Good Coding Style

Dr Andrew J. Stewart

Fellow of the Software Sustainability Institute

E: drandrewjstewart@gmail.com
T: @ajstewart_lang
G: ajstewartlang

Writing Scripts in R

Ultimately you will be writing scripts in R that will allow yourself and others to recreate your analysis just by running the script (or a group of scripts).

It’s important to structure the script(s) that you write clearly and consistently.

A good template is that the code at the start of a script loads the packages your analysis needs, followed by code for loading your data, then tidying, transforming and visualising your data, before moving on to the code for your statistical modelling.

Your code should include lots of comments (prefaced by the # symbol) that will explain to others (and to future you!) what each code section does.

Use Clear Labels for Files and Variables

File names (both scripts and data files) should be meaningful:

`regression_model.R` (Good)
`mymodelfinalv3.R` (Bad)

Variable names should be meaningful and in lowercase:

`age` - Good
`A1` - Bad
`expt1_raw_data` - Good
`Expiriment1datawithoutliersremoved` - Bad

Use Whitespace to Help Readability

Place spaces around operators like `+`, `-`, `=`, `<-`, `%<%`

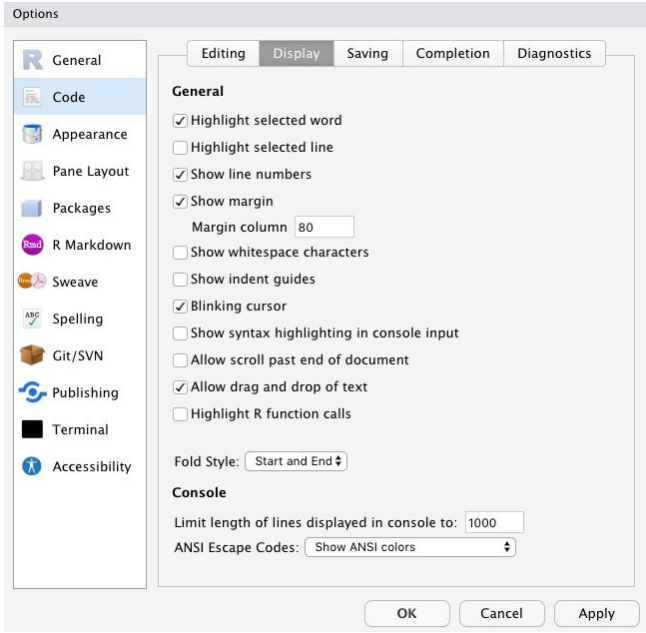
Whitespace used sensibly makes your code easier to read. Separate discrete sections of code in your script with a blank line (or two).

When writing scripts, continue on a subsequent line rather than writing one very long line.

I set a margin of 80 characters in my script window and make sure my code doesn't go beyond this.

Comment, comment, comment…

```
# First we load our datafile
```



Sharing your code

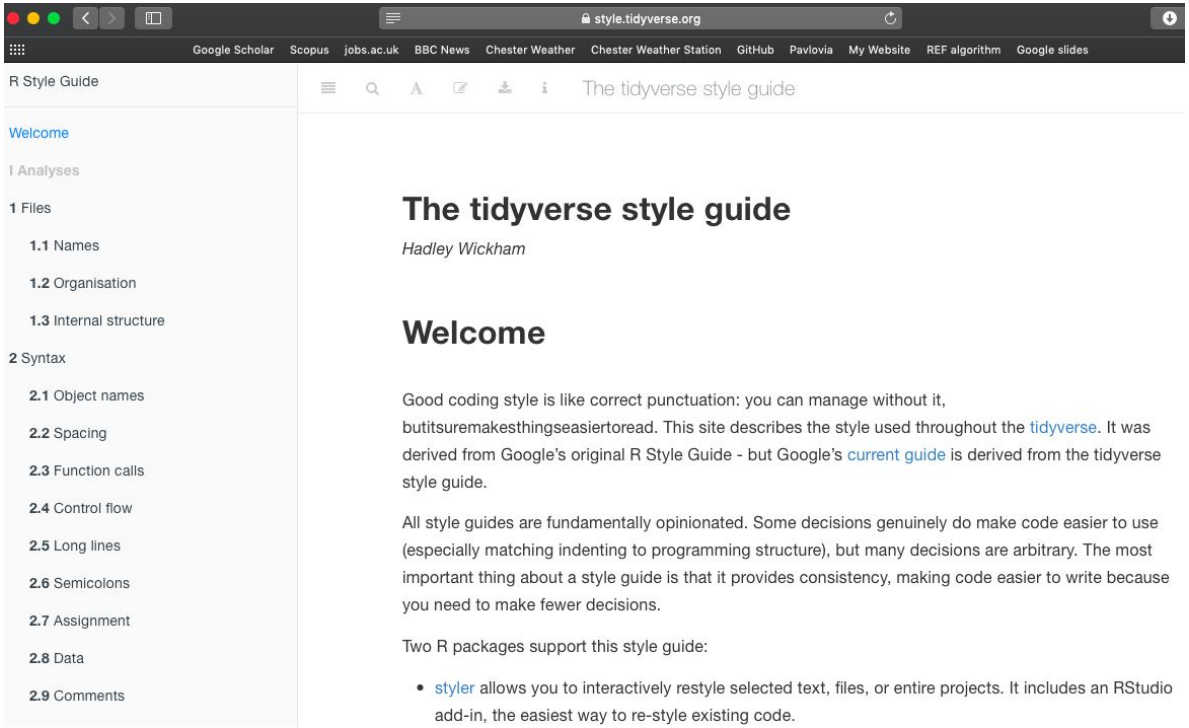
Analysis scripts can be saved and uploaded to (e.g.) GitHub, or submitted as an electronic supplement alongside your journal submission.

Even when journals don't require you to adhere to Open Research practices, it's a good idea to make your code and data available during the reviewing process - and publicly available once your paper is published.

Remember to add an open source license so others can (re)use your code - here is a good resource about which one you might want to adopt: https://choosealicense.com

If you aren't sharing your data and code, you aren't engaged in generating reproducible or open research.

The Tidyverse Style Guide



The guide can be found at: https://style.tidyverse.org

There's also a great package called `styler` which allows you to automatically re-style code.