

BIENVENIDOS A

R-LADIES RESISTENCIA-CORRIENTES



```
library(dplyr)
library(magrittr)
```

```
rladies_global %>% filter(city == 'Resistencia') && filter(city == 'Corrientes')
```



TERCER ENCUENTRO

R-LADIES RESISTENCIA-CORRIENTES

Viernes 12 de Octubre de 2018

Hoy hablamos sobre...

PARTE 1: LatinR 2018 -

Te contamos que vimos y que nos dejó el evento

Breve Repaso

¿Cómo instalamos un paquete?

¿Cómo importamos datos?

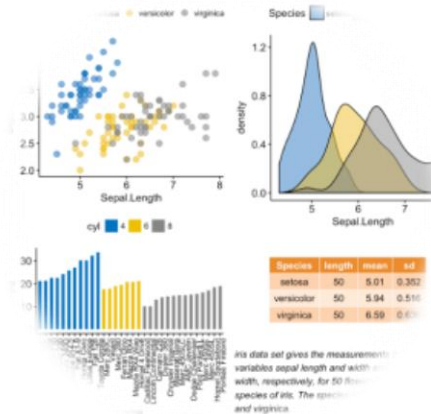
Visualización

Importancia de la visualización

Paquete ggplot2

Elementos de la visualización

Ejemplos – Práctica Hands on



PARTE 2: La Dra. Laura nos contará sobre

Pronóstico de cosecha citrus 🍊 🍊

con geoR.

Stickers:

- Rladies
- Rladies Resistencia - Corrientes
- Mujeres en STEM



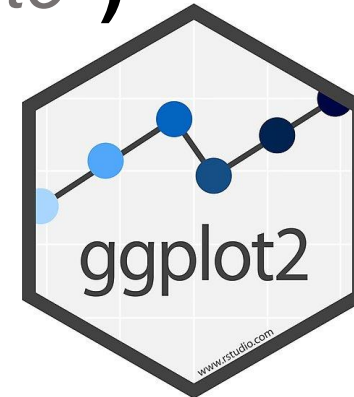


PARTE 1

Breve Repaso

¿Cómo instalamos un paquete?

- **install.packages**("nombre_del_paquete")
- **library**("nombre_del_paquete ")



Importación de datos en R



#Por medio de read.csv()

- Iris <- **read.csv**("E:/DATASETS/iris.csv")
- View(iris)

#Por medio de read.table()

- mtcars <- **read.table**("E:/DATASETS/mtcars.txt")
- mtcars <-**read.table**("E:/DATASETS/mtcars.txt",
header=TRUE)
- View(mtcars)

Importación de datos en R



#También podemos importar otro tipo de datos mediante el paquete **readxl**. Permite leer archivos con formato (.xls and .xlsx) into R

Ejemplo:

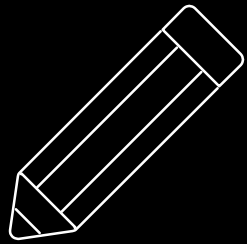
- `install.packages("readxl")`
- `library("readxl")`
- `estadis <- read_xl ("estadistica2009.xlsx")`





PARTE 2

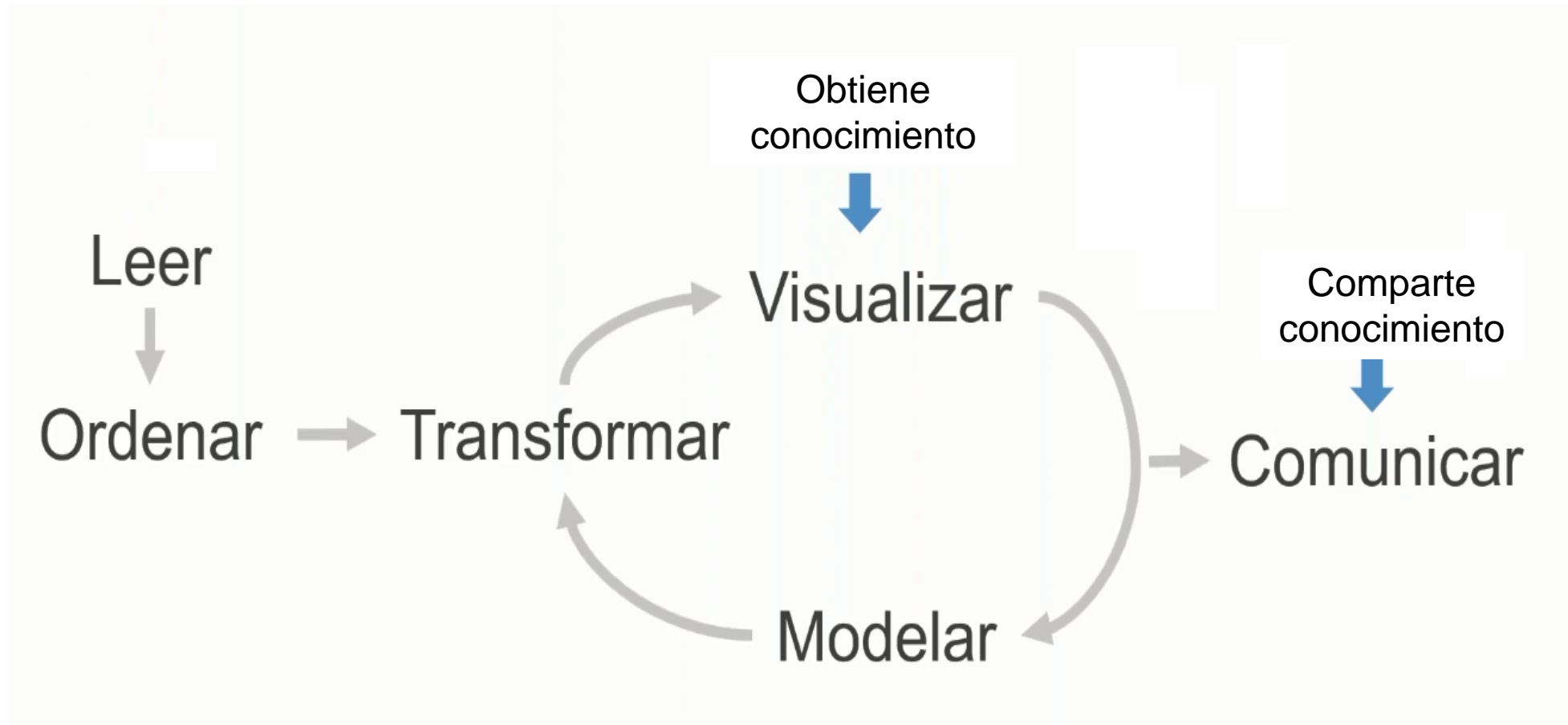




Visualización con ggplot2



La visualización dentro del esquema de trabajo en ciencia de datos



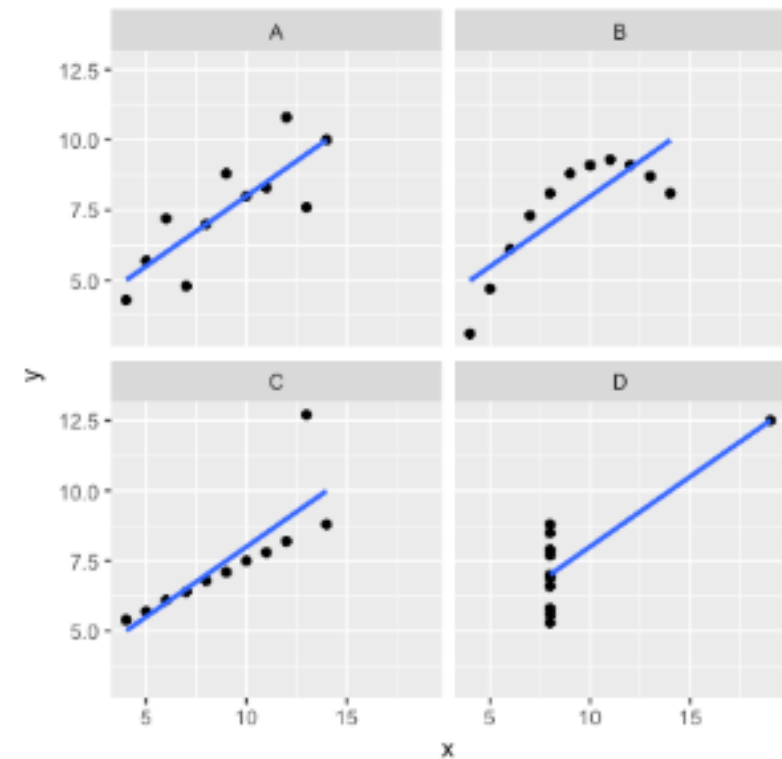
¿Porqué usamos visualización?



- Los gráficos proveen más información que los resúmenes numéricos

Por ejemplo:

- $n=11$
- $\bar{x}=9.0$
- $\bar{y}=7.5$
- $\hat{\beta}_1=0.5$
- $y=3+0.5x$
- $R^2=0.667$



Visualización Estadística



- La visualización juega un rol importante en todas las etapas del análisis estadístico.
- **Exploración:** Encontrar patrones generales y específicos en los datos.
- **Modelado:** Chequear supuestos sobre los datos antes de modelar.
- **Diagnóstico:** Visualizar el modelo en el espacio de los datos ó los datos en el espacio del modelo.



¿Cuál es la relación entre X e Y?

X	Y
1.972	1.236
1.112	1.994
0.000	1.009
0.665	1.942
0.235	0.356
0.247	1.658
1.275	1.961
0.702	0.045
1.760	0.350
1.691	0.277
1.628	1.778
1.957	1.290

¿Porqué usar el ggplot2?



- **ggplot2** es un paquete para producir gráficos estadísticos o de datos desarrollado por [Wickham \(2016\)](#).
- Tiene una teoría que lo sustenta basada en el *Grammar of Graphics* [Wilkinson \(2006\)](#).
- Es el paquete de gráficos estadísticos más dominante en R, tiene ya 10 años.
- Es parte de un conjunto de paquetes que tiene foco en la ciencia de datos llamado *Tidyverse*.
- [ggplot2: Elegant Graphics for Data Analysis](#)



Visualizaciones efectivas

- No todas las visualizaciones son igualmente efectivas
- Las mejores visualizaciones son aquellas que requieren el uso de la visión "pre-attentive" (instantáneo, sin aparente esfuerzo visual) [Cleveland William and McGill \(1985\)](#).



Visualizaciones efectivas

Para que una visualización sea fácilmente entendible:

- Pensar siempre en el público objetivo
- ¿Qué queremos mostrar con el gráfico?
- Resalta la información más importante
- Proporciona contexto a los datos
- Usa el color de forma inteligente
- El título del gráfico proporciona información



ggplot2

- Conjunto de componentes independientes, nos da flexibilidad
- No limitado a gráficos pre determinados, puedes crear lo que quieras
- Definido en base a un conjunto de principios, fácil de aprender
- Puedes producir gráficos estándar de calidad publicable en poco tiempo
- Diseñado para trabajar iterativamente agregando capas



¿Porqué usar una gramática de los gráficos ?

Para:

- Graficar datos con los que trabajamos
- Crear nuevas visualizaciones
- Encontrar mejores gráficos para visualizar nuestros datos

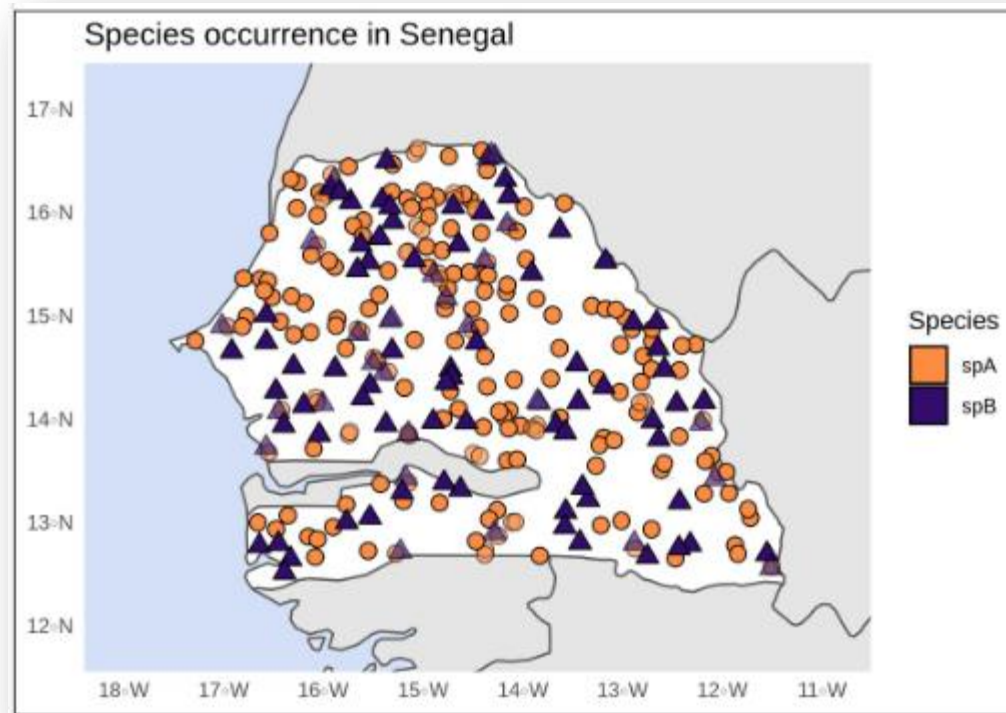


Grammar of Graphics o gramática de los gráficos

- La Gramática de los gráficos nos responde las preguntas:
- ¿Qué es un gráfico?
- ¿Cómo describir un gráfico?
- ¿Cómo crear un gráfico?

Grammar of Graphics

- Un gráfico es un mapeo de los datos a atributos estéticos de objetos geométricos.





Gramática de gráficos - Componentes

- **data:** con mapeos (aes) entre los datos y los elementos gráficos
- **layers:** elementos geométricos (geoms, son puntos, líneas, rectángulos, etc.) y transformaciones estadísticas (identidades, promedio, mediana, conteos, etc.)
- **scales:** mapea valores en el espacio de los datos a valores en el espacio estético (ej. color, tamaño, forma o posición)
- **coord:** normalmente Cartesianas, pero por ej; polares.
- **facets:** como se arregla el display cuando son muchos gráficos
- **theme,** items para mejorar el gráfico como fuente, tamaño, color, background.



Instalar ggplot2

```
install.packages ("ggplot2")  
library (ggplot2)
```

```
install.packages ("devtools")  
library (devtools)  
install_github ("tidyverse/ggplot2")
```

Versión en desarrollo: <https://github.com/tidyverse/ggplot2>



Ejemplo propina

```
##      total propina sexo  fuma  dia  momento cantidad
##      <dbl>   <dbl> <chr> <chr> <chr> <chr>      <int>
## 1    17.0     1.01 F    No   Do   Noche        2
## 2    10.3     1.66 M    No   Do   Noche        3
## 3    21.0     3.5  M    No   Do   Noche        3
## 4    23.7     3.31 M    No   Do   Noche        2
## 5    24.6     3.61 F    No   Do   Noche        4
## 6    25.3     4.71 M    No   Do   Noche        4
```



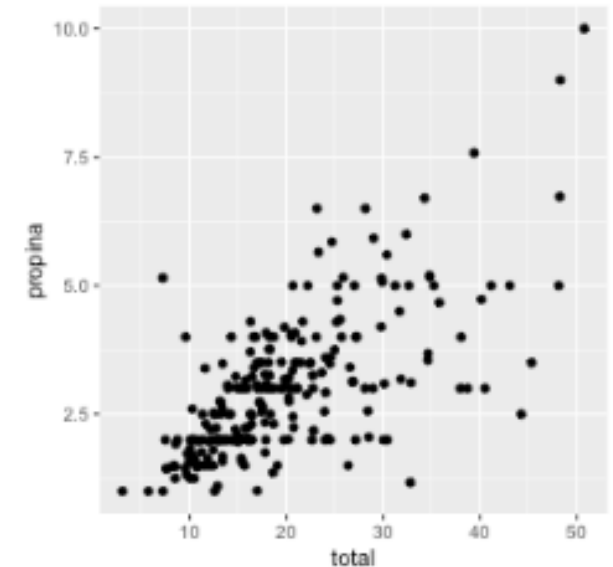
Tres componentes de todo gráfico (plot)

1. **Datos (data)**
2. **Atributos estéticos (aes)** : un conjunto de mapeos estéticos entre las variables en los datos y las propiedades visuales.
3. **Capas (layers)**: al menos una capa que describe cómo representar cada observación. Las capas son creadas con la función **geom**.

Tres componentes de todo gráfico: Veamos un ejemplo

- **data:** `propina`
- **aes:** `total` mapeado a la posición x , `propina` mapeado a la posición y.
- **layer:** puntos mediante `geom_point`.

```
ggplot(data = propinas, aes(x = total, y = propina)) +  
  geom_point() +  
  theme(aspect.ratio = 1)
```



Ejemplo propina

- ¿Qué vemos?
 - Hay una relación lineal débil entre propina y la cuenta total
- Hay mucha variabilidad
- Las líneas horizontales indican la preferencia de la gente a dar de a 1 dólar de propina c/u



Color, tamaño, forma y otros aes

- Para agregar otras variables al gráfico podemos usar otros **aes** o **componentes estéticas** como color, forma o tamaño:

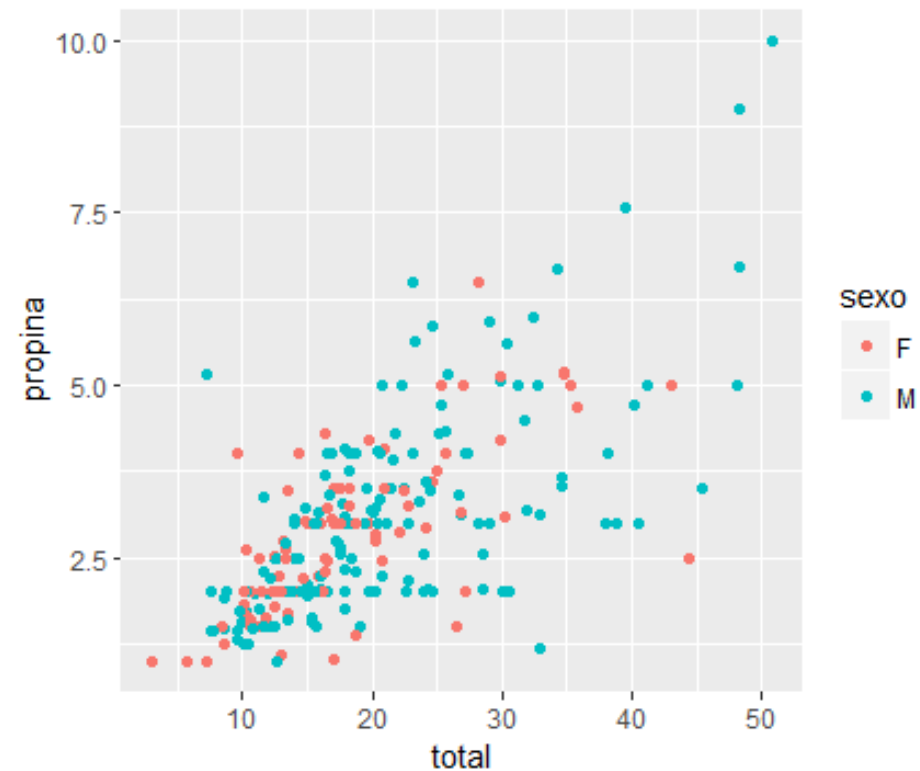
```
aes(x = total, y = propina, colour = sexo)
```

```
aes(x = total, y = propina, shape = sexo)
```

```
aes(x = total, y = propina, size = cantidad)
```

Color

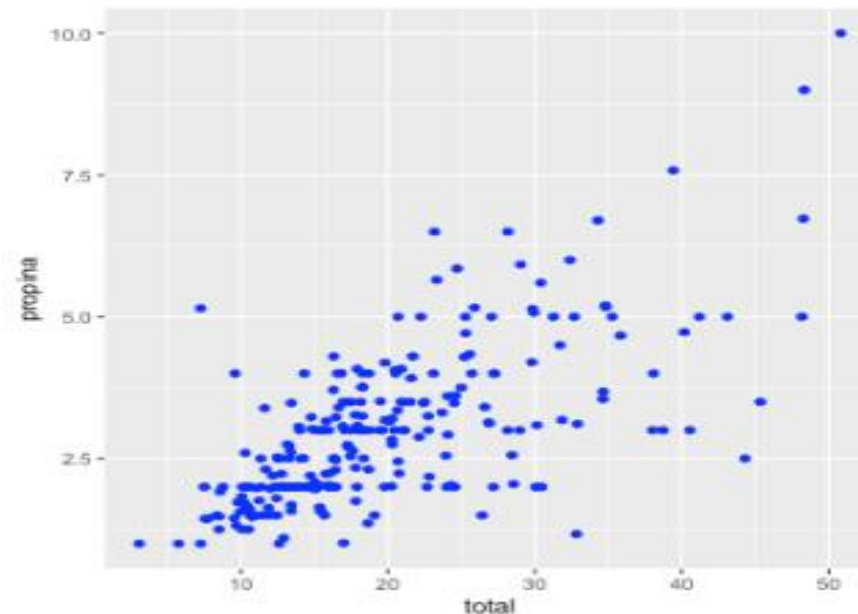
```
ggplot(data = propinas, aes(x = total, y = propina, colour = sexo)) +  
  geom_point() + theme(aspect.ratio = 1)
```



Color fijo

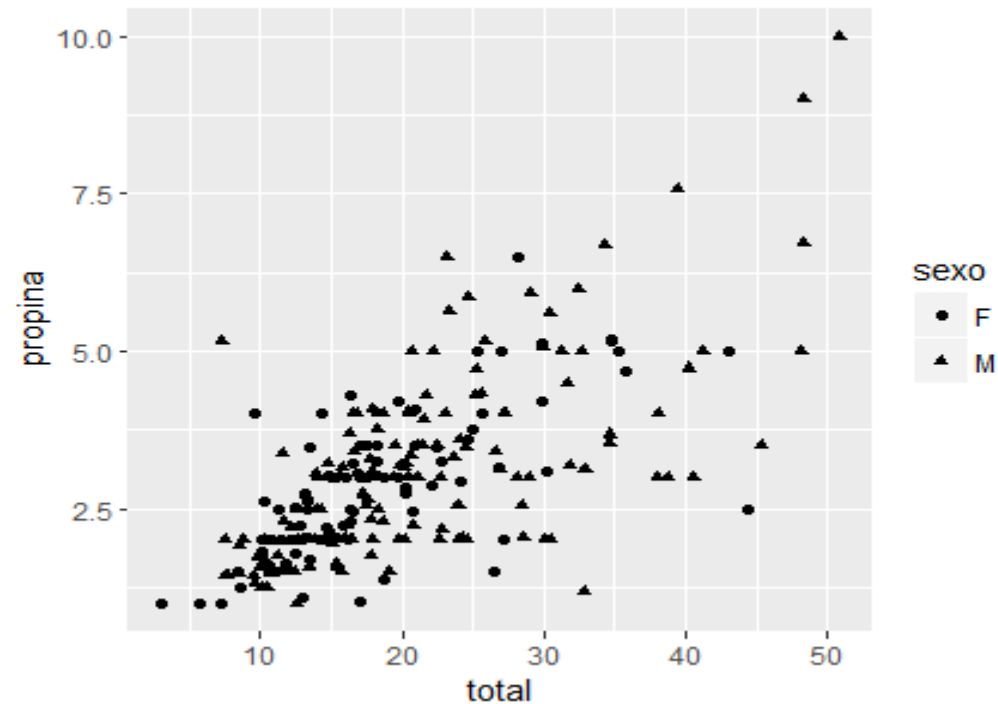
- Si queremos fijar un elemento estético a un valor fijo hay que hacerlo en la capa de afuera sin aes o usar I ('blue') en aes

```
ggplot(data = propinas, aes(x = total, y = propina) )+  
  geom_point(colour = "blue") + theme(aspect.ratio = 1)
```



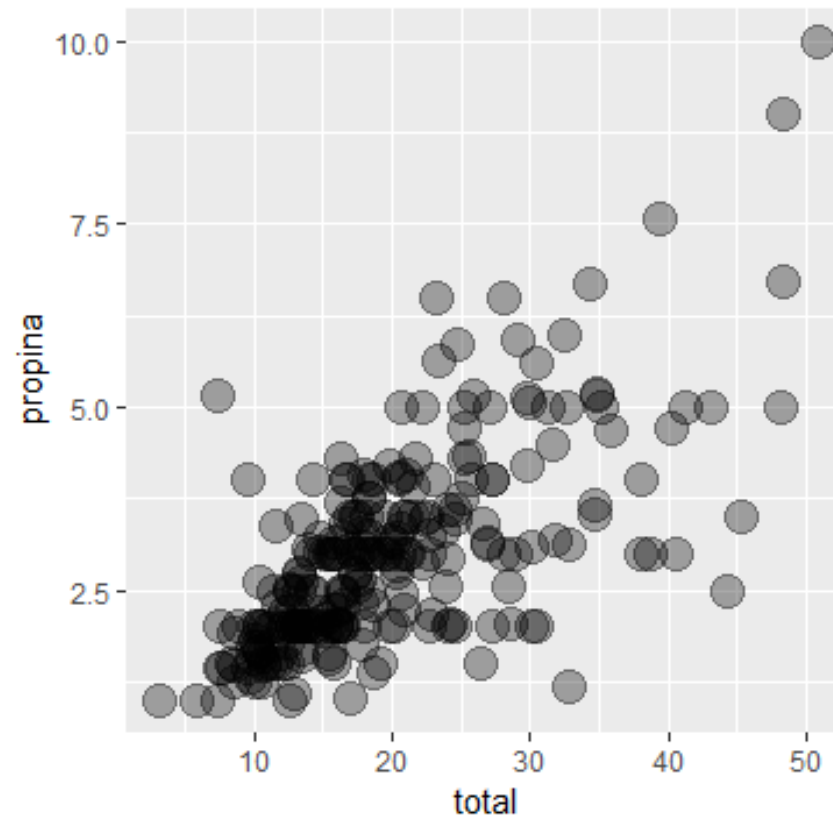
Forma

- `ggplot(data = propinas, aes(x = total, y = propina, shape = sexo)) + geom_point() + theme(aspect.ratio = 1)`



Tamaño

- `ggplot(data = propinas, aes(x = total, y = propina)) +
 geom_point(size = 5, alpha = 1 / 3) + theme(aspect.ratio = 1)`





scales

- Los datos pueden ser convertidos desde unidades de datos como moneda, sexo, día, etc. a unidades gráficas como color, forma, tamaño, etc. que se pueden representar en la computadora
- **La conversión se llama escalado y es realizada por scales o escalas.**
- Cada color es representado por una cadena de seis letras y números, cada size por números y cada shape por enteros.
- Estas especificaciones estéticas que tienen sentido para R se describen en `vignette("ggplot2-specs")`
- <https://cran.r-project.org/web/packages/ggplot2/vignettes/ggplot2-specs.html>
- <http://research.stowers.org/mcm/efg/R/Color/Chart/>



scales

- Mapeo estético solamente dice que una variable debe ser mapeada a un elemento estético pero no dice cómo debe ser.
- Cuando mapeo una variable a shape usando `aes(shape = x)` no especifico la forma (shape) específica que debe tomar.
- Cuando uso `aes(color = z)` no digo de que color debe ser
- Describir el color, la forma o el tamaño (color, shape, size) a usar se hace mediante transformaciones en scale.



scales

- color y fill
 - size
 - shape
 - Linetype
- Las scales se modifican con una serie de funciones con el siguiente esquema de nombrado `scale_<aesthetic>_<type>`. Mirar `scale_<tab>` ver la lista de las funciones de scale.



scales disponibles

Scale	Types	Examples
<code>scale_color_</code>	<code>identity</code>	<code>scale_fill_continuous</code>
<code>scale_fill_</code>	<code>manual</code>	<code>scale_color_discrete</code>
<code>scale_size_</code>	<code>continuous</code>	<code>scale_size_manual</code>
	<code>discrete</code>	<code>scale_size_discrete</code>
<code>scale_shape_</code>	<code>discrete</code>	<code>scale_shape_discrete</code>
<code>scale_linetype_</code>	<code>identity</code>	<code>scale_shape_manual</code>
	<code>manual</code>	<code>scale_linetype_discrete</code>
<code>scale_x_</code>	<code>continuous</code>	<code>scale_x_continuous</code>
<code>scale_y_</code>	<code>discrete</code>	<code>scale_y_discrete</code>
	<code>reverse</code>	<code>scale_x_log</code>
	<code>log</code>	<code>scale_y_reverse</code>
	<code>date</code>	<code>scale_x_date</code>
	<code>datetime</code>	<code>scale_y_datetime</code>

- https://ggplot2.tidyverse.org/reference/scale_shape.html#arguments

Gramática de gráficos

```
ggplot(propinas, aes(total, propina, colour = fuma)) +  
  geom_point()
```





¿Qué hay atrás de este gráfico?

- Cada observación representada como un punto cuya posición está de acuerdo a dos variables (posición horizontal y vertical)
- Cada punto tiene tamaño, color y forma estos atributos son denominados elementos estéticos (inglés aesthetics aes)
- Los aes son propiedades que pueden ser percibidas en el gráfico cada aes puede ser mapeado a una variable o fijado en un valor constante
- **total** es mapeado a la posición horizontal, **propina** a la posición vertical y **fuma** al color. Tamaño y forma no son mapeados a variables (valor por defecto)



Capas de un gráfico

- Los datos, mapeos estéticos, objetos geométricos y las transformaciones estadísticas forman una **capa**.
- Podemos tener un gráfico con muchas capas.

La gramática de capas define componentes de un gráfico:

- datos y conjunto de mapeos de variables a elementos estéticos
- una o más capas, cada capa tiene un elemento geométrico, una transformación estadística y una posición.



Capas de un gráfico

```
ggplot() +  
  layer(  
    data = propinas, mapping = aes(x = total, y = propina),  
    geom = "point", stat = "identity", position = "identity"  
  ) +  
  scale_x_continuous() +  
  scale_y_continuous() +  
  coord_cartesian()
```



Capas de un gráfico

Equivalente a

```
ggplot(data = propinas, aes(x = total, y = propina)) +  
geom_point()
```

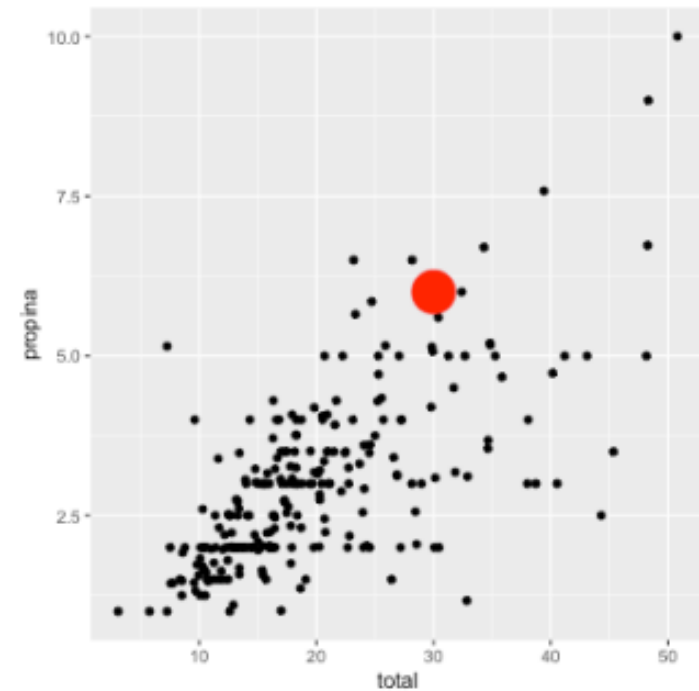
Capas de un gráfico

ggplot() +

geom_point (data = propinas, aes(x = total, y = propina)) +

geom_point(data = data.frame(x = 30, y = 6), aes(x, y),

color = "red", size = 10)



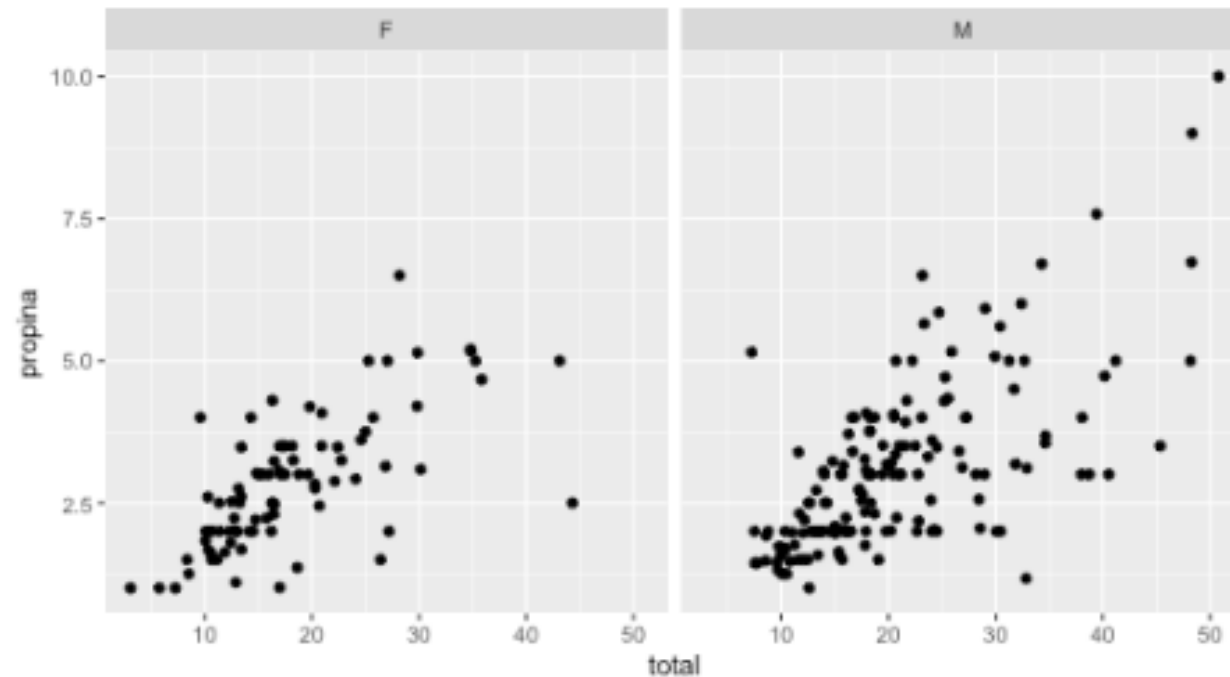


Paneles

- Se pueden desplegar variables categóricas adicionales en un gráfico particionando el panel.
- Crea tablas de gráficos partiendo los datos en subconjuntos y mostrándolos el mismo gráfico para cada subconjunto.
- Dos tipos: `facet_grid` y `facet_wrap`

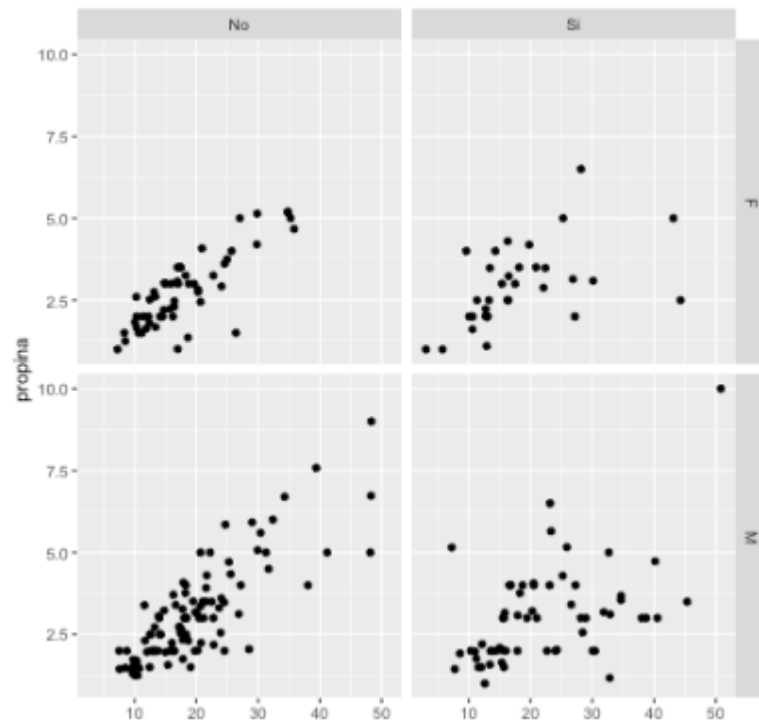
facet_wrap

- `ggplot(data = propinas, aes(x = total, y = propina)) +
geom_point() + theme(aspect.ratio = 1) +
facet_wrap(~sexo)`



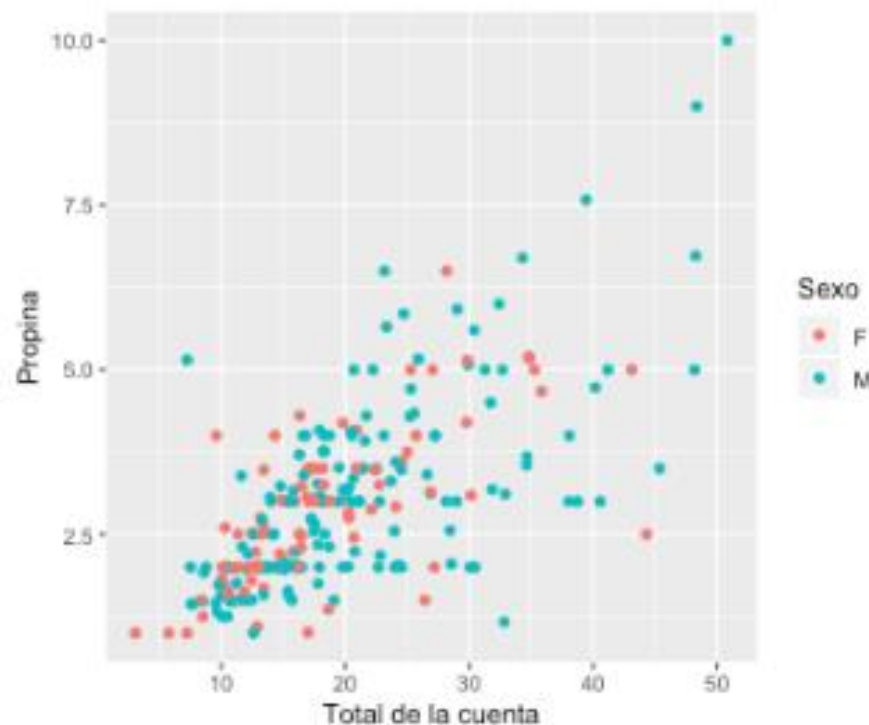
facet_grid

```
ggplot(data = propinas, aes(x = total, y = propina)) +  
  geom_point() + theme(aspect.ratio = 1) +  
  facet_grid(sexo ~ fuma)
```



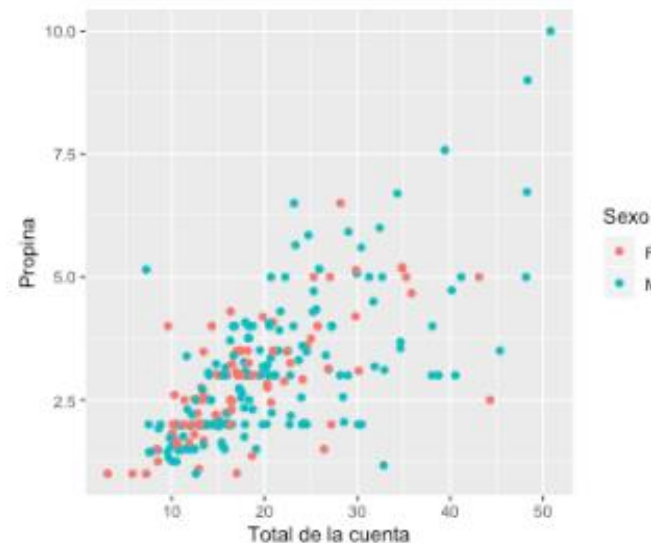
Incluir rótulos (labs)

```
ggplot(data = propinas, aes(x = total, y = propina, color = sexo)) +  
  geom_point() + theme(aspect.ratio = 1) +  
  labs(x = "Total de la cuenta", y = "Propina", color = "Sexo")
```



Incluir rótulos (*labs*)

```
ggplot(data = propinas, aes(x = total, y = propina, colour = sexo)) +  
  geom_point() + theme(aspect.ratio = 1) +  
  scale_x_continuous (name = "Total de la cuenta") +  
  scale_y_continuous (name = "Propina") +  
  scale_color_discrete ("Sexo")
```





Theme

- theme en un gráfico permite controlar los elementos que no son datos en el mismo.
- Ayuda a hacer tu gráfico estéticamente como lo querés no afecta el mapeo de datos.
- theme te da control sobre **fuente, tamaño, color, background.**



Material de ayuda

▪ Más geoms

<http://ggplot2.tidyverse.org/reference/>

▪ Extensiones

Hay 40 extensiones de ggplot2

<http://www.ggplot2-exts.org/gallery/>

▪ ggplot2 ayuda

Lista de mails: <http://groups.google.com/group/ggplot2>

stackoverflow: <http://stackoverflow.com>



Referencias

- <http://natydasilva.com>
- https://natydasilva.github.io/taller_LatinR
- Libro: <https://github.com/hadley/ggplot2-book>.

