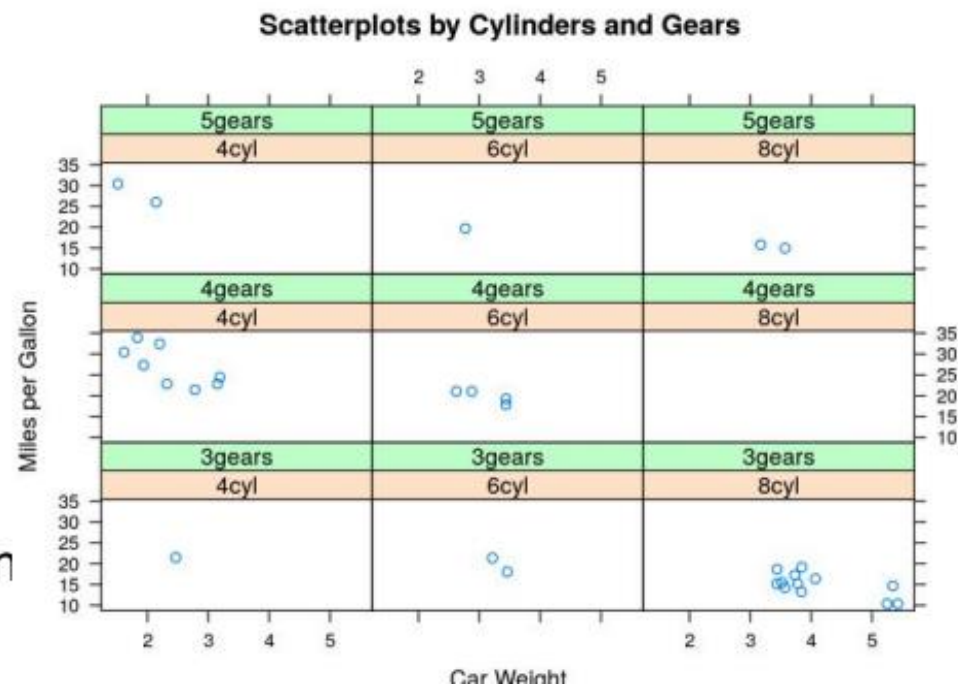


Introducción: Gráficos en R

- En la mayor parte de los lenguajes de programación, la capacidad de crear gráficos la proporcionan librerías adicionales, **ajenas a su núcleo**.
- En **R**, los gráficos son **nativos**.
- En **R** existen dos motores gráficos:
 - Funciones de alto nivel (*plot, hist, barplot, boxplot*) que a su vez invocan funciones de bajo nivel que nos permiten hacer modificaciones en los gráficos.



ggplot2



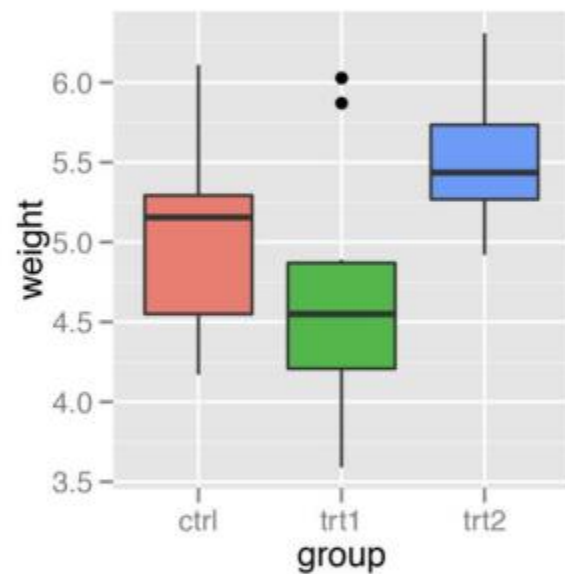
- Es una librería de R implementada por [Hadley Wickham](#) para [visualización de datos](#)
- Forma parte de un conjunto de librerías llamado [tidyverse](#)
- ggplot2 esta basado en “[The Grammar of Graphics](#)” de Leland Wilkinson (2000).
- Tiene más de 10 años y es usado por miles de personas en todo el mundo.
 - [Ventaja](#): Hay muchas comunidades que te ayudan a resolver dudas

Instalación de ggplot2

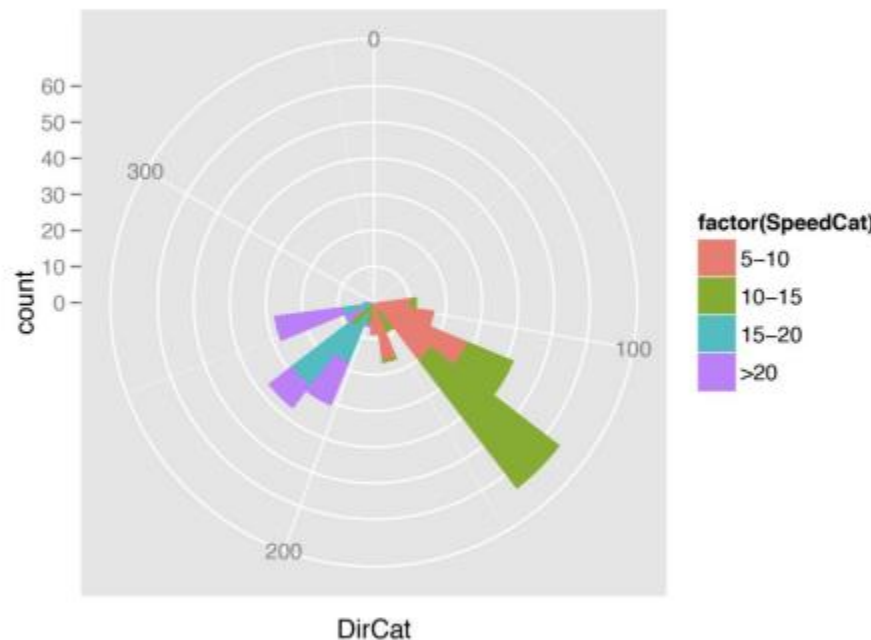
- Instalando la librería de *tidyverse* completa
 - `install.packages("tidyverse")`
- Instalándola desde su repositorio de preferencia
 - `install.packages("ggplot2", dependencies=TRUE)`
- O instalar la versión en desarrollo desde GitHub
 - `install.packages("devtools")`
 - `devtools::install_github("tidyverse/ggplot2")`



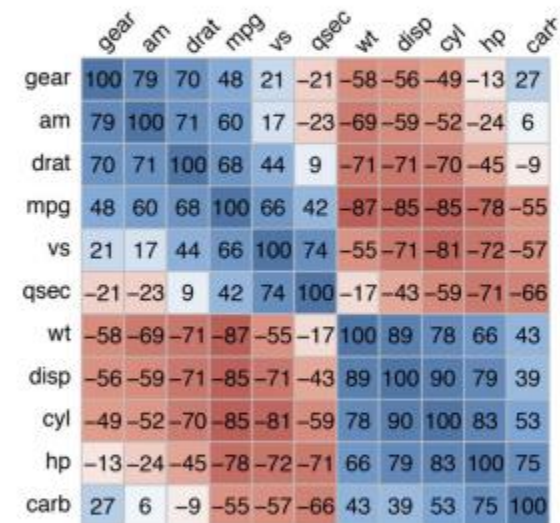
Repertorio de gráficos



Box plots



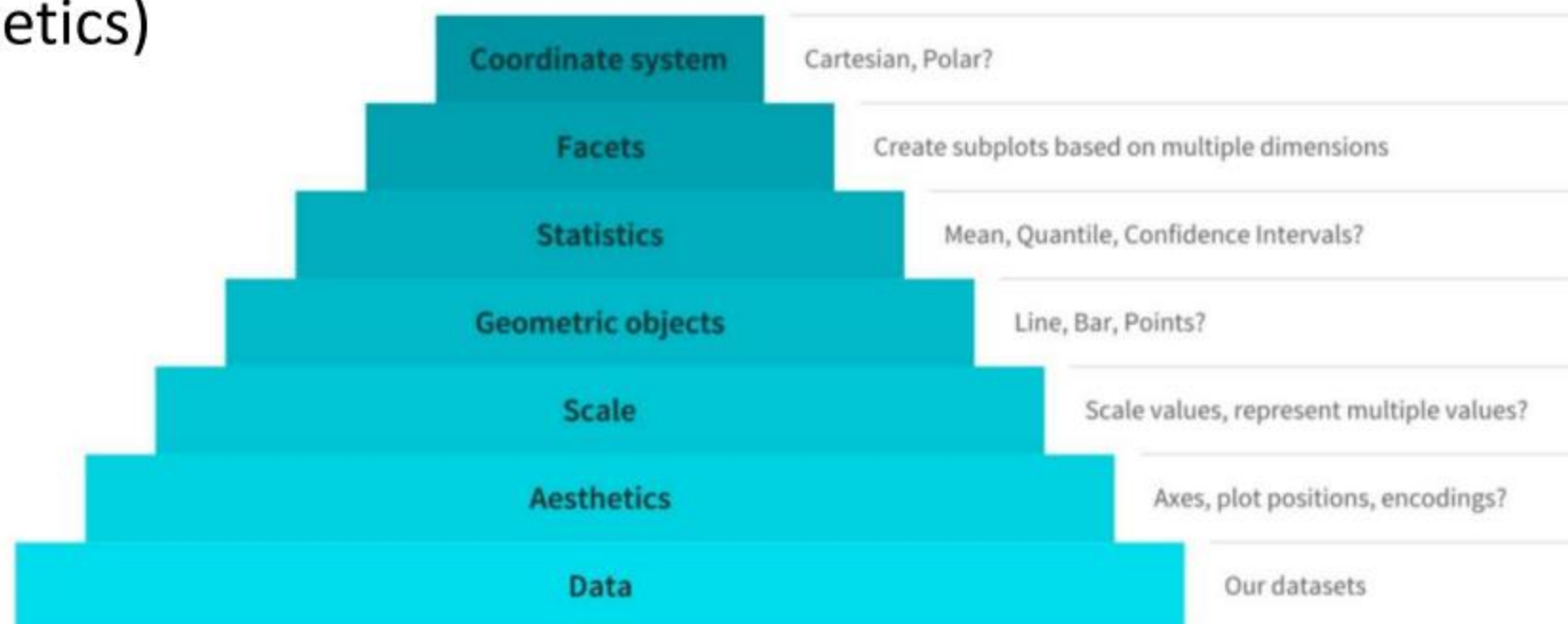
Polar plots



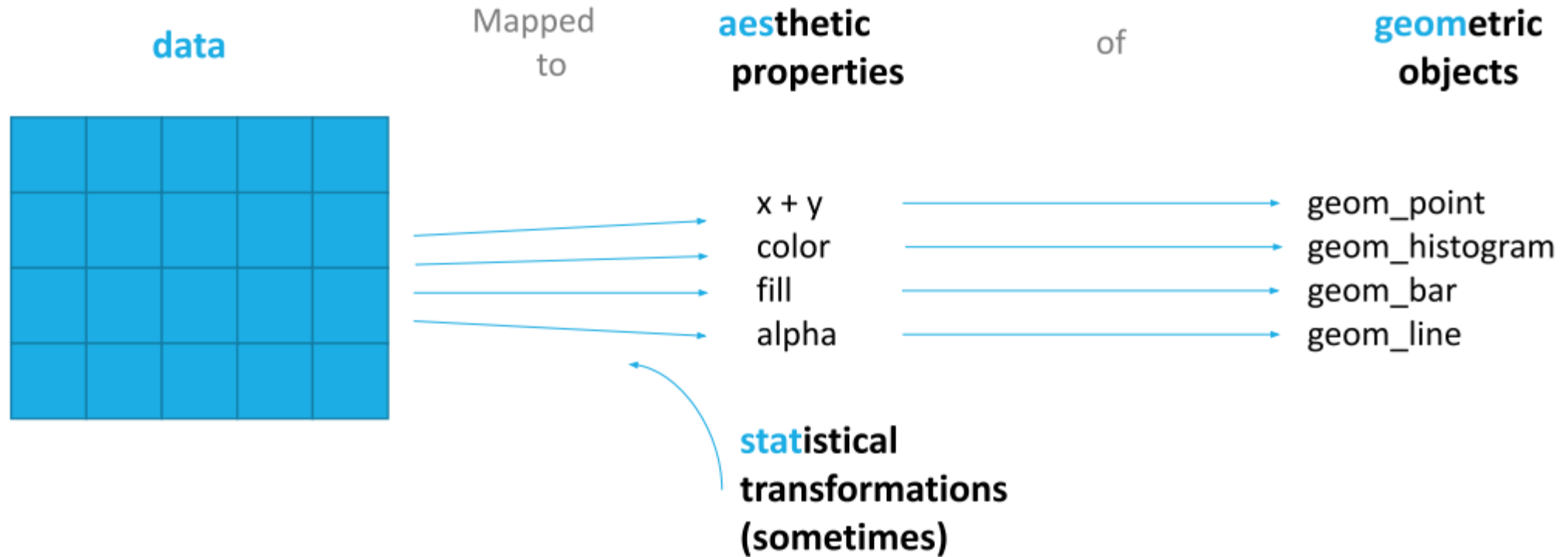
corplots

Elementos de un gráfico en ggplot

- Datos
- Estéticas (**a**esthetics)
- Capas
- Facetas
- Temas



The grammar of the graphics and ggplot2



Funciones principales en ggplot2

- Hay dos funciones principales en ggplot2:
- Una función que nos permite echarle un vistazo rápido a los datos
 - `qplot()`
- Una función más compleja que nos va a permitir explorar más a fondo los datos
 - `ggplot()`

Sintaxis `qplot()`

- Crea un gráfico completo con los datos, geom y mapeos. Proporciona muchos valores por defecto.

Mapeos estéticos

Datos

- `qplot(x= variable_X, y= variable_Y, color=variable_color, data= data.frame, geom= "point")`

Objetos
geométricos

Datos: mtcars

- Es un **data.frame** que viene con la librería de **ggplot2**
- Los datos se extrajeron de la revista **Motor Trend** de EE. UU. de 1974
- Comprenden el **consumo de combustible y 10 aspectos del diseño** y el **rendimiento del automóvil** para **32 automóviles** (modelos 1973–74).

A data frame with 32 observations on 11 (numeric) variables.

```
[, 1] mpg Miles/(US) gallon  
[, 2] cyl  Number of cylinders  
[, 3] disp Displacement (cu.in.)  
[, 4] hp   Gross horsepower  
[, 5] drat Rear axle ratio  
[, 6] wt   Weight (1000 lbs)  
[, 7] qsec 1/4 mile time  
[, 8] vs   Engine (0 = V-shaped, 1 = straight)  
[, 9] am   Transmission (0 = automatic, 1 = manual)  
[,10] gear Number of forward gears  
[,11] carb Number of carburetors
```

Sintaxis ggplot()

- Crea un gráfico al que se le añaderan capas. Sin valores por defecto, pero que proporciona más control.

Datos



```
ggplot(data= data.frame, aes(x= variable_X, y= variable_Y)) +
```

Geom



estadística



Mapeos
específicos



```
geom_point( aes(color= variable_color)) +  
geom_smooth(method= "lm") +  
coord_cartesian () +  
scale_color_gradient() +  
theme_bw()
```

Añadimos capas,
elementos con un +



Elementos adicionales



Datos -> iris

- Este famoso conjunto de datos de **iris** (de Fisher o Anderson) da las medidas en centímetros de las variables:
 - Longitud y ancho del sépalo
 - Largo y ancho del pétalo
 - Para 50 flores de cada una de las 3 especies de iris.
 - Iris setosa, versicolor y virginica.



Estéticas -> aesthetics

- En un conjunto de datos hay muchas **variables** que asignamos a columnas dentro del data.frame
- Estas **variables** dentro de la terminología de ggplot2 se les llama **estéticas**
- Por lo que nosotros vamos a darle un significado a las variables dentro del gráfico.

```
p <- p + aes(x= Petal.Length, y = Petal.Width, colour = Species)
```

- Estamos añadiendo al objeto **p** la información sobre las estéticas que obtiene de los datos “iris”

Capas -> Layers

- Las capas o los objetos geométricos en ggplot (**geom**) indican que hacer con los datos y las estéticas que le indicaste previamente.
- Es la representación gráfica de tus datos
- ggplot2 nos permite ir añadiendo “**capas**” al lienzo para irlo modificando de acuerdo a nuestras necesidades.

```
# Add layers
p <- p + geom_point()
# Display the object "p"
p
p <- p + geom_smooth()
```


Facetas -> Faceting

- Las facetas implementan los gráficos de Trellis
- Las facetas nos permiten **hacer comparaciones** entre las diferentes variables
- Para poder hacer comparaciones adecuadas, una característica esencial es que se **comparten ejes**.

```
## Facet grid
```

```
### Vertical
```

```
ggplot(iris, aes(x = Petal.Length, y = Petal.Width)) + geom_point()  
+ geom_smooth() + facet_grid(~ Species)
```

```
### Horizontal
```

```
ggplot(iris, aes(x = Petal.Length, y = Petal.Width)) + geom_point()  
+ geom_smooth() + facet_grid(Species ~ .)
```

Temas -> Themes

- Los temas en ggplot2 permiten hacer modificaciones a los aspectos estéticos del gráfico.
- Lo que incluye son:
 - los ejes, etiquetas, colores de fondo, el tamaño de los márgenes, paletas de colores.
- Los temas son una colección de elementos modificables.
- El tema por defecto en ggplot es el `theme_grey`
- Podemos elegir entre una variedad de temas que tiene ggplot2
- Es posible construir temas propios y personalizados

```
## Change the Theme
```

```
p <- p + theme_bw()  
p
```


¿Cómo guardo mis gráficos?

- Hay **dos maneras** de guardar tus imágenes

1. Con funciones propias de la base de R

- **png()**
- **pdf()**
- Tu les indicas las especificaciones de tamaño y calidad así como la ruta y el nombre de la imagen.
- La función se abre antes de generar la gráfica y se tiene que cerrar el archivo con la función **dev.off()**

2. Con la función **ggsave()** de ggplot2

- Guarda la última imagen que creaste en el directorio en donde estás trabajando.

Gráficos Interactivos de ggplot: esquisse



- Paquete de R que permite realizar análisis exploratorios graficando con ggplot de manera interactiva.
- Genera una interfaz gráfica donde puedes seleccionar con botones las opciones del plot
- Genera el código de ggplot para poder reproducir las graficas
- Permite de manera rápida e intuitiva explorar los datos.

Gráficos Interactivos de ggplot:

esquisse

- Instalándola desde su [repositorio](#) de preferencia
 - `install.packages("esquisse")`
- O instalar la versión en desarrollo desde [GitHub](#)
 - `remotes::install_github("dreamRs/esquisse")`

```
install.packages("esquisse")  
library(esquisse)
```

Gráficos Interactivos de ggplot: esquisse

- Cargar datos
- Para lanzar la aplicación se utiliza el comando “**esquisser()**”, dentro del cual se pone el **data.frame** a explorar

```
esquisser(iris)
```

Gráficos Interactivos de ggplot: esquisse



```
esquisser(iris)
```

Geom (Tipo de gráfico) →

Variables →

Mapeos →

Área del plot →

Controles →

The image shows the 'ggplot2 builder' web application interface. At the top, there's a dark blue header with 'Data' on the left, 'ggplot2 builder' in the center, and 'Close' on the right. Below the header, a row of variable buttons is shown: 'Sepal.Length', 'Sepal.Width', 'Petal.Length', 'Petal.Width', and 'Species'. The 'Species' button is highlighted in orange. To the right of these buttons is a 'Variables' label with an arrow pointing to the variable selection area. Below the variable buttons is a row of mapping boxes: 'X', 'Y', 'Fill', 'Color', 'Size', 'Group', 'Facet', 'FacetRow', and 'FacetCol'. An arrow labeled 'Mapeos' points to this row. The main area of the interface is a large white box labeled 'Área del plot' with an arrow pointing to it. In the bottom right corner of this area is a green 'Play' button. At the bottom of the interface is a dark blue footer with four sections: 'Labels & Title', 'Plot options', 'Data', and 'Export & code'. An arrow labeled 'Controles' points to this footer.