

# Introducción a R y Tidyverse

## Sesión 06

### Laboratorio de Innovación en Salud

2022-06-05

# Contenidos

- Introducción a Rmarkdown
  - Componentes de un Rmd: Encabezado YAML y cuerpo (texto y chunks)
  - Modificar texto
  - Insertar imágenes
- Introducción a visualización de datos con ggcharts.

# Exploración competencial

# Introducción a Rmarkdwon

R Markdown es la implementación de Markdown en R, el cual es un lenguaje de marcado ligero. Es decir, permite configurar formatos de texto en forma básica siendo fácil de editar y fácil de leer.

**R Markdown** (.Rmd) puede exportar a múltiples formatos y estos se incrementan aún más con el apoyo de otros paquetes: html, pdf, word, notebook, presentaciones (xaringan, ioslides, revealjs), dashboard, libros, páginas web, shiny apps, pósteres, etc.



# Primeros pasos

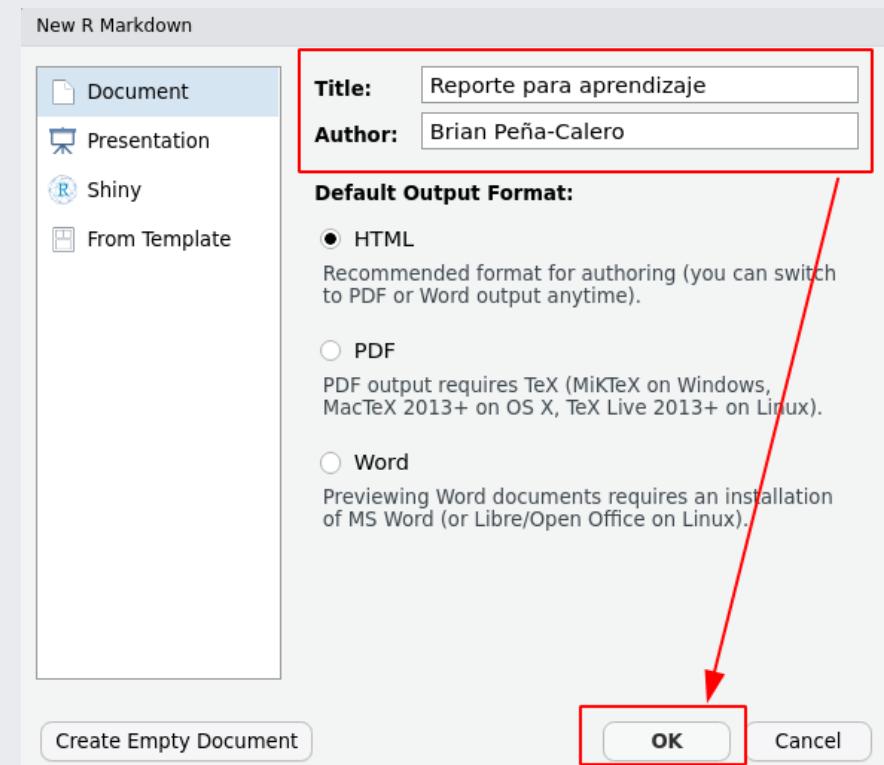
Necesitas tener instalado el paquete `rmarkdown` y posteriormente generar un archivo de esa extensión.

## Instalación del paquete

```
install.packages("rmarkdown")
```

## Generar un archivo Rmarkdown

1. Ir a File -> New File -> R Markdown
2. En la sección de **Document** indicar el título y nombre de autor (esto se puede modificar luego).
3. Guardar el archivo **.Rmd**
4. Con la opción de Knit se puede generar un archivo **html**.



# Encabezado YAML - Básico

El encabezado de un [Rmd](#) contiene básicamente metadatos del archivo, formato general del output y, en ocasiones, estilos personalizados (css), entre otros aspectos.

Este es el formato [YAML](#) básico con el que se inicia al generar un archivo [Rmd](#).

```
---
```

```
title: "Reporte de Proyecto de Análisis"
subtitle: "Introducción a R y Tidyverse"
date: 2022-06-05
output: html_document
---
```

El formato [YAML](#) puede contener más información y personalización. Esto dependerá de la configuración del output.

Algunos metadatos adicionales, que podrían usarse:

```
---
```

```
title: "Reporte de Proyecto de Análisis"
subtitle: "Introducción a R y Tidyverse"
author:
  - name: Brian N. Peña-Calero
    email: brianmsm@gmail.com
    affiliation: Laboratorio de Innovación en
date: 2022-06-05
output: html_document
---
```

# Cuerpo: Formato de texto

El formato de texto en Rmarkdown es bastante simple. Para títulos y subtítulos tenemos el uso de la **almohadilla/numeral/michi/#**:

```
# Título 1  
## Título 2  
### Título 3  
#### Título 4  
##### Título 5  
##### Título 6
```

**Título 1**  
**Título 2**  
**Título 3**  
**Título 4**  
**Título 5**  
**Título 6**

# Cuerpo: Formato de texto

Formato *cursiva*, **negrita**, ***cursiva-negrita***, código e hipervínculo

\*Este texto está en cursiva\*

*Este texto está en cursiva*

\*\*Ahora, este texto está en negrita\*\*

**Ahora, este texto está en negrita**

\*\*\*Este último contiene ambos: negrita-cursiva\*\*\*

***Este último contiene ambos: negrita-cursiva***

`Este texto será mostrado como código`

Este texto será mostrado como código

Esta data de `r nrow(mtcars)` casos ...

Esta data de 32 casos...

[Rmarkdown](<https://bookdown.org/yihui/rmarkdown>)

Rmarkdown

# Cuerpo: Insertar listas y viñetas

También es posible adicionar listas enumeradas y/o viñetas dentro de la redacción.

1. Primera numeración

1. Segunda numeración

- Primera viñeta

+ Una subviñeta viñeta

- Segunda viñeta

1. Tercera numeración

1. Primera numeración

2. Segunda numeración

◦ Primera viñeta

▪ Una subviñeta viñeta

◦ Segunda viñeta

3. Tercera numeración



# Cuerpo: Insertar código R

Se puede utilizar código R o de otro lenguaje soportado en una línea `r nrow(mtcars)` así como en un bloque (chunk) cuando se trata de análisis de varias líneas de código.

## Código:

```
```{r}
as_tibble(mtcars) %>%
  print(n = 3)
```

```

## Salida:

```
## # A tibble: 32 × 11
##   mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 21      6   160   110   3.9   2.62  16.5     0     1     4     4
## 2 21      6   160   110   3.9   2.88  17.0     0     1     4     4
## 3 22.8    4   108    93   3.85  2.32  18.6     1     1     4     1
## # ... with 29 more rows
```

# Cuerpo: Insertar código R

Los chunks pueden ser insertados mediante la combinación de teclas **Ctrl + alt + I** o mediante el botón de insertar.

Los chunks pueden contener nombres para ser referenciados posteriormente y/o configurar su ejecución/output dependiendo del contenido del código.

Por ejemplo:

```
```{r ejemplo, eval=FALSE}
tibble::as_tibble(mtcars) %>%
  print(n = 3)
```

```

```
tibble::as_tibble(mtcars) %>%
  print(n = 3)
```

Este chunk tiene un nombre llamado **ejemplo** y una configuración **eval=FALSE** que indica que el código no se evaluará (ejecutará), y solamente se mostrará. Puede ser útil para demostraciones y/o tutoriales.

# Cuerpo: Insertar código R

A continuación se muestran algunas configuraciones consideradas comunes para los chunks. Estas pueden combinarse entre sí (separadas por coma) en tanto tengan sentido. Existen alrededor de 60 configuraciones y dependen del contenido del código:

- `eval = FALSE`: No ejecuta código pero muestra el código.
- `include = FALSE`: Ejecuta código pero no muestra el resultado ni el código.
- `echo = FALSE`: Ejecuta código y muestra el output. Pero el código se mantiene oculto.
- `message = FALSE`: Evita que se muestren mensajes de ejecución.
- `warning = FALSE`: No muestra mensajes warnings (advertencia)
- `error = TRUE`: La compilación del documento continúa apesar de errores.

# Cuerpo: Insertar imágenes

Para insertar **imágenes**, existen 2 formas de hacerlo:

---

Código R      Imagen

---

## 1. Código CSS

```
{width=30%}
```

## 2. Código HTML

```

```

## 3. Código R

```
```{r, echo=FALSE, , out.width = '30%'}
knitr::include_graphics('https://rubenfcasal.github.io/bookdown_intro/images/rmarkdown.png',
  dpi = 300)
```
```

# Cuerpo: Insertar imágenes

Para insertar **imágenes**, existen 2 formas de hacerlo:

Código R

Imagen



# ¡Hazlo tú mismo!

1. Crear en un nuevo proyecto un archivo Rmarkdown
2. Completar los datos personales en **YAML** header.
3. Agregar algún texto descriptivo introductorio. Utiliza *cursivas*, **negritas**, **negritas y cursivas** y **código** para nombrar alguna función por ejemplo.

08 : 00

# Introducción a visualización de datos: ggcharts

En R el paquete por excelencia para poder realizar gráficos es `ggplot2`. Sin embargo, este en particular amerita un curso entero dedicado a su enseñanza total y en detalle. Por otro lado, `ggcharts` es un paquete que utiliza por debajo a `ggplot2` para generar gráficos rápidos y de forma sencilla.

Lo mejor de todo es que es completamente compatible con las funciones de `ggplot2`.

```
install.packages("ggcharts")
```



# Uso de ggcharts

Vamos a ejemplificar de forma sencilla cómo realizar un gráfico de barras usando la data `biomedicalrevenue` que muestra los ingresos anuales de las principales empresas biomédicas.

```
library(dplyr)
library(ggplot2)
library(ggcharts)
data("biomedicalrevenue")
biomedicalrevenue %>%
  as_tibble() %>%
  filter(year %in% c(2012, 2015, 2018))

## # A tibble: 84 × 3
##   company      year revenue
##   <chr>        <int>   <dbl>
## 1 Johnson & Johnson 2018     81.6
## 2 Johnson & Johnson 2015     70.1
```

# Uso de ggcharts

```
biomedicalrevenue %>%  
  as_tibble()
```

```
## # A tibble: 224 × 3  
##   company      year revenue  
##   <chr>        <int>   <dbl>  
## 1 Johnson & Johnson 2018     81.6  
## 2 Johnson & Johnson 2017     76.5  
## 3 Johnson & Johnson 2016     71.9  
## 4 Johnson & Johnson 2015     70.1  
## 5 Johnson & Johnson 2014     74.3  
## 6 Johnson & Johnson 2013     71.3  
## 7 Johnson & Johnson 2012     67.2  
## 8 Johnson & Johnson 2011     65.0  
## 9 Roche          2018     56.9  
## 10 Roche         2017     57.4  
## # ... with 214 more rows
```

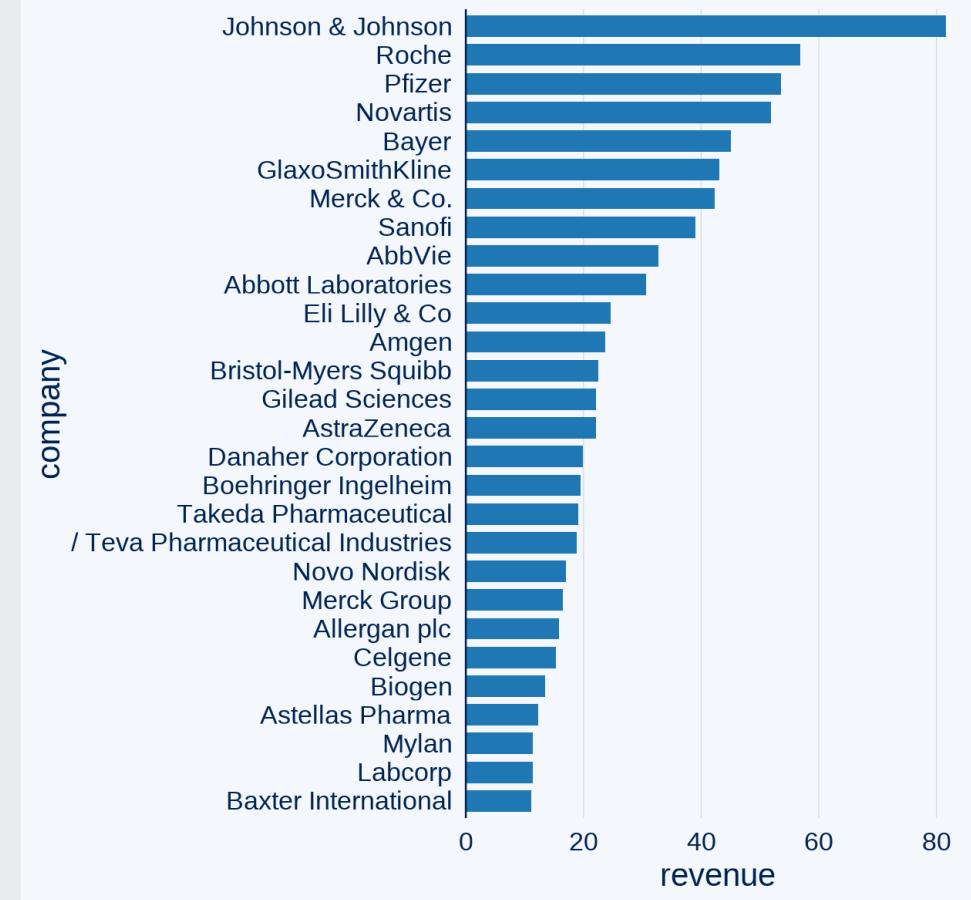
# Uso de ggcharts

```
biomedicalrevenue %>%  
  as_tibble() %>%  
  filter(year == 2018)
```

```
## # A tibble: 28 × 3  
##   company      year revenue  
##   <chr>        <int>   <dbl>  
## 1 Johnson & Johnson 2018     81.6  
## 2 Roche          2018     56.9  
## 3 Pfizer          2018     53.6  
## 4 Novartis        2018     51.9  
## 5 Bayer           2018     45.1  
## 6 GlaxoSmithKline 2018     43.1  
## 7 Merck & Co.      2018     42.3  
## 8 Sanofi          2018     39.1  
## 9 AbbVie          2018     32.8  
## 10 Abbott Laboratories 2018    30.6  
## # ... with 18 more rows
```

# Uso de ggcharts

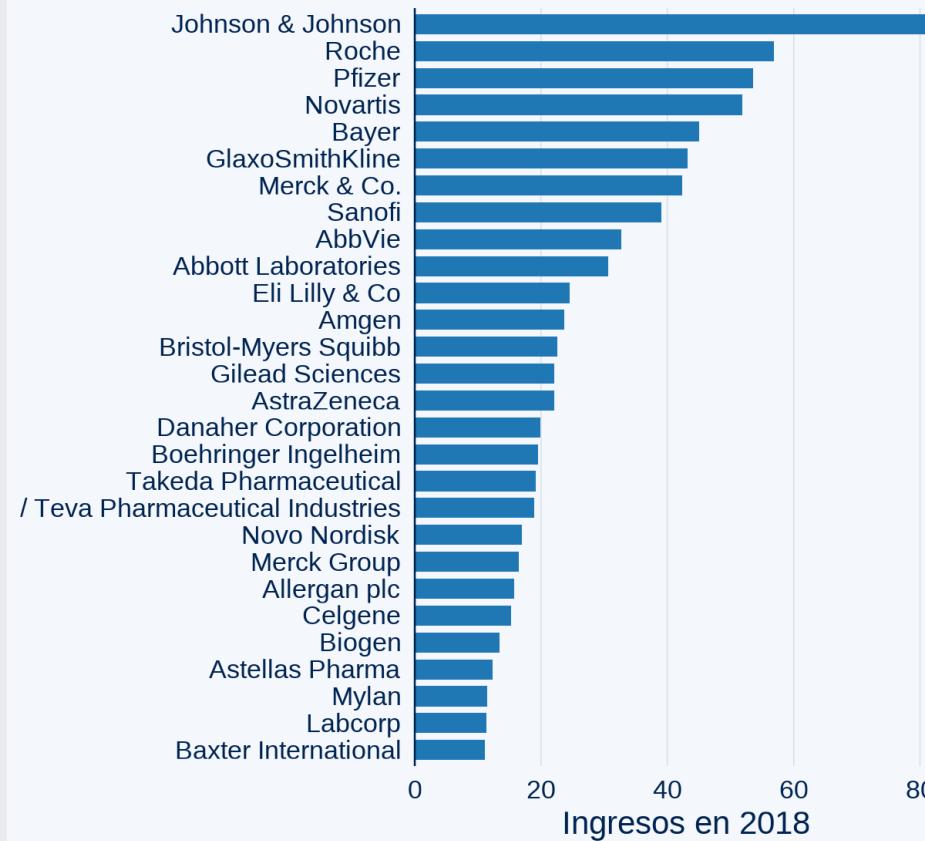
```
biomedicalrevenue %>%  
  as_tibble() %>%  
  filter(year == 2018) %>%  
  bar_chart(x = company, y = revenue)
```



# Uso de ggcharts

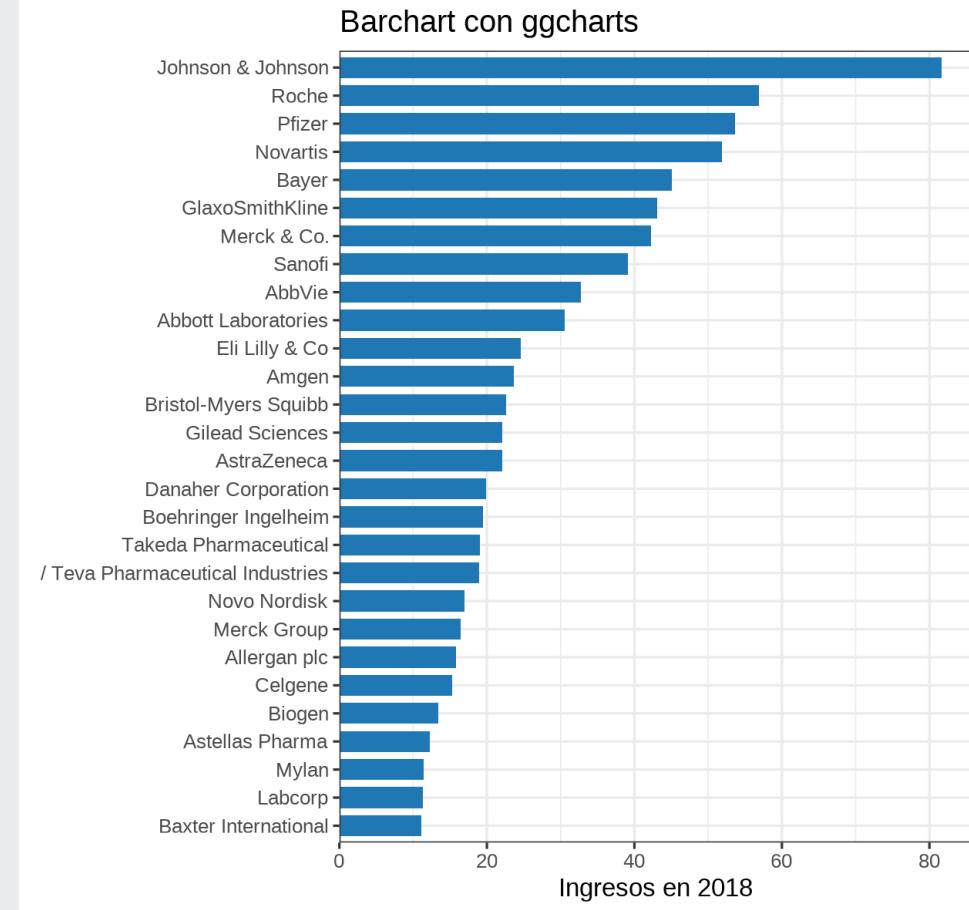
```
biomedicalrevenue %>%
  as_tibble() %>%
  filter(year == 2018) %>%
  bar_chart(x = company, y = revenue)
  labs(
    title = "Barchart con ggcharts",
    y = "Ingresos en 2018",
    x = NULL
  )
```

Barchart con ggcharts



# Uso de ggcharts

```
biomedicalrevenue %>%
  as_tibble() %>%
  filter(year == 2018) %>%
  bar_chart(x = company, y = revenue)
  labs(
    title = "Barchart con ggcharts",
    y = "Ingresos en 2018",
    x = NULL
  ) +
  theme_bw() # BREAK
```



# Uso de ggcharts

```
biomedicalrevenue %>%  
  as_tibble()
```

```
## # A tibble: 224 × 3  
##   company      year revenue  
##   <chr>        <int>   <dbl>  
## 1 Johnson & Johnson 2018     81.6  
## 2 Johnson & Johnson 2017     76.5  
## 3 Johnson & Johnson 2016     71.9  
## 4 Johnson & Johnson 2015     70.1  
## 5 Johnson & Johnson 2014     74.3  
## 6 Johnson & Johnson 2013     71.3  
## 7 Johnson & Johnson 2012     67.2  
## 8 Johnson & Johnson 2011     65.0  
## 9 Roche          2018     56.9  
## 10 Roche         2017     57.4  
## # ... with 214 more rows
```

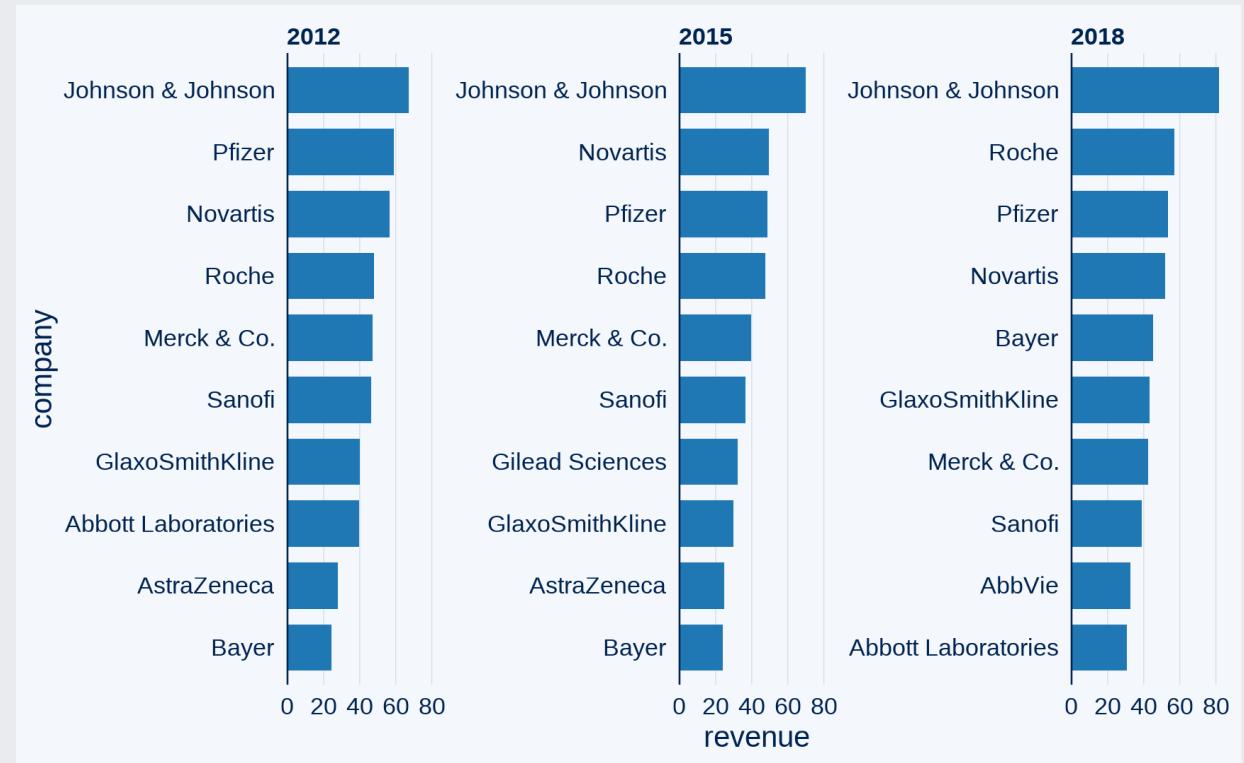
# Uso de ggcharts

```
biomedicalrevenue %>%  
  as_tibble() %>%  
  filter(year %in% c(2012, 2015, 2018))
```

| company                 | year | revenue |
|-------------------------|------|---------|
| Johnson & Johnson       | 2018 | 81.6    |
| Johnson & Johnson       | 2015 | 70.1    |
| Johnson & Johnson       | 2012 | 67.2    |
| Roche                   | 2018 | 56.9    |
| Roche                   | 2015 | 47.7    |
| Roche                   | 2012 | 47.8    |
| Pfizer                  | 2018 | 53.6    |
| Pfizer                  | 2015 | 48.8    |
| Pfizer                  | 2012 | 59.0    |
| Novartis                | 2018 | 51.9    |
| # ... with 74 more rows |      |         |

# Uso de ggcharts

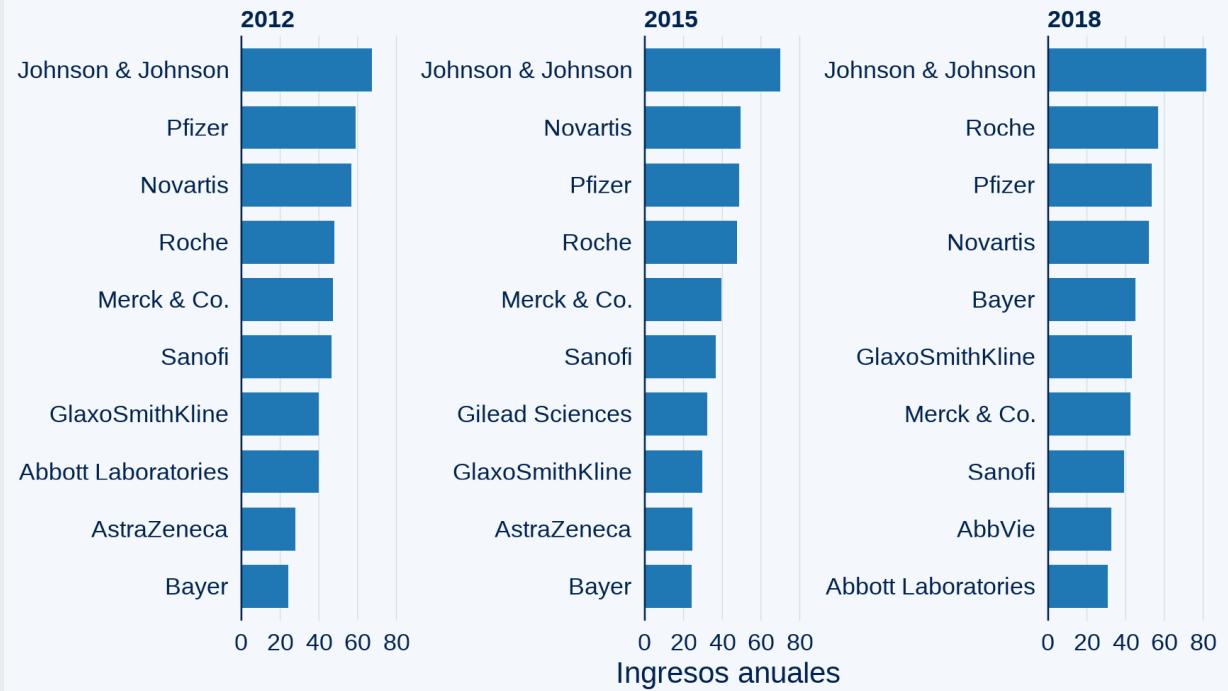
```
biomedicalrevenue %>%
  as_tibble() %>%
  filter(year %in% c(2012, 2015, 2018))
  bar_chart(x = company, y = revenue,
            facet = year, top_n = 10)
```



# Uso de ggcharts

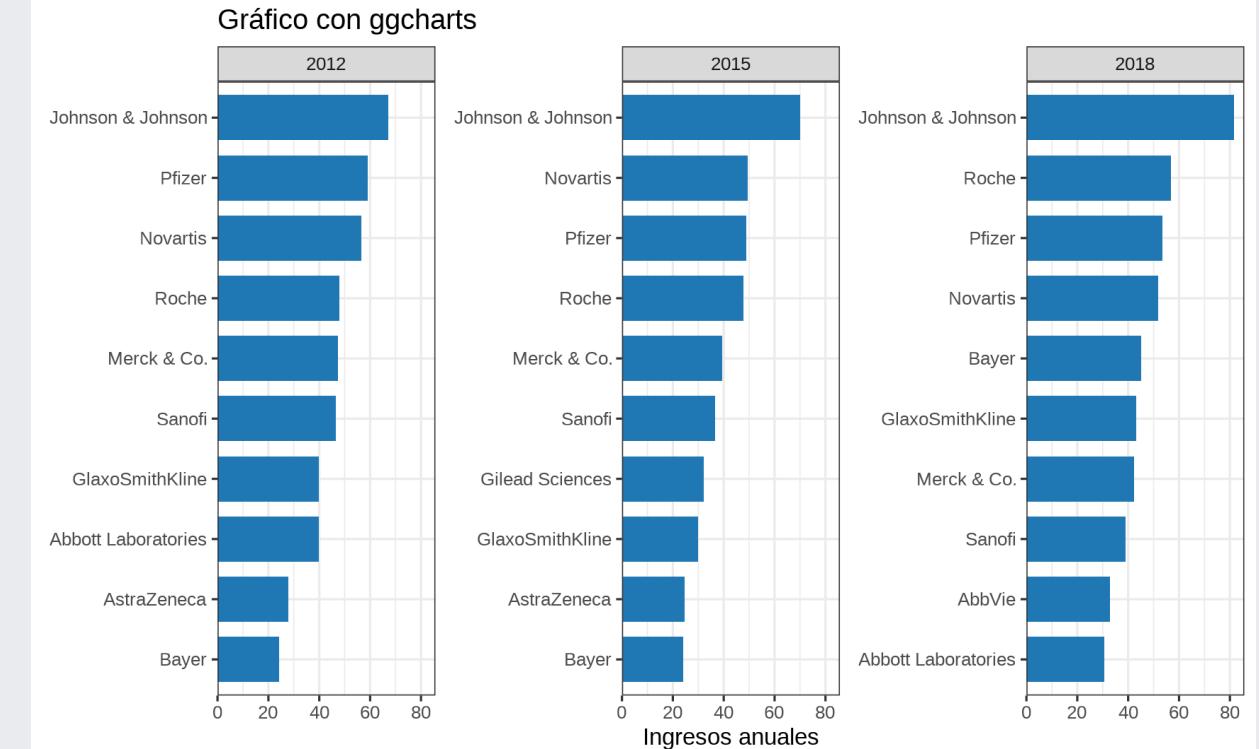
```
biomedicalrevenue %>%  
  as_tibble() %>%  
  filter(year %in% c(2012, 2015, 2018)  
  bar_chart(x = company, y = revenue,  
            facet = year, top_n = 10)  
  labs(  
    title = "Gráfico con ggcharts",  
    y = "Ingresos anuales",  
    x = NULL  
  )
```

Gráfico con ggcharts



# Uso de ggcharts

```
biomedicalrevenue %>%
  as_tibble() %>%
  filter(year %in% c(2012, 2015, 2018))
  bar_chart(x = company, y = revenue,
            facet = year, top_n = 10)
  labs(
    title = "Gráfico con ggcharts",
    y = "Ingresos anuales",
    x = NULL
  ) +
  theme_bw()
```



# Retroalimentación

# ¡Gracias!

✉ [imt.innovlab@oficinas-upch.pe](mailto:imt.innovlab@oficinas-upch.pe)

🗣 [@healthinnovation](https://twitter.com/@healthinnovation)

🐦 [@innovalab\\_imt](https://twitter.com/@innovalab_imt)

Estas diapositivas fueron creadas mediante el paquete `xaringan` y `xaringanthemer`.