

> 01

> Herramientas de análisis de datos

> Curso breve de técnicas modernas de  
análisis

 Material

 Twitter

Carlos A. Haro  
29/febrero/2020

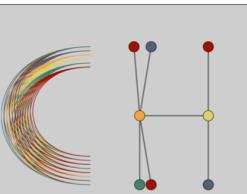
# Temario

## HERRAMIENTAS DE ANÁLISIS DE DATOS

### Curso breve de técnicas modernas de análisis

De una forma o de otra, es común enfrentarse con datos. Este curso pretende ser una introducción de las diversas herramientas disponibles para su análisis. Se cubrirá programación básica-intermedia en R y en Python, así como buenas prácticas para controlar versiones de código, datos, y modelos.

TEMARIO	
2 horas	<b>Introducción</b> Flujos de trabajo y pipelines Lenguajes de programación Editores de texto e IDEs Control de versiones Ambientes productivos Dataframes vs. databases
12 horas	<b>R</b> Editores e IDEs   <i>RStudio, VS Code, Jupyter</i> APIs de manejo de datos   <i>dplyr, data.table, base</i> Visualización de datos   <i>ggplot2, ggforce, ggraph</i> Modelos   <i>tidymodels</i> Comunicación de resultados   <i>RMarkdown, Shiny</i>
12 horas	<b>Python</b> Editores e IDEs   <i>Jupyter, VS Code</i> APIs de manejo de datos   <i>pandas</i> Visualización de datos   <i>seaborn, matplotlib</i> Modelos   <i>scikit-learn</i> Comunicación de resultados   <i>Jupyter</i>
5 horas	<b>Control de versiones</b> Código   <i>git, GitHub</i> Datos   <i>DVC</i> Modelos   <i>MLflow</i>
A disp. de tiempo	<b>Temas adicionales</b> Makefiles   <i>CNU Makefiles</i> Modelos en producción   <i>Docker, Flask, Kubernetes, Spark, unittest, testthis, plumber</i> Análisis de redes   <i>Neo4j</i>
INFORMACIÓN ADICIONAL	
	<b>Audiencia</b> Cualquier persona con interés o necesidad de trabajar con datos. Cada sesión comenzará desde cero e irá construyendo hacia ejemplos de mayor complejidad. Sin embargo, dada la disponibilidad de tiempo, el ritmo será acelerado.



## CONTACTO

- ✉ haro.ca@outlook.com
- 🐦 @haro.ca
- 🐙 github.com/haro-ca
- Ⓜ medium.com/@haro\_ca

## OBJETIVO

El objetivo del curso es proporcionar un panorama completo del ecosistema de trabajo con datos. Se expondrán los principales retos que es común enfrentar, así como una introducción a las herramientas necesarias para poder entregar un producto completo. Los temas cubiertos son, a mi consideración, los más importantes (e interesantes) para llevar a cabo un análisis de datos reproducible y en línea con buenas prácticas del ámbito. El curso no será suficiente para dar un conocimiento profundo, su intención es desarrollar un lenguaje mínimo para comenzar a practicar, indagar, y profundizar en el trabajo con datos. En ese sentido, este temario, las sesiones, y el material adicional, fungen como un compendio y una galería de técnicas.

Este temario fue realizado en R con [pagedown](#).

Última actualización 2020-02-29.

PDF en éste link

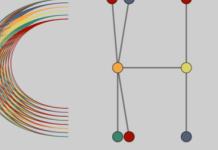
# Temario

## HERRAMIENTAS DE ANÁLISIS DE DATOS

### Curso breve de técnicas modernas de análisis

De una forma o de otra, es común enfrentarse con datos. Este curso pretende ser una introducción de las diversas herramientas disponibles para su análisis. Se cubrirá programación básica-intermedia en R y en Python, así como buenas prácticas para controlar versiones de código, datos, y modelos.

TEMARIO	
2 horas	<ul style="list-style-type: none"><li>Introducción<ul style="list-style-type: none"><li>Flujos de trabajo y pipelines</li><li>Lenguajes de programación</li><li>Editores de texto e IDEs</li><li>Control de versiones</li><li>Ambientes productivos</li><li>Dataframes vs. databases</li></ul></li></ul>
12 horas	<ul style="list-style-type: none"><li>R<ul style="list-style-type: none"><li>Editores e IDEs   <i>RStudio, VS Code, Jupyter</i></li><li>APIs de manejo de datos   <i>dplyr, data.table, base</i></li><li>Visualización de datos   <i>ggplot2, ggforce, ggraph</i></li><li>Modelos   <i>tidymodels</i></li><li>Comunicación de resultados   <i>RMarkdown, Shiny</i></li></ul></li></ul>
12 horas	<ul style="list-style-type: none"><li>Python<ul style="list-style-type: none"><li>Editores e IDEs   <i>Jupyter, VS Code</i></li><li>APIs de manejo de datos   <i>pandas</i></li><li>Visualización de datos   <i>seaborn, matplotlib</i></li><li>Modelos   <i>scikit-learn</i></li><li>Comunicación de resultados   <i>Jupyter</i></li></ul></li></ul>
5 horas	<ul style="list-style-type: none"><li>Control de versiones<ul style="list-style-type: none"><li>Código   <i>git, GitHub</i></li><li>Datos   <i>DVC</i></li><li>Modelos   <i>MLflow</i></li></ul></li></ul>
A disp. de tiempo	<ul style="list-style-type: none"><li>Temas adicionales<ul style="list-style-type: none"><li>Makefiles   <i>GNU Makefiles</i></li><li>Modelos en producción   <i>Docker, Flask, Kubernetes, Spark, unittest, testthis, plumber</i></li><li>Análisis de redes   <i>Neo4j</i></li></ul></li></ul>
INFORMACIÓN ADICIONAL	
<ul style="list-style-type: none"><li>Audiencia<ul style="list-style-type: none"><li>Cualquier persona con interés o necesidad de trabajar con datos.</li><li>Cada sesión comenzará desde cero e irá construyendo hacia ejemplos de mayor complejidad. Sin embargo, dada la disponibilidad de tiempo, el ritmo será acelerado.</li></ul></li></ul>	



CONTACTO

[haro.ca@outlook.com](mailto:haro.ca@outlook.com)  
[@haro.ca](https://twitter.com/haro_ca)  
[github.com/haro-ca](https://github.com/haro-ca)  
[medium.com/@haro\\_ca](https://medium.com/@haro_ca)

OBJETIVO

El objetivo del curso es proporcionar un panorama completo del ecosistema de trabajo con datos. Se expondrán los principales retos que es común enfrentar, así como una introducción a las herramientas necesarias para poder entregar un producto completo. Los temas cubiertos son, a mi consideración, los más importantes (e interesantes) para llevar a cabo un análisis de datos reproducible y en línea con buenas prácticas del ámbito.

El curso no será suficiente para dar un conocimiento profundo, su intención es desarrollar un lenguaje mínimo para comenzar a practicar, indagar, y profundizar en el trabajo con datos.

En ese sentido, este temario, las sesiones, y el material adicional, fungen como un compendio y una galería de técnicas.

Este temario fue realizado en R con [pagedown](#).

Última actualización 2020-02-29.

PDF en [éste link](#)

- +33 horas en total
- 4 lenguajes de programación (R, Python, Bash y SQL)
- 6 softwares
- +13 APIs

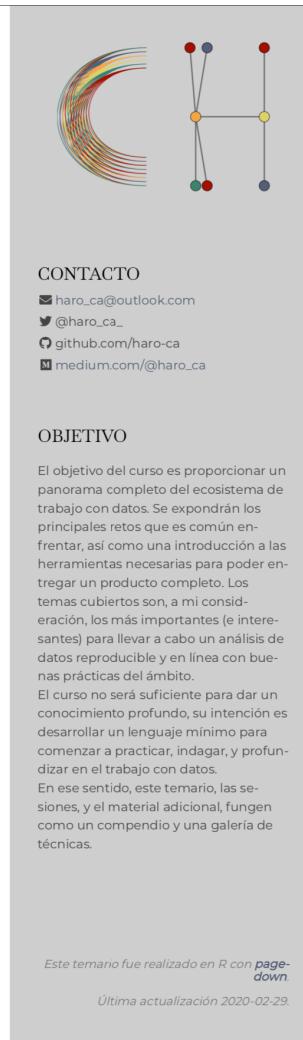
# Temario

## HERRAMIENTAS DE ANÁLISIS DE DATOS

### Curso breve de técnicas modernas de análisis

De una forma o de otra, es común enfrentarse con datos. Este curso pretende ser una introducción de las diversas herramientas disponibles para su análisis. Se cubrirá programación básica-intermedia en R y en Python, así como buenas prácticas para controlar versiones de código, datos, y modelos.

TEMARIO	
2 horas	<ul style="list-style-type: none"><li>Introducción<ul style="list-style-type: none"><li>Flujos de trabajo y pipelines</li><li>Lenguajes de programación</li><li>Editores de texto e IDEs</li><li>Control de versiones</li><li>Ambientes productivos</li><li>Dataframes vs. databases</li></ul></li></ul>
12 horas	<ul style="list-style-type: none"><li>R<ul style="list-style-type: none"><li>Editores e IDEs   RStudio, VS Code, Jupyter</li><li>APIs de manejo de datos   dplyr, data.table, base</li><li>Visualización de datos   ggplot2, ggeforce, ggraph</li><li>Modelos   tidyverse</li><li>Comunicación de resultados   RMarkdown, Shiny</li></ul></li></ul>
12 horas	<ul style="list-style-type: none"><li>Python<ul style="list-style-type: none"><li>Editores e IDEs   Jupyter, VS Code</li><li>APIs de manejo de datos   pandas</li><li>Visualización de datos   seaborn, matplotlib</li><li>Modelos   scikit-learn</li><li>Comunicación de resultados   Jupyter</li></ul></li></ul>
5 horas	<ul style="list-style-type: none"><li>Control de versiones<ul style="list-style-type: none"><li>Código   git, GitHub</li><li>Datos   DVC</li><li>Modelos   MLflow</li></ul></li></ul>
A disp. de tiempo	<ul style="list-style-type: none"><li>Temas adicionales<ul style="list-style-type: none"><li>Makefiles   GNU Makefiles</li><li>Modelos en producción   Docker, Flask, Kubernetes, Spark, unittest, testthis, plumber</li><li>Análisis de redes   Neo4j</li></ul></li></ul>
INFORMACIÓN ADICIONAL	
<ul style="list-style-type: none"><li>Audiencia<ul style="list-style-type: none"><li>Cualquier persona con interés o necesidad de trabajar con datos.</li><li>Cada sesión comenzará desde cero e irá construyendo hacia ejemplos de mayor complejidad. Sin embargo, dada la disponibilidad de tiempo, el ritmo será acelerado.</li></ul></li></ul>	



PDF en éste link

- +33 horas en total
- 4 lenguajes de programación (R, Python, Bash y SQL)
- 6 softwares
- +13 APIs

La intención es desarrollar un lenguaje mínimo para comenzar a practicar, indagar y profundizar en el trabajo con datos.

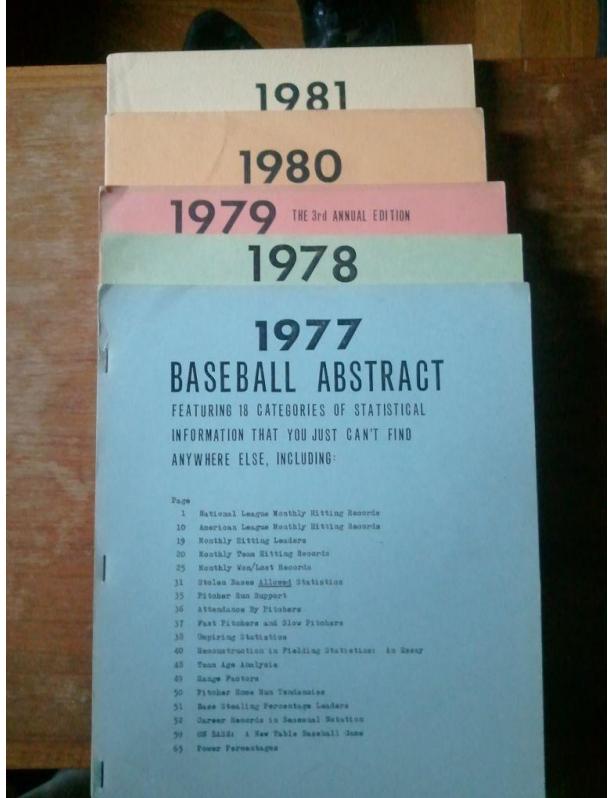
# Justificación

## ¿Por qué es necesario?

# Justificación

## ¿Por qué es necesario?

### 1. The Bill James Baseball Abstract

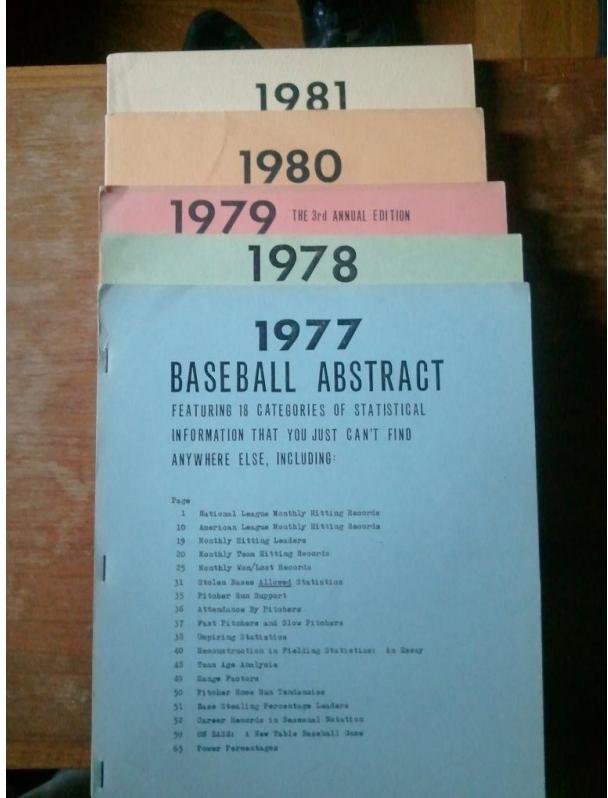


Page	
1	National League Monthly Hitting Records
10	American League Monthly Hitting Records
19	Monthly Hitting Leaders
20	Monthly Team Hitting Records
25	Monthly Win/Loss Records
31	Season Summarized Statistics
35	Player Record Holders
36	Attendance By Cities
37	Fast Pitches and Slow Pitches
38	Deceptive Statistics
40	Reinterpretation in Fielding Statistics: An Essay
49	Teen Age Analysis
49	Rage Pictures
50	Pitcher Home Run Testimonials
51	Base Stealing Percentage Leaders
52	Career Statistics in Seasonal Relation
59	Off Base: A New Table Baseball Game
65	Power Percentages

# Justificación

## ¿Por qué es necesario?

### 1. The Bill James Baseball Abstract

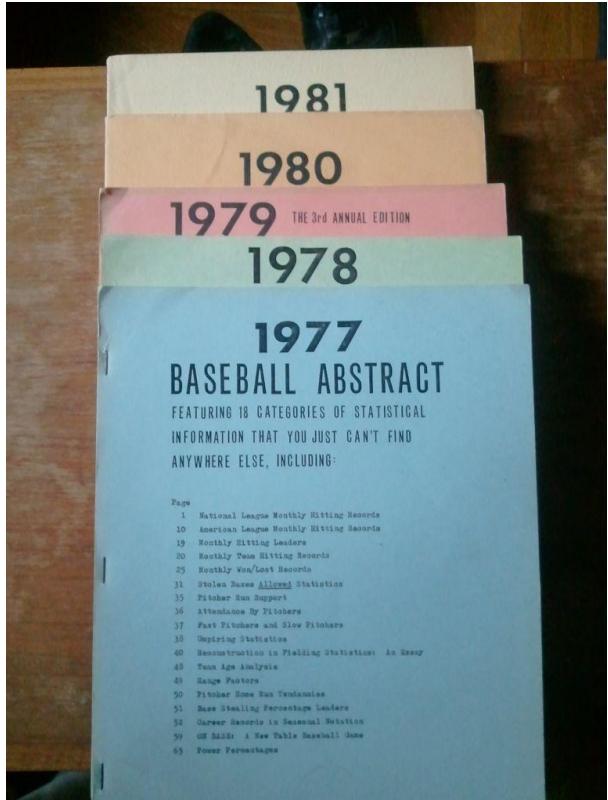


*Which pitchers and catchers allow runners to steal the most bases?*

# Justificación

## ¿Por qué es necesario?

### 1. The Bill James Baseball Abstract



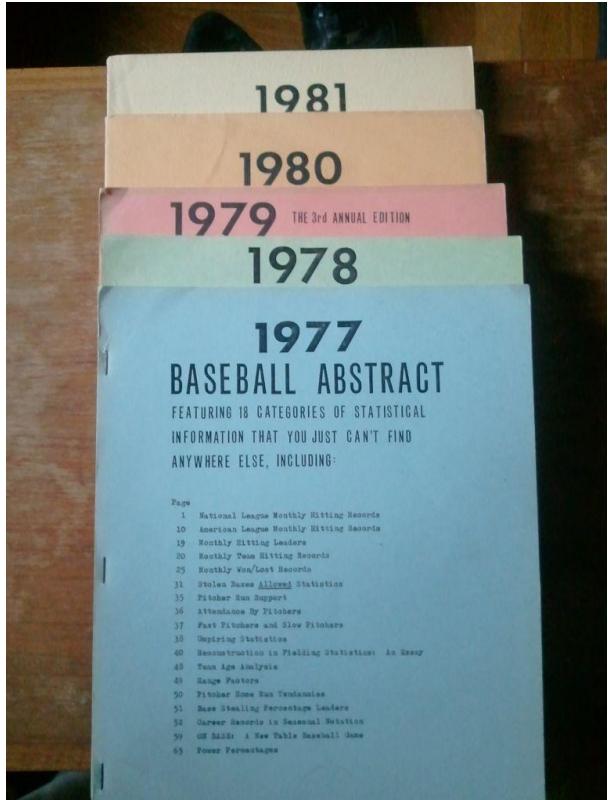
*Which pitchers and catchers allow runners to steal the most bases?*

"14-year old me was shocked to learn that most of what I've learned from baseball experts and insiders is entirely wrong"  
~ J.J. Allaire en la  
#rstudioconf2020  
(fundador de RStudio)

# Justificación

## ¿Por qué es necesario?

### 1. The Bill James Baseball Abstract



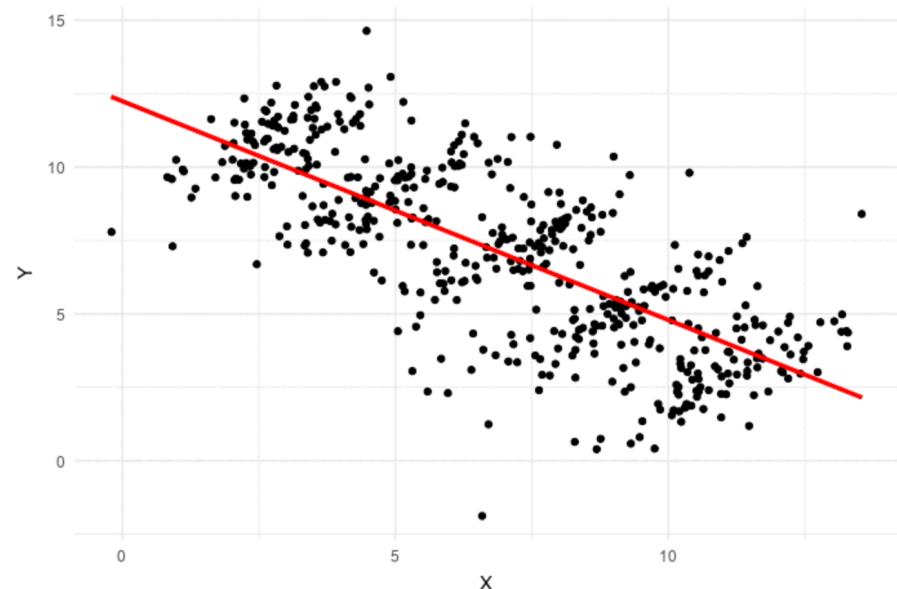
*Which pitchers and catchers allow runners to steal the most bases?*

"14-year old me was shocked to learn that most of what I've learned from baseball experts and insiders is entirely wrong"  
~ J.J. Allaire en la  
#rstudioconf2020  
(fundador de RStudio)

# Justificación

## ¿Por qué es necesario?

### 2. Simpson's paradox

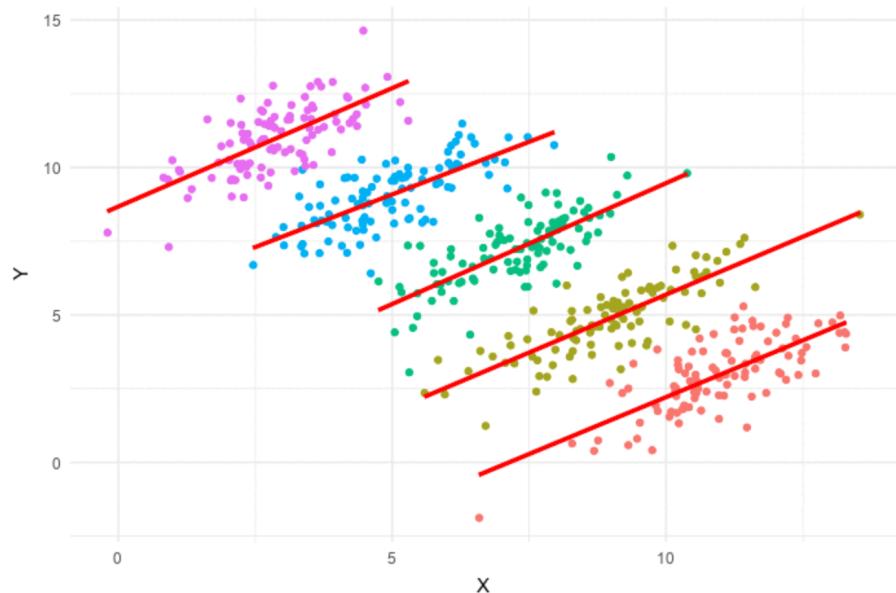


Al solo considerar una variable,  
la correlación parece negativa.

# Justificación

## ¿Por qué es necesario?

### 2. Simpson's paradox



Al considerar dos variables, la relación se invierte por completo

# Justificación

## ¿Por qué es necesario?

### 2. Simpson's paradox (ejemplo)

Intentas decidir a qué restaurante ir con tu novi@:  
Sophia's o Carlo's.

Tú le dices que en google Sophia's tiene mayor porcentaje de recomendaciones.

Tu novi@ te dice que tiene otros datos: "En Facebook tienen una encuesta en la que tanto hombres como mujeres prefieren Carlo's"

# Justificación

## ¿Por qué es necesario?

### 2. Simpson's paradox (ejemplo)

Intentas decidir a qué restaurante ir con tu novi@: Sophia's o Carlo's. Tú le dices que en google Sophia's tiene mayor porcentaje de recomendaciones. Tu novi@ te dice que tiene otros datos: "En Facebook tienen una encuesta en la que tanto hombres como mujeres prefieren Carlo's"

	Recommend Sophia's	Recommend Carlo's
Male	$\frac{50}{150} = 30\%$	$\frac{180}{360} = 50\%$
Female	$\frac{200}{250} = 80\%$	$\frac{36}{40} = 90\%$
Combined	$\frac{250}{400} = 62.5\%$	$\frac{216}{400} = 54\%$

# Justificación

## ¿Por qué es necesario?

### 2. Simpson's paradox (ejemplo)

Intentas decidir a qué restaurante ir con tu novi@: Sophia's o Carlo's. Tú le dices que en google Sophia's tiene mayor porcentaje de recomendaciones. Tu novi@ te dice que tiene otros datos: "En Facebook tienen una encuesta en la que tanto hombres como mujeres prefieren Carlo's"

	Recommend Sophia's	Recommend Carlo's
Male	$\frac{50}{150} = 30\%$	$\frac{180}{360} = 50\%$
Female	$\frac{200}{250} = 80\%$	$\frac{36}{40} = 90\%$
Combined	$\frac{250}{400} = 62.5\%$	$\frac{216}{400} = 54\%$

La perspectiva es importante

# Justificación

## ¿Por qué es necesario?

### 3. (GRAN) Etc.

Info muy resumida [aquí](#)

Un buen ejemplo de uso real [aquí](#)

Y mucho más en el material del temario.

.



# Herramientas

Calma. Nadie usa todas...

# Herramientas

Calma. Nadie usa todas...

Es una lista (demasiado extensa, y repetitiva) de herramientas (y compañías), no un checklist de requisitos.

# Herramientas

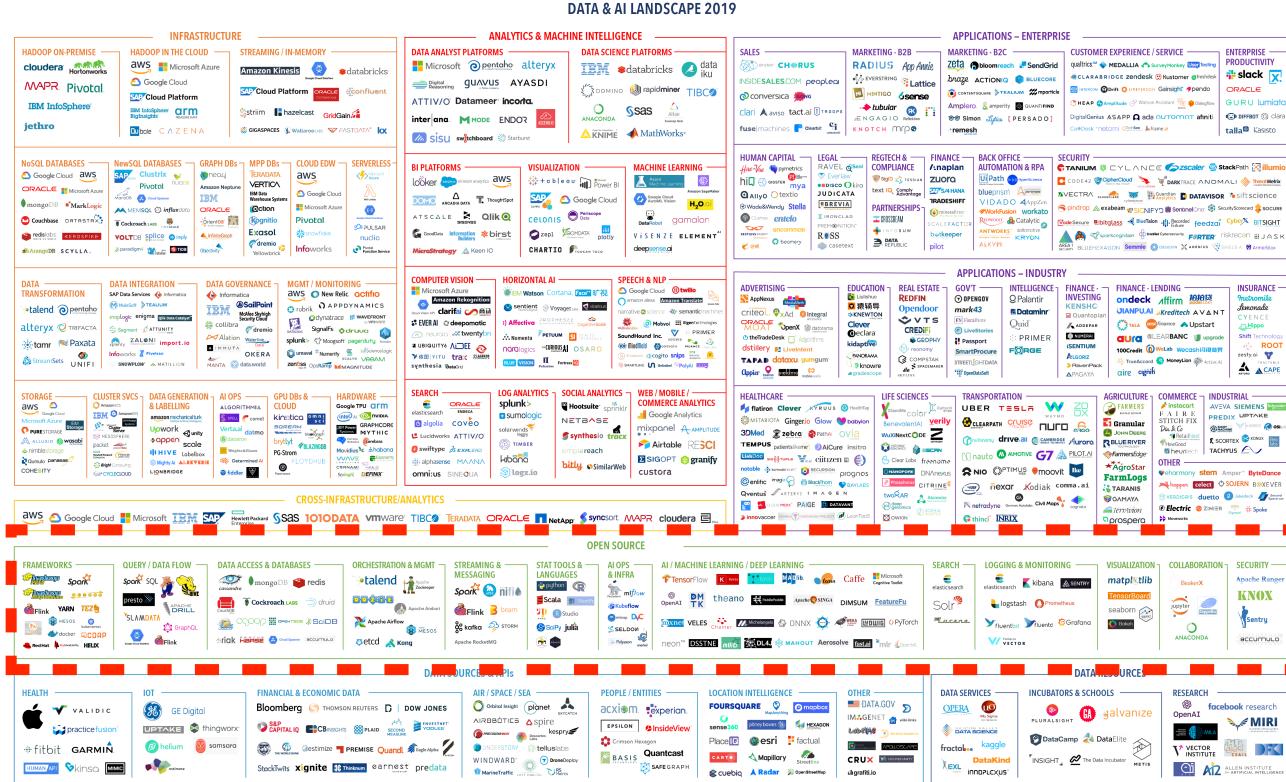
Calma. Nadie usa todas...

Es una lista (demasiado extensa, y repetitiva) de herramientas (y  
compañías), no un checklist de requisitos.

Muchas sirven el mismo propósito

# Herramientas

Pero, en lo que sí nos vamos a enfocar es en esta sección:



# Herramientas

## ¿Qué es open source?

# Herramientas

## ¿Qué es open source?

Definición de opensource.com:

*"Open source software is software with source code that anyone can inspect, modify, and enhance."*

# Herramientas

## ¿Qué es open source?

Definición de opensource.com:

*"Open source software is software with source code that anyone can inspect, modify, and enhance."*

## Ejemplos?

# Herramientas

## ¿Qué es open source?

Definición de opensource.com:

*"Open source software is software with source code that anyone can inspect, modify, and enhance."*

## Ejemplos?

Uno

.

# Herramientas

## ¿Qué es open source?

Definición de opensource.com:

*"Open source software is software with source code that anyone can inspect, modify, and enhance."*

## Ejemplos?

Uno (y es su cumpleaños! 🎂)

# Herramientas

## ¿Qué es open source?

Definición de opensource.com:

*"Open source software is software with source code that anyone can inspect, modify, and enhance."*

## Ejemplos?

Uno (y es su cumpleaños! 🎂),

Dos

.

# Herramientas

## ¿Qué es open source?

Definición de opensource.com:

*"Open source software is software with source code that anyone can inspect, modify, and enhance."*

## Ejemplos?

Uno (y es su cumpleaños! 🎂),

Dos (no es su cumpleaños)

.

# Herramientas

## ¿Qué es open source?

Definición de opensource.com:

*"Open source software is software with source code that anyone can inspect, modify, and enhance."*

## Ejemplos?

Uno (y es su cumpleaños! 🎂),

Dos (no es su cumpleaños),

Tres

.

# Herramientas

## ¿Qué es open source?

Definición de opensource.com:

*"Open source software is software with source code that anyone can inspect, modify, and enhance."*

## Ejemplos?

Uno (y es su cumpleaños! 🎂),

Dos (no es su cumpleaños),

Tres,

Cuatro

.

# Herramientas

## ¿Qué es open source?

Definición de opensource.com:

*"Open source software is software with source code that anyone can inspect, modify, and enhance."*

## Ejemplos?

Uno (y es su cumpleaños! 🎂),

Dos (no es su cumpleaños),

Tres,

Cuatro,

Cinco, Seis

# Herramientas

## ¿Qué es open source?

Definición de opensource.com:

*"Open source software is software with source code that anyone can inspect, modify, and enhance."*

## Ejemplos?

Uno (y es su cumpleaños! 🎂),

Dos (no es su cumpleaños),

Tres,

Cuatro,

Cinco, Seis,

Siete, Ocho, Nueve

# Herramientas

## ¿Qué es open source?

Definición de opensource.com:

*"Open source software is software with source code that anyone can inspect, modify, and enhance."*

## Ejemplos?

Uno (y es su cumpleaños! 🎂),

Dos (no es su cumpleaños),

Tres,

Cuatro,

Cinco, Seis,

Siete, Ocho, Nueve

Y mucho más: pandas, ggplot2, git, mlflow, docker, PostgreSQL

# Herramientas

*Solo veremos herramientas open source, porque:*

.

# Herramientas

*Solo veremos herramientas open source, porque:*

1.

.

# Herramientas

*Solo veremos herramientas open source, porque:*

1. 

# Herramientas

*Solo veremos herramientas open source, porque:*

1. 
- 2.

# Herramientas

*Solo veremos herramientas open source, porque:*

1. 
2. Viviran por siempre en un repositorio de código en la Antartica.

# Herramientas

*Solo veremos herramientas open source, porque:*

1. 
2. Viviran por siempre en un repositorio de código en la Antartica.  
En serio! [Véan](#)

# Herramientas

*Solo veremos herramientas open source, porque:*

1. 
2. Viviran por siempre en un repositorio de código en la Antartica.  
En serio! [Véan](#)
3. **Todas las empresas, y todo el análisis, se está moviendo hacia allá.**

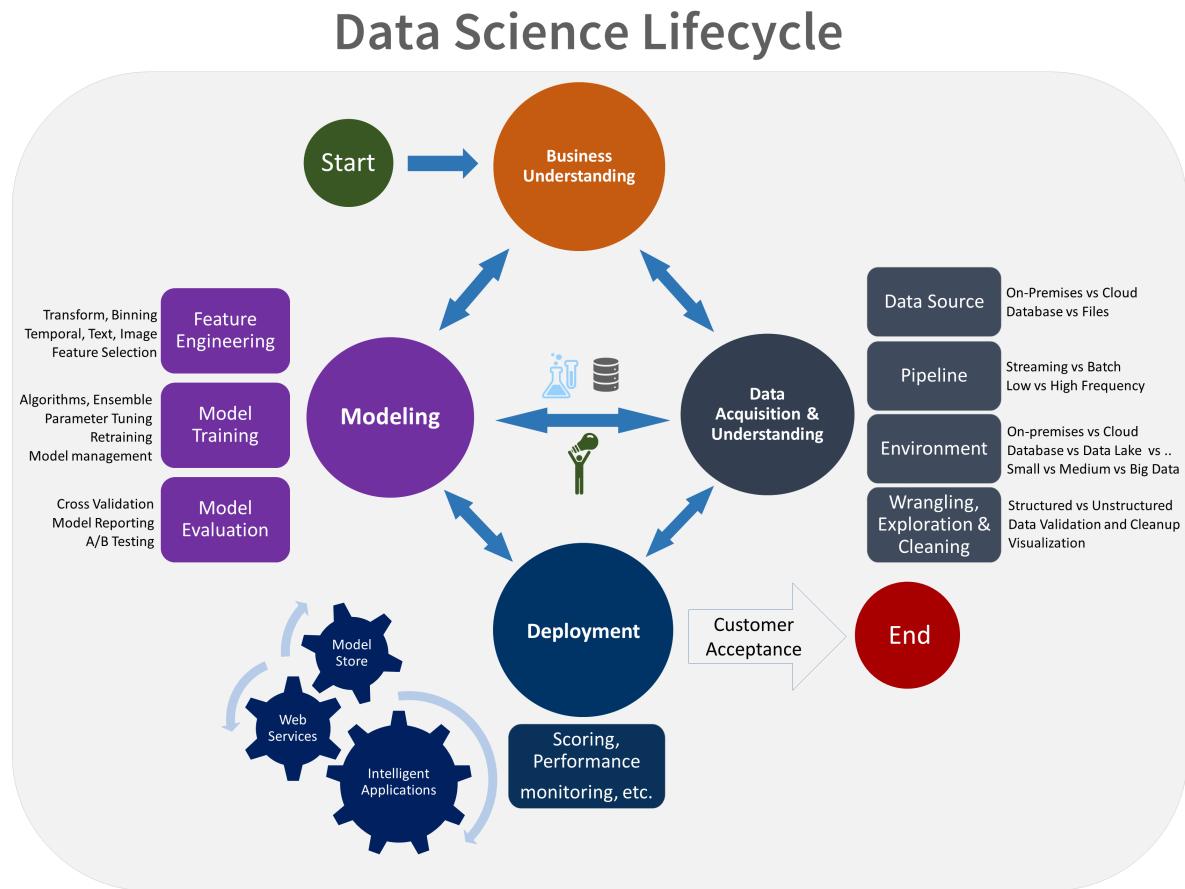
# Herramientas

*Solo veremos herramientas open source, porque:*

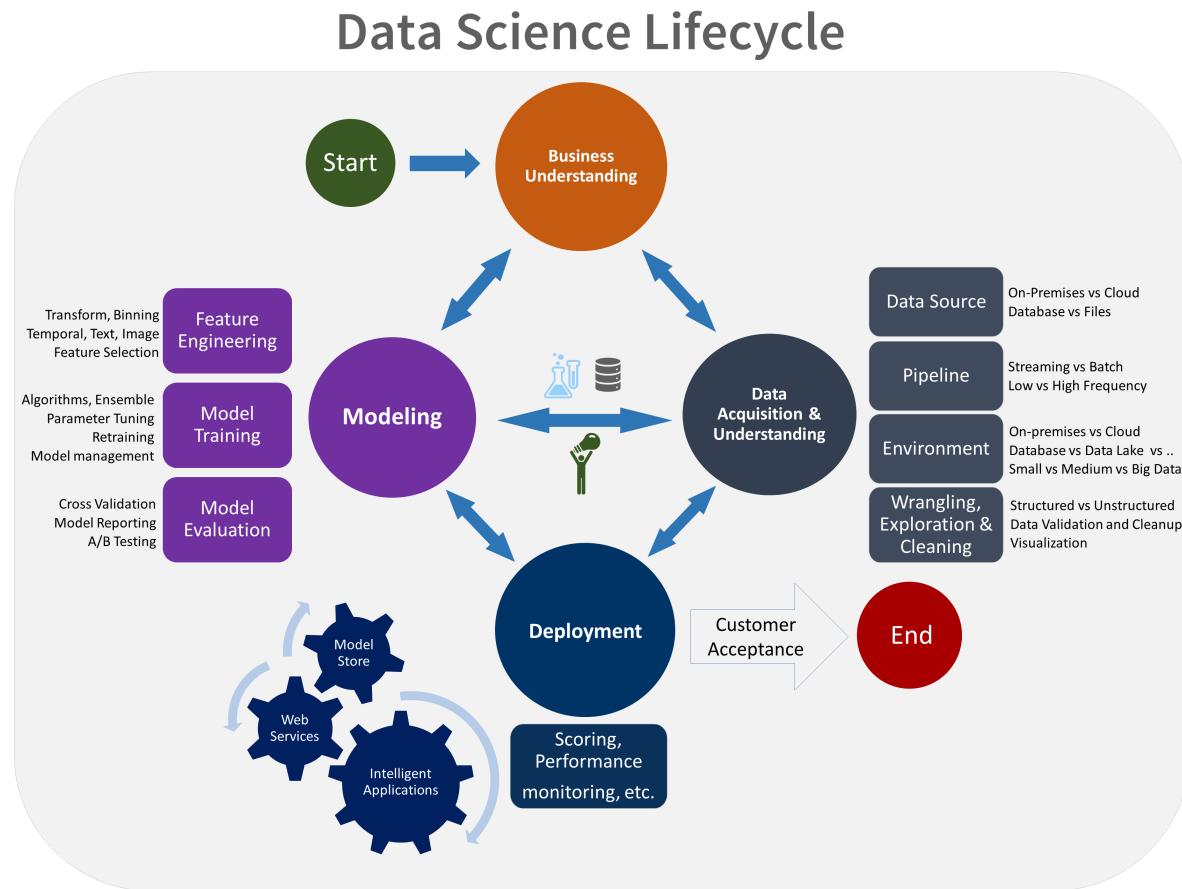
1. 
2. Viviran por siempre en un repositorio de código en la Antartica.  
En serio! [Véan](#)
3. **Todas las empresas, y todo el análisis, se está moviendo hacia allá.**  
[Evidencia aquí](#)

.

# Workflow



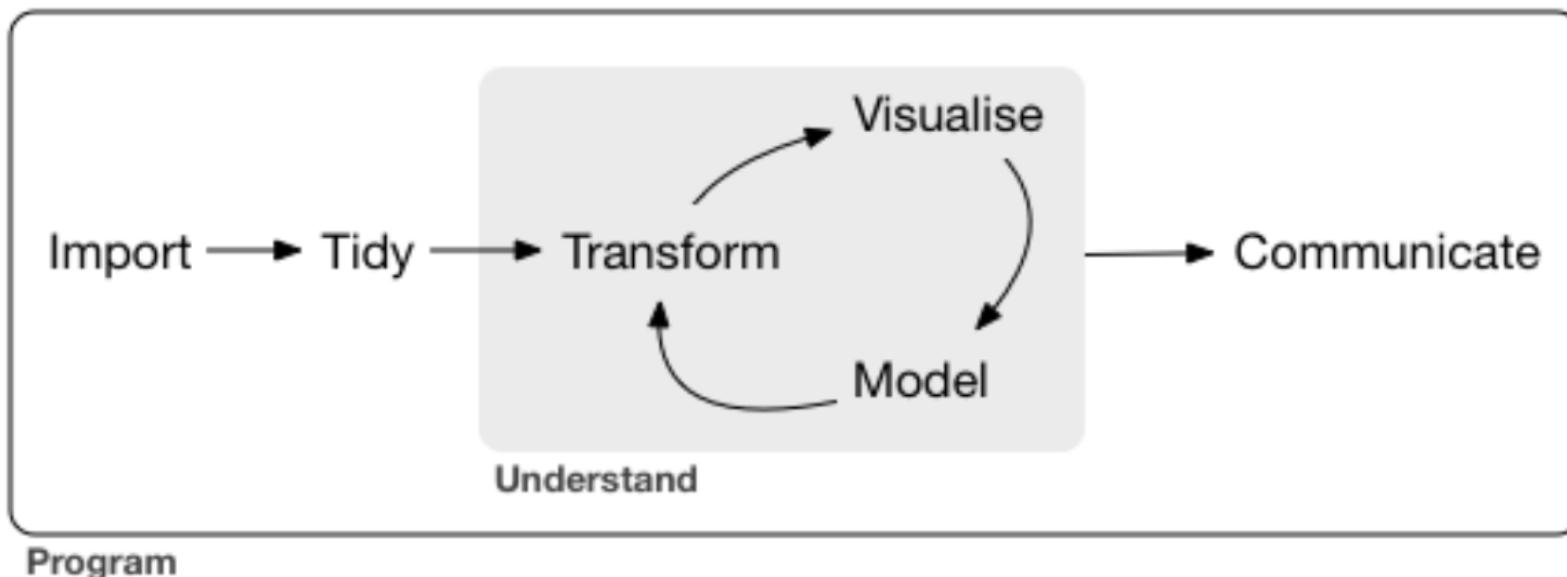
# Workflow



También es conocido como *pipeline*.

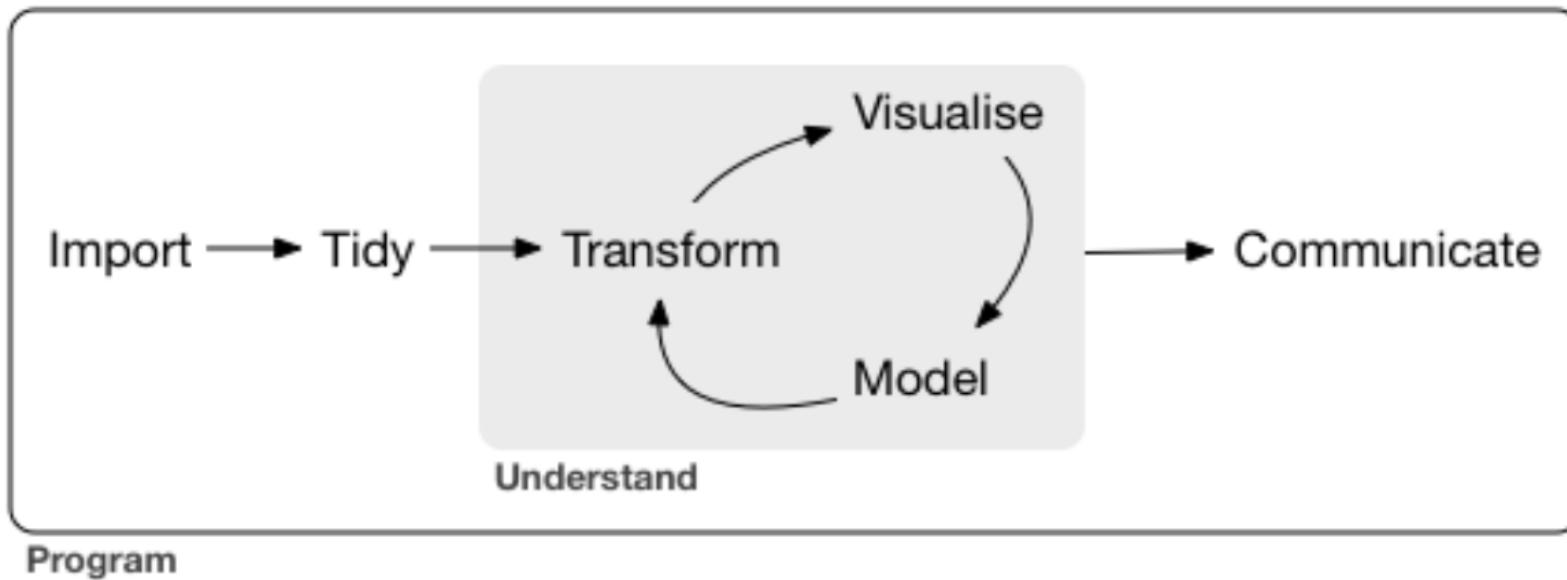
# Workflow

Pero comencemos más sencillo:



# Workflow

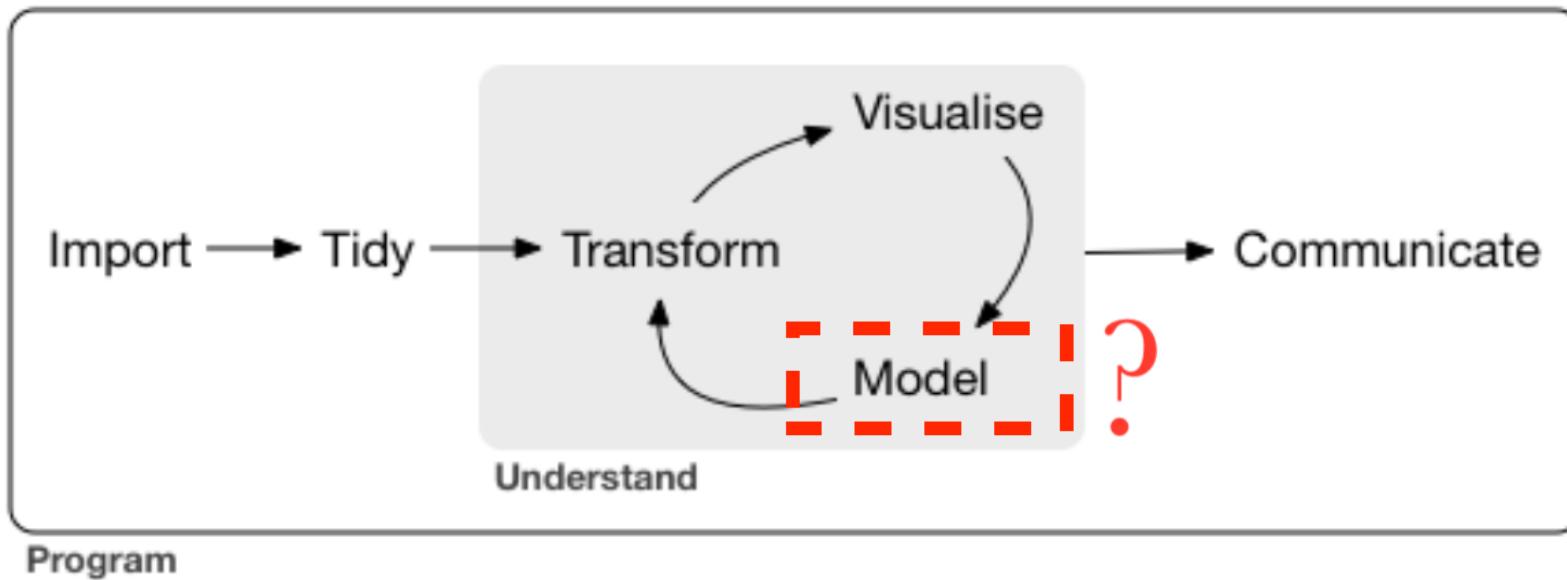
Pero comencemos más sencillo:



¿Qué parte es la que parece la más emocionante/interesante?

# Workflow

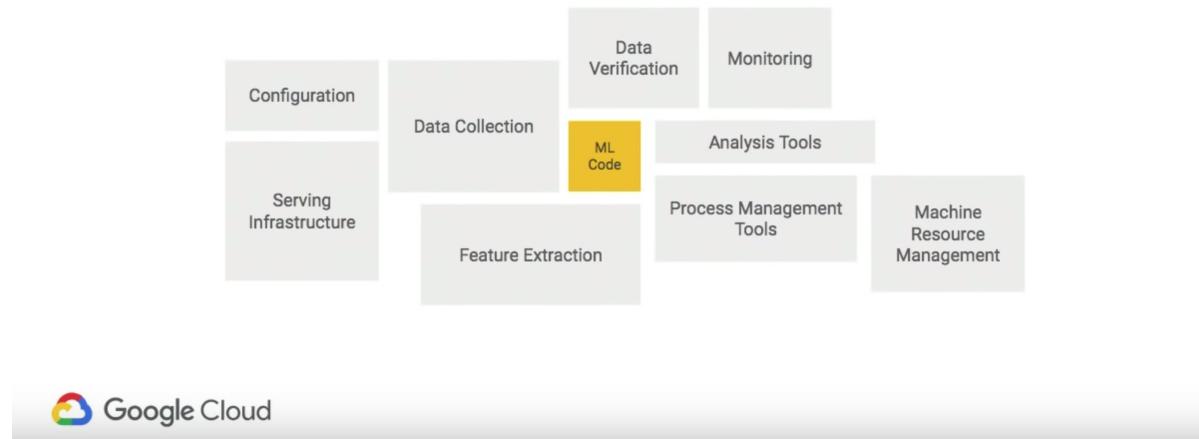
Pero comencemos más sencillo:



¿Qué parte es la que parece la más emocionante/interesante?

# Disclaimer

La parte importante de este curso no será hacer modelos elaborados de ML; tanto en academia, como en industria, ML es una parte muy pequeña.



Tener buenos datos y un buen control sobre ellos es lo que lleva a un buen análisis.

# A programar!

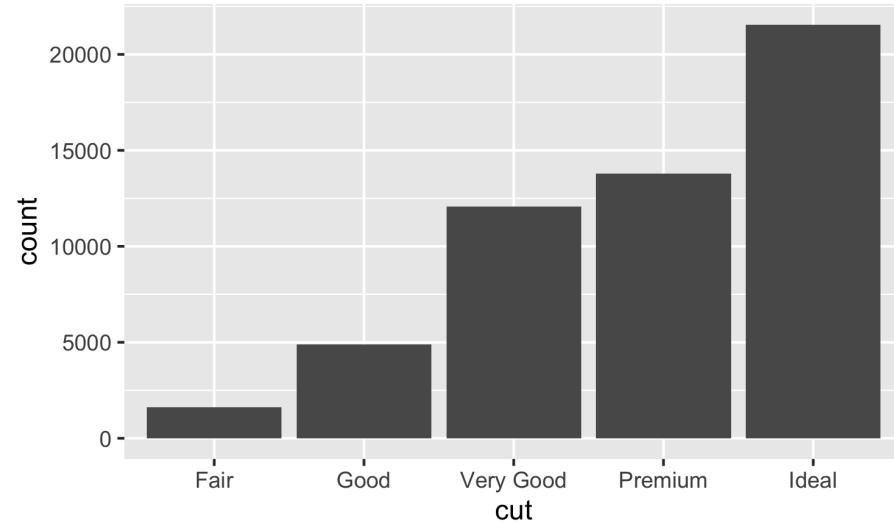
# Esquema de una gráfica de ggplot

**GGPLOT + AESthetics + GEOM**

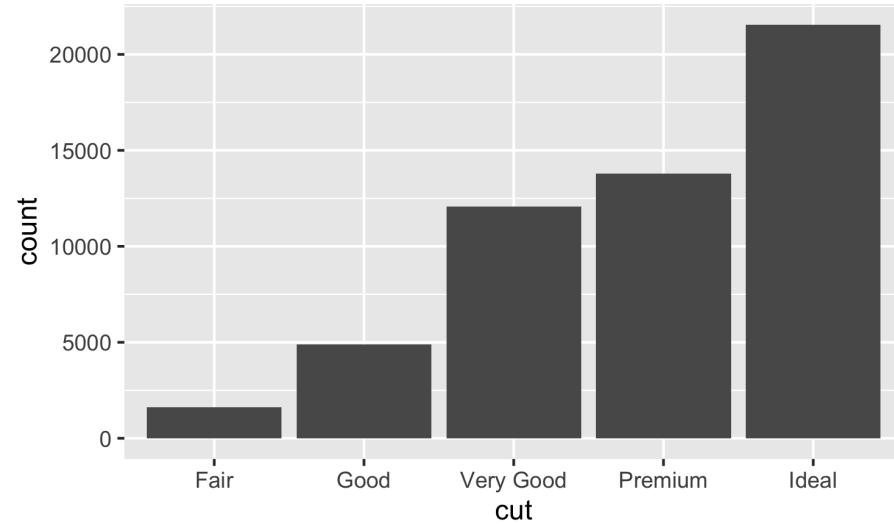
GGPLOT := la inicialización de la gráfica

AESthetics := ¿Qué variables veremos en la gráfica?

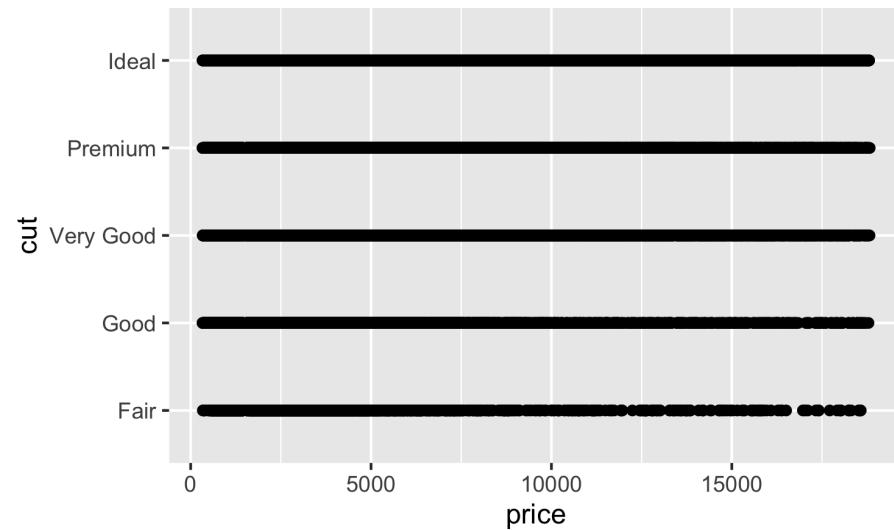
GEOM := ¿Cómo veremos esas variable en la gráfica? En forma de puntos, barras, líneas, etc



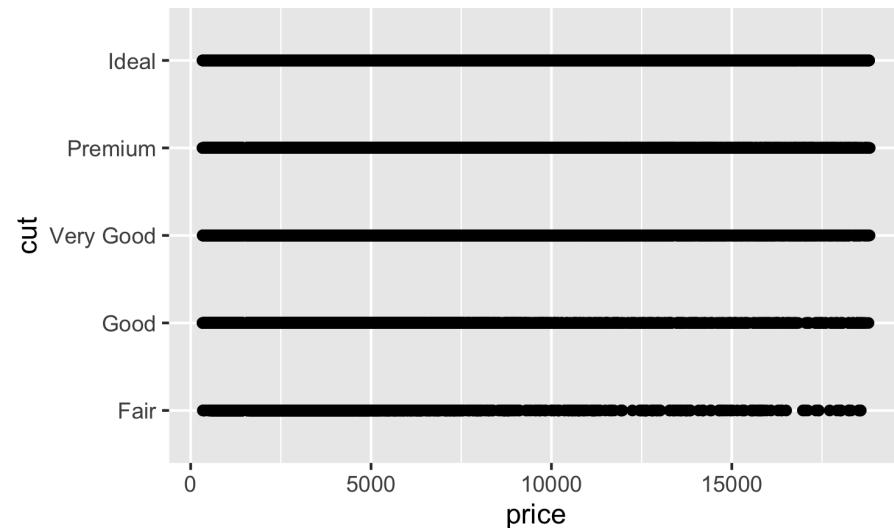
04 : 00



```
library(ggplot2)
ggplot(diamonds) +
  aes(x = cut) +
  geom_bar()
```



04 : 00



```
library(ggplot2)
ggplot(diamonds) +
  aes(x = price, y = cut) +
  geom_point()
```