



## Finanzas en R

### Visualization

Sebastián Egaña Santibáñez 

Nicolás Leiva Díaz 

---

### Enlaces del profesor

 <https://segana.netlify.app>

 <https://github.com/sebaegana>

 <https://www.linkedin.com/in/sebastian-egana-santibanez/>

---

### Visualización en R

La principal librería para fines de visualización corresponde a GGPlot2. Crearemos un gráfico simple utilizando la librería ggplot2<sup>1</sup>.

Revisemos el Cheatsheet asociado:

[Enlace acá](#)

La construcción de un gráfico utilizando GGPlot2 se realiza a través de los siguientes elementos:

1. Un set de datos.

---

<sup>1</sup>Recordar instalar y cargar la librería

2. Un sistema de coordenadas.
3. Figuras geométricas (geoms) o representaciones visuales de los datos.

Para cambiar la forma en que se muestran los datos se pueden editar las figuras geométricas (geoms), cambiando color, tamaño y la ubicación de los puntos en x y/o en y.

Inclusive se plantea una plantilla para esto:

```
ggplot (data = <DATA> ) +  
<GEOM_FUNCTION> (mapping = aes( <MAPPINGS> ),  
stat = <STAT> , position = <POSITION> ) +  
<COORDINATE_FUNCTION> +  
<FACET_FUNCTION> +  
<SCALE_FUNCTION> +  
<THEME_FUNCTION>
```

En donde las dos primeras líneas son obligatorias.

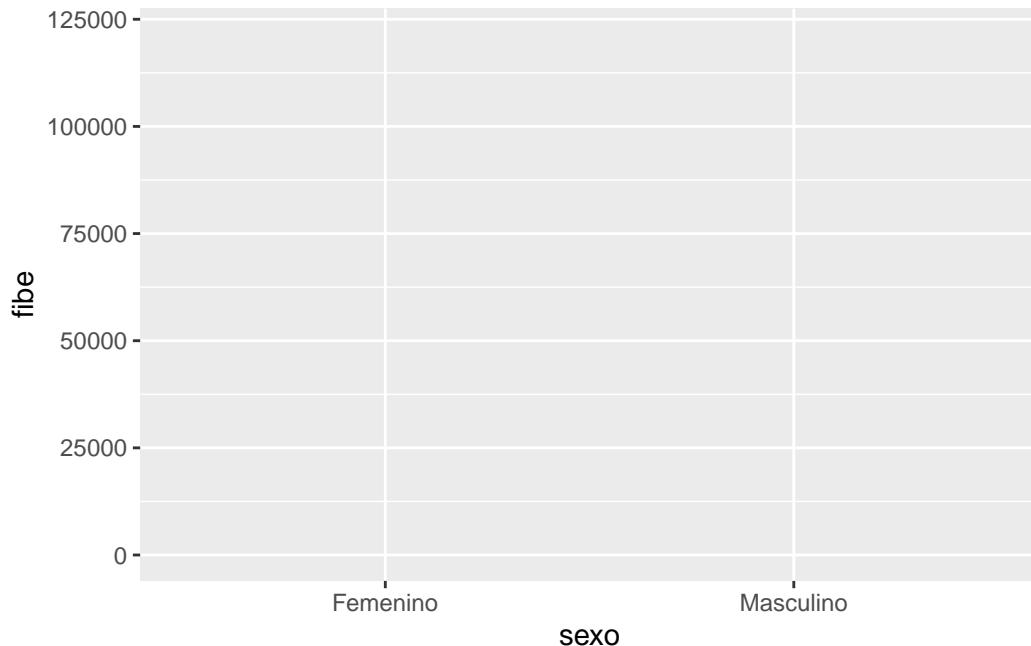
Aplicaremos esto a los datos importados. Referencia relevante para esta parte [Enlace acá](#)

1. Asignamos el set de datos:

```
library(tidyverse)  
  
p <- ggplot(data)  
  
p
```

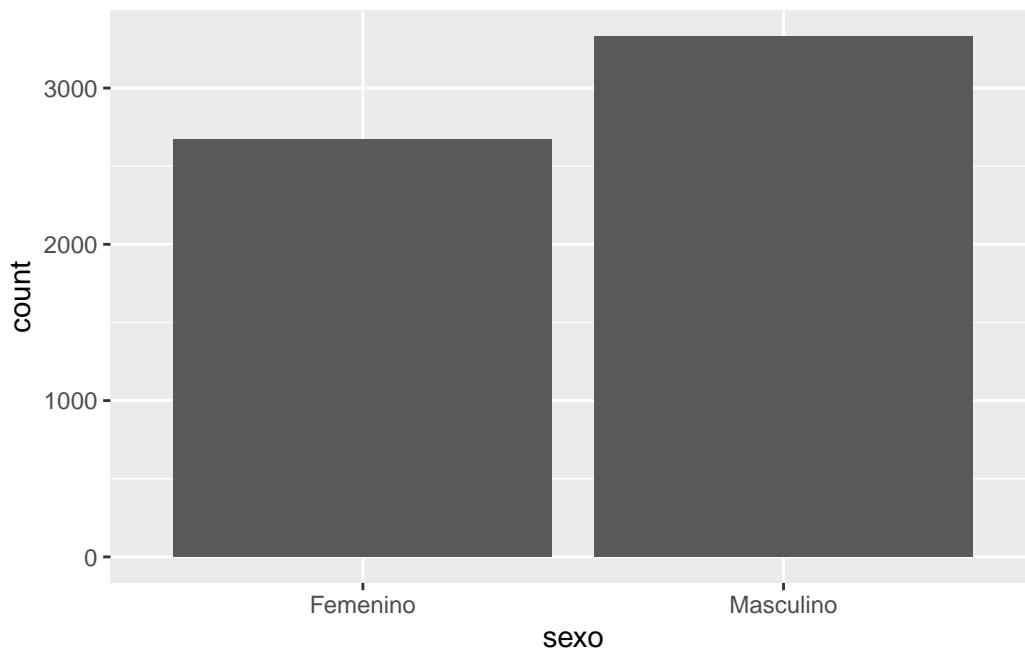
2. Ahora declaramos el sistema de coordenadas:

```
p <- ggplot(data) + aes(x = sexo, y = fibe)  
p
```



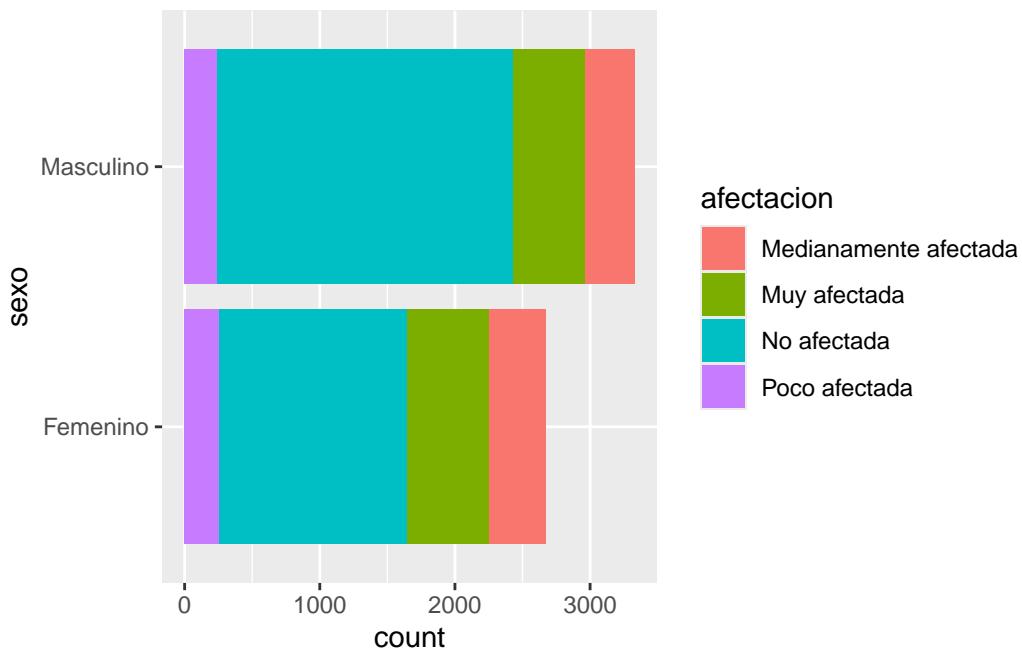
- Cabe mencionar, que cada elemento que se agrega dentro de un gráfico en GGPLOT2, se realiza utilizando el signo “+” (cómo ir añadiendo capas).
3. Declaremos ahora el tipo de figura, Por ejemplo, para un gráfico de barras el sistema de coordenadas solo debe contener el eje x (el eje y viene dado por el conteo de las ocurrencias):

```
p <- ggplot(data) + aes(x = sexo) + geom_bar()  
p
```



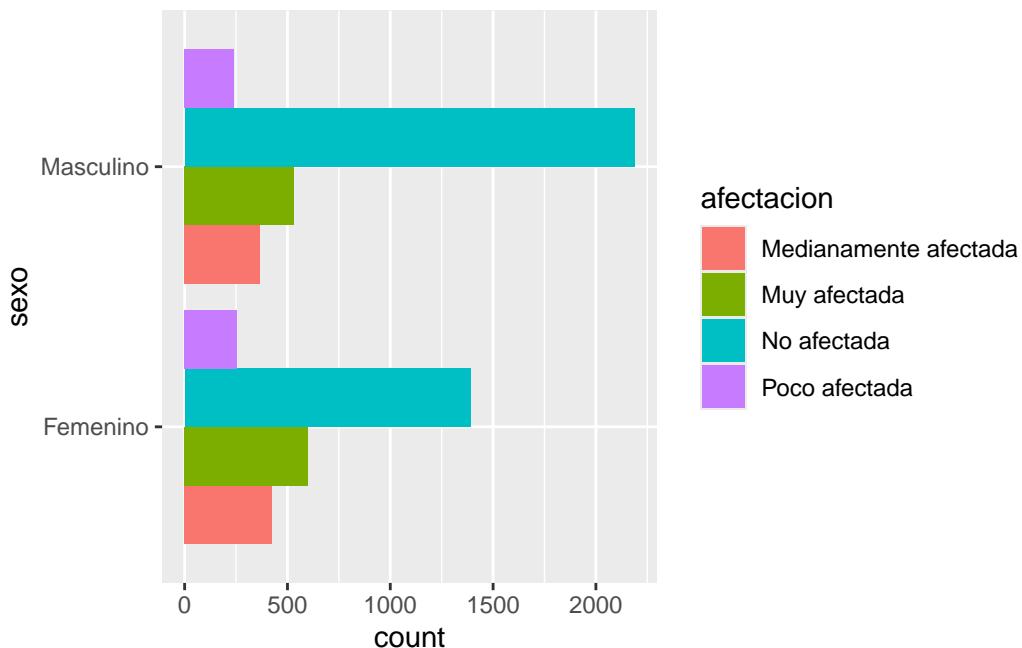
Podemos voltear el gráfico, y agregar diferenciación en base a otra variable:

```
p <- ggplot(data) + aes(y = sexo) + geom_bar(aes(fill = afectacion))  
p
```



Podríamos graficar de manera separada cada columna, en base a esta página [Enlace acá:](#)

```
p <- ggplot(data) + aes(y = sexo, fill = afectacion) +  
  geom_bar(position=position_dodge())  
  
p
```



## Actividad

Repliquemos los gráficos en el siguiente enlace.

[Enlace acá](#)

## Reflexiones sobre cómo visualizar

La visualización, es en parte un arte y en parte una ciencia (Wilke, 2019).

La relevancia de la visualización se genera como una aproximación visual resumida a los datos. El problema, es que no todas las visualizaciones son buenas, en otras palabras no todos los gráficos son construidos de manera correcta.

Un buen gráfico no responde necesariamente sobre el *¿cómo se ve?*, sino que debe responder a la necesidad de *¿quién lo ve?* y *¿por qué lo ve?*

## **Lo que no se debe hacer y lo que se debe hacer.**

Veamos un primer ejemplo:



Figure 1: Wilke (2019)

¿Qué le parece a usted? ¿es un buen o mal gráfico?

Otro ejemplo:

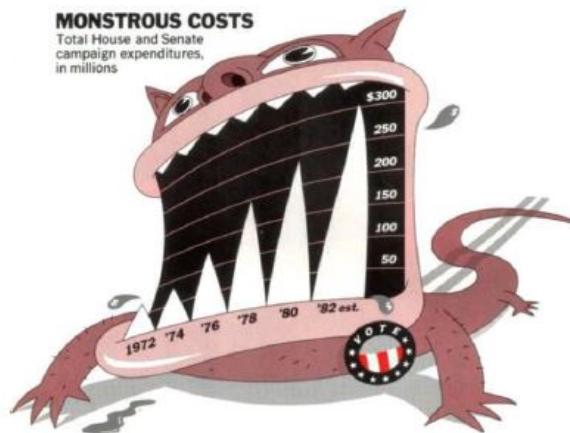


Figure 1.6: "Monstrous Costs" by Nigel Holmes (1982). Also a classic of its kind.

Figure 2: Wilke (2019)

## Feo, malo y malísimo (Wilke, 2019).

Un gráfico puede tener tres errores:

- Feo (Ugly): falla la estética, pudiendo ser claro e informativo.
- Malo (Bad): problemas de percepción en el gráfico, pudiendo ser poco claro, confuso o engañoso.
- Malísimo (Wrong): existen errores matemáticos detrás del gráfico.

De manera aplicada:

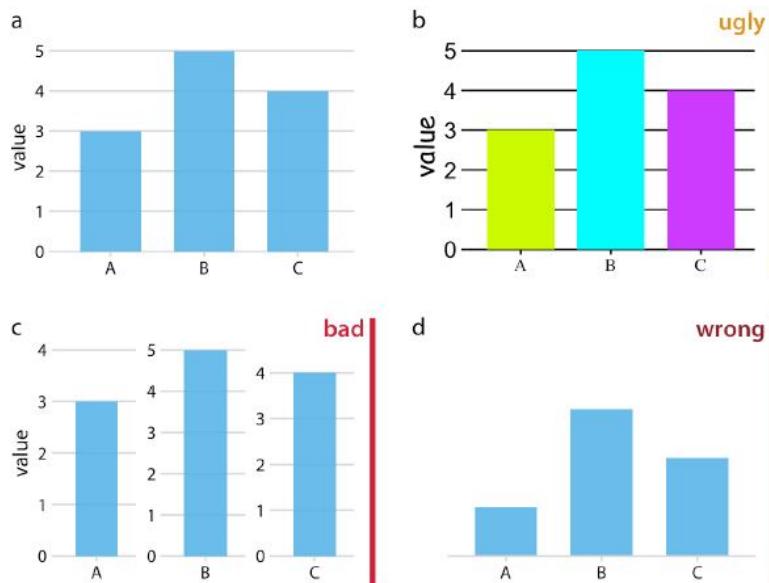


Figure 3: Wilke (2019)

Un ejemplo más realista de lo antes mencionado. Este gráfico, ¿está feo, malo o malísimo?

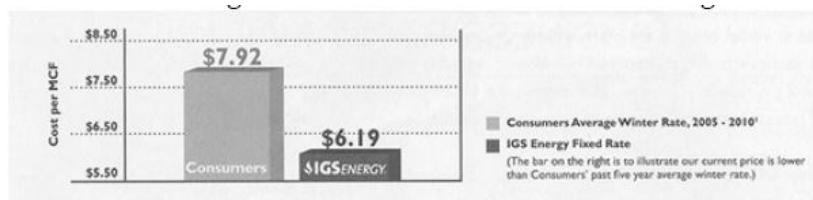


Figure 4: Evergreen (2019)

## La intencionalidad detrás del gráfico.

Dos maneras distintas de graficar lo mismo: una mejor que otra.

**Figure 1.1** Traditional clustered bar graphs can cloud the point.  
Parent vs. Student Perspectives

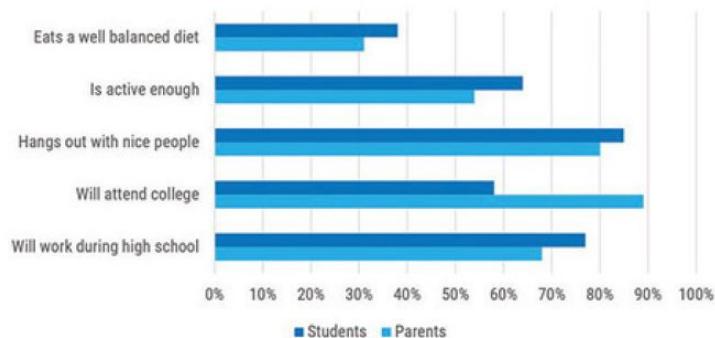


Figure 5: Evergreen (2019)

**Surprisingly, students have lower expectations to go to college than their parents have of them.**

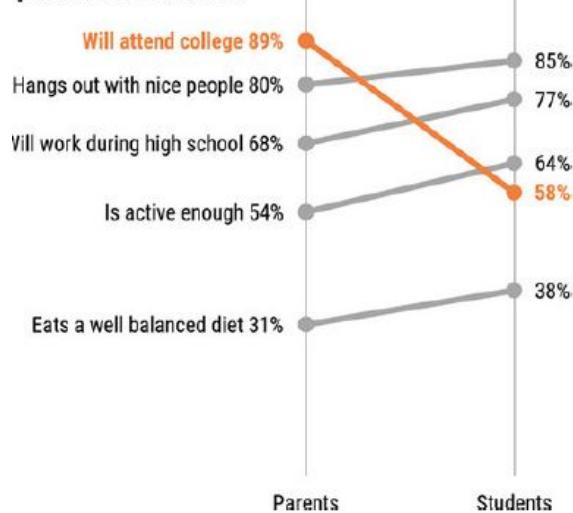


Figure 6: Evergreen (2019)

## La visualización ideal en seis imágenes (Roam, 2016).

1. Who and what is involved (¿Quién o qué está involucrado?): Se debe iniciar con un resumen visual sobre lo que se hablará.
2. How many are involved (¿Qué cantidad está involucrada?): Se deben generar medidas cuantitativas de lo hablado. Cambios en los números también son relevantes.
3. Where the pieces are located (¿Dónde se ubica?): Presentar alguna relación visual entre lo hablado y su ubicación.
4. When things occur (¿Cuándo ocurre?): Mostrar algo relacionado con la temporalidad o la secuencia de los eventos en las que ocurren las interacciones relevantes.
5. How things impact each other (¿Cómo las cosas se relacionan?): Generar una visualización que presente la relación causa - efecto que afectan lo mostrado anteriormente.
6. Why this matters (¿Por qué es importante?): Se debe concluir visualizado anteriormente.

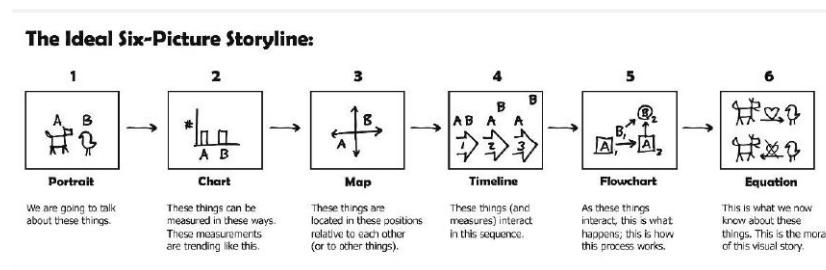


Figure 7: Evergreen (2019)

## Storytelling with data

Seis puntos claves:

1. Comprender el contexto
2. Elegir el gráfico adecuado
3. Eliminar el caos
4. Dirija la atención hacia donde le interese
5. Piense como un diseñador
6. Cuente una historia

En relación a este enfoque, se puede consultar el siguiente material:

[Página del libro](#)

## **Referencias**

- Wilke, C. O. (2019). Fundamentals of data visualization: a primer on making informative and compelling figures. O'Reilly Media.
- Evergreen, S. D. (2019). Effective data visualization: The right chart for the right data. SAGE publications.