

Themes

Session 5

PMAP 8921: Data Visualization with R
Andrew Young School of Policy Studies
Fall 2023

Plan for today

CRAP and ggplot

The anatomy of a ggplot theme

CRAP and ggplot

Universal principles

Contrast

Repetition

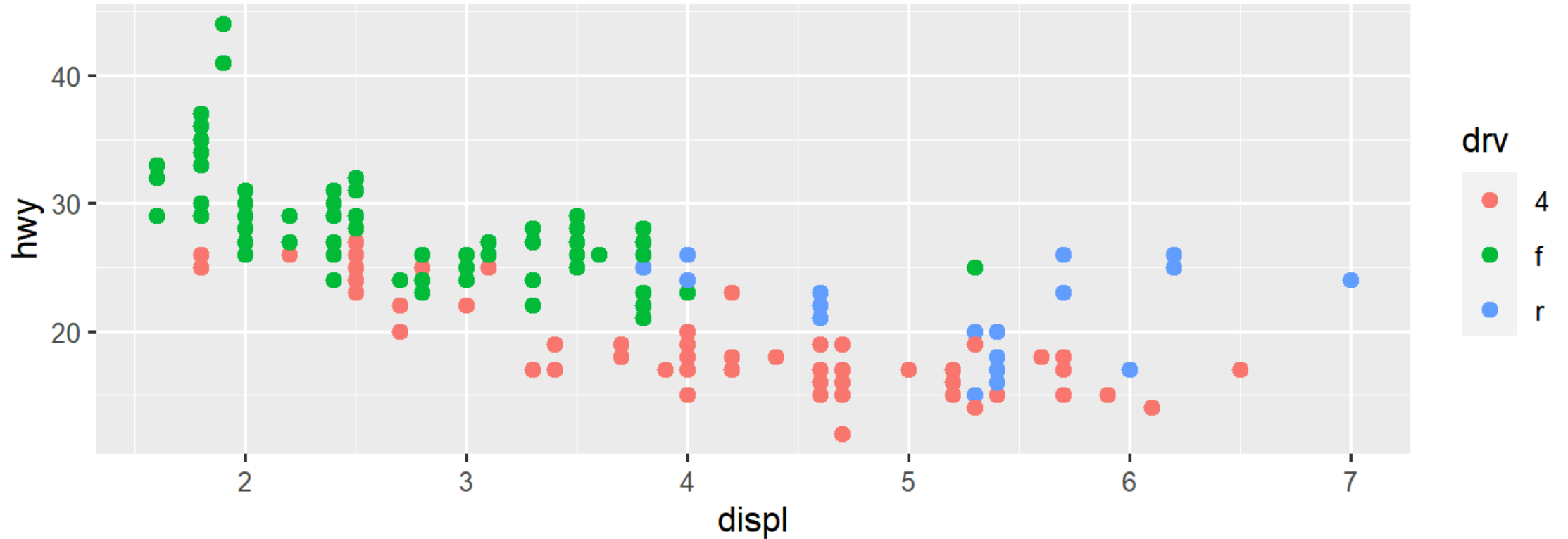
Alignment

Proximity

These design principles apply everywhere!

Graphic design, art, music, architecture... and graphs!

Is that gray background okay?



Applying CRAP to ggplot

We can follow CRAP principles to make big improvements to our plots

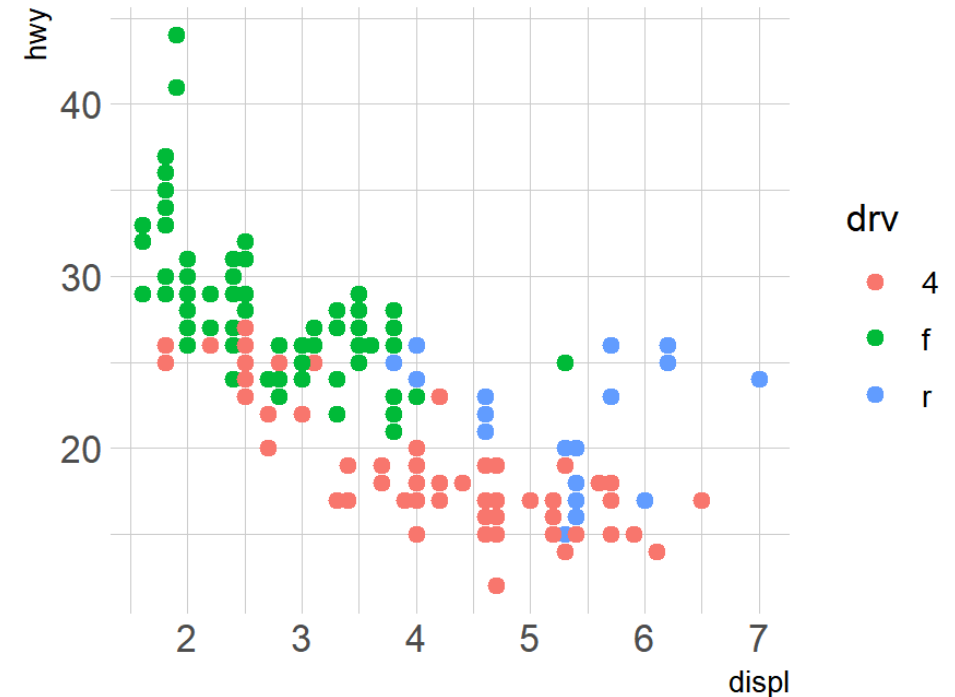
Claus Wilke's chapter covers lots of these graph-specific principles

We can apply these principles to ggplot plots

Like this!

```
library(hrbrthemes)

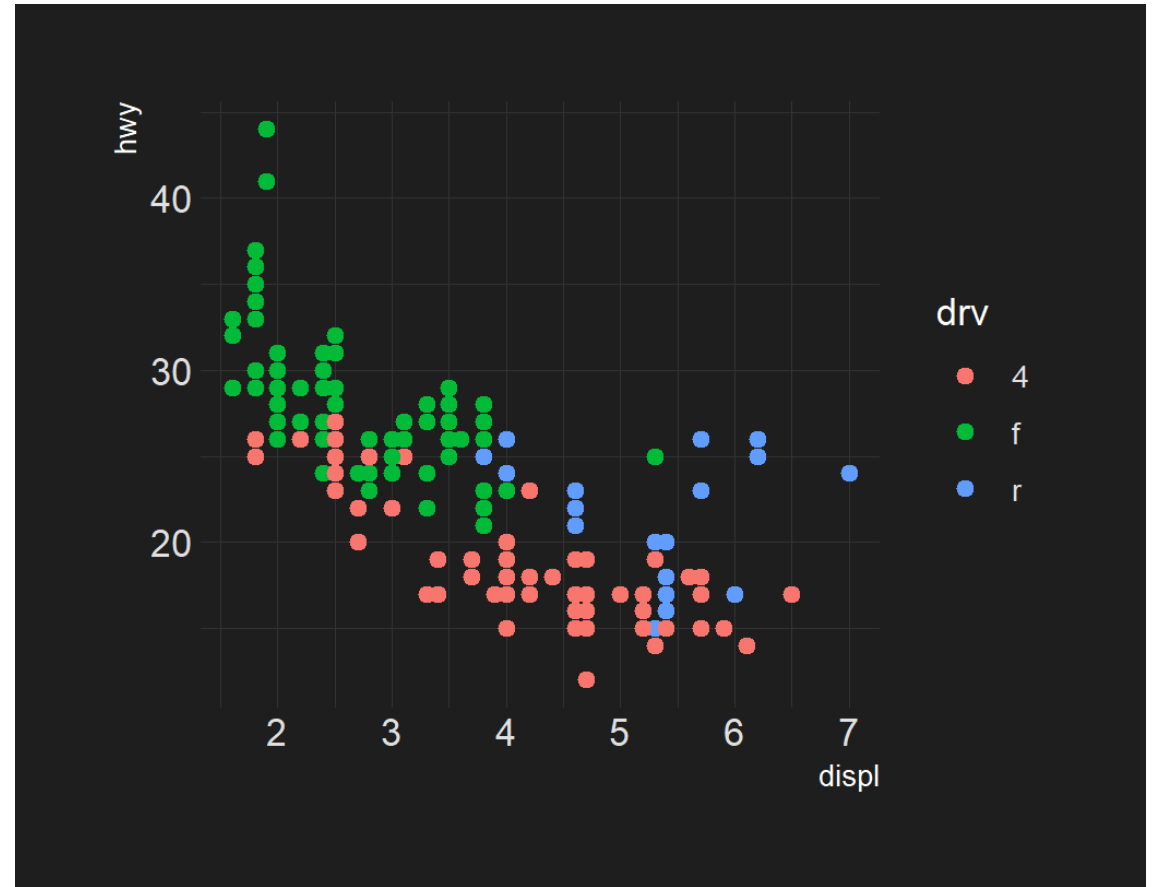
ggplot(mpg, aes(x = displ, y = hwy,
                color = drv)) +
  geom_point(size = 2) +
  theme_ipsum()
```



And this!

```
library(hrbrthemes)

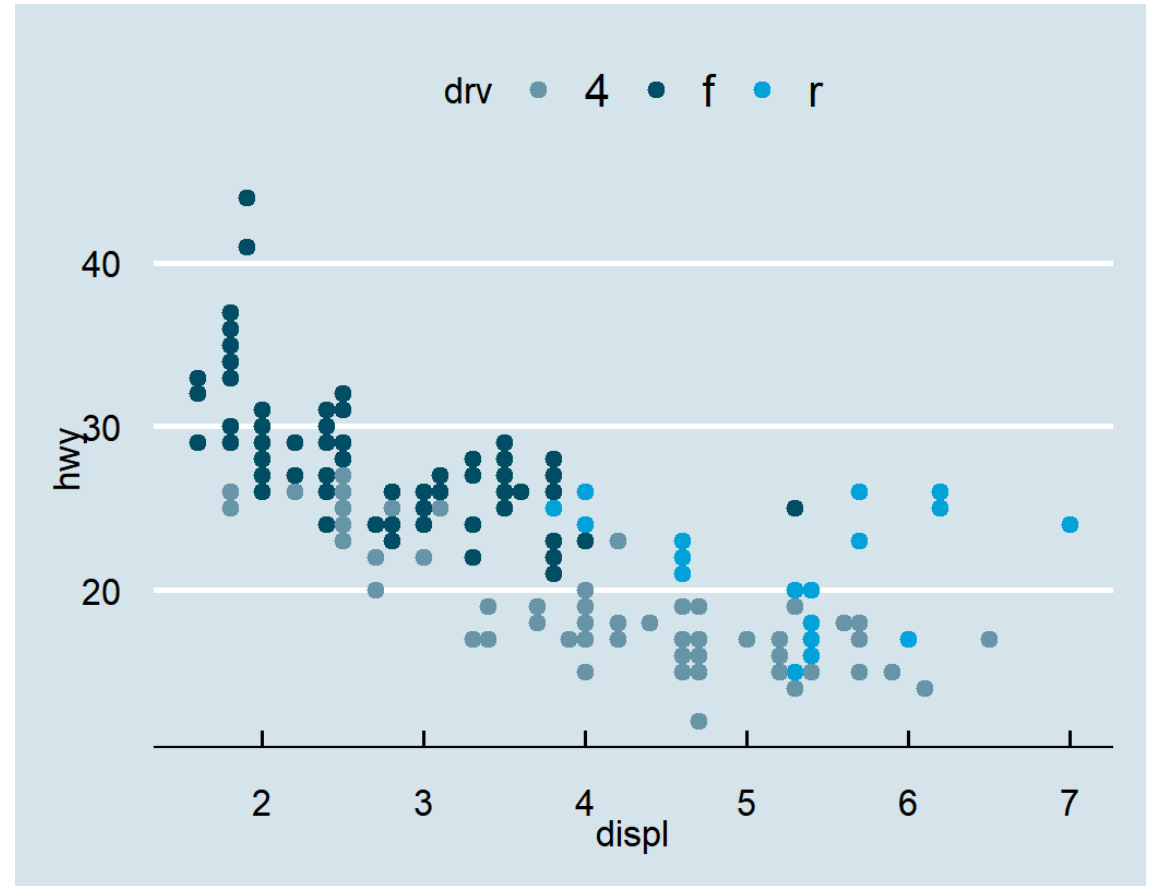
ggplot(mpg, aes(x = displ, y = hwy,
                color = drv)) +
  geom_point(size = 2) +
  theme_modern_rc()
```



Or this!

```
library(ggthemes)

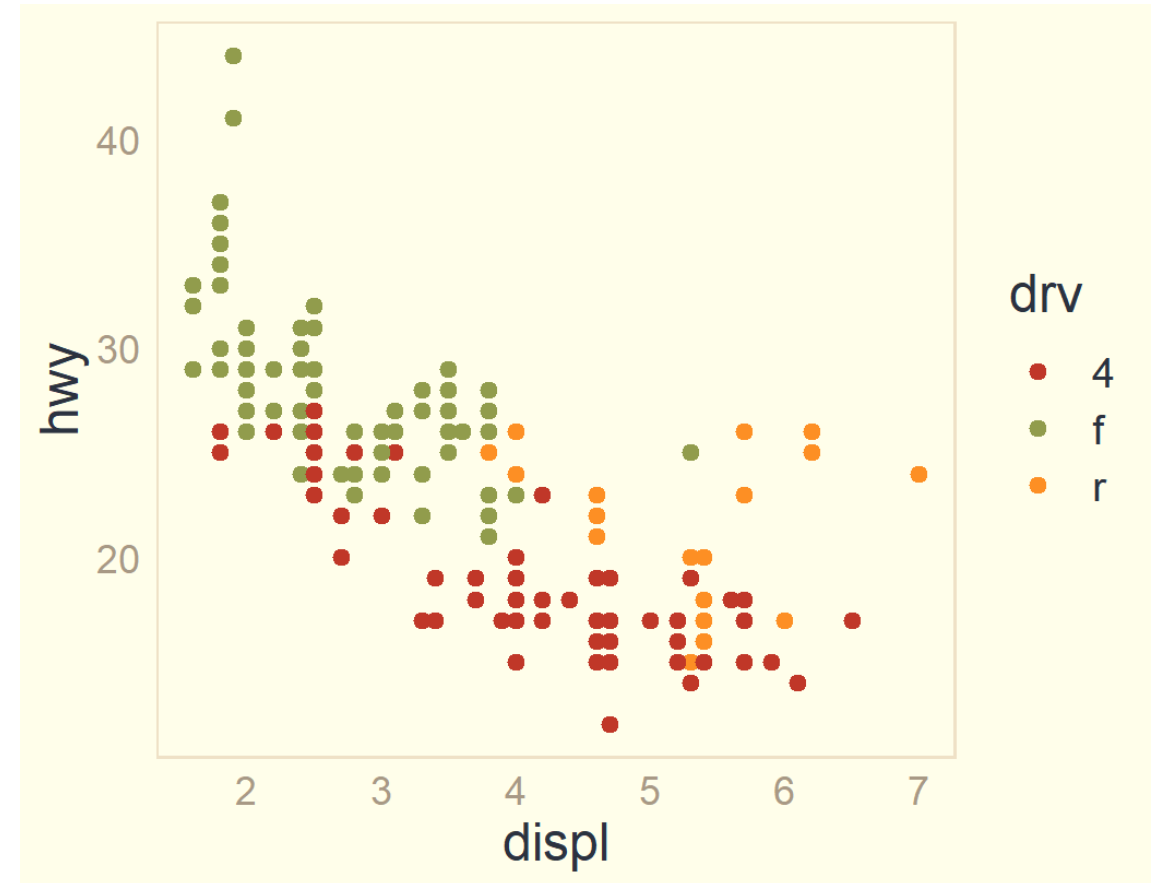
ggplot(mpg, aes(x = displ, y = hwy,
                color = drv)) +
  geom_point(size = 2) +
  scale_color_economist() +
  theme_economist()
```



And even this!

```
library(ggpomological)

ggplot(mpg, aes(x = displ, y = hwy,
                color = drv)) +
  geom_point(size = 2) +
  scale_color_pomological() +
  theme_pomological_fancy()
```



One magic, powerful function

theme()

The anatomy of a `ggplot()` theme

Theme system

ggplot2 Theme Elements

`theme(element_name = element_function())`

- `element_text()`
- `element_line()`
- `element_rect()`
- `element_blank()`

Axis elements:

`axis.ticks`
`element_line()`

`axis.title`
`element_text()`

`axis.text`
`element_text()`

`axis.line`
`element_line()`

Plot elements:

`plot.background`
`element_rect()`

`plot.title`
`element_text()`

`plot.margin`
`margin()`

Facetting elements:

`strip.background`
`element_rect()`

`panel.spacing`
`unit()`

`strip.text`
`element_text()`

Legend elements:

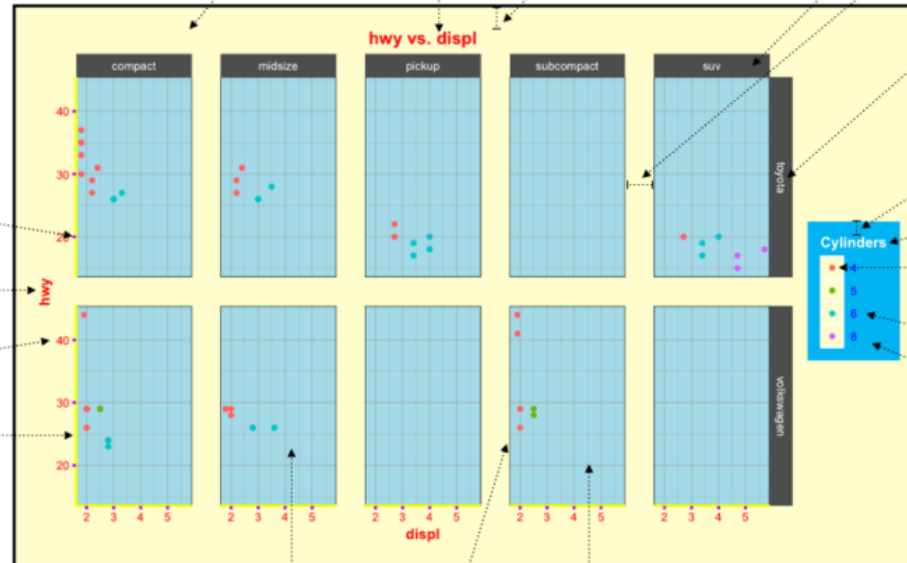
`legend.margin`
`margin()`

`legend.title`
`element_text()`

`legend.key`
`element_rect()`

`legend.text`
`element_text()`

`legend.background`
`element_rect()`



`panel.background`
`element_rect()`

`panel.grid`
`element_line()`

`panel.border`
`element_rect(fill = NA)`

Panel elements:

henrywang.nl

Derived from "ggplot2: Elegant Graphics for Data Analysis"

Theme elements

Each element in the plot can be targeted

Plot title = `plot.title`

Grid lines = `panel.grid`

Legend background = `legend.background`

Theme functions

Use special functions to
manipulate specific elements

Text-based things = `element_text()`

Rectangular things (backgrounds) = `element_rect()`

Line-based things (axis lines, grid lines) = `element_line()`

Disable element completely = `element_blank()`

How to learn `theme()`

The `theme()` function has
94 possible arguments(!!!)

You can get hyper-specific with things like
`axis.ticks.length.x.bottom`

The only way to learn how to use `theme()`
is to use it and tinker with it

How to learn theme()

I cannot show you everything

That's why we have the lesson, example, and exercise!