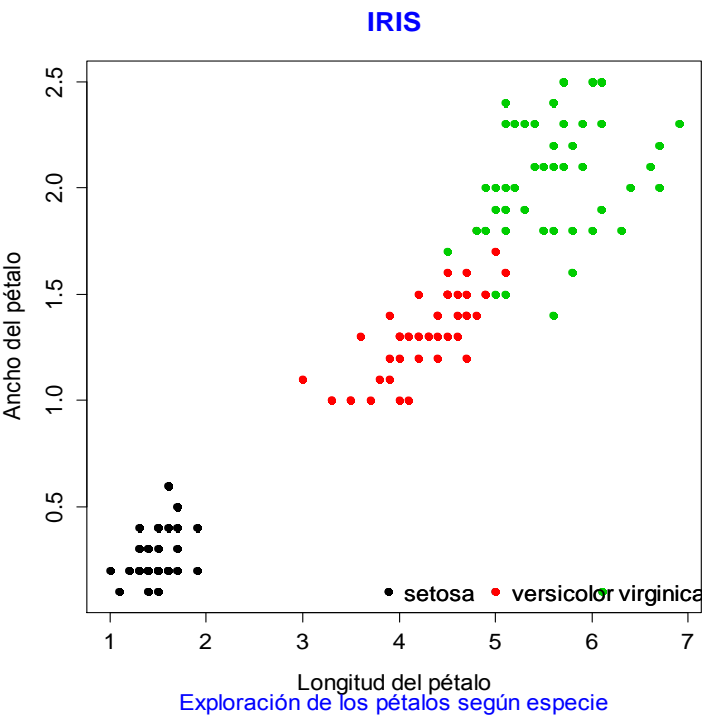
Contenidos

- ▣ Gráficos básicos
- ▣ Histogramas
- ▣ Agrupación de gráficos
- ▣ Almacenamiento de gráficos
- ▣ Introducción a ggplot2
- ▣ Otras posibilidades gráficas

>> `graficosBasicos.R` <<

>> `graficosAvanzados.R` <<

Gráficos básicos ► Nube de puntos

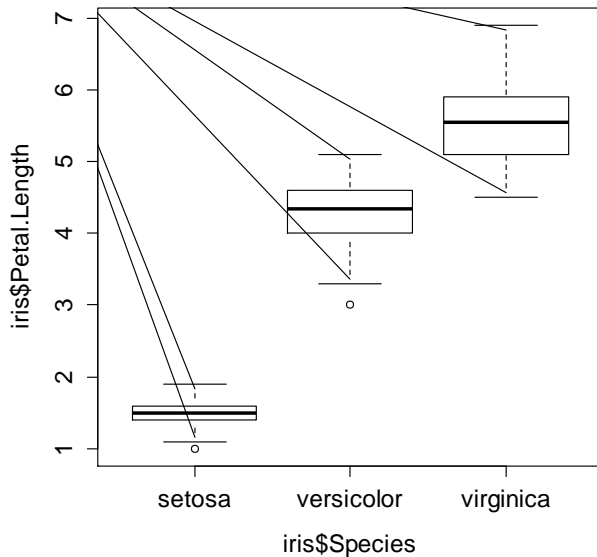


```
plot(iris$Petal.Length,
     iris$Petal.width,
     col = iris$Species, pch = 19,
     xlab = 'Longitud del pétalo',
     ylab = 'Ancho del pétalo')

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")
```

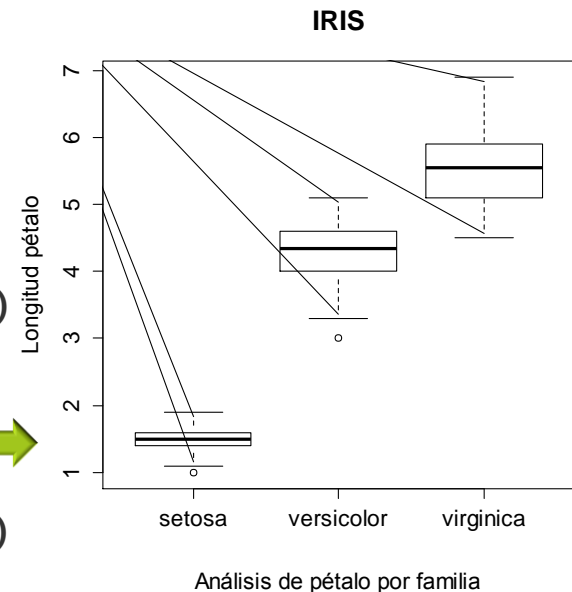
Gráficos básicos ▶ Gráfico de cajas (bigotes)



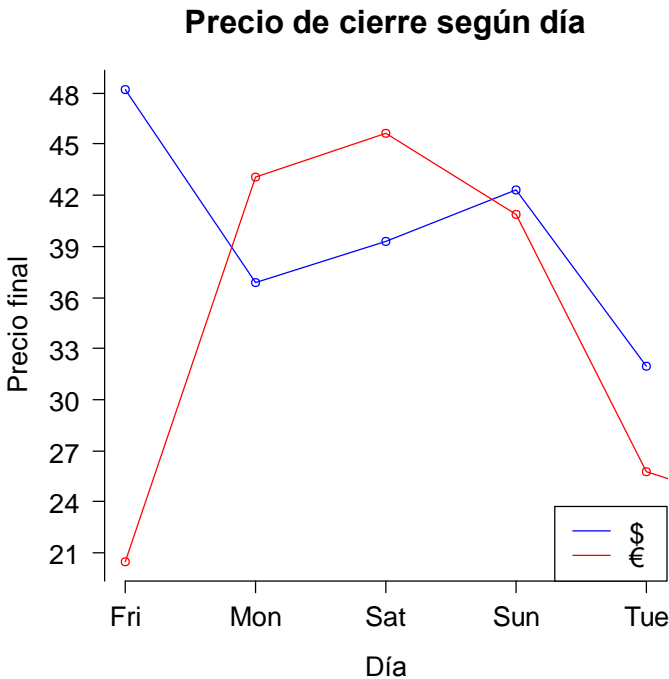
← `plot(iris$Petal.Length ~ iris$Species)`

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

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



Gráficos básicos ► Líneas



```
plot(meanPricesUS, type = "o",
     axes = F, ann = F, col = "blue")
```

```
lines(meanPricesEUR, type="o", col="red")
```

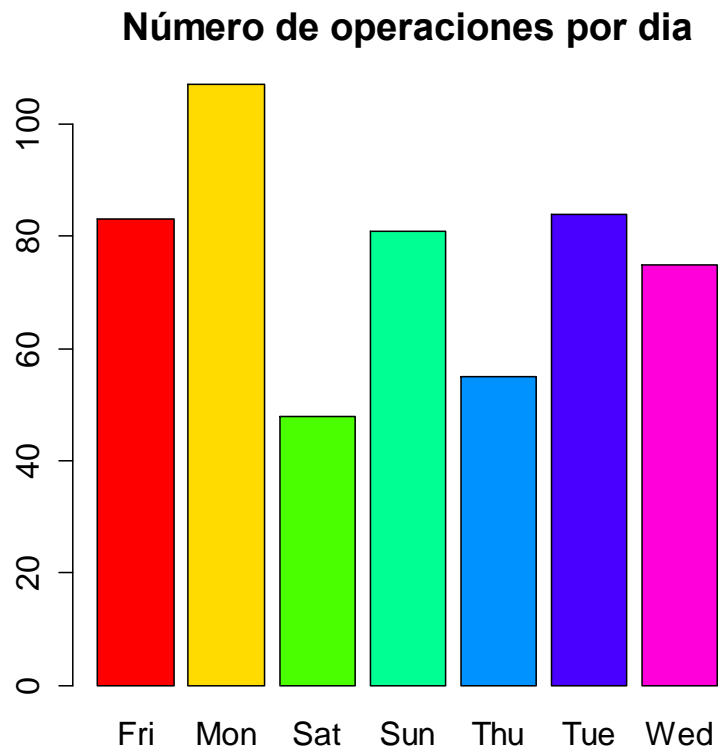
```
axis(1, at=1:length(meanPricesUS),
     lab = names(meanPricesUS))
```

```
axis(2, at = 3*0:rango[2], las = 1)
```

```
title(main='Precio de cierre según día',
      xlab = 'Día', ylab = 'Precio final')
```

```
legend("bottomright", c("$", "€"),
      col = c("blue", "red"), lty = c(1,1))
```

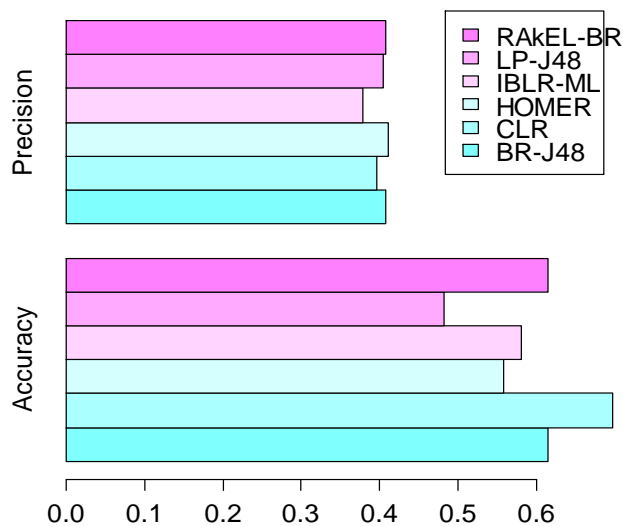
Gráficos básicos ► Barras



```
barplot(sapply(
  endPricePerDay$EUR, length),
  col = rainbow(7))

title(main = 'Operaciones por día')
```

Gráficos básicos ► Barras

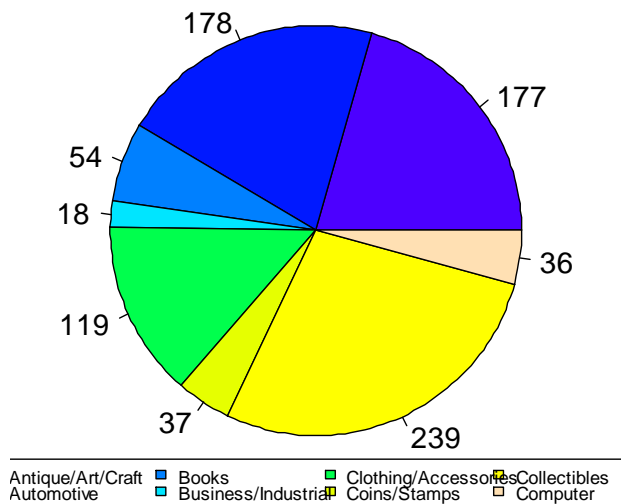


```
accuracy <- aggregate(
  Accuracy ~ Algorithm, results, mean)
precision <- aggregate(
  Precision ~ Algorithm, results, mean)
valMedios <- matrix(c(
  precision$Precision,
  accuracy$Accuracy),
  nrow=6, ncol=2)
rownames(valMedios) <-
  accuracy$Algorithm
```

```
barplot(valMedios, beside=T, horiz=T,
  col = cm.colors(6),
  legend.text=T, names.arg =
  c('Accuracy', 'Precision'))
```

Gráficos básicos ► Sectores (circular o tarta)

Productos por categoría



```
opPorCategoria <- aggregate(
  ClosePrice ~ Category,
  ebay, length)[1:8,]
```

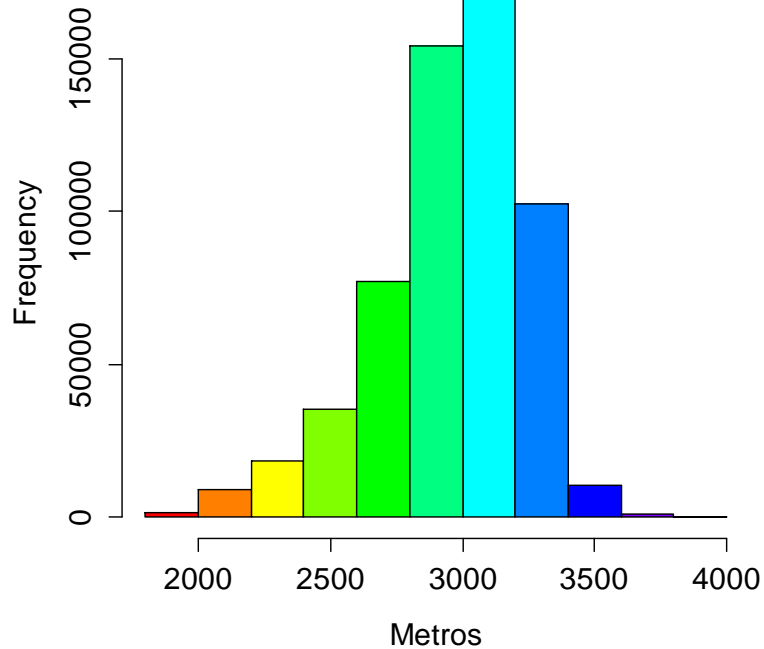
```
colores <- topo.colors(length(
  opPorCategoria$Category))
```

```
pie(opPorCategoria$ClosePrice,
  labels=opPorCategoria$ClosePrice,
  col=colores,
  main='Productos por categoría')
```

```
legend("bottom",
  opPorCategoria$Category,
  cex=0.6, fill=colores, ncol=4)
```


Histogramas ▶ Frecuencia

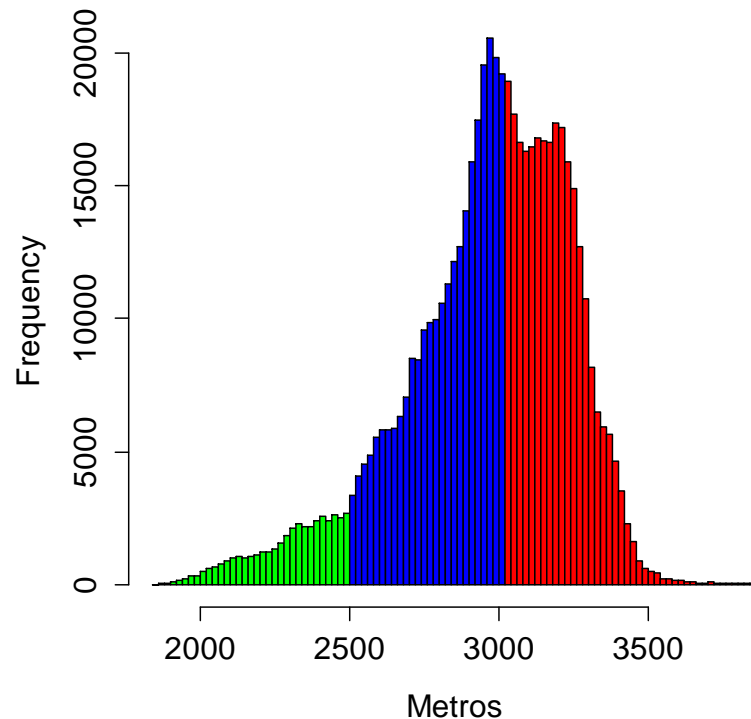
Elevación del terreno



```
hist(covertime$elevation,
     breaks = 12,
     col = rainbow(12),
     main = 'Elevación del terreno',
     xlab = 'Metros')
```

Histogramas ► Frecuencia

Elevación del terreno

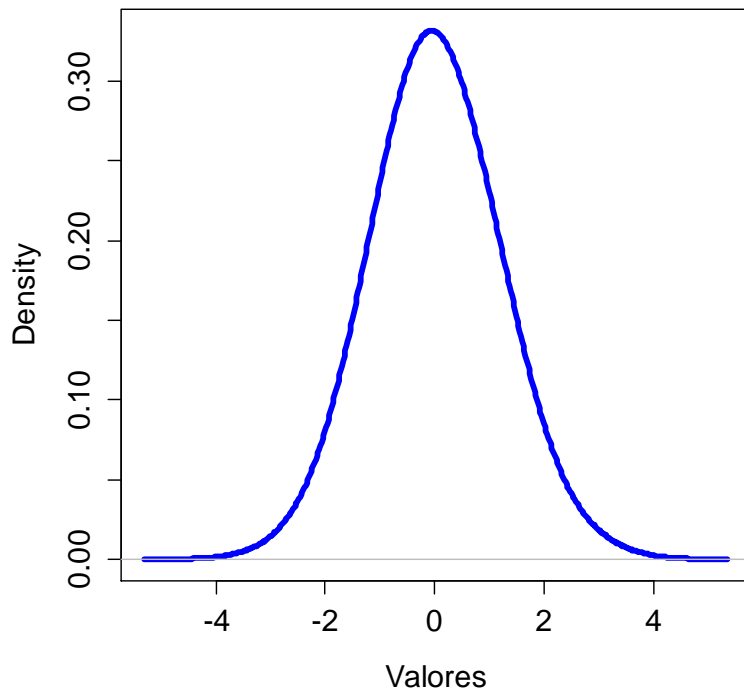


```
histograma <- hist(covertypes$elevation,
                    breaks = 100)
```

```
plot(histograma, col =
     ifelse(histograma$breaks < 2500,
           'green',
           ifelse(histograma$breaks > 3000,
                 "red", "blue")),
     main = 'Elevación del terreno',
     xlab = 'Metros')
```

Histogramas ► Densidad

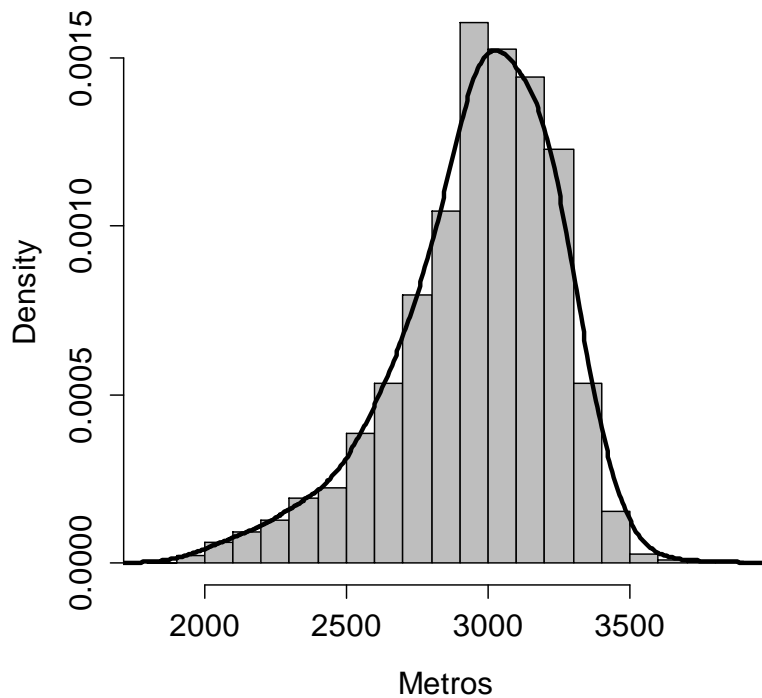
Distribución normal



```
plot(density(rnorm(1000),
  adjust = 3),
  col = 'blue', lwd = 4,
  main = 'Distribución normal',
  xlab = 'valores')
```

Histogramas ► Histograma + Densidad

Elevación del terreno

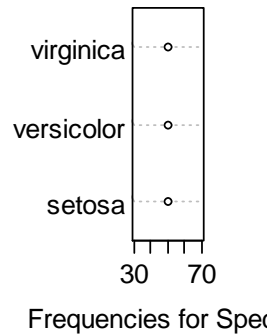
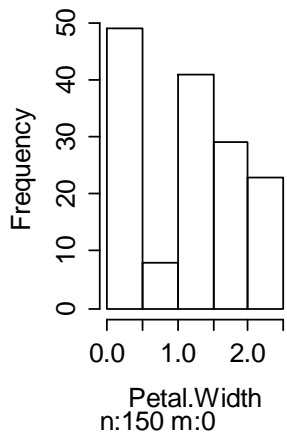
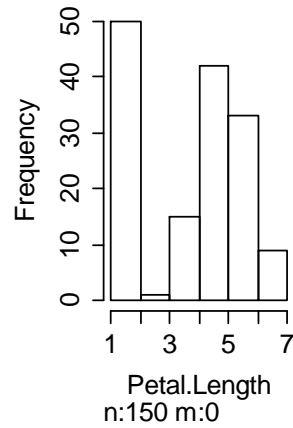
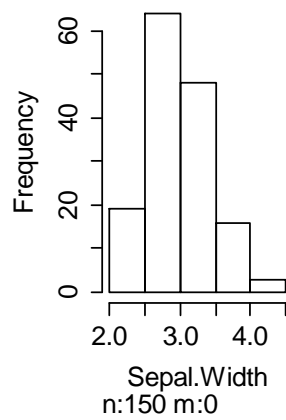
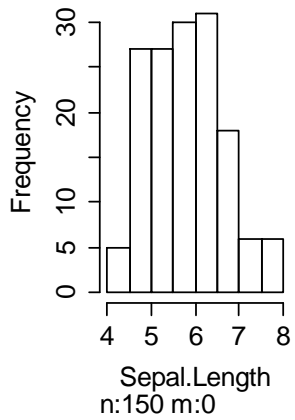


```
hist(covertype$elevation, prob = T,
     col = "grey",
     main = 'Elevación del terreno',
     xlab = 'Metros')
```

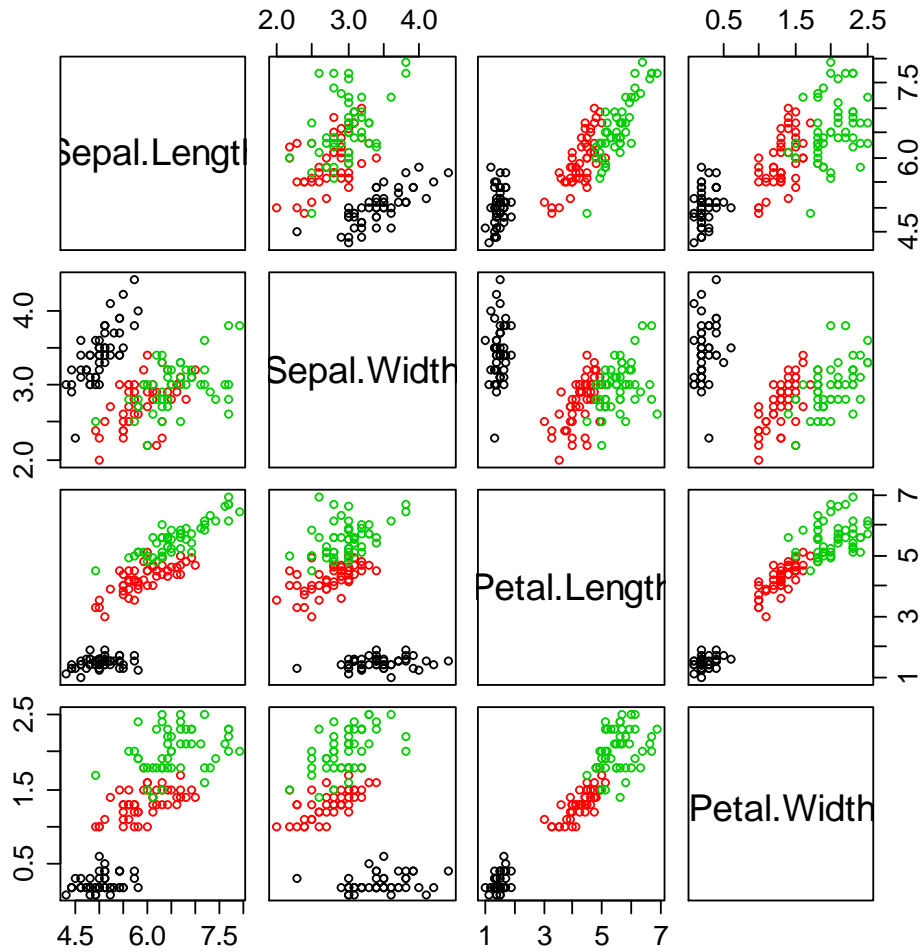
```
lines(density(covertype$elevation,
              adjust = 5), col = 'black',
      lwd = 3)
```

Histogramas ▶ Histogramas de objetos compuestos

hist(iris)

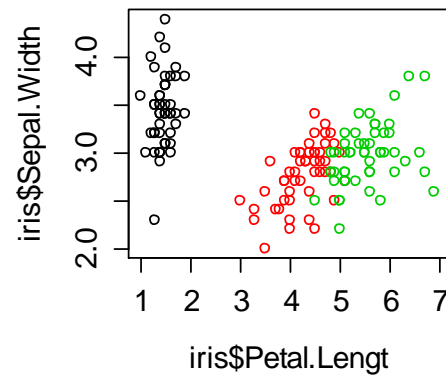
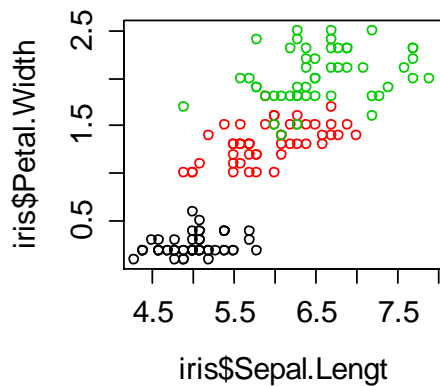
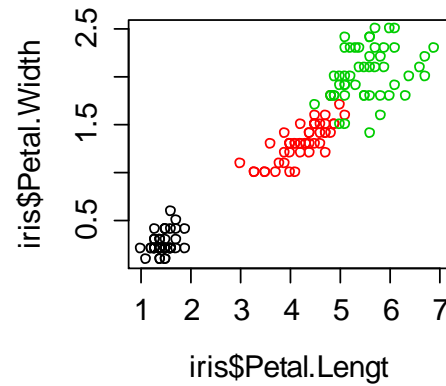
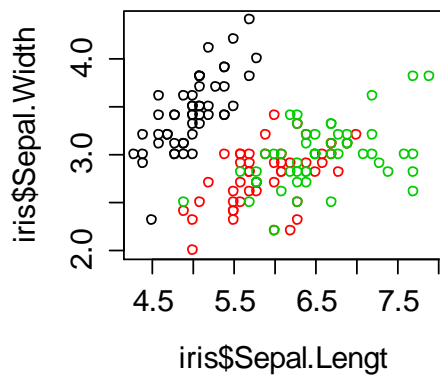


Agrupación de gráficos ► Proyecciones 2D



```
plot(iris[,1:4],
     col = iris$Species)
```

Agrupación de gráficos ► mfrow/mfcol

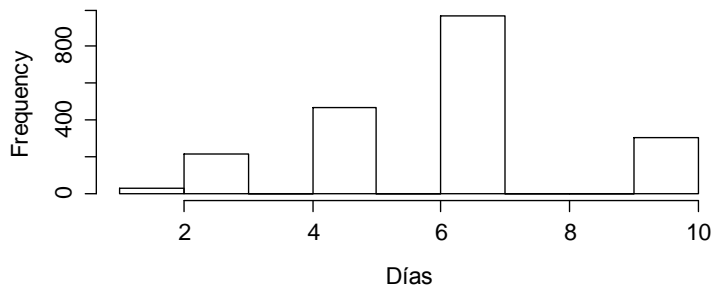


```

prev <- par(mfrow=c(2,2))
plot(iris$Sepal.Length,
      iris$Sepal.Width,
      col=iris$Species)
plot(iris$Petal.Length,
      iris$Petal.Width,
      col=iris$Species)
plot(iris$Sepal.Length,
      iris$Petal.Width,
      col=iris$Species)
plot(iris$Petal.Length,
      iris$Sepal.Width,
      col=iris$Species)
par(prev)
  
```

Agrupación de gráficos ► layout

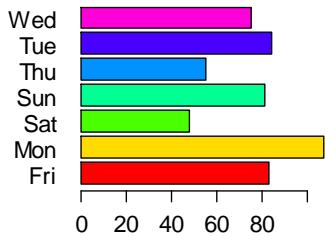
Duración subasta



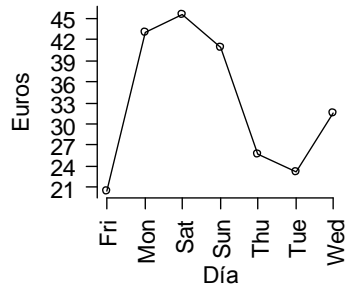
```
# 2x2 con 3 gráficas
layout(matrix(c(1,1,2,3), 2, 2,
              byrow = T))
```

```
# Gráfico 1 ocupa 1ª fila
hist(ebay$Duration,
     main='Duración subasta',
     xlab='Días')
```

Operaciones por día



Precio cierre por día



```
# Gráfico 2 en 1ª col de 2ª fila
barplot(sapply(
  endPricePerDay$EUR, length),
  col = rainbow(7),
  horiz=T, las=1)
```

```
# Gráfico 3 en 2ª col de 2ª fila
plot(meanPricesEUR, type="o")
```


Almacenamiento de gráficos ▶ opciones de RStudio

Save Plot as Image

Image format: **PNG** (selected)
Directory: /book/data
File name:
Width: 553 Height: 336
 Maintain aspect ratio
Update Preview

Elevación

Frequency

0 5000 15000

2000 2500 3000 3500

Metros

View plot after saving
Save Cancel

Almacenamiento de gráficos ► Funciones de R

- Generan un archivo en el formato adecuado
 - Mapas de bits: `png()`, `bmp()`, `jpeg()`, `tiff()`
 - Vectoriales: `win.metafile()`
 - Otros: `postscript()`, `pdf()`

- Parámetros generales
 - `filename`, `width`, `height`, `pointsize`, `quality`

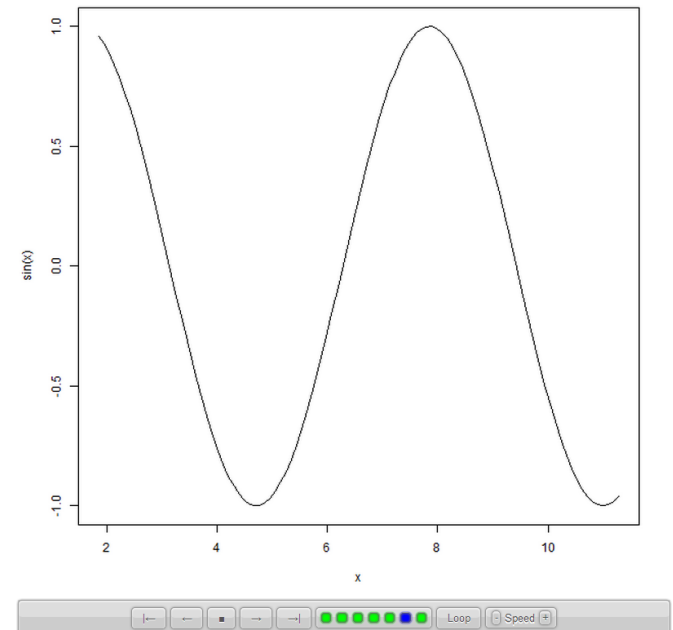
- Desactivación de la salida al archivo
 - `dev.off()`

- Ejemplo
 - ```
pdf('Subastas.pdf', width=8.3, height=11.7) # A4
hist(iris) # Una gráfica por página
dev.off() # Cerrar PDF
```

## Almacenamiento de gráficos ► Animaciones

- Funciones en paquete `animations`
  - `saveGIF()`, `saveVideo()`, `saveHTML()`, `saveLatex()`
- Ejemplo de uso

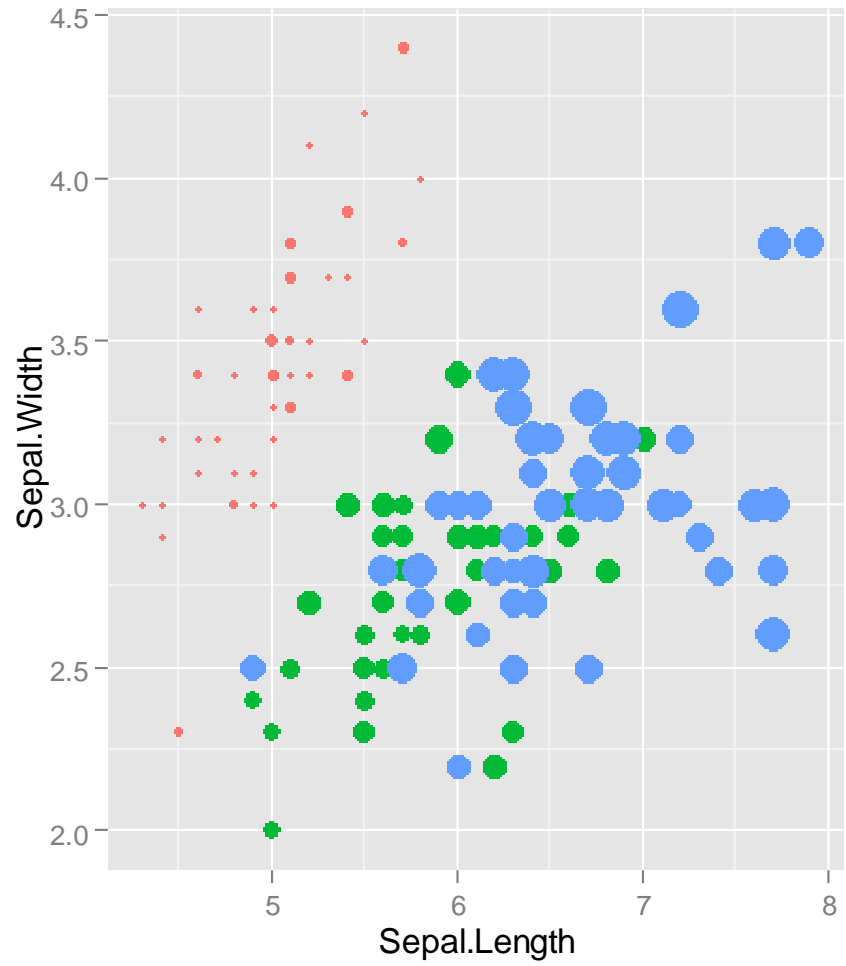
```
saveGIF({
 for(lim in seq(-3.14,3.14,by=0.1))
 curve(sin, from=lim,
 to=lim + 9.42)
}, movie.name="animacion.gif",
 interval=0.2, ani.width=640,
 ani.height=640)
```



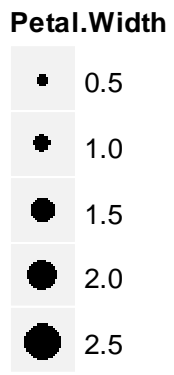
## Introducción a ggplot2

- Ventajas sobre los gráficos básicos
  - Estética mejorada y altamente personalizable
  - Mayor flexibilidad y control
  - Más facilidad para generar gráficos con datos complejos
  
- Recursos necesarios
  - Instalar paquete `ggplot2`
  - Documentación en <http://docs.ggplot2.org/current>
  - *ggplot2: Elegant Graphics for Data Analysis*
    - Ejemplos disponibles en <http://ggplot2.org/book>

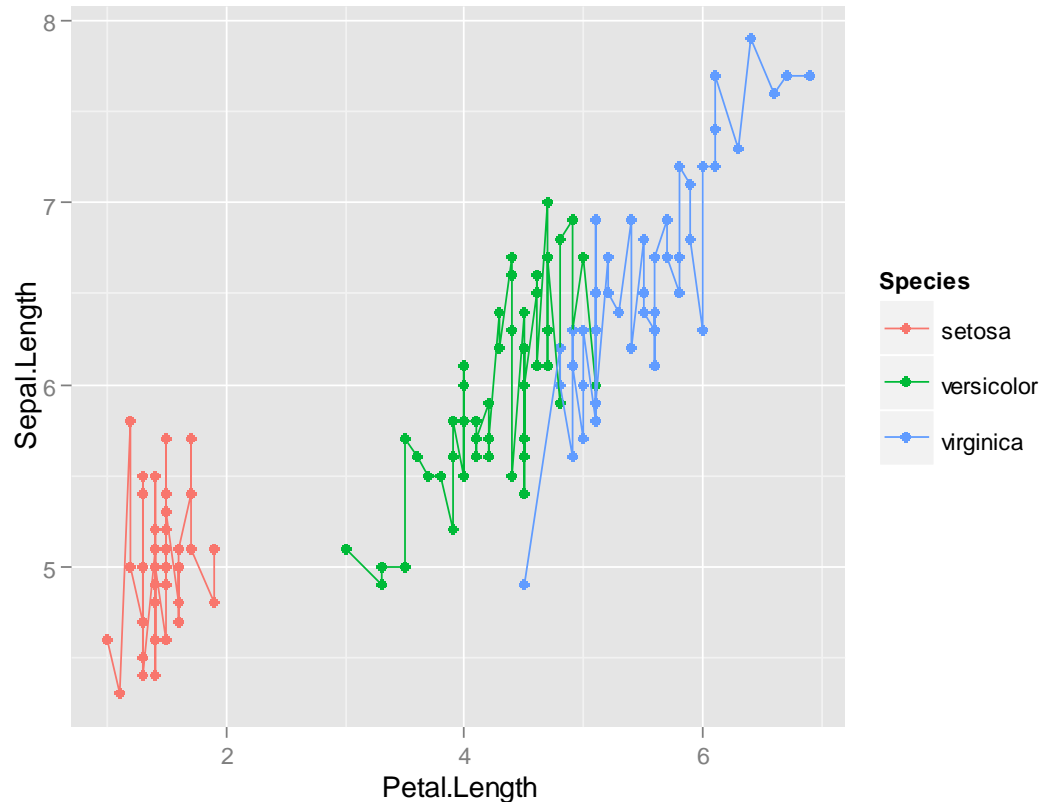
# Introducción a ggplot2 ► Nube de puntos



```
ggplot(Sepal.Length,
 Sepal.Width,
 data=iris,
 colour=Species,
 size=Petal.Width)
```



# Introducción a ggplot2 ► Líneas

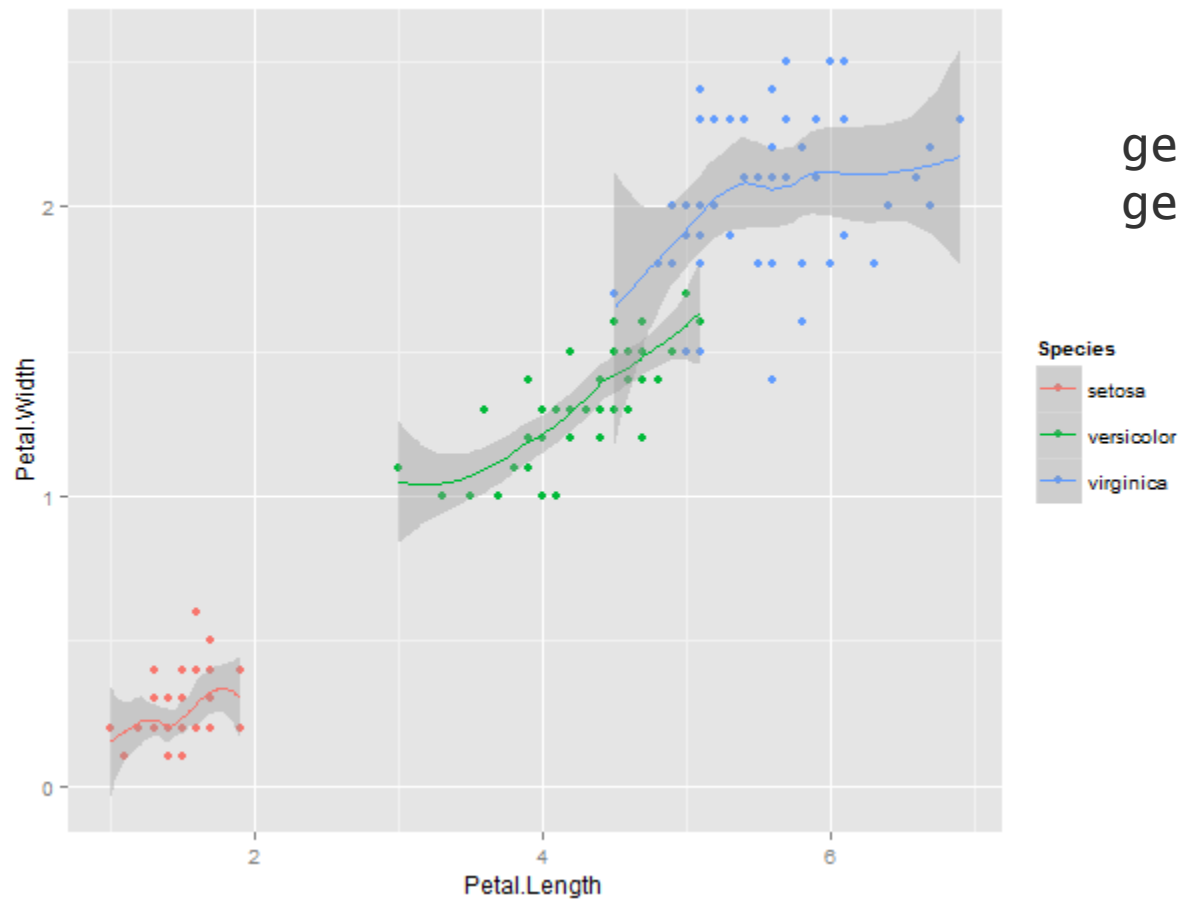


```

ggplot(Petal.Length,
 Sepal.Length,
 data=iris,
 color=Species) +
geom_line()

```

# Introducción a ggplot2 ► Nube de puntos + Curvas de regresión



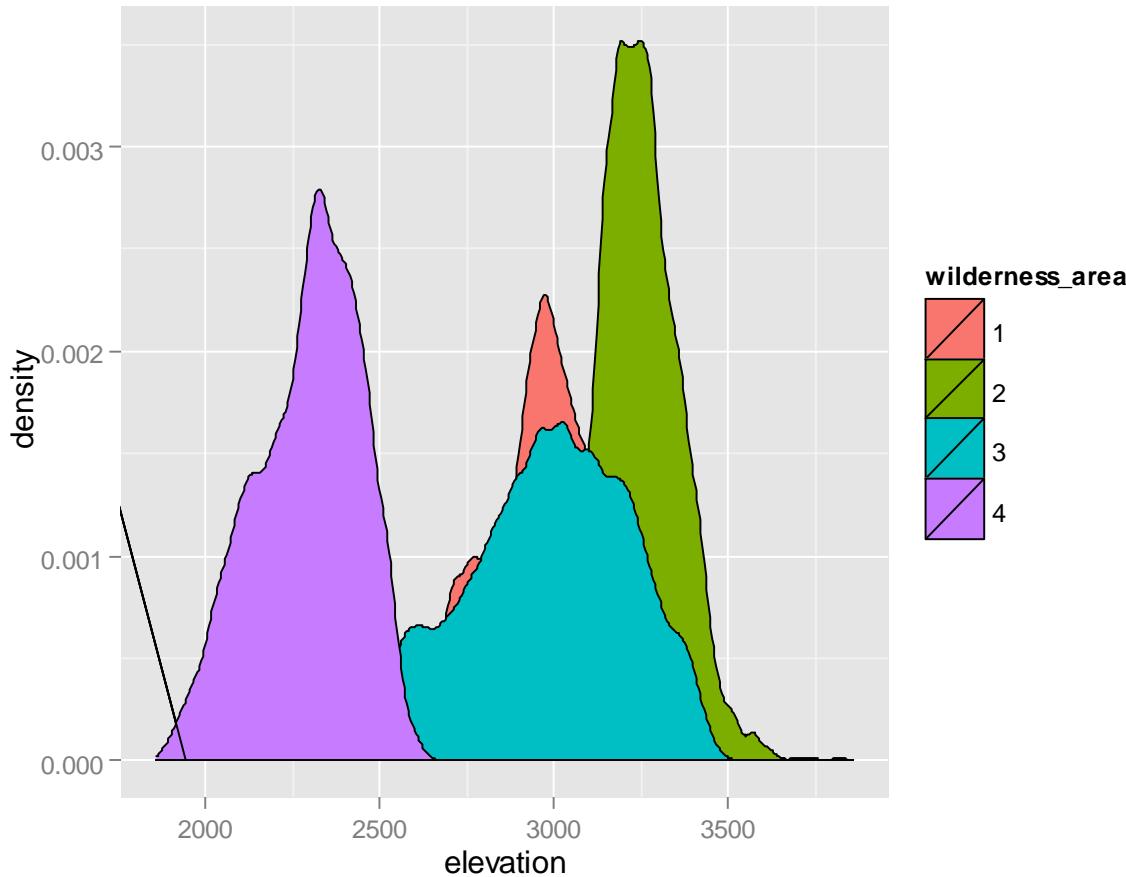
```

qplot(Petal.Length,
 Petal.Width, data=iris,
 color=Species) +
geom_point() +
geom_smooth()

```

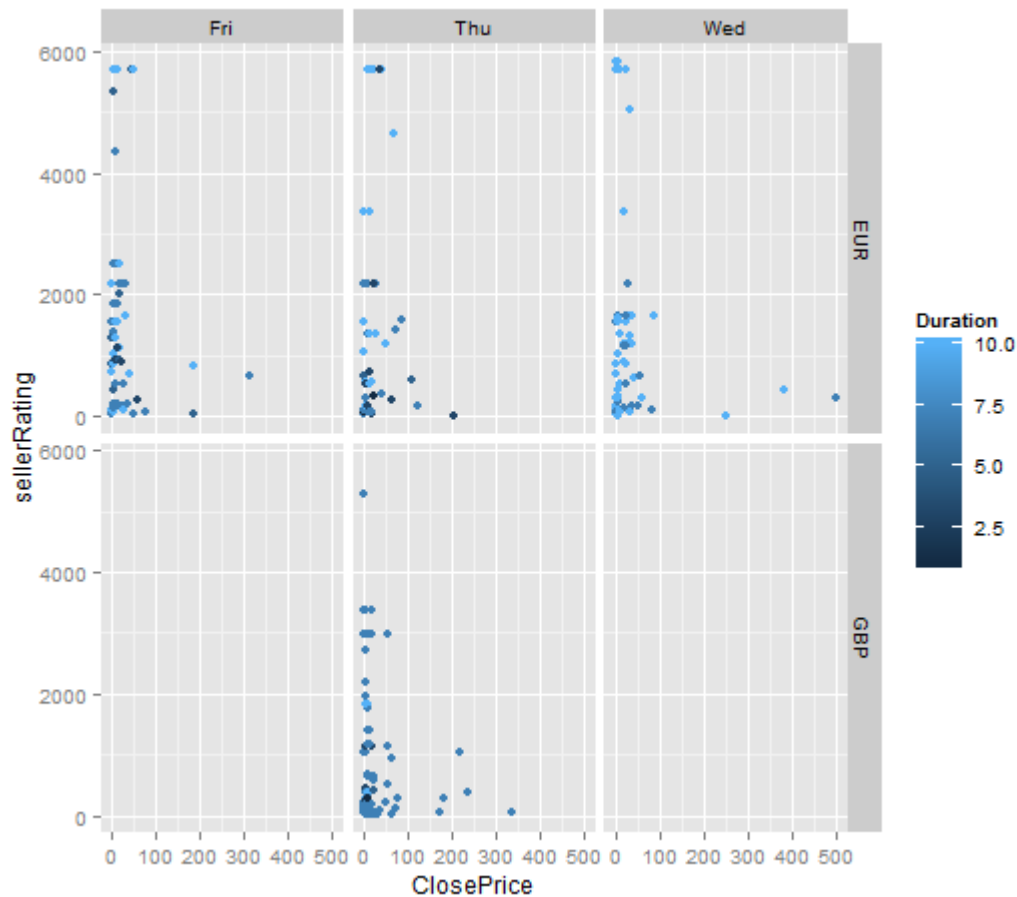
# Introducción a ggplot2 ► Densidad

```
qplot(elevation,
 data=covertypes,
 geom="density",
 fill=wilderness_area)
```





# Introducción a ggplot2 ► Facets



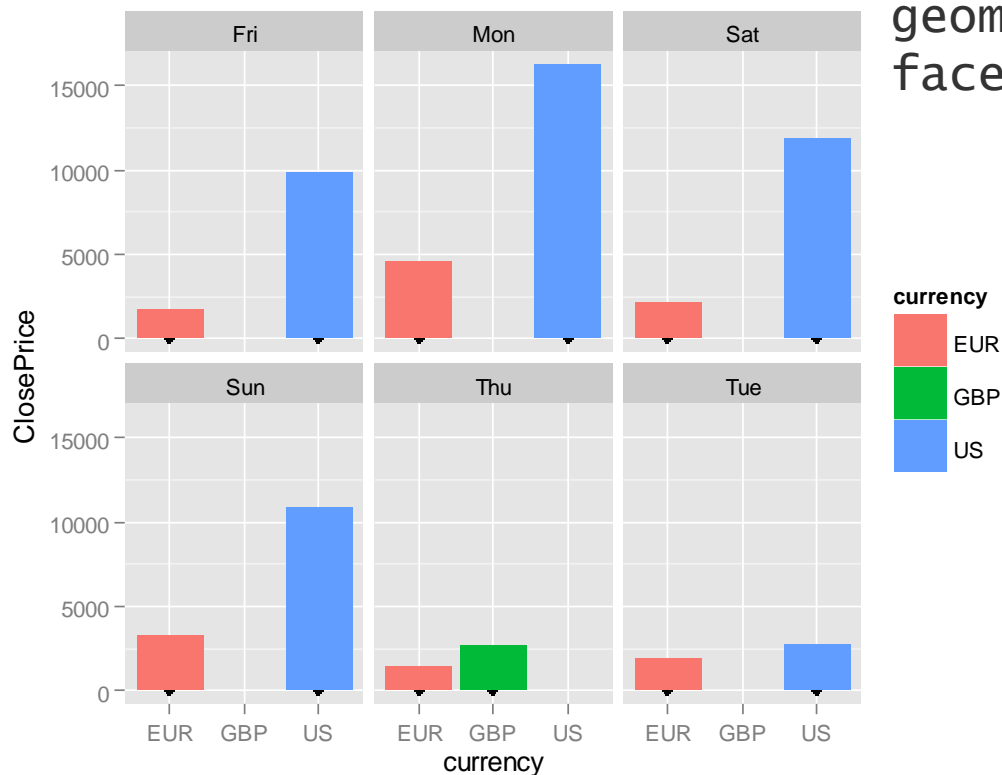
```
Representar 5 variables
qplot(ClosePrice,
 sellerRating,
 data=ebay[ebay$endDay %in%
 c('Wed', 'Thu', 'Fri') &
 ebay$currency != 'US',],
 facets=currency ~ endDay,
 color=Duration)
```

# Introducción a ggplot2 ▶ Facets

```

ggplot(currency, ClosePrice,
 data=ebay[ebay$endDay != 'wed',],
 fill=currency) +
 geom_bar(stat='identity') +
 facet_wrap(~endDay)

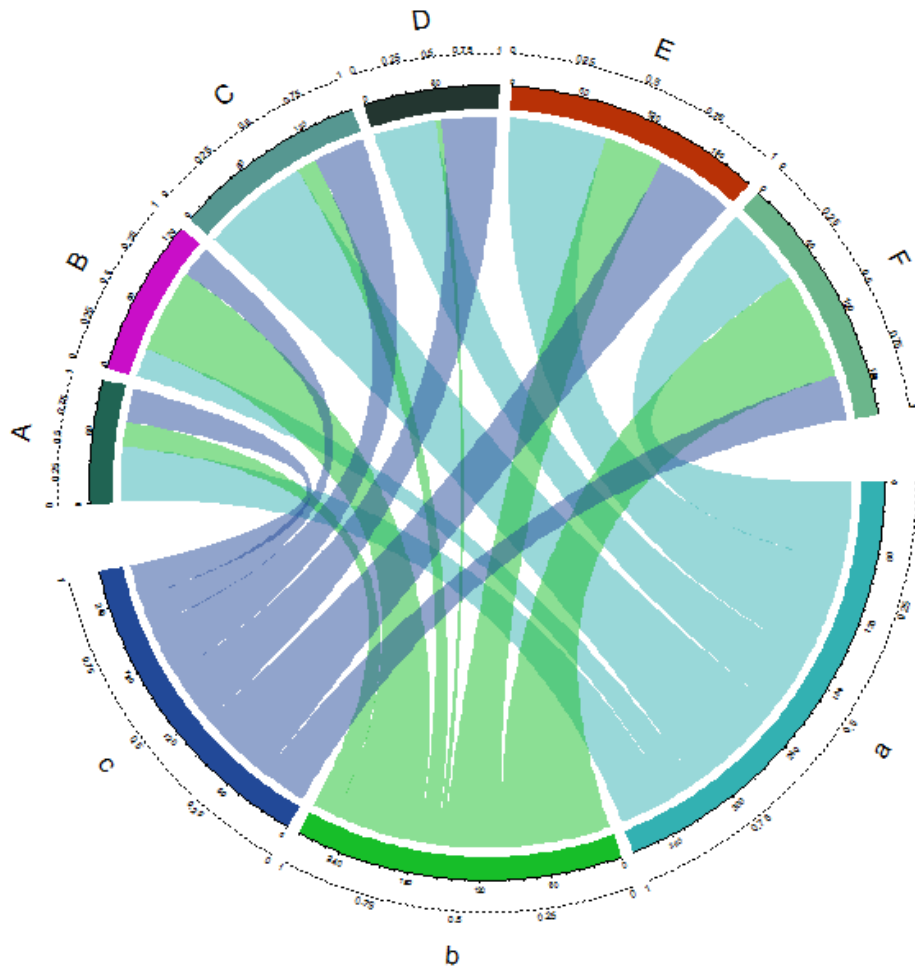
```



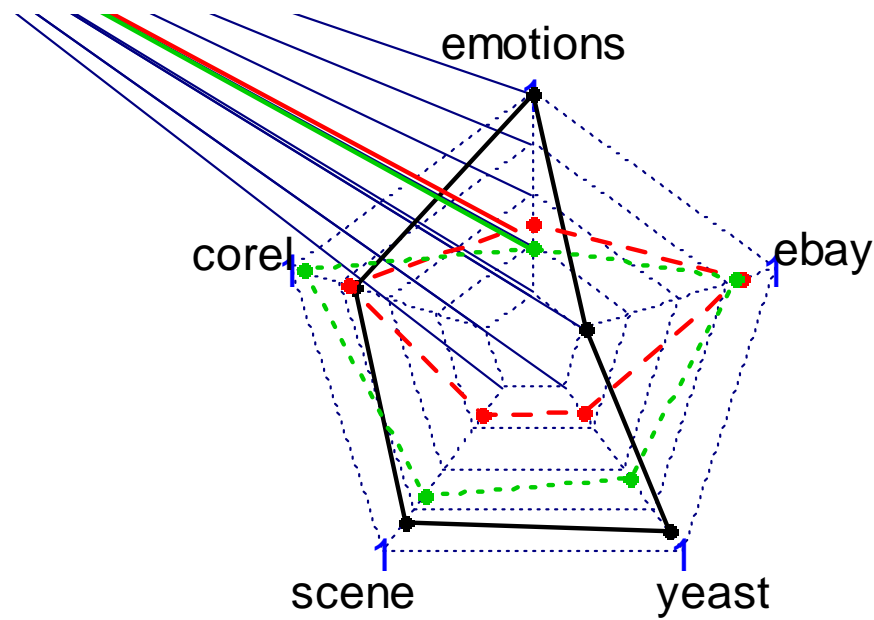
## Otras posibilidades gráficas

- Medio centenar de paquetes gráficos sólo en CRAN
  - <http://cran.r-project.org/web/views/Graphics.html>
- Interacción con los gráficos desde navegador web
  - Shiny: <http://shiny.rstudio.com/>
- Multitud de tipos de gráficas alternativas
  - circos, radar, 3D, etc.
- Gráficos no generados a partir de datos
  - TurtleGraphics

# Otras posibilidades gráficas ► circlize

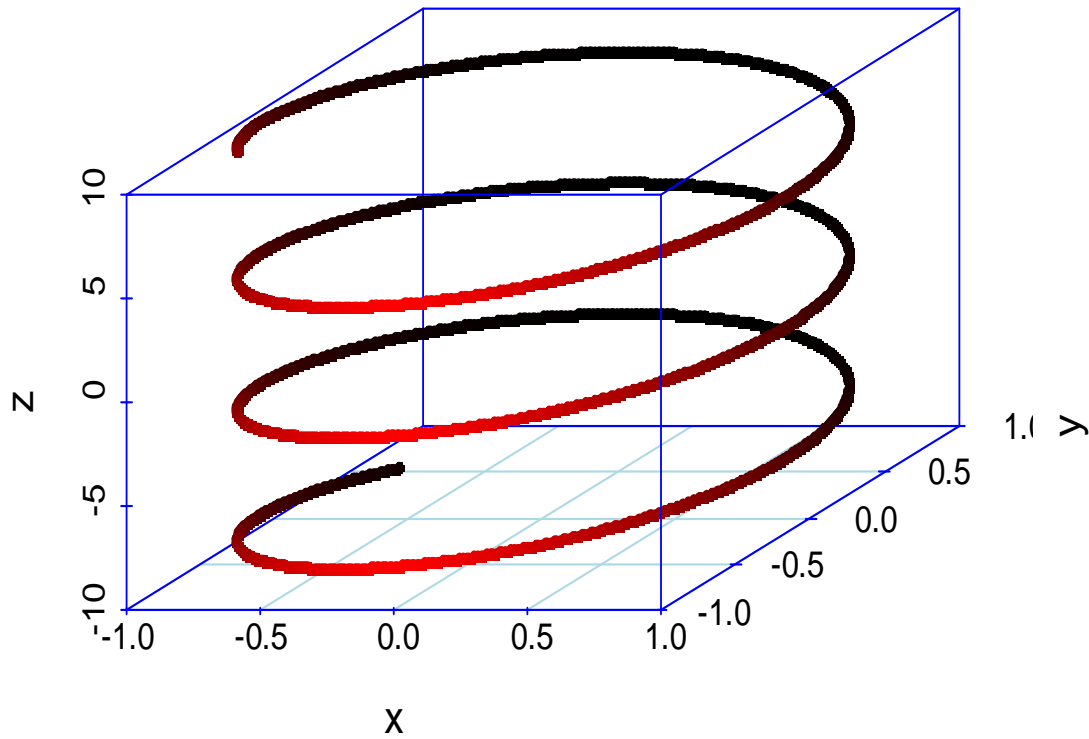


Otras posibilidades gráficas ► radarchart (paquete fmsb)

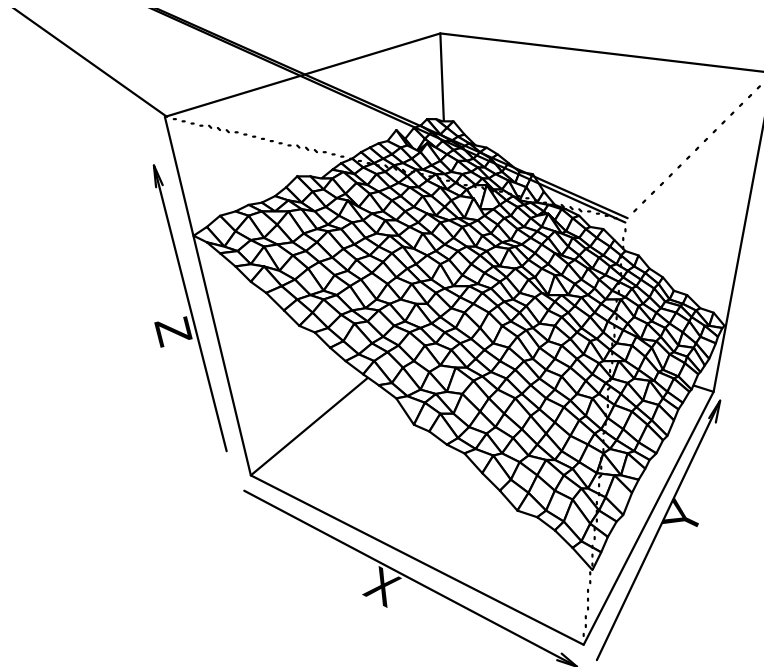


Otras posibilidades gráficas ► scatterplot3d

Helix

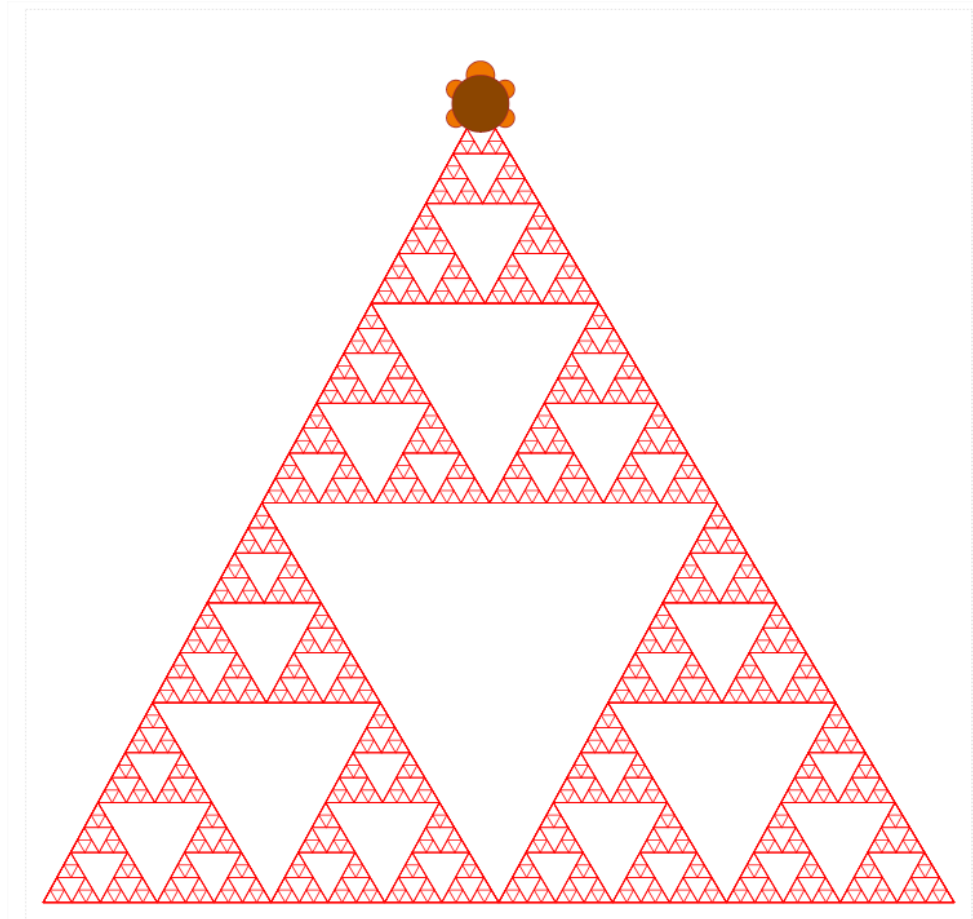
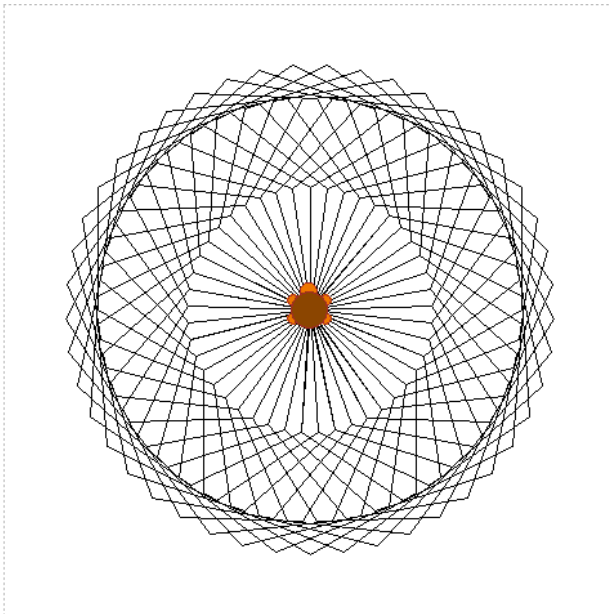


## Elevación del terreno



## Otras posibilidades gráficas ► TurtleGraphics

```
turtle_do({
 for(j in 1:45) {
 for(i in 1:6) {
 turtle_forward(20)
 turtle_right(360/6)
 }
 turtle_right(360/45)
 })
```







# CURSOS DE VERANO 2014

**APROXIMACIÓN PRÁCTICA A LA CIENCIA DE DATOS Y BIG DATA: HERRAMIENTAS KNIME, R, HADOOP Y MAHOUT**

**Visualización de datos con R**

**Francisco Charte Ojeda**