One Button Reproducibility

June 5, 2014

Links and Contact Info

- `http://github.com/dexy/repro-demo`
- `http://dexy.it`
- `http://twitter.com/dexyit`
- `http://twitter.com/ananelson`

Soapbox

Invest in the command line (and working with text-based files).

Reproducibility Conversation

Have you ever tried to reproduce someone else's work? Have you ever tried to reproduce your own work from the past?

How Transparent Is It?

All the source code for a paper is in a github repository. How transparent is it? What is the cost of reproducing this paper? (mean/median/mode/stdev)

We have to:

- Reverse engineer which software was installed.
- Install this software in our dev environment.
- Reverse engineer which Python/R packages are used in each script.
- Install these packages in our dev environment.
- Try to determine the correct order of running the scripts, and how data is shared between them.
- Run all scripts in correct order.
- Figure out how to link scripts to the paper.

All of this with *no guarantee* that I will be able to get your code working, because:

- you might not have included all data

- you might not have included all the scripts
- you might be using some proprietary software or some antiquated software which I can't install
- it might be impossible to find out which data was used for which results in your paper
- you might have changed some scripts after they were used to generate results in your paper
- I might just not have enough time to figure all this out

But, the author wrote a README!

A README is great, but it doesn't guarantee reproducibility, because:

- the author might have left some steps out
- the author might have written it a year ago and not kept it up to date
- it might be totally correct, but we have no idea how accurate it is (back to stdev idea)

How do you know a README is correct?

Ask a colleague with no experience of your project to follow your README step-by-step with no help from you.

Great, but other people still have no idea how trustworthy a README is.

One Button Reproducibility Example

`http://github.com/dexy/repro-demo`

Comparison of IPython Notebook and Dexy

The notebook metaphor: a single document containing code and annotations. Very intuitive, self-contained. Great for exploratory work.

Dexy is a command line tool which wraps lots of other command line tools in a common interface. It's similar to GNU Make, but designed specifically for reproducible research and writing documents.