This is the presentation title

This is a presentation subtitle

Dr. Calum Webb
Sheffield Methods Institute, the University of Sheffield
c.j.webb@sheffield.ac.uk

This is a panelset: good for learning objectives

What will I learn? How does this week fit into my course?

By the end of this week you will:

- LO1
- LO2
- LO3
- LO4

This is a panelset: good for learning objectives

What will I learn? | How does this week fit into my course?

- Text
- Text
- Text

This is text on a regular slide

This is a smaller heading

This is a smaller heading

This is bold text - do not use on inverse slide

This is regular text

This is a link.

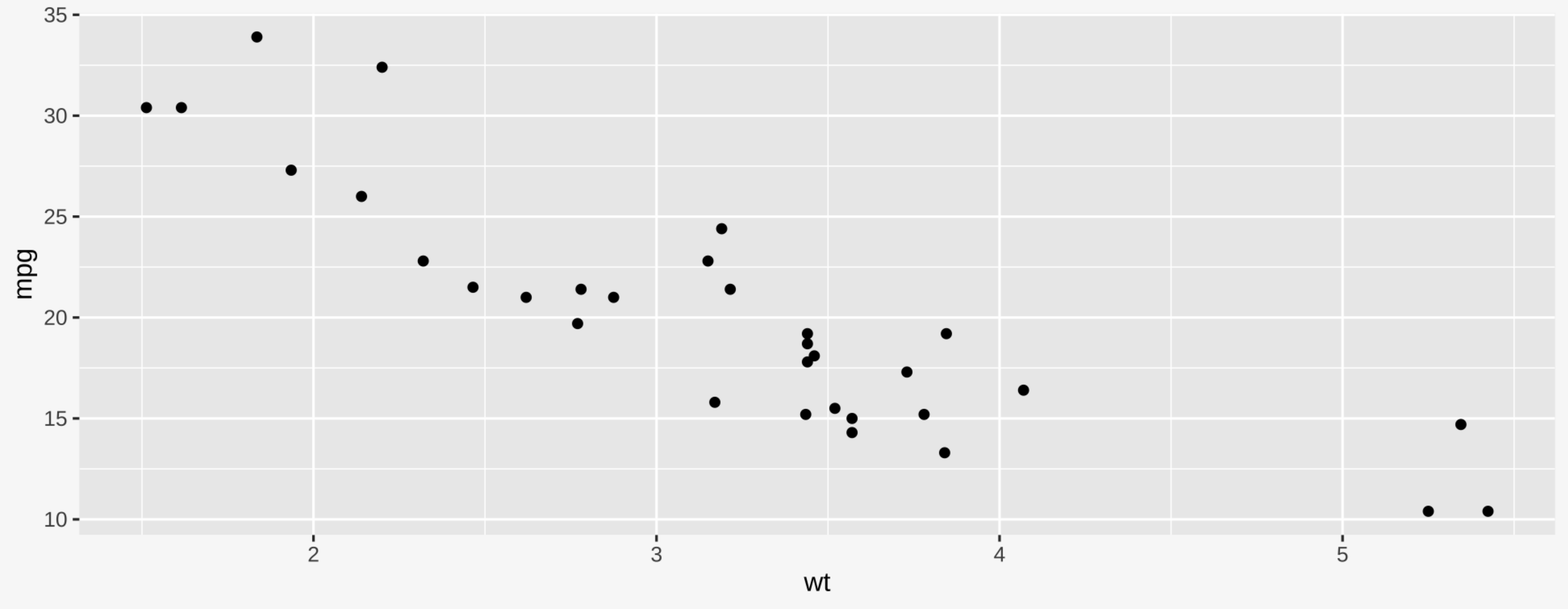
This is an image:

This is an image caption

```
# This is a code snippet
mean(mtcars$mpg)
```

This is text on the inverse slide: useful for big statements, quotes, or section headings

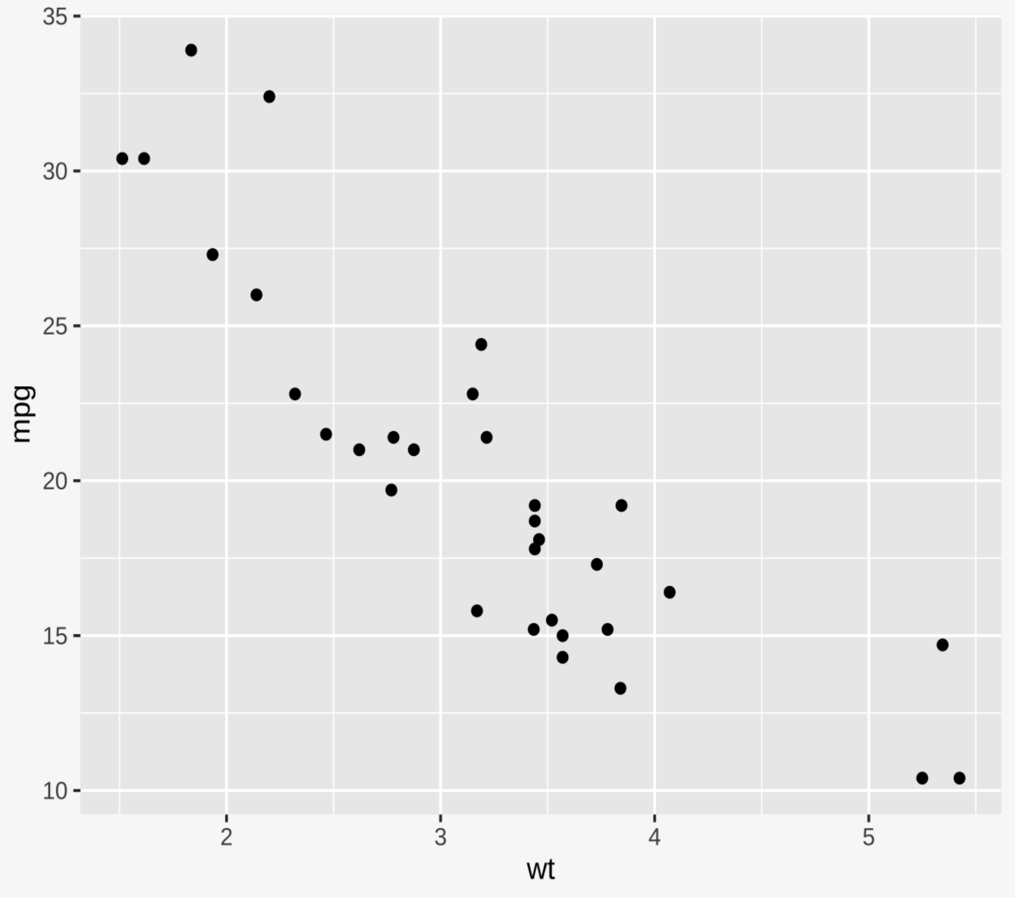
This is an R plot that fills the slide



This is an R plot that fills one half of a slide with code shown.

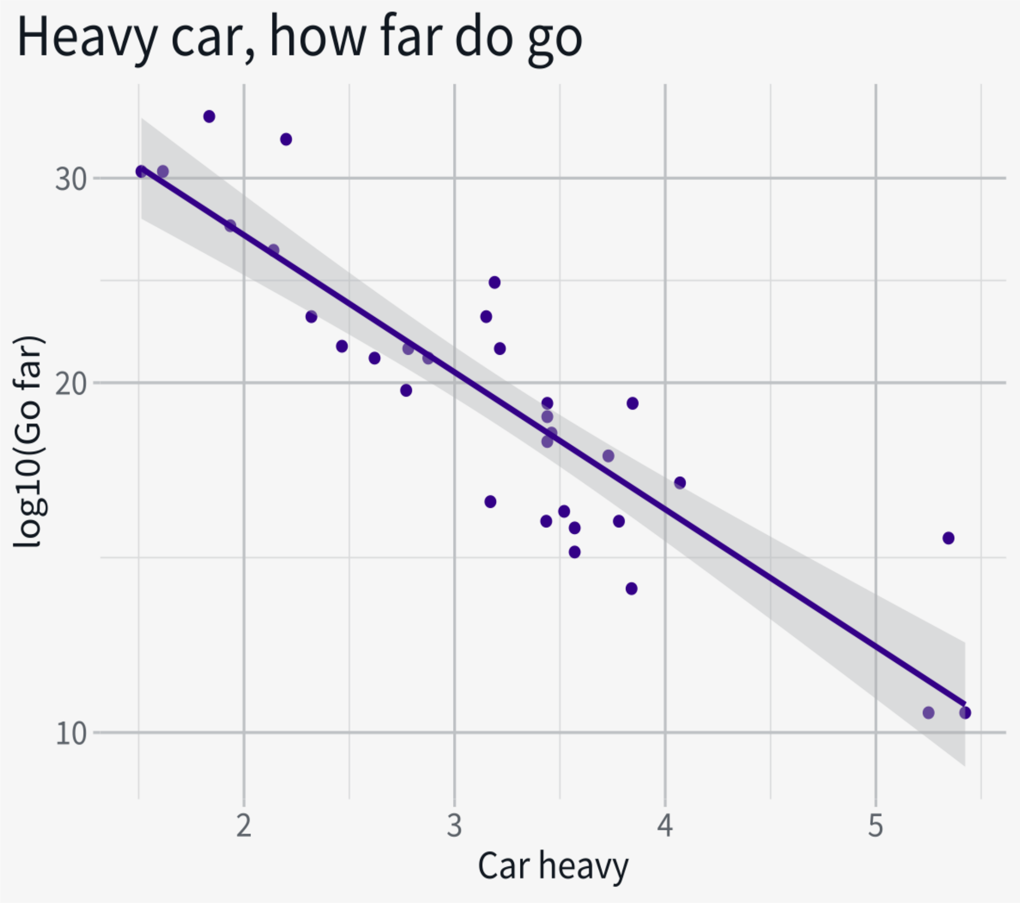
> This is a quote blah blah blah blah blah blah blah blah blah blah blah blah

```r
mtcars %>%
  ggplot() +
  geom_point(aes(x = wt, y = mpg))
```



If I am using a very long bit of code, I will sometimes have the code on the left (but not evaluated using `eval = FALSE` in the chunk header) so students can see it, and then an evaluated piece of code that doesn't echo on the right, e.g.

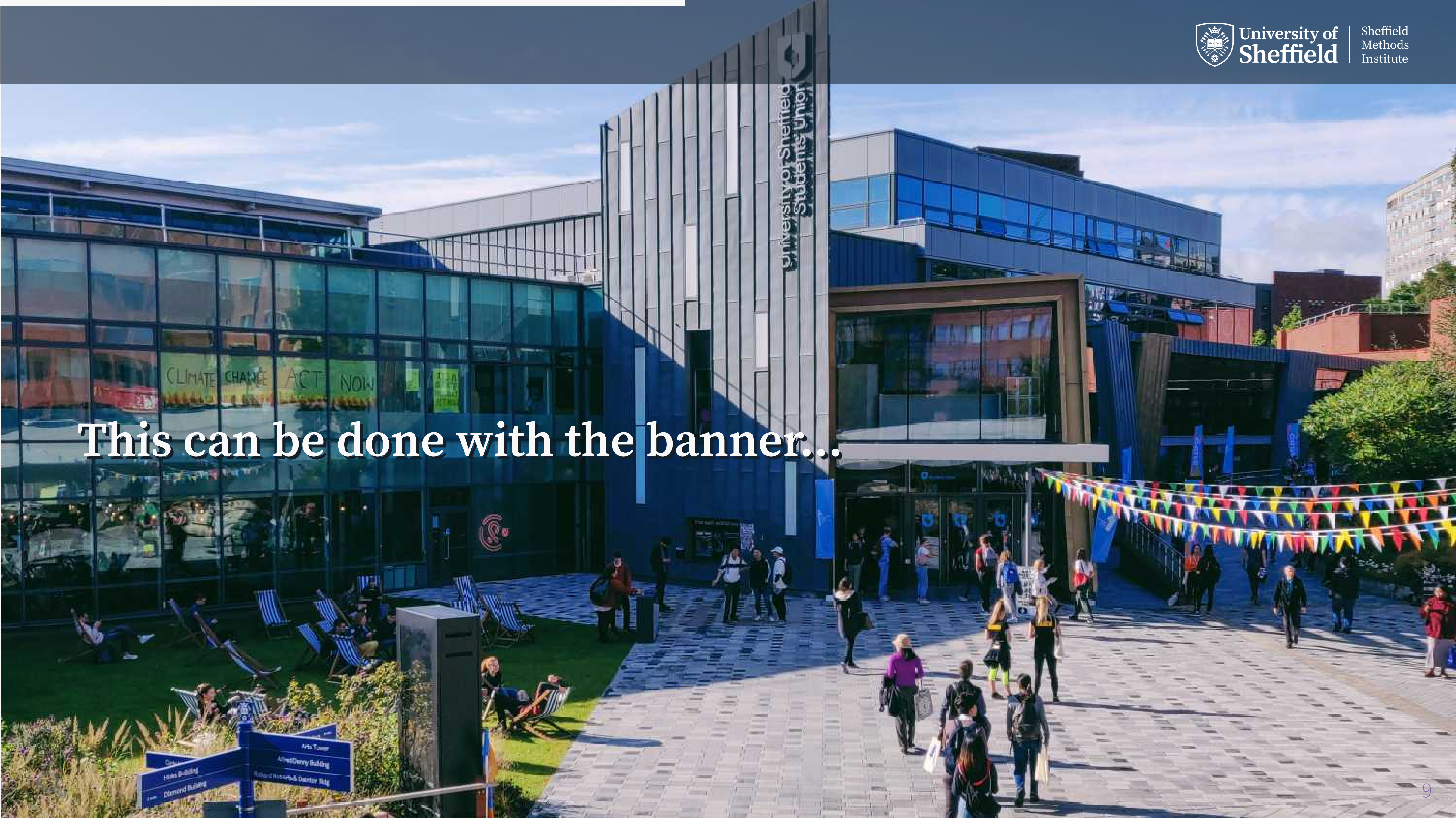
```
#' A lot of code for a plot styled like
#' the slides

mtcars %>%
  ggplot(aes(x = wt, y = mpg)) +
  geom_point(col = "#440099") +
  geom_smooth(method = "lm") +
  scale_y_log10() +
  theme_minimal() +
  ggtitle("Heavy car, how far do go") +
  xlab("Car heavy") +
  ylab("log10(Go far)") +
  theme_xaringan(title_font_size = 18,
                 text_font_size = 16,
                 title_font = google_font("Source Sans Pro"),
                 text_font = google_font("Source Sans Pro")) +
  theme(plot.background = element_rect(fill = "#f8f8f8",
                                       colour = "#f8f8f8"),
        panel.background = element_rect(fill = "#f8f8f8",
                                        colour = "#f8f8f8"))
```



You can split up content with a horizontal rule using the html tag `<hr>`

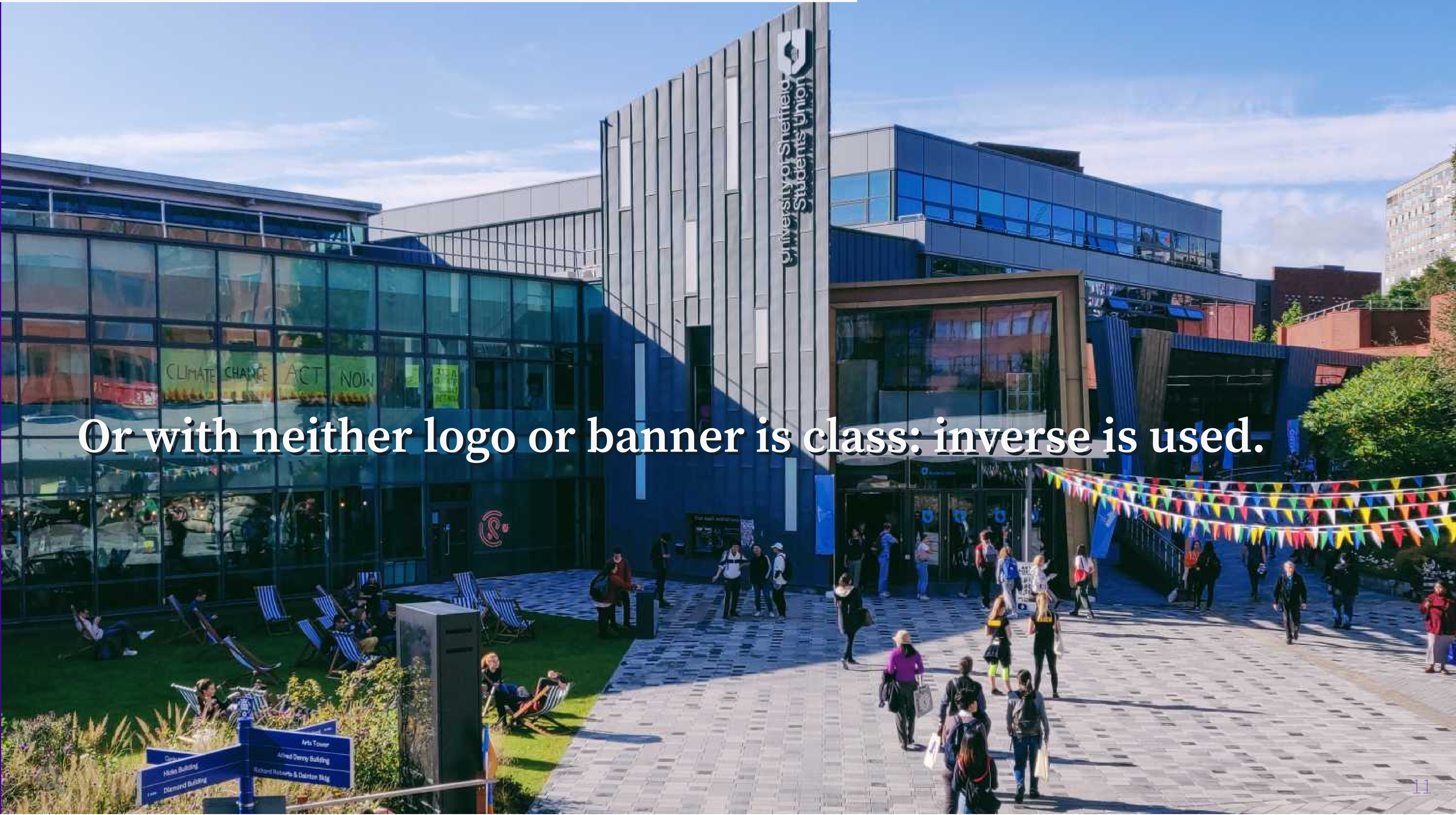
If you need a bit more space for an image, and less space for text, you can use the `pull-left-small[]`, `pull-right-small[]`, `pull-left-big[]`, and `pull-right-big[]` versions of the pull class.

If you want an image to fill an entire slide, you can use the `background-image`, `background-position` and `background-size` options...



This can be done with the banner…

Or without, if class: hide-logo is used...



Or with neither logo or banner is class: inverse is used.

This text is centered using `.center[]`

This text is right-aligned using `.right[]`

- This is a bullet list
- Alignment also works
- for
- images/code output/etc

This text is in italics.

If you want text to be slightly larger on a specific slide, you can use the custom `.big[]` class.

If you want text to be slightly smaller, you can use the custom `.small[]` class

You can also use webicons with the `fontawesome()` function from the `icons` github package, with backticks around them e.g.

🐦 `r icons::fontawesome("twitter")`

🚀 `r icons::fontawesome("rocket")`

`r icons::fontawesome("github")`

⊗ `r icons::fontawesome("times-circle")`

✓ `r icons::fontawesome("check")`

? `r icons::fontawesome("question-circle")`

💬 `r icons::fontawesome("comments")`

☕ `r icons::fontawesome("coffee")`

You can search all available icons here.

$$Pr(\bar{Y} = 1) = \frac{e^{\beta_0+\beta_1 X_1+\beta_2 x_2}}{(1 + e^{\beta_0+\beta_1 X_1+\beta_2 x_2})}$$

This is an equation: wow, it's so beautiful!

Equations are also supported in slides

You can use single $ signs to enclose an inline equation (e.g. X_1), or double dollar signs $$ to create a block equation like on the left.

Something that annoys me...

You might have noticed that these slides irritatingly float left and right blocks to be top-aligned rather than middle aligned when they have different sized content. You can fix this either with repeated breaks spacers `<br>` or you can use the custom `.middle-left[]` or `.middle-right[]` classes I wrote and added to the custom CSS.

P.S. You can add incremental slides like these using `--` between paragraphs, but this doesn't work *within* `.pull-left[]`, `.pull-right[]` split column slides, unfortunately.

$$Pr(Y = 1) = \frac{e^{\beta_0+\beta_1 X_1+\beta_2 x_2}}{(1 + e^{\beta_0+\beta_1 X_1+\beta_2 x_2})}$$

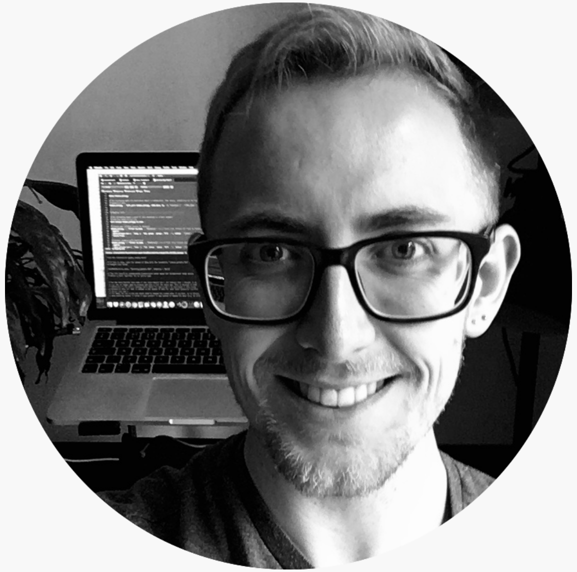
This is an equation: wow, it's even more beautiful!

Equations are also supported in slides

You can use single $ signs to enclose an inline equation (e.g. X_1), or double dollar signs $$ to create a block equation like on the left.

Something that annoys me...

You might have noticed that these slides irritatingly float left and right blocks to be top-aligned rather than middle aligned when they have different sized content. You can fix this either with repeated breaks spacers `<br>` or you can use the custom `.middle-left[]` or `.middle-right[]` classes I wrote and added to the custom CSS.



Dr. Calum Webb

Sheffield Methods Institute
The University of Sheffield
The Wave, 2 Whitham Road
Sheffield
S10 2AH

c.j.webb@sheffield.ac.uk

These custom classes also have equivalent `-big` and `-small` versions, e.g. `.middle-left-small[]`, `.middle-right-big[]`, that can be paired up. Perfect for staff profiles.

PDF versions of slides can be created using the excellent `renderthis` package. You will need to install `chromote` and `pdftools` as well.

These slides were saved as pdf using the following code:

```
renderthis::to_pdf(from = "smi-xaringan-template.html", to = "smi-xaringan-template.pdf", complex_slides = TRUE)
```