

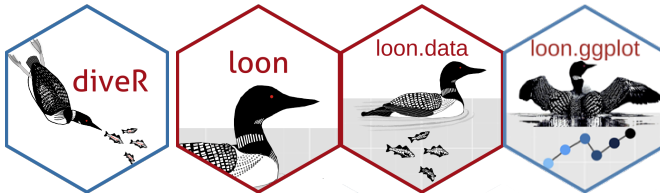
Interactive Exploratory Data Visualization in R



Maynooth University
National University
of Ireland Maynooth



A workshop introduction to loon and related packages



Wayne Oldford
May 12, 2021



Making this work

1. You should already have your machine set up as described in the preparations

- ▶ Most recent R and RStudio and all necessary packages installed and tested
- ▶ A mouse/trackpad with two buttons and scrolling enabled
- ▶ ideally two screens:
 - ▶ one for my presentation
 - ▶ one for your work in RStudio (the larger of your two screens)
- ▶ a microphone and headphones/earbuds
- ▶ if your machine is not working, there is not much we can do about it today

2. Communication

- ▶ I (the presenter) will be speaking and sharing my screen
- ▶ audio/video communication is mostly one way:
 - ▶ this may be the biggest challenge (100 registrations)
 - ▶ everyone (I think) joins muted so unmute yourself if called on
- ▶ there is a text chat facility (public and private)
 - ▶ you can “raise your hand”, make it urgent (“!”), and give some canned comments from a menu (“Yes”, “No”, “Too fast”, “Applause”, . . .)
 - ▶ I will try to pay attention to this but forgive me if I overlook a comment/question,,
 - ▶ Zehao Xu (Waterloo PhD student and author of a significant part of today’s software) will also be able to answer questions (hopefully)



What is going to happen?

Mainly I will be leading you through two R scripts

- ▶ You should have these downloaded already
- ▶ I will be sharing my screen throughout
 - ▶ sharing the code onscreen and explaining it as we go
 - ▶ encouraging you to try stuff and waiting for you to explore
 - ▶ showing the results when I run the code
 - ▶ showing my interactions with `loom` and with `RStudio`
- ▶ So mainly, I will be following you
 - ▶ which makes pacing a challenge
 - ▶ PLEASE stop me if/when you have a question. Guaranteed you will not be the only person with that question.

I will also occasionally switch to slides like these.



Rough schedule

Part 1 Introducing interactive visualization (approx 1 hour)

- ▶ 1. covidNZ.R will be the script covered
- ▶ this is where the major functionality of loon will be demonstrated on a topical dataset.

Break: There will be a 15 minute break after about an hour of covidNZ

Part 2: An interactive grammar of graphics (approx 1 hour)

- ▶ finish the covidNZ analysis if needed
- ▶ 2. airquality.R will be the script covered
- ▶ this part is primarily for those users who work with ggplot2
- ▶ shows how to easily create **interactive ggplots**
- ▶ extends ggplot2's grammar of graphics

Questions?



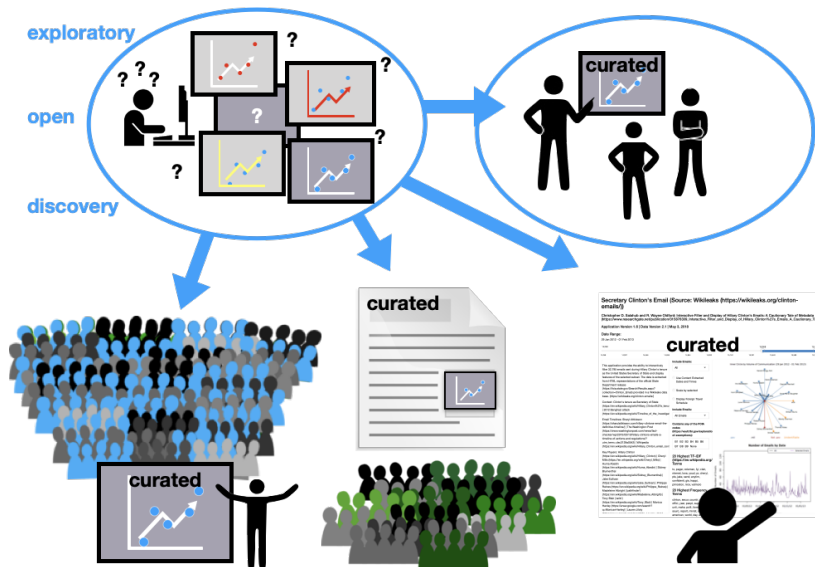
Homework

Part 3: Extra material

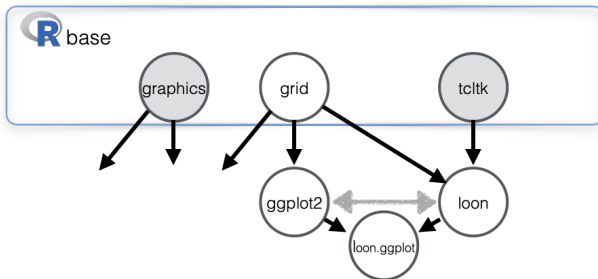
- ▶ the directory Extra contains a number of files for you to learn more
 - ▶ `moreInformation.R` contains pointers to accessing more material
(e.g., includes how to save the state of loon plots to be later recovered)
 - ▶ `loon.ggplot.R` more detail and examples on how loon and ggplot2 work together
 - ▶ `plots3D.R` more examples and detail on exploratory analyses using three dimensional plots
 - ▶ `higher_dimensional.R` for visualizing more than 3 dimensions
 - ▶ `pipes.R` using loon with pipes (`magrittr`) and an interesting Canadian data set on minority distributions in cities across Canada. This also illustrates some high-dimensional methods.
 - ▶ `teachingDemo.R` illustrates some visualizations that can be created for teaching.



Exploratory visualization



Graphics systems in R



R ships with three fundamental graphics packages on which most graphics packages are built:

- ▶ **graphics** ... simple but powerful traditional base graphics
 - ▶ e.g. `hist()`, `plot()`, `pairs()`
- ▶ **grid** ... powerful classically designed computer graphics system
 - ▶ coordinate systems, viewports, ...
- ▶ **tcltk** ... a classic widget packing system
 - ▶ widgets, events, sliders, ... interactive

E.g. `ggplot2` is built on `grid`, `loon` on `tcltk`. One complements the other.



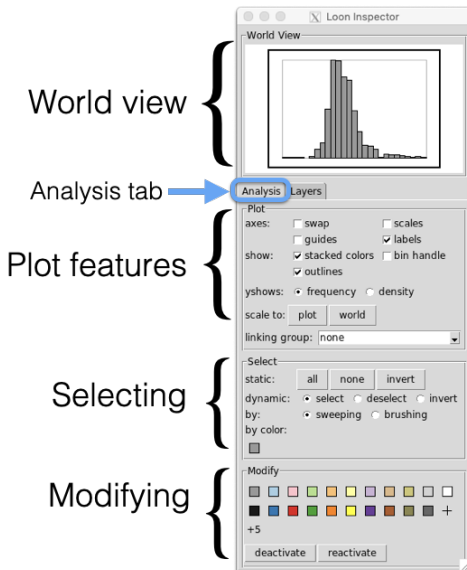
Covid-19 cases in New Zealand

Let's get started!

Open up the file 1. covidNZ.R



The loon inspector

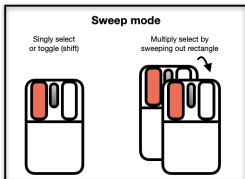


Mouse movements - selection

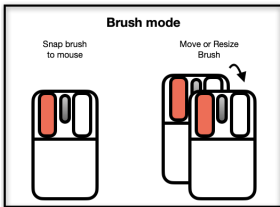
Selecting

SELECTION

SHIFT makes selection "sticky"

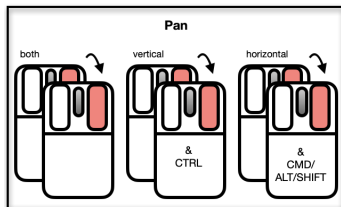


Brushing

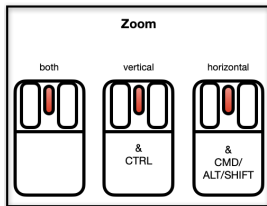


Mouse movements - panning and zooming

Panning

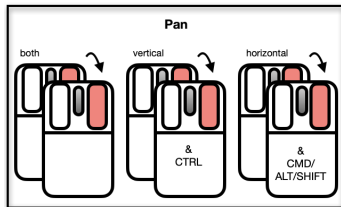


Zooming



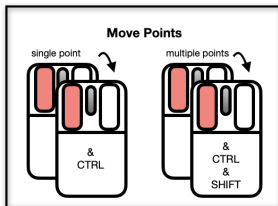
Mouse movements - moving

Panning is like moving the background



Moving points (scatterplots)

SCATTERPLOT



Back to 1. covidNZ.R



Interactive query logic

Classic logical queries:

- ▶ NOT A
 - ▶ select A and invert – everything selected is NOT A
 - ▶ select A and deactivate A – everything remaining (active) is NOT A
- ▶ A OR B
 - ▶ shift-select A and B – everything selected is A OR B
 - ▶ shift-select A and B, invert and deactivate – everything remaining is A OR B
- ▶ A AND B
 - ▶ select A, invert, deactivate not A; select B in what remains (or B then A)
 - ▶ everything selected is A AND B
 - ▶ do as above, then invert and deactivate. Remaining is A AND B

The result of the query is either - what remains selected, or - what remains active

For more complete information on interactive logical queries see

```
library(loon)
l_web(page = "logicalQueries", directory = "articles", package = "loon")
```



Back to '1. covidNZ.R'



BREAK

End of Part 1?

Back in an 15 minutes!!



A grammar of **interactive** graphics

Beginning Part 2

Open up the file '2. airquality.R'



Credits

loon has had a lot of person-years work into it:

- ▶ began with Adrian Waddell's 2016 PhD thesis (though we went through several design iterations in the preceding years)
- ▶ continues with Zehao Xu's current PhD thesis (loon, loon.ggplot, ggmulti, loon.tourr, loon.shiny)
- ▶ other named contributors are Martin Gauch (l_plot3D)

Many undergraduate and graduate students have also contributed in their own way, not least being the guinea pigs for testing code!

Some students might also be interested in my 2017 Data Visualization lectures on YouTube: <https://youtu.be/Khj94kaJjWo> (you can at least fast-forward through them!)



Fin

