Getting started with R, RStudio and R Markdown

Professor Andy Field

@profandyfield

www.youtube.com/user/ProfAndyField/

www.discoveringstatistics.com

www.milton-the-cat.rocks

www.discovr.rocks

Key information

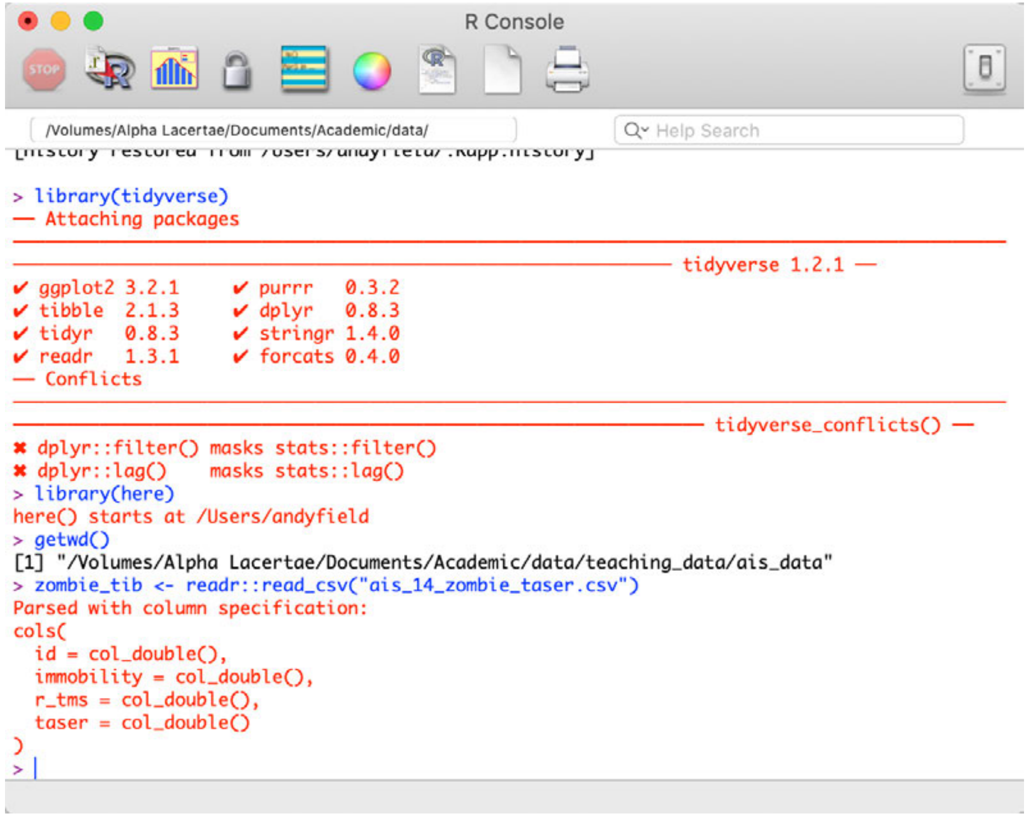
Teaching

- 1 × weekly lecture (2 in weeks 1 and 2) on theory
- 1 × weekly practical class (2 in weeks 1 and 2) about R and RStudio

Assessment

- 2 × 24-hour Take Away Papers (TAPs)
  - TAP 1: 30%
  - TAP 2: 30%
- 1 × Report (40%)

Part 1: Introducing R and RStudio


R Console
/Volumes/Alpha Lacertae/Documents/Academic/data/
Help Search
> library(tidyverse)
— Attaching packages
tidyverse 1.2.1 —
✔ ggplot2 3.2.1 ✔ purrr 0.3.2
✔ tibble 2.1.3 ✔ dplyr 0.8.3
✔ tidyr 0.8.3 ✔ stringr 1.4.0
✔ readr 1.3.1 ✔ forcats 0.4.0
— Conflicts
tidyverse_conflicts() —
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag() masks stats::lag()
> library(here)
here() starts at /Users/andyfield
> getwd()
[1] "/Volumes/Alpha Lacertae/Documents/Academic/data/teaching_data/ais_data"
> zombie_tib <- readr::read_csv("ais_14_zombie_taser.csv")
Parsed with column specification:
cols(
id = col_double(),
immobility = col_double(),
r_tms = col_double(),
taser = col_double()
)
>

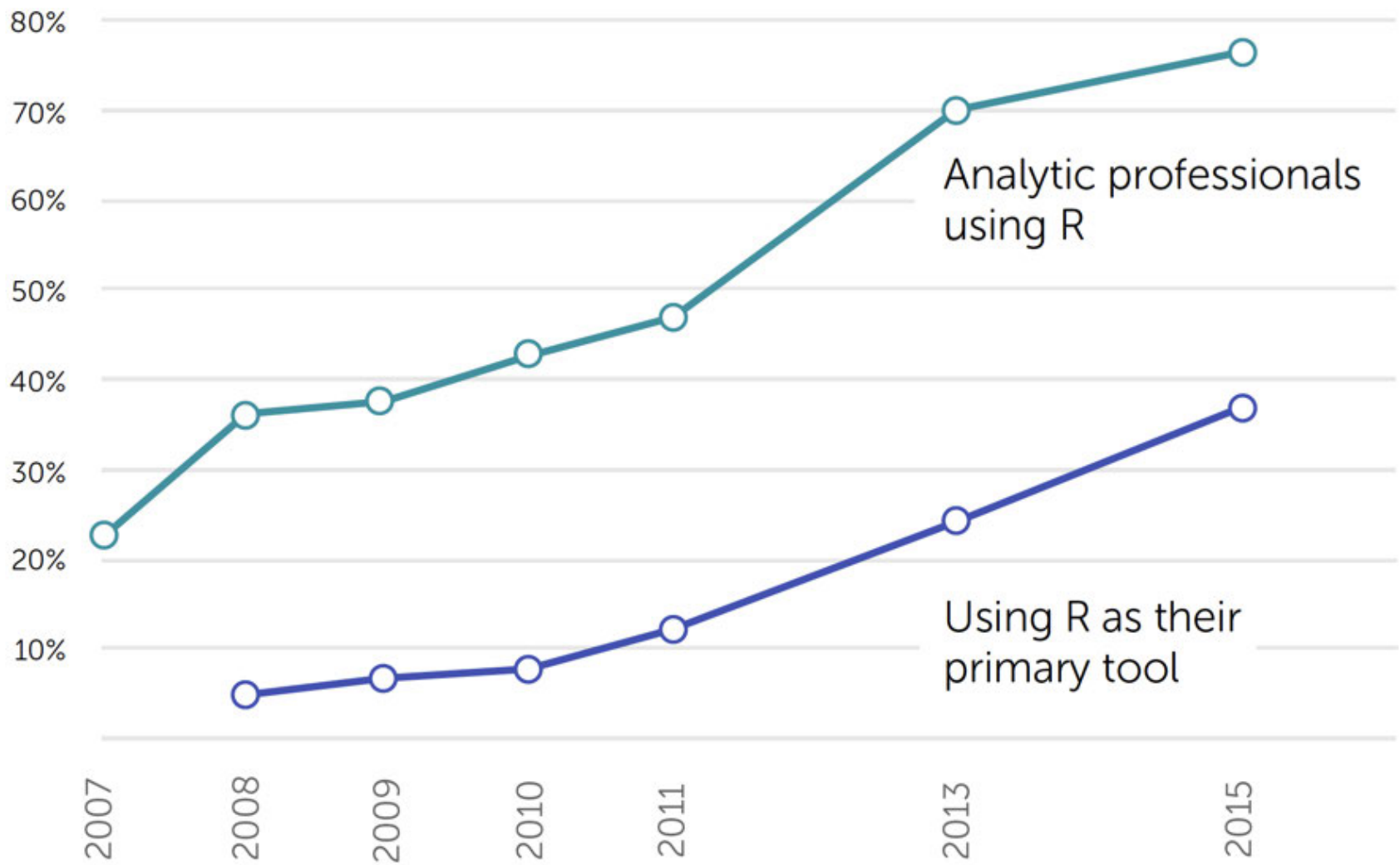

RStudio®

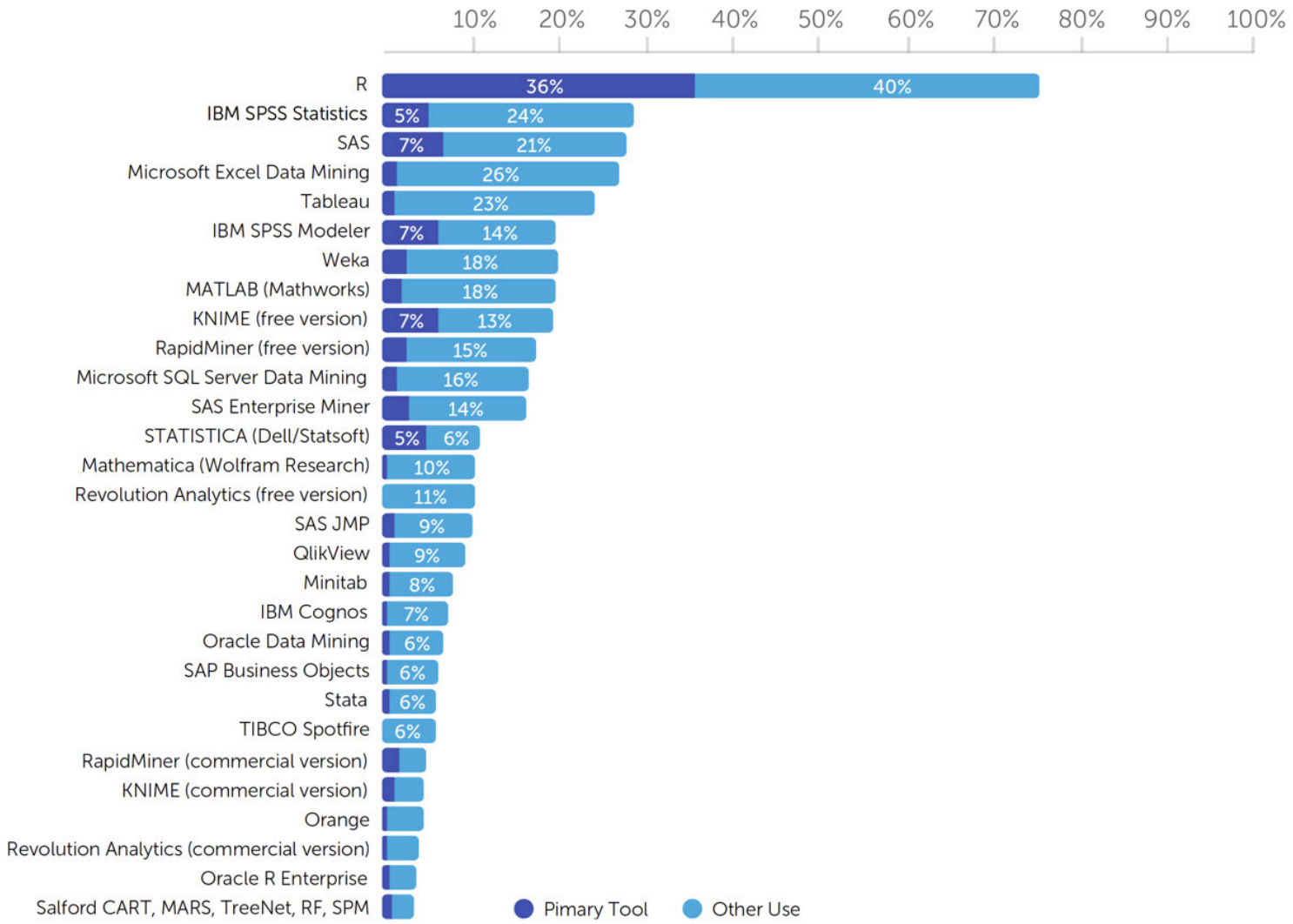
- Free software environment for statistical analysis and graphics

- A free integrated development environment (IDE) for R
- You use RStudio as a way to interact with R
  - Install R and forget about it until such time that you need to update it
  - Install RStudio
  - Set up RStudio to suit your personality and enjoy your status as rad data scientist hipster

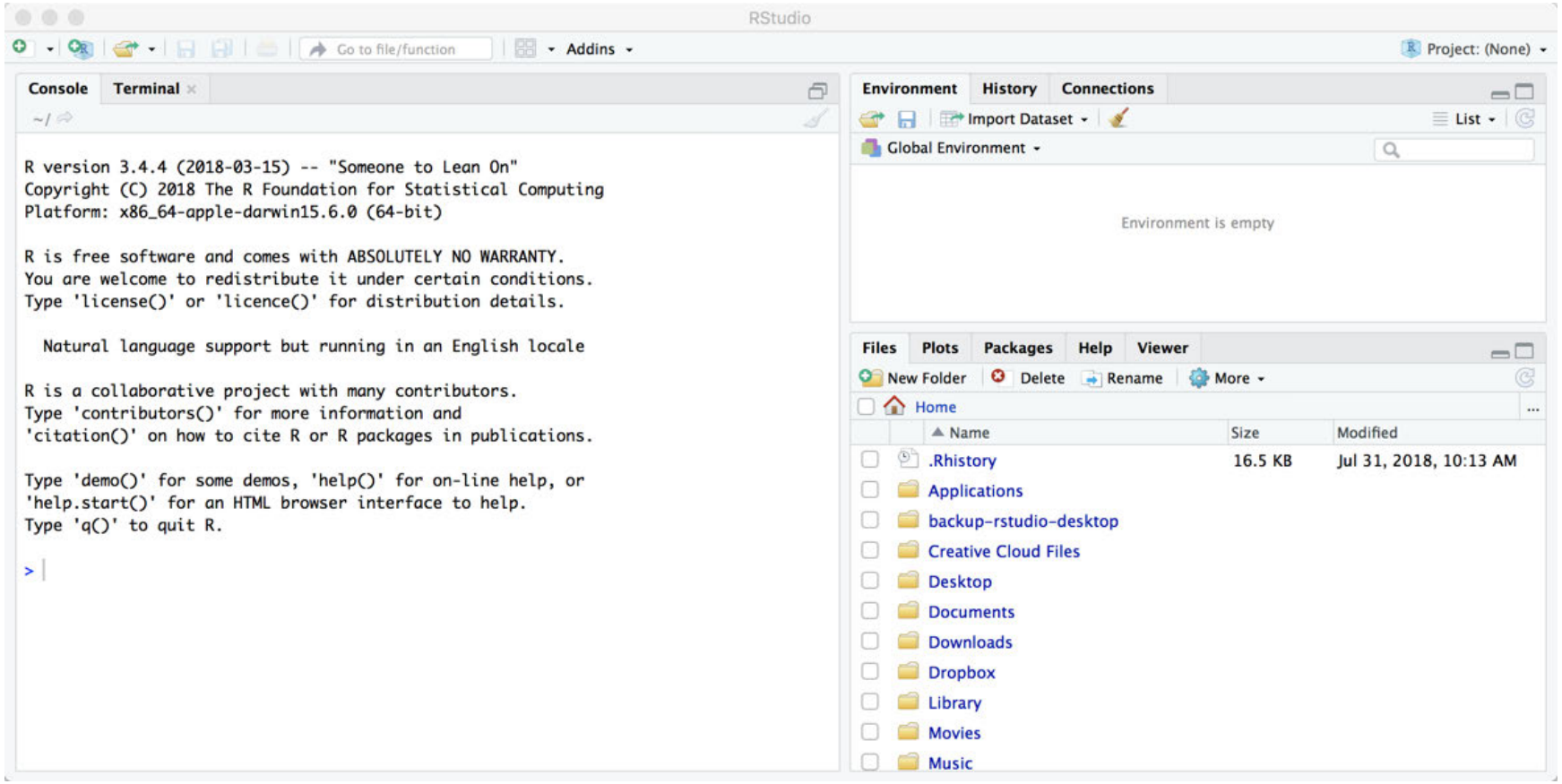
Why R?

- Transferable skills
- R is the most widely used data analysis software
- Reproducible science
- Cutting edge
- One stop shop
- RStudio is amazeballs


80%
70%
60%
50%
40%
30%
20%
10%
Analytic professionals using R
Using R as their primary tool
2007
2008
2009
2010
2011
2013
2015

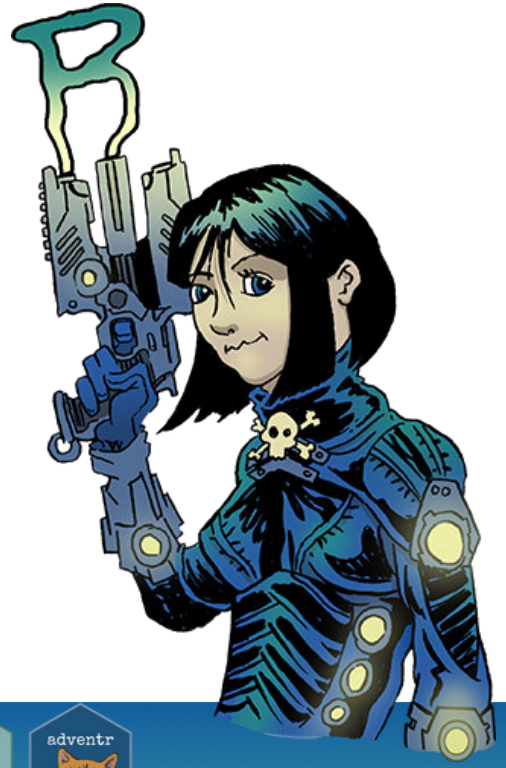

10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
R
36%
40%
IBM SPSS Statistics
5%
24%
SAS
7%
21%
Microsoft Excel Data Mining
26%
Tableau
23%
IBM SPSS Modeler
7%
14%
Weka
18%
MATLAB (Mathworks)
18%
KNIME (free version)
7%
13%
RapidMiner (free version)
15%
Microsoft SQL Server Data Mining
16%
SAS Enterprise Miner
14%
STATISTICA (Dell/Statsoft)
5%
6%
Mathematica (Wolfram Research)
10%
Revolution Analytics (free version)
11%
SAS JMP
9%
QlikView
9%
Minitab
8%
IBM Cognos
7%
Oracle Data Mining
6%
SAP Business Objects
6%
Stata
6%
TIBCO Spotfire
6%
RapidMiner (commercial version)
KNIME (commercial version)
Orange
Revolution Analytics (commercial version)
Oracle R Enterprise
Salford CART, MARS, TreeNet, RF, SPM
Pimary Tool
Other Use

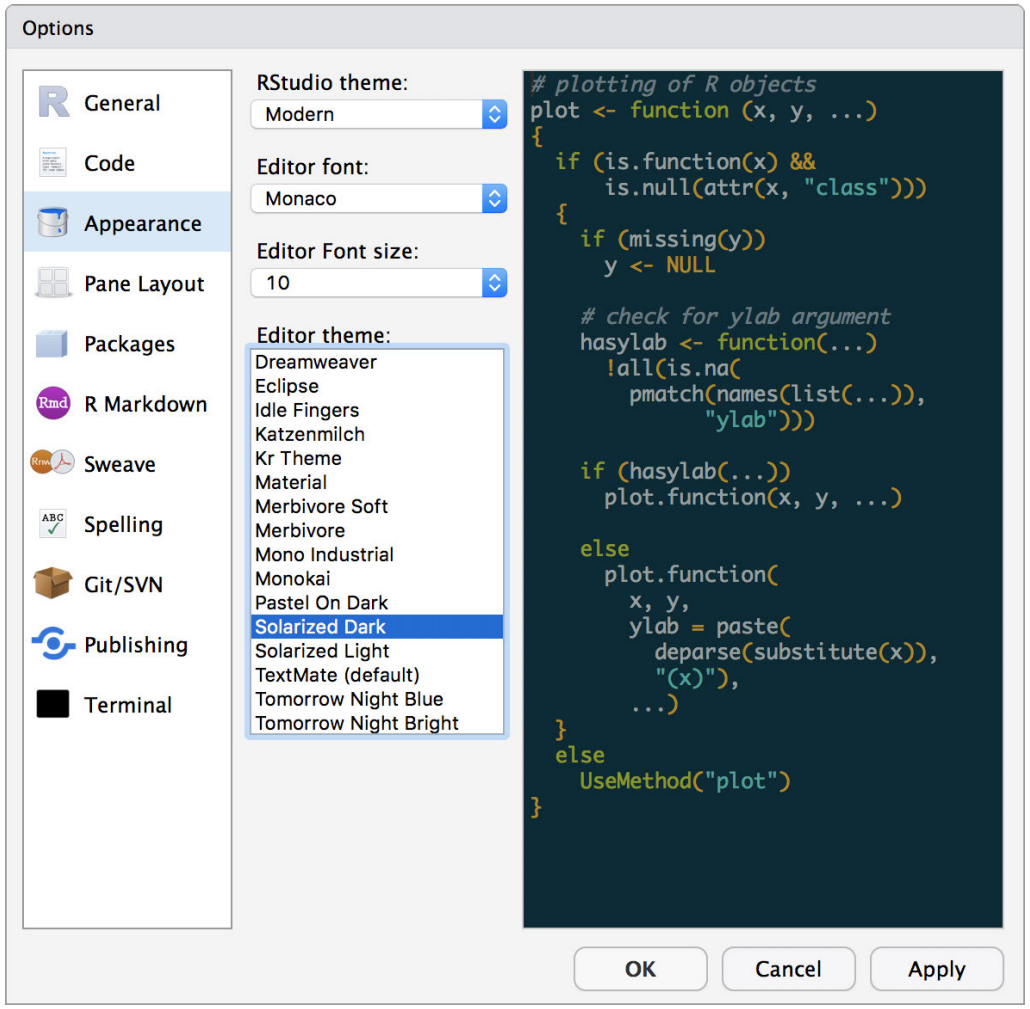
Starting up RStudio



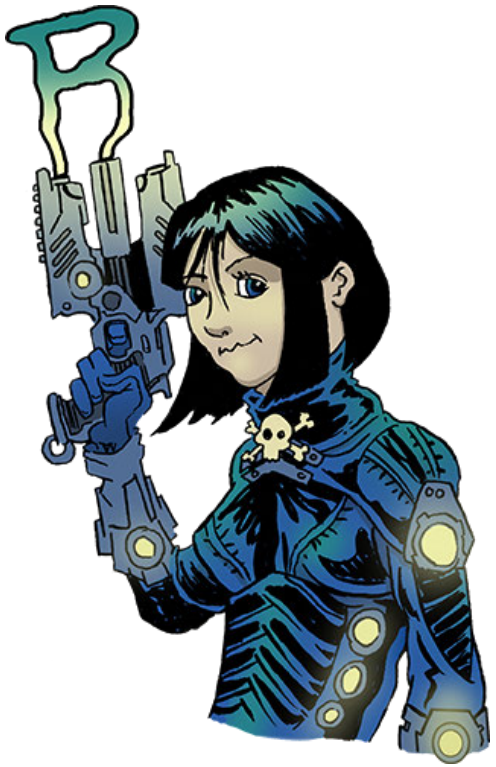
Try it! Colour scheme

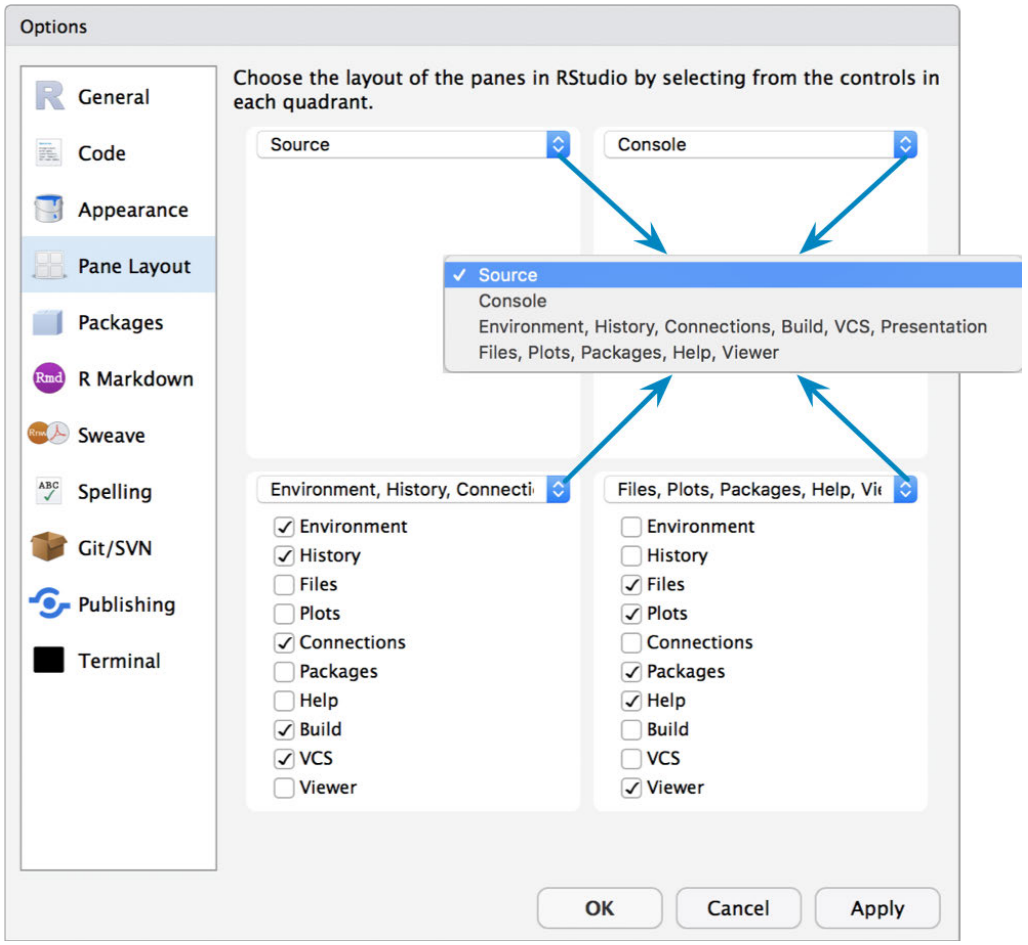
- Windows
  - Tools > Options
- MacOS
  - Tools > Global Options
  - RStudio > Preferences

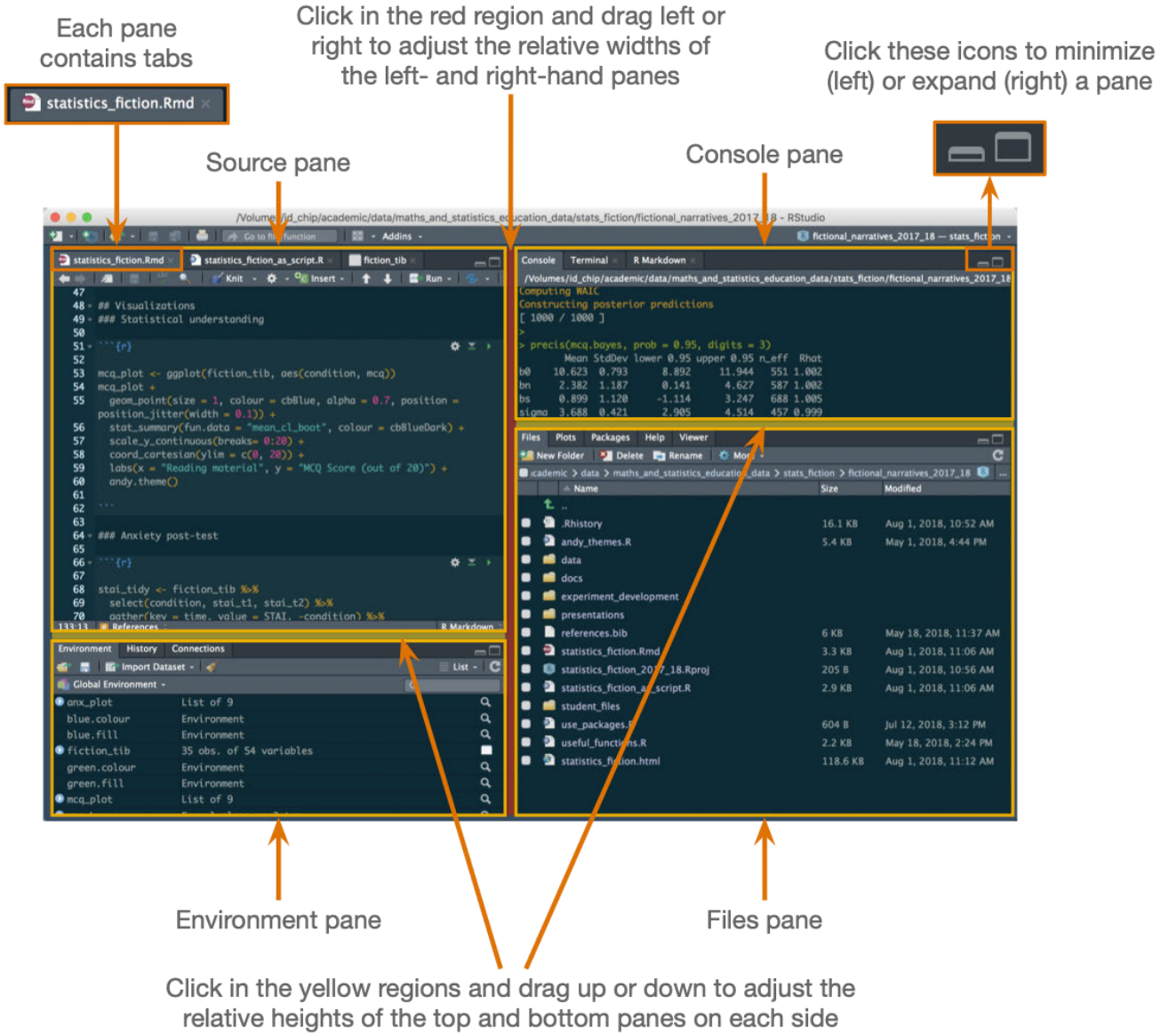




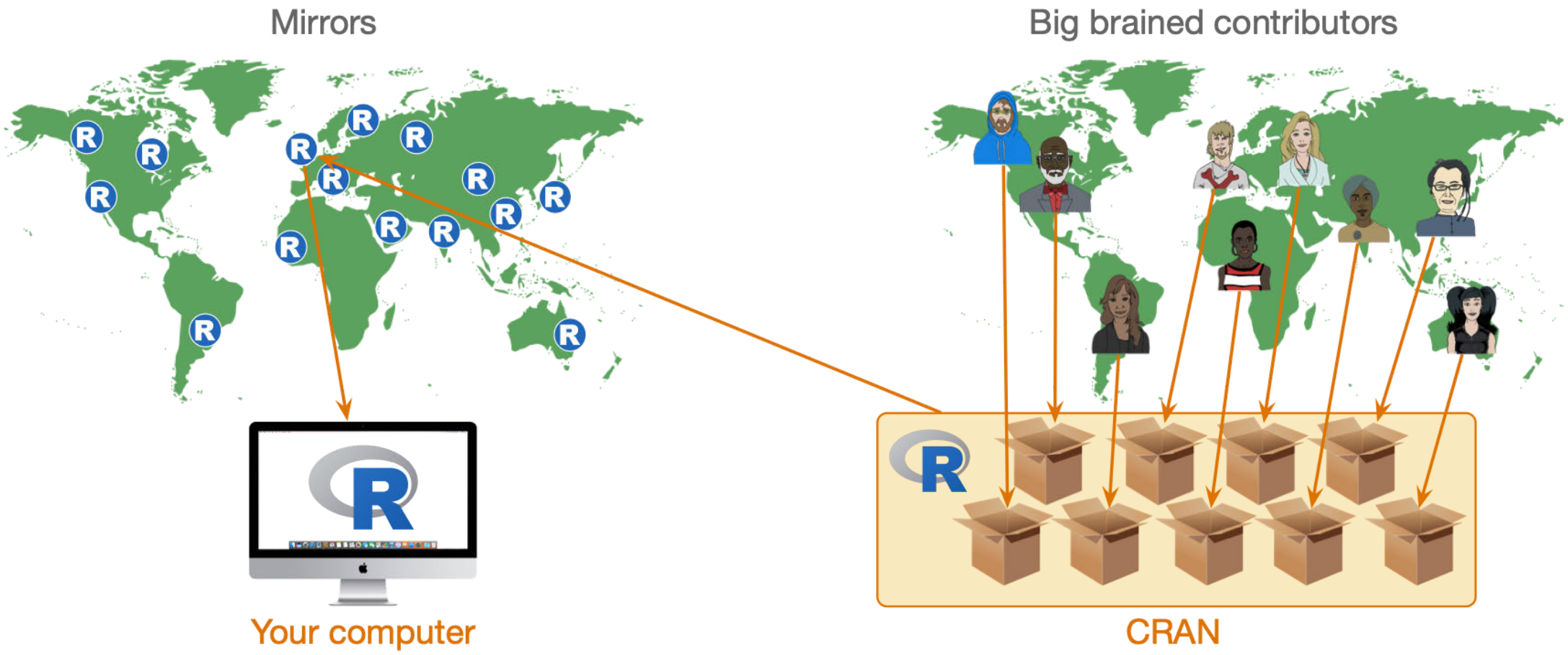
Try it! Pane locations






Each pane contains tabs
statistics_fiction.Rmd
Click in the red region and drag left or right to adjust the relative widths of the left- and right-hand panes
Click these icons to minimize (left) or expand (right) a pane
Source pane
Console pane
Environment pane
Files pane
Click in the yellow regions and drag up or down to adjust the relative heights of the top and bottom panes on each side

Packages



Installing and loading packages

Install the package from CRAN

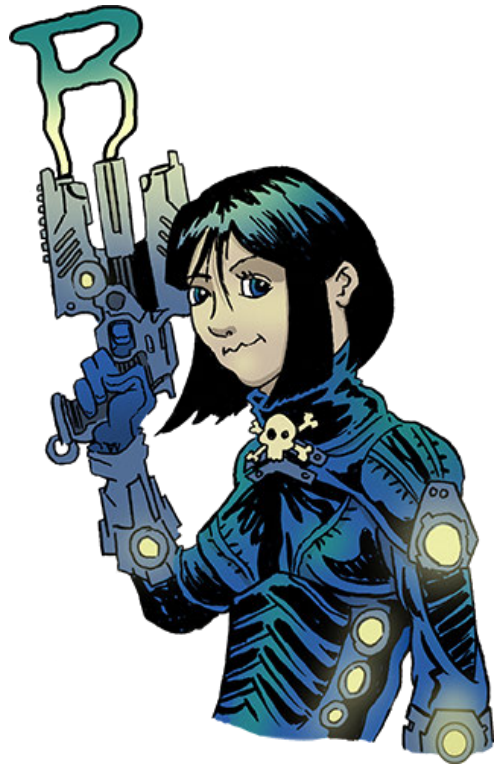
- You need to install the package into R's repository of packages on your computer.
- Every time you update or re-install R you need to re-install packages to use them.
- `install.packages("package_name")`

Load the package

- To use a particular package in a current session load it from the repository.
- `library(package_name)`

Try it!

- Execute `install.packages("readr")` in the console.
- Execute `install.packages("remotes")` in the console.



Installing the discovr package

```
remotes::install_github("profandyfield/discovr")
```

Part 2: Interacting with R and RStudio

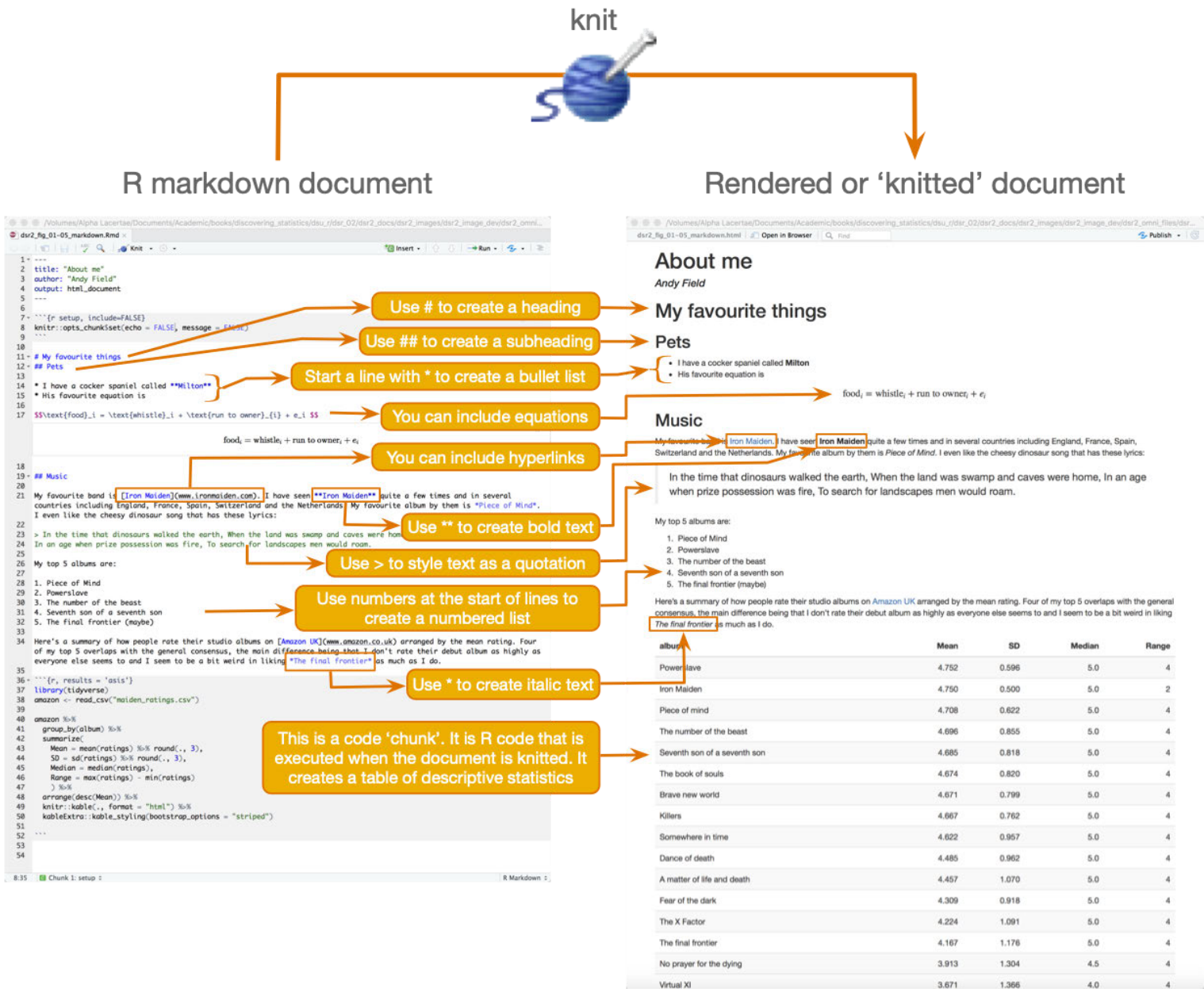
Interacting with R

- The console (boo!)
  - Type commands at the console prompt
  - Bad for reproducibility/your sanity
  - Great for getting help, trying things out

- Script files (meh!)
  - Great for reproducibility
  - Bad for integrating analysis into documents

- R markdown document (hooray!)
  - A document that combines marked-up text and code
  - 'Knits' to a nicely-formatted .html, .docx or .pdf (LaTeX required) file.
  - Great for reproducibility
  - Great for integrating analysis into documents
  - All coursework submitted as html file

Open the file my_first_markdown.Rmd and click  to 'knit' the document.


knit
R markdown document
Rendered or 'knitted' document
Use # to create a heading
Use ## to create a subheading
Start a line with * to create a bullet list
You can include equations
You can include hyperlinks
Use ** to create bold text
Use > to style text as a quotation
Use numbers at the start of lines to create a numbered list
Use * to create italic text
This is a code 'chunk'. It is R code that is executed when the document is knitted. It creates a table of descriptive statistics
About me
Andy Field
My favourite things
Pets
I have a cocker spaniel called Milton
His favourite equation is
Music
In the time that dinosaurs walked the earth, When the land was swamp and caves were home, In an age when prize possession was fire, To search for landscapes men would roam.
My top 5 albums are:
1. Piece of Mind
2. Powerslave
3. The number of the beast
4. Seventh son of a seventh son
5. The final frontier (maybe)

Inserting code chunks

Windows

- *ctrl + alt + i*
- Code > Insert Chunk

Mac

- *cmd + opt + i* (⌘ + ⌥ + i)
- Code > Insert Chunk

Creating objects

```
metallica <- c("Lars","James","Jason", "Kirk")
metallica <- metallica[metallica != "Jason"]
metallica <- c(metallica, "Rob")
```

object

instructions

Assignment operator

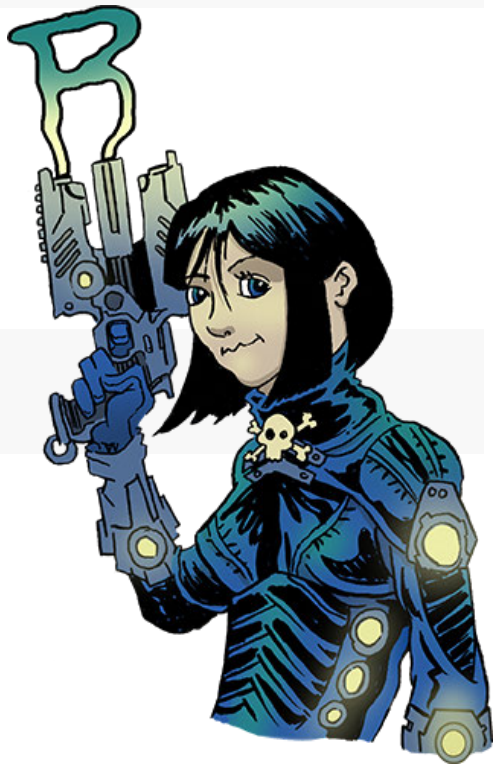
Try it

- In my_first_markdown.Rmd create this code chunk

```
energy_tib <- eddie_tib[c("energy", "valence")]
```

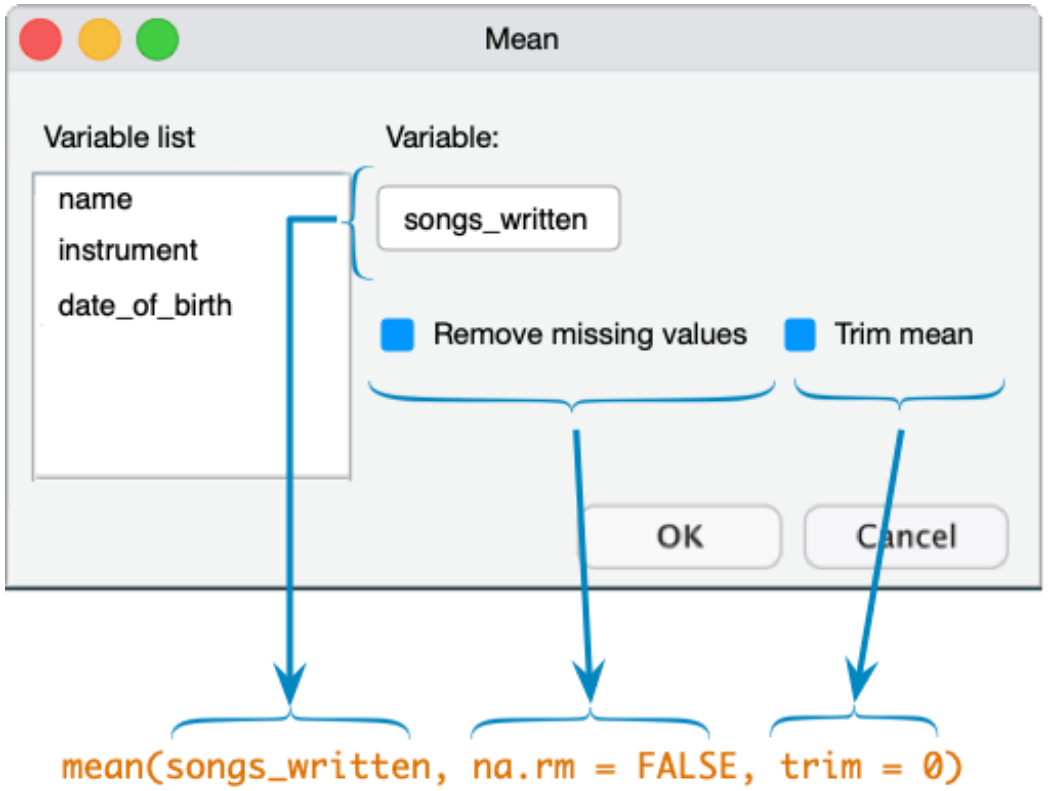
- Click ▶ to execute the code in the code chunk
- Click ⏬ to execute all previous code chunks
- Add the second line to the code chunk and click ▶

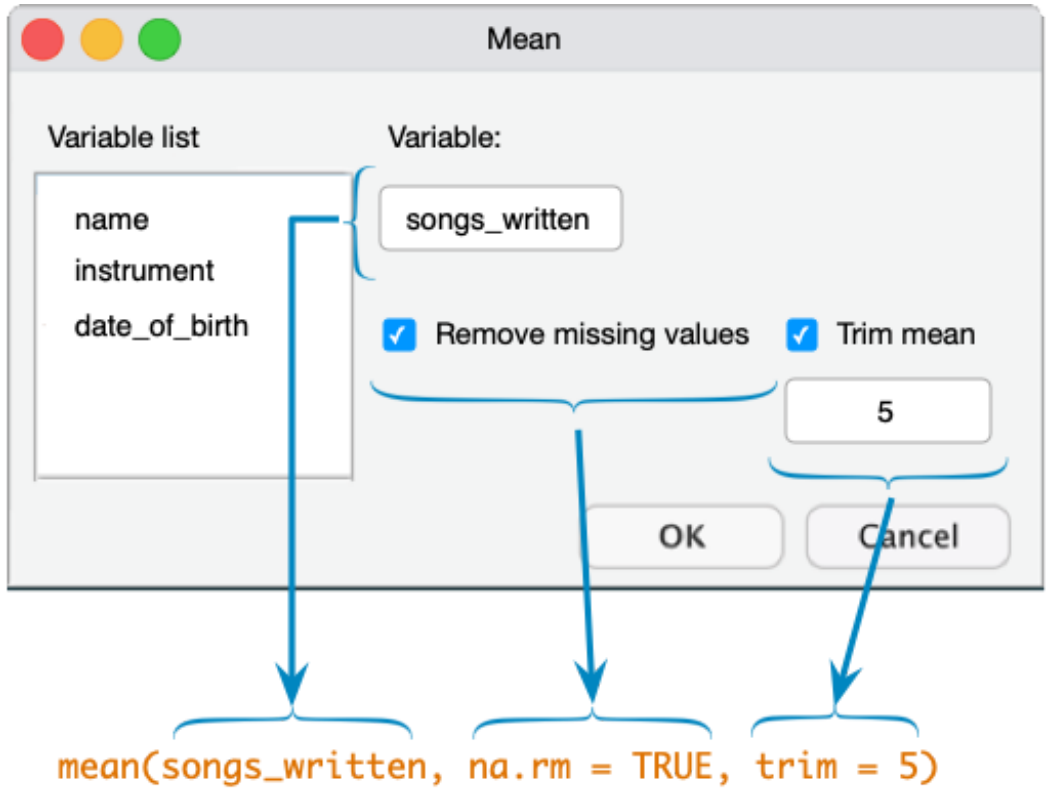
```
energy_tib <- eddie_tib[c("energy", "valence")]
energy_tib
```



Functions

```
mean(variable_name, ... arguments/options ...)
mean(variable_name, na.rm = TRUE, trim = 0)
```



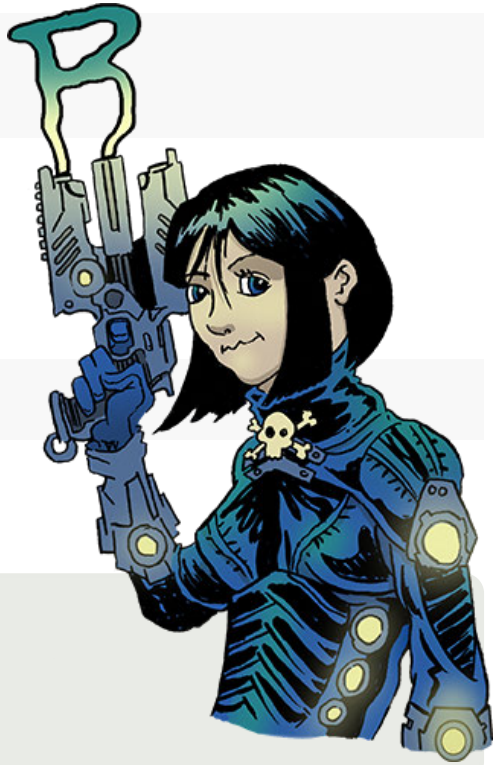


Try it

In your my_first_markdown.Rmd file

- Insert a code chunk, type the first command into it and execute.

```
mean_energy <- mean(energy_tib$energy)
mean_energy
```



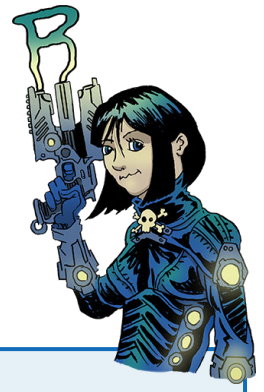
- Click ▶ to execute the code in the code chunk
- Insert a second code chunk, type the second command into it and execute.

```
plot(energy_tib)
```

- Click ▶ to execute the code in the code chunk

- Click  to knit the document.

Try it

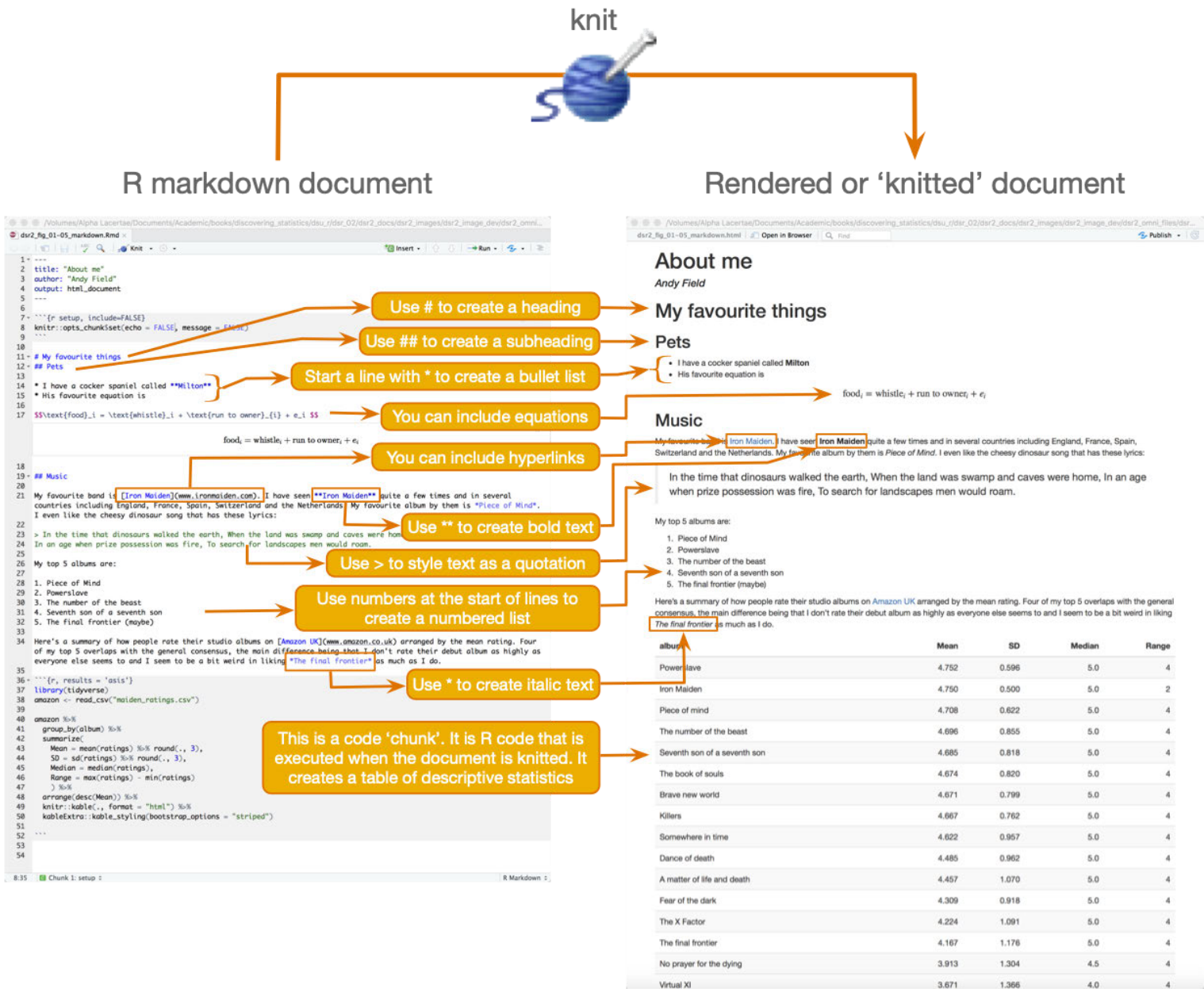


- Move one of your newly-created code chunks to before the code chunk called `subset_data`.
- Click  to knit the document.

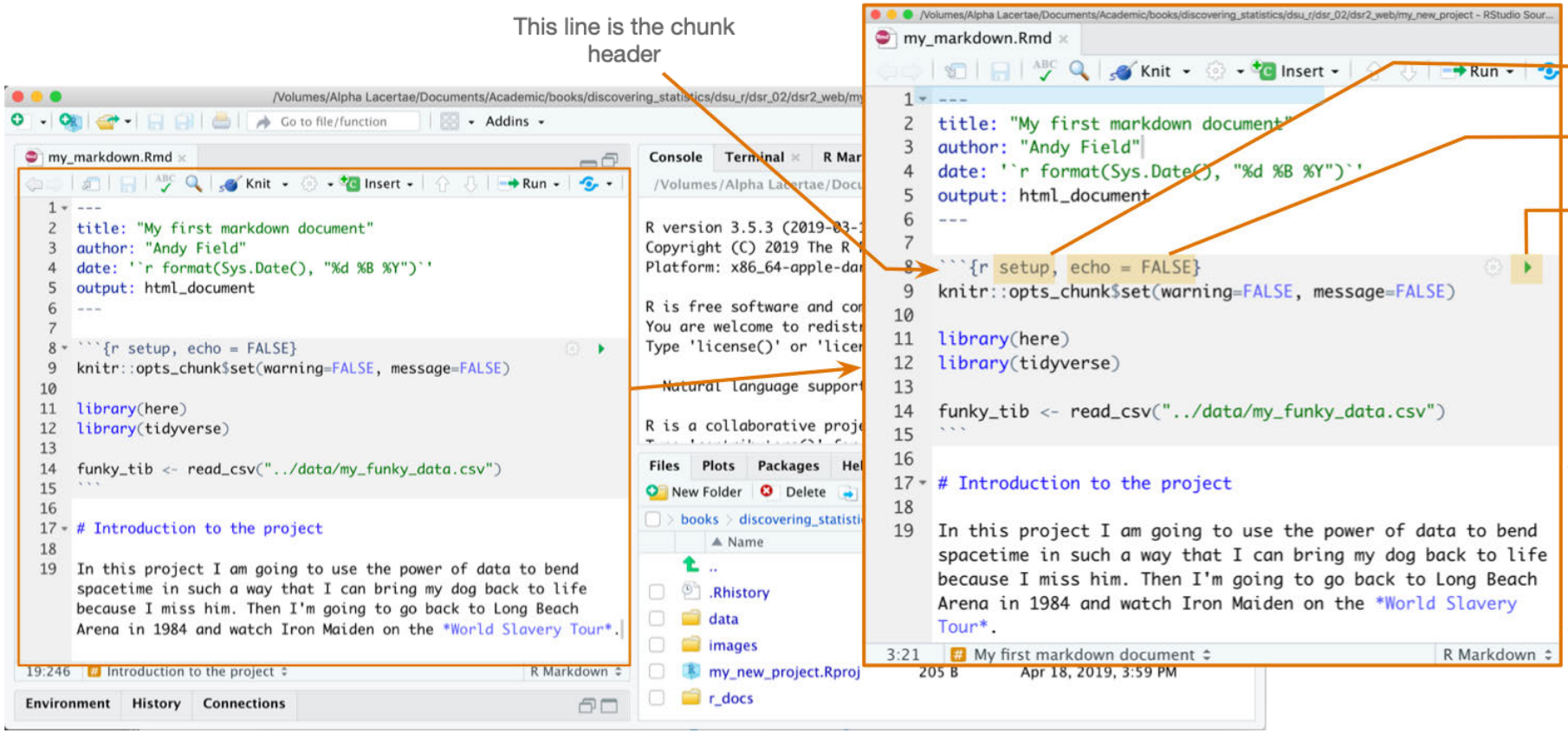
Knitting Tips

- Code chunks are knitted in the order they appear in the markdown document.
  - Make sure you create objects BEFORE you try to use them.
- Do NOT include `install.packages()` in markdown files or the package will be installed every time you knit the document!
  - Execute `install.packages()` commands at the command line

Part 3: More R Markdown


knit
R markdown document
Rendered or 'knitted' document
Use # to create a heading
Use ## to create a subheading
Start a line with * to create a bullet list
You can include equations
You can include hyperlinks
Use ** to create bold text
Use > to style text as a quotation
Use numbers at the start of lines to create a numbered list
Use * to create italic text
This is a code 'chunk'. It is R code that is executed when the document is knitted. It creates a table of descriptive statistics
About me
Andy Field
My favourite things
Pets
I have a cocker spaniel called Milton
His favourite equation is
Music
In the time that dinosaurs walked the earth, When the land was swamp and caves were home, In an age when prize possession was fire, To search for landscapes men would roam.
My top 5 albums are:
1. Piece of Mind
2. Powerslave
3. The number of the beast
4. Seventh son of a seventh son
5. The final frontier (maybe)

This is a markdown file

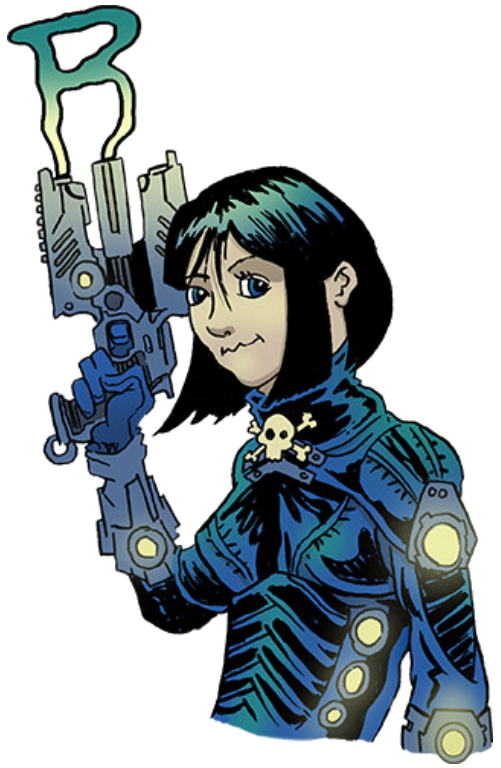


Code chunk options

- `echo=FALSE`: the code will be evaluated but not reproduced in the knitted document.
- `eval=FALSE`: the code will not be evaluated.
- `include=FALSE`: RStudio will evaluate the code but neither the code nor output are displayed in the knitted document.
- `results="hide"`: the code will be evaluated and displayed in the knitted document but the output will be omitted.
- Include `fig.width=9` and `fig.height=5` to set the dimensions of an image
  - Obviously replace 9 and 5 with the values you want
- Use `results = 'asis'` when creating tables from code/output

Try it!

- Using the code chunks we have already created
- Change the code chunk headers to include `echo=FALSE`, re-knit the document and compare the result.
- Repeat this process but including `eval=FALSE`



Markdown: headers

Markdown

```
# Level 1 heading
## Level 2 heading
### Level 3 heading
#### Level 4 heading
```

Knitted text

Level 1 heading

Level 2 heading

Level 3 heading

Level 4 heading

Markdown: lists

Markdown

```
* This is the first bullet point
  + this is a sub-bullet
  + so is this
* This is the second bullet
  + This is a sub-bullet
    - A third layer of bullet madness
    - It had to be done
* This is the third main bullet
```

Knitted text

- This is the first bullet point
  - this is a sub-bullet
  - so is this
- This is the second bullet
  - This is a sub-bullet
    - A third layer of bullet madness
    - It had to be done
- and this is the third bullet

Markdown: emphasizing text

Markdown

```
*italic text*
**bold text**
Text^superscript^
Text~subscript~
```

Knitted text

italic text

bold text

$\text{Text}^{\text{superscript}}$

$\text{Text}_{\text{subscript}}$

Markdown: images

```
![Caption for image](filepath_for_image)

![Figure 1: my lovely spaniel]
(media/milton_circle_2019.png)
```

Figure 1: my lovely spaniel

Markdown: hyperlinks

```
[text_describing_link](url)
```

Markdown

```
The **discovr** package is a collection of interactive tutorials for learning R. Get it
from [www.discovr.rocks/discovr/](https://www.discovr.rocks/discovr/)
```

Knitted text

The discovr package is a collection of interactive tutorials for learning R. Get it from www.discovr.rocks/discovr/

Markdown: equations

Markdown

We can include the linear model in its own paragraph like this:

```
$$ Y_i = b_0 + b_1X_i + \epsilon_i $$
```

Knitted text

We can include the linear model in its own paragraph like this:

$$Y_i = b_0 + b_1X_i + \epsilon_i$$

Tables

Raw markdown

```
tap_tib
```

```
## # A tibble: 1,740 × 9
##    id    intervention efficacy did_bhv id_school time_cat intention time_num positivity
##    <chr> <chr>           <dbl>   <dbl> <chr>     <chr>        <dbl>    <dbl>      <dbl>
##  1 zy56g Group               3       0 school_1  baseline        35        0          5
##  2 xt30h Leaflet             3       1 school_1  baseline        36        0          3
##  3 st59q Group               4       0 school_1  baseline        34        0          4
##  4 vh02j Group               7       1 school_1  baseline        33        0          2
##  5 vn97y Leaflet             3       1 school_1  baseline        31        0          3
##  6 av87d Leaflet             5       1 school_1  baseline        36        0          4
##  7 nu84x Leaflet             3       0 school_1  baseline        33        0          6
##  8 zr06y Group               7       1 school_1  baseline        34        0          5
##  9 ls25l Leaflet             2       0 school_1  baseline        32        0          3
## 10 zp67x Group               5       0 school_1  baseline        35        0          2
## # … with 1,730 more rows
```

Tables

Using `results = 'asis'` and `kable()`

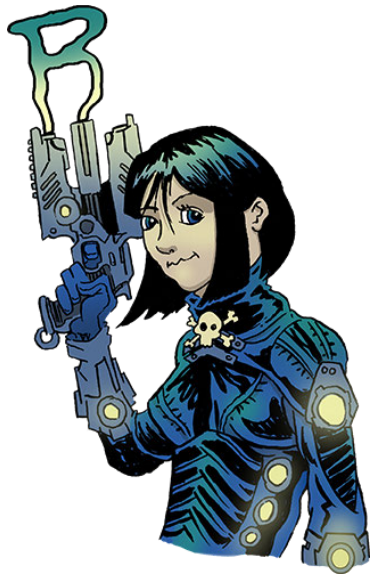
```
tap_tib %>%
  knitr::kable(caption = "Table 1: My nice table", digits = 2)
```

Table 1: My nice table

id	intervention	efficacy	did_bhv	id_school	time_cat	intention	time_num	positivity
zy56g	Group	3	0	school_1	baseline	35	0	5
xt30h	Leaflet	3	1	school_1	baseline	36	0	3
st59q	Group	4	0	school_1	baseline	34	0	4
vh02j	Group	7	1	school_1	baseline	33	0	2
vn97y	Leaflet	3	1	school_1	baseline	31	0	3
av87d	Leaflet	5	1	school_1	baseline	36	0	4

Try it!

- In your markdown file create a level 1 header that reads 'About me'
- Write a sentence or two about yourself using bold and italic somewhere
- Create a level two heading 'My favourite things'
- Within that section write a list of five of your favourite things. Include a URL to one of them.
    - (It could be books, movies, bands, songs, statisticians).



Part 4: Workflow in RStudio

RStudio project files

A file created by RStudio with the extension .Rproj

- Stores information about the containing folder
- Restores the previous state of the project (i.e. what documents/tabs were open)

Opening a project file sets the working directory to the folder containing the project file

- You can use relative file paths
- The project will work on any machine you care to use
- You can share your project folder with others and it'll work for them

Use project files!

Filepaths made easy (well, easier ...)

Relative paths

If you use an RStudio project, you can use relative paths

- `tap_tib <- readr::read_csv("../data/tap_parenting.csv")`

The here package

If you use an RStudio project, `here::here()` returns the project folder

- Text within `here()` returns the folder or file within the project folder that matches the text
- `tap_tib <- here::here("data/tap_parenting.csv") %>% readr::read_csv()`

```
library(here)
here::here()
```

```
## [1]
"/Volumes/alpha_lacertae/documents/academic/teaching/disc_stats_module/disc_stat_presentations"
```

```
here::here("data")
```

```
## [1]
"/Volumes/alpha_lacertae/documents/academic/teaching/disc_stats_module/disc_stat_presentations/data"
```

```
here::here("data/tap_parenting.csv")
```

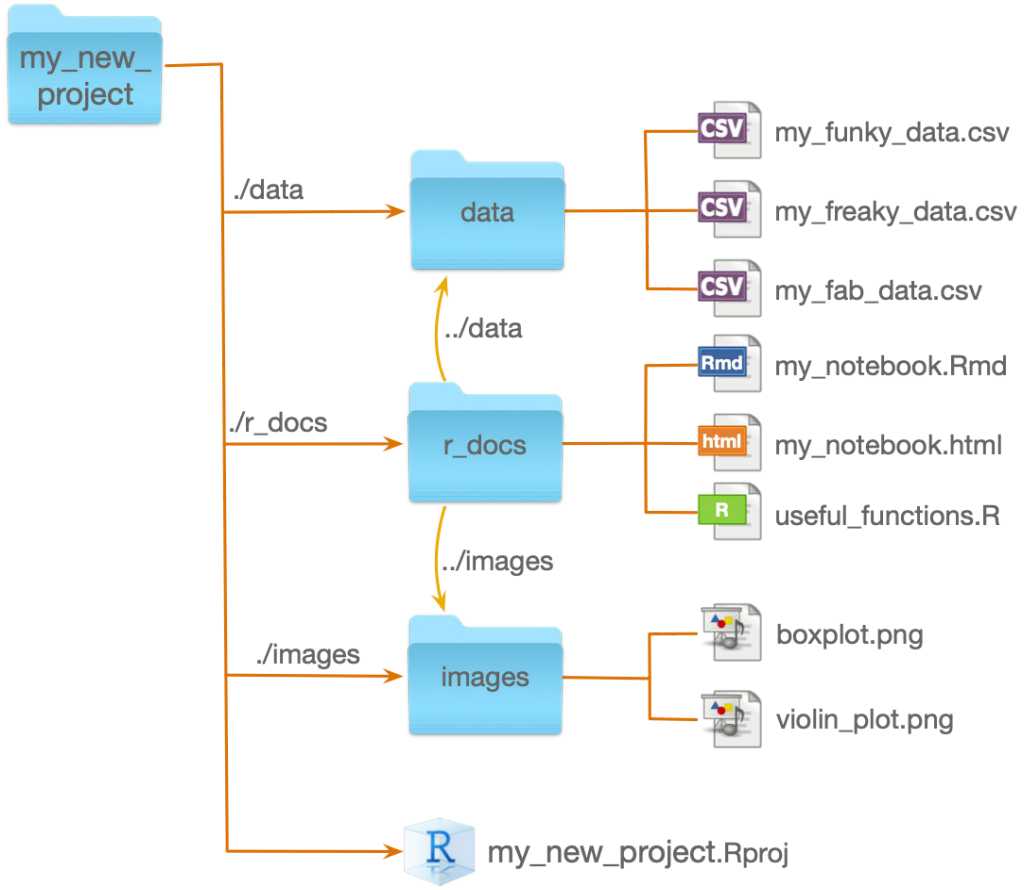
```
## [1]
"/Volumes/alpha_lacertae/documents/academic/teaching/disc_stats_module/disc_stat_presentations/data/tap_pa
```

Get organized!

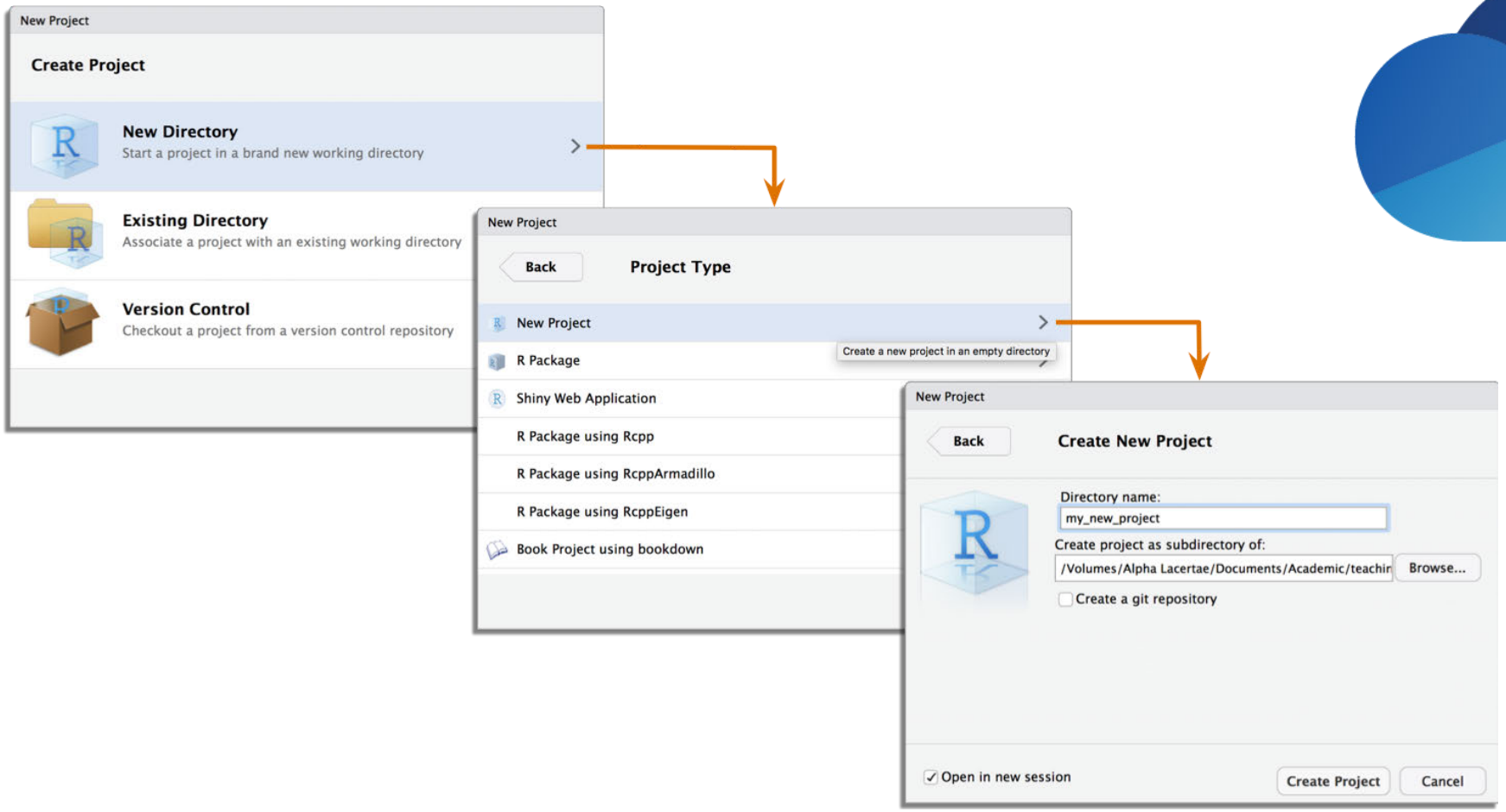
```
C:/Users/andyfield/Documents/my_new_project/data/my_funky_data.csv
```

```
./data/my_funky_data.csv
```

```
../data/my_funky_data.csv
```



Creating a project: File > New Project



Try it!

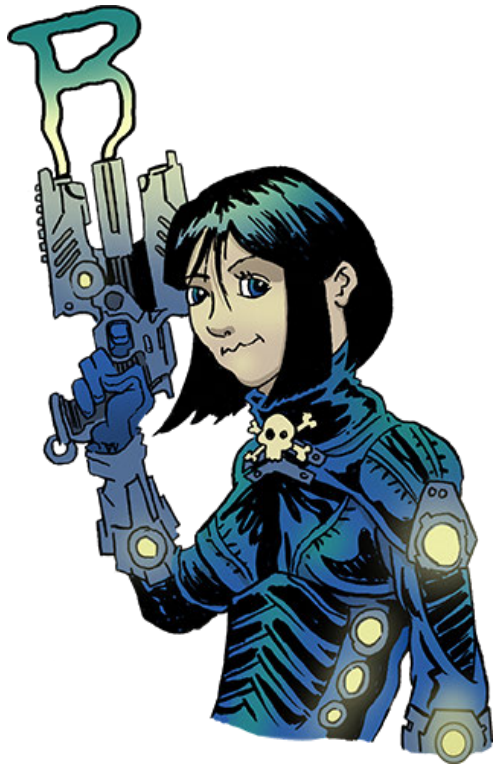
Create an RStudio project called my_adventr on Sussex One Drive

Within it create folders called

- data
- r_docs

Create a markdown file

- Use the File > New File > R Markdown ... menu
- Save it to the r_docs folder with the name sample_tap.Rmd



Now, let's save some data

On CANVAS go to Files > data_files

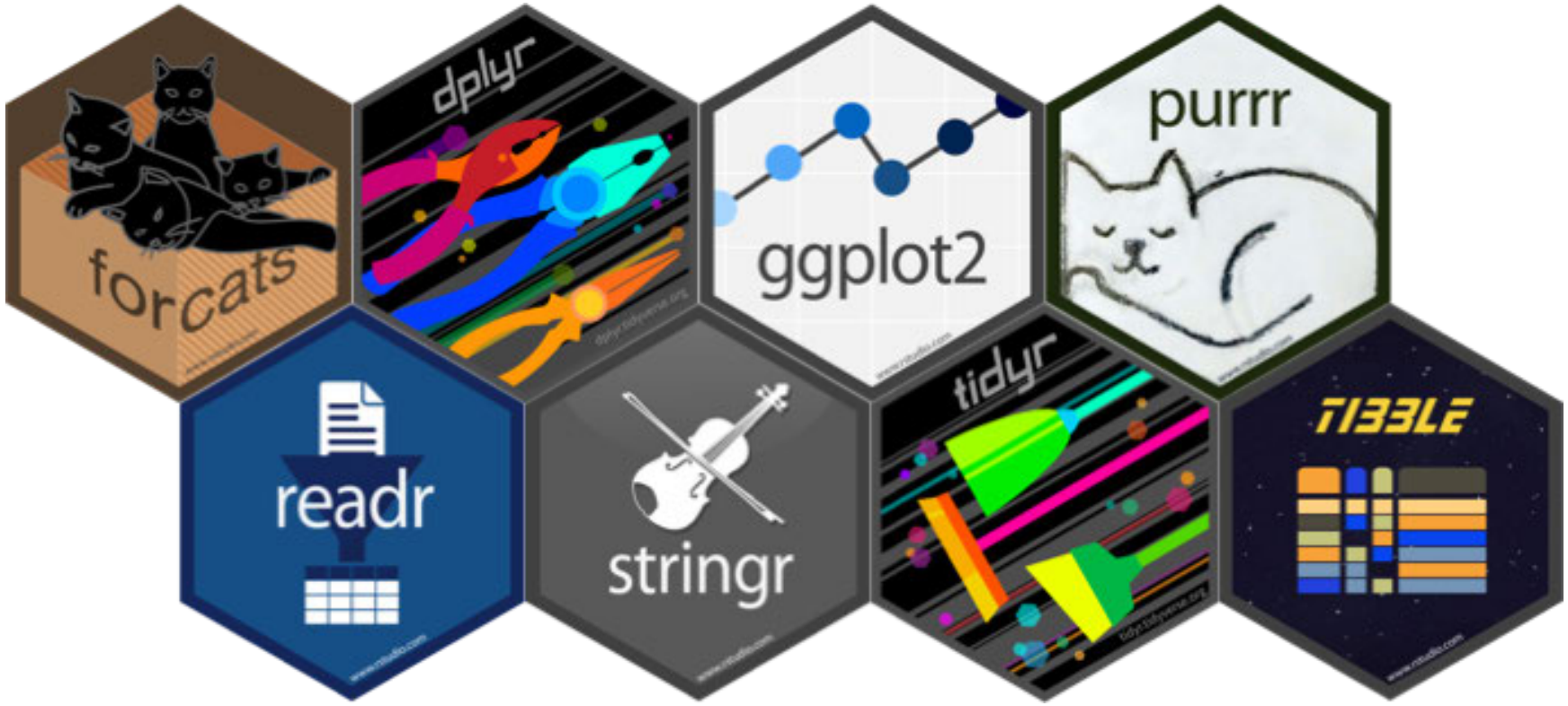
Download the sample TAP data (tap_parenting.csv)

Copy the file into the data folder of your project

Part 5: Tidyverse

The tidyverse

A set of packages built upon a common philosophy of data science



The pipe operator (%>%)

```
metallica
```

```
## [1] "Lars"  "James" "Kirk"  "Rob"
```

Multiple commands:

```
metalli_core <- subset(metallica, metallica != "Rob")
metalli_core <- sort(metalli_core)
```

Nested commands:

```
metalli_core <- sort(subset(metallica, metallica != "Rob"))
```

Piped commands:

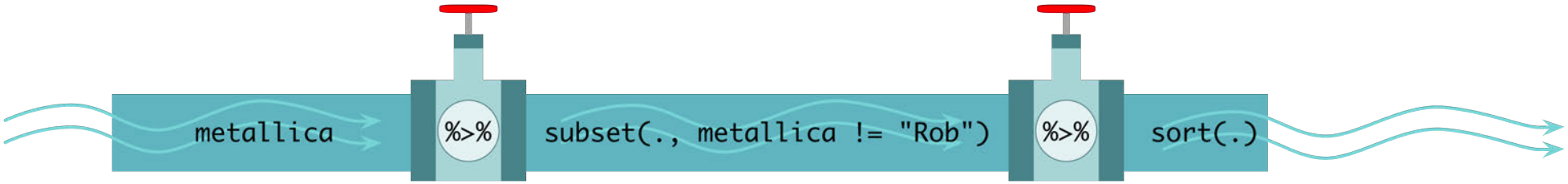
```
metalli_core <- metallica %>%
  subset(., metallica != "Rob") %>%
  sort(.)
```

Piped commands

```
metalli_core <- metallica %>%
  subset(., metallica != "Rob") %>%
  sort(.)

metalli_core
```

```
## [1] "James" "Kirk"  "Lars"
```



R Style (Wickham, 2014)

R is case sensitive. In code chunks use lower case wherever possible

Variable and function names should be lowercase with an underscore (_) to separate words

- Names should be concise and meaningful
- A variable representing children's' anxiety levels might be named `child_anxiety`
- `scores_on_the_child_manifest_anxiety_scale` is meaningful but too long, and `ca` is concise but not meaningful

Place spaces around all operators (`=`, `+`, `-`, `<-`) to make code easier on the eye!

My practice: use suffixes that identify types of objects. Some examples

- `_tib` to denote a tibble (more them later), e.g. `anx_tib` for a tibble of data relating to anxiety)
- `_mod` to denote a model (e.g., `anx_mod` for a model in which child anxiety is predicted)
- `_lm` to denote a model built using `lm()`
- `_out` to denote output (e.g., `anx_out` contains the summary output from the above model

Messy vs. Tidy data

Messy data (aka 'wide' data)

What IBM SPSS Statistics uses

- Each row represents a unique case/entity

Tidy data (aka 'long' data)

What many (but not all) R functions require

- Each row represents an instance of the outcome measure
- Rows code information about that 'instance'

Tidy data example

Are invisible people mischievous?

- Placed participants in an enclosed community riddled with hidden cameras
- Measured how many mischievous acts participants performed in a week
- Manipulated whether or not there was access to an invisibility cloak

Independent design

- 12 participants given an invisibility cloak
- 12 participants not given an invisibility cloak
- A random sample of *N* = 7 from each group is shown

id	cloak	mischief
Anupama	No cloak	1
Collin	No cloak	5
Darrell	No cloak	4
Kinaana	No cloak	5
Steven	No cloak	2
Tyler	No cloak	6
Vanessa	No cloak	6
Conan	Cloak	6
Devante	Cloak	7
Jerry	Cloak	5
Kathryn	Cloak	2
Sage	Cloak	3
Shajee'a	Cloak	4
Tamara	Cloak	8

Repeated measures design

- 12 participants given an invisibility cloak for one week
- During a different week they did not have an invisibility cloak
- A random subsample of *N* = 7 participants is displayed

id	cloak	mischief
Anupama	No cloak	1
Anupama	Cloak	3
Collin	No cloak	5
Collin	Cloak	6
Darrell	No cloak	4
Darrell	Cloak	5
Kinaana	No cloak	5
Kinaana	Cloak	5
Steven	No cloak	2
Steven	Cloak	4
Tyler	No cloak	6
Tyler	Cloak	5
Vanessa	No cloak	6
Vanessa	Cloak	8

```
tap_tib <- here::here("data/tap_parenting.csv") %>%
  readr::read_csv()

tap_tib
```

tap_parenting.csv

id	intervention	efficacy	did_bhv	id_school	time_cat	intention	time_num	positivity
aa54j	Group	7	1	school_7	baseline	28	0	1
aa54j	Group	7	1	school_7	1 month	46	1	5
aa54j	Group	7	1	school_7	6 months	29	6	2
ac09v	Group	4	1	school_6	baseline	54	0	5
ac09v	Group	4	1	school_6	1 month	48	1	5
ac09v	Group	4	1	school_6	6 months	59	6	5
ad17o	Leaflet	6	1	school_7	baseline	26	0	2
ad17o	Leaflet	6	1	school_7	1 month	28	1	5
ad17o	Leaflet	6	1	school_7	6 months	26	6	2
ad43a	Group	3	1	school_6	baseline	57	0	4

Part 6: Tips and good habits

Getting help

To get help, use the `help()` function or `?`

- `help(thing_you_want_help_with)`
- `?thing_you_want_help_with`

The R help files are frequently incomprehensible to mere mortals, but they are getting better!

> Tip: Execute help commands at the command line
>
> - Do NOT include `help()` of `?` in R markdown files

Try it!

Access the help files for the `mean()` function, by executing:

- `?mean`

Using a 'setup' code chunk

Use a setup code chunk that

- Sets global options for all of your code chunks
- Loads all of the packages you plan to use (in alphabetic order)
- Loads any data that you plan to use

```
knitr::opts_chunk$set(echo = TRUE, warning=FALSE, message=FALSE)

library(here)
library(kableExtra)
library(tidyverse)

tap_tib <- here::here("data/tap_parenting.csv") %>%
  readr::read_csv()
```

Part 7: Getting the most from practical classes

Practical classes

The practical classes are based on a package of interactive tutorials called `discovr` that I wrote

- You work at your own pace
- You can work with friends/peers to support each other
- Tutors will wander around giving you one-to-one help when you need it

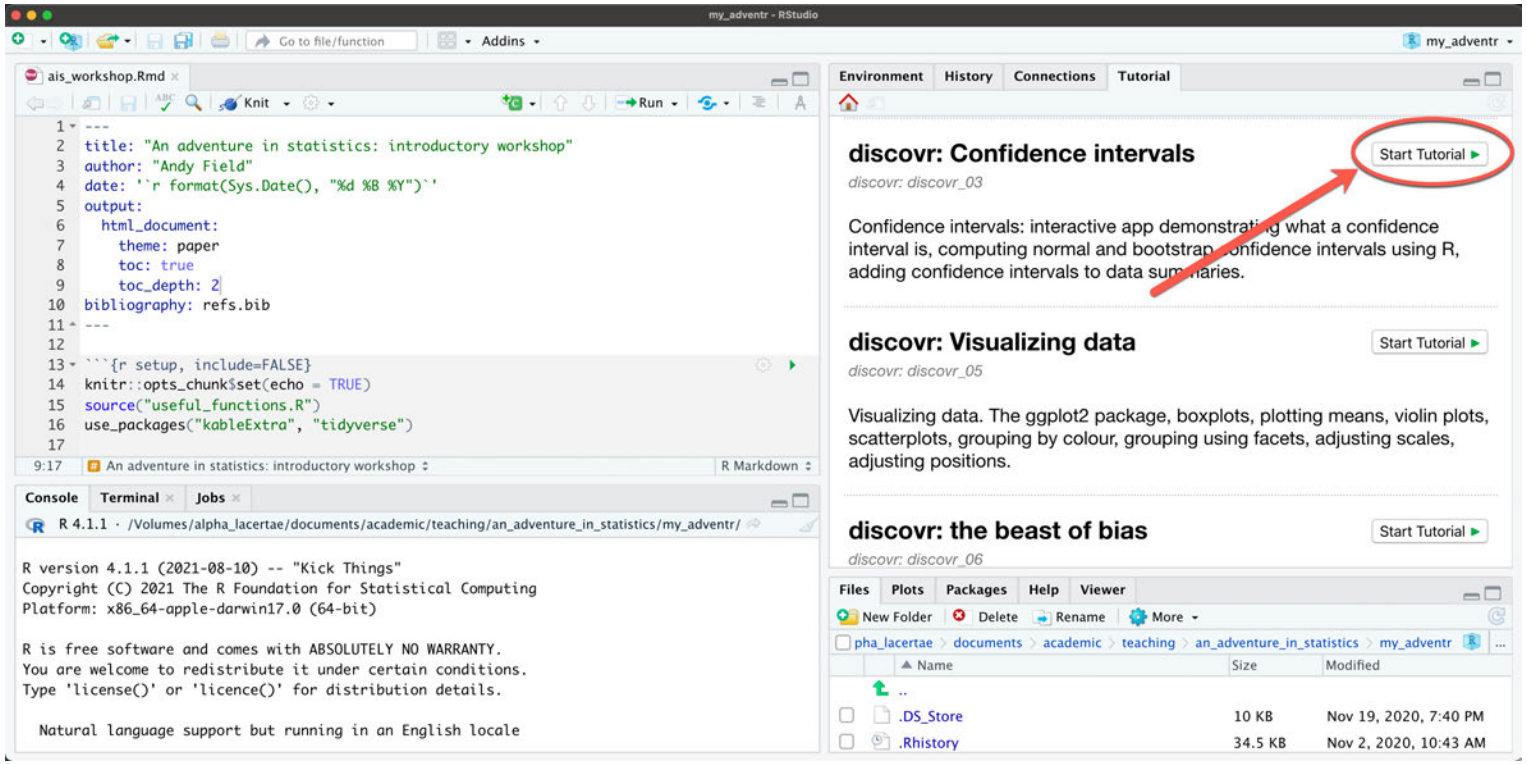
The discovr package

Installing discovr

```
if(!require(remotes)){
  install.packages('remotes')
}
remotes::install_github("profandyfield/discovr")
```

Running a tutorial

```
library(discovr)
learnr::run_tutorial("name_of_tutorial", package = "discovr")
learnr::run_tutorial("discovr_03", package = "discovr")
```



Suggested workflow

Create an RStudio project called my_adventr

- Within it create folders called data and r_docs
- Save all of the data files for the tutorials (on Canvas) into the data folder
- Save the file tutorial_template.Rmd into your r_docs folder

Run a tutorial and open it in a separate window

Create a learning journal for the tutorial

- Open the file tutorial_template.Rmd and save it with a name related to the tutorial
- As you work through the tutorial, copy the code you've written in the tutorial into code chunks in the R markdown file
- Make notes (for example, anything you didn't understand at first, or things to help you remember what you did and why you did it). This will help with your reflective statements
- Save the markdown file for future reference, and/or knit it into an html document

Watch the video at https://youtu.be/FhoYCsZttGc