Visualización de datos con ggplot2

Diego Delgado Palomares

6/12/2020

La visualización.

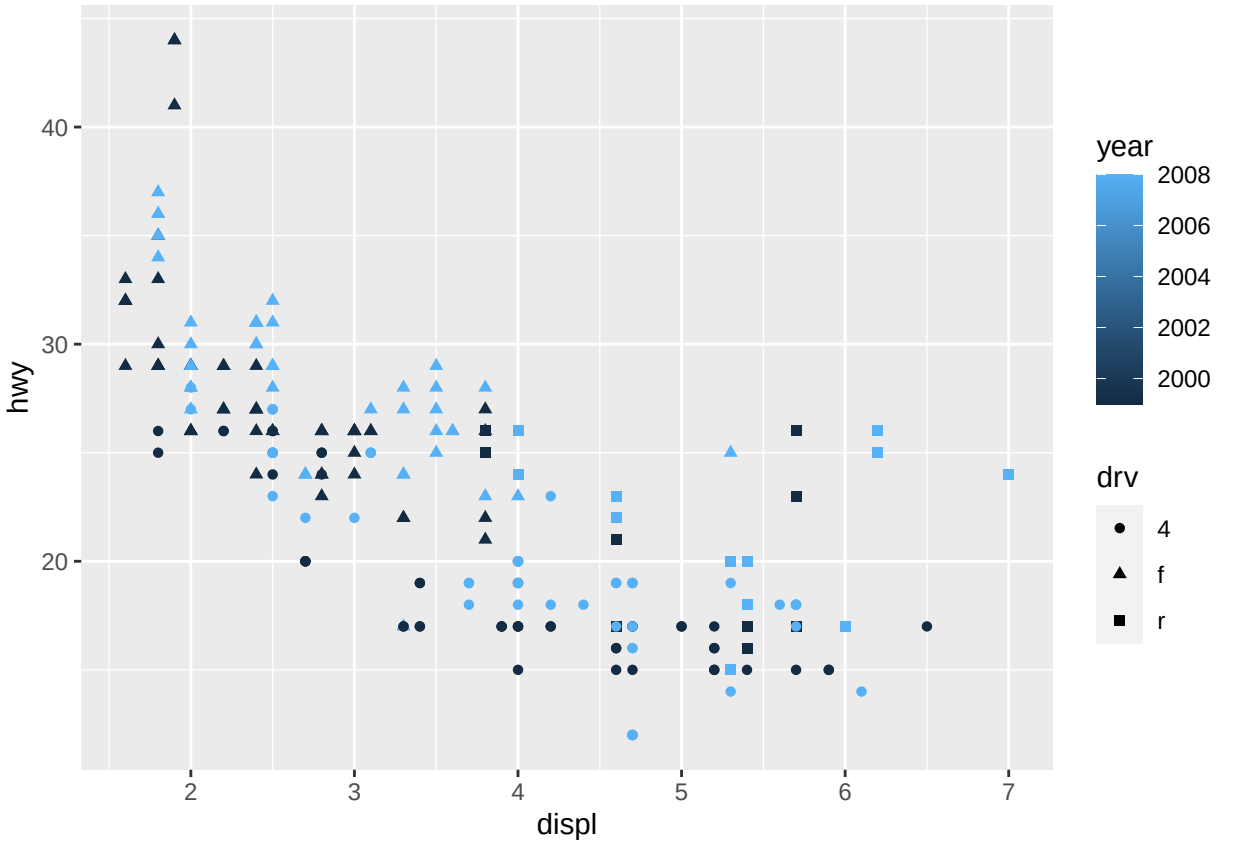
Para comenzar, vamos a trabajar con el dataset `"mpg"`, el cual contiene 234 observaciones a diferentes tipos de autómovil.

- Las Variables (incluyendo algunos datos dentro de estas)
  - manufacturer: manufacturer name
    - audi, chevrolet, dodge, ford, honda, hyundai, jeep, land, rover, lincoln, mercury, nissan, pontiac, subaru, toyota, volkswagen.
  - model: model name
  - displ: engine displacement, in litres
    - entre 1.6 y 7.
  - year: year of manufacture
    - 1999 , 2008
  - cyl: number of cylinders
    - 4, 5, 6, 8.
  - trans: type of transmission
  - drv: the type of drive train
    - f = front-wheel drive, r = rear wheel drive, 4 = 4wd
  - cty: city miles per gallon
    - 9, de 11 a 33, 35.
  - hwy: highway miles per gallon
    - 12, de 14 a 37, 41, 44.
  - fl: fuel type
    - c, d, e, p, r.
  - class: "type" of car
    - 2seater, compact, midsize, minivan, pickup, subcompact, suv.

Ya con los datos expuestos, podemos comenzar el análisis gráfico con ggplot2.

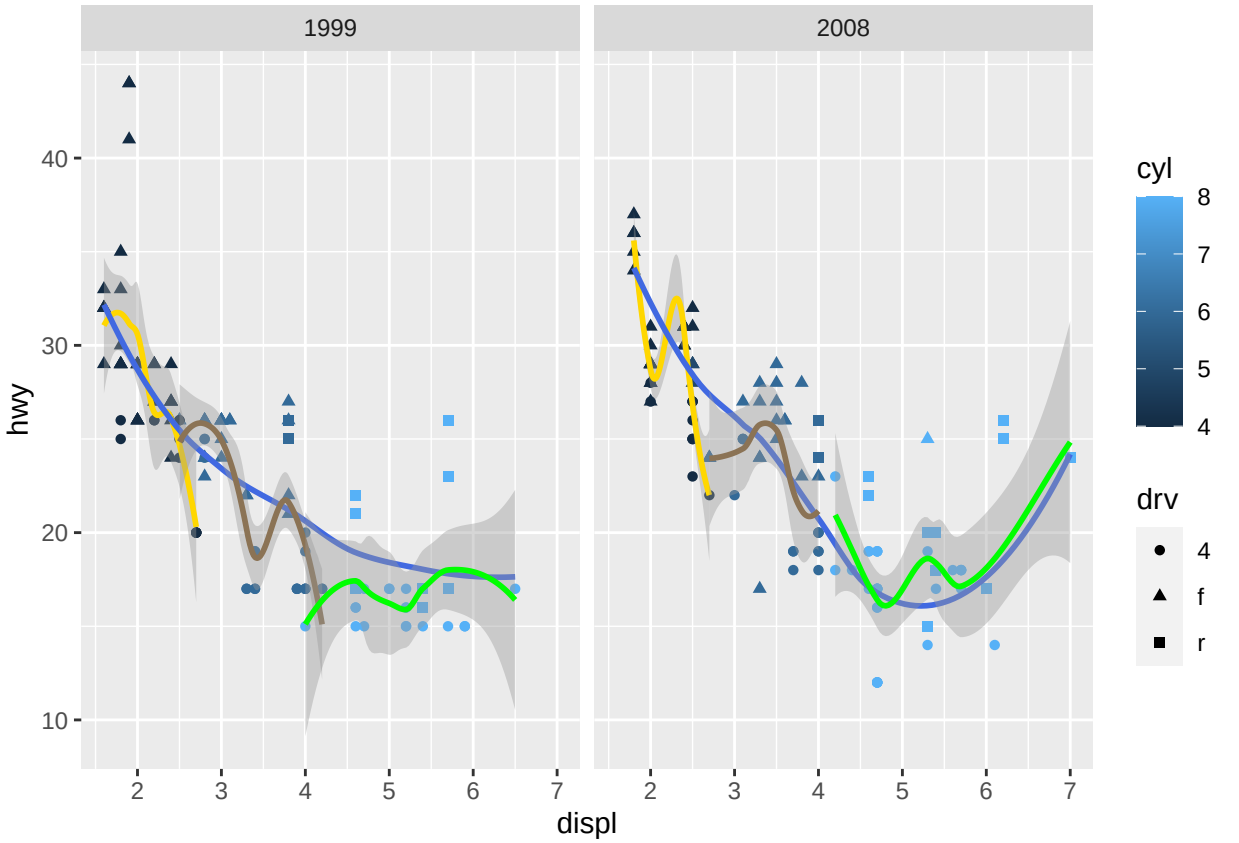
```
library(tidyverse)
library(ggplot2)

# un gráfico que presenta la relación entre rendimiento de combustible  en carretera y la capacidad en
```

```
ggplot(mpg) +
  geom_point(mapping = aes(x = displ, y = hwy, colour = year, shape = drv))
```



El gráfico anterior nos deja observar que los vehículos con un mayor volumen de motor, tienen en general un menor rendimiento de gasolina.

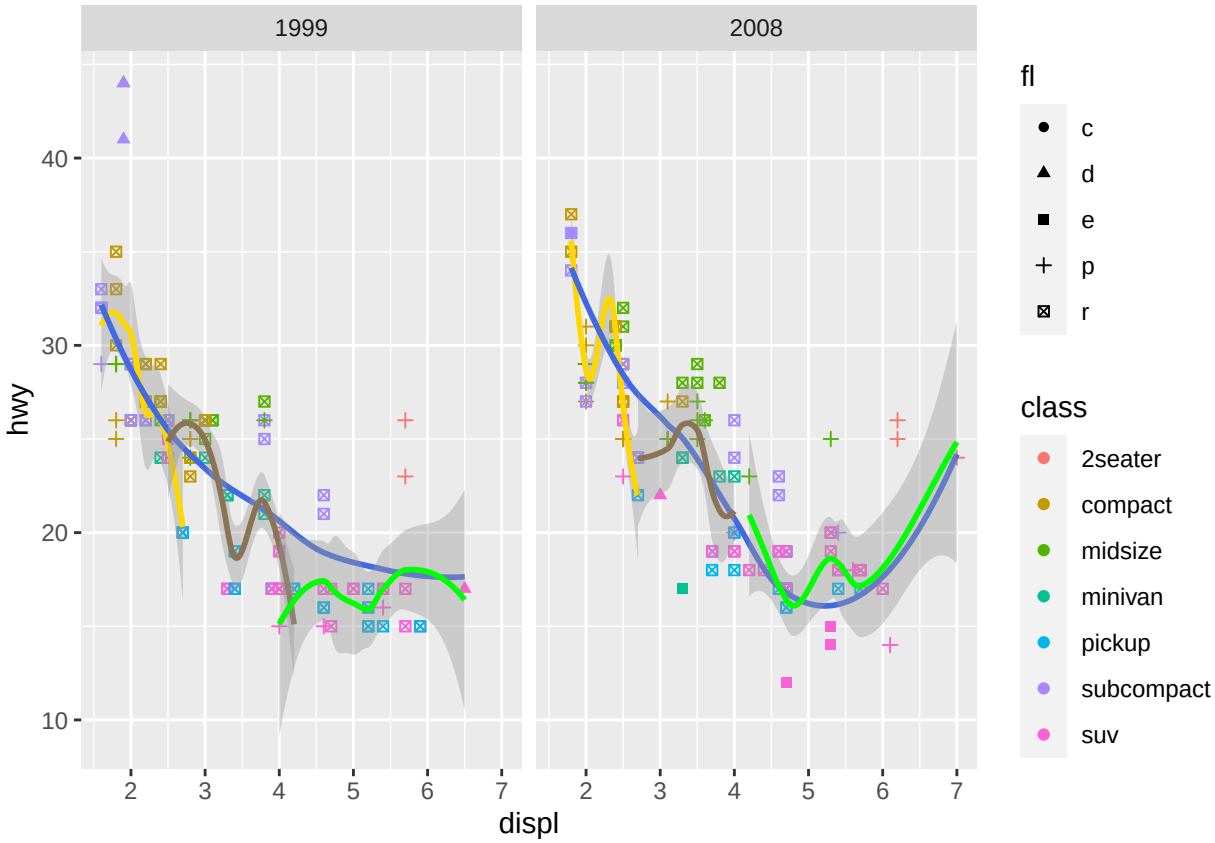
```
ggplot(mpg) +
  geom_point(mapping = aes(x = displ, y = hwy, colour = cyl, shape = drv)) +
  geom_smooth(mapping = aes(x = displ, y = hwy), data = filter( mpg, cyl == 4),
              colour="gold") +
  geom_smooth(mapping = aes(x = displ, y = hwy),
              data = filter( mpg, cyl == c(4,5,6,8)),
              colour="royalblue", se=F) +
  geom_smooth(mapping = aes(x = displ, y = hwy),
              data = filter( mpg, cyl == 6), colour="burlywood4") +
  geom_smooth(mapping = aes(x = displ, y = hwy),
              data = filter( mpg, cyl == 8), colour="green") +
  facet_wrap(~year)
```



El código anterior nos presenta dos gráficos, podemos ver la relación entre los cilindros, parece ser que los autos con menor cilindraje tienen mayor rendimiento (algo perceptible con el gráfico anterior, pues a mayor volumen de motor, se usan mas cilindros).

Podemos ver más relaciones, por ejemplo los autos con mayor potencia,

```
ggplot(mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(colour = class , shape = fl)) +
  geom_smooth(data = filter( mpg, cyl == 4), colour="gold") +
  geom_smooth(data = filter( mpg, cyl == c(4,5,6,8)),
              colour="royalblue", se=F) +
  geom_smooth(data = filter( mpg, cyl == 6), colour="burlywood4") +
  geom_smooth(data = filter( mpg, cyl == 8), colour="green") +
  facet_wrap(~year)
```



las posibilidades son bastantes, aunque este dataset esta muy limitado. por ello cambiaremos a uno con una cantidad bastante superior de datos.

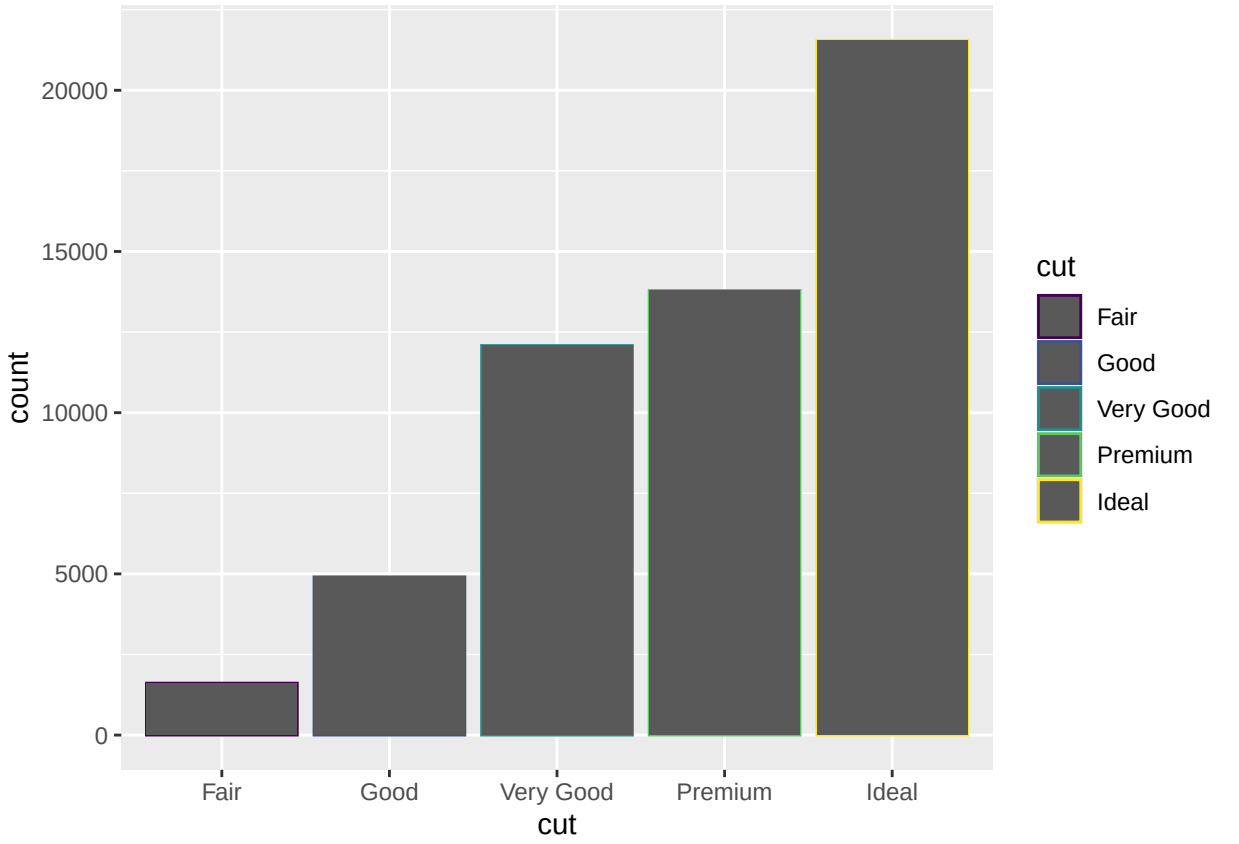
Diamantes

El dataset diamonds cuenta con una muestra de 53940 diamantes.

- Las Variables
  - price
    - price in US dollars ($326–$18,823)
  - carat
    - weight of the diamond (0.2–5.01)
  - cut
    - quality of the cut (Fair, Good, Very Good, Premium, Ideal)
  - color
    - diamond colour, from D (best) to J (worst)
  - clarity
    - a measurement of how clear the diamond is (I1 (worst), SI2, SI1, VS2, VS1, VVS2, VVS1, IF (best))
  - x
    - length in mm (0–10.74)
  - y
    - width in mm (0–58.9)
  - z
    - depth in mm (0–31.8)
  - depth

    * total depth percentage = z / mean(x, y) = 2 * z / (x + y) (43–79)
  - table
  - width of top of diamond relative to widest point (43–95)

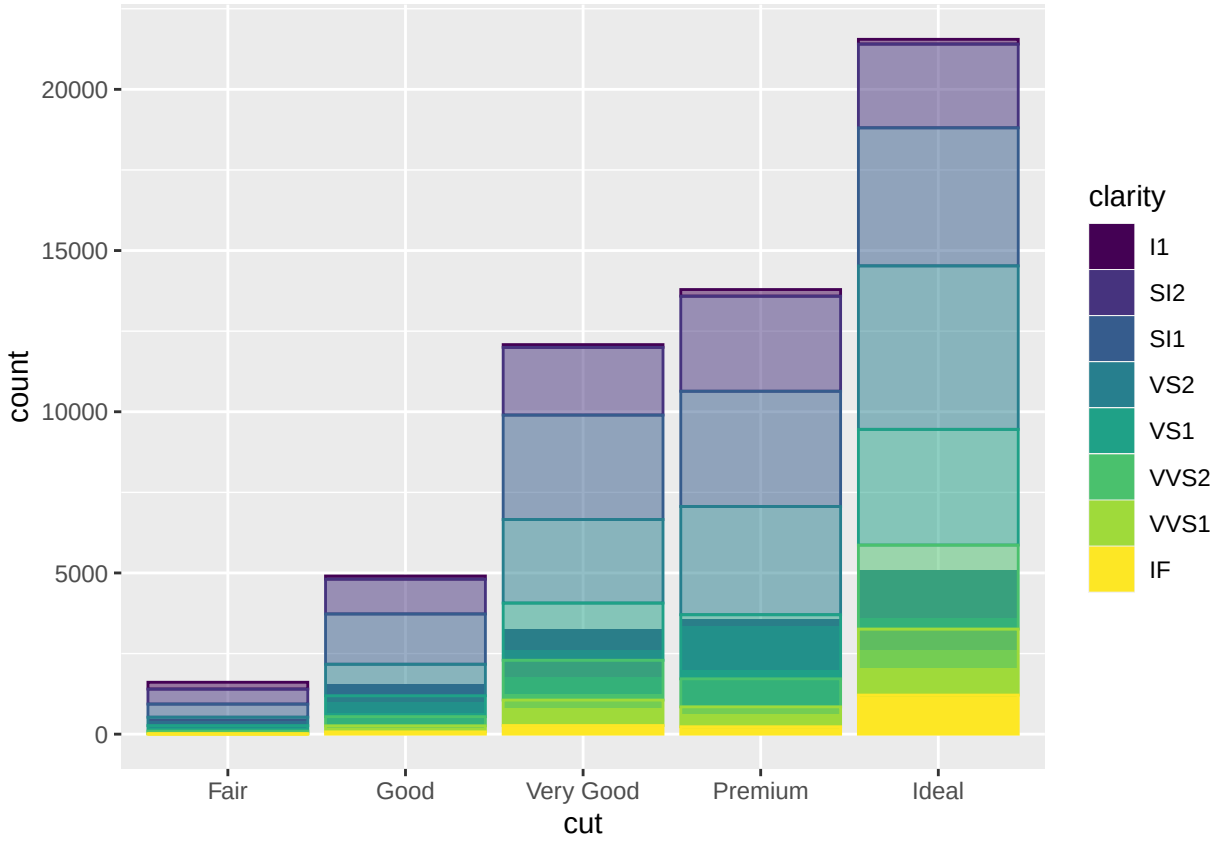
```
ggplot(diamonds) +
  geom_bar(mapping = aes(x = cut, colour = cut)) +
  stat_count(mapping =  aes(x=cut))
```



Esteticas

Aprenderemos el uso de las estéticas con los siguientes ejemplos.

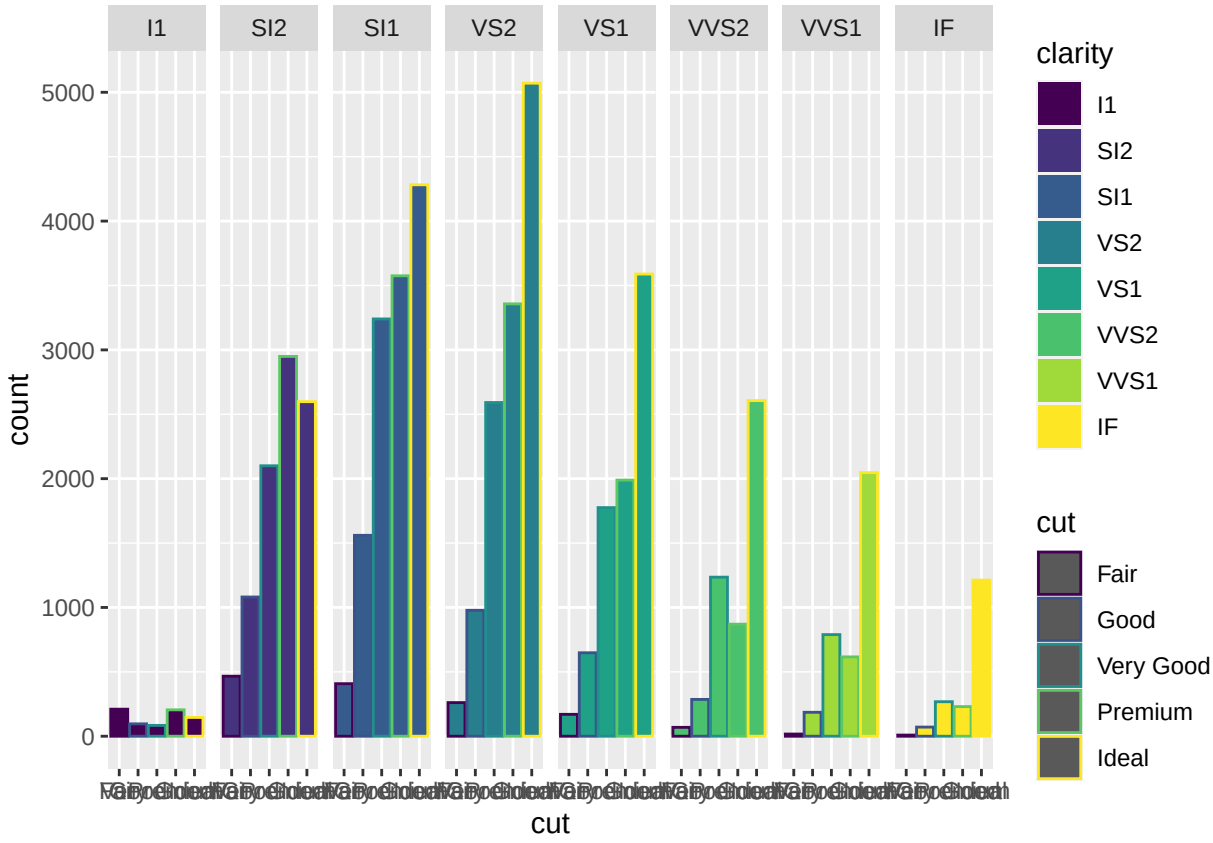
```
ggplot(diamonds, mapping = aes(x = cut, fill = clarity )) +
  geom_bar(position = "identity") +
  geom_bar(aes(colour=clarity), alpha = .5)
```



Quizá el anterior gráfico a simple vista no nos ayude mucho a entender lo que hace la estética `position="identity"`, eso es debido a que es un gráfico de barras, pero aquí se muestra con una claridad superior[1] que esta estética se encarga de no apilar, sino simplemente colocar la medida exacta de los datos, notemos que la acumulación en ideal es bastante mayor a los 20000 en el gráfico apilado, mientras en el de identity, apenas llega sobre los 5000.

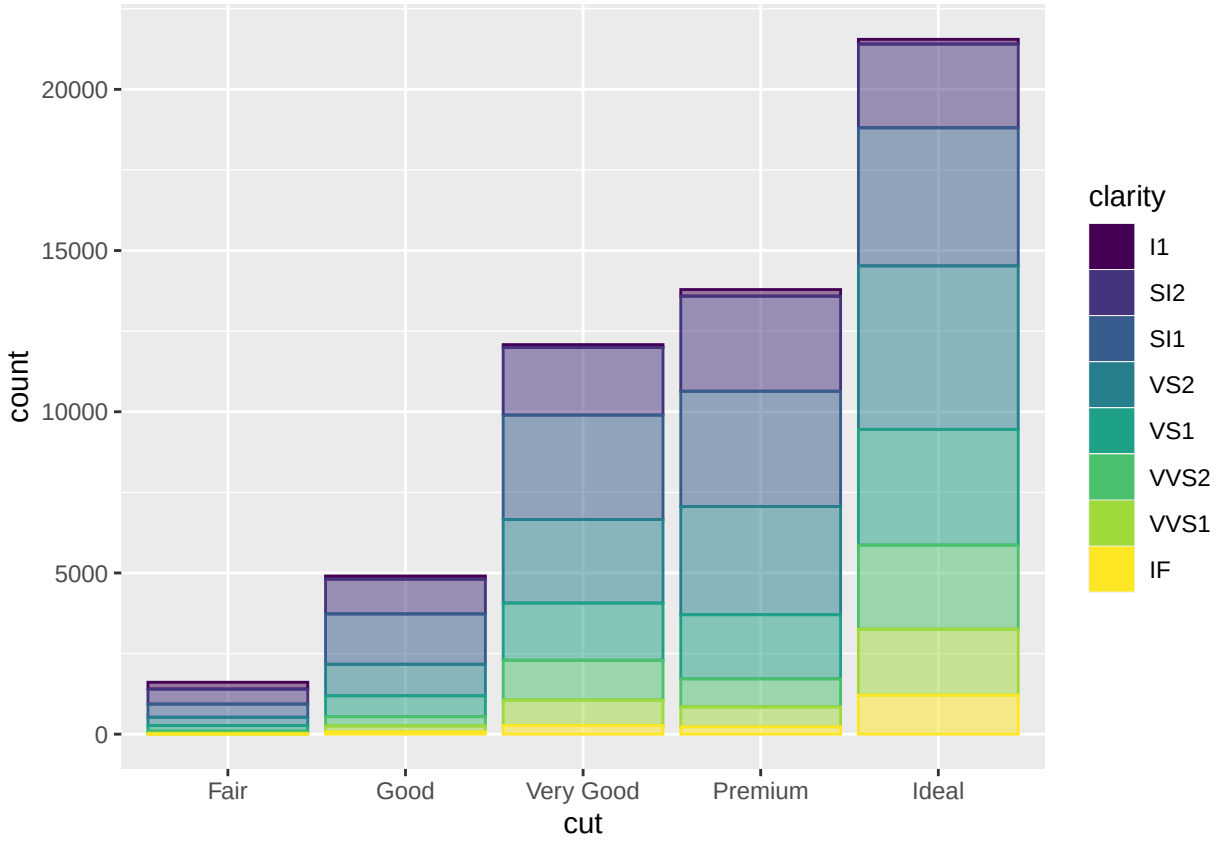
```
ggplot(diamonds, mapping = aes(x = cut, fill = clarity, colour = cut, xlab = F )) +
  geom_bar() +
  facet_grid(~clarity)
```

[1] El problema de nombres en los ejes será tratado mas adelante.




```
ggplot(diamonds, mapping = aes(x = cut, fill = clarity )) +
  geom_bar(position = "fill") +
  geom_bar(aes(colour=clarity), alpha = .5)
```

```
position = "fill"
```



para resolverlos necesitas hacer en este caso una pequeña factorización. o sea que metes todo lo que se vea igual dentro de un parentesis y usas la propiedad de distribución de los números reales. Y en estos casos particulares, tienes que pensar cuando un numero da cero. por ejemplo, si multiplicas cualquier número por 0, te va a dar 0, entonces tienes que volver en cero los números en la factorización.

$$x^2 - 9x = 0$$

sabemos que x^2 es igual a $x \cdot x$ y asi la ecuación se puede ver como esto $x \cdot x - 9x = 0$ y viendo que podemos factorizar x

$$x(x - 9) = 0, \text{ estos son justo los numeros que te tienen que dar } 0$$