

Getting started with Rmarkdown presentations

Brad Duthie

25 JAN 2023

Before starting to code

When and how to start making Rmarkdown slides

- ▶ Start for an informal presentation
 - ▶ Will not need to do anything fancy
 - ▶ Minor errors not a big deal
 - ▶ Can get started quickly
- ▶ Build confidence and learn new style tricks over time
 - ▶ Add cascading style sheets (CSS) for HTML
 - ▶ Integrate LaTeX for PDF slides
- ▶ Avoid anxiety of having presentation limited by technical skill

Four types of outputs using knit

1. HTML (ioslides): Not very elegant, but can use columns and shiny easily
2. HTML (slidy): Nicer looking, but columns are tricky and not for shiny
3. PDF (LaTeX): Nice looking, but no shiny and might need to know LaTeX
4. PPTX (Powerpoint): Probably looks okay for some things, but never works for me in LibreOffice

Need to install rmarkdown and knitr packages

The long-term goal is to do what you cannot (easily) in PowerPoint

- ▶ Quickly and easily produce a consistent slide layout
- ▶ Integrate R analysis and plots **directly** into a presentation
- ▶ Make interactive slides using Rshiny
- ▶ Use version control when writing and maintaining slides
- ▶ Ultimately produce slides more quickly through text and reuse of old code

Some tricks to get started with the code

Select a new Rmarkdown presentation

- ▶ The output format is completely unimportant
- ▶ You can get rid of everything but the first lines

```
---  
title: "Untitled"  
author: "Brad Duthie"  
date: "25/01/2023"  
output: ioslides_presentation  
---
```

- ▶ The YAML specifies some meta-data
 - ▶ 'Yet Another Markup Language'
 - ▶ Can add to it (e.g., reference styles, format options, etc.)
 - ▶ Everything below the YAML is meant to introduce you to Rmarkdown slides

Getting started is really easy

The syntax can be used for any type of slide (HTML, PDF, PPTX)

```
# This makes a title slide
```

```
## This is a normal slide
```

- Bullet 1 on normal slide
- Bullet 2 on normal slide

```
## Links are easy
```

- Link to [UoS website] (<https://www.stir.ac.uk/>)

Simple text slides are especially easy to write quickly

Adding images to an Rmarkdown slide

Images can be added with a single line of code:

```
![Optional figure legend](logo.png){width=20%}
```



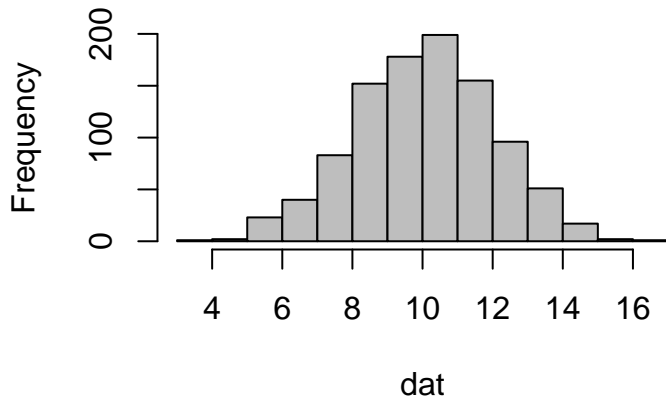
Figure 1: Optional figure legend

Could have also used the below

```
include_graphics("logo.png",  
                fig.cap = "Optional figure legend");
```

Running code in an Rmarkdown slide

```
# The code below is actually being run  
dat <- rnorm(n = 1000, mean = 10, sd = 2);  
  
# We can plot a histogram of it below  
hist(dat, main = "", col = "grey");
```



Separating a slide into columns gets tricky

Dividing an entire slide into two columns can be done using a bit of code at the end of the title.

```
## Slide title {.columns-2}
```

This applies to the whole slide though, even if you only want columns for a portion of the slide.



It also only works for ioslides presentations.

Separating a slide into columns gets tricky

- ▶ Work-arounds to force breaks between columns and make everything look better
- ▶ Figuring out the columns even just for ioslides is not much fun
- ▶ At this point you might need to settle on an output type (ioslides, slidy, PDF)



Separating a slide into columns gets tricky

Using LaTeX can be powerful inside Rmarkdown, but takes time to learn, and only works for PDFs.

```
\begin{columns}
```

```
\begin{column}{0.5\textwidth}
```

```
''',{r, out.width = "100%", echo = FALSE}
```

```
library("knitr")
```

```
include_graphics("logo.png");
```

```
''',
```

```
\end{column}
```

```
\begin{column}{0.5\textwidth}
```

```
\begin{itemize}
```

```
\setlength\itemsep{1.0em}
```

```
\item Using LaTeX can make things look nice
```

```
\item Only possible for PDFs
```

```
\item Need to know yet more code
```

```
\end{itemize}
```

Separating a slide into columns gets tricky



- ▶ Using LaTeX can make things look nice
- ▶ Only possible for PDFs
- ▶ Need to know yet more code

Maths can be expressed especially well in PDF

$$e^{i\pi} + 1 = 0$$

$$e^{i\pi} + 1 = 0$$

$$P(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x - \mu}{\sigma}\right)^2}$$

$$P(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x - \mu}{\sigma}\right)^2}$$

Maths can be expressed especially well in PDF

$$E[X] = \int x f(x) dx$$

$$E[X] = \int x f(x) dx$$

Equation editors can convert to LaTeX

You will probably have to make compromises

If all else fails, you can just make a picture of what you want and insert it as an image

This is a new slide

- ▶ Bullet point 1
- ▶ Bullet point 2