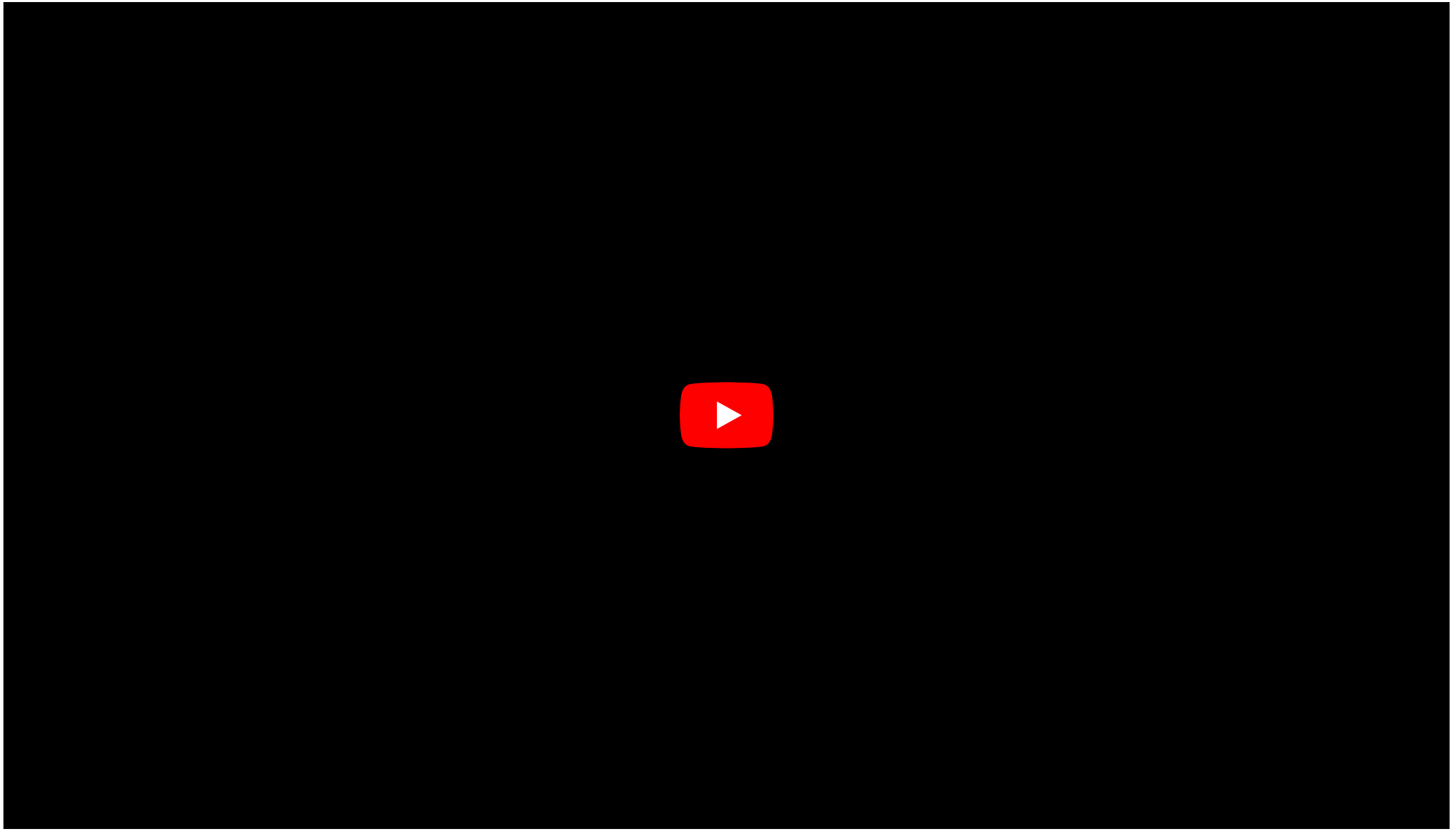
Academic Writing with R Markdown

Hello, R Markdown!

ILD Webinar Series, UiTM Kedah
27 July 2023

Kamarul Ariffin Mansor
UiTM Expert Profile Link

Department of Mathematical Sciences
UiTM Kedah



Plan

Module 1: Fundamentals

- Why R Markdown?
- Basic syntax: Markdown & code
- Final pieces: Citations & cross-referencing

Module 2: Applications

- Customising your output
- Workflows, collaboration & teaching
- Putting it all together

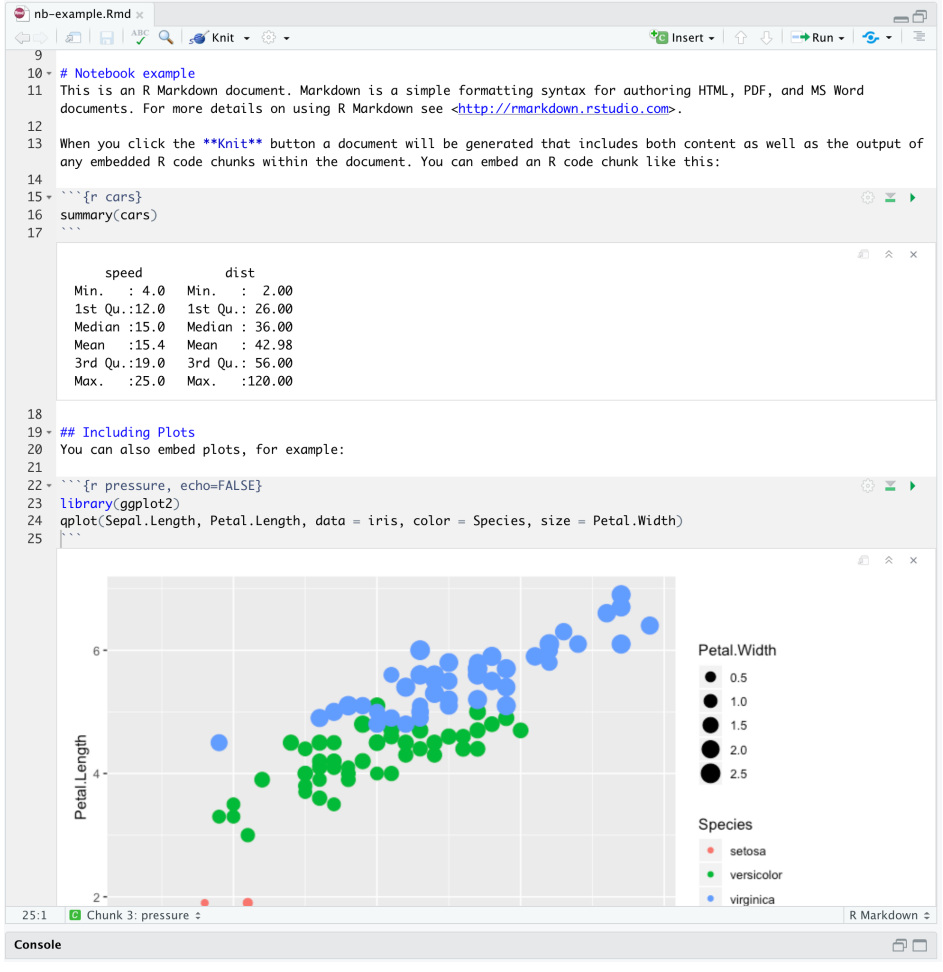
Module 1: Fundamentals

Wouldn't it be great if...

- You could have code, results, and text in the same document?
- Your results and plots were automatically generated from your data, so your documents were updated if your data changed?
- The file format of your documents was future-proof?
- The syntax for this was easy?

Wouldn't it be great if...

- You could have code, results, and text in the same document?
- Your results and plots were automatically generated from your data, so your documents were updated if your data changed?
- The file format of your documents was future-proof?
- The syntax for this was easy?



Interactive notebook

Wouldn't it be great if...

- You could have code, results, and text in the same document?
- Your results and plots were automatically generated from your data, so your documents were updated if your data changed?
- The file format of your documents was future-proof?
- The syntax for this was easy?

````
---
title: "Notebook example"
output: html_document
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```

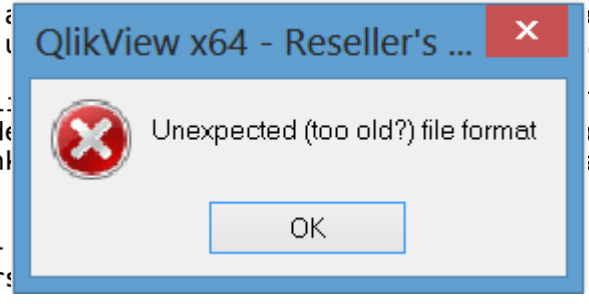
# Notebook example
This is an R Markdown document. Markdown is a simple formatting
syntax for a                                                    ents. For more
details on u                                                    .rstudio.com>.

When you cli                                                    l be generated
that include                                                    of any embedded
R code chunk                                                    an R code chunk
like this:

```{r cars}
summary(cars
```

## Including Plots
You can also embed plots, for example:

```{r pressure, echo=FALSE}
library(ggplot2)
qplot(Sepal.Length, Petal.Length, data = iris, color = Species,
size = Petal.Width)
```
````



Plain text

Wouldn't it be great if...

- You could have code, results, and text in the same document?
- Your results and plots were automatically generated from your data, so your documents were updated if your data changed?
- The file format of your documents was future-proof?
- The syntax for this was easy?

```
\setlength{\droptitle}{-2em}

  \title{Notebook example}
    \pretitle{\vspace{\droptitle}\centering\huge}
  \posttitle{\par}
    \author{}
    \preauthor{}\postauthor{}
    \date{}
    \predate{}\postdate{}


\begin{document}
\maketitle

\hypertarget{notebook-example}{%
\section{Notebook example}\label{notebook-example}}

This is an R Markdown document. Markdown is a simple formatting syntax
for authoring HTML, PDF, and MS Word documents. For more details on
using R Markdown see \url{http://rmarkdown.rstudio.com}.

When you click the \textbf{Knit} button a document will be generated
that includes both content as well as the output of any embedded R code
chunks within the document. You can embed an R code chunk like this:

\begin{Shaded}
\begin{Highlighting}[]
\KeywordTok{summary}\NormalTok{(cars)}
\end{Highlighting}
\end{Shaded}

\begin{verbatim}
##      speed           dist
##  Min.   : 4.0   Min.   :  2.00
##  1st Qu.:12.0   1st Qu.: 26.00
##  Median :15.0   Median : 36.00
```

Wouldn't it be great if...

- You could have code, results, and text in the same document?
- Your results and plots were automatically generated from your data, so your documents were updated if your data changed?
- The file format of your documents was future-proof?
- The syntax for this was easy?

R Markdown is an easy-to-write plain text format for creating dynamic documents and reports. See Using R Markdown to learn more.

Emphasis

```
*italic*   **bold**

_italic_   __bold__
```

Headers

```
# Header 1

## Header 2

### Header 3
```

Lists

Unordered List

```
* Item 1
* Item 2
    + Item 2a
    + Item 2b
```

Markdown

Anatomy of an R Markdown Document

R Markdown file = plain text file with extension *.Rmd*

````
---
title: "Diamond sizes"
date: 2016-08-25
output: html_document
---

```{r setup, include=FALSE}
library(ggplot2)
library(dplyr)

smaller <- diamonds %>%
  filter(carat <= 2.5)
```

# Shine bright like a diamond
We have data about `r nrow(diamonds)` diamonds.
Only `r nrow(diamonds) - nrow(smaller)` are larger than 2.5 carats.
The distribution of the remainder is shown below:

```{r}
smaller %>%
  ggplot(aes(carat)) +
  geom_freqpoly(binwidth = 0.01)

```
````

YAML header ("YAML Ain't Markup Language")

````
---
title: "Diamond sizes"
date: 2016-08-25
output: html_document
---

```{r setup, include=FALSE}
library(ggplot2)
library(dplyr)

smaller <- diamonds %>%
  filter(carat <= 2.5)
```

# Shine bright like a diamond
We have data about `r nrow(diamonds)` diamonds.
Only `r nrow(diamonds) - nrow(smaller)` are larger than 2.5 carats.
The distribution of the remainder is shown below:

```{r}
smaller %>%
  ggplot(aes(carat)) +
  geom_freqpoly(binwidth = 0.01)

```
````

Text

````
---
title: "Diamond sizes"
date: 2016-08-25
output: html_document
---

```{r setup, include=FALSE}
library(ggplot2)
library(dplyr)

smaller <- diamonds %>%
  filter(carat <= 2.5)
```

# Shine bright like a diamond
We have data about `r nrow(diamonds)` diamonds.
Only `r nrow(diamonds) - nrow(smaller)` are larger than 2.5 carats.
The distribution of the remainder is shown below:

```{r}
smaller %>%
  ggplot(aes(carat)) +
  geom_freqpoly(binwidth = 0.01)

```
````

Code

````
---
title: "Diamond sizes"
date: 2016-08-25
output: html_document
---

```{r setup, include=FALSE}
library(ggplot2)
library(dplyr)

smaller <- diamonds %>%
  filter(carat <= 2.5)
```

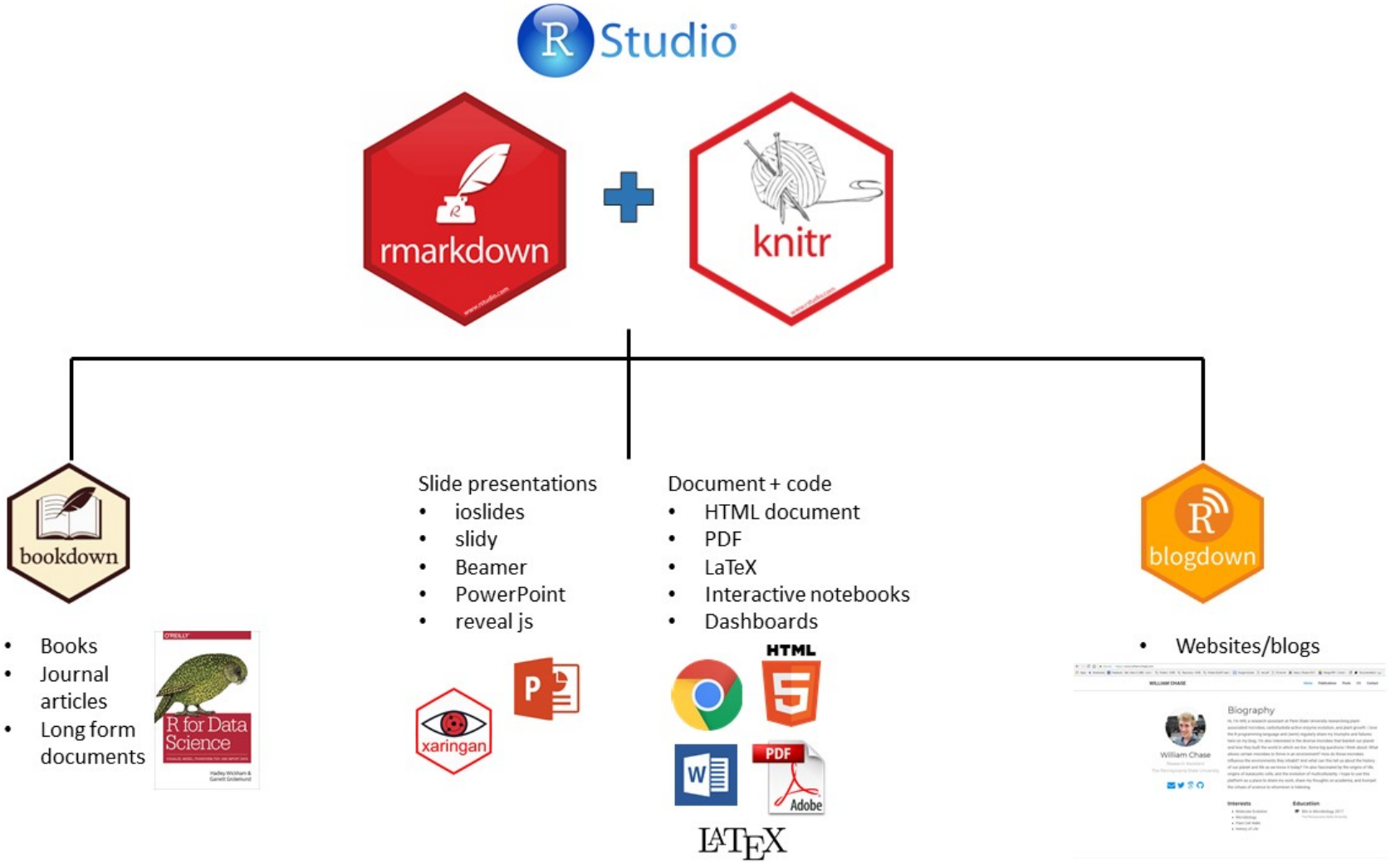
# Shine bright like a diamond
We have data about `r nrow(diamonds)` diamonds.
Only `r nrow(diamonds) - nrow(smaller)` are larger than 2.5 carats.
The distribution of the remainder is shown below:

```{r}
smaller %>%
  ggplot(aes(carat)) +
  geom_freqpoly(binwidth = 0.01)

```
````

What Can It Do?
Output formats

Document your analyses, make a **website**, write your **paper**, make **slides**... the world is your oyster!




Rmd
knitr
md
pandoc

Basic syntax

This...

italics and **bold**

`inline code`

sub~2~/superscript^2^

~~strikethrough~~

escaped: * _ \\

endash: --, emdash: ---

> blockquote

Header 1

Header 2

Line break: End line with 2+ spaces, or backslash:
Roses are red
Violets are blue

Roses are red \
Violets are blue

turns into this...

italics and **bold**

`inline code`

sub$_2$/superscript^2

~~strikethrough~~

escaped: * _ \

endash: –, emdash: —

> blockquote

Header 1

Header 2

Roses are red Violets are blue

Roses are red
Violets are blue

This...

```
- unordered list
   - sub-item
   - sub-item 2
     - sub-sub-item

1. ordered list
2. item 2
   - sub-item 1
   - sub-item 2


inline-math: $A = \pi*r^{2}$

math-block: $$A = \pi*r^{2}$$

[text for hyperlink](https://www.google.com)

A footnote [^1]

[^1]: here is the footnote text.

<!-- this is a comment that won't be shown -->
```

turns into this...

- unordered list
  - sub-item
  - sub-item 2
    - sub-sub-item

1. ordered list

2. item 2
   i. sub-item 1
   ii. sub-item 2

   inline-math: $A = \pi * r^2$

   math-block:

$$A = \pi * r^2$$

   text for hyperlink

   A footnote[1]

[1] Here is the footnote text.

Time for practice!

Everybody should already have on their laptops...

- R and RStudio + R packages `bookdown` and `tidyverse` (`install.packages("package-name")`)
- a LaTeX installation, for knitting to PDF (`tinytex::install_tinytex()`)

Alternatively, we can use the cloud version at Posit Cloud

Beginner

Create a new R Markdown file
(File > New File > R Markdown...)

Knit to HTML, PDF, Word

Tweak the content

- add your name and today's date to YAML header
- add a paragraph, containing a header, **bold**, and *italics*
- knit to output of your choice
- what creates linebreaks and new paragraphs?

Intermediate

In a new R Markdown file, add a paragraph that contains

- a hyperlink
- a blockquote
- a comment
- some math

Skim through the references available from within RStudio:

- Help > Cheatsheets > R Markdown Cheat Sheet
- Help > Cheatsheet > R Markdown Reference Guide

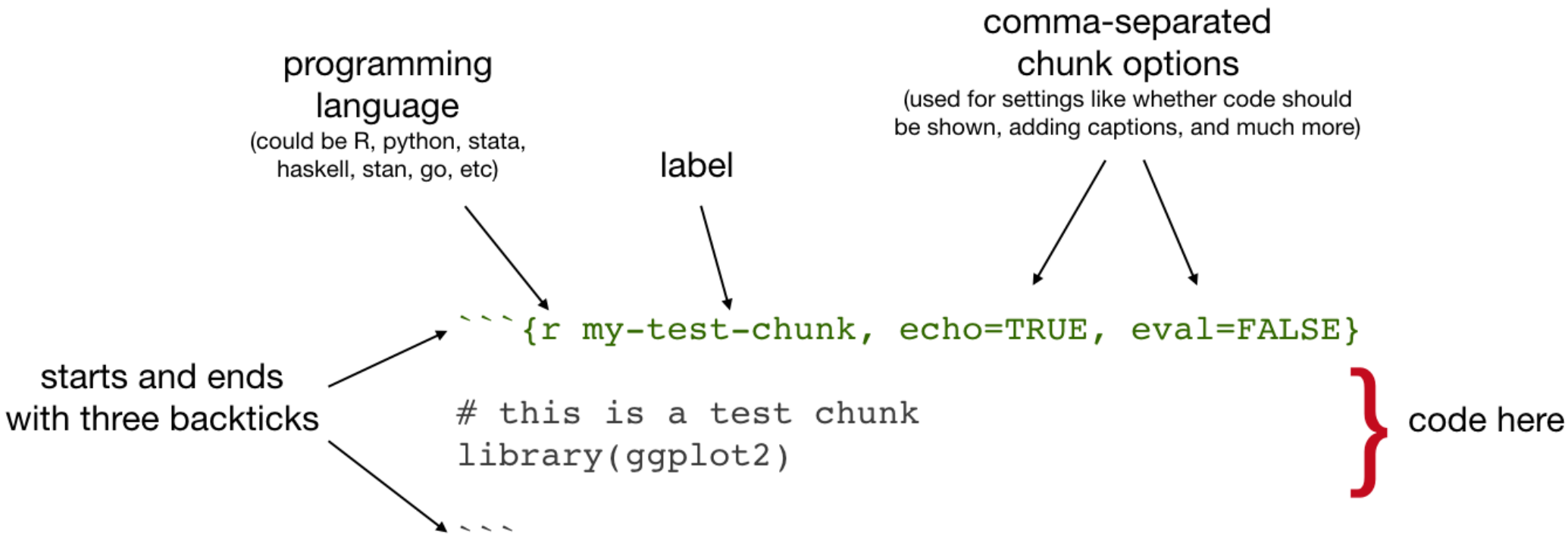
Advanced

- When making footnotes, what are the two ways to create the actual footnote text?
- How do you get literal backticks (`) in your output?
- What's the difference between outputting to `rmarkdown::pdf_document` and `bookdown::pdf_document2`?


NOT SURE IF I'M A GOOD PROGRAMMER
OR JUST GOOD AT GOOGLING
VIA 9GAG.COM
MEMEFUL.COM

Code

Code chunks



Some common chunk options (see e.g. bookdown.org)

- `echo`: whether or not to display code in knitted output
- `eval`: whether or to to run the code in the chunk when knitting
- `include`: wheter to include anything from the from a code chunk in the output document
- `fig.cap`: figure caption

Typical chunks

Setup chunk

````
```{r setup, include=FALSE}
# don't show code unless we explicitly set echo = TRUE
knitr::opts_chunk$set(echo = FALSE)

library(tidyverse)
```
````

- Normally, an R Markdown document starts with a chunk that's used to set some options and load required libraries.
- `knitr::opts_chunk$set` sets default options for all chunks.

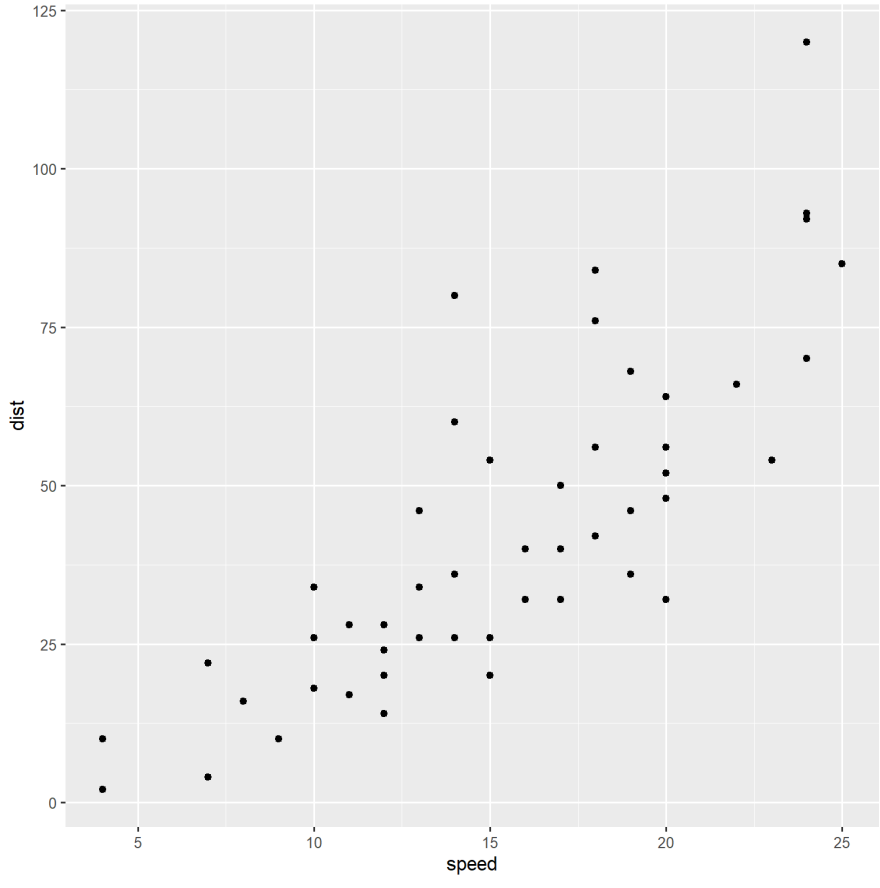
Including images

```
```{r, fig.cap="R Markdown logo"}
knitr::include_graphics("figures/markdown.png")
```
```

R Markdown logo

Including plots

```
```{r, fig.cap = "A ggplot of car stuff"}
cars %>%
  ggplot() +
    aes(x = speed, y = dist) +
    geom_point()
```
```



A ggplot of car stuff

Including tables

```
```{r}
cars %>%
  head() %>%
  knitr::kable(caption = "A knitr kable table")
```
```

A knitr kable table

speed	dist
4	2
4	10
7	4
7	22
8	16

- Gotcha: when using `kable`, captions are set inside the `kable` function
- The `kable` package is often used with the `kableExtra` package
- A number of other packages are available for making pretty tables, see rmarkdown.rstudio.com

Inline code

Inside your text you can include code with the syntax `` `r code here` ``.

For example, `` `r 4 + 4` `` would output 8 in your text.

```
print(diamonds, n = 5)
```

```
## # A tibble: 53,940 x 10
##   carat cut     color clarity depth table price     x     y     z
##   <dbl> <ord>   <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>
## 1  0.23 Ideal   E     SI2      61.5    55   326  3.95  3.98  2.43
## 2  0.21 Premium E     SI1      59.8    61   326  3.89  3.84  2.31
## 3  0.23 Good    E     VS1      56.9    65   327  4.05  4.07  2.31
## 4  0.29 Premium I     VS2      62.4    58   334  4.2   4.23  2.63
## 5  0.31 Good    J     SI2      63.3    58   335  4.34  4.35  2.75
## # ... with 53,935 more rows
```

```
num_diamonds <- nrow(diamonds)
```

There are `` `r num_diamonds` `` rows in the `diamonds` dataset.

There are 53940 rows in the `diamonds` dataset.

Inline code with python

At the moment, syntax like `python code here` is not valid.

However, you can use the `reticulate` package to access variables from python chunks.

````
```{python}
my_number = 4 + 8

print(my_number)

```
````

```
## 12
```

````
```{r}
library(reticulate)

py$my_number

```
````

```
## [1] 12
```

Inline you can then refer to this python variable with `r py$my_number` - `my_number` is 12.

Time for practice!

Beginner

In a new R Markdown file, use code chunks to

1. include an image with `knitr::include_graphics`
2. include a plot (e.g. `plot(pressure)`)
3. include a table (using `kable`)
4. report a calculation inline

Intermediate

1. Try resizing plots with `out.width` and `fig.width` - what's the difference?
2. How do you set `knitr`'s global options to hide code by default?
3. What other options are available to control if a code block is executed and what results are inserted in the finished report? (Hint: see R for Data Science, 27.4.2)

Advanced

1. What's the use of `cache = TRUE`? How does it relate to the `dependson` chunk option?
2. How might you create new chunk options, if the ones provided by `knitr` are not sufficient?

Final pieces

Citations

1. Put references in a plain text file with the extension **.bib**, in **BibTex** format (most reference managers can do this - Zotero works best).[1]
   In the highlighed section, 'Shea2014' is the **citation identifier**.

```
@article{Shea2014,
  author =        {Shea, Nicholas and Boldt, Annika},
  journal =       {Trends in Cognitive Sciences},
  pages =         {186--193},
  title =         {{Supra-personal cognitive control}},
  volume =        {18},
  year =          {2014},
  doi =           {10.1016/j.tics.2014.01.006},
}
```

2. Reference this file in your YAML header

```
---
title: "Citation test"
bibliography: example.bib
output: html_document
---
```

[1] The bibliography can be in other formats as well, including EndNote (**.enl**) and RIS (**.ris**), see rmarkdown.rstudio.com/authoring_bibliographies_and_citations

Citations

3. In your text, citations go inside brackets and separated by semicolons.
   By default the Chicago author-date format is used in the output:

This...

Blah blah [@Shea2014; @Lottridge2012].

turns into this...

Blah blah (Shea et al. 2014; Lottridge et al. 2012).

Citations

3. In your text, citations go inside brackets and separated by semicolons.
   By default the Chicago author-date format is used in the output:

This...

Blah blah [@Shea2014; @Lottridge2012].

Shea et al. says blah [-@Shea2014].

@Shea2014 says blah.

Blah blah [see @Shea2014, pp. 33-35; also @Wu2016, ch. 1].

turns into this...

Blah blah (Shea et al. 2014; Lottridge et al. 2012).

Shea et al. says blah (2014).

Shea et al. (2014) says blah.

Blah blah (see Shea et al. 2014, 33–35; also Wu 2016, ch. 1).

Citations

3. In your text, citations go inside brackets and separated by semicolons.
   By default the Chicago author-date format is used in the output:

This...

Blah blah [@Shea2014; @Lottridge2012].

Shea et al. says blah [-@Shea2014].

@Shea2014 says blah.

Blah blah [see @Shea2014, pp. 33-35; also @Wu2016, ch. 1].

turns into this...

Blah blah (Shea et al. 2014; Lottridge et al. 2012).

Shea et al. says blah (2014).

Shea et al. (2014) says blah.

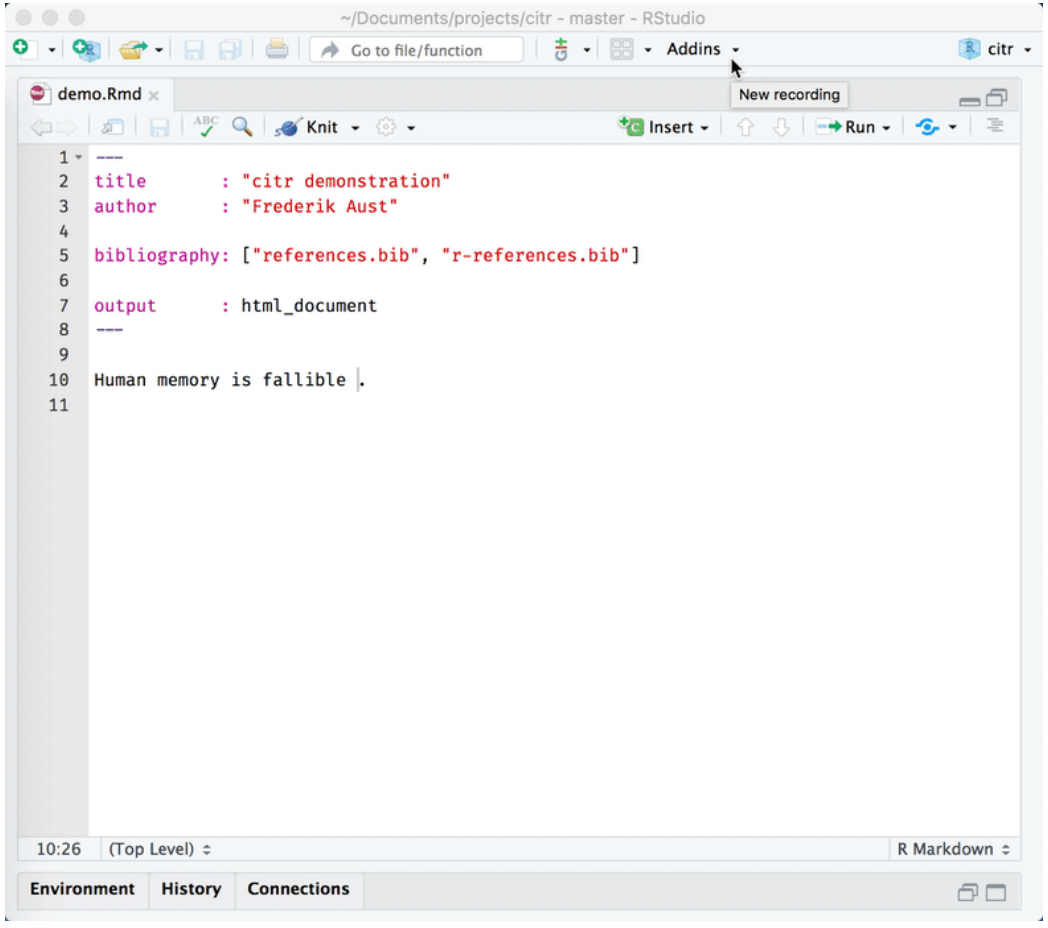
Blah blah (see Shea et al. 2014, 33–35; also Wu 2016, ch. 1).

You can add e.g `csl: my-style.csl` in the YAML header to change to other formats - browse through and download styles at zotero.org/styles

Citations

For an easy way to insert citations, try the `citr` RStudio add-in.

```
install.packages("citr")
```



Cross-referencing

- The `bookdown` package adds capability to do cross-referencing. We can refer to **sections** within our document, to **figures** and **tables**, and even pieces of **text**.
- To enable this ability, set output to e.g. `bookdown::html_document2`, `bookdown::pdf_document2`, `bookdown::word_document2`, etc. (see bookdown.org)

```
---
title: "Cross-referencing test"
output: bookdown::html_document2
---
```

Section references

- Headers are automatically assigned a reference label, which is the text in lower caps separated by dashes
  - For example, the label for `# My header` is `my-header`
- The cross-reference syntax is `\@ref(label)`
  - So `# My header` can be referenced with `\@ref(my-header)`
  - .Rmd: `See section \@ref(my-header)` -> output: 'See section 1'

Section references

- You can also use hyperlink syntax and add # before the label
  - .Rmd: `[see this section](#my-header)` -> output: 'see this section'
- Create custom labels by adding `{#label}` after a header, e.g. `# My section {#my-label}`.

Examples here: Rmd file and HTML output.

Figure and table references

- **GOTCHA**: Figures and tables must have captions if you wish to cross-reference them.

```
```{r captain, fig.cap="A marvelous idea"}
knitr::include_graphics("figures/captain.jpeg")
```
```

A marvelous idea

- To refer to figures (plots and images) use the syntax `\@ref(fig:label)`
- So we'd refer to this image with `\@ref(fig:captain)`[1]
- Again, examples here: Rmd, HTML

[1] Recall that the first chunk option after the language is the label (we could also be explicit with `label=captain`).

Figure and table references

````
```{r cars}
knitr::kable(cars[1:5,],
             caption="Stopping cars")
```
````

Table: Stopping cars

speed	dist
4	2
4	10
7	4
7	22
8	16

- To refer to tables use the syntax `\@ref(tab:label)`
- So we'd refer to this table with `\@ref(tab:cars)`
- ...examples here: Rmd, HTML

Time for practice!

Beginner

1. Install the `citr` RStudio add-in
2. Open **examples/citation_test.Rm** and add a new paragraph in which you use `citr` to add two citations - what happens when you knit?
3. Make cross-references to a figure and a table

Intermediate

1. Open **examples/citation_test.Rm** and change the citation style with `csl: citation_styles/apa.csl`. What happens when you knit?
2. Have a look through zotero.org/styles; download another styles and test it.
3. Add a custom label to a header and reference it

Advanced

1. How do you stop pandoc from automatically inserting cited references by the end of the document?
2. How would you handle a situation where special characters made `fig.cap` mess up? (hint)

That's all for today