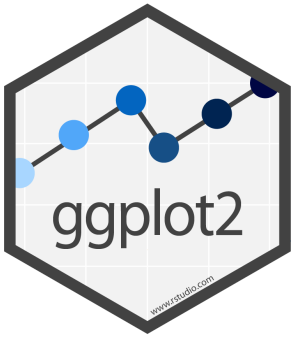
20 gráficos con 

Eduardo Guamán

2019-06-07

Who I am?

- Aprendiz de RStats
  - comencé a aprender con más intensidad aproximadamente hace 3 meses
  - **Fan absoluto de R Markdown y Tidyverse**
- Where I am?

@faestadistica

@guamandseduardo

@guamandseduardo

Instalar {ggplot2}

"Fichero R Markdown"

```
---
title: "Instalar {ggplot2}"
author: "Eduardo GuamÃ¡n"
output:
  pdf_document
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = FALSE, message = FALSE, warning =
  FALSE)
```


Texto

ggplot2 es un sistema para crear grÃ¡ficos de forma declarativa,
  basado en [The Grammar of Graphics]
  (https://www.amazon.com/Grammar-Graphics-Statistics-Computing-
  ebook-dp-B00HWUVHXK/dp/B00HWUVHXK/). Usted proporciona los
  datos, le dice a ggplot2 cÃ³mo mapear las variables en la
  estÃ©tica, quÃ© primitivas usar para grÃ¡ficar, y el paquete
  se encarga de los detalles.

InstalaciÃ³n.

```{r, eval=FALSE, echo=TRUE}
```
```

Data frame y librerías

```
library(tidyverse)
library(ggrepel)
library(plotly)
library(gganimate)
library(gapminder)
library(ggExtra)
library(ggcorrplot)
library(quantmod)
library(ggthemes)
df <- read_csv2("titanic3.csv")
```

El conjunto de datos titanic3 solo se utilizó para el gráfico 1 y el gráfico 20

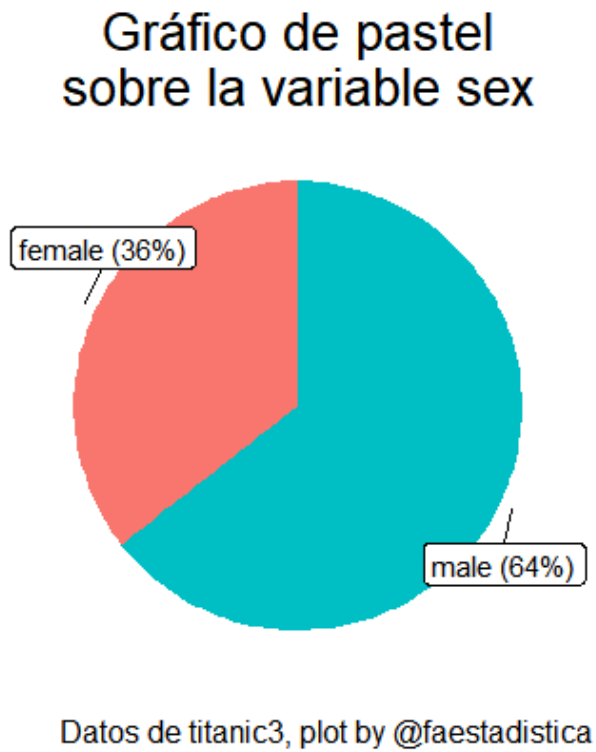
1: Gráfico de sectores 1

```
df <- read_csv2("titanic3.csv")
pies <- df %>%
  select(sex) %>%
  filter(!is.na(sex)) %>%
  group_by(sex) %>%
  tally(sort=T) %>%
  mutate(pie_cat = factor(c(sex[1:2]), levels=c(sex[1:2])),
         sex = factor(sex, levels=sex)) %>%
  group_by(pie_cat) %>%
  tally() %>%
  mutate(perc = round(n/sum(n)*100))
# Calcular pos para la posición de la etiqueta en el gráfico - inicio de cada segmento + tamaño de segme
# Esto coloca las etiquetas en el medio de cada segmento.
pies$pos = (cumsum(c(0, pies$n)) + c(pies$n/2, .01))[1:nrow(pies)]
```

1: Gráfico de sectores 1

```
ggplot(data=pies) +
  geom_col(aes(x=1, y=n, fill=fct_rev(pie_cat)), position="fill") +
  geom_label_repel(aes(x=1.5, y=pos/sum(pies$n), label=paste0(pie_cat, " (", perc, "%)")),
                   nudge_x = 0.5,
                   show.legend = FALSE) +
  coord_polar("y", start=0) +
  labs(title="Gráfico de pastel\nsobre la variable sex", caption="Datos de titanic3, plot by @faestadist
  theme_void() +
  theme(legend.position = "none",
        text=element_text(family="Roboto"),
        plot.title = element_text(size=20, hjust = 0.5),
        plot.caption = element_text(size = 12, hjust = 1),
        plot.margin = unit(c(0.5,0.5,0.5,1), "cm")
  )
```

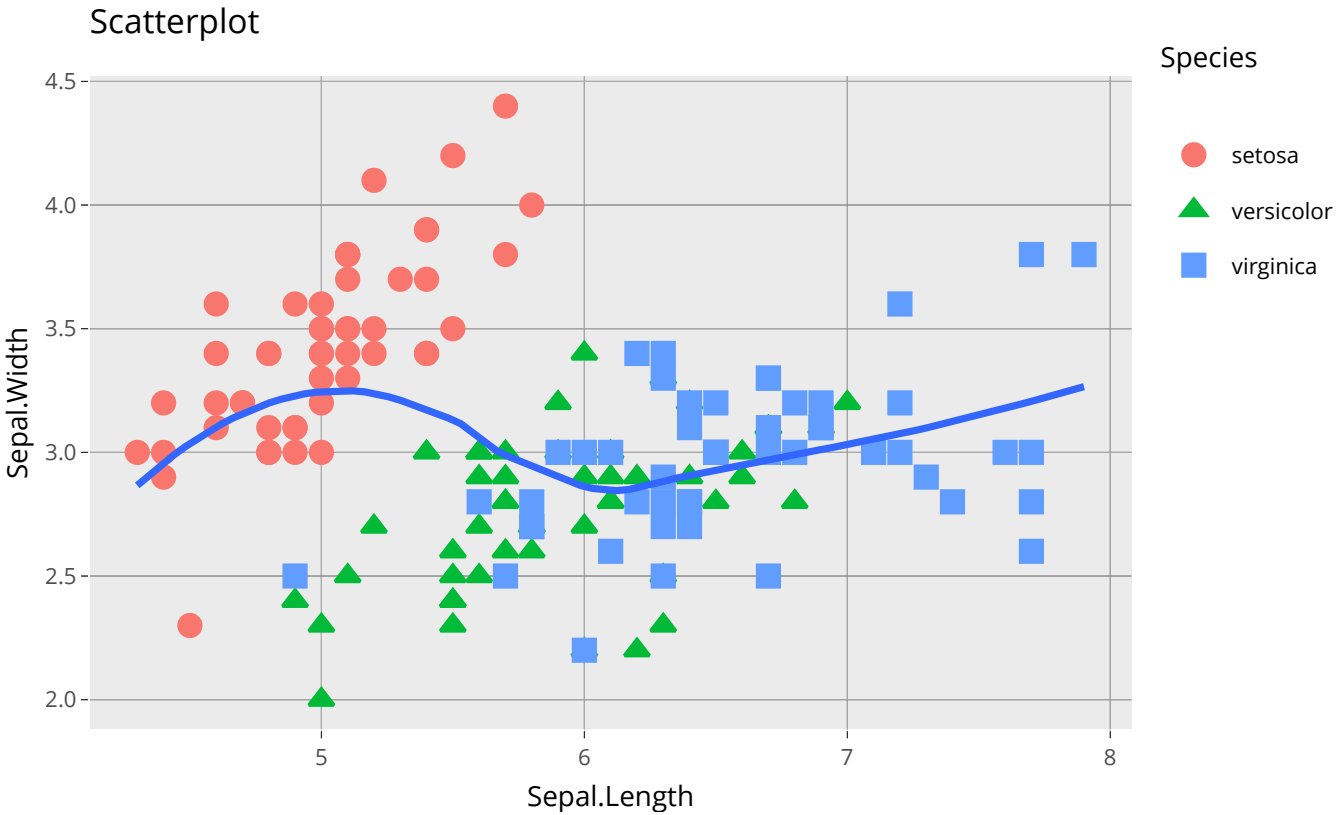
1: Gráfico de sectores 1



2: Gráfico de dispersión

```
p2 <- ggplot(iris, aes(Sepal.Length, Sepal.Width)) +
  geom_point(aes(col=Species, shape=Species), size=3) +
  geom_smooth(method="loess", se=F) +
  labs(subtitle="Sepal.Length Vs Sepal.Width",
       y="Sepal.Width",
       x="Sepal.Length",
       title="Scatterplot",
       caption = "plot by @faestadistica")
ggplotly(p2)
```

2: Gráfico de dispersión

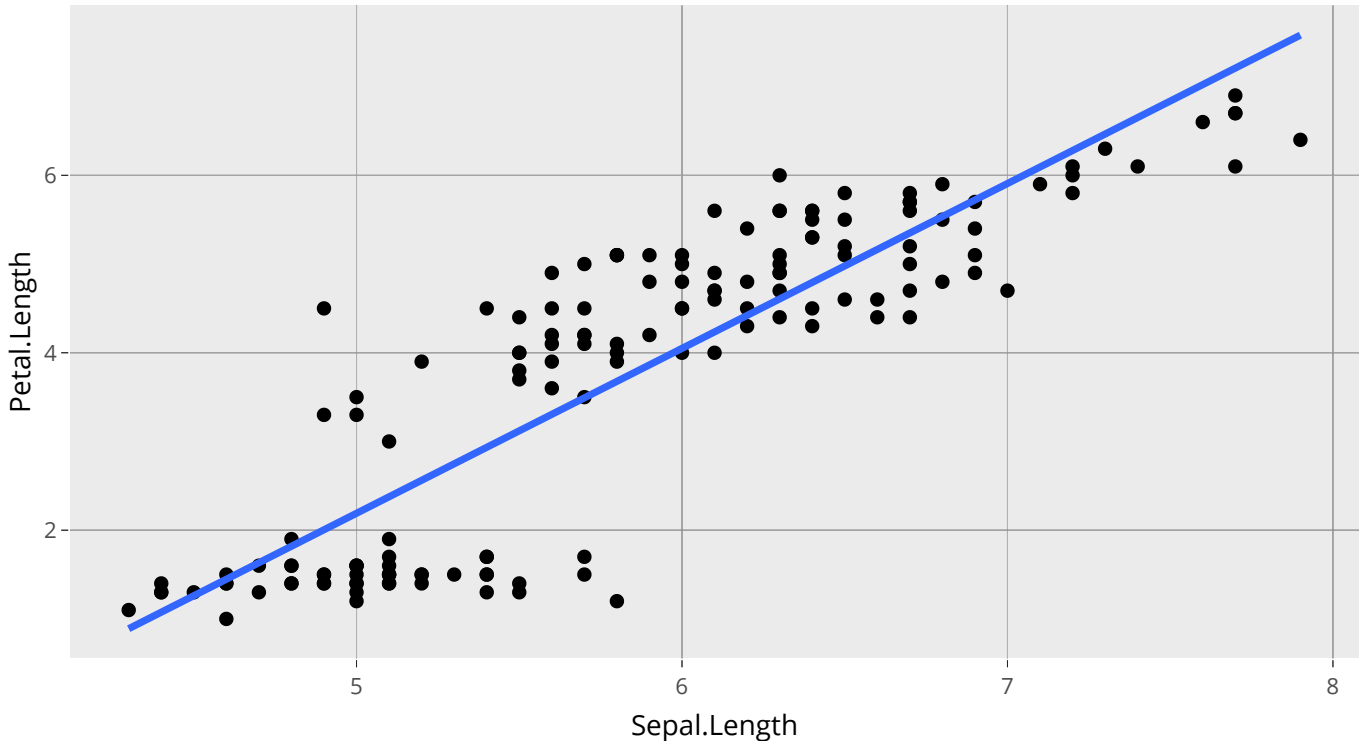


3: Scatterplot con puntos de solapamiento

```
p3<- ggplot(iris, aes(Sepal.Length, Petal.Length)) +
  geom_point() +
  geom_smooth(method="lm", se=F) +
  labs(subtitle="iris: Sepal.Length vs Petal.Length",
       y="Petal.Length",
       x="Sepal.Length",
       title="Diagrama de dispersión con puntos de solapamiento",
       caption="plot by @faestadistica")
ggplotly(p3)
```

3: Scatterplot con puntos de solapamiento

Diagrama de dispersión con puntos de solapamiento

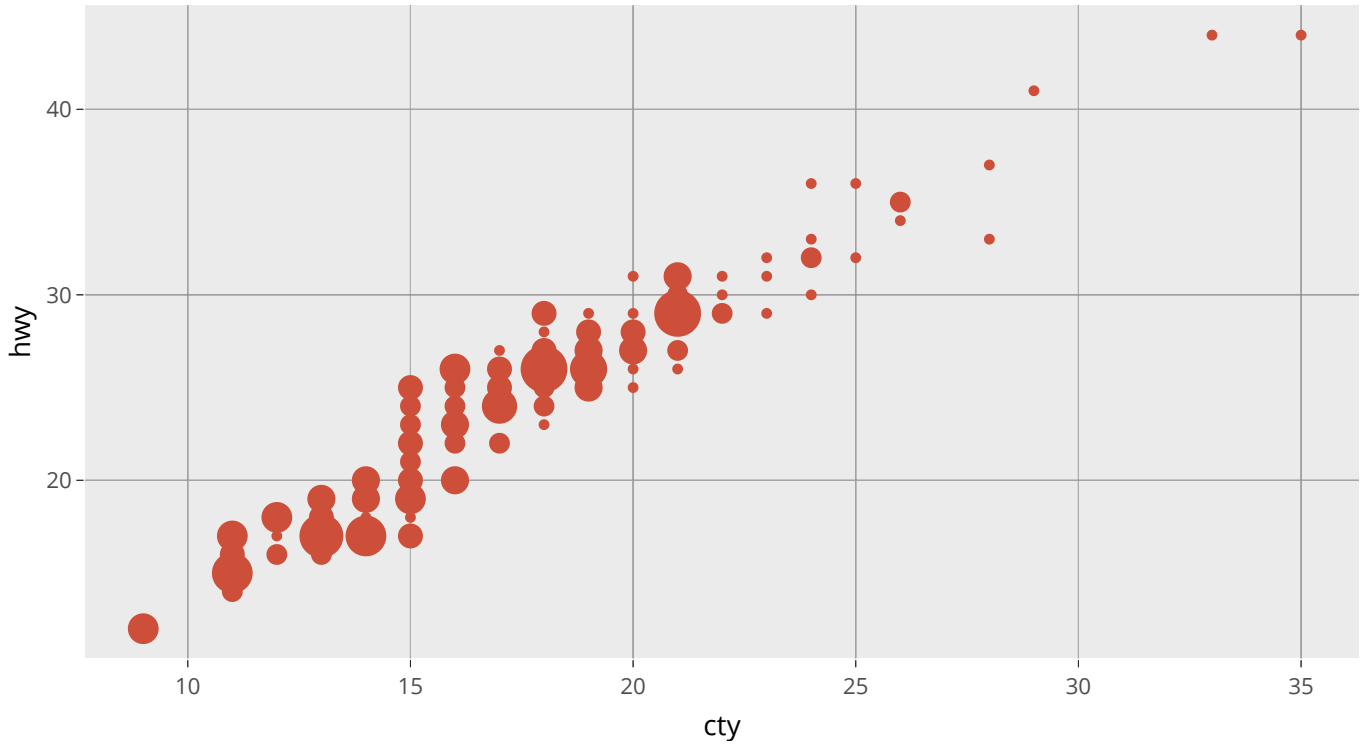


4: Gráfico de conteos

```
p4 <- ggplot(mpg, aes(cty, hwy)) +
  geom_count(col="tomato3", show.legend=F) +
  labs(subtitle="mpg: hwy vs cty",
       y="hwy",
       x="cty",
       title="Gráfico de conteo")
ggplotly(p4)
```

4: Gráfico de conteos

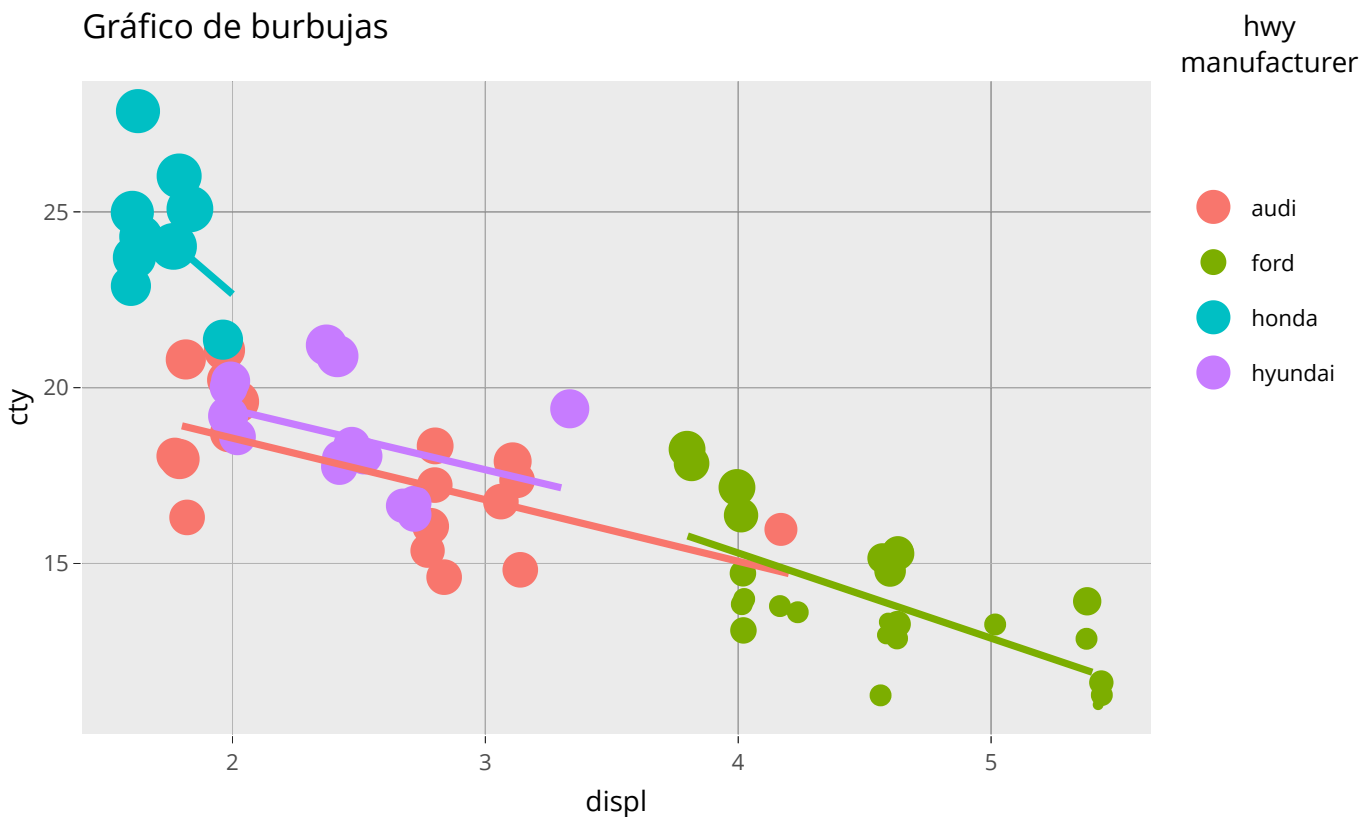
Gráfico de conteo



5: Gráfico de burbujas

```
mpg_select <- mpg[mpg$manufacturer %in% c("audi", "ford", "honda", "hyundai"), ]
p5 <- ggplot(mpg_select, aes(displ, cty)) +
  labs(subtitle="mpg: cty vs displ",
       title="Gráfico de burbujas")+
  geom_jitter(aes(col=manufacturer, size=hwy)) +
  geom_smooth(aes(col=manufacturer), method="lm", se=F)
ggplotly(p5)
```

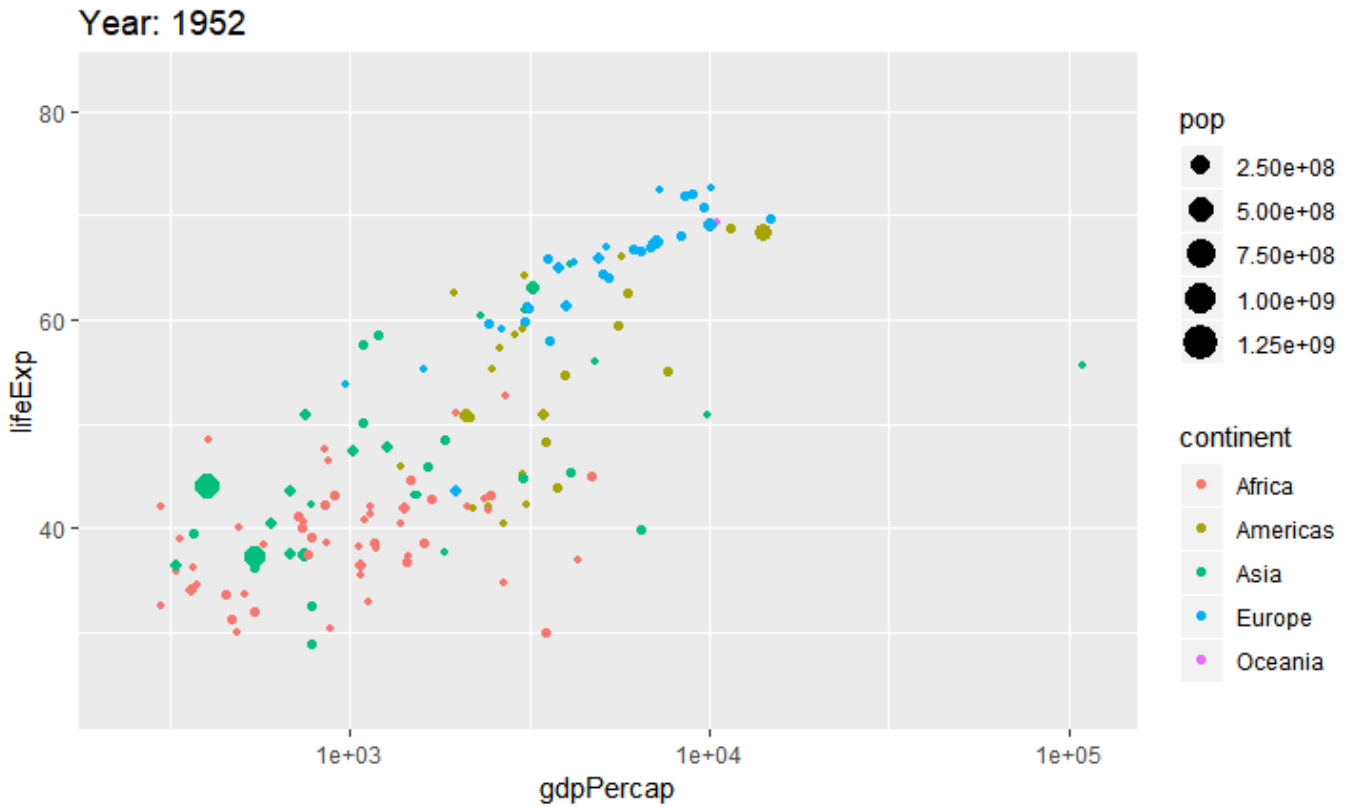
5: Gráfico de burbujas



6: Gráfico de burbujas animado {gganimate}

```
ggplot(gapminder, aes(gdpPercap, lifeExp, color = continent)) +
  geom_point(aes(size = pop, frame = year, ids = country)) +
  scale_x_log10() +
  labs(title = 'Year: {frame_time}', x = 'gdpPercap', y = 'lifeExp') +
  transition_time(year) +
  ease_aes('linear')
```

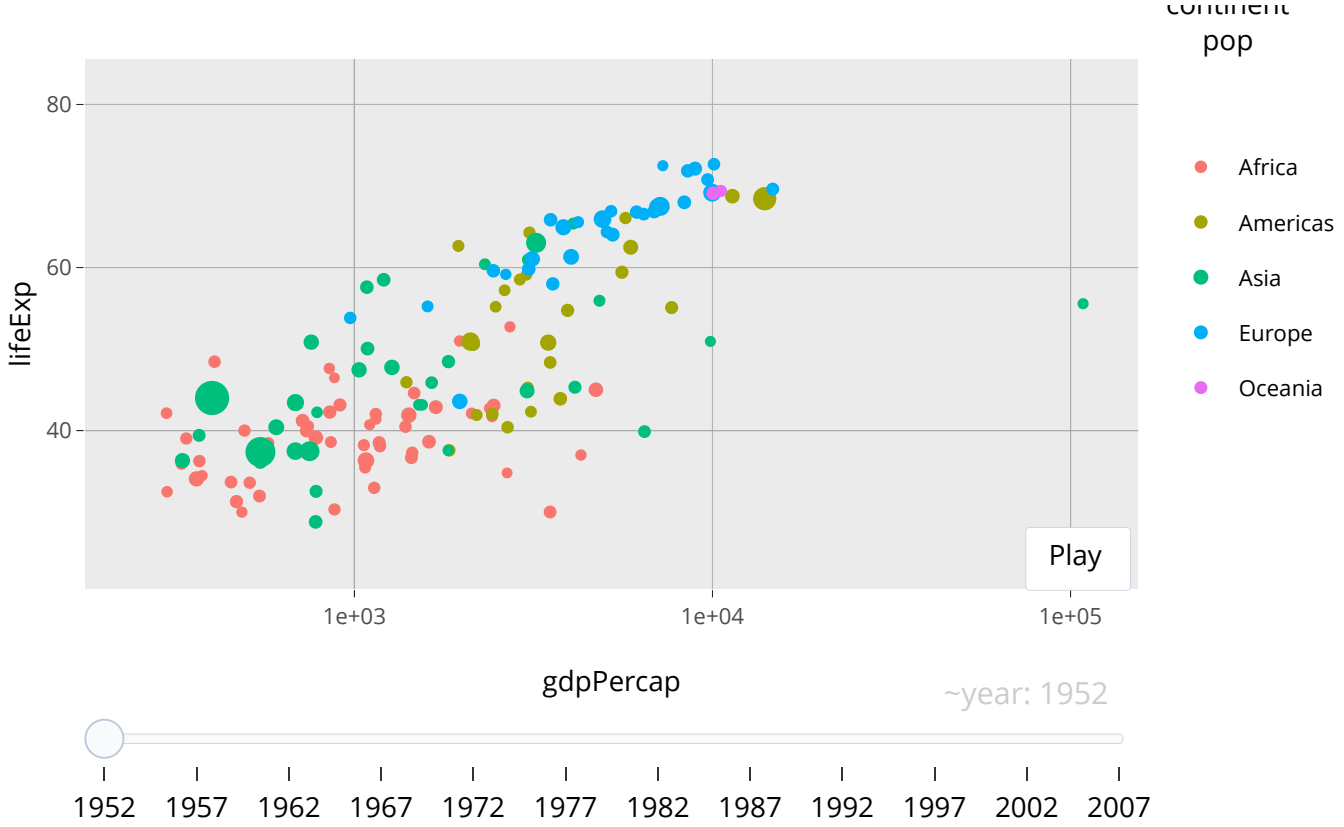
6: Gráfico de burbujas animado {gganimate}



7: Gráfico de burbujas animado {plotly}

```
p7 <- ggplot(gapminder, aes(gdpPercap, lifeExp, color = continent)) +
  geom_point(aes(size = pop, frame = year, ids = country)) +
  scale_x_log10()
animation_button(p7, x = 1, xanchor = "right", y = 0, yanchor = "bottom")
```

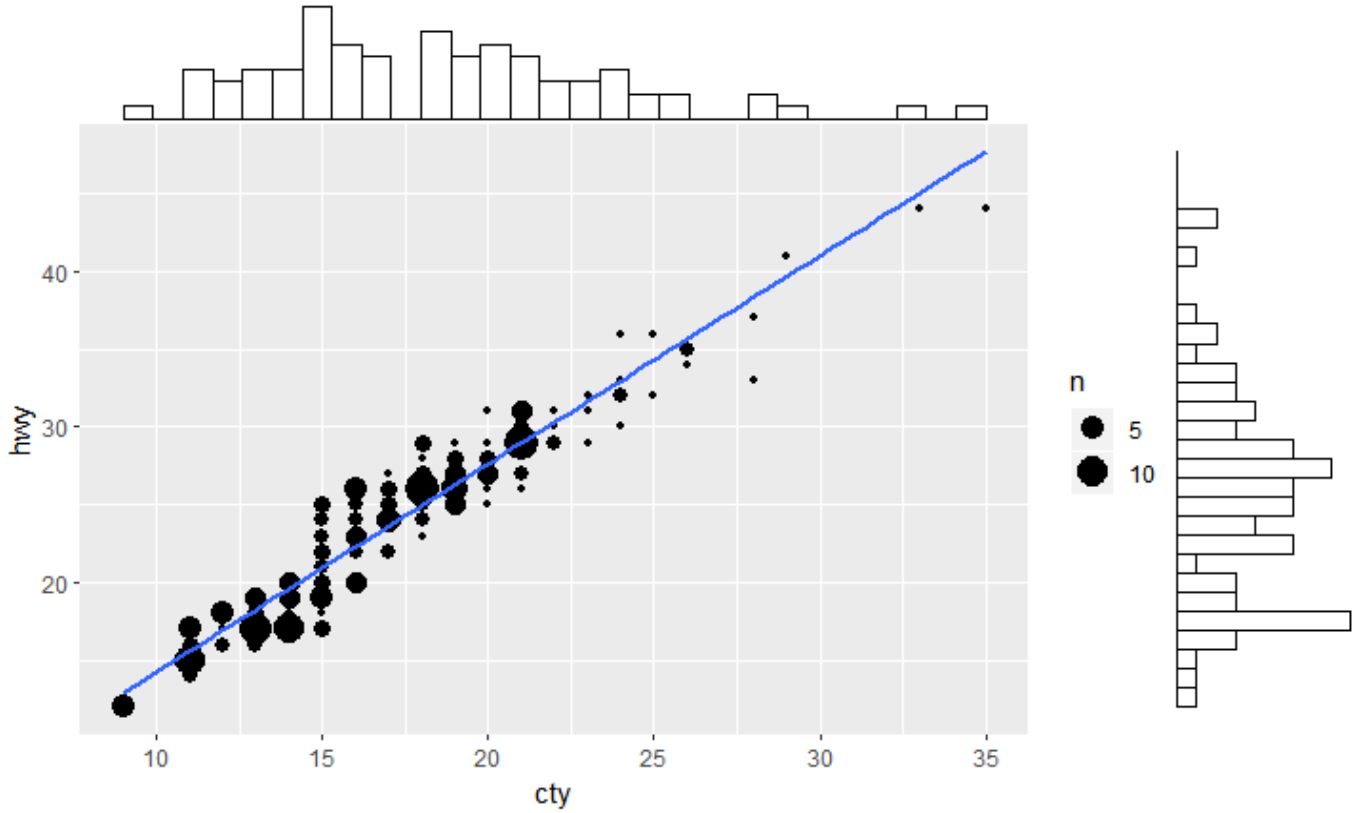
7: Gráfico de burbujas animado {plotly}



8: Histograma Marginal / Boxplot

```
p8 <- ggplot(mpg, aes(cty, hwy)) +
  geom_count() +
  geom_smooth(method="lm", se=F)
ggMarginal(p8, type = "histogram", fill="transparent")
# ggMarginal(p8, type = "boxplot", fill="transparent")
# ggMarginal(p8, type = "density", fill="transparent")
```

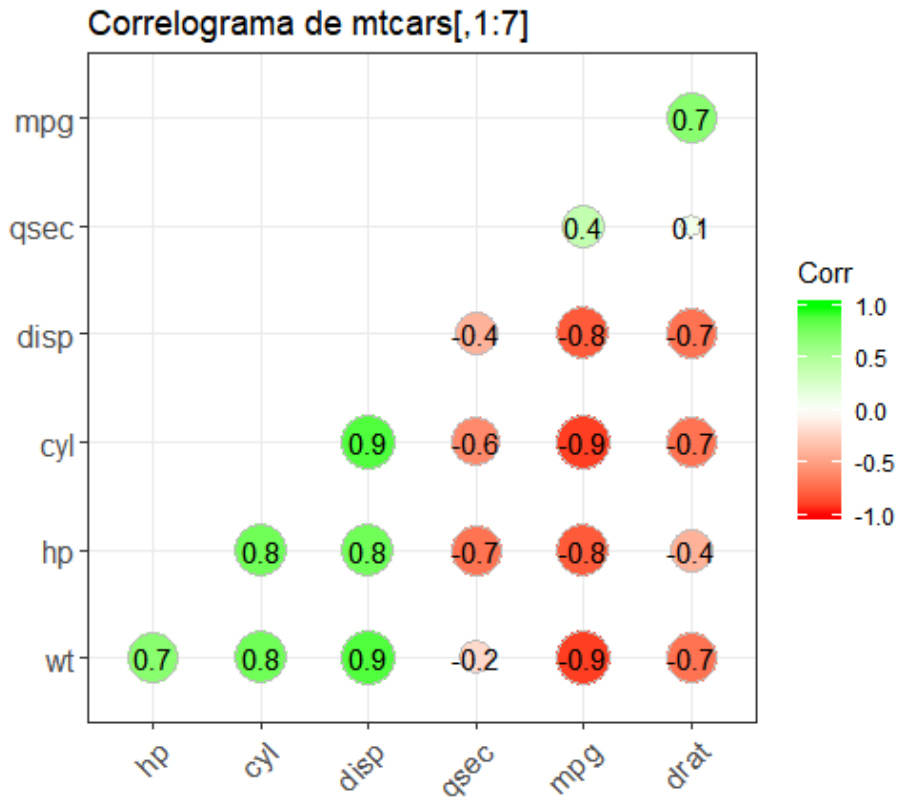
8: Histograma Marginal / Boxplot



9: Correlograma

```
corr <- round(cor(mtcars[,1:7]), 1)
ggcorrplot(corr, hc.order = TRUE,
           type = "lower",
           lab = TRUE,
           lab_size = 4,
           method="circle",
           colors = c("red", "white", "green"),
           title="Correlograma de mtcars[,1:7]",
           ggtheme=theme_bw)
```

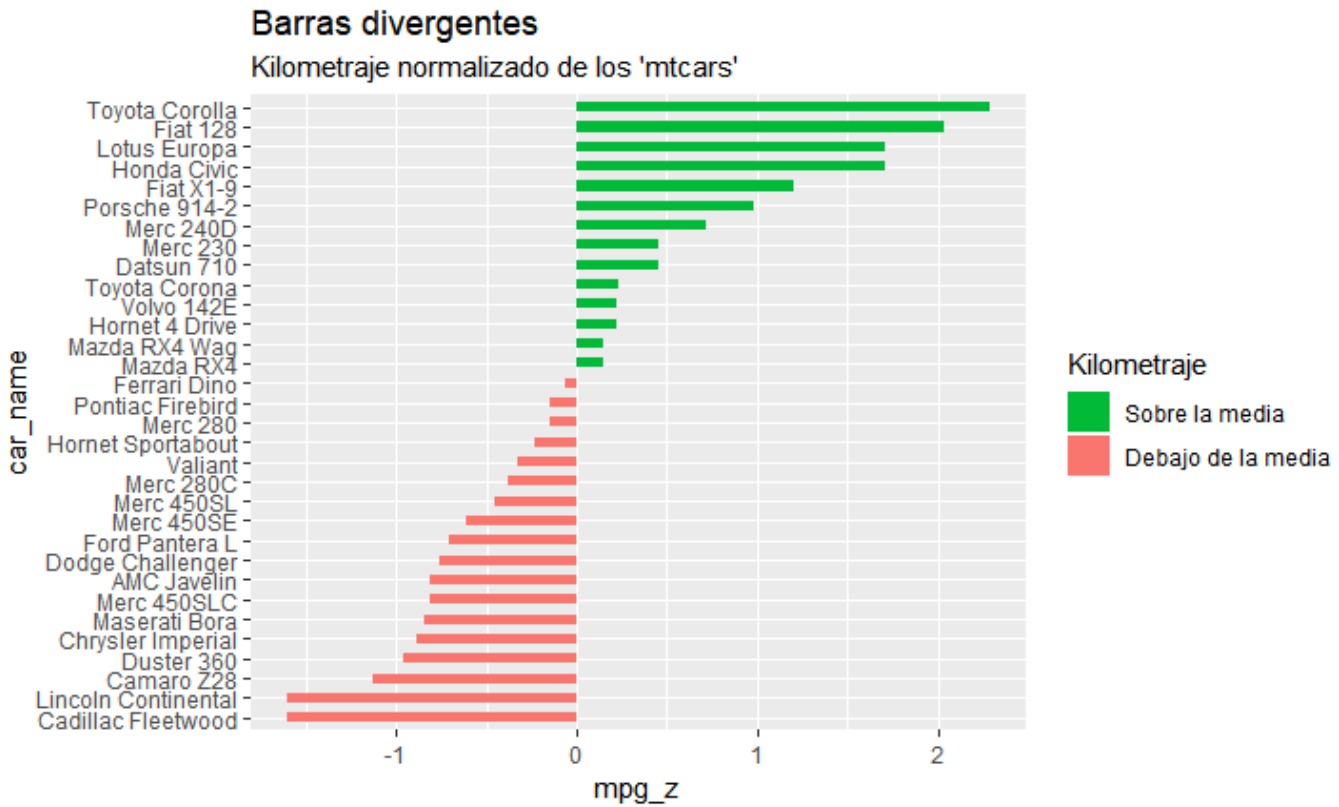
9: Correlograma



10: Barras divergentes

```
mtcars <- mtcars %>%
  mutate(car_name = rownames(mtcars),
         mpg_z = round((mpg - mean(mpg))/sd(mpg), 2),
         mpg_type = ifelse(mpg_z < 0, "below", "above")) %>%
  arrange(mpg_z)
mtcars$car_name <- factor(mtcars$car_name, levels = mtcars$car_name)
# Diverging Barcharts
ggplot(mtcars, aes(x=car_name, y=mpg_z, label=mpg_z)) +
  geom_bar(stat='identity', aes(fill=mpg_type), width=.5)  +
  scale_fill_manual(name="Kilometraje",
                    labels = c("Sobre la media", "Debajo de la media"),
                    values = c("above"="#00ba38", "below"="#f8766d")) +
  labs(subtitle="Kilometraje normalizado de los 'mtcars'",
       title= "Barras divergentes") +
  coord_flip()
```

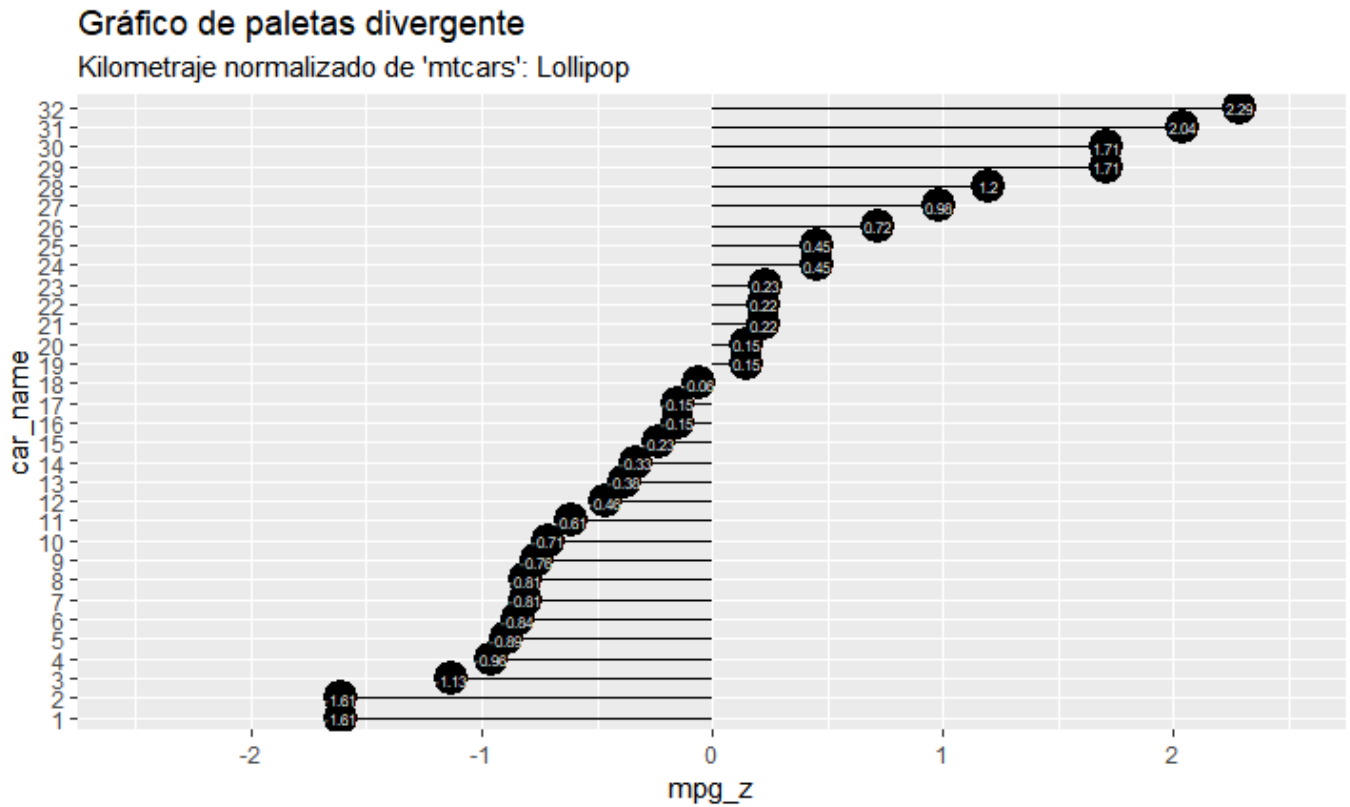
10: Barras divergentes



11: Gráfico de paletas divergentes

```
mtcars <- mtcars %>%
  mutate(car_name = rownames(mtcars),
         mpg_z = round((mpg - mean(mpg))/sd(mpg), 2),
         mpg_type = ifelse(mpg_z < 0, "below", "above")) %>%
  arrange(mpg_z)
mtcars$car_name <- factor(mtcars$car_name, levels = mtcars$car_name)
# Diverging Lollipop Chart
ggplot(mtcars, aes(x=car_name, y=mpg_z, label=mpg_z)) +
  geom_point(stat='identity', fill="black", size=6)  +
  geom_segment(aes(y = 0, x = car_name,
                   yend = mpg_z, xend = car_name),
               color = "black") +
  geom_text(color="white", size=2) +
  labs(title="Gráfico de paletas divergente",
       subtitle="Kilometraje normalizado de 'mtcars': Lollipop") +
  ylim(-2.5, 2.5) +
  coord_flip()
```

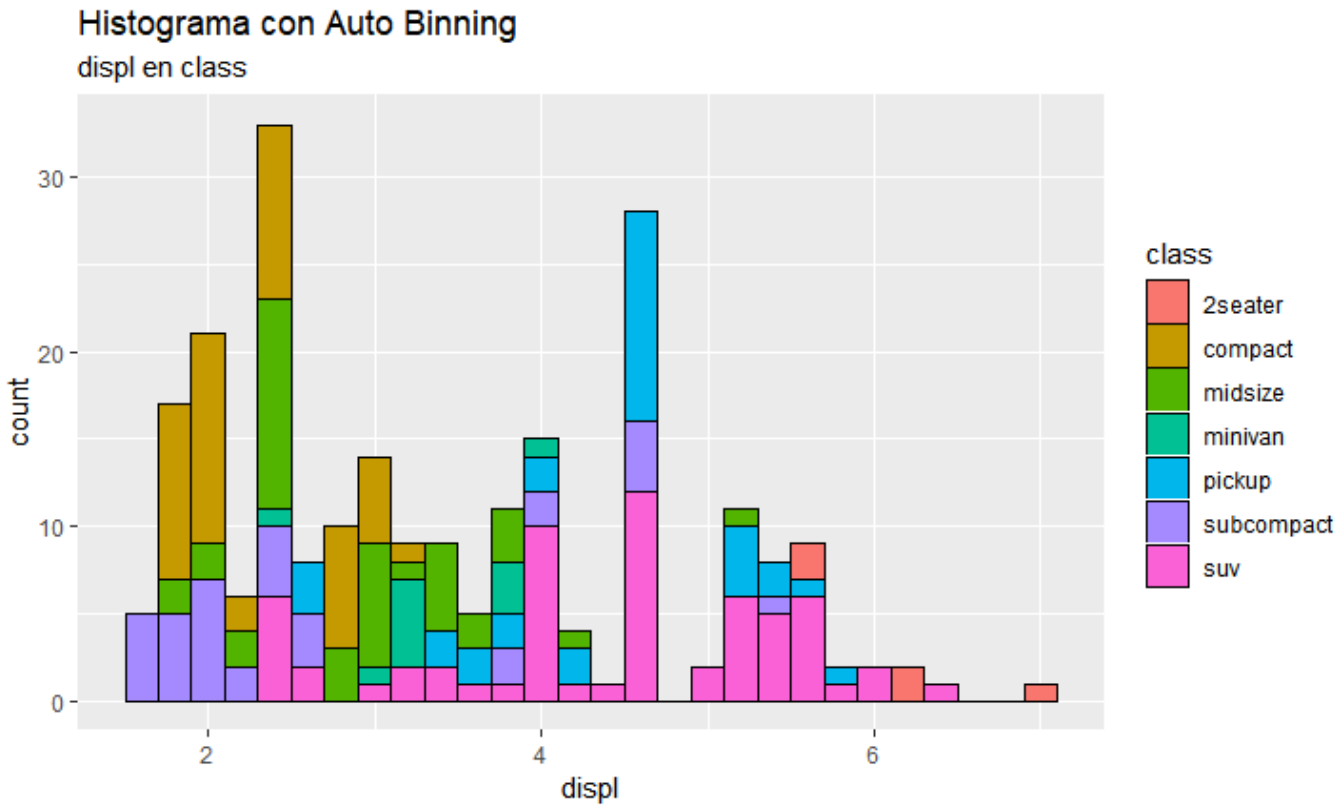
11: Gráfico de paletas divergentes



12: Histograma

```
# Histograma para una variable continua (numérica)
## Auto Binning
ggplot(mpg, aes(displ)) +
  geom_histogram(aes(fill=class), binwidth = .2, col="black", size=.1) +
  labs(title="Histograma con Auto Binning",
       subtitle="displ en class")
## Fixed Bins
ggplot(mpg, aes(displ)) +
  geom_histogram(aes(fill=class), bins=5, col="black", size=.1) +
  labs(title="Histograma con Fixed Bins",
       subtitle="displ en class de vehículos")
```

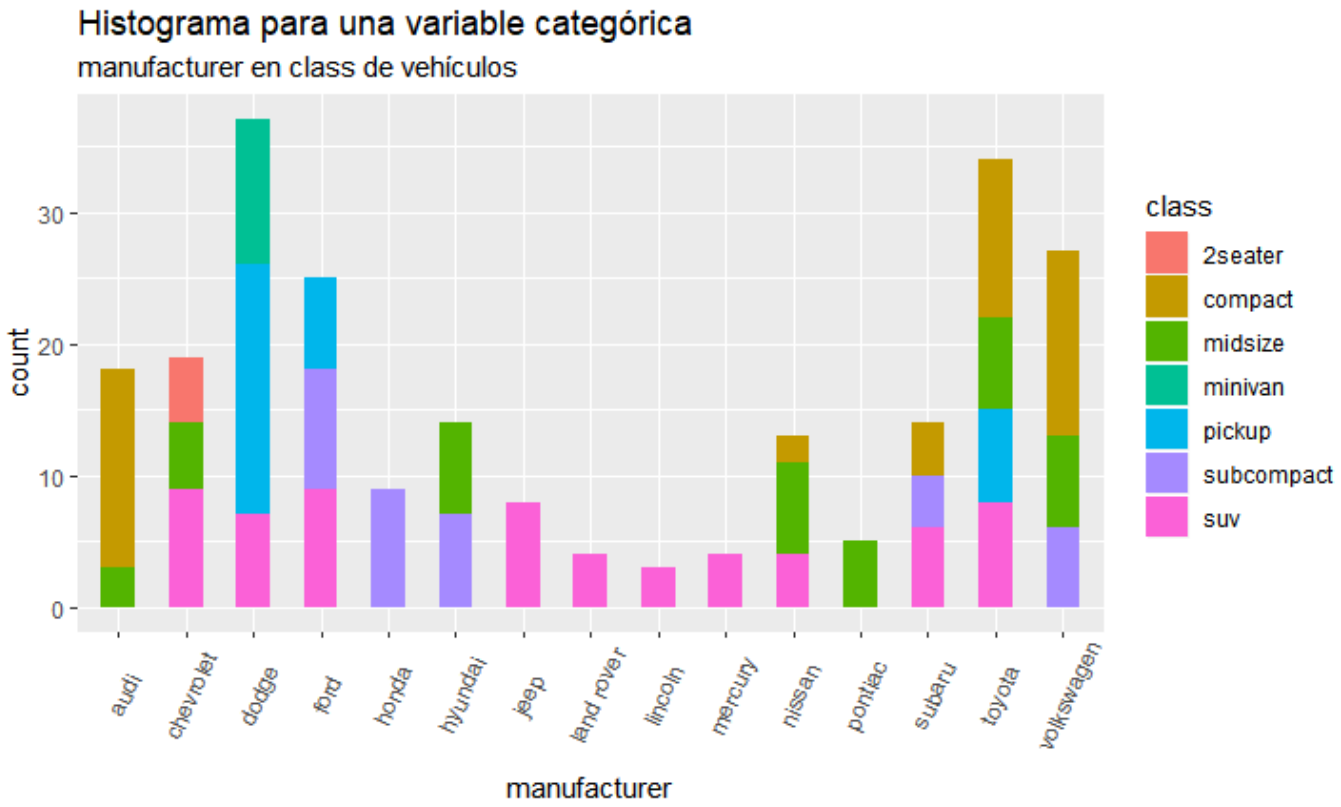
12: Histograma



13: Barras

```
# Gráfico de barras para una variable categórica
ggplot(mpg, aes(manufacturer)) +
  geom_bar(aes(fill=class), width = 0.5) +
  theme(axis.text.x = element_text(angle=65, vjust=0.6)) +
  labs(title="Histograma para una variable categórica",
       subtitle="manufacturer en class de vehículos")
# Gráfico de barras para una variable categórica
ggplot(mpg, aes(class, fill = drv)) +
  geom_bar(aes(y = ..count..*100/sum(..count..))) +
  labs(title="Barras apiladas para una variable categórica",
       subtitle="class en drv de vehículos")
# Gráfico de barras para una variable numérica y categórica
ggplot(mpg, aes(class, displ, fill = drv)) +
  geom_bar(stat = "identity") +
  labs(title="Barras apiladas para una variable numérica y categórica",
       subtitle="class, displ en drv de vehículos")
```

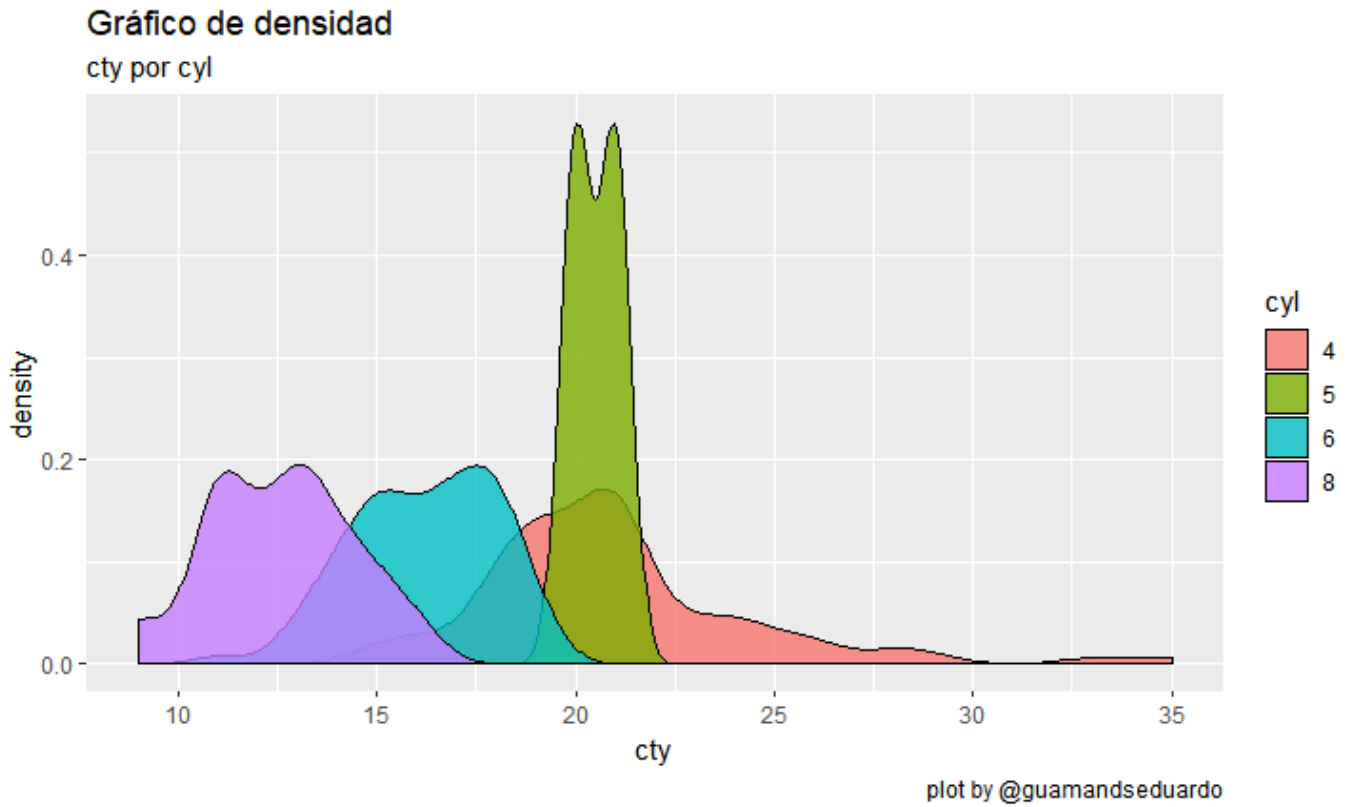
13: Barras


Histograma para una variable categórica
manufacturer en class de vehículos
count
30
20
10
0
audi
chevrolet
dodge
ford
honda
hyundai
jeep
land rover
lincoln
mercury
nissan
pontiac
subaru
toyota
volkswagen
manufacturer
class
2seater
compact
midsize
minivan
pickup
subcompact
suv

14: Gráfico de densidad

```
ggplot(mpg, aes(cty)) +
  geom_density(aes(fill=factor(cyl)), alpha=0.8) +
  labs(title="Gráfico de densidad",
       subtitle="cty por cyl",
       caption="plot by @guamandseduardo",
       x="cty",
       fill="cyl")
```

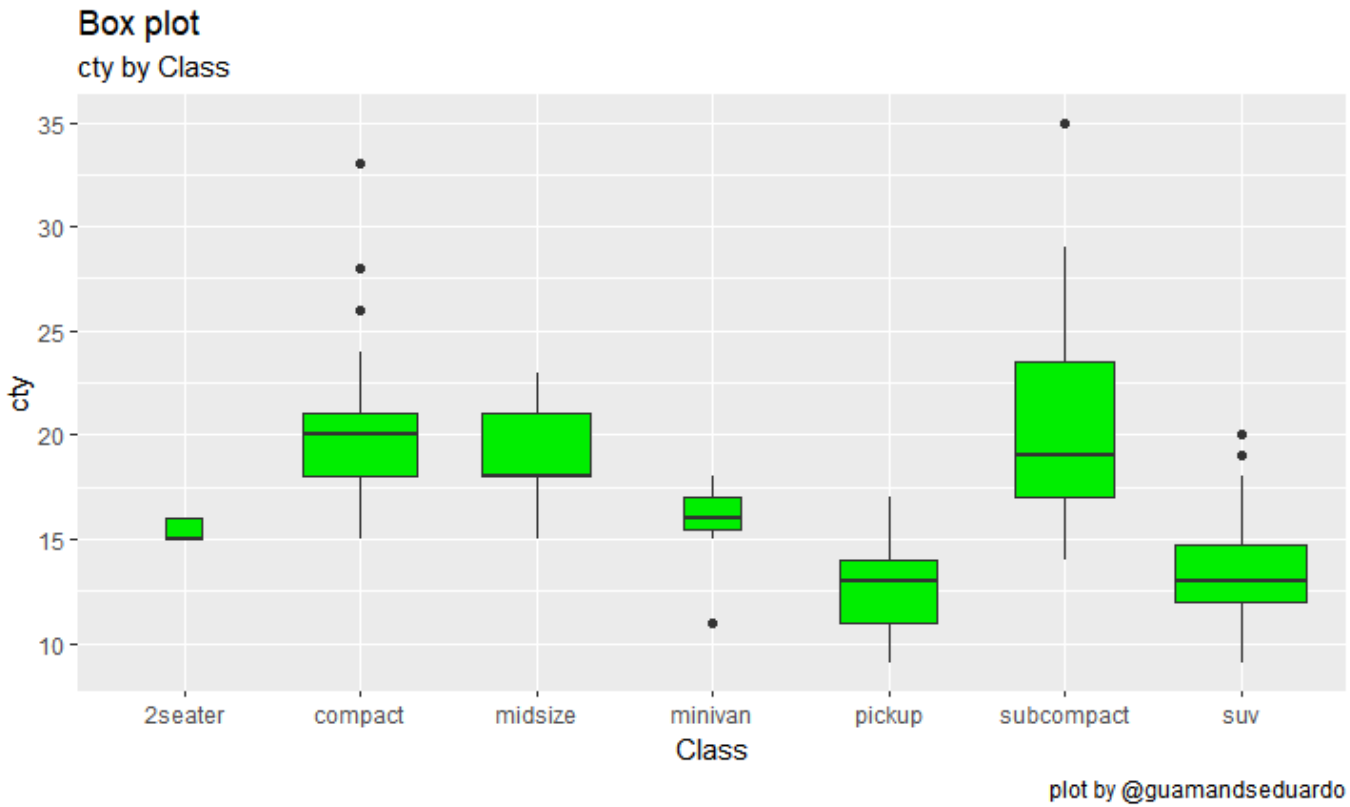
14: Gráfico de densidad



15: Box plot

```
ggplot(mpg, aes(class, cty)) +
  geom_boxplot(varwidth=T, fill="green2") +
  labs(title="Box plot",
       subtitle="cty by Class",
       caption="plot by @guamandseduardo",
       x="Class",
       y="cty")
# Box plot agrupado por una variable categótica
ggplot(mpg, aes(class, cty)) +
  geom_boxplot(aes(fill=factor(cyl))) +
  theme(axis.text.x = element_text(angle=65, vjust=0.6)) +
  labs(title="Box plot",
       subtitle="cty agrupado por Class",
       caption="plot by @guamandseduardo",
       x="Class",
       y="City Mileage")
```

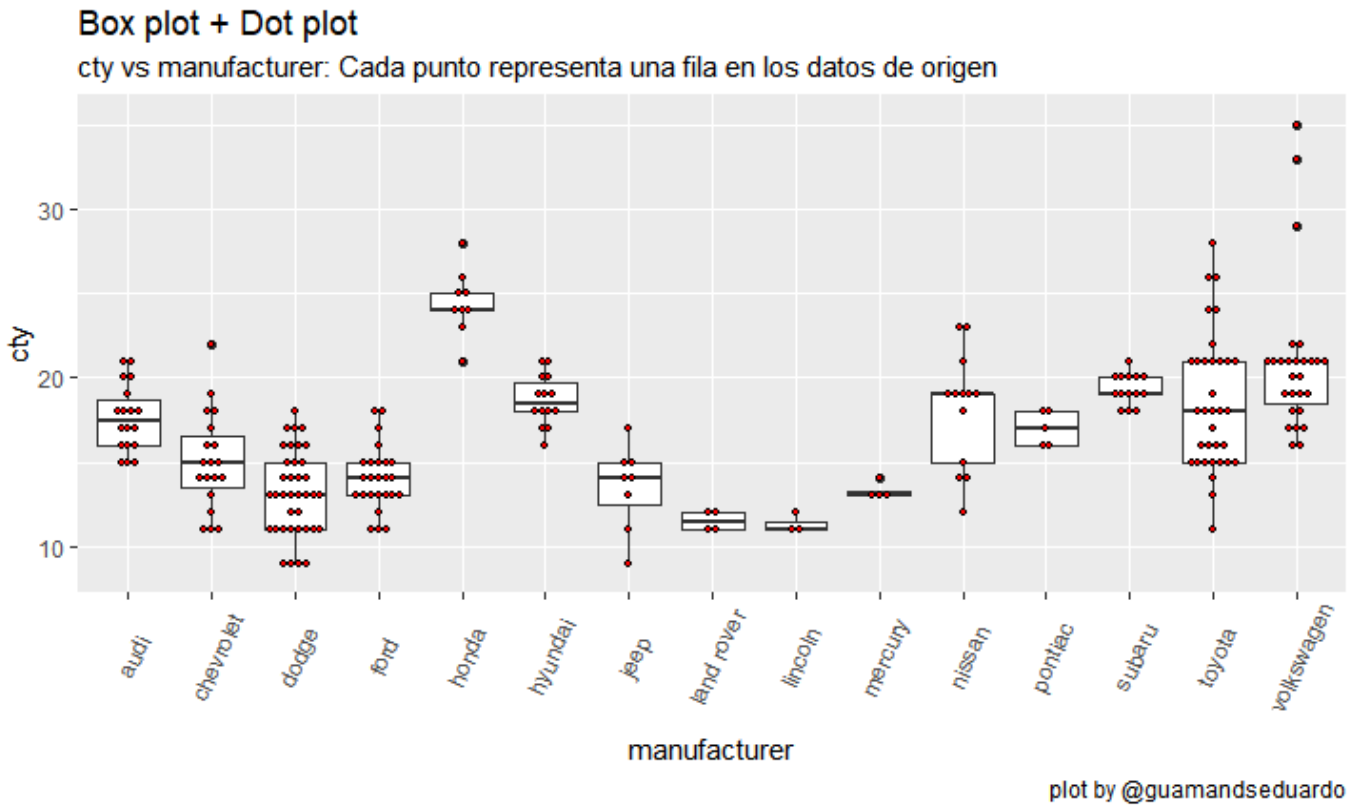
15: Box plot


Box plot
cty by Class
35
30
25
20
15
10
cty
2seater
compact
midsize
minivan
pickup
subcompact
suv
Class
plot by @guamandseduardo

15: Box plot + Dot plot

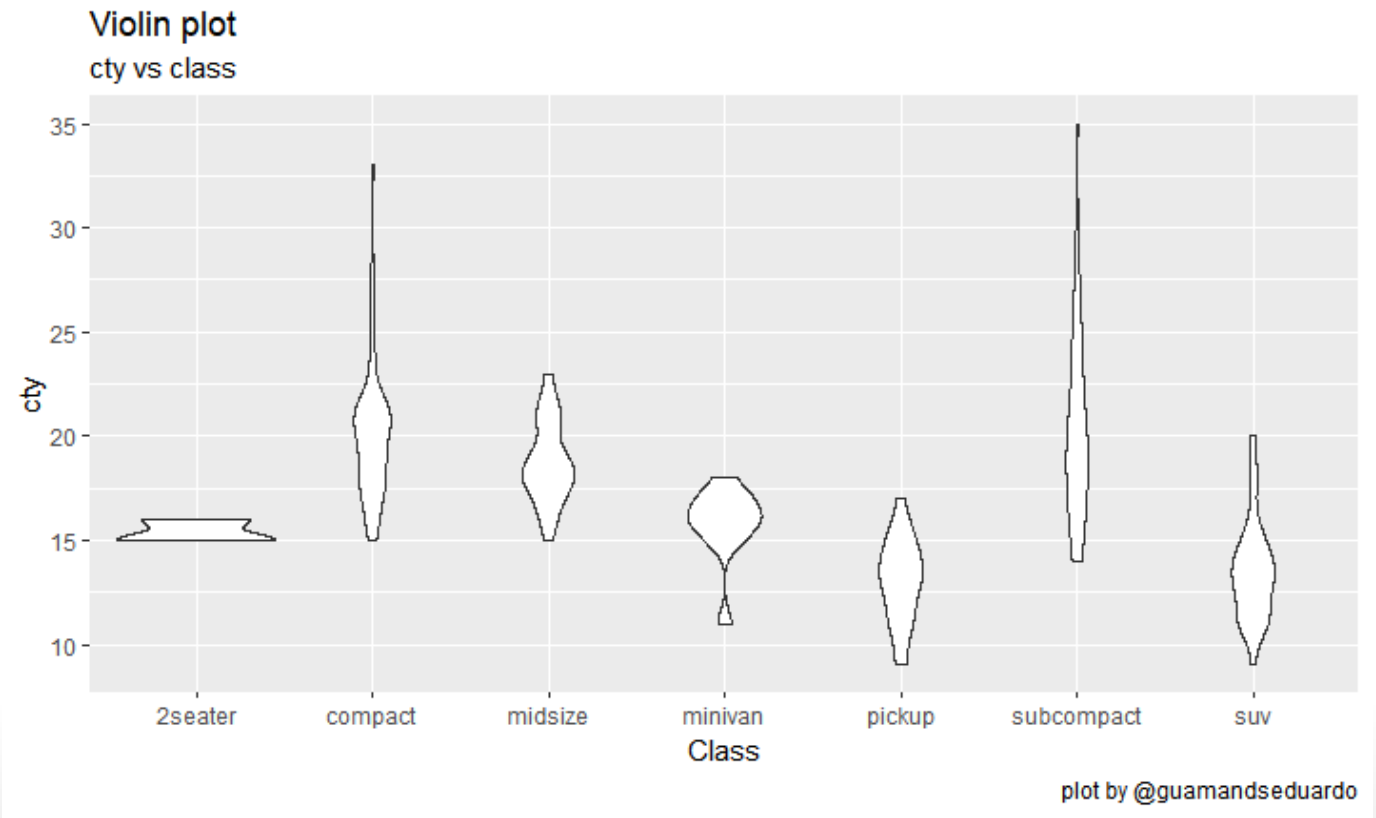
```
ggplot(mpg, aes(manufacturer, cty)) +
  geom_boxplot() +
  geom_dotplot(binaxis='y',
               stackdir='center',
               dotsize = .5,
               fill="red") +
  theme(axis.text.x = element_text(angle=65, vjust=0.6)) +
  labs(title="Box plot + Dot plot",
       subtitle="cty vs manufacturer: Cada punto representa una fila en los datos de origen",
       caption="plot by @guamandseduardo",
       x="manufacturer",
       y="cty")
```

15: Box plot + Dot plot



16: Gráfico de violín

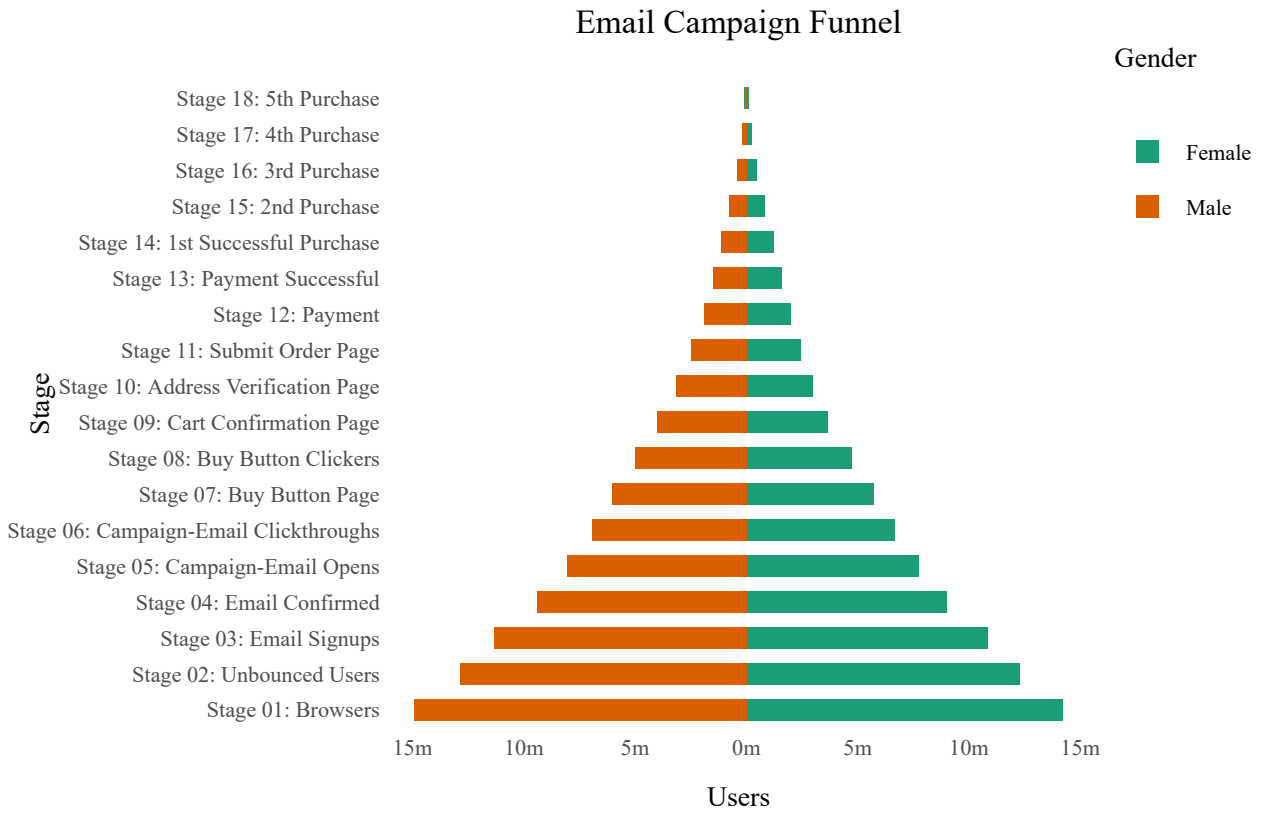
```
ggplot(mpg, aes(class, cty))+ geom_violin() +
  labs(title="Violin plot", subtitle="cty vs class", x="Class", y="cty",
       caption="plot by @guamandseduardo")
```



17: Pirámide poblacional

```
options(scipen = 999)  # notaciones científicas
email_campaign_funnel <- read.csv(
  "https://raw.githubusercontent.com/selva86/datasets/master/email_campaign_funnel.csv")
# Roturas y etiquetas del eje X
brks <- seq(-15000000, 15000000, 5000000)
lbls = paste0(as.character(c(seq(15, 0, -5), seq(5, 15, 5))), "m")
lbls1 = c(seq(15, 0, -5), seq(5, 15, 5))
g17 <- ggplot(email_campaign_funnel, aes(x = Stage, y = Users, fill = Gender)) +
  geom_bar(stat = "identity", width = .6) +
  scale_y_continuous(breaks = brks, labels = lbls) +
  coord_flip() +
  labs(title="Email Campaign Funnel") +
  theme_tufte() +
  theme(plot.title = element_text(hjust = .5), axis.ticks = element_blank()) +
  scale_fill_brewer(palette = "Dark2")
ggplotly(g17)
```

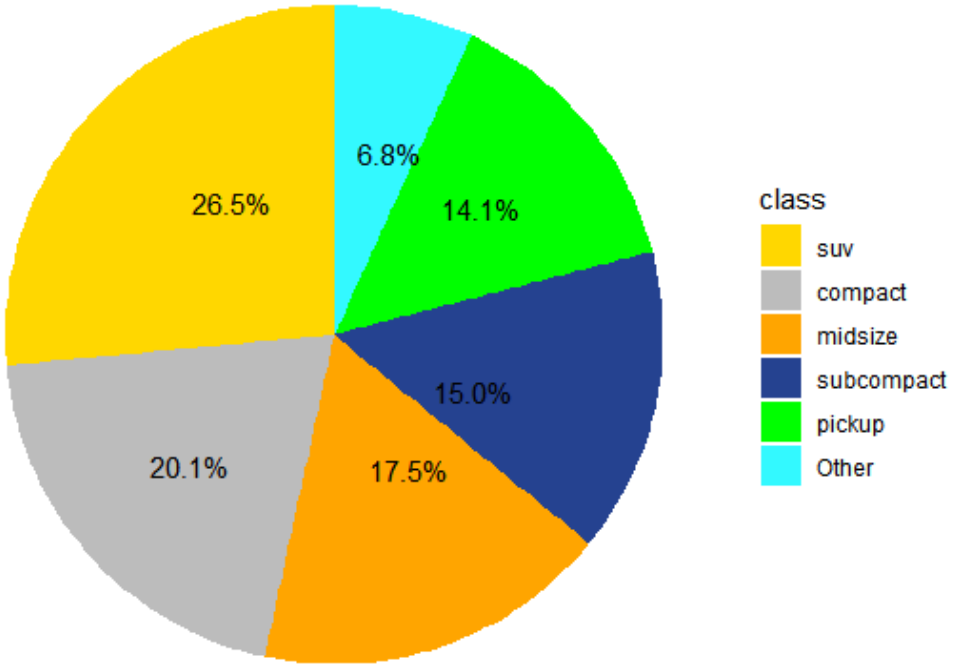
17: Pirámide poblacional



18: Gráfico circular 2

```
mpg %>%
  group_by(class) %>%
  summarize(freq = n()) %>%
  mutate(porc = freq / sum(freq)) %>%
  arrange(desc(porc)) %>%
  mutate(class = factor(c(class[1:5], rep("Other",2)), levels = as.character(c(class[1:5], "Other")))) %
  group_by(class) %>%
  summarize(porc = sum(porc)) %>%
  ungroup() %>%
  ggplot(aes(x= "", y = porc, fill = class)) +
  geom_col() +
  geom_text_repel(aes(label = scales::percent(round(porc,3))), position = position_stack(vjust = 0.5))+
  coord_polar(theta = "y") +
  scale_fill_manual(values = c("#ffd700", "#bcbcbc", "#ffa500", "#254290", "green", "#33F9FF"))+
  theme_void()
```

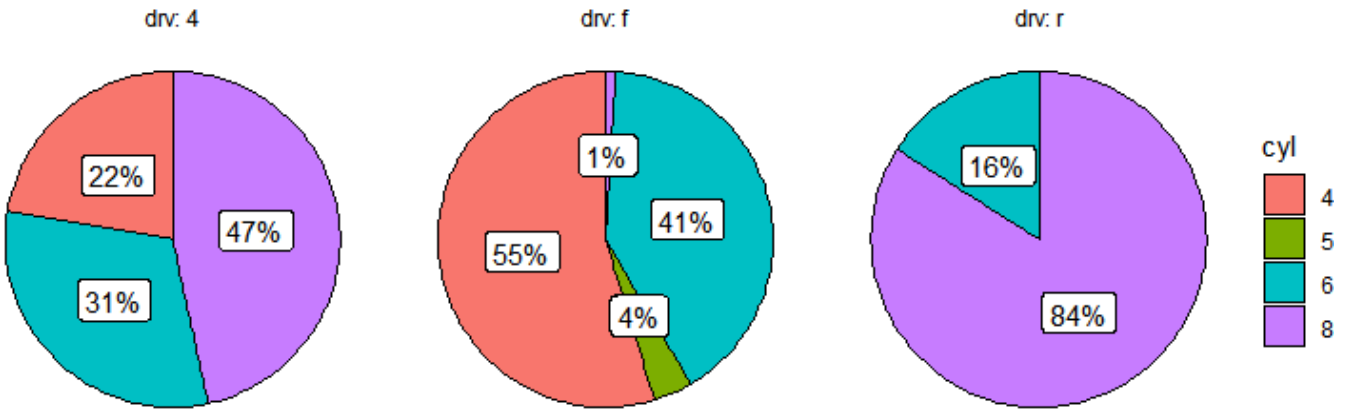
18: Gráfico circular 2



19: Gráfico circular 3

```
mpg %>%
  mutate_at(vars("cyl"), as.character) %>%
  mutate(drv = factor(drv, levels=c("4", "f", "r")),
         cyl = factor(cyl, levels=c("4", "5", "6", "8"))) %>%
  group_by(.dots = c('drv', 'cyl')) %>%
  summarize(counts = n()) %>%
  mutate(perc = (counts/sum(counts)) * 100) %>%
  arrange(desc(perc)) %>%
  ggplot(aes('', counts)) +
  geom_col(position = 'fill', color = 'black', width = 1, aes(fill = cyl)) +
  facet_wrap(~drv, labeller = "label_both") +
  geom_label(aes(label = paste0(round(perc), "%"), group = cyl),
             position = position_fill(vjust = 0.5), color = 'black', size = 4, show.legend = FALSE) +
  coord_polar(theta = "y") +
  theme_void()
```

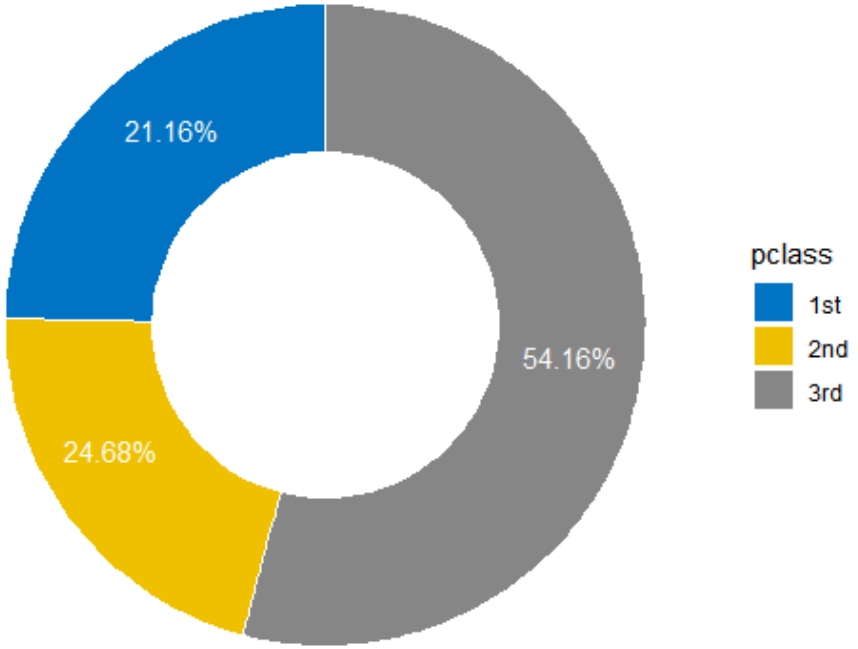
19: Gráfico circular 3



20: Gráfico de anillos

```
df %>%
  filter(!is.na(pclass)) %>%
  group_by(pclass) %>%
  tally(sort=T) %>%
  mutate(pclass = factor(pclass, levels = c("1st", "2nd", "3rd")),
         prop = round((n/sum(n))*100, 2),
         lab.ypos = cumsum(prop) - 0.5*prop) %>%
  ggplot(aes(x = 2, y = prop, fill = pclass)) +
  geom_bar(stat = "identity", color = "white") +
  coord_polar(theta = "y", start = 0) +
  geom_text(aes(y = lab.ypos, label = paste0(prop, "%")), color = "white") +
  scale_fill_manual(values = c("#0073C2FF", "#EFC000FF", "#868686FF")) +
  theme_void() +
  xlim(0.5, 2.5)
```

20: Gráfico de anillos



Referencias

Datanovia *Datanovia* https://www.datanovia.com/

Data Carpentry contributors *Data visualization with* `ggplot2` https://datacarpentry.org/R-ecology-lesson/04-visualization-ggplot2.html

En cours de rédaction :
JMU2017 *Advanced Data Visualization with* `ggplot2` https://4va.github.io/biodatasci/r-viz-gapminder.html

Otros:

- https://plot.ly/ggplot2/animations/
- https://plot.ly/r/animations/