TECNOLÓGICO NACIONAL DE MÉXICO

INSTITUTO TECNOLÓGICO DE TIJUANA

SUBDIRECCIÓN ACADÉMICA

DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN SEMESTRE SEPTIEMBRE-ENERO 2021

INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIONES

MINERÍA DE DATOS

LÓPEZ GONZÁLEZ ÁNGEL URIEL C16210548



El enfoque de gramática de gráficos en capas se implementa en ggplot2, una biblioteca de gráficos ampliamente utilizada para R. Todos los gráficos de esta biblioteca se crean utilizando un enfoque en capas, creando capas para crear el gráfico final.

Componentes de la gramática en capas de gráficos

Capa

● Datos

● Cartografía

● Transformación estadística (stat)

● Objeto geométrico (geom)

● Ajuste de posición (posición)

Escala

● Sistema de coordenadas (coord)

● Facetas (facetas)

● Defaults

Data

Mapping

Capa

Las capas se utilizan para crear los objetos en un gráfico. Están definidos por cinco partes básicas:

1. Datos

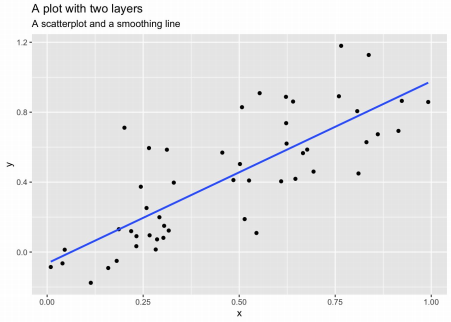
2. Cartografía

3. Transformación estadística (stat)

4. Objeto geométrico (geom)

5. Ajuste de posición (posición)



**Las capas​** suelen estar relacionadas entre sí y comparten muchas características comunes. Por ejemplo, se pueden crear varias capas utilizando los mismos datos subyacentes. Un ejemplo sería un diagrama de dispersión superpuesto con una línea de regresión suavizada para resumir la relación entre dos variables:



Datos y mapeo

Los datos definen la fuente de la información a visualizar, pero son

independientes de los demás elementos. Por lo tanto, se puede construir un gráfico en capas que utiliza diferentes fuentes de datos mientras mantiene los otros componentes iguales.

Para nuestro ejemplo de ejecución, usemos el conjunto de datos mpg en el paquete ggplot2.

**manufacturer model displ year cyl trans drv cty hwy fl class** audi a4 1.8 1999 4 auto(l5) f 18 29 p compact audi a4 1.8 1999 4 manual(m5) f 21 29 p compact audi a4 2.0 2008 4 manual(m6) f 20 31 p compact audi a4 2.0 2008 4 auto(av) f 21 30 p compact audi a4 2.8 1999 6 auto(l5) f 16 26 p compact



**El mapeo** define cómo se aplican las variables a la gráfica. Entonces, si estuviéramos graficando información de mpg, podríamos mapear el desplazamiento del motor de un automóvil a la posición y el kilometraje de la carretera a la posición y.

mpg %>%

select(displ, hwy) %>%

rename(x = displ,

y = hwy)

## # A tibble: 234 x 2

## x y

## <dbl> <int>

## 1 1.8 29

## 2 1.8 29

## 3 2 31

## 4 2 30

## 5 2.8 26

## 6 2.8 26

## 7 3.1 27

## 8 1.8 26

## 9 1.8 25

## 10 2 28

## # ... with 224 more rows

Transformación estadística

Una transformación estadística (stat) transforma los datos, generalmente resumiendo la información. Por ejemplo, en un gráfico de barras, por lo general, no intenta graficar los datos sin procesar porque esto no tiene ningún sentido inherente. En su lugar, puede resumir los datos graficando el número total de observaciones dentro de un conjunto de categorías. O si tiene un conjunto de datos con muchas observaciones, puede transformar los datos en una línea de suavizado que resume el patrón general de la relación entre las variables calculando la media de condicional.



Una estadística es una función que toma un conjunto de datos como entrada y devuelve un conjunto de datos como salida; una estadística puede agregar nuevas variables al conjunto de datos original o crear un conjunto de datos completamente nuevo. Entonces, en lugar de graficar estos datos en su forma original:

mpg %>%

select(cyl)

## # A tibble: 234 x 1

## cyl

## <int>

## 1 4

## 2 4

## 3 4

## 4 4

## 5 6

## 6 6

## 7 6

## 8 4

## 9 4

## 10 4

## # ... with 224 more rows

Lo que lo transformaría a:

mpg %>%

count(cyl)

## # A tibble: 4 x 2

## cyl n

## <int> <int>

## 1 4 81

## 2 5 4

## 3 6 79

## 4 8 70



Objetos geométricos

Los objetos geométricos (geoms) controlan el tipo de trama que crea. Las geomas se clasifican por su dimensionalidad:

● 0 dimensiones - punto, texto

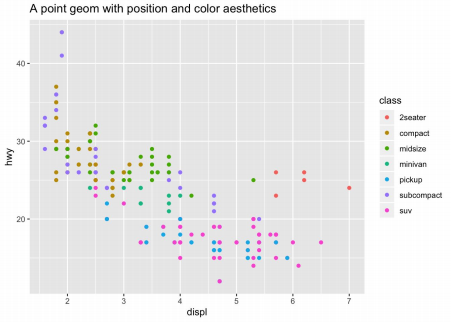
● 1 dimensión - ruta, línea

● 2 dimensiones: polígono, intervalo

Cada geom solo puede mostrar ciertos atributos estéticos o visuales de la geom. Por ejemplo, un punto geom tiene estética de posición, color, forma y tamaño.

ggplot(**data** = mpg, mapping = aes(**x** = displ, y = hwy, color = class)) + geom\_point() +

ggtitle("A point geom with position and color aesthetics")





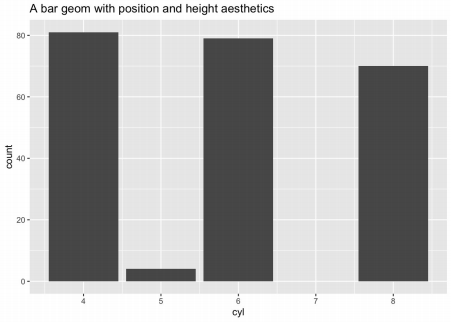
● La posición define dónde se dibuja cada punto en la gráfica.

● El color define el color de cada punto. Aquí el color está determinado por la clase del automóvil (observación)

Mientras que una barra geom tiene posición, altura, ancho y color de relleno.

ggplot(**data** = mpg, aes(**x** = cyl)) +

geom\_bar() +

ggtitle("A bar geom with position and height aesthetics")

● La posición determina la ubicación inicial (origen) de cada barra ● La altura determina la altura de la barra. Aquí, la altura se basa en el número de observaciones en el conjunto de datos para cada número posible de cilindros.



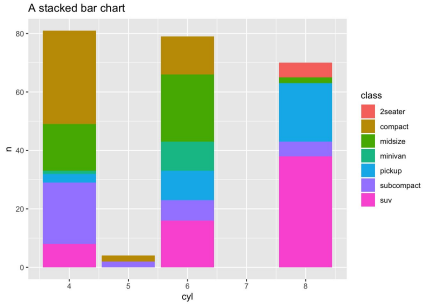
Ajuste de posición

A veces, con datos densos, necesitamos ajustar la posición de los elementos en el gráfico, de lo contrario, los puntos de datos podrían oscurecerse entre sí. Los diagramas de barras con frecuencia se apilan o esquivan las barras para evitar la superposición:

count(x = mpg, class, cyl) %>%

ggplot(mapping = aes(x = cyl, y = n, fill = class)) + geom\_bar(stat = "identity") +

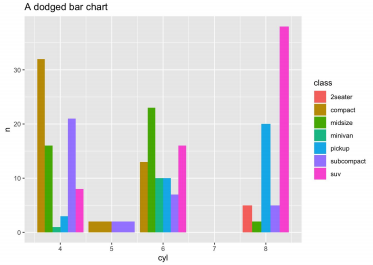
ggtitle(label = "A stacked bar chart")





count(x = mpg, class, cyl) %>%

ggplot(mapping = aes(x = cyl, y = n, fill = class)) + geom\_bar(stat = "identity", position = "dodge") + ggtitle(label = "A dodged bar chart")

Escala

Una escala controla cómo se asignan los datos a los atributos estéticos, por lo que necesitamos una escala para cada propiedad estética empleada en una capa. Por ejemplo, este gráfico define una escala de color:

ggplot(**data** = mpg, mapping = aes(**x** = displ, y = hwy, color = class)) + geom\_point() +

guides(**color** = guide\_legend(**override**.aes = list(**size** = 4)))



Tenga en cuenta que la escala es consistente: cada punto para un automóvil compacto se dibuja en color canela, mientras que los SUV se dibujan en rosa. La escala se puede cambiar para usar una paleta de colores diferente:

ggplot(**data** = mpg, mapping = aes(**x** = displ, y = hwy, color = class)) +

geom\_point() +

guides(**color** = guide\_legend(**override**.aes = list(**size** = 4))) + scale\_color\_brewer(**palette** = "Dark2")

Ahora estamos usando una paleta diferente, pero la escala sigue siendo consistente: todos los autos compactos utilizan el mismo color, mientras que los SUV usan un nuevo color, pero cada SUV todavía usa el mismo color uniforme.