

# R Markdown Workshop

## Reproducible Reports

Presented by Emi Tanaka  
School of Mathematics and Statistics




THE UNIVERSITY OF  
SYDNEY




dr.emi.tanaka@gmail.com



@statsgen

! These slides are viewed best by Chrome and occasionally need to be refreshed if elements did not load properly. See here for PDF .

19th Nov 2019 @ SSA Vic | Melbourne, Australia

 Open and inspect the file  
demo-header.Rmd

# Cross Reference

- When you make a header via Rmd

```
# Some Header
```

an **id** is created automatically.

- The id is created by replacing **space** with **-** and making it **all lower case**.
- Now you can link to this header by `[some text](#some-header)`.
- Cross references work for both pdf and html outputs.

## Demo: header cross-references

```
library(tidyverse)
library(knitr)
```

### A look at iris

Let's have a look at the `iris` data set. The dataset contains 150 observations. We'll also have a look at some chicken weights later.

### Count

```
iris %>%
  group_by(Species) %>%
  count(name = "Count")
```

Species	Count
setosa	50
versicolour	50

# Direct Reference for html

---

- For html output, you can also give a link directly to the relevant section
- E.g. open `demo-header.html` in a web browser
- Append say `#chicken-data` to the url. It should look like

`demo-header.html#chicken-data`


- It should have taken you straight to the corresponding header 🏃

# User-defined id

- You can define your own id by appending `{#your-id}`.

```
# Some header {#header1}
```

- Now you can link to this header with the id `header1`.
- Note there should be no space in the id name!

 Open and inspect the file  
demo.bib

# Bibliography

BibTeX citation style format is used to store references in `.bib` files.

Remember that you can get most BibTeX citation for R packages `citation` function. (Scroll below to see the BibTeX citation).


```
citation("xaringan")
```

To cite package `'xaringan'` in publications use:

Yihui Xie (2019). `xaringan`: Presentation Ninja. R package version 0.9. <https://CRAN.R-project.org/package=xaringan>

A BibTeX entry **for** LaTeX users is

```
@Manual{,  
  title = {xaringan: Presentation Ninja},  
  author = {Yihui Xie},  
  year = {2019},  
  note = {R package version 0.9},  
  url = {https://CRAN.R-project.org/package=xaringan},
```

 Open, inspect and knit the file  
`demo-citation.Rmd`



# Citations

- You can include BibTeX by specifying the `bib` file at  
YAML as:

```
bibliography: bibliography.bib
```

`[@bibtex-key]` → (Author et al. 2019)

or

`@bibtex-key` → Author et al. 2019

- See `demo-citation.Rmd`

# Figure References

- Support for figure references are included for output format type `bookdown::pdf_document2` for pdf or `bookdown::html_document2` for html.

```
```${r plot1, fig.cap = "Caption"}  
ggplot(cars, aes(dist, speed)) + geom_point()  
```
```

- Above figure number can be referenced as `\@ref(fig:plot1)`
- The reference label has the prefix `fig:` before the chunk label.

# Table References

- Support for table references are also included for output format type `bookdown::pdf_document2` for pdf or `bookdown::html_document2` for html.

```
```${r table1}  
knitr::kable(cars, booktabs = TRUE, caption = "Caption")  
```
```

- Above table number can be referenced as `\@ref(tab:table1)`
- The reference label has the prefix `tab:` before the chunk label.


# Markdown for Captions

```
```${r plot1, fig.cap = "(ref:label)"}  
ggplot(cars, aes(dist, speed)) + geom_point()  
```
```

- Then the caption can be entered in a separate paragraph with empty lines above and below it

```
(ref:label) This is the *caption* with **markdown**.
```

- You can substitute `label` with another unique label composed of alphanumeric characters, `:`, `-`, or `/`
- This caption supports markdown syntax
- This is great for long captions
- It also works for tables!

 Open and work through  
challenge-08-references.Rmd

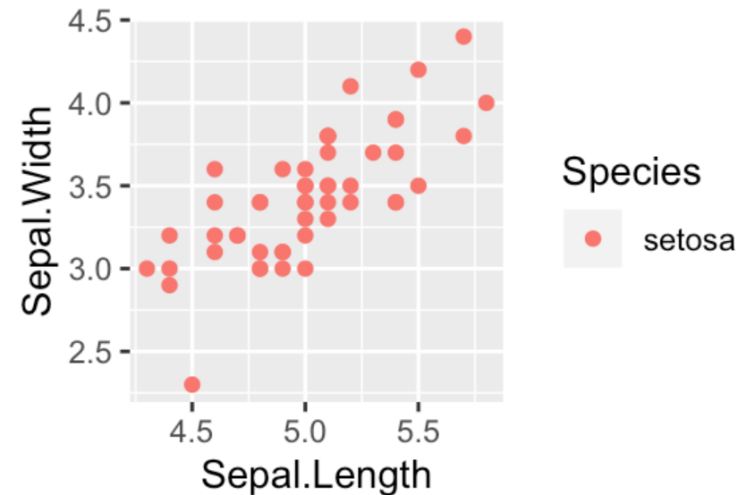
# Parametrized Report

```
---  
title: "Parameterized Report"  
params:  
  species: setosa  
output: html_document  
---  
  
````{r, message = FALSE, fig.dim = c(3,2)}  
library(tidyverse)  
iris %>%  
  filter(Species==params$species) %>%  
  ggplot(aes(Sepal.Length, Sepal.Width))  
  geom_point(aes(color=Species))  
````
```

output

## Parameterized Report

```
library(tidyverse)  
iris %>%  
  filter(Species==params$species) %>%  
  ggplot(aes(Sepal.Length, Sepal.Width)) +  
  geom_point(aes(color=Species))
```

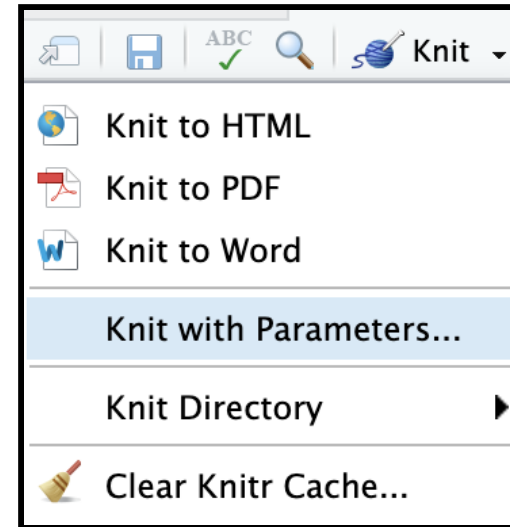


# Knit with Parameters

```
---  
title: "Parameterized Report"  
params:  
  species:  
    label: "Species"  
    value: setosa  
    input: select  
    choices: [setosa, versicolor, virginica]  
  color: red  
  max:  
    label: "Maximum Sepal Width"  
    value: 4  
    input: slider  
    min: 4  
    max: 5  
    step: 0.1
```

**output: html\_document**

```
---
```



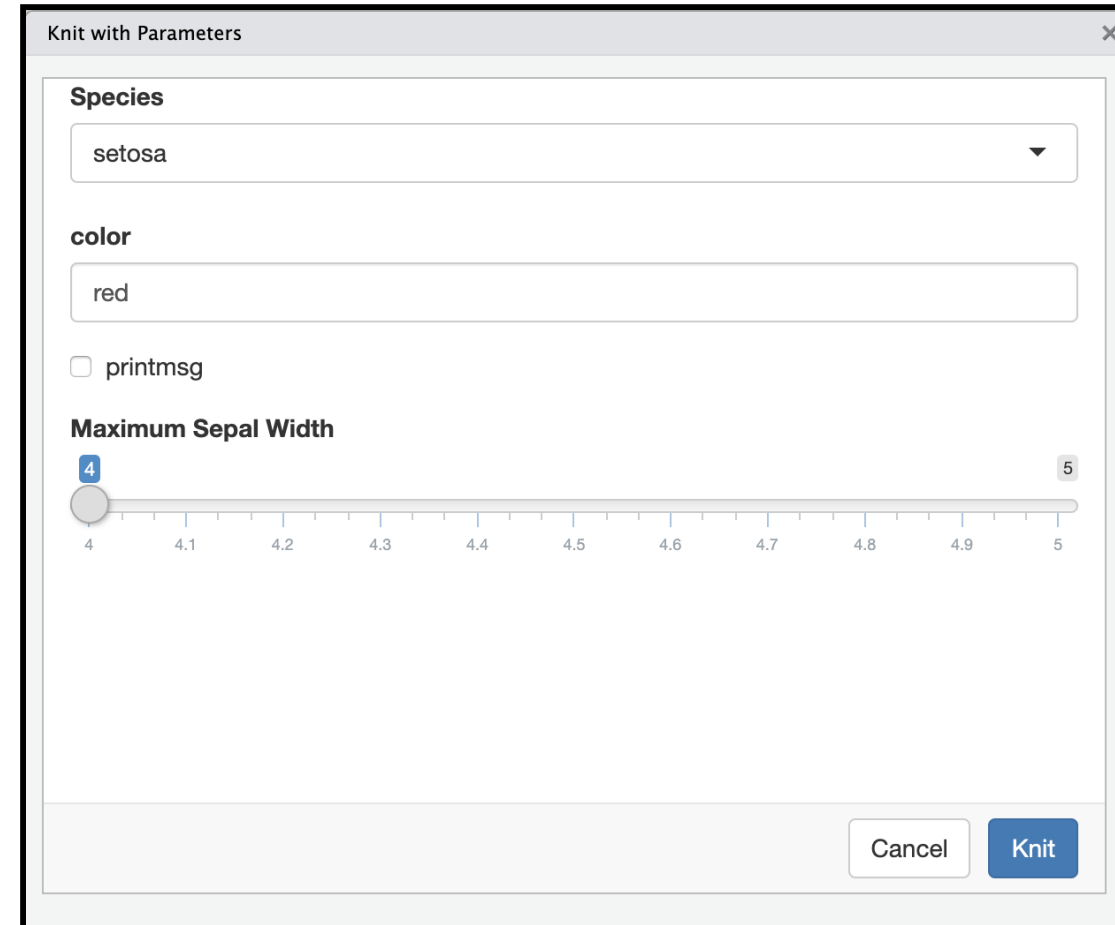
```
```{r, message = params$printmsg}  
library(tidyverse)  
iris %>%  
  filter(Species==params$species) %>%  
  filter(Sepal.Width < params$max) %>%  
  ggplot(aes(Sepal.Length, Sepal.Width)) +  
  geom_point(color = params$color) +  
  labs(title = params$species)  
```
```

# Shiny Report Generator

```
---  
title: "Parameterized Report"  
params:  
  species:  
    label: "Species"  
    value: setosa  
    input: select  
    choices: [setosa, versicolor, virginica]  
  color: red  
  max:  
    label: "Maximum Sepal Width"  
    value: 5  
    input: slider  
    min: 4  
    max: 5  
    step: 0.05
```

**output: html\_document**

---



Knit with Parameters

**Species**  
setosa

**color**  
red


printmsg

**Maximum Sepal Width**  
4 5

4 4.1 4.2 4.3 4.4 4.5 4.6 4.7 4.8 4.9 5

Cancel Knit



 Open and work through  
challenge-09-params.Rmd

# R Markdown via Command Line

## demo-render.Rmd


```
---  
title: "Parameterized Report"  
params:  
  species: setosa  
output: html_document  
---  
  
````{r, message = FALSE, fig.dim = c(3,2)}  
library(tidyverse)  
iris %>%  
  filter(Species==params$species) %>%  
  ggplot(aes(Sepal.Length, Sepal.Width)) +  
  geom_point(aes(color=Species))  
````
```

You can knit this file via R command by using the `render` function:

```
library(rmarkdown)  
render("demo-render.Rmd")
```
















You can overwrite the YAML values by supplying arguments to `render`:

```
library(rmarkdown)  
render("demo-render.Rmd",  
      output_format = "pdf_document",  
      params = list(species = "virginica"))
```

 Open and work through  
challenge-10-letters.Rmd

# Themes: `html_document`

You can change the look of the html document by specifying themes:

- `default` 
- `cerulean` 
- `journal` 
- `flatly` 
- `darkly` 
- `readable` 
- `spacelab` 
- `united` 
- `cosmo` 
- `lumen` 
- `paper` 
- `sandstone` 
- `simplex` 
- `yeti` 
- `NULL` 


```
output:
```






```
  html_document:
```

```
    theme: cerulean
```

These bootswatch themes attach the whole bootstrap library which makes your html file size larger.

# prettydoc

prettydoc  is a community contributed theme that is light-weight:

- cayman 
- tactile 
- architect 
- leonids 
- hpstr 


output:





```
prettydoc::html_pretty:  
  theme: cayman
```

See more about it below:

<https://prettydoc.statr.me/>

# rmdformats

rmdformats  contains four built-in html formats:

- readthedown 
- html\_clean 
- html\_docco 
- material 


























You can use these formats by simply specifying the output in YAML as below:

```
output: rmdformats::readthedown
```

See more about it below:

<https://github.com/juba/rmdformats>

# rticles - LaTeX Journal Article Templates

- acm 
- acs 
- aea 
- agu 
- amq 
- ams 
- asa 
- biometrics 
- copernicus 
- elsevier 
- frontiers 
- ieee 
- jss 
- mdpi 
- mnras 
- peerj 
- plos 
- pnas 
- rjournal 
- rsos 
- rss 
- sage 
- sim 
- springer 
- tf 

Go to RStudio > File > New File > R Markdown ... > From Template

# External Files in Templating

- When using `rticles`, each journal usually require external files (e.g. `cls` or image files).
- These external components are stored within the package.
- So use `draft` instead of `render`!

## GUI

- RStudio > File > New File > R Markdown ... > From Template

## Command line

```
rmarkdown::draft("file.Rmd",  
  template = "biometrics_article",  
  package = "rticles")
```




# Making your own R Markdown template

- You need to make an R package first!  
Go to RStudio > New Project > New Directory > R Package or `usethis::create_package()`

- When you are in your R package project,

```
usethis::use_rmarkdown_template("<Name>")
```

- Modify the `skeleton/skeleton.Rmd` to how you want and add all external files to the `skeleton` folder.
- Install your package.
- 🎉 And now find it at RStudio > File > New File > R Markdown > From Template.

 Create your own  
R Markdown Template Package!

# Session Information

---

```
devtools::session_info()
```

```
– Session info _____  
  setting      value  
  version      R version 3.6.0 (2019-04-26)  
  os           macOS Mojave 10.14.6  
  system       x86_64, darwin15.6.0  
  ui           X11  
  language     (EN)
```

These slides are licensed under

