

Research Tools and Methods for the Mathematical Science

Lecture 6: Backups and Revision Control

Matthew Roughan

`<matthew.roughan@adelaide.edu.au>`

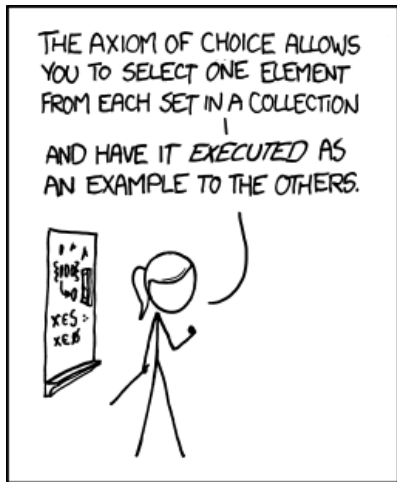
[http://www.maths.adelaide.edu.au/matthew.roughan/
Lecture_notes/ResearchToolsCourse/](http://www.maths.adelaide.edu.au/matthew.roughan/Lecture_notes/ResearchToolsCourse/)

School of Mathematical Sciences,
University of Adelaide

March 30, 2015

Proof by Intimidation: "Don't be stupid, of course it's true."

Proof by Terror: When intimidation fails ...



MY MATH TEACHER WAS A BIG BELIEVER IN PROOF BY INTIMIDATION.

<http://xkcd.com/982/>

Data storage requirements

The ARC and NHMRC's "Australian Code for the Responsible Conduct of Research", requires, amongst many others issues:

In general, the minimum recommended period for retention of research data is 5 years from the date of publication.

Section 2.1.1

Your funding (for those with scholarships) implicitly commits you to this policy.

- retention isn't just about holding it
- it must be readable (and understandable)

Retain research data, including electronic data, in a durable, indexed and retrievable form.

Section 2.6.4

[Adelaide Uni Policy](#)

[Adelaide Uni Information](#)

[Adelaide Uni Training](#)

The scary stories

- Student lost their entire thesis (almost finished) when a hacker cracked their laptop, and encrypted the drive
- I asked a colleague for a dataset they had used in a paper, as I wanted to do a comparison with a new algorithm. They couldn't find the data.
- I have a nice example of a problem that happened, but I can't find where I wrote it up.

Backups

- **MANDATORY**
- Your responsibility (even if someone says they are doing them for you)
 - ▶ I once lost a nice chunk of data when a disk crashed, and when I asked for the backups, was told some disks had been corrupted and they didn't work.
- Backups are **NOT REAL** unless you test them (regularly)

Backup models

- incremental/differential vs full
 - ▶ most things on your system change rarely, so why record the things that haven't changed?
- unstructured vs system
 - ▶ do you just record the files you “need”?
- continuous (e.g. RAID)
- off-site vs local
- how do you access the data (online, database, snapshot, ...)

External hard-drive

- unstructured:
 - ▶ copy some files to it, sometimes
 - ▶ requires discipline and care
- time machine
 - ▶ differential++
 - ▶ nice interface
 - ▶ BTA (Better Trust Apple)
- questions:
 - ▶ how do you do off-site?
 - ▶ how reliable is an external drive?

Cloud storage: e.g. Dropbox

Nice features:

- auto-backup (one-month history)
- distributed (access almost anywhere)
- sharable (can give access to others)

Problems:

- sync
- easy for someone to delete or break something
 - ▶ in theory can recover
 - ▶ recovery is manual – what if lots was lost?
- churn – puts stress on networks, potentially

Does protect against a disk failure.

Doesn't protect (properly) against a hacker deleting everything.

Collaborative writing

Co-authors won't save you any time, but they do help filter out your idiosyncrasies.

Jeff Ullman [KLR89, p.67]

- Most of your writing will be collaborative
 - ▶ often participants are distributed
 - ▶ there are lots of ways to deal with this
 - ▶ even when they are local, these techniques help
- Models:
 - ▶ One person acts as editor, and incorporates changes
 - ★ others communicate proposed changeslots of work for editor, but only they end up happy.
 - ▶ Token: one person has the “token” (for all or part)
 - ★ edit as please when have token
 - ★ pass it when finished (e.g. by email)great with timezones, but requires trust.
 - ▶ Truly distributed:
 - ★ all have access, and can edit
 - ★ conflicts are mergedvery powerful, but requires tools.

Collaborative writing models

Truly distributed collaboration

Equally applicable for code or LaTeX or ...

- We want tools that support
 - ▶ distributed access – e.g. Dropbox
 - ▶ revision control [Hen07]
- Examples:
 - ▶ payfor: Scribetex, writeLaTeX, SpanDex, ShareLaTeX
 - ★ often have a free plan, but are they free forever?
 - ★ focussed on latex, not the rest (e.g. accompanying code and data)
 - ▶ Latexlab (google docs integration) <http://docs.latexlab.org/docs>
 - ▶ free: standard open source tools
 - ★ revision control
- This is a field in flux ...

Revision (or version) control

Truly distributed collaboration

- Examples:
 - ▶ git
 - ▶ svn
 - ▶ cvs
- Features:
 - ▶ allow you to see all revisions of paper
 - ★ e.g. revert back to an old version if you don't like changes
 - ▶ trace activity
 - ★ volume
 - ★ also what changed, with comments
 - ▶ atomic operations (cvs lacks this)
 - ★ even if something is interrupted, system is left in consistent state
 - ▶ file locking (some systems)
- Good for code, and LaTeX, and (some) data
- **Comparison** of revision control systems

Git instructions

Many howtos and primers, e.g.

- <http://stackoverflow.com/questions/315911/git-for-beginners-the-definitive-practical-guide>
- <http://sixrevisions.com/resources/git-tutorials-beginners/>
- <http://starlink.jach.hawaii.edu/starlink/GitPrimer>
- http://wiki.kokuaviewer.org/wiki/Git_Primer
- <http://www.doblock.com/articles/a-git-primer-fit-for-linus-himself>
- <http://software-carpentry.org/v5/novice/git/index.html>

Git instructions

Only a few simple operations needed to get started:

clone create a copy of a repository

```
git clone git://github.com/something.git
```

add add files to the repository

```
git add filename
```

commit commit your changes

```
git commit filename
```

push push a set of committed changes to the repository

```
git push
```

pull pull (update) from the repository

```
git pull
```

but there are lots of other things you can see and do.

GitHub <https://github.com/>

- Provides an online (cloud) version of git with a nice set of interfaces.
 - ▶ can use web interface for many things
 - ▶ also a Mac GUI client
- **2011 numbers**
 - ▶ used by 2 million projects
 - ▶ 4,500 new GitHub projects per day
- We have an institutional account
 - ▶ I am the admin
 - ▶ I can set up a private project for you
- Getting started requires you jump through a few hoops:
<https://help.github.com/categories/53/articles>
<https://help.github.com/categories/54/articles>
- Lots of help, e.g., <https://help.github.com/>

Why not Dropbox + git?

- Sync – state in Dropbox might not be sync'd when you commit
 - ▶ non-atomic
- Overlap of functionality
 - ▶ both keep some type of history
 - ▶ inefficient and inelegant
- Dropbox is chatty
 - ▶ do you want to colleagues to see every single change you make? or just the new draft?
- Simultaneous edits are bad in Dropbox

Collaborative writing

Using revision control effectively for writing

- Line-wraps: don't have one para per line
 - ▶ every change changes the whole paragraph
- Break into segments (using LaTeX includes)
 - ▶ can be independently edited
- Use standard packages (everyone has)
- Sharing .bib files
 - ▶ agreed keyname format
- Use “ignores” to keep “products” out of the repository
 - ▶ in coding “binaries” are the products
 - ▶ are PDFs a “product”?
 - ▶ what about images?
- Verify LaTeX compiles before committing it

See [http:](http://)

[//stackoverflow.com/questions/6188780/git-latex-workflow](http://stackoverflow.com/questions/6188780/git-latex-workflow)
and <https://www.sharelatex.com/blog/2012/10/16/collaborating-with-latex-and-git.html> for more discussion.

Summary

- YOU are responsible for your backups
 - ▶ Backups are insanely important, but you won't realise that until you don't have them
- Revision control will make your life easier

Assignment

Get a GitHub account

- If you need help follow the instructions on <https://help.github.com/categories/53/articles>
- Let me know your account name
- I'll give you (read) access to the course repository
- I'll set up a “thesis” repository for you.
 - ▶ if you get your supervisors to join GitHub (or they already have), then I will include them into your repository.

Further reading I



Arne Henningsen, *Tools for collaborative writing of scientific LaTeX documents*, The PracTEX Journal (2007).



Donald E. Knuth, Tracy L. Larrabee, and Paul M. Roberts, *Mathematical writing*, Mathematical Association of America, 1989, jmlr.csail.mit.edu/reviewing-papers/knuth_mathematical_writing.pdf, contains a huge amount of very good advice, but loosely organised (just reports of a set of lectures).