

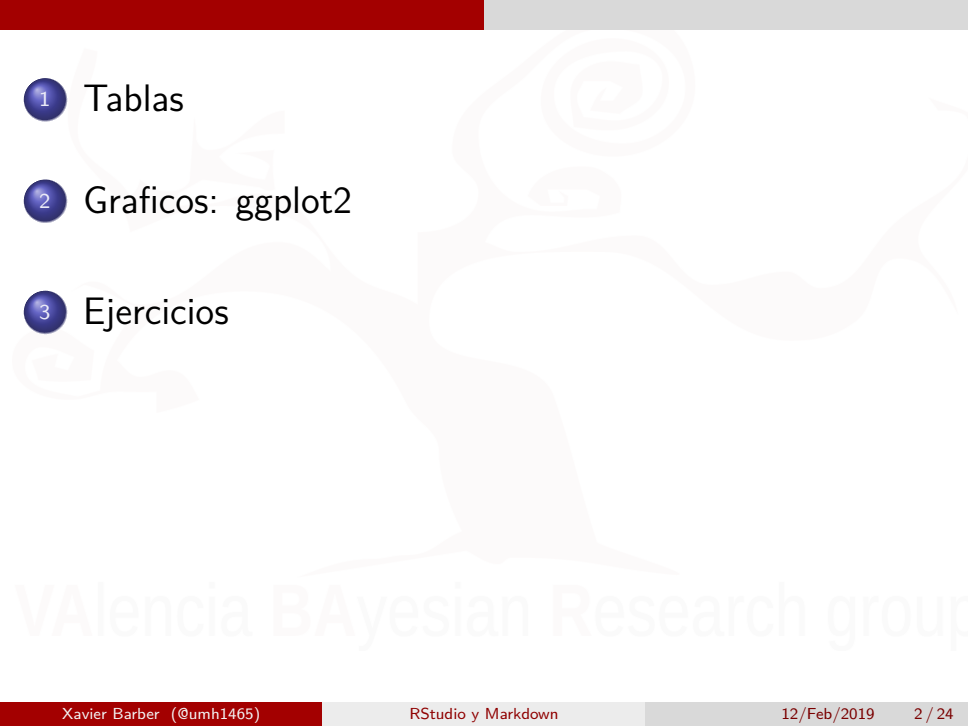
# RStudio y Markdown: Las salidas de R dejan de ser feas

Análisis estadístico de series económicas

Xavier Barber

Departamento de Estadística, Matemáticas e Informática  
Centro de Investigación Operativa  
Universitas Miguel Hernández de Elche

12/Feb/2019

- 
- 1 Tablas
  - 2 Graficos: ggplot2
  - 3 Ejercicios

Valencia Bayesian Research group

# Introducción

-Ya hemos visto que el RStudio es algo más que un editor de sintaxis de Word.

- Veamos ahora ejemplos claros de lo se puede hacer con Markdown y los paquetes `rmarkdown` y `knitr`.
- Este último paquete es el encargado de “compilar” lo que aparece en pantalla y convertirlo en HTML, pdf o WORD.

# Tablas

Valencia Bayesian Research group

# kable

- El primer comando que debemos aprender es el comando `kable`.
  - Convierte cualquier estructura tipo `data.frame` o `matrix` en una tabla bonita.
  - el paquete `kableExtra` nos ayudará a conseguir tablas todavía más profesionales. Remarcar que la máxima potencialidad de este paquete se obtiene si utilizamos LaTeX o HTML en nuestros documentos.
- Por desgracia hay funciones que no están implementadas todavía para que se muestren en WORD, pero si en HTML.

# Consas a considerar (I)

Muy importante aprender el significado de los “comandos” del chunk.

Yo suelo utilizar al principio del documento:

```
knitr::opts_chunk$set(echo = TRUE,  
  message=FALSE, warning=FALSE,  
  cache=TRUE, comment=" ")
```

Valencia Bayesian Research group

# Ejemplo 1

```
library(knitr)
library(rmarkdown)
library(fpp2)
data(elecdemand)
kable(elecdemand[1:5,])
```

Demand	WorkDay	Temperature
3.914647	0	18.2
3.672550	0	17.9
3.497539	0	17.6
3.339145	0	16.8
3.204313	0	16.3

# Insertar espacios y otras cosas de formato

- Para insertar un pequeño espacio entre párrafos se utiliza `&nbsp;` y esto hace que se separen los párrafos
- Es muy importante para un buen funcionamiento que nos creamos una plantilla en WORD y así podamos enviarle “información” al documento, como por ejemplo saltos de página o formato de los encabezados, etc.: enlace
- Y ahora voy a insertar un salto de página:

`<br>`



# Tabla de descriptivos

- Existen diferentes opciones de realizar una tabla de descriptivos.
- Paquetes como `stargazer` o `reporttools` funcionan muy bien en HTML o LaTeX, pero no para WORD

```
library(stargazer)
#aunque esto no va en WORD
stargazer(data.frame(elecdemand), header=FALSE)
```

# Tabla de descriptivos

```
library(stargazer)  
stargazer(data.frame(elecdemand), header=FALSE)
```

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)
Demand	17,520	4.610	0.878	2.858	3.927	5.159
WorkDay	17,520	0.688	0.463	0	0	1
Temperature	17,520	16.505	5.614	1.500	12.600	19.400

# Tabla de descriptivos

- La mejor opción para crear un `data.frame` o `data.table` con el contenido que queremos imprimir:

```
library(dplyr)
temp<-data.frame(elecddemand)
temp$WorkDay<-factor(temp$WorkDay, labels=c("Laboral","Festivo"))
tabla_descriptivos <- summarise(group_by( temp,WorkDay),
                                Media=mean(Demand, na.rm=TRUE),
                                Desv=sd(Demand, na.rm=TRUE),
                                Maximo=max(Demand, na.rm=TRUE))
kable(tabla_descriptivos)
```

# Tabla de descriptivos

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

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

WorkDay	Media	Desv	Maximo
Laboral	4.104919	0.6587216	7.819034
Festivo	4.839323	0.8688780	9.345004

# Mejorando la tabla

- para mejorar la tabla obtenida tenemos en primer lugar las opciones del comando `kable`
  - `kable(x, format, digits = getOption("digits"), row.names = NA, col.names = NA, align, caption = NULL, format.args = list(), escape = TRUE, ...)`
- Y como segunda opción la utilización del paquete `flextable`: [enlace](#)

# flextable

```
library(flextable)
myft <- flextable(
  tabla_descriptivos,
  col_keys = c("WorkDay", "Media" , "Desv", "Max")
myft2<-regulartable(tabla_descriptivos)
tbl<-autofit(myft2)
```

# flextable

WorkDay	Media	Desv	Maximo
Laboral	4.105	0.659	7.819
Festivo	4.839	0.869	9.345

# fletable

```
myft <- flextable(data.frame(dt), col_keys = c("WorkDay", "Med",  
  "Desv", "Maximo"))  
myft <- italic(myft, j = 1)  
myft <- bg(myft, bg = "#C90000", part = "header")  
myft <- color(myft, color = "white", part = "header")  
myft <- color(myft, ~Maximo > 8, ~Maximo, color = "red")  
myft <- bold(myft, ~Maximo > 8, ~Maximo, bold = TRUE)  
myft <- autofit(myft)
```



## flextable

WorkDay	Media	Desv	Maximo
<i>Laboral</i>	4.105	0.659	7.819
<i>Festivo</i>	4.839	0.869	<b>9.345</b>

# Graficos: ggplot2

Valencia Bayesian Research group

# ggplot2

- Aunque inicialmente su sintaxis se aleja de los comandos habituales del R , es cierto que al final se pueden conseguir gráficos muy “profesionales”.

ggplot2. Elegant Graphics for Data Analysis. Hadley Wickham (2016). enalce

Learn ggplot2 Using Shiny App. Keon-Woong Moon (2016).

Valencia Bayesian Research group

# ggplot2

```

# install.packages("ggplot2")
# load package and data
options(scipen=999) # turn-off scientific notation like 1e+48
library(ggplot2)
theme_set(theme_bw()) # pre-set the bw theme.
data("midwest", package = "ggplot2")
# midwest <- read.csv("http://goo.gl/G1K41K") # bkup data source

# Scatterplot
gg <- ggplot(midwest, aes(x=area, y=poptotal)) +
  geom_point(aes(col=state, size=popdensity)) +
  geom_smooth(method="loess", se=F) +
  xlim(c(0, 0.1)) +
  ylim(c(0, 500000)) +
  labs(subtitle="Area Vs Population",
       y="Population",
       x="Area",
       title="Scatterplot",
       caption = "Source: midwest")

plot(gg)

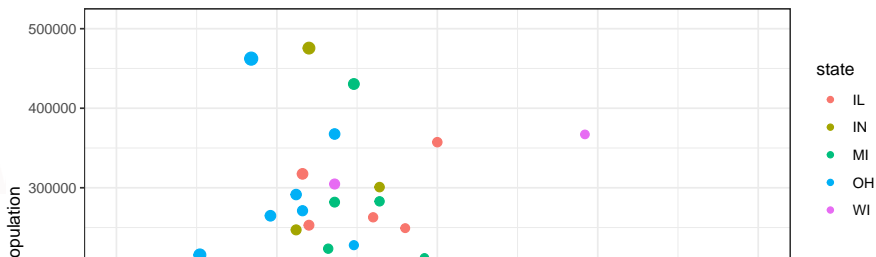
```

# ggplot2

```
## Warning: Removed 15 rows containing non-finite
## (stat_smooth).
```

```
## Warning: Removed 15 rows containing missing v
## (geom_point).
```

Scatterplot  
Area Vs Population



# Páginas de referencia

- Advanced Plots with ggplot [enlace()] <https://uc-r.github.io/ggplot>
- Top 50 ggplot2 Visualizations - The Master List (With Full R Code) [enlace](#)
- Plot Time Series Data Using GGPlot [enlace] (<http://www.sthda.com/english/articles/32-r-graphics-essentials/128-plot-time-series-data-using-ggplot/>)

# Ejercicios

Valencia Bayesian Research group

# 1,2,3.. probando

Tienes que crear un documento Word en el personalices una tabla de un análisis descriptivo y varios gráficos diferentes utilizando la sintaxis de los ggplot2.

+ Base de datos propuesta: `economics`

*date: Month of data collection*

*psavert: personal savings rate*

*pce: personal consumption expenditures, in billions of dollars,*

*unemploy: number of unemployed in thousands,*

*uempmed: median duration of unemployment, in weeks,*

*pop: total population, in thousands*