L02 Data Visualization

Data Science I (STAT 301-1)

Shay Lebovitz

Contents

Overview **1**

Datasets **1**

Exercises **1**

Challenges **18**

Overview

The goal of this lab is to start building the skills to visualize data using the `ggplot2` package in R. Students will also learn to access and utilize R documentation.

Datasets

This lab utilizes the `mpg` and `diamonds` datasets. Both come with `ggplot2` and their documentation/codebooks can be accessed with `?mpg` and `?diamonds`, provided you have installed and loaded `ggplot2` to your current R session.

Exercises

Please complete the following exercises. Be sure your solutions are clearly indicated and that the document is neatly formatted.

Load Packages You should always begin by loading all necessary packages towards the beginning of your documents. Assume that that all necessary packages have been installed. User should be able to determine if a package needs to be installed either through knowing their R repository or an error message. **Your code should never have install commands.**

```
# Loading package(s)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.4     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

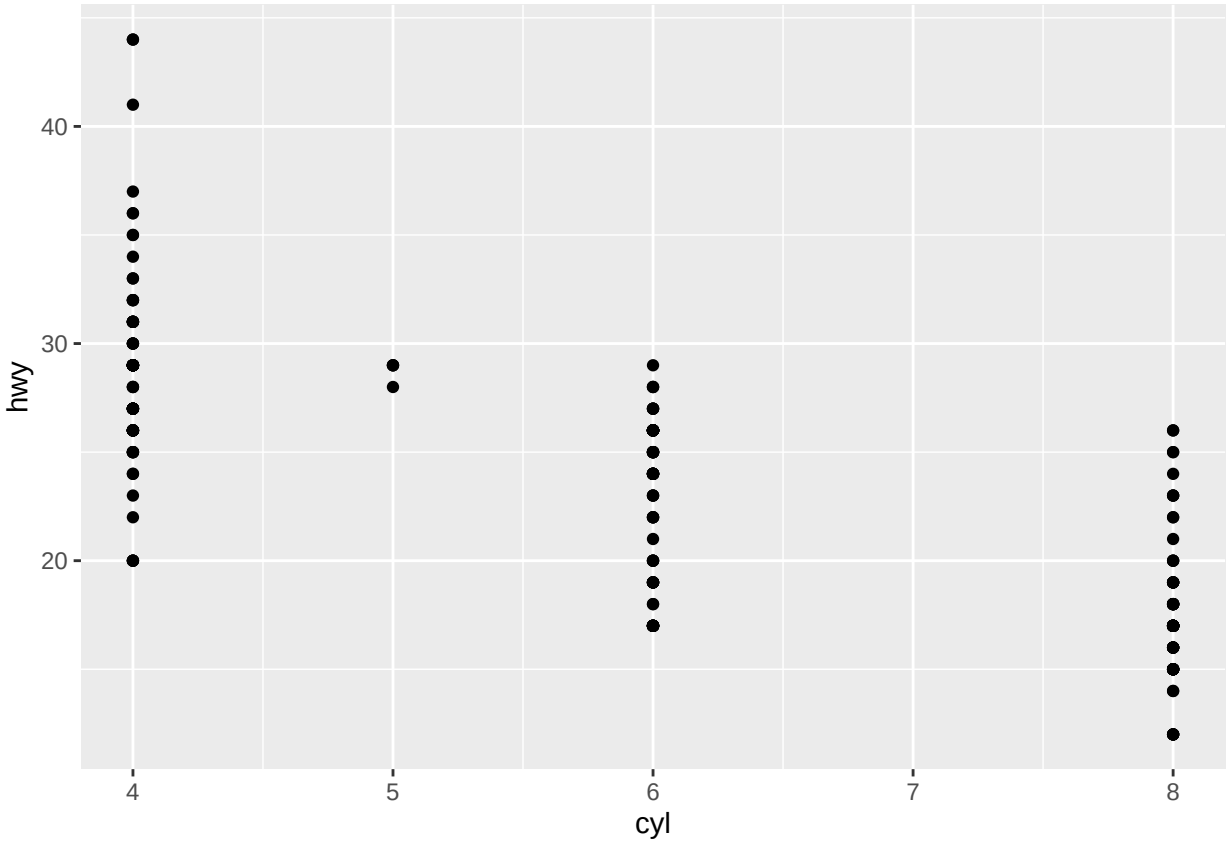
Exercise 1 There are three particularly important parameters to our template for building a graphic with `ggplot2`. They are `<DATA>`, `<GEOM_FUNCTION>`, and `<MAPPINGS>`. The importance of `<DATA>` is obvious. `<GEOM_FUNCTION>` is referring to the selection of a **geom**. `<MAPPINGS>`, specifically `aes(<MAPPINGS>)`, is referring to the process of defining **aesthetic mappings**.

- What is a **geom**?
- What is an **aesthetic mapping**?

A `geom` is a geometric object, a layer of a graph. An `aesthetic` mapping describes how objects are mapped aesthetically

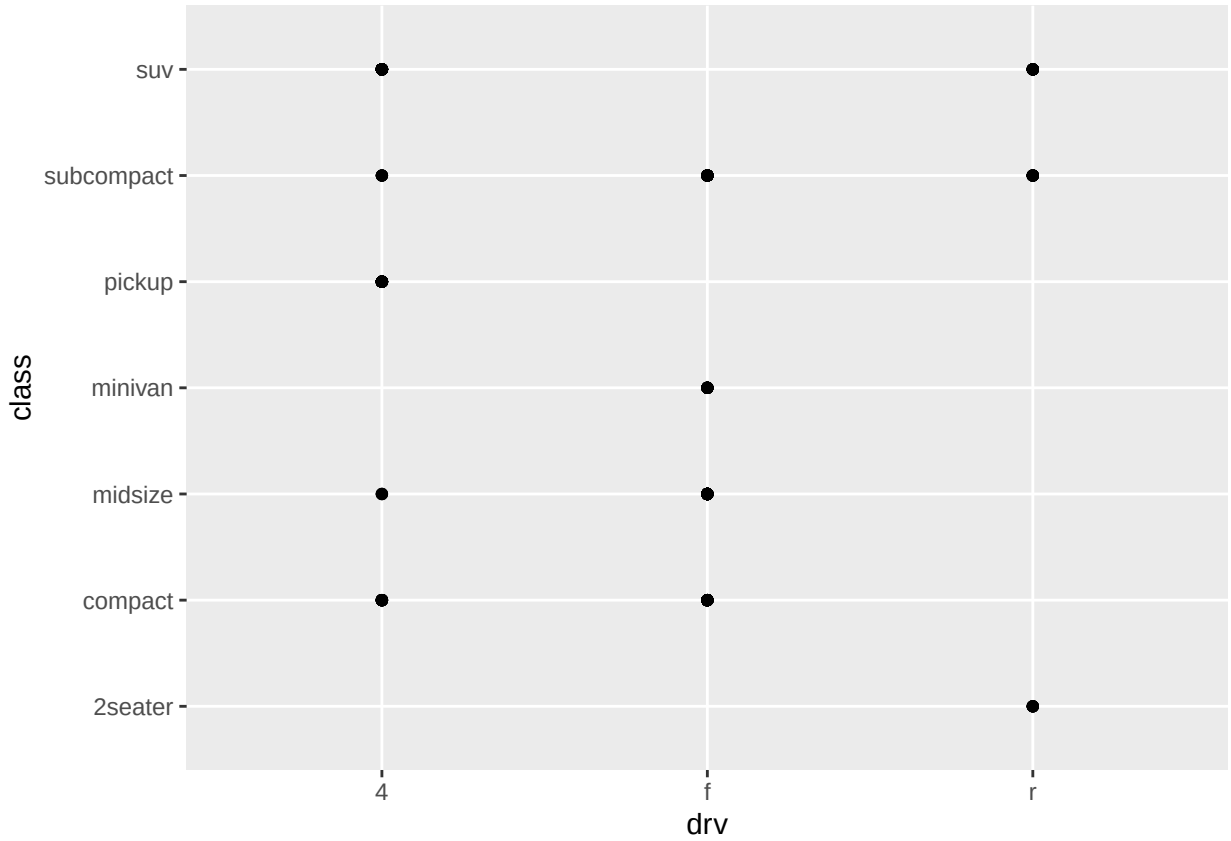
Exercise 2 (Website: 3.2.4 Ex. 4) Make a scatterplot of `hwy` vs `cyl`.

```
ggplot(data = mpg) +
  geom_point(mapping = aes(x = cyl, y = hwy))
```



Exercise 3 (Website: 3.2.4 Ex. 5) What happens if you make a scatterplot of `class` vs `drv`? What is the major drawback of this plot — really limits the plots usefulness?

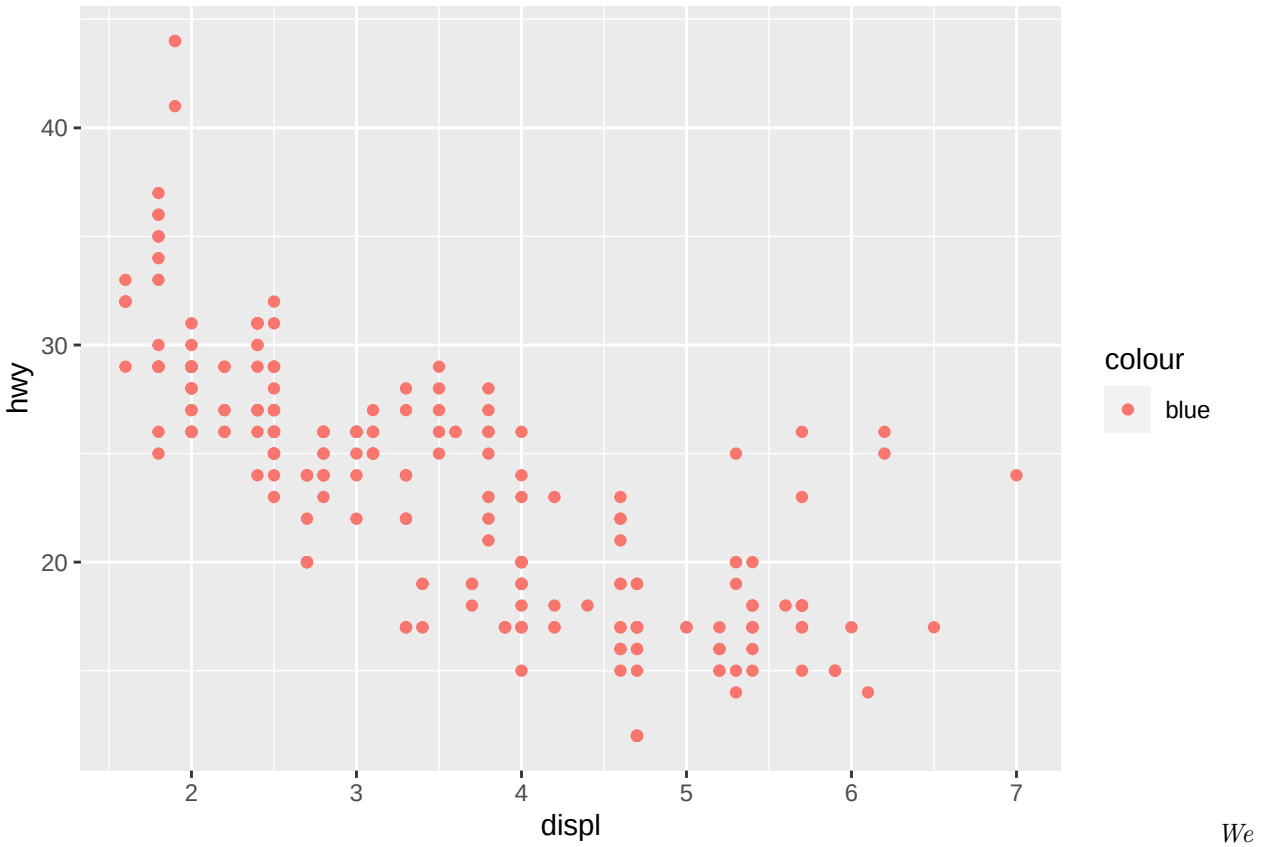
```
ggplot(data = mpg) +
  geom_point(mapping = aes(x = drv, y = class))
```



This plot is not numerical, making it a scatter plot doesn't allow you to see the shape of data

Exercise 4 (Website: 3.3.1 Ex. 1) What's gone wrong with this code? Why are the points not blue?

```
ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ, y = hwy, color = "blue"))
```

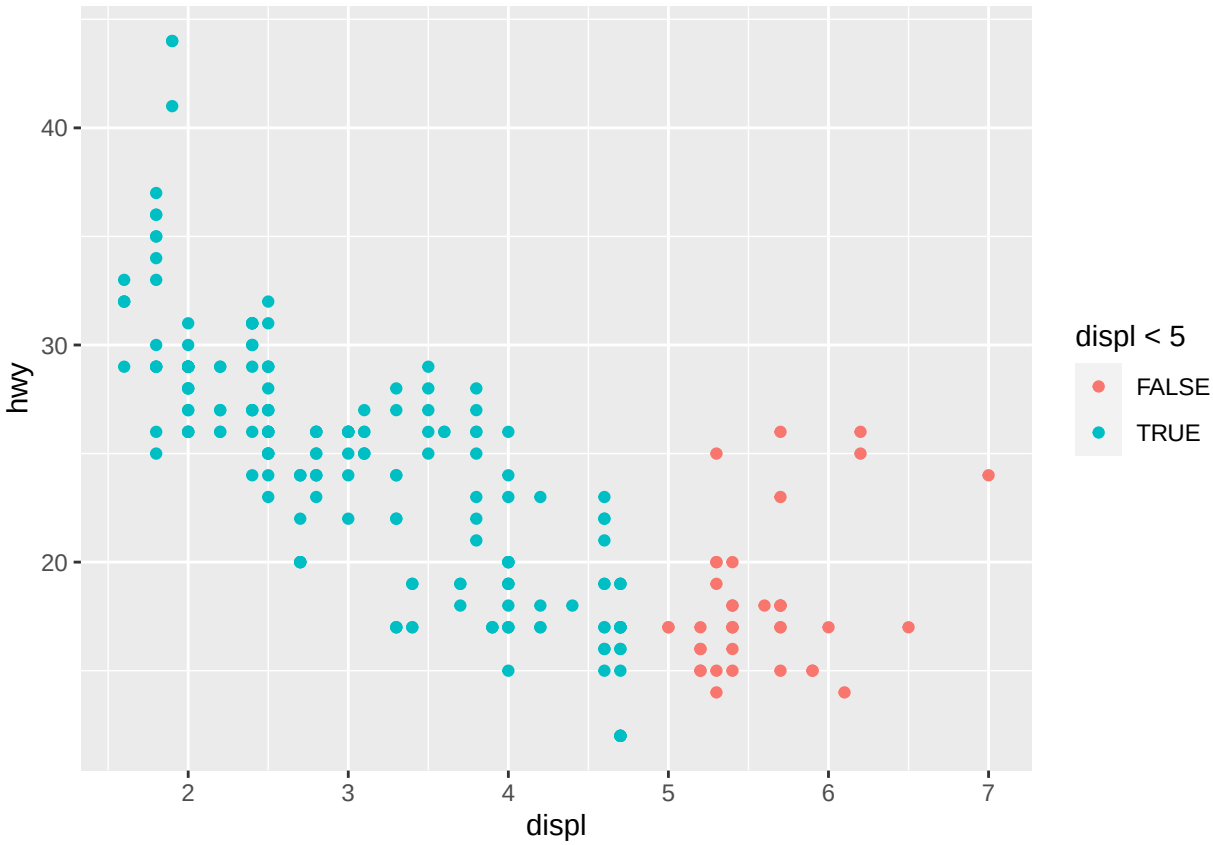


We mapped the string "blue" to the color aesthetic, but "blue" is not a variable. To fix this, put color outside the `aes()` function.

Exercise 5 (Website: 3.3.1 Ex. 5) What does the `stroke` aesthetic do? What shapes does it work with? (Hint: use `?geom_point`) *`stroke` controls the width of the border of certain points. It works for shapes that have a border, so 0,1,2,5,6,7,9,10,11,12,13,14,21-25.*

Exercise 6 (Website: 3.3.1 Ex. 6) What happens if you map an aesthetic to something other than a variable name, like `aes(colour = displ < 5)`?

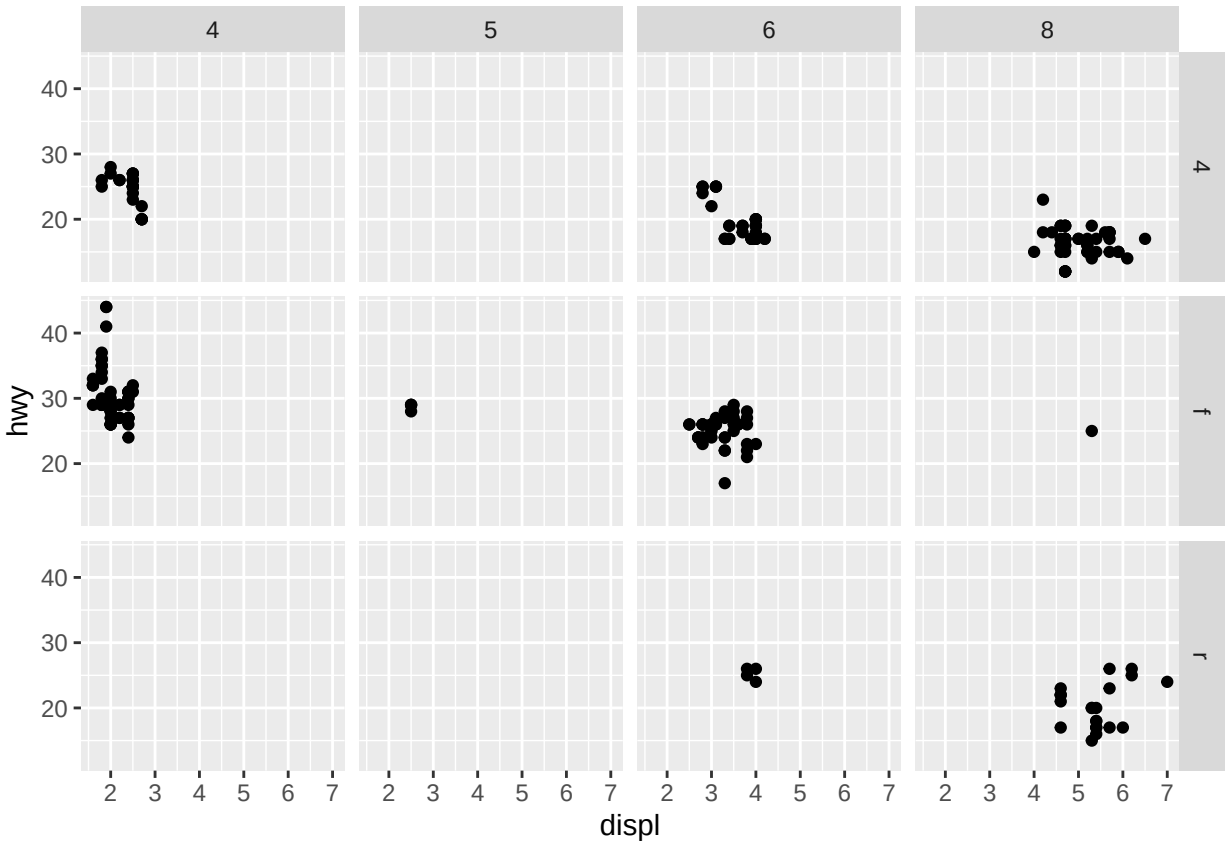
```
ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ, y = hwy, color = displ < 5))
```



Colors points depending on whether they meet the conditional statement

Exercise 7 (Website: 3.5.1 Ex. 2) What do the empty cells in the plot below plot with `facet_grid(drv ~ cyl)` mean?

```
ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ, y = hwy)) +
  facet_grid(drv ~ cyl)
```



The empty cells imply that there are no data points that fall into those subcategories

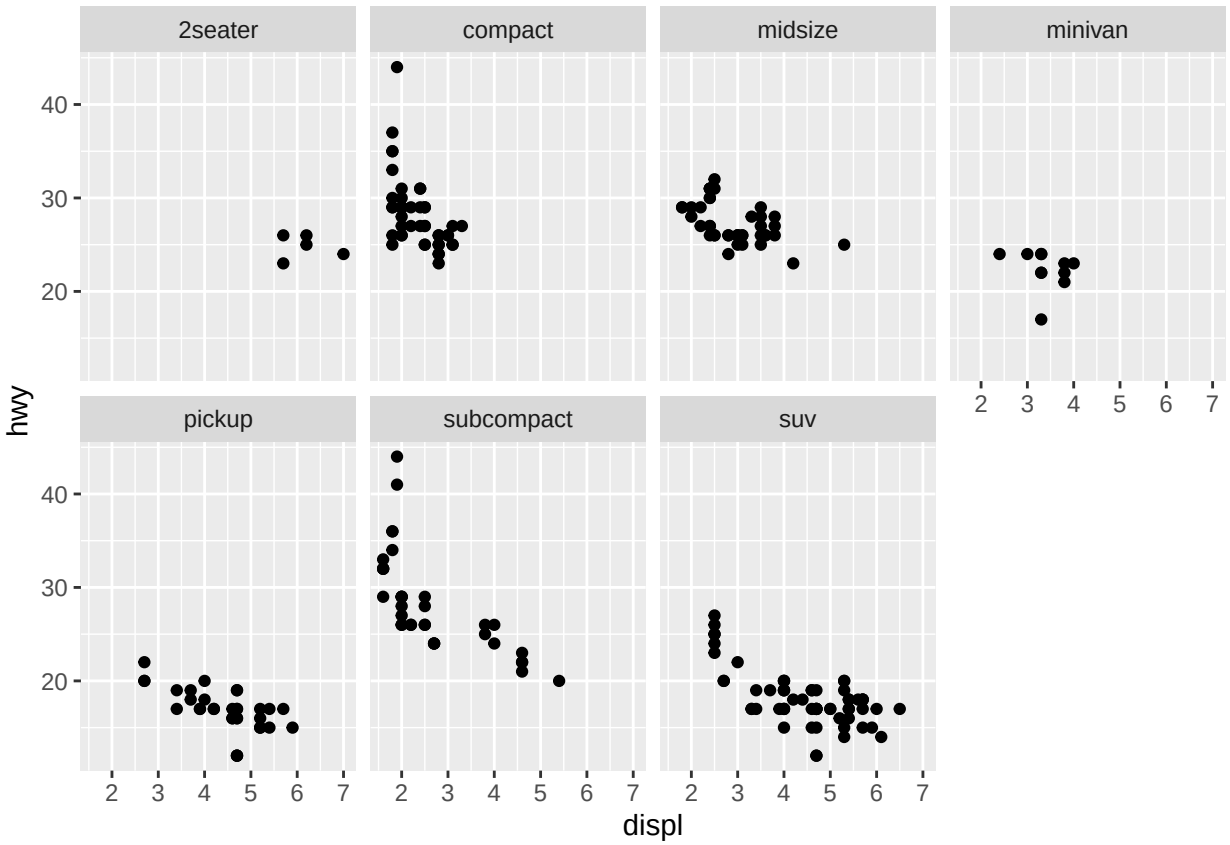
How do they relate to this plot?

```
ggplot(data = mpg) +
  geom_point(mapping = aes(x = drv, y = cyl))
```

This plot tells you if there are points that fall into the subcategories defined by `drv` *and* `cyl`*.*

Exercise 8 (Website: 3.5.1 Ex. 4) Given the faceted plot:

```
ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ, y = hwy)) +
  facet_wrap(~ class, nrow = 2)
```



What are the advantages to using faceting instead of the color aesthetic? What are the disadvantages? How might the balance change if you had a larger dataset?

Faceting allows for easier visualization for large data sets, as it splits up the subcategories into their own graphs, and thus overlap is not as big of a problem. For small data sets, it might be useful to use the color aesthetic so that all the points are on the same graph

Exercise 9 (Website: 3.5.1 Ex. 5) Read `?facet_wrap`. What does `nrow` do? What does `ncol` do? What other options control the layout of the individual panels? Why doesn't `facet_grid()` have `nrow` and `ncol` argument?

`nrow` and `ncol` determine the number of rows and columns for the graph. `scales`, `shrink`, `as.table`, `switch`, `drop`, and `dir` also affect the layout of the graph. `facet_grid()` doesn't have an nrow or ncol argument because they will be pre-determined by the number of factors in the variables called.

Exercise 10 (Website: 3.5.1 Ex. 6) When using `facet_grid()` you should usually put the variable with more unique levels in the columns. Why?

Screens are normally wider than they are tall, so it is a better use of space to put the variable with more unique levels in the columns section so it can go horizontally across the screen.

Exercise 11 (Website: 3.6.1 Ex. 1) What geom would you use to draw a line chart? A boxplot? A histogram? An area chart?

Use `geom_line()` for line plot, geom_boxplot()`for box plot`,geom_histogram()`for a histogram`,geom_area()‘ for an area plot.

Exercise 12 Suppose we have a dataset named `dat` containing the variables `weight`, `height`, and `gender`. Predict what the output/graphic will look like for the code below.

```
ggplot(data = dat, mapping = aes(x = height, y = weight, color = gender)) +
  geom_point() +
  geom_smooth(se = FALSE)
```

You would greate a scatter plot of `weight` vs. `height`, color separated by `gender`, with smooth line showing the pattern in the data without standard error.

Exercise 13 (Website: 3.6.1 Ex. 5) Will these two graphs look different? Why/why not? — Try answering without running code and then check.

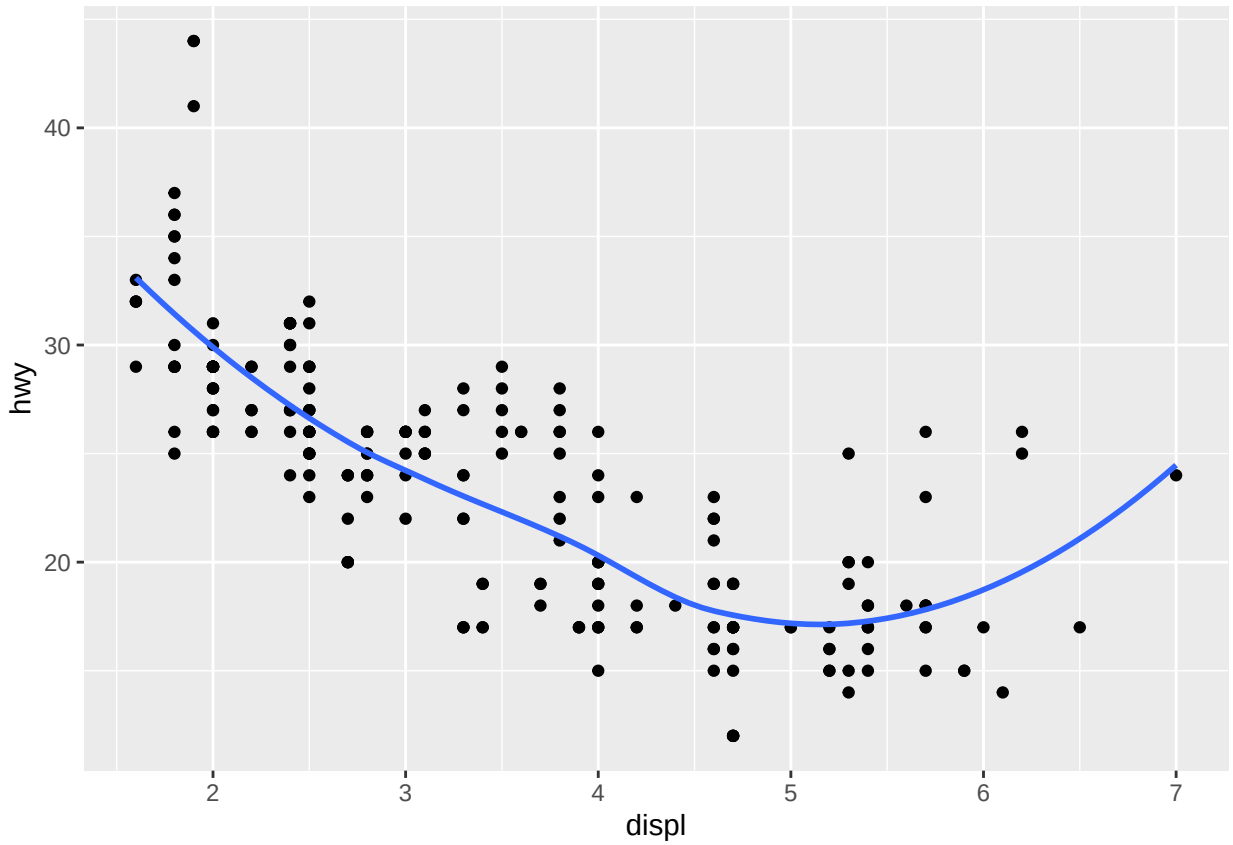
```
# Graph 1
ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point() +
  geom_smooth()

# Graph 2
ggplot() +
  geom_point(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_smooth(data = mpg, mapping = aes(x = displ, y = hwy))
```

At first glance it looks like they will do the same thing, the details on `ggplot()` should extend to the `geom_point()` and `geom_smooth` calls. And yes, they are the same.

Exercise 14 (Website: 3.6.1 Ex. 6) Recreate the R code necessary to generate the following graphs (6 total).

```
#a)
ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ, y = hwy)) +
  geom_smooth(mapping = aes(x = displ, y = hwy), se = FALSE)
```

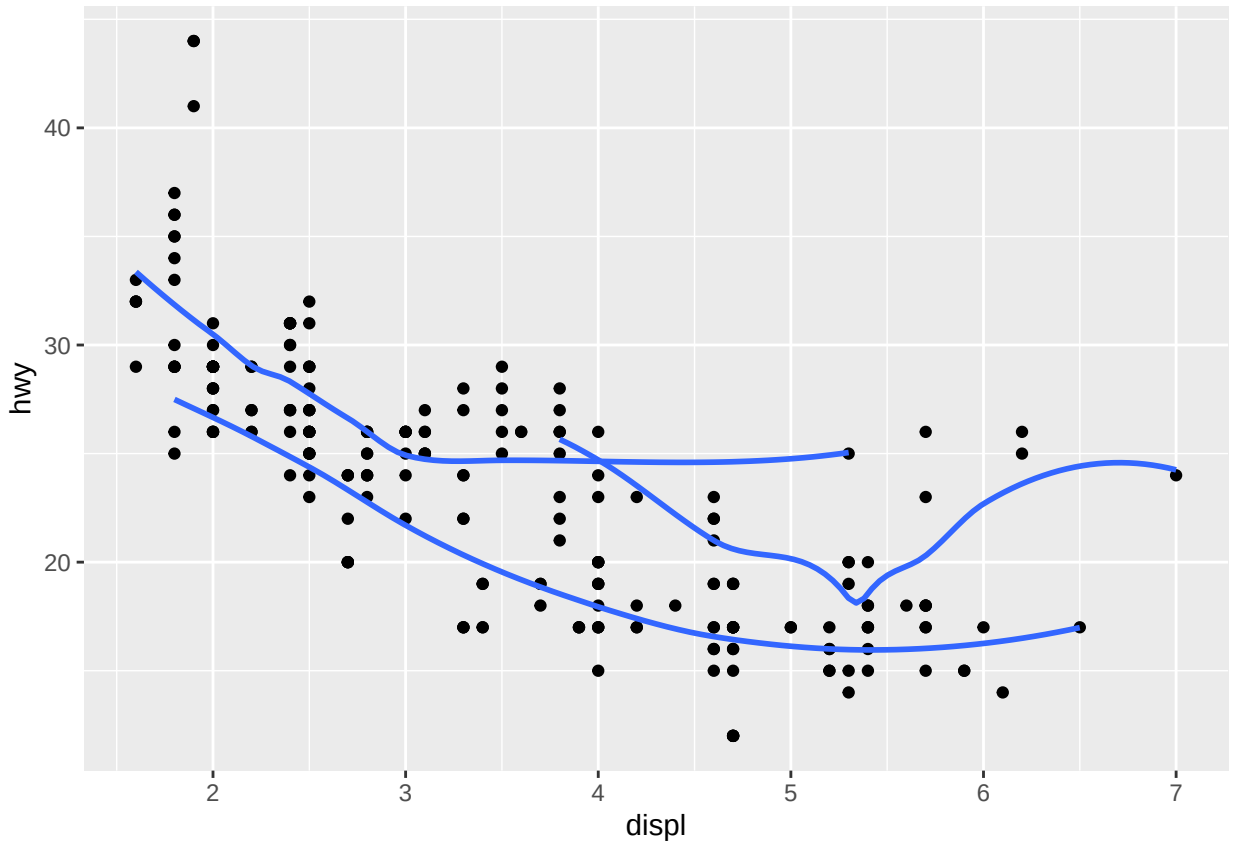
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```




```
#b)
ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ, y = hwy)) +
  geom_smooth(mapping = aes(x = displ, y = hwy, line = drv),
              se = FALSE)
```

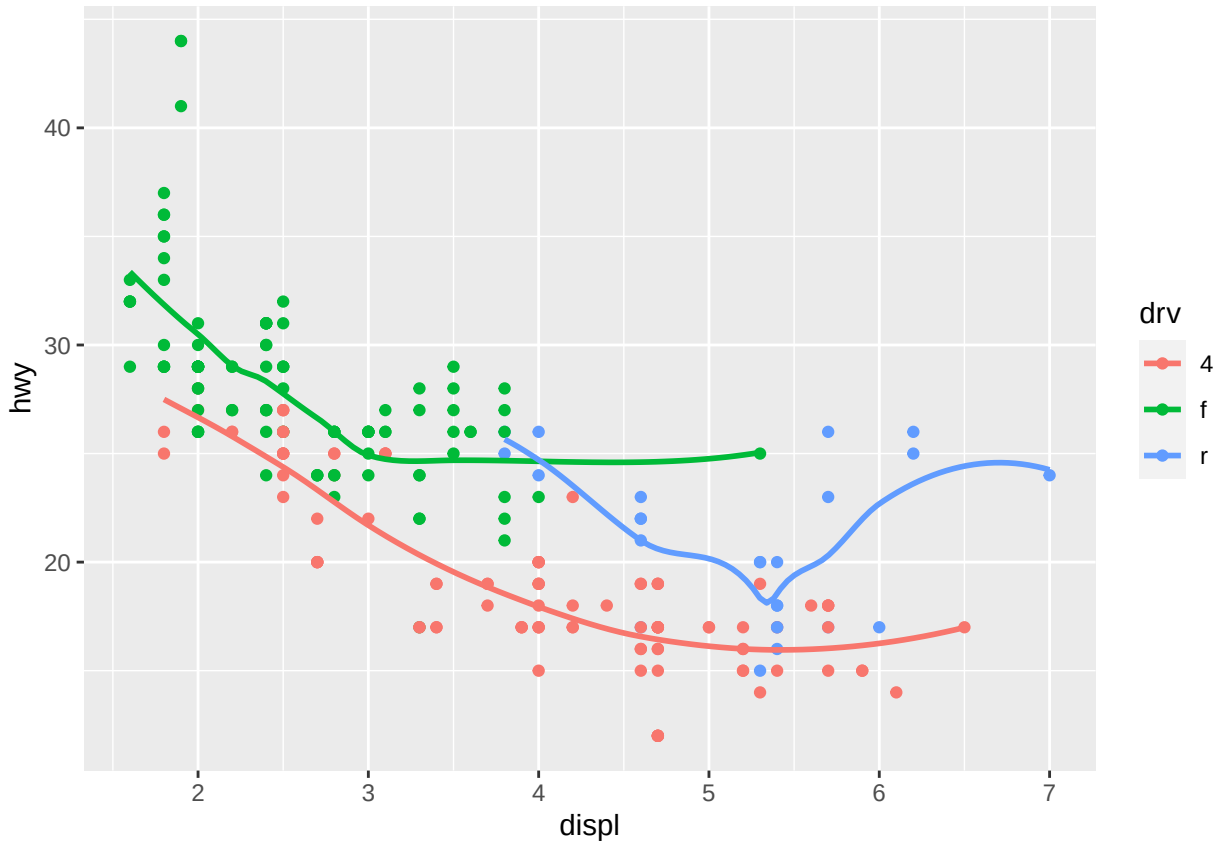
```
## Warning: Ignoring unknown aesthetics: line
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



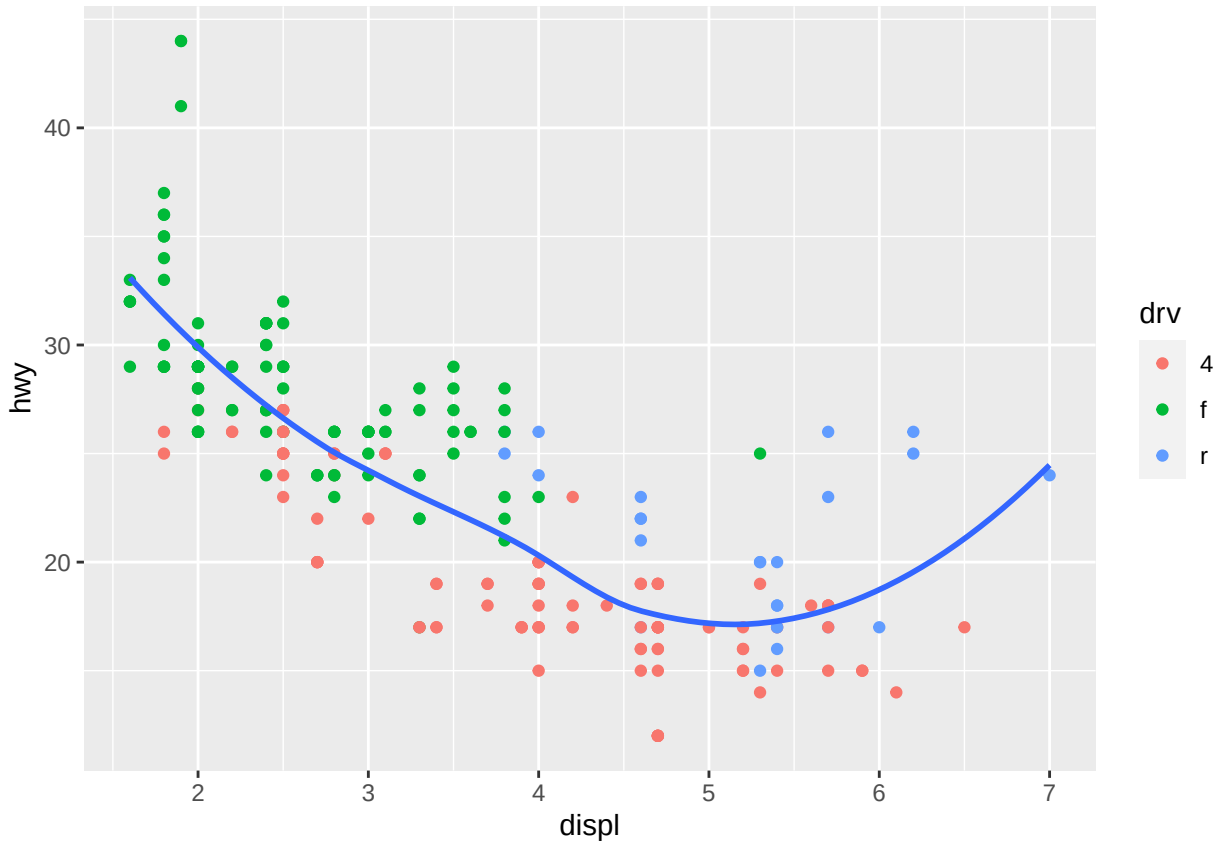

```
#c)
ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ, y = hwy, color = drv)) +
  geom_smooth(mapping = aes(x = displ, y = hwy, linetype = drv, color = drv),
              se = FALSE, linetype = 'solid')
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



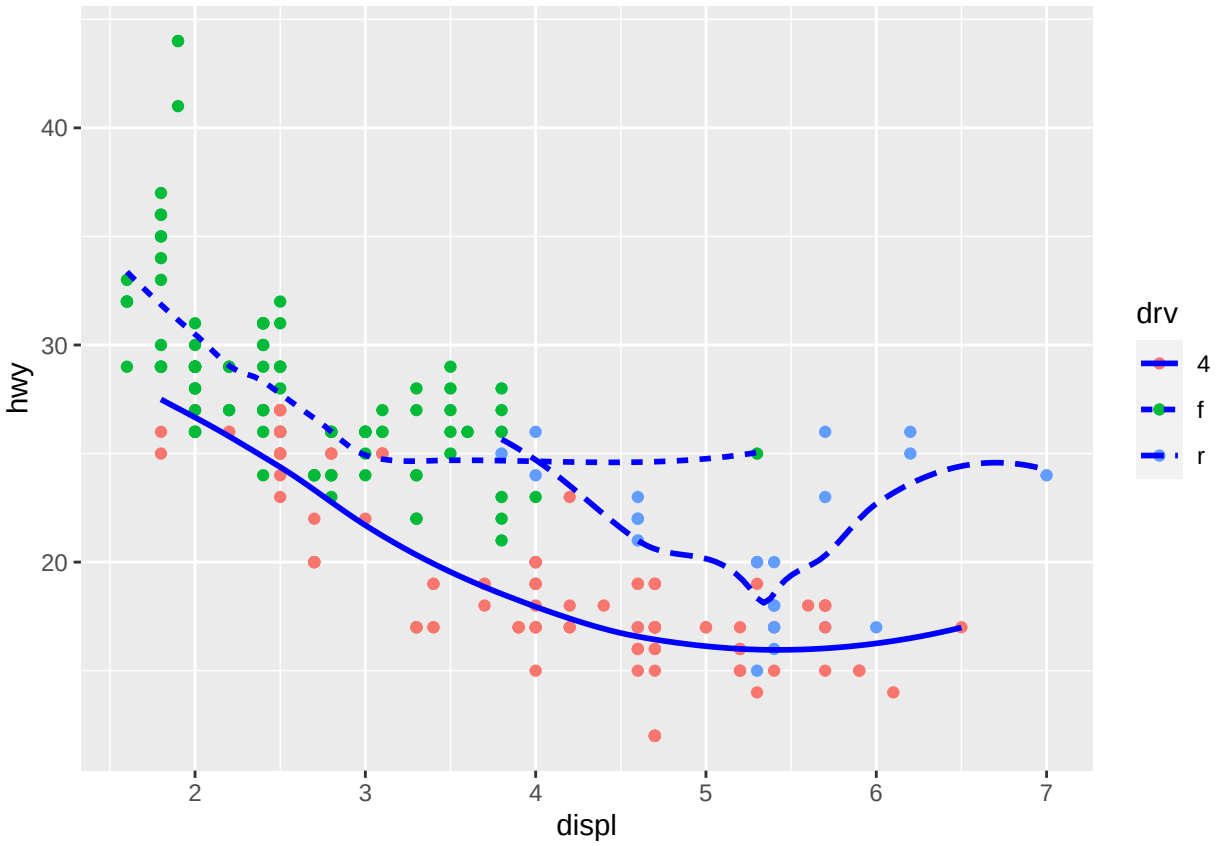

```
#d)
ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ, y = hwy, color = drv)) +
  geom_smooth(mapping = aes(x = displ, y = hwy), se = FALSE)
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```




```
#e)
ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ, y = hwy, color = drv)) +
  geom_smooth(mapping = aes(x = displ, y = hwy, linetype = drv),
              color = 'blue', se = FALSE)
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



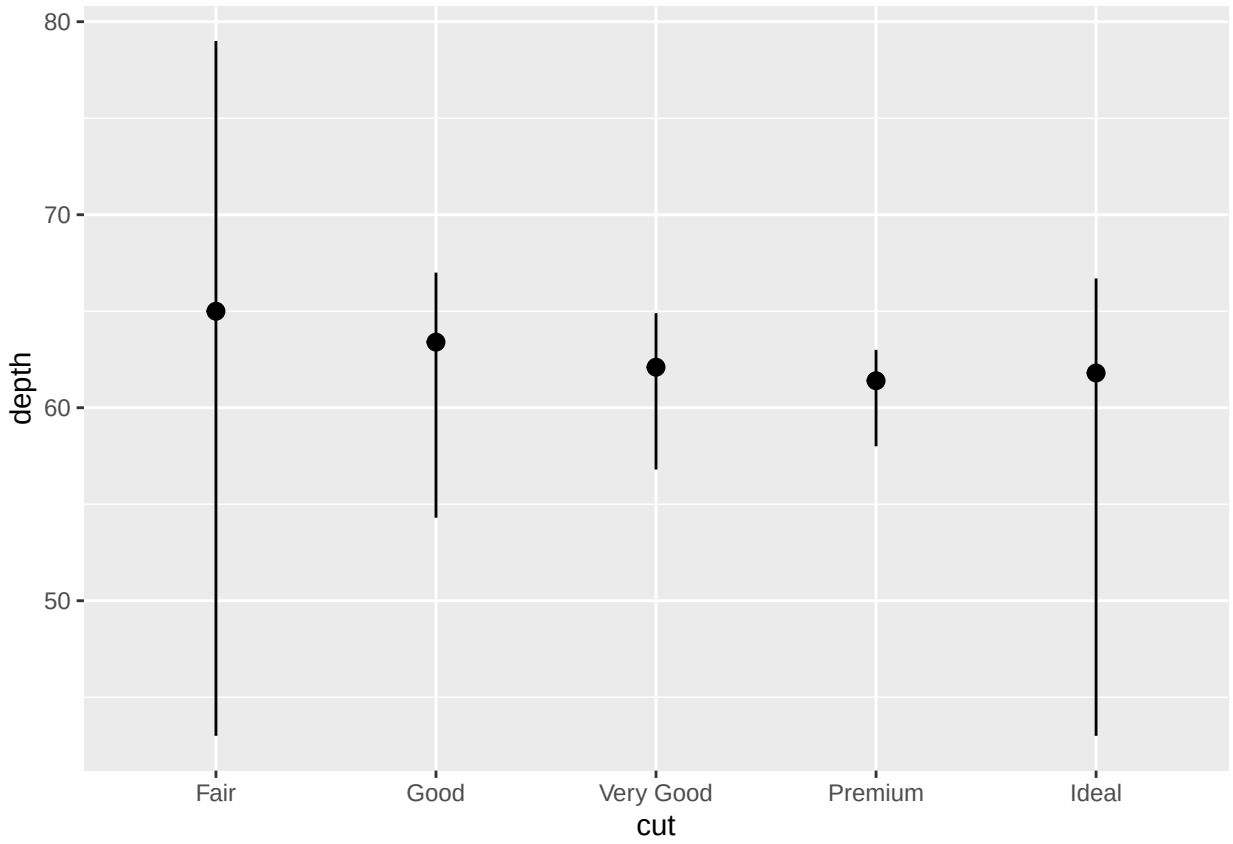
Exercise 15 (Website: 3.7.1 Ex. 1) What is the default geom associated with `stat_summary()`? How could you rewrite the plot below to use that geom function instead of the stat function? *`geom_pointrange()` is the default associated with `stat_summary()`*

```
ggplot(data = diamonds) +
  stat_summary(
    mapping = aes(x = cut, y = depth),
    fun.ymin = min,
    fun.ymax = max,
    fun.y = median
  )
```

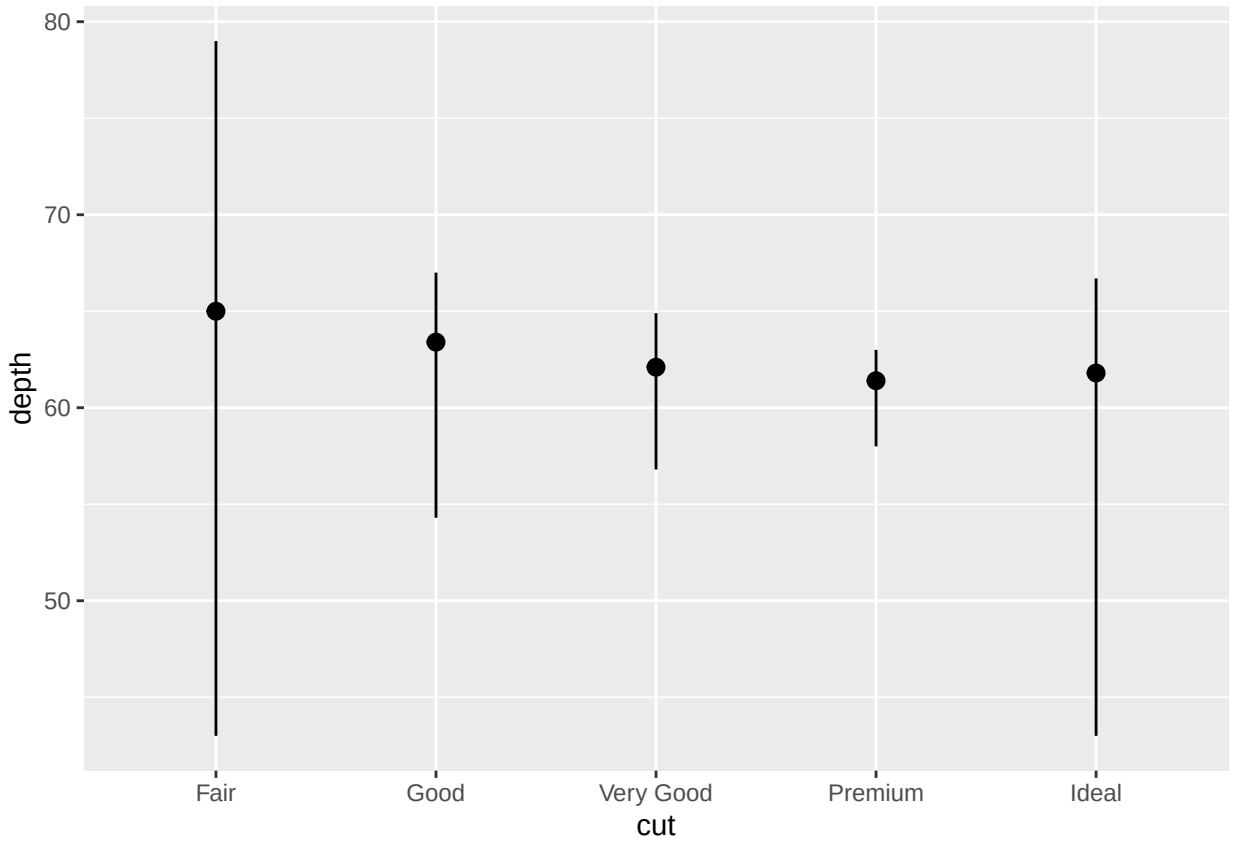
```
## Warning: `fun.y` is deprecated. Use `fun` instead.
```

```
## Warning: `fun.ymin` is deprecated. Use `fun.min` instead.
```

```
## Warning: `fun.ymax` is deprecated. Use `fun.max` instead.
```




```
ggplot(data = diamonds, mapping = aes(x = cut, y = depth)) +
  geom_pointrange(
    stat = 'summary',
    fun.min = min,
    fun.max = max,
    fun = median
  )
```

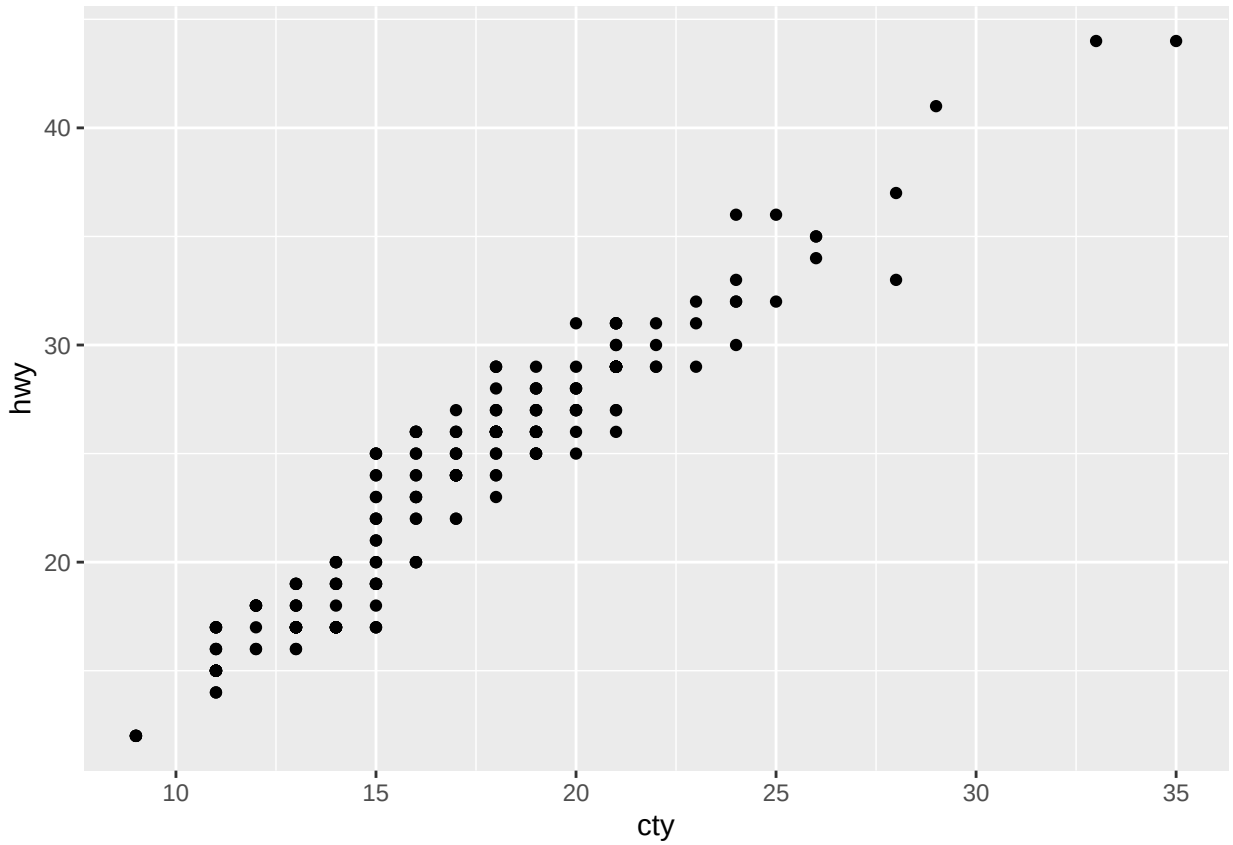


Exercise 16 (Website: 3.7.1 Ex. 4) What variables does `stat_smooth()` compute? In your own words, describe how the parameters `method`, `formula`, and `span` effect its behavior.

`stat_smooth` computes y (predicted value), `ymin` and `ymax` (lower and upper pointwise confidence interval around the mean), and `se` (standard error). `method` describes the smoothing mechanism that the function uses, Can be `lm`, `glm`, `gam`, `loess`, or a function of your choosing. Some work better than others, but also take different computation times. `formula` describes the formula used during the smoothing process. `span` describes to what extent the smoothing function works. More smooth or less smooth

Exercise 17 (Website: 3.8.1 Ex. 1) What is the problem with this plot? How could you improve it?

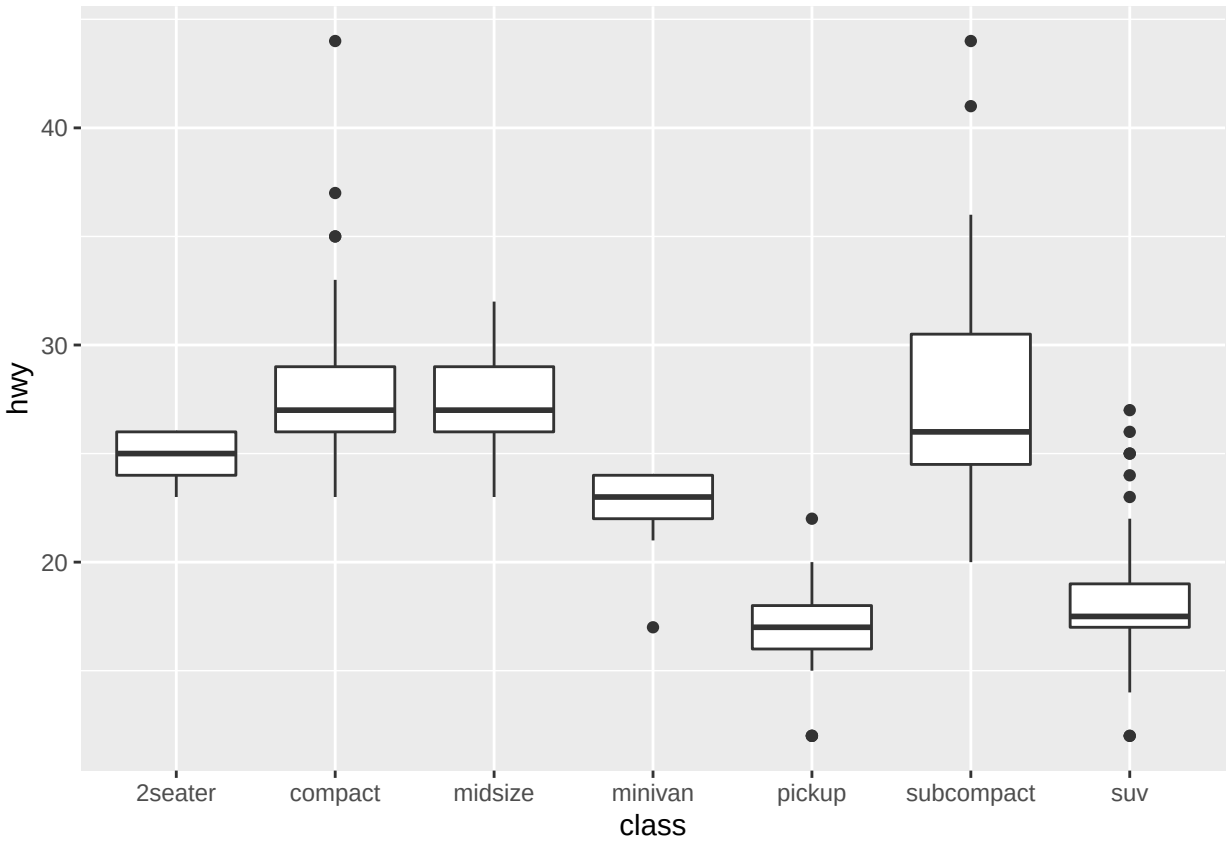
```
ggplot(data = mpg, mapping = aes(x = cty, y = hwy)) +
  geom_point()
```



The data points are rounded to reduce clutter, but that doesn't let you see where the mass of the data lies. Use `position = 'jitter'` *to fix it by introducing small variations in the data points so no two are exactly equal*

Exercise 18 (Website: 3.8.1 Ex. 4) What's the default position adjustment for `geom_boxplot()`? Create a visualization of the mpg dataset that demonstrates it.

```
ggplot(data = mpg) +
  geom_boxplot(mapping = aes(x = class, y = hwy))
```



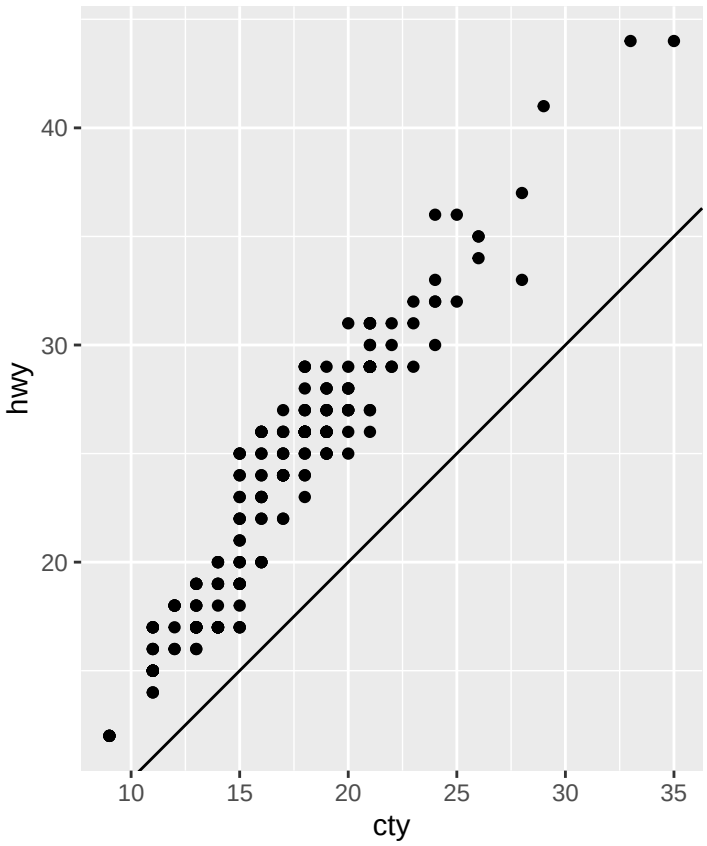
The default position is `dodge 2`

Exercise 19 (Website: 3.9.1 Ex. 2) What does `labs()` do? Read the documentation.

`labs()` allows you to modify the axis, legend, and plot labels

Exercise 20 (Website: 3.9.1 Ex. 4) What does the plot below tell you about the relationship between city and highway mpg? Why is `coord_fixed()` important? What does `geom_abline()` do?

```
ggplot(data = mpg, mapping = aes(x = cty, y = hwy)) +
  geom_point() +
  geom_abline() +
  coord_fixed()
```



A fairly linear relationship between city and highway mpg, with a slope of approximately 1. `geom_abline()` *adds a reference horizontal, vertical, or diagonal line.* `coord_fixed()` *ensures that the axes have a specific scale ratio, with the default set to 1. So, in this case, it ensures that the plot is square*

Exercise 21 In a few sentences, describe the approach to building graphics that is implemented in `ggplot2`.

In general, ggplot2 uses a layered system to build graphics. There are many different `geom` *functions available to create any type of plot you need, and* `ggplot2` *allows you to add multiple elements to the same graph to enhance data conveyance. It also allows you to facet or distinguish many different variables at once by groupings, colors, or sections. ****

Challenges

Students are not required to complete these. This section is for those wanting to go a little further with `ggplot2`.

Challenge A Attempt to recreate the following graphic. *Hint:* `ggthemes` *package*

Challenge B Recreate the following graphic. *Hint:* `scale_y_continuous()`