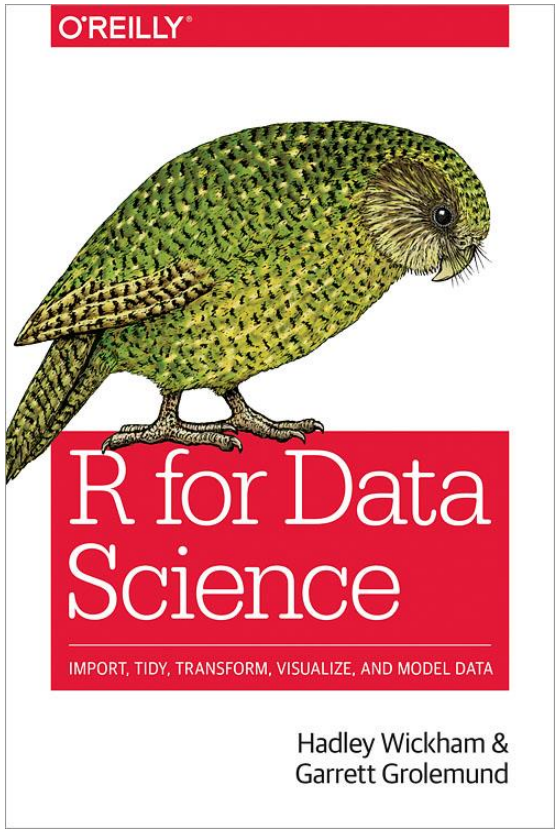
Session 3: Introduction to ggplot2

Andrew Jones | Strategy Unit

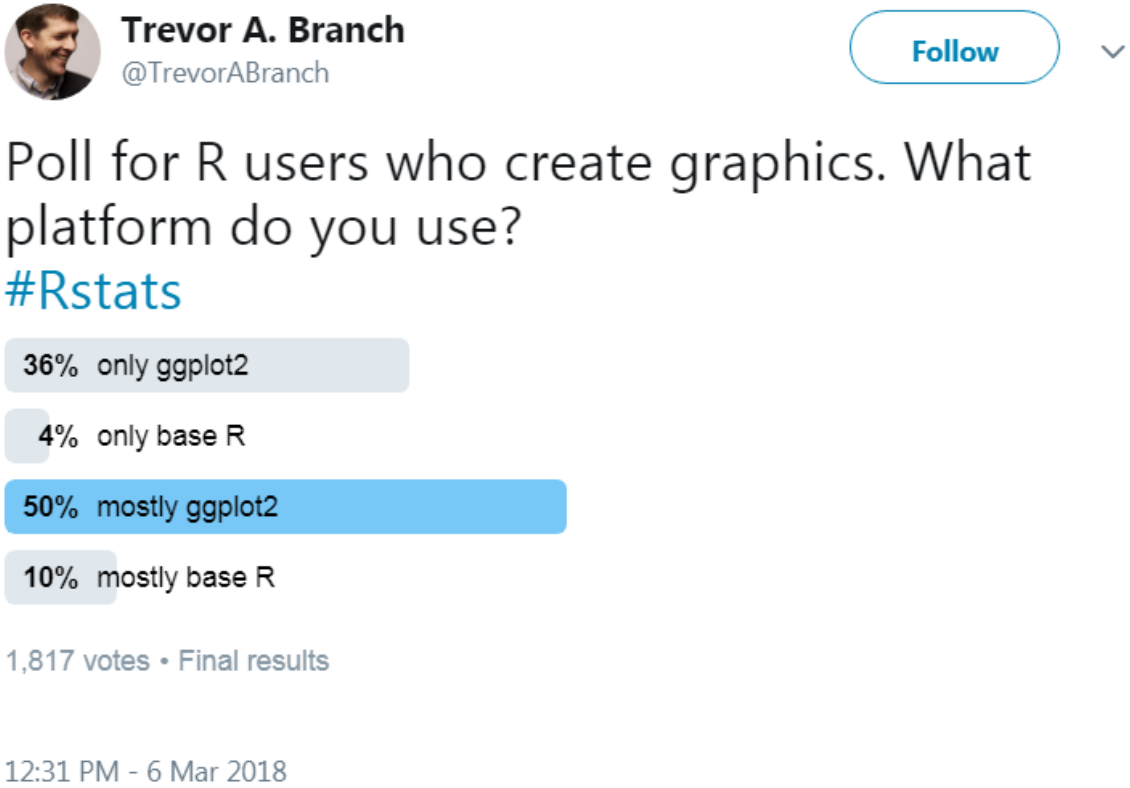
Acknowledgements

This session shadows Chapter 3 of *R for Data Science*



ggplot2

Is one of several plotting systems in R



Why ggplot2?

1. Easy to make good-looking plots

   (be careful that form doesn't eclipse function)

2. It meshes well with other principles we will be learning

Why: http://varianceexplained.org/r/teach_ggplot2_to_beginners/ Why not: https://simplystatistics.org/2016/02/11/why-i-dont-use-ggplot2/

ggplot2

ggplot2 is part of the tidyverse, so:

```
library(tidyverse)
```

mpg data

Data on car efficiency*. 38 models produced in both 1999 and 2008. Type:

```
mpg <- mpg
```

Source: US Environment Protection Agency https://fueleconomy.gov/

What is a data frame?

A data frame is a rectangular collection of variables (in columns) and observations (in rows).

id	gender	score
1	F	10.24
2	F	5.98
3	M	7.62

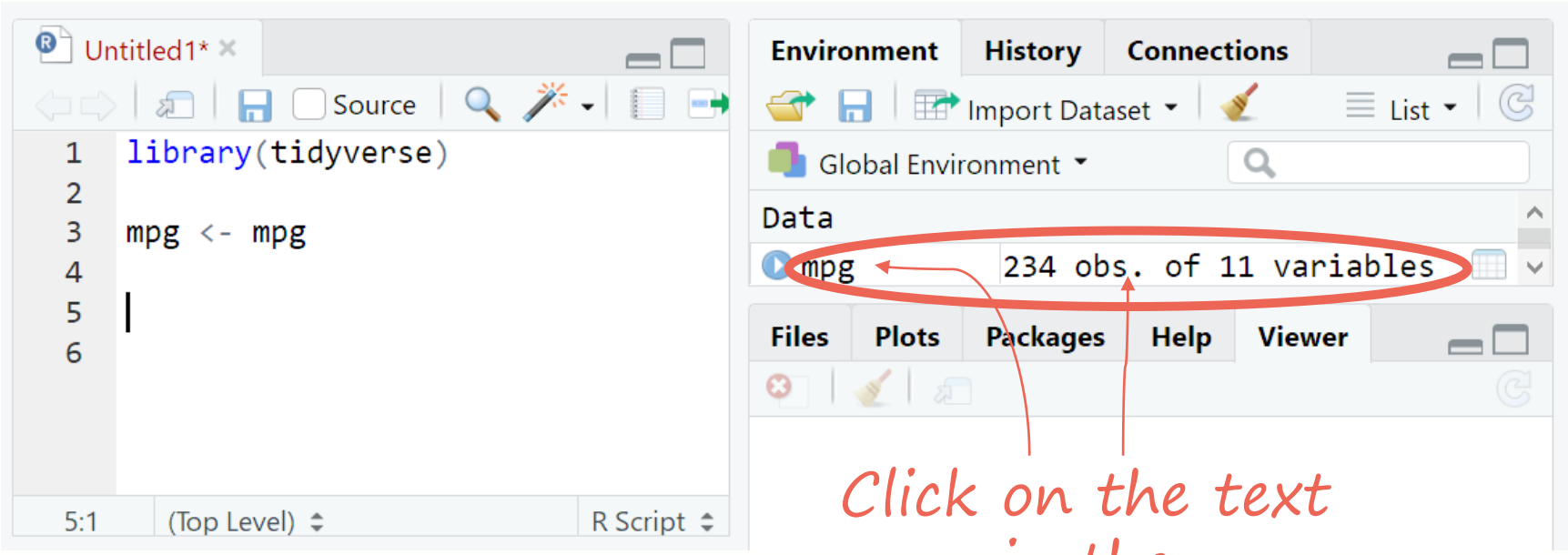
tibble = data frame

You may also come across the term "tibble". We'll take "tibble" to be synonymous with "data frame".

id	gender	score
1	F	10.24
2	F	5.98
3	M	7.62

Viewing the data

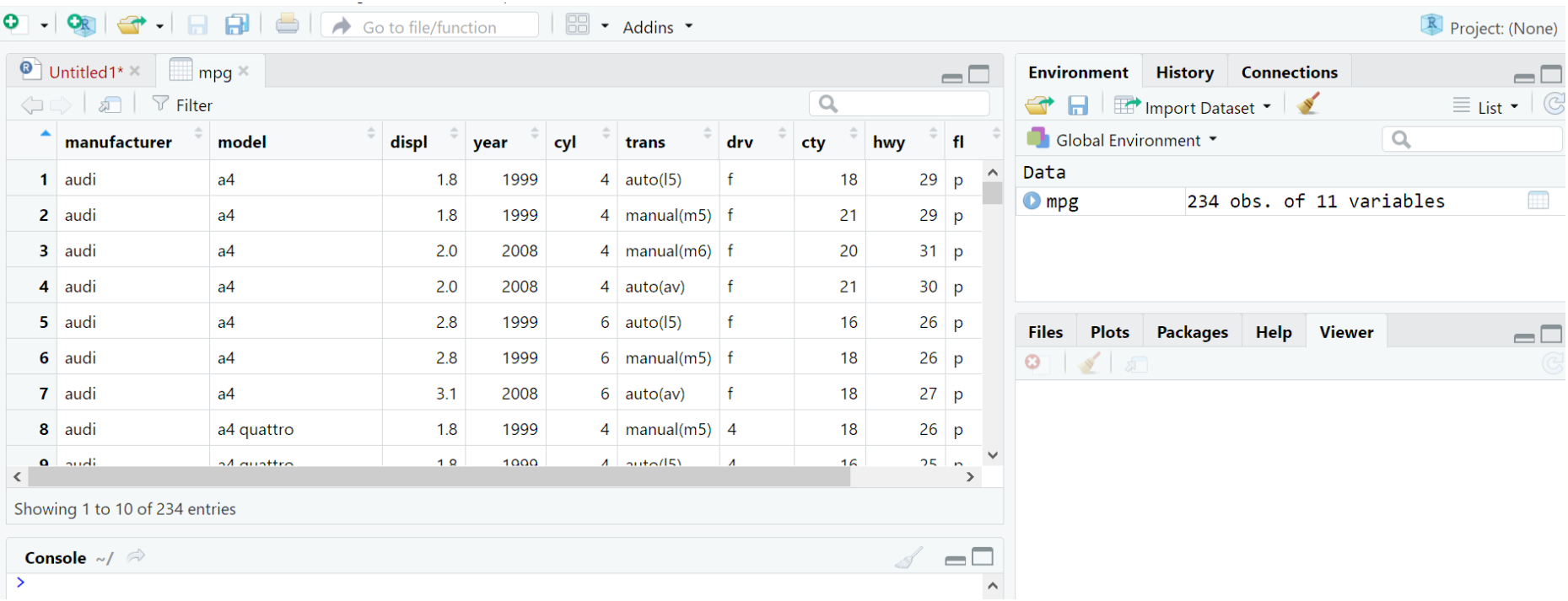
Several ways to look at a data frame. Option 1:



Click on the text in the Environment pane

Viewing the data

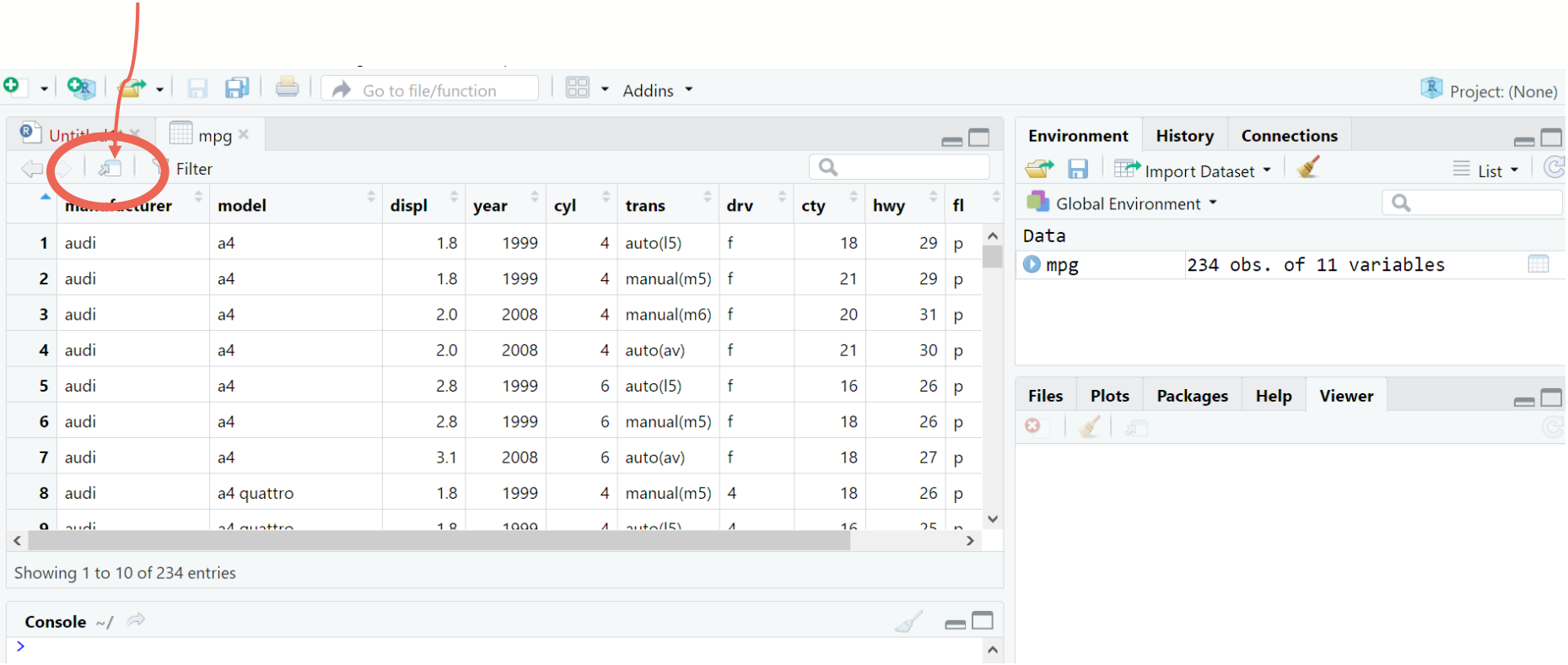
This brings up a view of the data in a new tab:



	manufacturer	model	displ	year	cyl	trans	drv	cty	hwy	fl
1	audi	a4	1.8	1999	4	auto(l5)	f	18	29	p
2	audi	a4	1.8	1999	4	manual(m5)	f	21	29	p
3	audi	a4	2.0	2008	4	manual(m6)	f	20	31	p
4	audi	a4	2.0	2008	4	auto(av)	f	21	30	p
5	audi	a4	2.8	1999	6	auto(l5)	f	16	26	p
6	audi	a4	2.8	1999	6	manual(m5)	f	18	26	p
7	audi	a4	3.1	2008	6	auto(av)	f	18	27	p
8	audi	a4 quattro	1.8	1999	4	manual(m5)	4	18	26	p

Viewing the data

Click here to show the data frame in a new window*



	manufacturer	model	displ	year	cyl	trans	drv	cty	hwy	fl
1	audi	a4	1.8	1999	4	auto(l5)	f	18	29	p
2	audi	a4	1.8	1999	4	manual(m5)	f	21	29	p
3	audi	a4	2.0	2008	4	manual(m6)	f	20	31	p
4	audi	a4	2.0	2008	4	auto(av)	f	21	30	p
5	audi	a4	2.8	1999	6	auto(l5)	f	16	26	p
6	audi	a4	2.8	1999	6	manual(m5)	f	18	26	p
7	audi	a4	3.1	2008	6	auto(av)	f	18	27	p
8	audi	a4 quattro	1.8	1999	4	manual(m5)	4	18	26	p

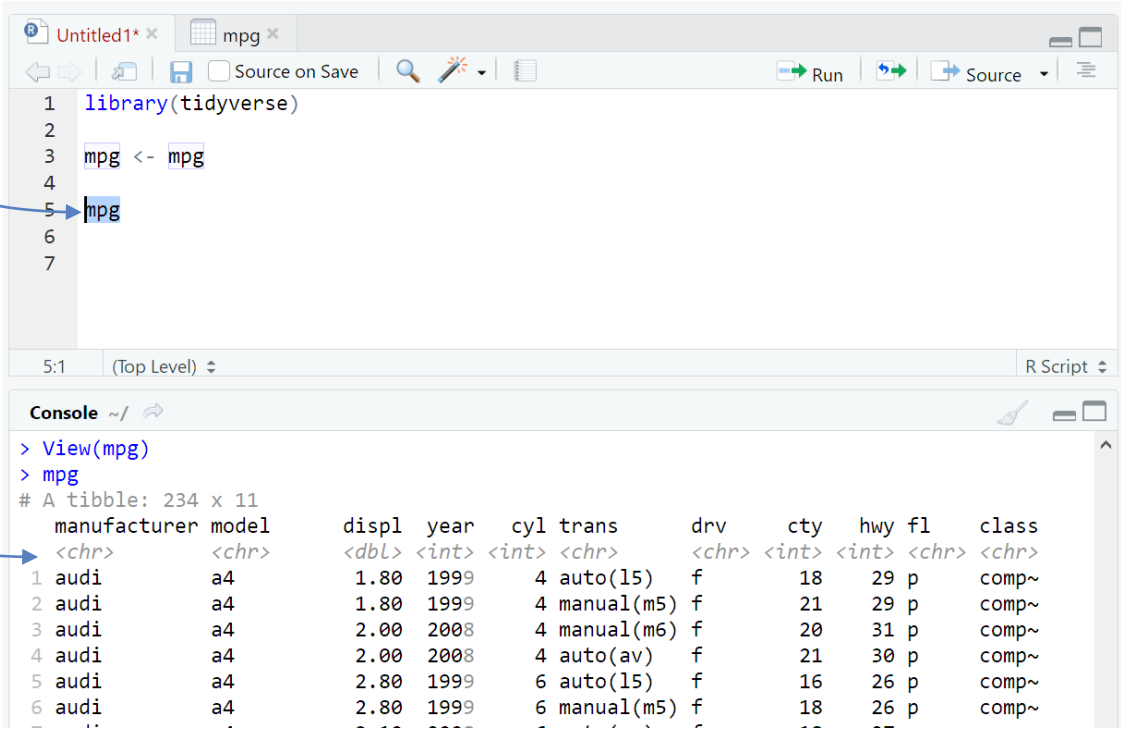
*Very useful with multiple monitors

Option 2: Preview in Console

Type the name of the dataset into editor/console, and run the line.

Run mpg

Prints data frame to console



Q. How many cars?
What variables do we have?

The simple graph has bought more information to the data analyst's mind than any other device

- John Tukey

Graphics to illuminate

Do cars with large engines (displ) use more fuel than cars with small engines?

Note that R is case sensitive

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

Breakdown

1. We begin our plot with ggplot()

2. Inside the ggplot() we name our dataset

3. Next we add layer(s) with +

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

How do we move from data to graphic?

Exercise:
Create a graphic from the data below.

year	time (s)
1930	12.0
1960	11.3
1990	10.5

Exercise:
Create a graphic from the data below. Then, note down all the choices you made.

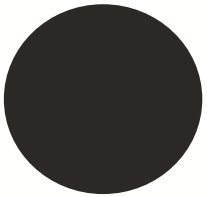
year	time (s)
1930	12.0
1960	11.3
1990	10.5

Choices

1. What shape will represent the data?

Choices

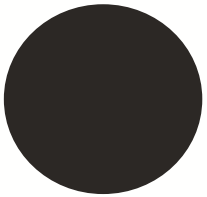
1. What shape will represent the data?



geometric object

Choices

1. What shape will represent the data? (geom)
2. What visual (aesthetic) attributes do we give to the geom?



Choices

1. What shape will represent the data? (geom)
2. What visual (aesthetic) attributes do we give to the geom?

Choices

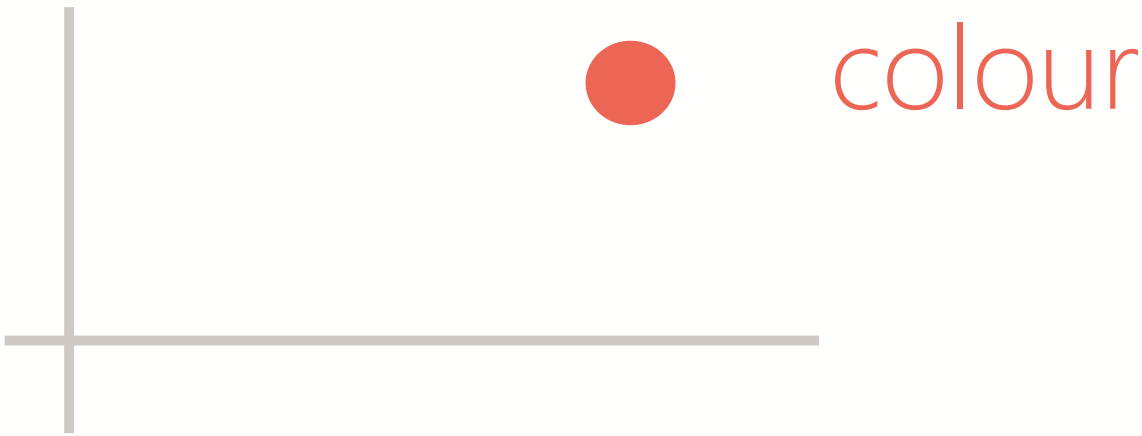
1. What shape will represent the data? (geom)
2. What visual (aesthetic) attributes do we give to the geom?

Choices

1. What shape will represent the data? (geom)
2. What visual (aesthetic) attributes do we give to the geom?

Choices

1. What shape will represent the data? (geom)
2. What visual (aesthetic) attributes do we give to the geom?



A statistical graphic

Maps data variables to geometric objects.

aesthetic
attributes of

A statistical graphic

Maps data variables to geometric objects.

aesthetic
attributes of

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

Here, other aes()
properties: size, colour,
etc. are set by default

A note on functions

`ggplot()`, **`geom_point()`**, and **`aes()`** are functions.

Arguments (inputs) in a function are separated by commas

Here, we provide geom_point() with one argument : aes()

We give aes() two (explicit) arguments

```
ggplot(mpg) +
  geom_point(aes(displ, hwy))
```

Unspecified arguments revert to default values

Shorthand

As ggplot2 knows the order of essential arguments, I will use this convention from now on:

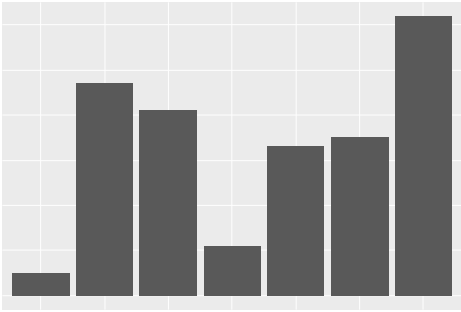
No need for "data ="

```
ggplot(mpg) +
    geom_point(aes(displ, hwy))
```

x goes first, y goes second

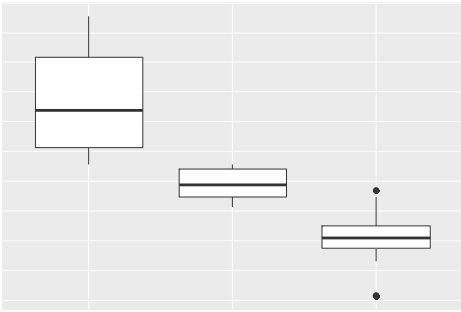
Geoms

We tend to describe plots by the geom used:

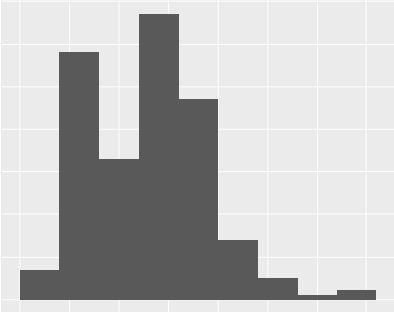


`geom_bar()`

`geom_line()`



`geom_boxplot()`



`geom_histogram()`

Layering geoms

We can display more than one geom in a plot:

```
ggplot(mpg) +
  geom_point(aes(displ, hwy)) +
  geom_line(aes(displ, hwy))
```

Note: `geom_line` *used to illustrate principle only*

Layering geoms

We can display more than one geom in a plot:

```
ggplot(mpg) +
  geom_point(aes(displ, hwy)) +
  geom_line(aes(displ, hwy))
```

duplication!

Note: `geom_line` *used to illustrate principle only*

Layering geoms

To avoid duplication, we can pass the local **`aes()`** to **`ggplot()`**. This will make it a global value:

```
ggplot(mpg, aes(displ, hwy)) +
  geom_point() +
  geom_line()
```

In ggplot aes() goes second

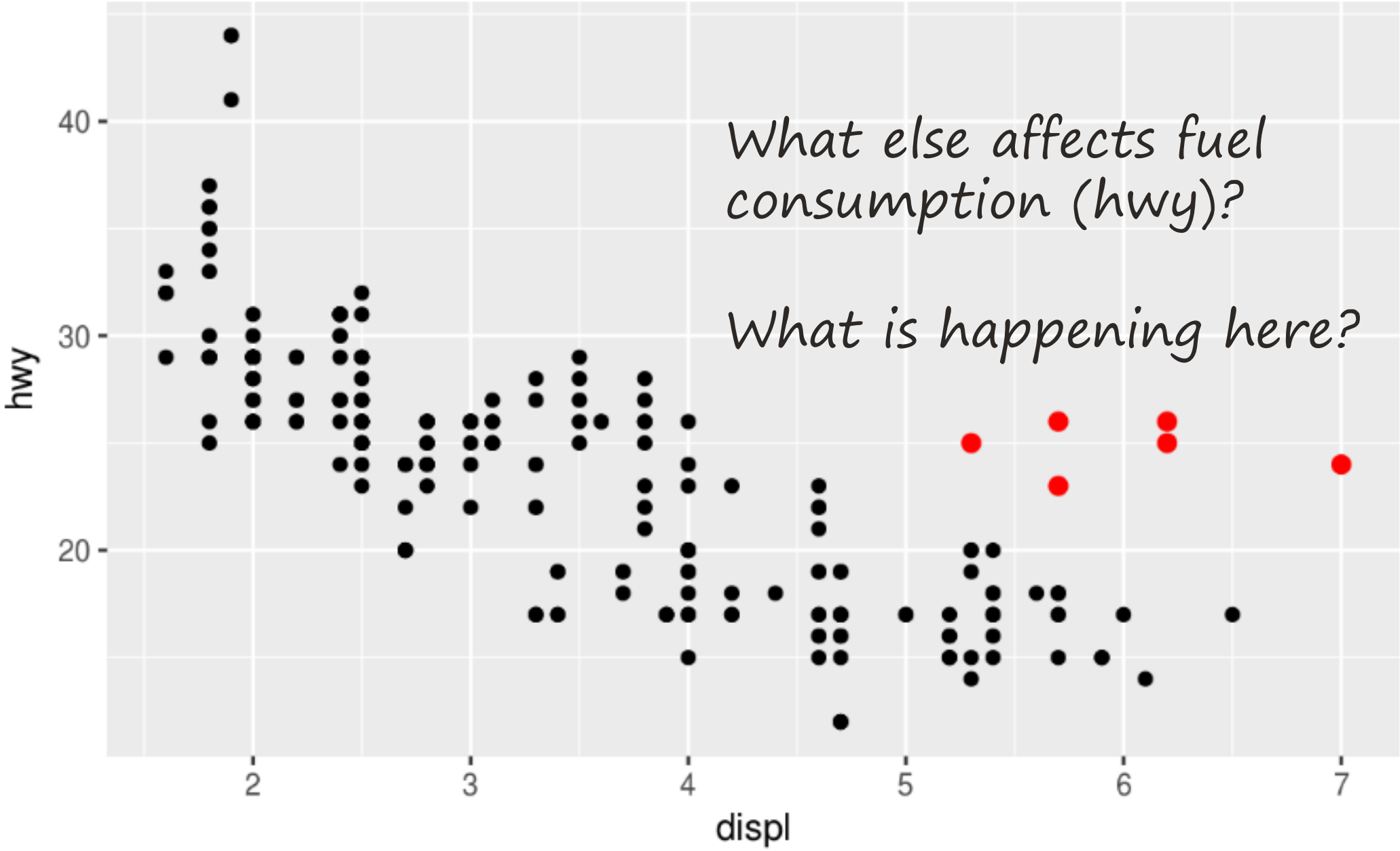
Note: `geom_line` used to illustrate principle only

Your turn

A **`geom_smooth()`** layer can help us identify patterns. Add this geom to our original plot:

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

And (if you like) re-write in shorthand

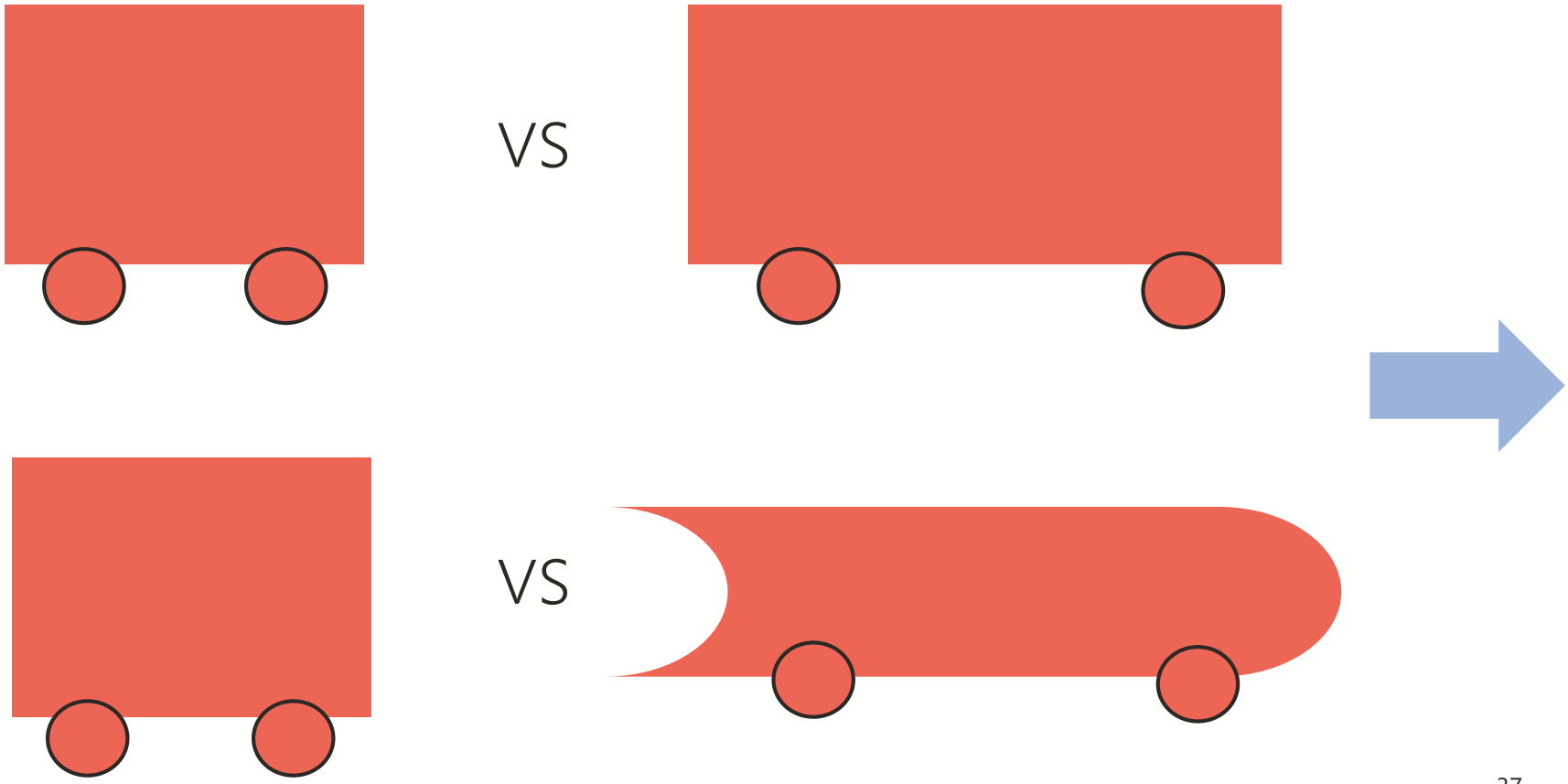


What else affects fuel consumption (hwy)?

What is happening here?

Play your cars right

Same engine, material, speed. Which is more fuel efficient?



Hypothesis

These cars are lighter and/or more aerodynamic.

Are they sports cars?

aesthetic attribute

We can map point colour to the class variable

(so a different colour for each class) to find out.

Adding another variable

Arguments within the aesthetic wrapper describe how variables are mapped:

```
ggplot(mpg) +
 geom_point(aes(displ, hwy, colour = class))
```

We could have chosen size or shape – but less clear

Outcome

The anomalous points are (mostly) two-seater cars.

Likely to be sports cars, therefore more aerodynamic and lighter.

All red

If you wish to apply the same colour to all points, the colour argument goes outside **aes()**:

```
ggplot(mpg) +
  geom_point(aes(displ,  hwy), colour = “red")
```

Small multiples

An alternative way to display additional variables is with small multiples. We do this with **`facet_wrap()`**

```
ggplot(mpg) +
  geom_point(aes(displ, hwy)) +
  facet_wrap(~ class)
```

tilde

Small multiples

An alternative way to display additional variables is with small multiples. We do this with **`facet_wrap()`**

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

facet_wrap is used with categorical variables

Demonstrating geoms:

Histogram:
Q. How are “cty” values distributed?

```
ggplot(mpg, aes(cty)) +
  geom_histogram()
```

Histogram:
Q. How are “cty” values distributed?

```
ggplot(mpg, aes(cty)) +
  geom_histogram(binwidth = 4)
```

Bar plot:
Q. Number of models by manufacturer?

```
ggplot(mpg, aes(manufacturer)) +
  geom_bar()
```

Bar plot:
Q. Number of models by manufacturer?

```
ggplot(mpg, aes(manufacturer)) +
  geom_bar()+
  coord_flip()
```

Bar plot:
Q. Number of models by manufacturer?

```
ggplot(mpg, aes(manufacturer)) +
  geom_bar(stat = “count”)
```

Note: Hidden (default) argument which obviously works if we're only counting one variable, but…

Two variable bar plot:

If two variables for your bar…

```
ggplot(mpg, aes(manufacturer, hwy)) +
  geom_bar(stat = “identity”)
```

… must specify this argument

Note: This plot is to illustrate principle only

Reorder a two variable bar plot:

Name of variable by which to reorder x

```
ggplot(data, aes(reorder(x, a), y)) +
  geom_bar(stat = “identity”)
```

Box plot:
Q. Distribution in each class?

```
ggplot(mpg, aes(class, displ)) +
  geom_boxplot()
```

Violin plot:
Q. Distribution in each class?

```
ggplot(mpg, aes(class, displ)) +
  geom_violin()
```

Plot labels

```
ggplot(mpg, aes(class, displ)) +
  geom_violin()+
  labs(title = "Displacement by class",
       subtitle = “Any subtitle",
       y = "Displacement",
       caption = "Source: US EPA")
```

or: pdf, jpg as you wish

```
ggsave("plot_name.png")
```

By default:

- saves most recent ggplot to your working directory
- saves a plot in the same dimensions as plot window

Tip for now: adjust dimensions of plot pane in RStudio as you wish, then save.

Save your script!

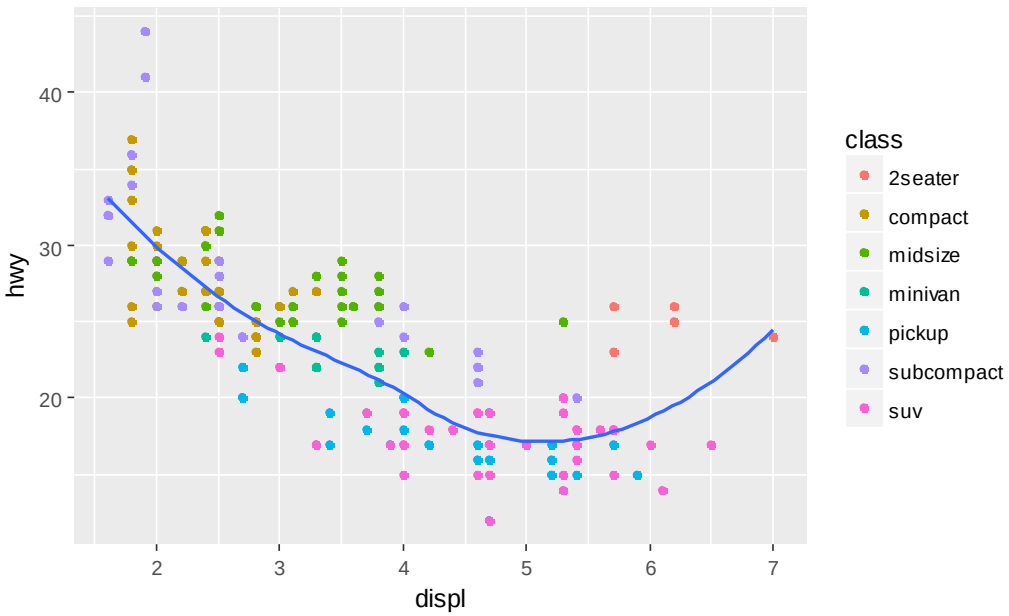
Think of your script as the “real” part of your analysis.

File → Save As… → ggplot_intro.R

Addendum:
Local and global aesthetics

Two layers:

```
ggplot(mpg) +
  geom_point(aes(displ, hwy, colour = class)) +
  geom_smooth(aes(displ, hwy))
```



Duplicate aes attributes:

Ideally, extract similar elements and put them in the global aes()

```
ggplot(mpg  ) +
  geom_point(aes(displ, hwy, colour = class)) +
  geom_smooth(aes(displ, hwy))
```

Global and local

Note that `colour = class` must remain within aes() itself

```
ggplot(mpg, aes(displ, hwy)) +
  geom_point(aes(colour = class)) +
  geom_smooth()
```

accidental aRt

https://twitter.com/accidental_aRt

This work is licensed as

End