Grammar of Graphics (ggplot2)

Tyson Whitehead

February 10, 2026

Material

Slides and R and python plot commands available at

▶ https://staff.sharcnet.ca/tyson/GGPlot-20260210-VizWinter/

Presentation source material available at

▶ https://git.sharcnet.ca/training/ggplot

A Grammar of Graphics

What is a graphic? How can we succinctly describe a graphic? And how can we create the graphic that we have described?

   Grammar   the principles or rules of an art, science, or technique

A good grammar will

- ▶ allow us to gain insight into the composition of complicated graphics,
- ▶ reveal unexpected connections between seemingly different graphics,
- ▶ provide a strong foundation for understanding a diverse range of graphics
- ▶ guide us on what a well-formed or correct graphic looks like

Grammar of Graphics (Work)

Well thought out and studied framework with a solid academic background.

1983 *Semiology of Graphics* (Bertin)
2005 *The Grammer of Graphics* (Wilkinson, Anand, and Grossman)
2010 *A Layered Grammar of Graphics* (Wickham)

The Layered Grammar of Graphics

```
> ggplot(data = <DATA>) +
    <GEOM_FUNCTION>(
      mapping = aes(<MAPPINGS>),
      stat = <STAT>,
      position = <POSITION>
    ) + ... +
    <COORDINATE_FUNCTION> +
    <FACET_FUNCTION> +
    <SCALE_FUNCTION> + ... +
    <GUIDE_FUNCTION> + ... +
    <THEME_FUNCTION> + ...
```

The Layered Grammar of Graphics (Components)

- ▶ DATA underlying data set providing the observations
- ▶ STAT statistical transformation (stat) of the information to be displayed
- ▶ GEOM_FUNCTION geometric object (geom) to represent information
- ▶ MAPPINGS how values to be display map to the levels of an aesthetic
- ▶ COORDINATE_FUNCTION coordinate system to place the geom into
- ▶ POSITION position adjustments in the coordinate system
- ▶ FACET_FUNCTION split the plot into subplots
- ▶ SCALE_FUNCTION how data values are translated to visual properties
- ▶ GUIDE_FUNCTION help readers interpret the plot
- ▶ THEME_FUNCTION controls the display of non-date items

The Layered Grammar of Graphics (Implementations)

Implementations

R	ggplot2 (https://ggplot2.tidyverse.org)
Python	plotnine (https://plotnine.readthedocs.io)

Excellent R books (good for Python too)

- ▶ ggplot2: Elegant Graphics for Data Analysis (ggplot2-book.org)
- ▶ R for Data Science (r4ds.had.co.nz)

Creating a Plot (1/2)

Sample data is a subset of the fuel economy data the US Environmental Protection Agency (EPA) provides on cars from 1999 and 2008.

```
> mpg
# A tibble: 234 x 11
  manufacturer model displ  year   cyl trans drv
  <chr>        <chr> <dbl> <int> <int> <chr> <chr>
1 audi         a4      1.8  1999     4 auto... f
2 audi         a4      1.8  1999     4 manu... f
3 audi         a4      2    2008     4 manu... f
4 audi         a4      2    2008     4 auto... f
5 audi         a4      2.8  1999     6 auto... f
# ... with 229 more rows, and 4 more variables:
#   cty <int>, hwy <int>, fl <chr>, class <chr>
```

Creating a Plot (2/2)
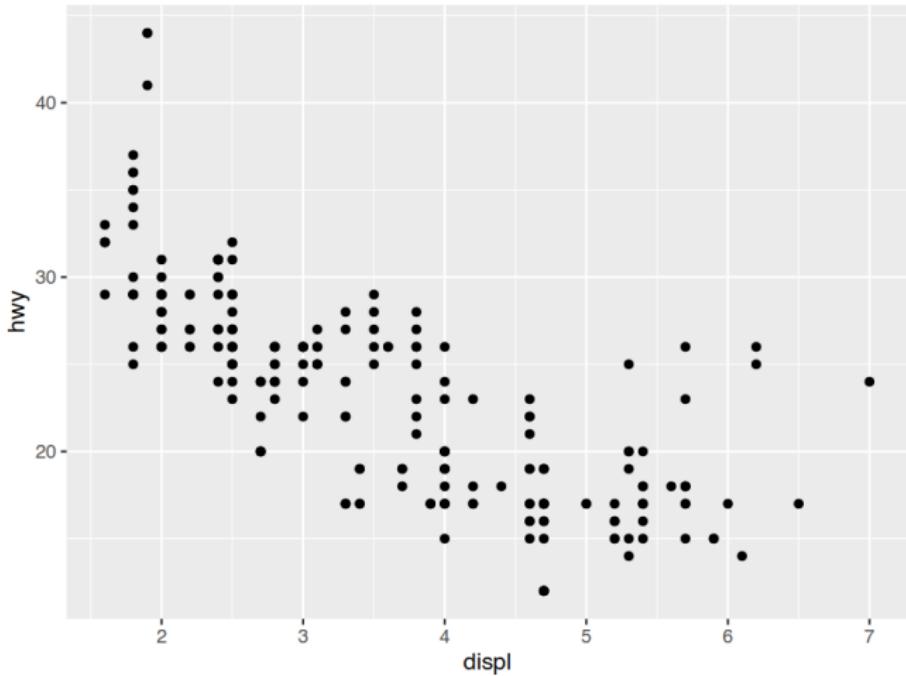
Figure 1: First ggplot showing highway mpg vs engine displacement.

Aesthetic Mappings (1/2)
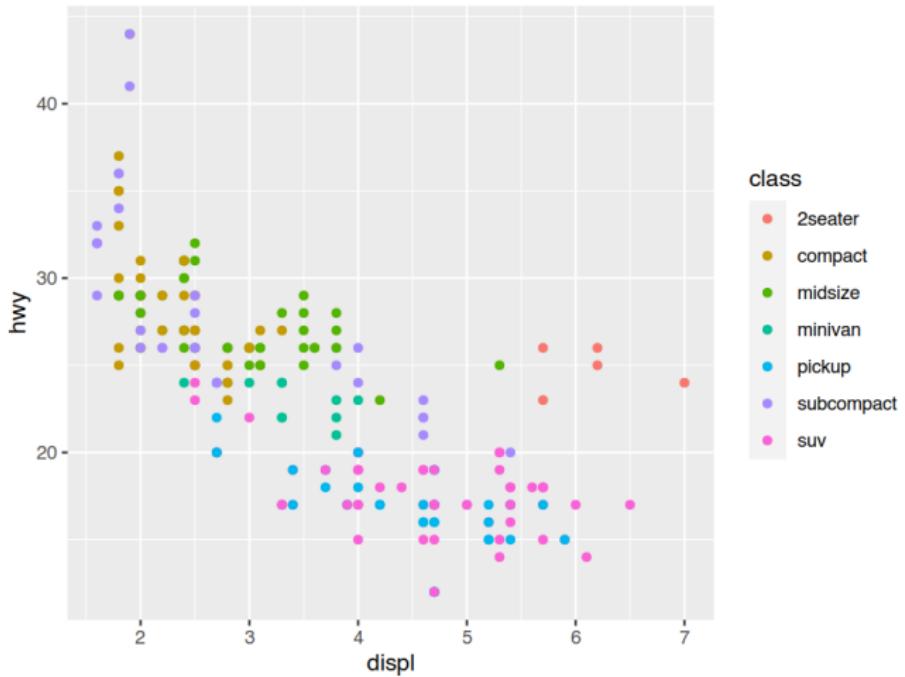
Figure 2: Using additional aesthetics to convey more information.

Aesthetic Mappings (2/2)
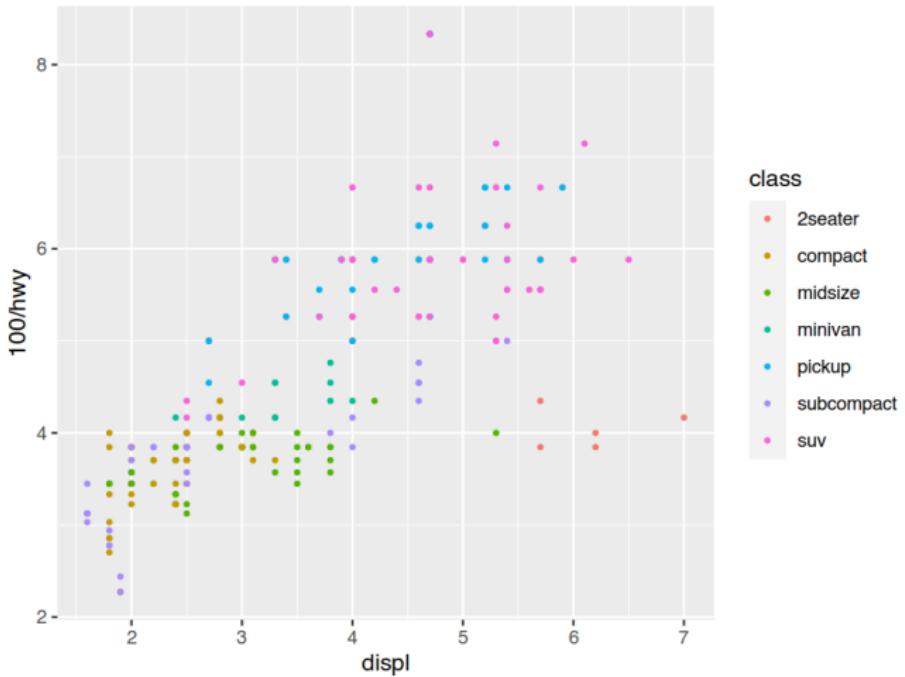
Figure 3: Formulas for aesthetics and setting defaults.

Faceting
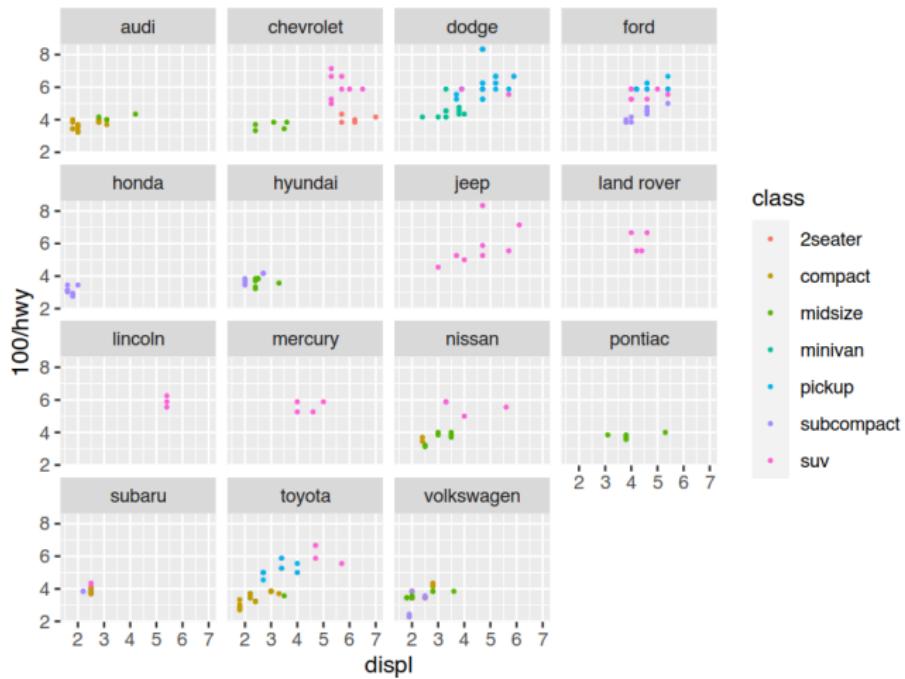
Figure 4: Facets for comparing subsets of the data.

Geometric Objects
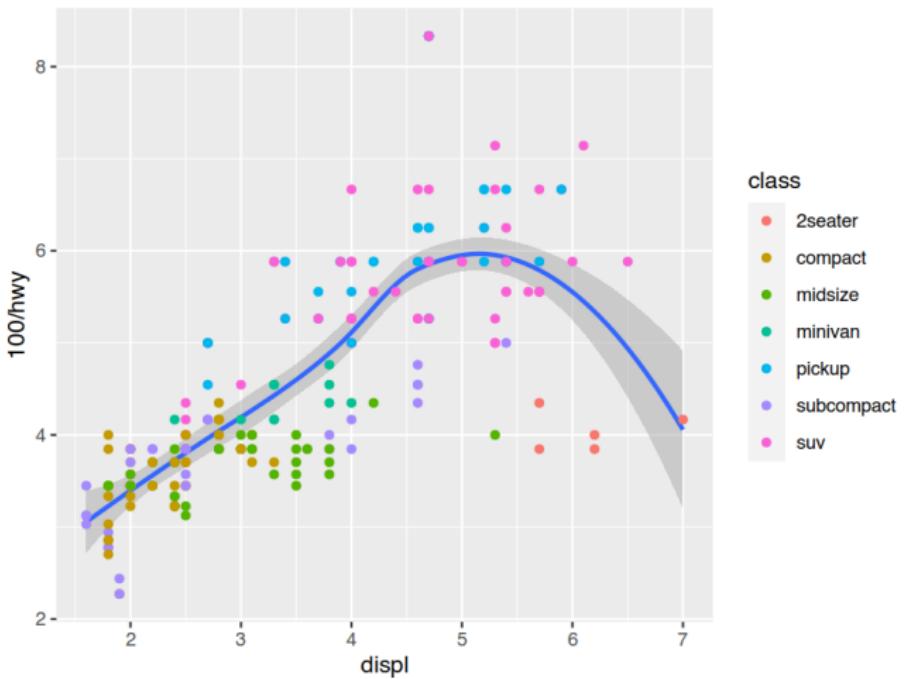
Figure 5: Adding additional geometry objects.

Statistical Transformations
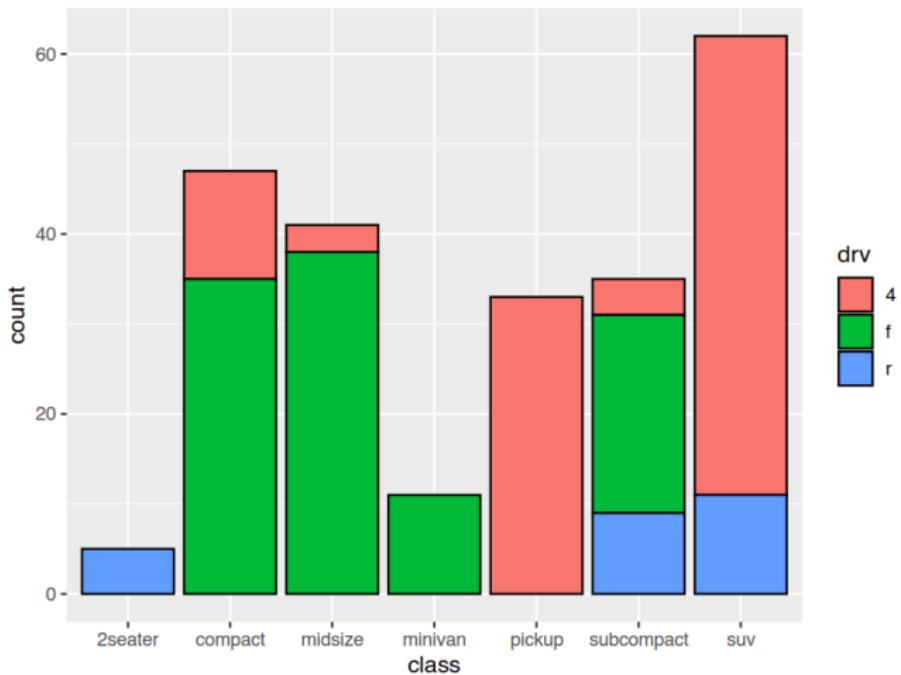
Figure 6: Number of vehicles for each class broken out by drivetrain.

Position Adjustments (1/2)
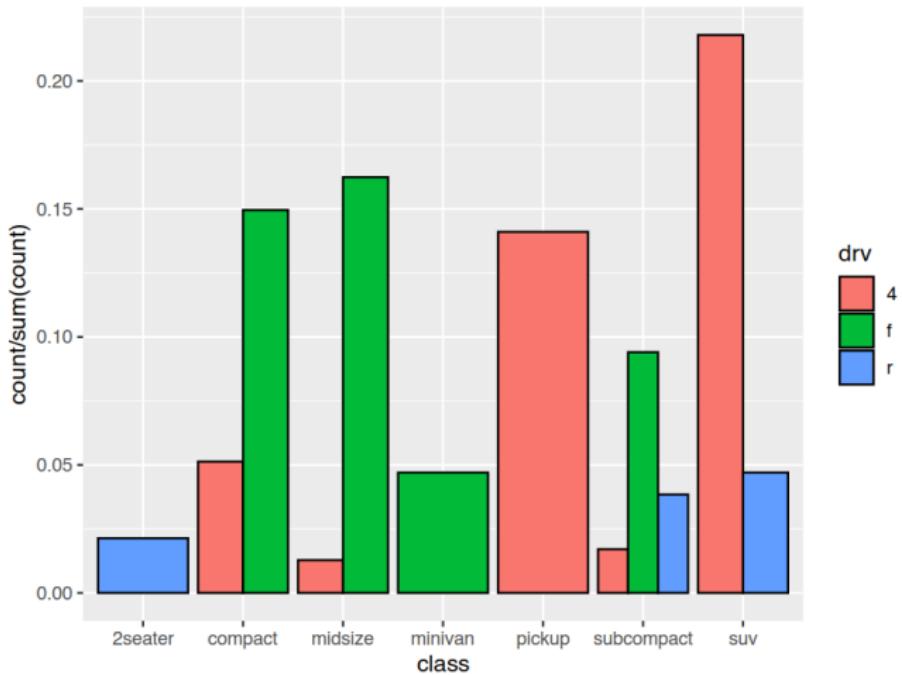
Figure 7: Side-by-side variant of the number of vehicles bar chart.

Position Adjustments (2/2)
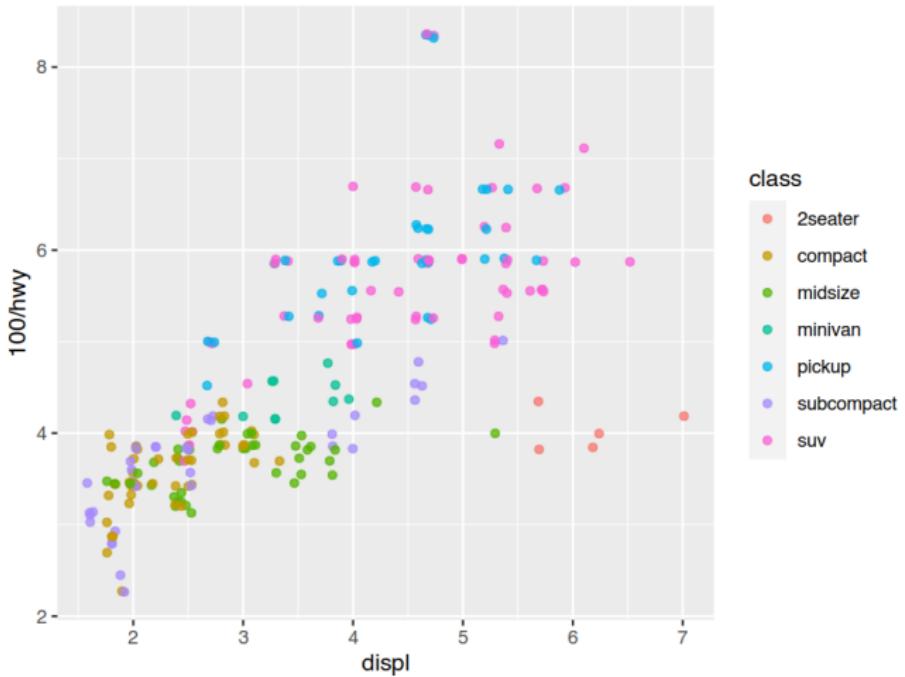
Figure 8: Jittering reveals a concentration of overlapping points.

Coordinate Systems
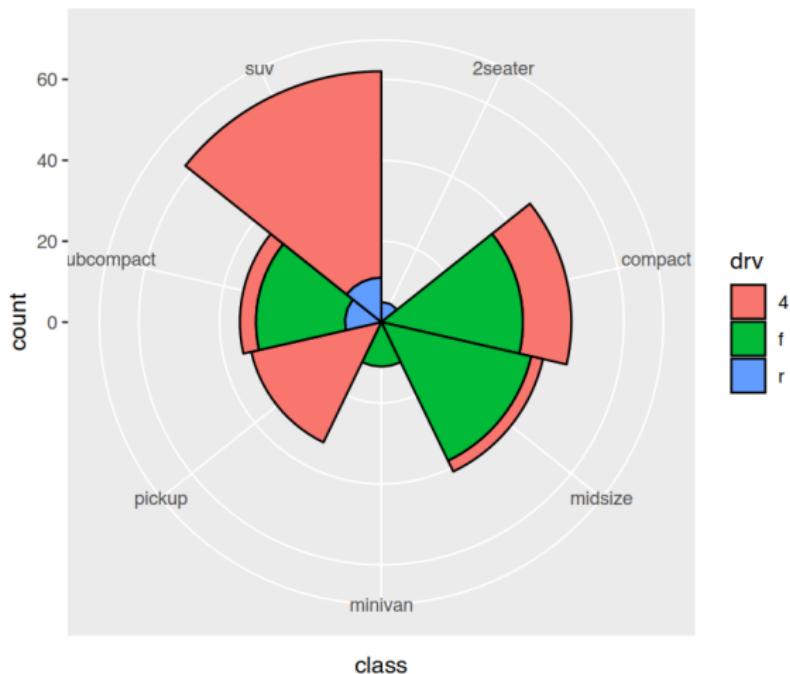
Figure 9: A wild rose, which isn't generally recommended despite looking sophisticated.

Scales and Labels (1/4)
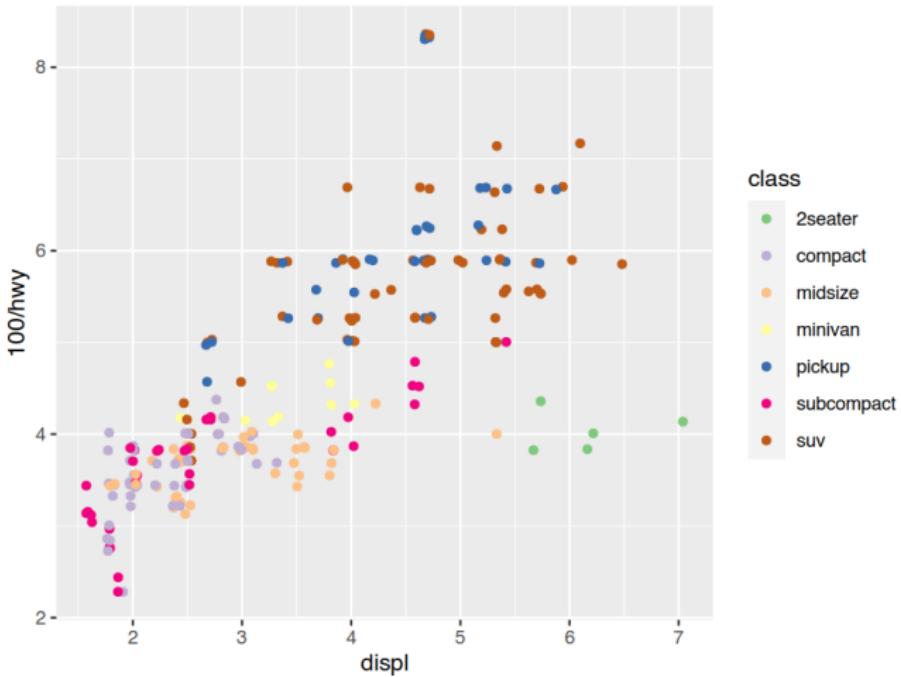
Figure 10: The ColorBrewer schemes are a good choice for discrete data.
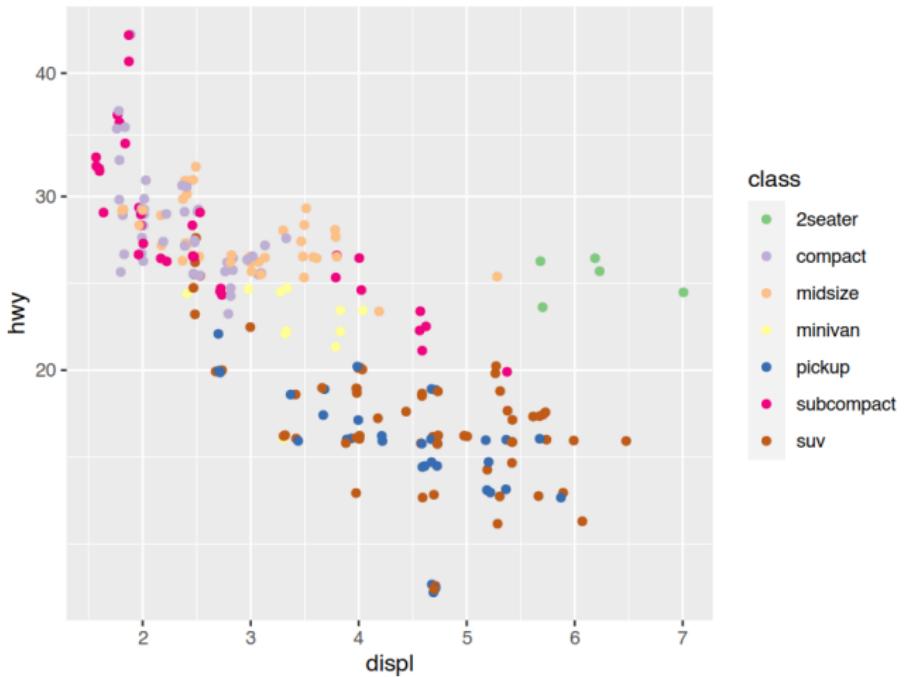
Figure 11: Using a log scale also expands out the lower end of miles-per-gallon.

Scales and Labels (3/4)
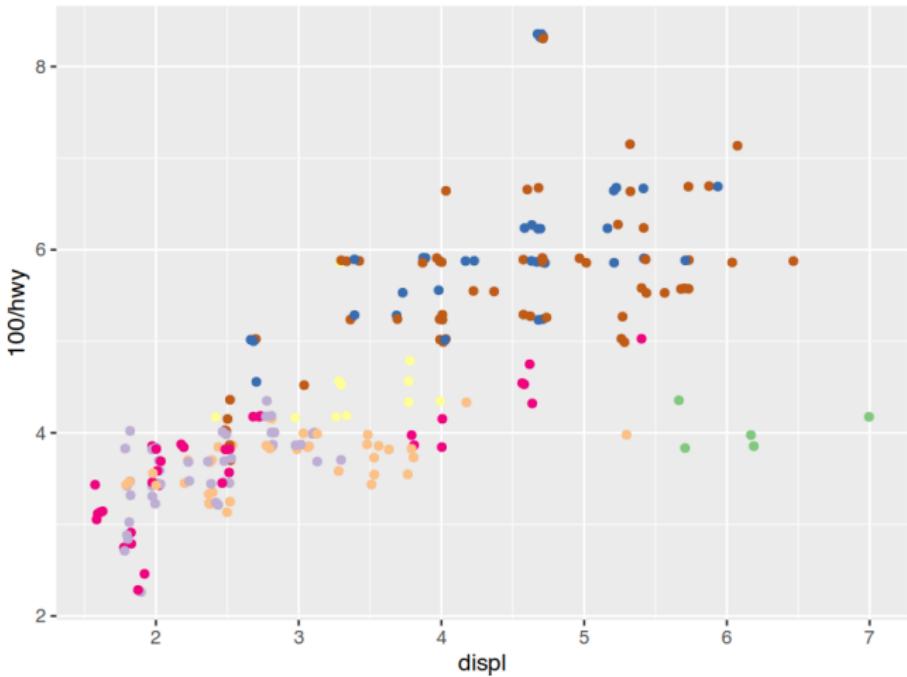
Figure 12: Legends can be customized (or removed) with guides.

Scales and Labels (4/4)
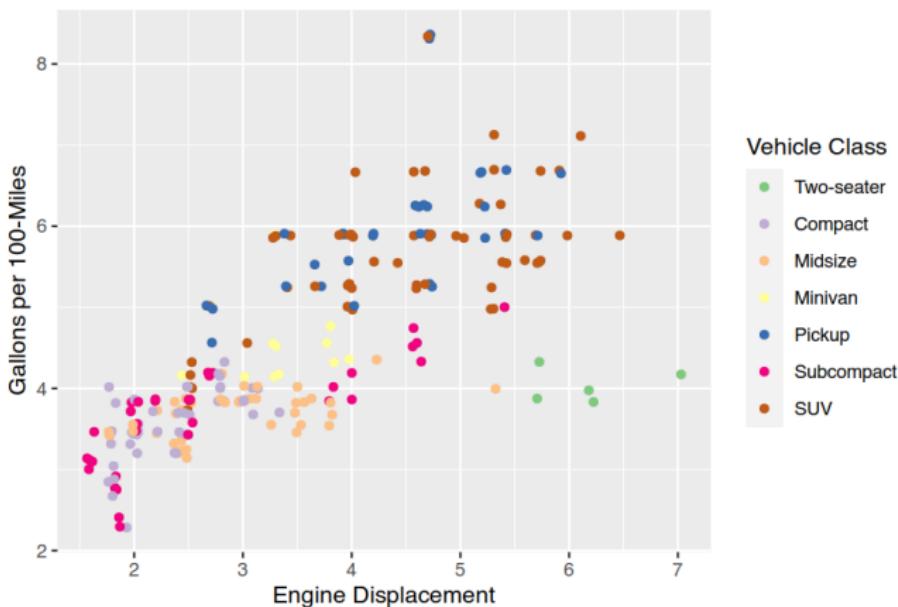
Figure 13: Labels are easily specified via the a variety of convenience functions.

Themes
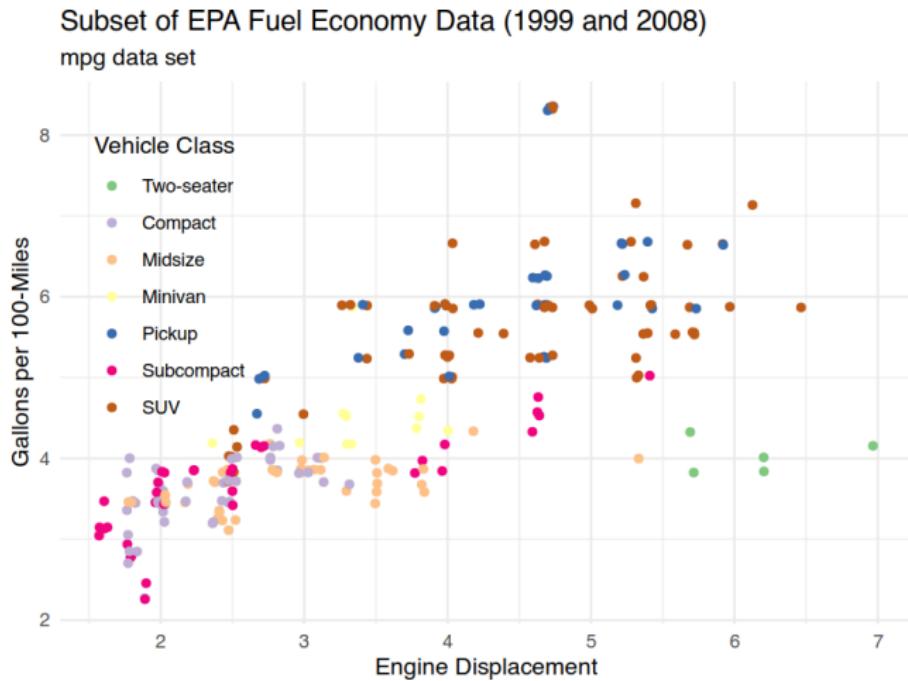
Figure 14: Themes make it easy to tweak non-data apperances.

Creativity
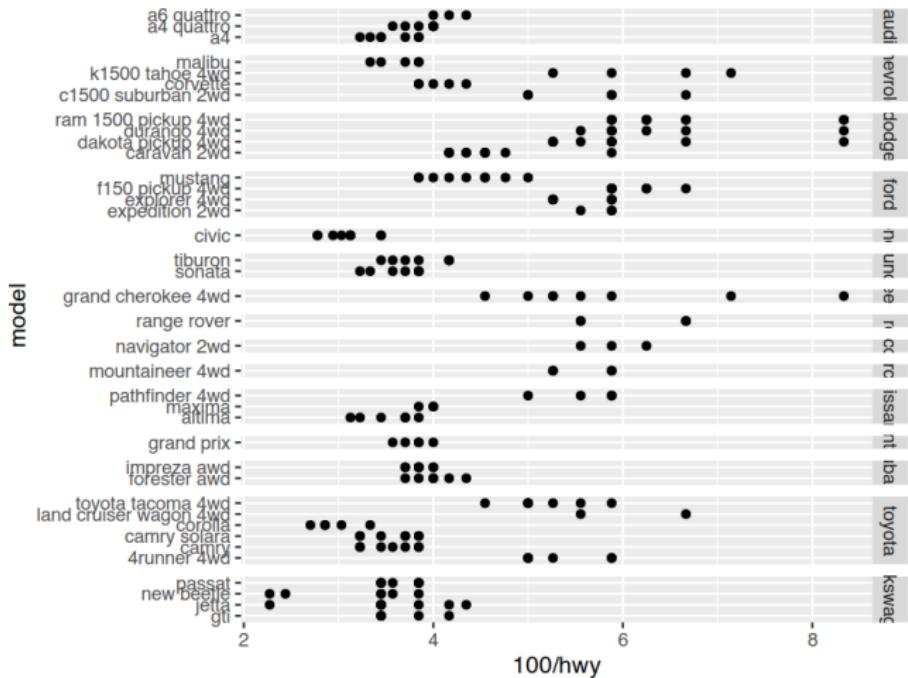
Figure 15: Creative applications of the grammar can produce almost any graph.