STATS 220

Data technology

Kia Ora!

- 🎓 I earned my PhD (Stats) @ Monash University, Australia.
- ❤️ My research interests lie in exploratory data analysis, data visualisation, software design, ...
- 👩‍💻 I turn ☕ into > 10 `#rstats` 📦.
- Outside of work, I play 🎾 and make ☕.

Contact

- ✉️ earo.wang@auckland.ac.nz
- 📌 Office 303.323
- 🕑 Thursday 2-3pm

We're looking for Class Reps

Nominate yourself today and get:
An important and recognised addition to your resume // Improve your leadership skill set // Ability to make significant changes to your education // End of Year Function // At the end of the semester, you will be eligible to receive a class rep certificate provided you have registered with AUSA.

NB: The deadline to register your details is Friday of week two.

Responsibilities:
Elicit feedback from your classmates // Attend department and faculty meetings // Help resolve issues that may arise in your class.

For more info visit www.ausa.org.nz/support/class-reps/
email classreps@ausa.org.nz or speak to your lecturer

ausa

FAIR • SUPPORTIVE
INDEPENDENT • CONFIDENTIAL

advocacy

We offer advice about your rights, university procedures, tenancy and more.

ausa advocacy OLD CHORAL, ALFRED ST, CITY CAMPUS
09 923 7294 WWW.AUSA.ORG.NZ ADVOCACY@AUSA.ORG.NZ

Data + Technology

➔ https://stats220.earo.me

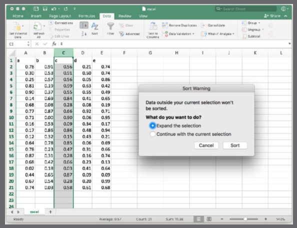
What I mean by "data"

🥫	🍅
✗ Stale, uninteresting, convenient	✓ Fresh, interesting, challenging
✗ Highly processed and archived	✓ Locally collected and impactful
✗ Example: `student tests`, `titanic`, `wages`	✓ Example: Modelling the travel time of transit vehicles in real-time

How I learn new technology

- 🗣 **Get hands dirty**!!
- 📖 Documentation! Documentation! Documentation!
- 🔍 (Not surprisingly) Learn to google: what that error message means (I google a lot 😶)

You can't do data science in a GUI



	A	B	C	D	E
1	a	b	c	d	e
2	0.78	0.91	0.56	0.21	0.74
3	0.30	0.53	0.91	0.50	0.74
4	0.25	0.57	0.56	0.05	0.86
5	0.81	0.19	0.99	0.63	0.42
6	0.90	0.37	0.55	0.55	0.49
7	0.14	0.69	0.84	0.41	0.65
8	0.68	0.08	0.28	0.08	0.19
9	0.77	0.87	0.66	0.92	0.71
10	0.71	0.00	0.90	0.06	0.95
11	0.16	0.53	0.09	0.34	0.17
12	0.17	0.86	0.86	0.48	0.94
13	0.12	0.32	0.15	0.43	0.21
14	0.64	0.78	0.85	0.06	0.69
15	0.78	0.23	0.47	0.31	0.66
16	0.87	0.31	0.28	0.16	0.74
17	0.68	0.42	0.66	0.23	0.13
18	0.02	0.18	0.03	0.41	0.64
19	0.44	0.65	0.87	0.09	0.09
20	0.67	0.54	0.28	0.20	0.99
21	0.74	0.03	0.58	0.61	0.68

Sort Warning

Data outside your current selection won't be sorted.

What do you want to do?

[✓ Expand the selection]
Continue with the current selection

reference: You can't do data science in a GUI

Why programme for data science?

- Programming languages are **languages**.

```
library(dplyr)
starwars %>%
  group_by(species) %>%
  summarise(
    n = n(),
    mass = mean(mass, na.rm = TRUE)
  ) %>%
  filter(n > 1, mass > 50)
```

- It's just **text**!
  - reproducible, readable, sharable
  - expressive

Why R

- A general-purpose programming language
- Originated by statisticians, a language for statistical analysis
- 292995 + packages on CRAN (Comprehensive R Archive Network, the official repository), Github, etc.
- The tidyverse, a domain specific language in R for data scientists

What R can do?

- for fun

📦 {cowsay} for generating ASCII picture

```
library(cowsay)
say("Kia Ora!")
```

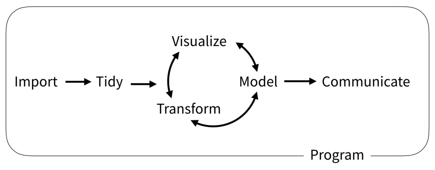
```
#>
#>  --------------
#> Kia Ora!
#>  --------------
#>     \
#>       \
#>         \
#>             |\___/|
#>           ==) ^Y^ (==
#>             \  ^  /
#>              )=*=(
#>             /     \
#>             |     |
#>            /| | | |\
#>            \| | |_|/\
#>       jgs  //_// ___/
#>                \_)
#>
```

What R can do?

- for fun

- for data

The data science workflow



tidyverse = readr tidyr dplyr ggplot2 purrr

What R can do?

- for fun
- for data
- **for communication**

R Markdown

> {rmarkdown} for assignments/reports/papers in `.html` and `.pdf`
> {blogdown} for blogs
> {bookdown} for books
> {xaringan} for slides (220 slides!)

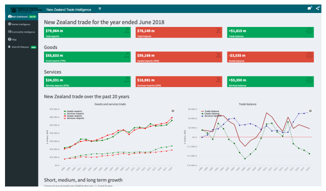
R Markdown documents are fully reproducible: weaving narrative text and code together.

What R can do?

- for fun
- for data
- **for communication**

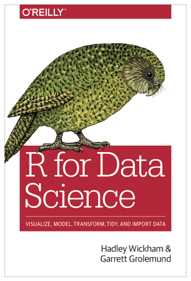
R shiny dashboard

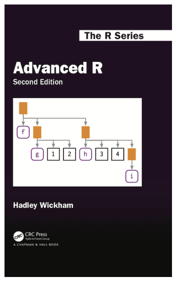
> Shiny is an R package that makes it easy to build interactive web apps straight from R.


New Zealand Trade Intelligence
New Zealand trade for the year ended June 2018
Goods
Services
New Zealand trade over the past 20 years
Short, medium, and long term growth

👆 click the image above will take you to the web app, and try to interact with the app.

Textbook 📚


O'REILLY
R for Data Science
VISUALIZE, MODEL, TRANSFORM, TIDY, AND IMPORT DATA
Hadley Wickham & Garrett Grolemund



At first, you may be like this...

But you can do it!

Assessments

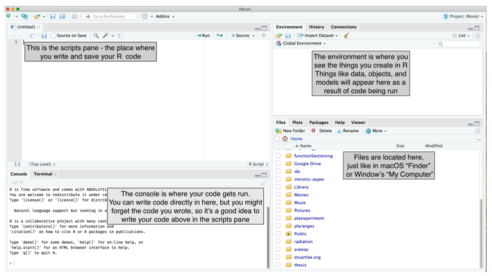
- 11 weekly labs 10% (best 10 out of 11)
- 3 assignments 30% (each 10%)
- 1 mid-term test 10% (TBD, possibly week 8)
- 1 final exam 50%

Project-oriented workflow

R Studio

> *If R were an airplane, RStudio would be the airport, providing many, many supporting services that make it easier for you, the pilot, to take off and go to awesome places. Sure, you can fly an airplane without an airport, but having those runways and supporting infrastructure is a game-changer.*
> -- *Julie Lowndes*

RStudio interface



This is the scripts pane - the place where you write and save your R code

The environment is where you see the things you create in R Things like data, objects, and models will appear here as a result of code being run

Files are located here, just like in macOS "Finder" or Window's "My Computer"

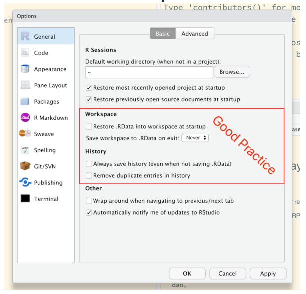
The console is where your code gets run. You can write code directly in here, but you might forget the code you wrote, so it's a good idea to write your code above in the scripts pane

```
R is free software and comes with ABSOLUTEL
You are welcome to redistribute it under ce
Type 'license()' or 'licence()' for distrib

  Natural language support but running in a

R is a collaborative project with many cont
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

>
```

image credit: Stuart Lee

Setting up RStudio (do this once)

Go to **Tools** > **Global Options**:



R Sessions

Default working directory (when not in a project): ~

[✓] Restore most recently opened project at startup

[✓] Restore previously open source documents at startup

Workspace

[ ] Restore .RData into workspace at startup

Save workspace to .RData on exit: Never

History

[ ] Always save history (even when not saving .RData)

[ ] Remove duplicate entries in history

Other

[ ] Wrap around when navigating to previous/next tab

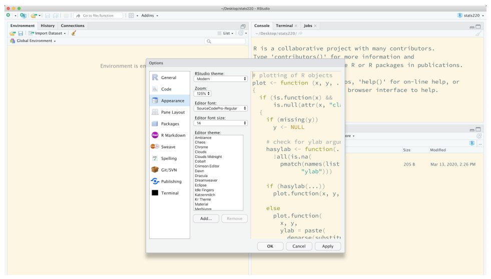
[✓] Automatically notify me of updates to RStudio

Uncheck `Workspace` and `History`, which helps to keep R working environment fresh and clean every time you switch between projects.

Your turn

01:00

Change the RStudio appearance up to your taste



What is a project?

- Each university course is a project, and get your work organised.
- A self-contained project is a folder that contains all relevant files, for example my `stats220/` 📁 includes:
  - `stats220.Rproj`
  - `data/`
    - `*.csv`, `*.xlsx`
  - `lectures/`
    - `01-intro.Rmd`, `02-import-export.Rmd`
  - `labs/`
    - `lab01.R`, `lab02.R`
- All working files are relative to the **project root** (i.e. `stats220/`).
- The project should just work on a different computer.

🛑 STOP DOING THIS!

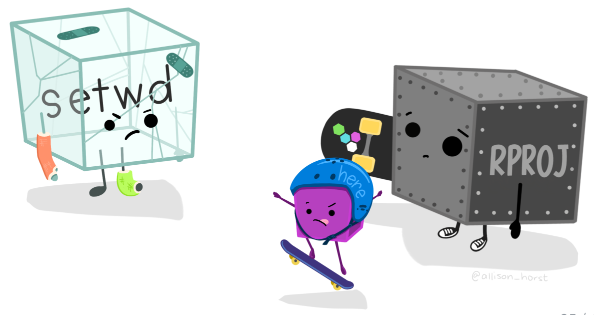
Jenny Bryan will set your computer on fire 🔥

1. if the first line of your R script is

```
setwd("C:\Users\jenny\path\that\only\I\have")
```

2. if the first line of your R script is

```
rm(list = ls())
```

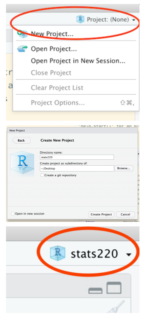


Create an RStudio project `.Rproj`

1. Click the **Project** icon on the top right corner

2. **New Directory/Existing Directory** > **New Project** > **Create Project**

3. Open the project



101: syntax and semantics

Get started
- assignment

```
akl_lon <- 174.76
akl_lat <- -36.85
```

⬆️ read as "assign the value of `174.76` to an object called `akl_lon`".

An assignment consists of:

- left-hand side: variable names or symbols (`akl_lon`)
- assignment operator: `<-` (RStudio shortcut: `Alt` + `-`)
- right-hand side: values (`174.76`)

Get started
- assignment
- **retrieval**

```
akl_lon
```

```
#> [1] 174.76
```

```
akl_lat
```

```
#> [1] -36.85
```

- Names are case sensitive.

```
akl_Lon
```

```
#> Error in eval(expr, envir, enclos): object 'akl_Lon' not found
```

Get started
- assignment
- retrieval
- **operation**

Perform calculations and comparisons

- Infix operators:
  - `+`, `-`, `*`, `/`, `^`, `%%` (modulo), `%/%` (integer division)
  - `==`, `!=`, `>`, `<`, `>=`, `<=`, `%in%`

```
akl_lon_region <- akl_lon + c(-1, 1)
akl_lat_region <- akl_lat + c(-.5, .5)

akl_lon_region
```

```
#> [1] 173.76 175.76
```

```
akl_lat_region
```

```
#> [1] -37.35 -36.35
```

Coding style

> *Good coding style is like correct punctuation: you can manage without it, butitsuremakesthingseasiertoread.*
> -- *The tidyverse style guide*

R style guide

- ✓ `snake_case`
- ✗ `camelCase` (Javascript)
- ✗ `PascalCase` (Python)

R 101: data structures

Atomic vectors

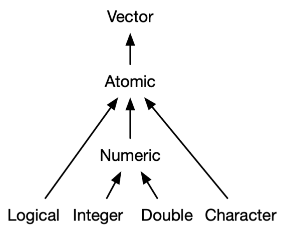


*image credit: Hadley Wickham's **Advanced R***

Scalars: ***length of 1***

- Logicals: `TRUE` or `FALSE`
- Doubles: `174.76`, `1.7476e2`, `Inf`, `-Inf`, `NaN` (Not a Number)
- Integers: `174L`
- Strings: `"hello"`, `'world'`

Vectors: ***values must all be the same type***

```
lgl_vec <- c(TRUE, FALSE)
int_vec <- c(174L, -36L)
dbl_vec <- c(174.76, -36.85)
chr_vec <- c("long", "lat")
```

Special values

Missing values

```
NA # Not Applicable
```

```
#> [1] NA
```

```
c(174.76, NA, -36.85)
```

```
#> [1] 174.76     NA -36.85
```

```
length(NA)
```

```
#> [1] 1
```

The `NULL` object

```
NULL
```

```
#> NULL
```

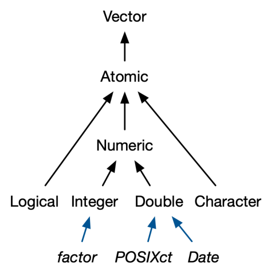
```
c(174.76, NULL, -36.85)
```

```
#> [1] 174.76 -36.85
```

```
length(NULL)
```

```
#> [1] 0
```

Atomic vectors



Subsetting vectors with `[]`

```
x <- c(akl_lon_region, akl_lat_region)
x
```

```
#> [1] 173.76 175.76 -37.35 -36.35
```

Positive indices

```
x[c(1, 3)]
```

```
#> [1] 173.76 -37.35
```

Negative indices

```
x[-c(3, 1)]
```

```
#> [1] 175.76 -36.35
```

Subsetting vectors with `[]`

Logical indices

```r
x[c(TRUE, FALSE, TRUE, FALSE)]
```

```
#> [1] 173.76 -37.35
```

```r
x[lgl_vec] # recycling
```

```
#> [1] 173.76 -37.35
```

```r
x[x > 0]
```

```
#> [1] 173.76 175.76
```

Special subsetting

```r
x[0]
```

```
#> numeric(0)
```

```r
x[]
```

```
#> [1] 173.76 175.76 -37.35 -36.35
```

Modifying vectors with `[]` on the LHS

```r
y <- x
y
```

```
#> [1] 173.76 175.76 -37.35 -36.35
```

```r
y[1:3] <- y[1:3] %/% 2
y
```

```
#> [1]  86.00  87.00 -19.00 -36.35
```

- RHS `[]` subsets vector `y`
- LHS `[]` modifies vector `y`

R 101: functions

Function

A function call consists of the function name followed by one or more argument within parentheses.

```r
mean(x = x)
```

```
#> [1] 68.955
```

- function name: `mean()`, a built-in R function to compute mean of a vector
- argument: the first argument (LHS `x`) to specify the data (RHS `x`)

Function help page

01:00

Check the function's help page with `?mean`

```
mean(x, trim = 0, na.rm = FALSE, ...)
```

- Read **Usage** section
  - What arguments have default values?
- Read **Arguments** section
  - What does `trim` do?
- Run **Example** code

Function arguments

Match by **positions**

```
mean(x, 0.1, TRUE)
```

```
#> [1] 68.955
```

Match by **names**

```
mean(x, na.rm = TRUE, trim = 0.1)
```

```
#> [1] 68.955
```

Use functions from packages

```
# install.packages("dplyr")
library(dplyr)
cummean(x)
```

```
#> [1] 173.7600 174.7600 104.0567  68.9550
```

```
first(x)
```

```
#> [1] 173.76
```

```
last(x)
```

```
#> [1] -36.35
```

Images sourced from https://www.wikihow.com/Change-a-Light-Bulb

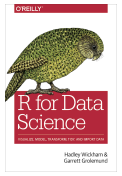
Write your own functions

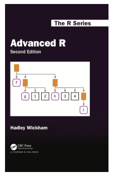
```
# function_name <- function(arguments) {
#   function_body
# }

my_mean <- function(x, na.rm = FALSE) {
  summation <- sum(x, na.rm = na.rm)
  summation / length(x)
}

my_mean(x)
```

```
#> [1] 68.955
```

Follow the `#rstats` community

Reading





- Workflow: basics
- Workflow: scripts
- Workflow: project

- Names and values
- Vectors
- Subsetting