Reproducible Research Checklist

What to Do and What Not to Do

Roger D. Peng, Associate Professor of Biostatistics
Johns Hopkins Bloomberg School of Public Health

DO: Start With Good Science

- Garbage in, garbage out
- Coherent, focused question simplifies many problems
- Working with good collaborators reinforces good practices
- Something that's interesting to you will (hopefully) motivate good habits

DON'T: Do Things By Hand

- Editing spreadsheets of data to "clean it up"
  - Removing outliers
  - QA / QC
  - Validating
- Editing tables or figures (e.g. rounding, formatting)
- Downloading data from a web site (clicking links in a web browser)
- Moving data around your computer; splitting / reformatting data files
- "We're just going to do this once...."

Things done by hand need to be precisely documented (this is harder than it sounds)

DON'T: Point And Click

- Many data processing / statistical analysis packages have graphical user interfaces (GUIs)
- GUIs are convenient / intuitive but the actions you take with a GUI can be difficult for others to reproduce
- Some GUIs produce a log file or script which includes equivalent commands; these can be saved for later examination
- In general, be careful with data analysis software that is highly *interactive*; ease of use can sometimes lead to non-reproducible analyses
- Other interactive software, such as text editors, are usually fine

DO: Teach a Computer

- If something needs to be done as part of your analysis / investigation, try to teach your computer to do it (even if you only need to do it once)
- In order to give your computer instructions, you need to write down exactly what you mean to do and how it should be done
- Teaching a computer almost guarantees reproducibilty

For example, by hand, you can

1. Go to the UCI Machine Learning Repository at http://archive.ics.uci.edu/ml/
2. Download the Bike Sharing Dataset by clicking on the link to the Data Folder, then clicking on the link to the zip file of dataset, and choosing "Save Linked File As..." and then saving it to a folder on your computer

DO: Teach a Computer

Or You can teach your computer to do the same thing using R:

```
download.file("http://archive.ics.uci.edu/ml/machine-learning-databases/00275/
              Bike-Sharing-Dataset.zip", "ProjectData/Bike-Sharing-Dataset.zip")
```

Notice here that

- The full URL to the dataset file is specified (no clicking through a series of links)
- The name of the file saved to your local computer is specified
- The directory in which the file was saved is specified ("ProjectData")
- Code can always be executed in R (as long as link is available)

DO: Use Some Version Control

- Slow things down
- Add changes in small chunks (don't just do one massive commit)
- Track / tag snapshots; revert to old versions
- Software like GitHub / BitBucket / SourceForge make it easy to publish results

DO: Keep Track of Your Software Environment

- If you work on a complex project involving many tools / datasets, the software and computing environment can be critical for reproducing your analysis
- **Computer architecture**: CPU (Intel, AMD, ARM), GPUs,
- **Operating system**: Windows, Mac OS, Linux / Unix
- **Software toolchain**: Compilers, interpreters, command shell, programming languages (C, Perl, Python, etc.), database backends, data analysis software
- **Supporting software / infrastructure**: Libraries, R packages, dependencies
- **External dependencies**: Web sites, data repositories, remote databases, software repositories
- **Version numbers**: Ideally, for everything (if available)

DO: Keep Track of Your Software Environment

```
sessionInfo()
```

```
## R version 3.0.2 Patched (2014-01-20 r64849)
## Platform: x86_64-apple-darwin13.0.0 (64-bit)
## 
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
## 
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  base     
## 
## other attached packages:
## [1] slidify_0.3.3
## 
## loaded via a namespace (and not attached):
## [1] evaluate_0.5.1 formatR_0.10   knitr_1.5      markdown_0.6.3
## [5] stringr_0.6.2  tools_3.0.2    whisker_0.3-2  yaml_2.1.8
```

DON'T: Save Output

- Avoid saving data analysis output (tables, figures, summaries, processed data, etc.), except perhaps temporarily for efficiency purposes.
- If a stray output file cannot be easily connected with the means by which it was created, then it is not reproducible.
- Save the data + code that generated the output, rather than the output itself
- Intermediate files are okay as long as there is clear documentation of how they were created

DO: Set Your Seed

- Random number generators generate pseudo-random numbers based on an initial seed (usually a number or set of numbers)
    - In R you can use the `set.seed()` function to set the seed and to specify the random number generator to use
- Setting the seed allows for the stream of random numbers to be exactly reproducible
- Whenever you generate random numbers for a non-trivial purpose, **always set the seed**

DO: Think About the Entire Pipeline

- Data analysis is a lengthy process; it is not just tables / figures / reports
- Raw data → processed data → analysis → report
- How you got the end is just as important as the end itself
- The more of the data analysis pipeline you can make reproducible, the better for everyone

Summary: Checklist

- Are we doing good science?
- Was any part of this analysis done by hand?
  - If so, are those parts *precisely* document?
  - Does the documentation match reality?
- Have we taught a computer to do as much as possible (i.e. coded)?
- Are we using a version control system?
- Have we documented our software environment?
- Have we saved any output that we cannot reconstruct from original data + code?
- How far back in the analysis pipeline can we go before our results are no longer (automatically) reproducible?