A battle plan

DEFENSIVE R PROGRAMMING

Dr. Colin Gillespie
Jumping Rivers

Starting small

> Great results, can be achieved with small forces.
>
> Sun Tzu, **The Art of War**

What's in a (file)name?

- All R scripts are stored in files
  - So filenames are important
  - Consistency in filenames is very important
    - What sort of rules could we have?

Multiple words

Filenames often contain multiple words

For example

- `cluster-analysis.R`
- `load-survival-data.R`
- `plot-residuals.R`

Spaces, dashes or underscores?

Simple question. How should words be separated?

- Space: `analysis clustering.R`
- Underscores: `analysis_clustering.R`
- Dashes: `analysis-clustering.R`

Take a second and answer these two questions

- Which do you use?
- What **should** you use?

Spaces in filenames

- Don't use them
- Really, just don't

Spaces in filenames and directories are a bad idea

- If you put the file on the web `file name.R` becomes `file%20name.R`
- On the command line harder filenames with spaces need to be surrounded by quotes
- Regular expressions are also more painful
- Spotting the difference between
  - `file name.R` - one space
  - `file name.R` - two spaces is hard

Dashes or underscores

There are a few **minor** problems with underscores

- Google treats `file_name` as a single word
  - So searching for just `file` won't work
- The regular expression character `\w` treats `_` as a character

The same problems don't apply to dashes

Confession time:

- I usually use underscores
- but I'm trying to change

Let's have some practice

DEFENSIVE R PROGRAMMING

Human Readable Filenames

DEFENSIVE R PROGRAMMING

Dr. Colin Gillespie
Jumping Rivers

The humble slug

> URL slugs are the end part of a web address

Which URL do you prefer?

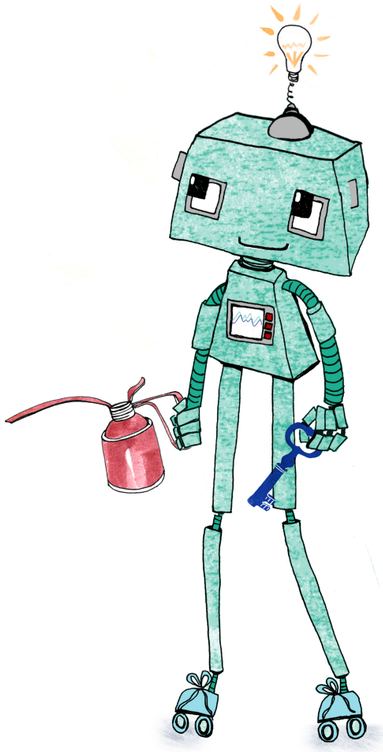
www.datacamp.com/courses/course1963.htm

We can learn from slugs

- Use sensible names
  - `ac.R` or `analysis-clustering.R`
  - `1.R` or `loading.R`
    - Be consistent
    - Use the same file extension - `.R`
    - Always lower case

Dates - what do we want?

- Unambiguous
  - So not
    - 01/02/032
  - Sortable in a file system



Dates - ISO8601

Dates should be

> YYYY-MM-DD

- All dates are now in an obvious and natural order
- Sorting just works!

```
2017-01-02
2018-01-01
2018-01-02
```

Numbers are good

For this course, I created directories called

- `chapter01`
- `chapter02`

Simple, yet effective

Let's practice!

DEFENSIVE R PROGRAMMING

Organizing a project

DEFENSIVE R PROGRAMMING

Dr. Colin Gillespie
Jumping Rivers

It starts with something small

All R analyses start with a little code, but then

- 1 line becomes 10
- 1 imported package becomes 5
- 1 file becomes a mess

Project Set-up

Every project I work on

- Has its own directory
- Has a sensible name

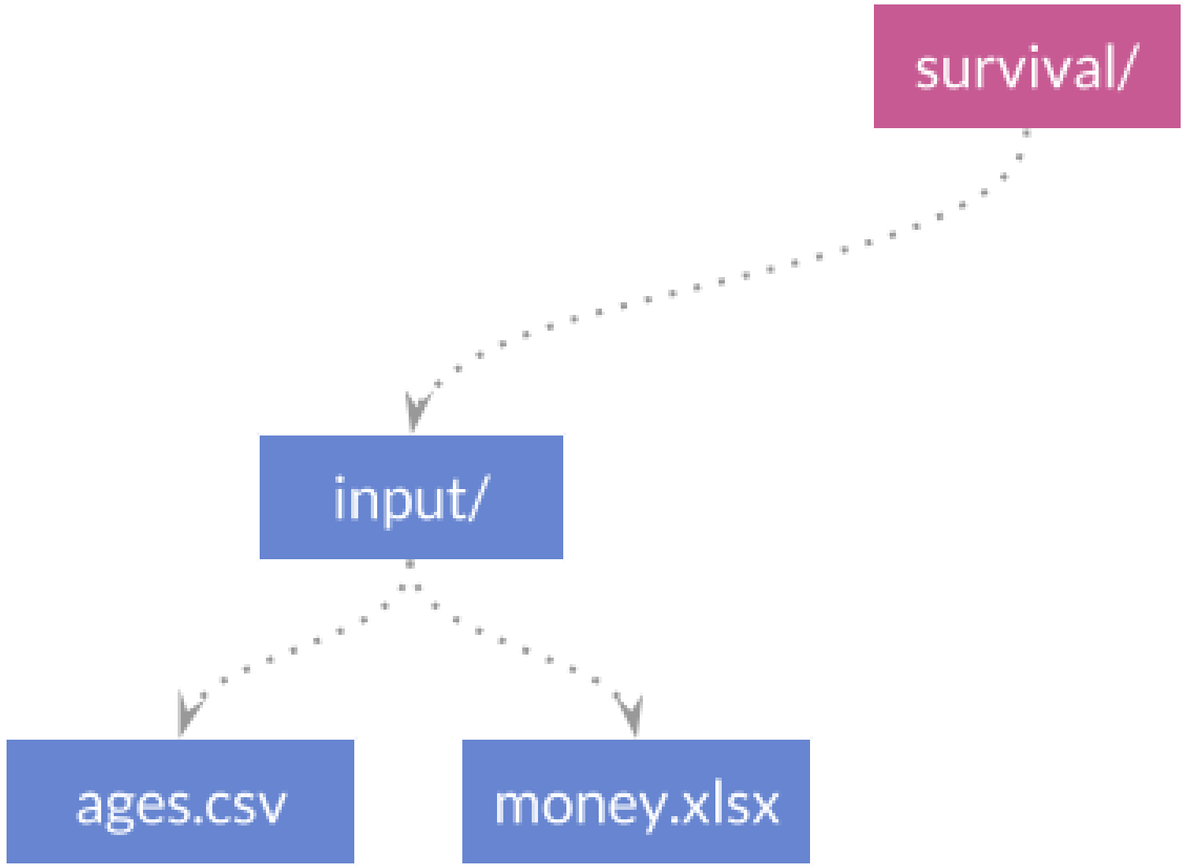
The directory name gives the **context** of the scripts

Directory: input/

This directory contains data, typically

- `csv` & excel files
- No R code

Data is only edited in R



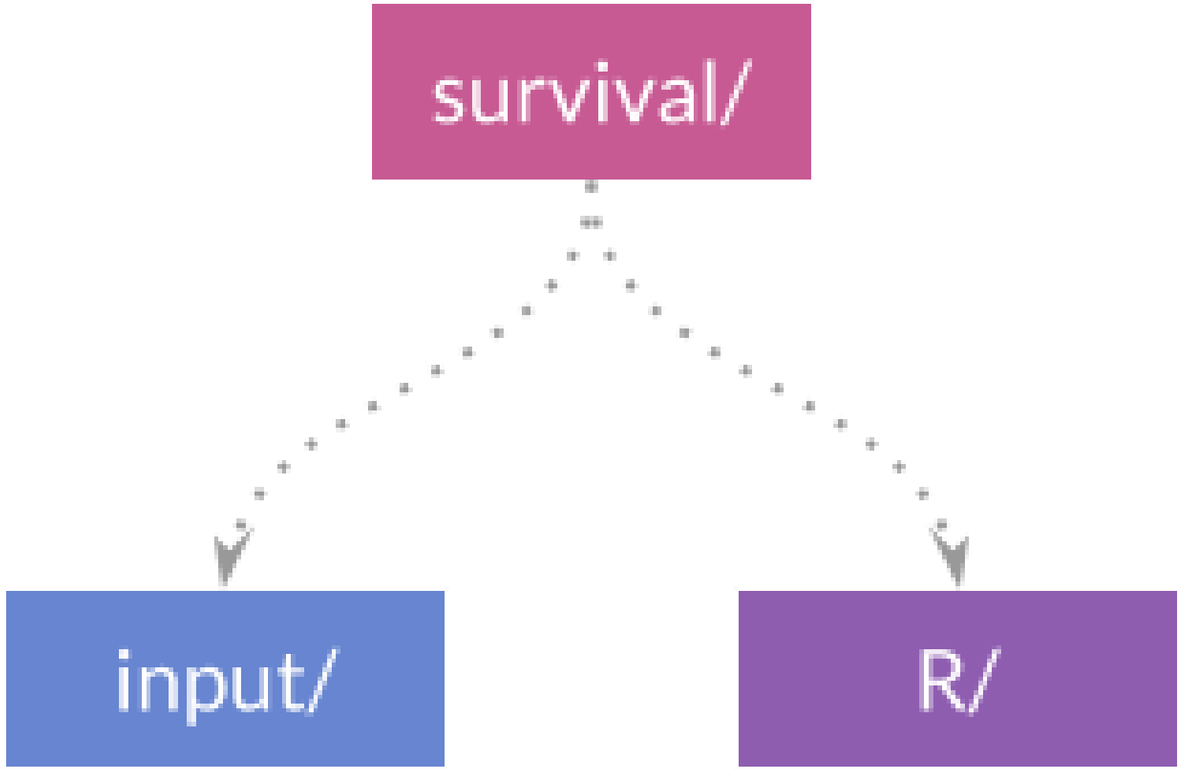
Directory: R/

All R code lives in this directory

Notice The directory isn't

- R_analysis
- R_code
- R_survival

just plain **R/**



Directory: R/

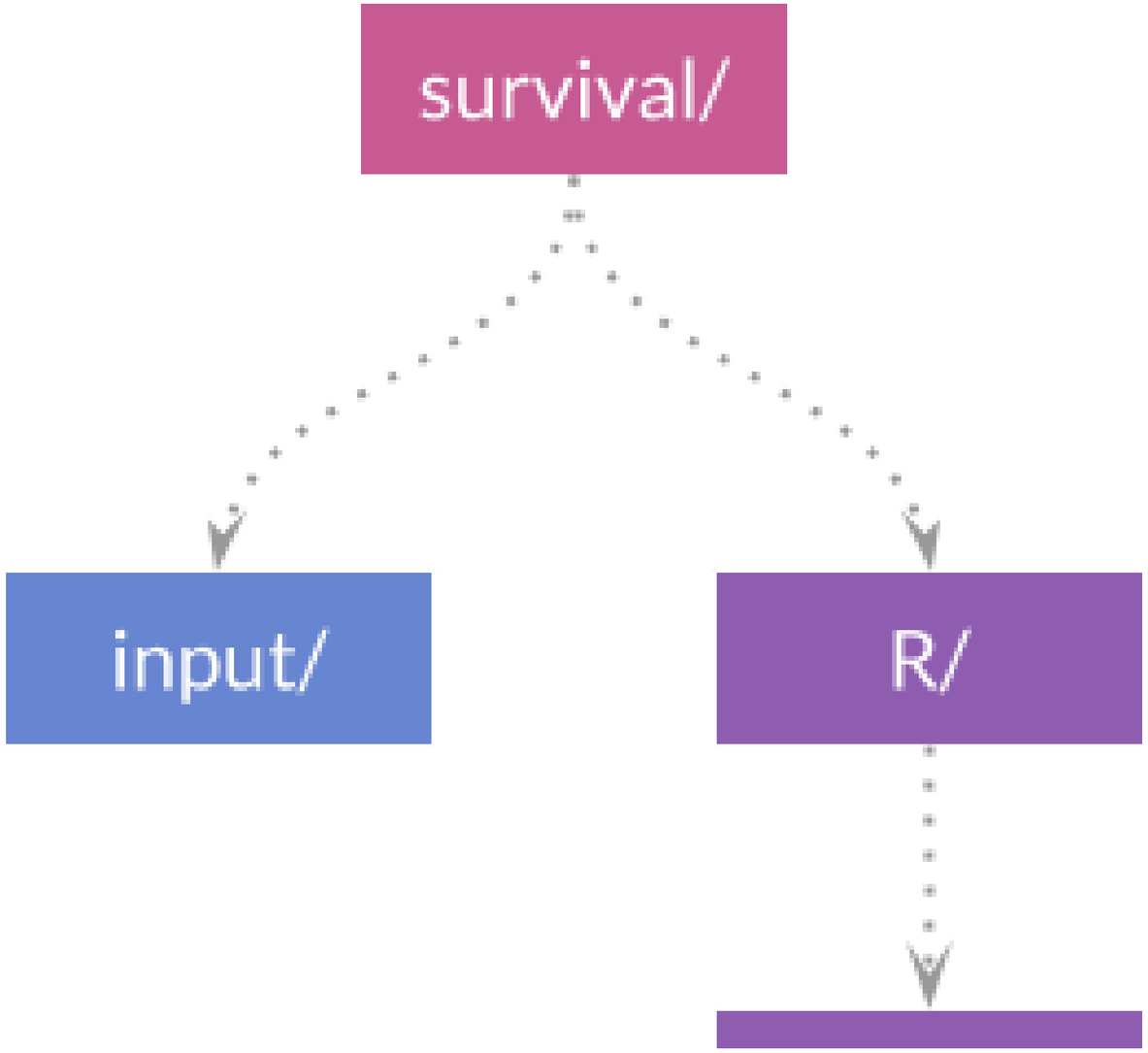
In this directory, I always have a file called

- `load.R`

This file loads the data from `input/`

Every project I've worked has a similar structure

- I can give you any project and you can load the data

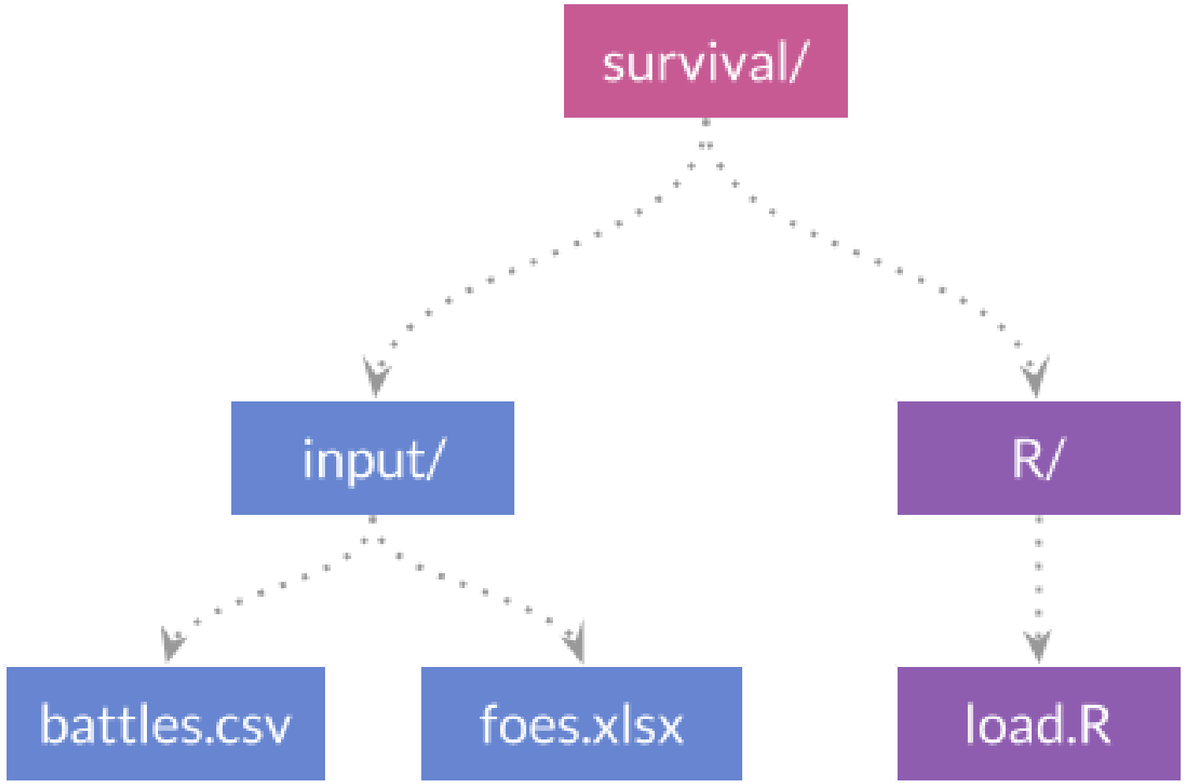


The load.R file

All paths are relative

```
battles <- read_csv("input/battles.csv")
foes <- read_xlsx("input/foes.xlsx")
```

My code is portable



Other R files

Remember, all R files live in the R directory!

- `clean.R` - for cleaning your data
- `function.R` - any helper functions
- `analysis.R` - the actual analysis

Standard names used in every project

Your turn

DEFENSIVE R PROGRAMMING

Graphics and Output

DEFENSIVE R PROGRAMMING

Dr. Colin Gillespie
Jumping Rivers

Project overview 1

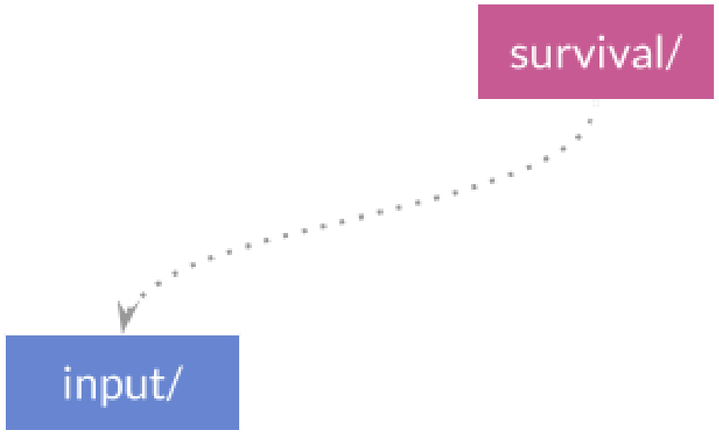
So far we have encountered

- the base project directory

Project overview 2

So far we have encountered

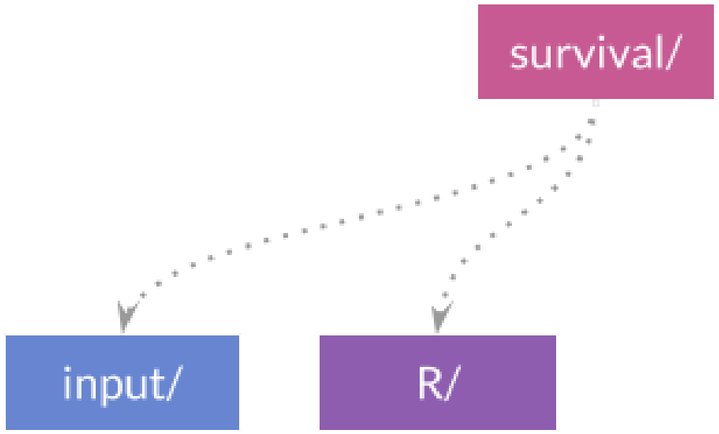
- the base project directory
- `input/` for data files



Project overview 3

So far we have encountered

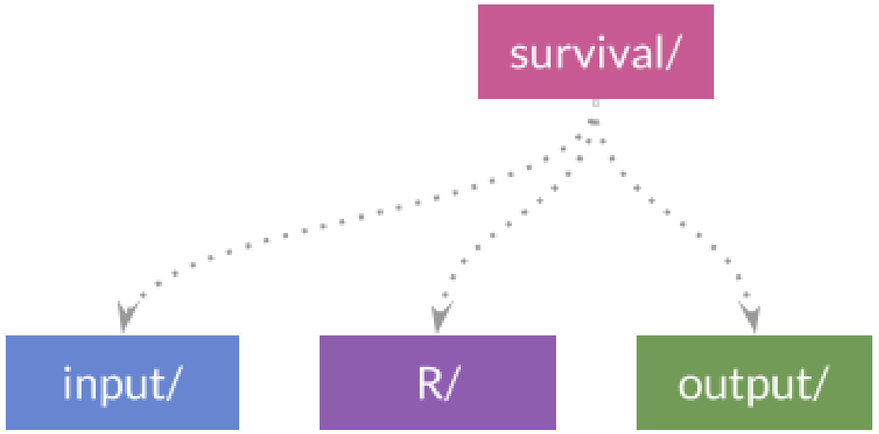
- the base project directory
- `input/` for data files
- `R/` for R scripts



Project overview 4

So far we have encountered

- the base project directory
- `R/` for R scripts
- `input/` for data sets



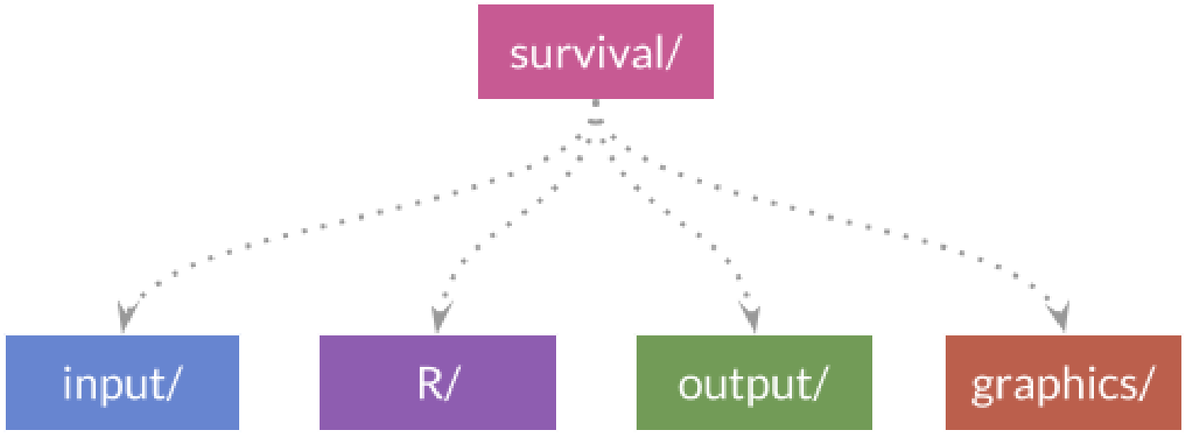
In this last video, we'll look at

- `output/` for output **generated** data files

Project overview 5

So far we have encountered

- the base project directory
- `R/` for R scripts
- `input/` for data sets



In this last video, we'll look at

- `output/` for output **generated** data files
- `graphics/` for **generated** plots

The difference

The scripts in

- the `R/` directory

create the contents of

- `output/` & `graphics/`

So in theory, we can delete `output/` & `graphics/` and not cry

The graphics/ directory

This directory just contains

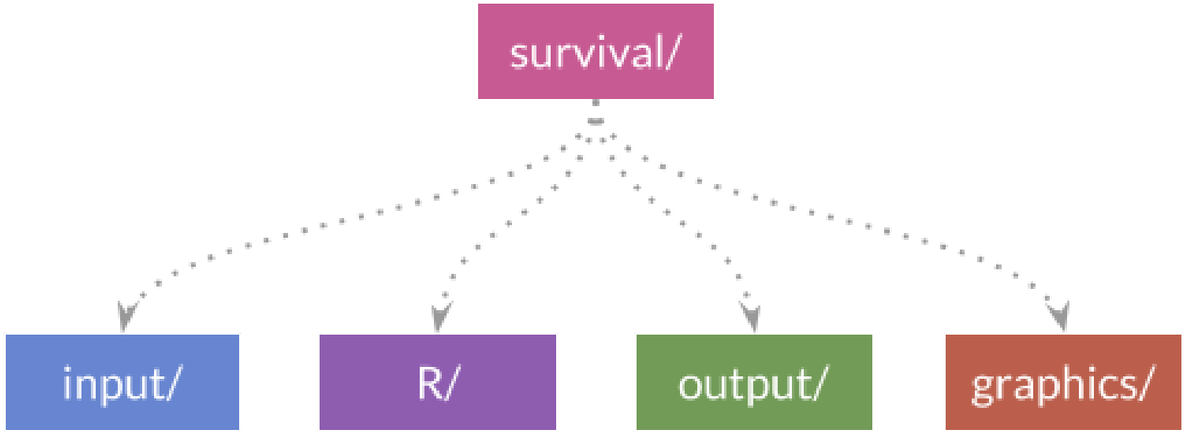
- **graphics**!

In my `R/` directory I have imaginatively named script

- `graphics.R`

that generates all graphics

Make sure to use *relative* paths!



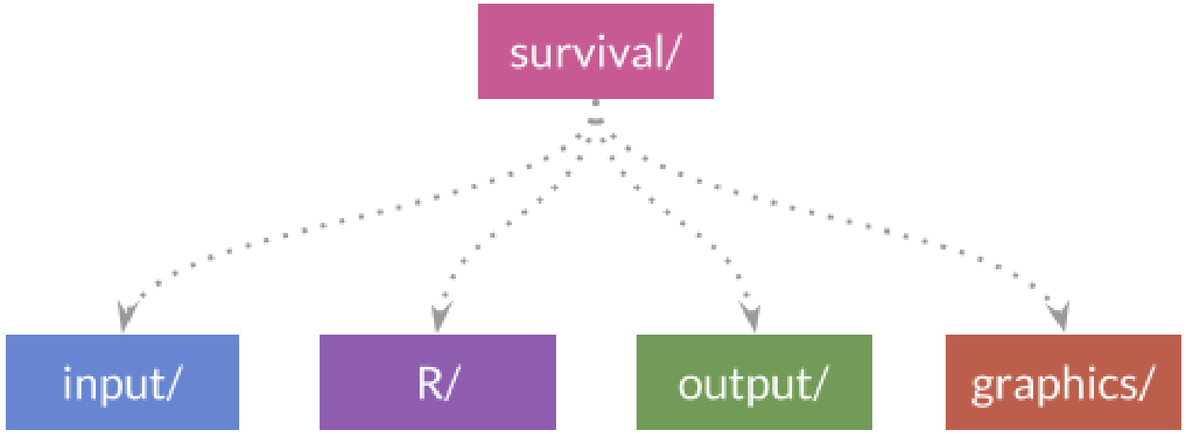
The output/ directory

This directory contains **output**

For example

- List of significant variables, perhaps p-value
- Data for the next analysis

Personally, I typically don't use this directory



Let's try it

DEFENSIVE R PROGRAMMING