Columbia University
in the City of New York

DSFBA: Visualization

Data Science for Business Analytics

Thibault Vatter

Department of Statistics, Columbia University

11/17/2021

Outline

1 Labels and annotations

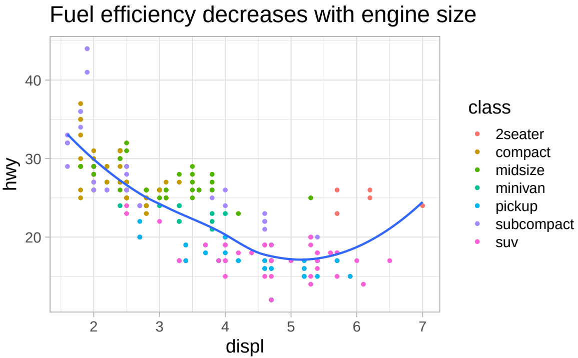
2 Guides and scales

3 Colors, zooming and themes

4 Bad graphs?

Label

```
ggplot(mpg, aes(displ, hwy)) + geom_point(aes(color = class)) +
  geom_smooth(se = FALSE) +
  labs(title = "Fuel efficiency decreases with engine size")
```

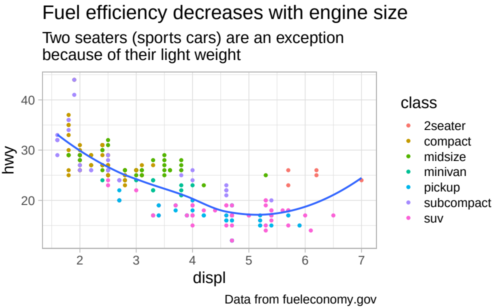


- Avoid titles that just describe what the plot is!

More text

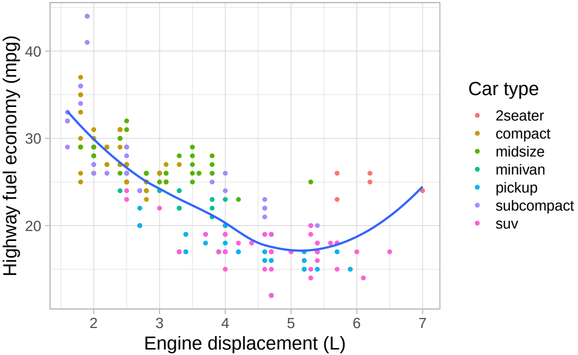
- subtitle: additional details beneath the title.
- caption: text at the bottom right of the plot.

```
ggplot(mpg, aes(displ, hwy)) +
  geom_point(aes(color = class)) + geom_smooth(se = FALSE) +
  labs(title = "Fuel efficiency decreases with engine size",
       subtitle = str_wrap("Two seaters (sports cars) are an exception
                            because of their light weight", width = 45),
       caption = "Data from fueleconomy.gov")
```



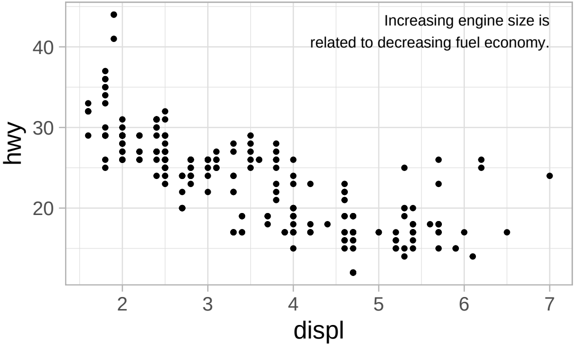
Axes

```r
ggplot(mpg, aes(displ, hwy)) +
  geom_point(aes(color = class)) +
  geom_smooth(se = FALSE) +
  labs(x = "Engine displacement (L)",
       y = "Highway fuel economy (mpg)",
       color = "Car type")
```



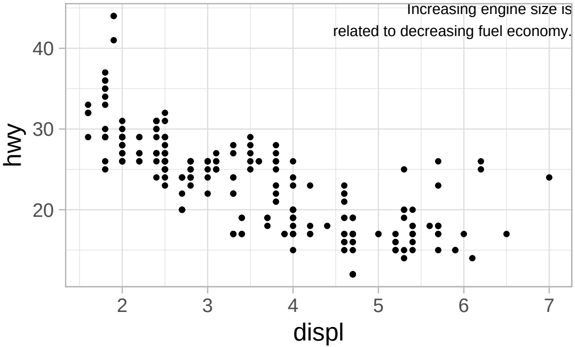
To add a single label to the plot

```
my_label <- mpg %>%
  summarize(displ = max(displ), hwy = max(hwy),
            txt = "Increasing engine size is
            related to decreasing fuel economy.")

ggplot(mpg, aes(displ, hwy)) + geom_point() +
  geom_text(aes(label = txt), data = my_label,
            vjust = "top", hjust = "right")
```



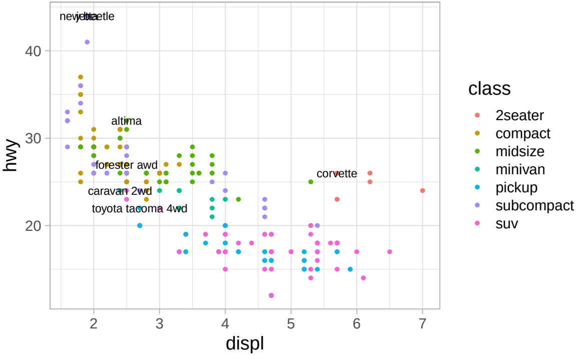
To add a single label to the plot II

```
my_label <- tibble(displ = Inf, hwy = Inf,
                   txt = "Increasing engine size is
                   related to decreasing fuel economy.")

ggplot(mpg, aes(displ, hwy)) + geom_point() +
  geom_text(aes(label = txt), data = my_label,
            vjust = "top", hjust = "right")
```



To add a multiple labels to the plot

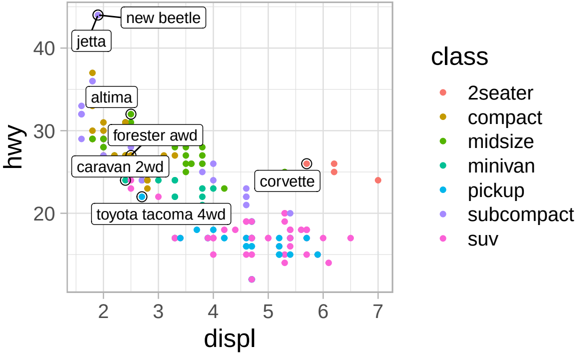
```
best_in_class <- mpg %>%
  group_by(class) %>%
  filter(row_number(desc(hwy)) == 1)

ggplot(mpg, aes(displ, hwy)) +
  geom_point(aes(color = class)) +
  geom_text(aes(label = model), data = best_in_class)
```



Or better

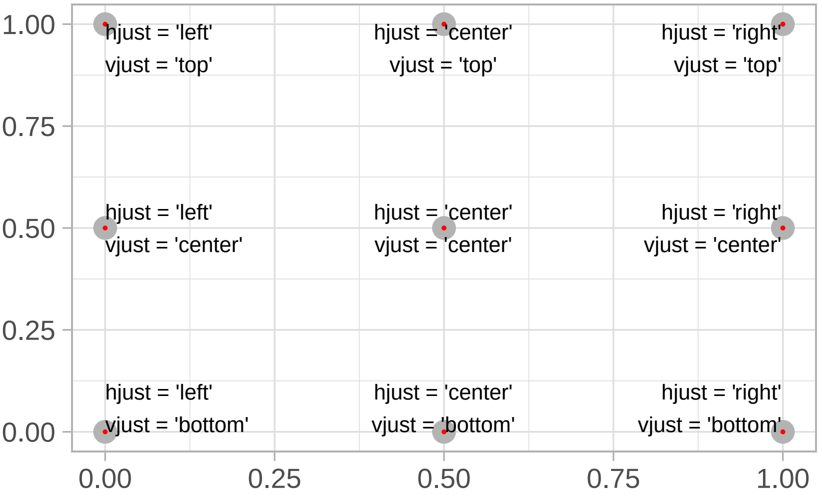
- Use the **ggrepel** package!

```
ggplot(mpg, aes(displ, hwy)) +
  geom_point(aes(color = class)) +
  geom_point(size = 3, shape = 1, data = best_in_class) +
  ggrepel::geom_label_repel(aes(label = model), data = best_in_class)
```



To control the alignment of the label

Columbia University in the City of New York



Geoms to help annotate your plot

- `geom_hline()` and `geom_vline()`:
  - Add reference lines.
  - Using e.g. `size = 2` is often a good idea.
- `geom_rect()`:
  - Draw a rectangle around points of interest.
  - Boundaries defined by `xmin`, `xmax`, `ymin`, `ymax`.
- `geom_segment()` with the `arrow` argument:
  - Draw attention to a point with an arrow.
  - `x`/`xend` and `y`/`yend` define the start/end locations.
- The only limit is your imagination (and patience)!

Outline

1 Labels and annotations

2 Guides and scales

3 Colors, zooming and themes

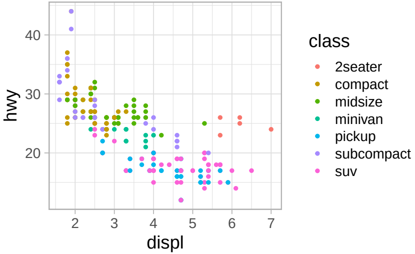
4 Bad graphs?

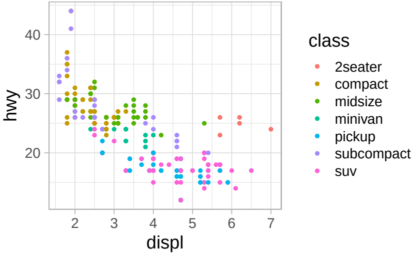
Guides and scales

- Collectively axes and legends are called **guides**:
  - Axes are used for x and y aesthetics.
  - Legends are used for everything else.
- **Scales** control mappings from data values to perceived values:

```
ggplot(mpg, aes(displ, hwy)) +
  geom_point(aes(color = class)) # +
  # scale_x_continuous() +
  # scale_y_continuous() +
  # scale_color_discrete()
```

```
ggplot(mpg, aes(displ, hwy)) +
  geom_point(aes(color = class)) +
  scale_x_continuous() +
  scale_y_continuous() +
  scale_color_discrete()
```

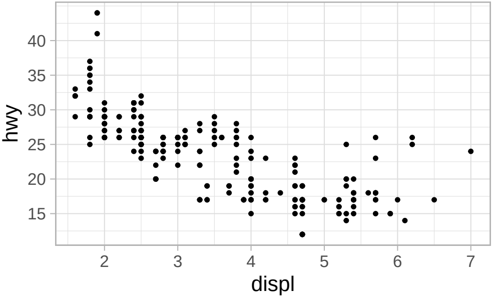




Axes ticks and legend keys

- To control the ticks on the axes and the keys on the legend:
  - `breaks`: ticks positions, or values associated with keys.
  - `labels`: text associated with each tick/key.
- The scales package gives you tools to override the defaults!

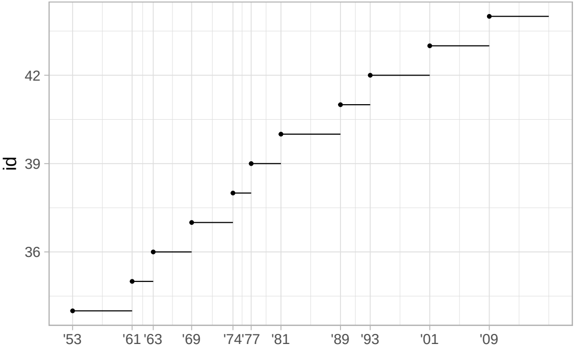
```
ggplot(mpg, aes(displ, hwy)) + geom_point() +
  scale_y_continuous(breaks = seq(15, 40, by = 5))
```



Breaks and labels for date/datetime

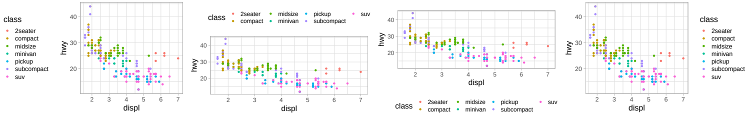
- date_labels: a format as in ?readr::parse_datetime().
- date_breaks: a string like "2 days" or "1 month".

```
presidential %>% mutate(id = 33 + row_number()) %>%
  ggplot(aes(start, id)) + geom_point() +
    geom_segment(aes(xend = end, yend = id)) +
    scale_x_date(NULL, breaks = presidential$start, date_labels = "'%y")
```



Legend layout

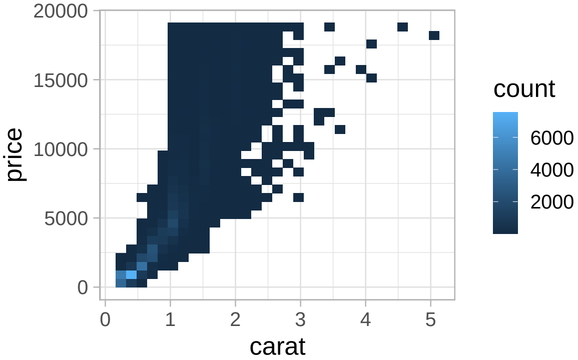
```
base <- ggplot(mpg, aes(displ, hwy)) + geom_point(aes(color = class))

base + theme(legend.position = "left")
base + theme(legend.position = "top")
base + theme(legend.position = "bottom")
base + theme(legend.position = "right") # the default
```



- legend.position = "none" suppresses the display!

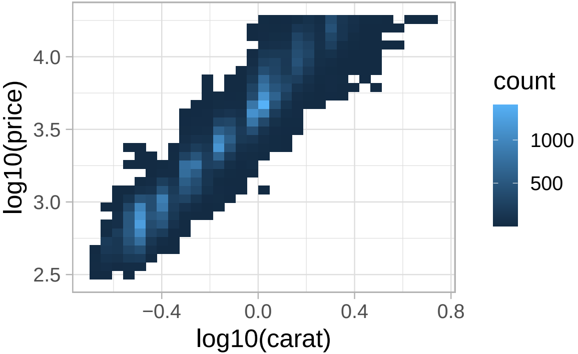
How could we improve the scale?

```
ggplot(diamonds, aes(carat, price)) +
  geom_bin2d()
```



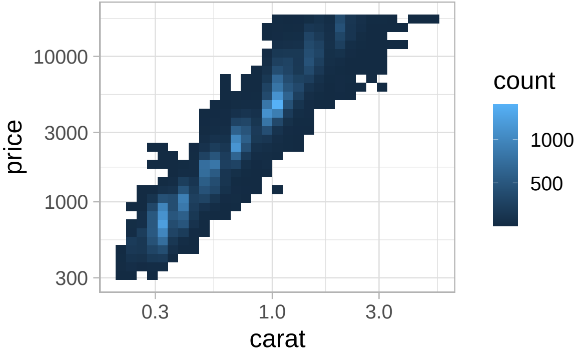
Log-transform the variables

```
ggplot(diamonds, aes(log10(carat), log10(price))) +
  geom_bin2d()
```



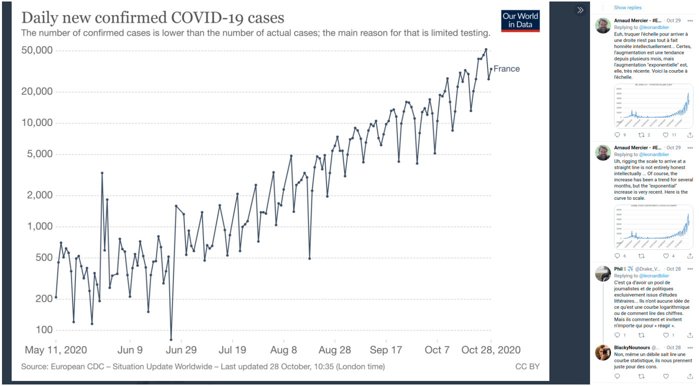
... or simply replace the scale

```
ggplot(diamonds, aes(carat, price)) +
  geom_bin2d() +
  scale_x_log10() +
  scale_y_log10()
```



Not everyone gets it :)

Tweet from Léonard Blier: “Many journalists and political leaders in France explain that the pandemic is growing faster than expected. Here are the daily new cases in France since May 11th (end of the lockdown), log scale. How could it be more predictable? Where is the surprise?”



Daily new confirmed COVID-19 cases

The number of confirmed cases is lower than the number of actual cases; the main reason for that is limited testing.

Source: European CDC – Situation Update Worldwide – Last updated 28 October, 10:35 (London time)

CC BY

Arnaud Mercier - #E... · Oct 29
Replying to @leonardblier
Euh, truquer l'échelle pour arriver à une droite n'est pas tout à fait honnête intellectuellement... Certes, l'augmentation est une tendance depuis plusieurs mois, mais l'augmentation "exponentielle" est, elle, très récente. Voici la courbe à l'échelle.

Arnaud Mercier - #E... · Oct 29
Replying to @leonardblier
Uh, rigging the scale to arrive at a straight line is not entirely honest intellectually ... Of course, the increase has been a trend for several months, but the "exponential" increase is very recent. Here is the curve to scale.

Phil ✈️ @Drake_V... · Oct 28
Replying to @leonardblier
C’est ça d’avoir un pool de journalistes et de politiques exclusivement issus d’études littéraires... Ils n’ont aucune idée de ce qu’est une courbe logarithmique ou de comment lire des chiffres. Mais ils commentent et invitent n’importe qui pour « réagir ».

BlackyNounours @... · Oct 28
Non, même un débile sait lire une courbe statistique, ils nous prennent juste pour des cons.

Outline

1 Labels and annotations

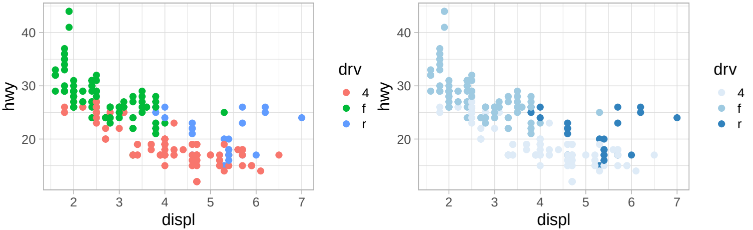
2 Guides and scales

3 Colors, zooming and themes

4 Bad graphs?

Replacing color scales

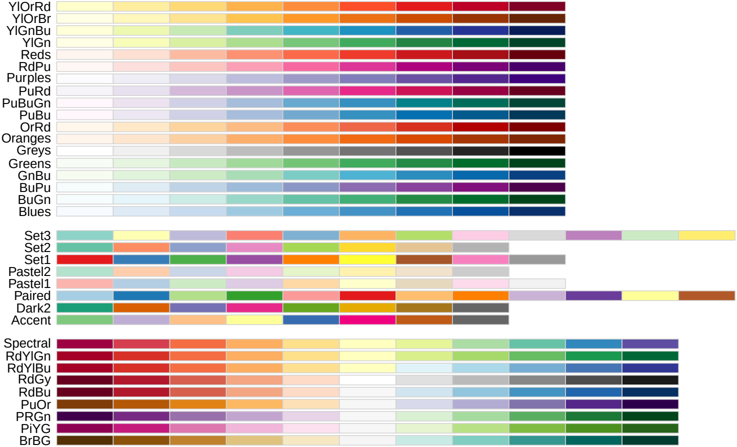
```
ggplot(mpg, aes(displ, hwy)) +
  geom_point(aes(color = drv), size = 3)

ggplot(mpg, aes(displ, hwy)) +
  geom_point(aes(color = drv), size = 3) +
  scale_color_brewer(palette = "Blues")
```



- Color scales come in two variety:
  - `scale_color_x()` for the `color` aesthetics (available in UK/US spellings).
  - `scale_fill_x()` for the `fill` aesthetics.

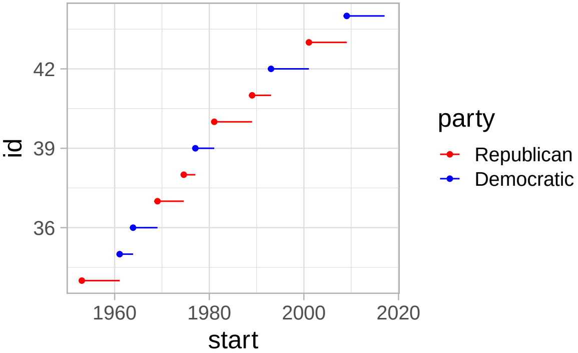
The ColorBrewer scales

- Documented online at http://colorbrewer2.org/
- Available via the **RColorBrewer** package.



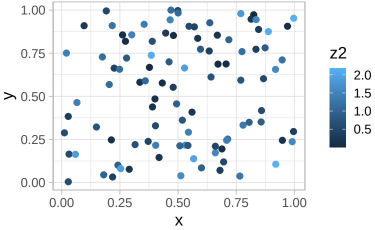
Using manually defined mappings

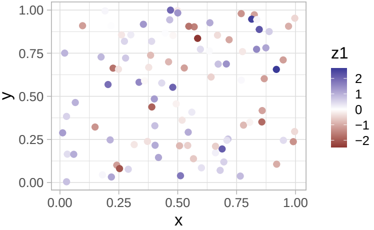
```
presidential %>%
  mutate(id = 33 + row_number()) %>%
  ggplot(aes(start, id, color = party)) +
    geom_point() +
    geom_segment(aes(xend = end, yend = id)) +
    scale_color_manual(values = c(Republican = "red",
                                  Democratic = "blue"))
```



Continuous vs diverging color scales

```r
df <- data.frame(x = runif(100), y = runif(100),
                 z1 = rnorm(100), z2 = abs(rnorm(100)))

ggplot(df, aes(x, y)) +
  geom_point(aes(color = z2), size = 3)

ggplot(df, aes(x, y)) +
  geom_point(aes(color = z1), size = 3) +
  scale_color_gradient2()
```

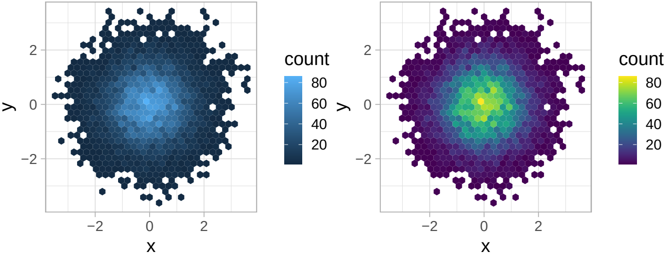




A continuous analog of ColorBrewer

- The viridis package!

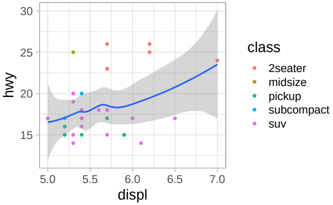
```
df <- tibble(x = rnorm(10000), y = rnorm(10000))

ggplot(df, aes(x, y)) +
  geom_hex() +
  coord_fixed()

ggplot(df, aes(x, y)) +
  geom_hex() +
  coord_fixed() +
  viridis::scale_fill_viridis()
```

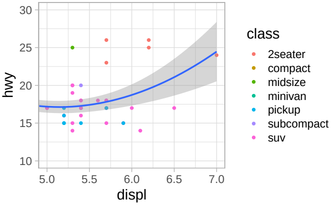


Zooming

- Three methods:
  - Adjust what data are plotted.
  - Set `xlim` and `ylim` in `coord_cartesian()`.
  - Set the limits in each scale.

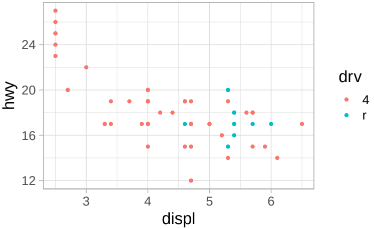
```
mpg %>%
  filter(displ >= 5, displ <= 7, hwy >= 10, hwy <= 30) %>%
  ggplot(aes(displ, hwy)) +
    geom_point(aes(color = class)) + geom_smooth()

ggplot(mpg, mapping = aes(displ, hwy)) +
  geom_point(aes(color = class)) + geom_smooth() +
  coord_cartesian(xlim = c(5, 7), ylim = c(10, 30))
```

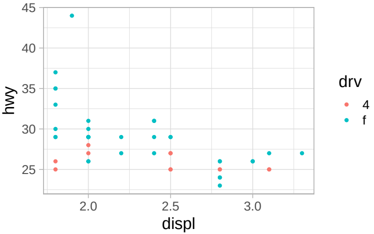




Zooming cont'd

```
suv <- mpg %>%
  filter(class == "suv")
compact <- mpg %>%
  filter(class == "compact")

ggplot(suv, aes(displ, hwy, color = drv)) +
  geom_point()
ggplot(compact, aes(displ, hwy, color = drv)) +
  geom_point()
```

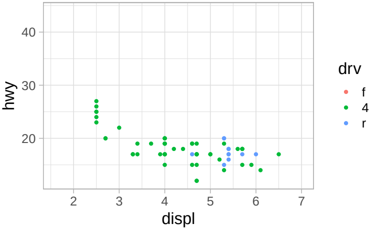


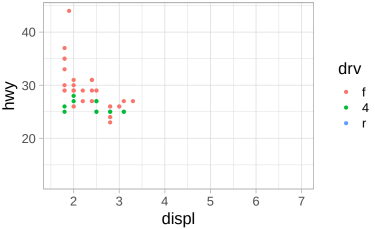


Share scales across multiple plots

- Training the scales with the `limits` of the full data:

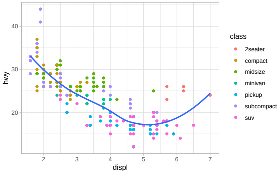
```
x_scale <- scale_x_continuous(limits = range(mpg$displ))
y_scale <- scale_y_continuous(limits = range(mpg$hwy))
col_scale <- scale_color_discrete(limits = unique(mpg$drv))

ggplot(suv, aes(displ, hwy, color = drv)) + geom_point() +
  x_scale + y_scale + col_scale
ggplot(compact, aes(displ, hwy, color = drv)) + geom_point() +
  x_scale + y_scale + col_scale
```



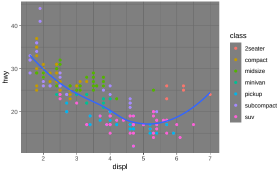


Themes

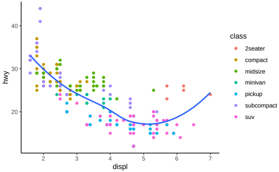
```
ggplot(mpg, aes(displ, hwy)) +
  geom_point(aes(color = class)) +
  geom_smooth(se = FALSE) +
  theme_light()
```



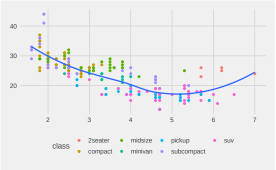
```
ggplot(mpg, aes(displ, hwy)) +
  geom_point(aes(color = class)) +
  geom_smooth(se = FALSE) +
  theme_dark()
```



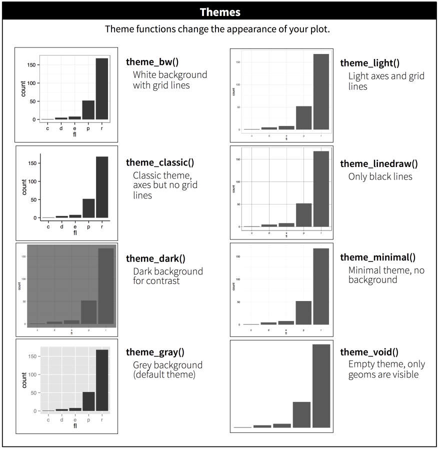
```
ggplot(mpg, aes(displ, hwy)) +
  geom_point(aes(color = class)) +
  geom_smooth(se = FALSE) +
  theme_classic()
```



```
ggplot(mpg, aes(displ, hwy)) +
  geom_point(aes(color = class)) +
  geom_smooth(se = FALSE) +
  ggthemes::theme_fivethirtyeight()
```



ggplot2 default themes



Themes

Theme functions change the appearance of your plot.

theme_bw()
White background with grid lines

theme_light()
Light axes and grid lines

theme_classic()
Classic theme, axes but no grid lines

theme_linedraw()
Only black lines

theme_dark()
Dark background for contrast

theme_minimal()
Minimal theme, no background

theme_gray()
Grey background (default theme)

theme_void()
Empty theme, only geoms are visible

- More in add-on packages like ggthemes!

Outline

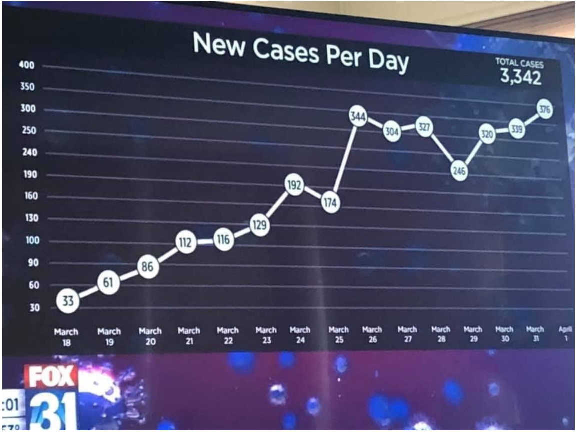
1 Labels and annotations

2 Guides and scales

3 Colors, zooming and themes

4 Bad graphs?

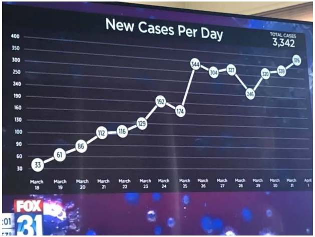
Remember?

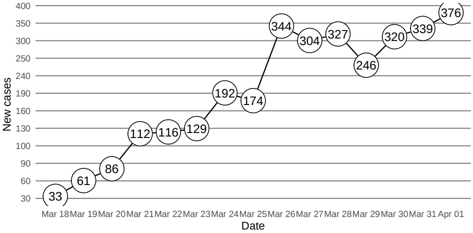

New Cases Per Day
TOTAL CASES
3,342
400
350
300
250
240
190
160
130
100
90
60
30
33
61
86
112
116
129
192
174
344
304
327
246
320
339
376
March 18
March 19
March 20
March 21
March 22
March 23
March 24
March 25
March 26
March 27
March 28
March 29
March 30
March 31
April 1
FOX 31

The code

```r
fox_data <- tibble(
  new_cases = c(33, 61, 86, 112, 116, 129, 192, 174,
                344, 304, 327, 246, 320, 339, 376),
  date = seq(as.Date("2020-03-18"), as.Date("2020-04-01"), by = 1))

trans_dumb <- function(breaks) {
  breaks <- c(0, breaks)
  trans_new(name = "dumb",
            trans = splinefun(breaks, seq_along(breaks)),
            inverse = splinefun(seq_along(breaks), breaks))
}

breaks <- c(30, 60, 90, 100, 130, 160, 190, 240, 250, 300, 350, 400)
ggplot(fox_data, aes(x = date, y = new_cases, label = new_cases)) +
  geom_line()  +
  geom_point(size = 10, colour = "white") +
  geom_point(size = 10, colour = "black", shape = 1) +
  geom_text() +
  scale_x_date(date_breaks = "1 day", date_labels = "%b %d") +
  scale_y_continuous(trans = trans_dumb(breaks), breaks = breaks) +
  labs(x = "Date", y = "New cases") +
  theme_minimal(base_size = 10) +
  theme(panel.grid = element_blank(),
        panel.grid.major.y = element_line(color = "grey50"))
```

The results


New Cases Per Day
TOTAL CASES
3,342
400
350
300
250
240
190
160
130
100
90
60
30
33
61
86
112
116
129
192
174
344
304
327
246
320
339
376
March 18
March 19
March 20
March 21
March 22
March 23
March 24
March 25
March 26
March 27
March 28
March 29
March 30
March 31
April 1
FOX 31


New cases
400
350
300
250
240
190
160
130
100
90
60
30
33
61
86
112
116
129
192
174
344
304
327
246
320
339
376
Mar 18
Mar 19
Mar 20
Mar 21
Mar 22
Mar 23
Mar 24
Mar 25
Mar 26
Mar 27
Mar 28
Mar 29
Mar 30
Mar 31
Apr 01
Date