Appendix: Tutorials, examples, resources

Supplement for the article:

Excuse me, do you have a moment to talk about version control?
by Jennifer Bryan

This appendix provides annotated links to step-by-step instructions, examples, and other resources, as promised in the main article. It is organized using the same section headers.

Many of these resources live primarily on the web and, therefore, are more perishable than the content of the article. As time goes on, it is possible that URLs may change, and it might be necessary to use a site's navigation or search features to re-locate the resource.

Why Git?

Software:

- Git is a version control system.
- RStudio is an integrated development environment for R. It can help you use Git and GitHub. More about that below.

Website:

- GitHub is a web hosting service for Git repositories.

What is Git?

Comic relief:

- "FINAL.doc" from PhD, pokes gentle fun at DIY filename-based version control.

I cite it in the article, but really want to underscore the accessibility of Alice Bartlett's talk "Git for Humans". Highly recommended. Slides on Speakerdeck.

https://github.com/jennybc/excuse-me-iris is the actual Git repository that lead to the screenshots in the figure.

Who should read this and what to expect

no links

What is GitHub?

Example of utility: Drill down into the code of any R package on CRAN by clicking around this read-only mirror of CRAN stewarded by Gábor Csárdi.

- https://github.com/cran Just click on a package or use the search box to find the packages you use most.
- Compare this to the alternative of downloading the `*.tar.gz` from CRAN, unpacking it, and exploring the code locally. And repeating that process whenever the package changes.

Example of utility: See the files behind the book "R for Data Science" by Garrett Grolemund and Hadley Wickham:

- https://github.com/hadley/r4ds
- Click on files to see their current state right now.
- Look at the commit history to see how things have evolved.
- Use the search box in upper left corner to search for specific words, like `join` or `factor`.
- See a summary of what the >100 contributors have added to this book: this is how readers submit fixes and small improvements.
- Look at the issues, such as the one where I discuss error handling in R Markdown documents, which is linked to the exact commit where the authors addressed my suggestion.

Comic relief

- xkcd 1597 is about Git and strategies for coping with its famously unfriendly user interface.

Initial system setup

Happy Git and GitHub for the useR holds concrete setup and early usage information, developed through several years of Git/GitHub use in STAT 545 at the University of British Columbia.

- Register a free GitHub account
- Install or update R and RStudio
- Install Git **We have specific reasons for our installation methods, so seriously consider using them!**
- Introduce yourself to Git
- Install a Git client. Good options – you might want more than one!
  - RStudio has basic Git client functionality.
  - SourceTree is free and excellent.
  - GitKraken is also well-regarded. It is notable in that it runs on Linux, in addition to Mac OS and Windows.
- Connect to GitHub will affirm that your system is set up properly.

Repositories and workflow

Continuing to link to Happy Git and GitHub for the useR.

- New project, GitHub first is a good introductory workflow for starting new projects.
- Existing project, GitHub first shows one way to take an existing project and make it a Git/GitHub repo and RStudio Project.
- Existing project, GitHub last shows another way to get an existing project onto GitHub.

Commits, diffs, and tags

https://github.com/jennybc/excuse-me-iris is the actual Git repository depicted in this figure.

Comic relief

- xkcd 1296 is about the challenge of writing highly informative commit messages all the time.

Repo, commit, and diff. dplyr is a widely used R package whose development is managed with Git and on GitHub.

- Visit its GitHub repo: https://github.com/tidyverse/dplyr.

- Look at some of the recent commits: https://github.com/tidyverse/dplyr/commits/master.
- Click on an individual commit to see exactly what changed, possibly across multiple files, from one snapshot to the next.
- CRAN releases are all marked with tags: https://github.com/tidyverse/dplyr/releases, which make it easy to see the exact state of the source code for any specific version.

General Git books:

- Git in Practice by Mike McQuaid is my favorite book for day-to-day Git operations.
- Pro Git by Scott Chacon and Ben Straub is an excellent and comprehensive resource. You can read it for free online.

Markdown is special on GitHub

Markdown is widely used to author content for the web:

- WordPress: https://en.support.wordpress.com/markdown/
- Stackoverflow: https://stackoverflow.com/editing-help
- GitHub: https://help.github.com/categories/writing-on-github/

Fantastic tutorial on Markdown, with step-by-step interactive exercises and immediate feedback:

- http://commonmark.org/help/tutorial/

Markdown is special for R users

R Markdown = R + Markdown:

- http://rmarkdown.rstudio.com is a landing page for a huge array of resources about R Markdown.
- Happy Git and GitHub for the useR has tutorials for:
  - Test drive R Markdown
  - Render an R script

Which files to commit

Happy Git and GitHub for the useR deals with some of these choices in Make a GitHub repo browsable.

Collaboration

How Google Solved the Version Control Problem is an excellent presentation of the fundamental problem with DIY version control and how cloud-based systems, like Google Drive and GitHub, offer a better way.

GitHub has many excellent resources for collaborative workflows, using branches, issues, and pull requests, including how to tackle merge conflicts:

- Collaborating with issues and pull requests

GitHub as web presence

Happy Git and GitHub for the useR covers this in Make a GitHub repo browsable.

If you decide to make a proper website (vs just making your GitHub repo more browsable), go here:

- GitHub Pages is the basic platform.
- GitHub Pages can create a simple website automatically.
- Jekyll static site generator for maximum flexibility.
- R- and R Markdown-specific frameworks:
  - R Markdown websites
  - bookdown
  - blogdown

Where to go next?

Alternatives to GitHub:

- Bitbucket
- GitLab

Call to action

Donoho's "50 Years of Data Science" examines Statistics' slowness to embrace a broader definition of scholarly statistical activity and training:

- http://courses.csail.mit.edu/18.337/2015/docs/50YearsDataScience.pdf