

Version control for reproducible science

https://bradduthie.github.io/version_control/vc_slides.pdf

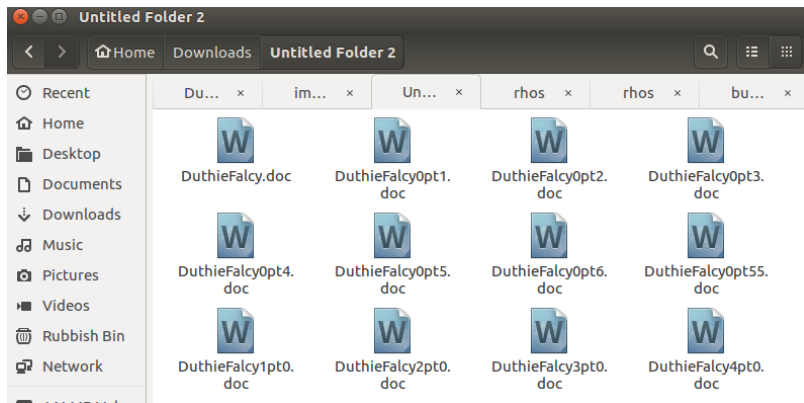
Brad Duthie

16 January 2020

Rough outline of version control workshop

1. **14:00** What is version control, and why use it?
2. **14:20** Getting set up – good file management
3. **14:30** The GitKraken interface and simple commits
4. **15:00** Setting up GitHub, pushing and pulling
5. **15:30** Branching using GitKraken
6. **16:00** Merging and merge conflicts
7. **16:30** Independent work using version control

What is version control, and why use it?



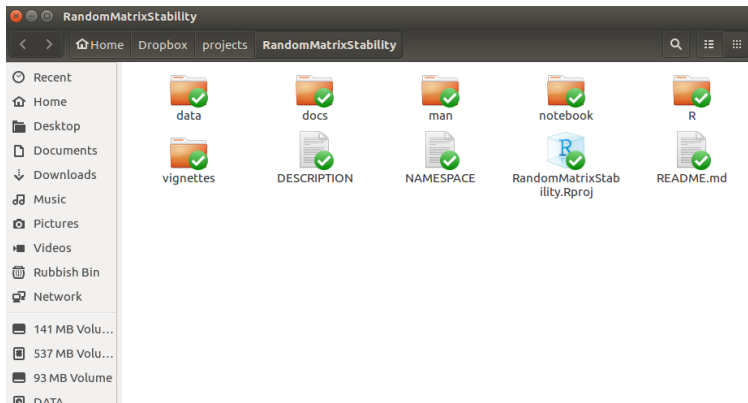
What version control software does

- ▶ Software that records changes you make to files over time
 - ▶ Manage different *versions* of files (no need to 'Save As...')
 - ▶ Recover old files, keep track of file changes
 - ▶ Collaborate with others on shared files

What version control software does


- ▶ Software that records changes you make to files over time
 - ▶ Manage different *versions* of files (no need to 'Save As...')
 - ▶ Recover old files, keep track of file changes
 - ▶ Collaborate with others on shared files
-
- ▶ **Put more intuitively**, version control takes a snapshot in time (called a '**commit**') of all the files in one of your folders (called '**repositories**')
 - ▶ Visualise changes to your files over time
 - ▶ Look at the differences between file versions
 - ▶ Record who changed files, and what they changed


Inside of a project on version control






Folders (a.k.a, 'repositories') include all data files, R code, notes, manuscript drafts, etc.



Full annotated timeline of folder changes (GitHub)


 Commits on Mar 1, 2019


Some work on the SI
 bradduthle committed on 1 Mar 2019 ✓



 **5d29331** 


Major restructure and revision of the Discussion to compare to Gibbs
 bradduthle committed on 1 Mar 2019 ✓
...et al.


 **a40ede5** 



 Commits on Feb 27, 2019


Edit the manuscript up to Reviewer 2 specific comment 3 -- these are
 bradduthle committed on 27 Feb 2019 ✓
...next on the list, specifically the new Discussion paragraph



 **d411fd5** 

 Commits on Feb 22, 2019

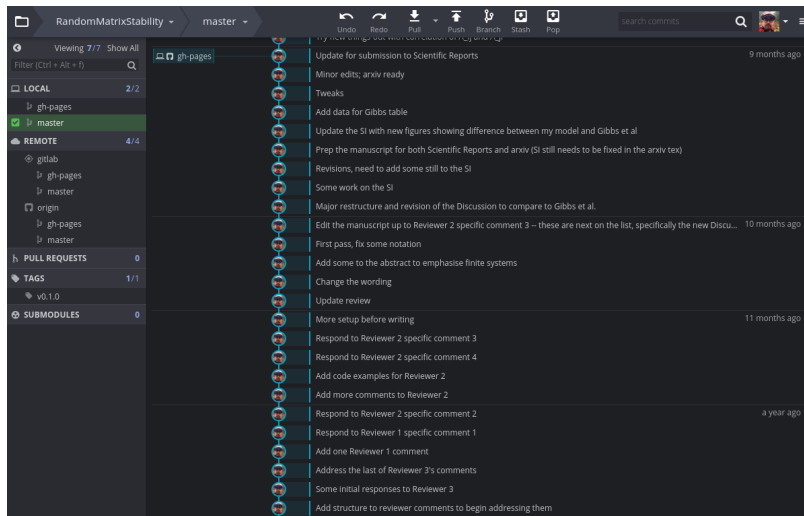
First pass, fix some notation
 bradduthle committed on 22 Feb 2019 ✓

 **39f854b** 

Add some to the abstract to emphasise finite systems
 bradduthle committed on 22 Feb 2019 ✓

 **612aa4b** 

Full annotated timeline of folder changes (GitKraken)



Parallel versions ('branches') of a folder (GitKraken)

The screenshot displays the GitKraken application interface for a repository named 'helicoverpa'. The 'version_control' branch is selected, and the 'gh-pages' branch is highlighted in the left sidebar. The central pane shows a commit history graph with a vertical timeline of commits. The right pane shows the commit details for the selected commit (5ad7ea), including the commit message 'Merge branch 'master' into gh-pages', the author 'Brad Duthie', and a list of modified files.

Repository: helicoverpa

Branch: version_control

Commit: 5ad7ea

Commit Message: Merge branch 'master' into gh-pages

Author: Brad Duthie (parent: 0f71de, ffd76d)

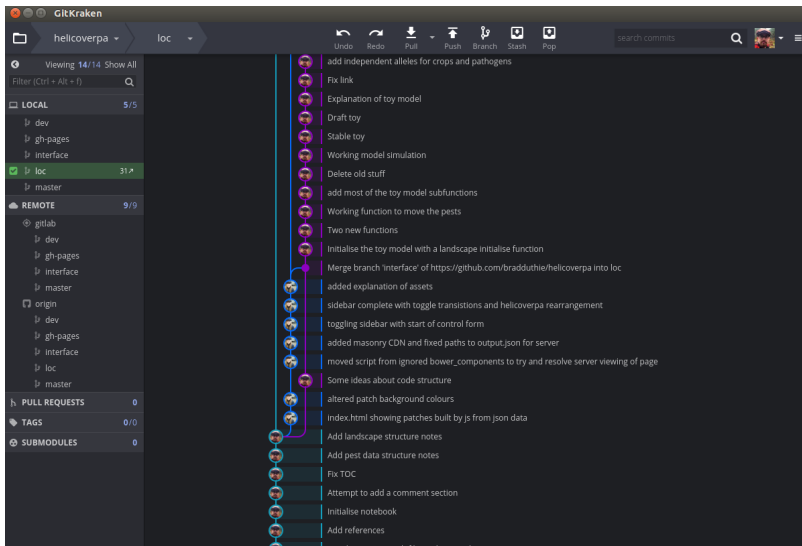
Files Modified:

- RStudio_and_git.html
- RStudio_and_git.Rmd
- vc_notes.html
- vc_notes.Rmd
- vc_presentation.html
- vc_presentation.pdf
- vc_presentation.Rmd
- vc_slides.pdf
- vc_slides.Rmd

Commit History (from bottom to top):

- Start merge conflict section
- Commit the first change with the list
- Change Lettuce to Cucumber on list
- Change Lettuce to Spinach
- Merged branch list_A into master
- Paragraph explaining merge conflict
- Change to Lettuce
- CLI merge conflict
- Change Apples to Pears
- Change Apples to Bananas
- list_C
- Add the notes I just wrote
- Merge branch 'list_C'
- More notes
- How do you link them apples?
- Nearing end of GitHub stuff
- First draft of notes added
- Typo and cheat sheets.
- Merged master into gh-pages
- Fix spelling
- Merged branch master into gh-pages
- Fix links and minor edits
- Merged branch master into gh-pages
- Change to avocado and start a new ...

Collaborative history or a shared folder (GitKraken)



Clear breakdown of what has changed (GitKraken)

The screenshot displays the GitKraken application interface. The top bar shows the repository name 'helicopterpa' and the branch 'RandomMatrixStability'. The central pane shows a diff for the file 'notebook/ms.Rmd'. The diff highlights changes in the manuscript text, including mathematical expressions and references. The right-hand panel shows the commit details for 'd411fd', authored by Brad Duthie on 27/2/2019. Below the commit details is a file tree showing the project structure, including 'notebook', 'ms_files', 'ms.pdf', 'ms.Rmd', 'ms.tex', 'PLOS_Compu_reviews.html', and 'PLOS_Compu_reviews.md'.

Repository: helicopterpa | Branch: RandomMatrixStability | Commit: d411fd

22 file changes in working directory | View changes

Edit the manuscript up to Reviewer 2 specific comment 3 -- these are next on the list, specifically the new Discussion paragraph

Brad Duthie
authored 27/2/2019 @ 18:00 | parent: 39f854

8 modified + 1 added

Expand All

- notebook
 - ms_files 4
 - ms.pdf
 - ms.Rmd
 - ms.tex
 - PLOS_Compu_reviews.html
 - PLOS_Compu_reviews.md

Diff View: notebook/ms.Rmd

251 251
252 252 Randomly assembled complex systems can be represented as large square matrices (\mathbf{M}) with S components (e.g.
253 253
254 254 -- May's [May1972; @Allesina2012] stability criterion $\sigma(\sqrt{SC}) < 1$ assumes that the expected response rates ($\sigma(\gamma)$
254+ May's [May1972; @Allesina2012] stability criterion $\sigma(\sqrt{SC}) < 1$ assumes that the expected response rates ($\sigma(\gamma)$
255 255
256 256 <--
257 257

360 360
361 361 It is important to emphasise that variation in component response rate is not stabilising per se; that is, adding variation in comp
362 362
363 363 --<-- Also important to emphasise Gibbs result -- I'm doing this for finite systems, and I deliberate stressed the system complexi
363+ <--
364+ <--
364+ Also important to emphasise Gibbs result -- I'm doing this for finite systems, and I deliberate stressed the system complexity to
366+ <--
367+ But Gibbs was more interested in first assuming a stable matrix and then showing that the vector of abundances would not cha
368+ <--
369+ <--
370+ <--
371 371
364 371
365 372 The potential importance of component response rate variation was most evident from the results of simulations in which the
366 373

387 394
388 395 $\frac{d\mathbf{v}(t)}{dt} = \mathbf{A}(\gamma)\mathbf{v}(t)$
389 396
390 396 -- In the above, $\mathbf{A}(\gamma)$ is a diagonal matrix in which elements correspond to individual component response rates. T
397+ In the above, $\mathbf{A}(\gamma)$ is a diagonal matrix in which elements correspond to individual component response rates. T
391 398
392 399 **Genetic algorithm**. Ideally, to investigate the potential of $\text{Var}(\gamma)$ for increasing the proportion of stable complex s
393 400

Version control makes science easier

- ▶ **Organises files** by avoiding 'save as' multiple versions
 - ▶ analysis_1.R
 - ▶ analysis_2.R
 - ▶ analysis_FINAL.R
 - ▶ analysis_FINAL_no_really_this_time.R

Version control makes science easier

- ▶ **Organises files** by avoiding 'save as' multiple versions
 - ▶ analysis_1.R
 - ▶ analysis_2.R
 - ▶ analysis_FINAL.R
 - ▶ analysis_FINAL_no_really_this_time.R
- ▶ **Provides a clear history** of what you have done, when, and why (through commit comments)

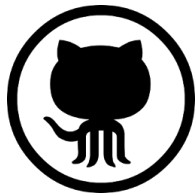
Version control makes science easier

- ▶ **Organises files** by avoiding 'save as' multiple versions
 - ▶ analysis_1.R
 - ▶ analysis_2.R
 - ▶ analysis_FINAL.R
 - ▶ analysis_FINAL_no_really_this_time.R
- ▶ **Provides a clear history** of what you have done, when, and why (through commit comments)
- ▶ **Saves time** by avoiding loss of data, analysis, or writing when integrating with GitHub

Version control makes science easier

- ▶ **Organises files** by avoiding 'save as' multiple versions
 - ▶ analysis_1.R
 - ▶ analysis_2.R
 - ▶ analysis_FINAL.R
 - ▶ analysis_FINAL_no_really_this_time.R
- ▶ **Provides a clear history** of what you have done, when, and why (through commit comments)
- ▶ **Saves time** by avoiding loss of data, analysis, or writing when integrating with GitHub
- ▶ **Gives peace of mind** to experiment by removing any fear of breaking something that you know works

Version control can help open science



- ▶ Transparent record of data collection, analysis, and writing
- ▶ Record publicly available on GitHub, Bitbucket, or GitLab
- ▶ GitHub repository can be copied, reproduced, and discussed
- ▶ git and GitHub can track individual contributions to a project

Most researchers use git (and GitHub)



- ▶ Free and open-source
- ▶ Separate from GitHub

Most researchers use git (and GitHub)



- ▶ Free and open-source
- ▶ Separate from GitHub
- ▶ Works across platforms
 - ▶ Windows
 - ▶ Linux
 - ▶ Mac
- ▶ Invented by Linus Torvalds

Why focus on using GitKraken?



- ▶ Free to download and use
- ▶ Easy GitHub integration
- ▶ Graphical user interface
- ▶ Visualisation of repository

Accompanying notes to these slides are available in the `version_control` repository, and include instructions for using the command line interface, and for editing directly in GitHub.

Objectives: using version control

Guided walkthrough of setting up a project in GitHub and GitKraken to manage a project with version control.

Slides: `https:`

`//bradduthie.github.com/version_control/vc_slides.pdf`

Notes: `https:`

`//bradduthie.github.com/version_control/vc_notes.html`

Discuss, share, and get additional help by raising an issue in the `version_control` repository on the Stirling Coding Club.