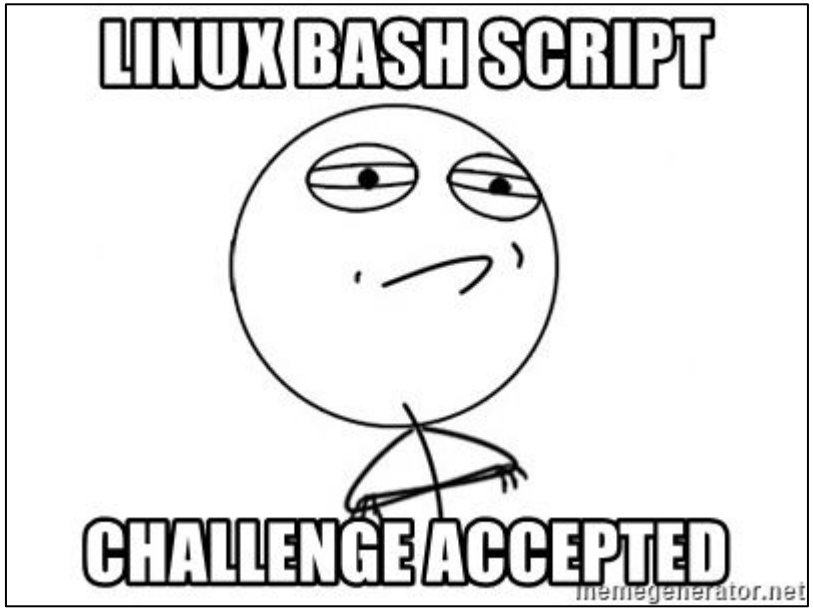
Bash scripting

Lecture 8
BIOS 6660, Spring 2019
Instructor: Pam Russell



https://memegenerator.net/instance/32147888/challenge-accepted-hd-linux-bash-script-challenge-accepted

Bash

The shell language we use on the command line

A full-featured programming language

Bash scripts

Save Bash commands in a text file

Script runs in shell environment

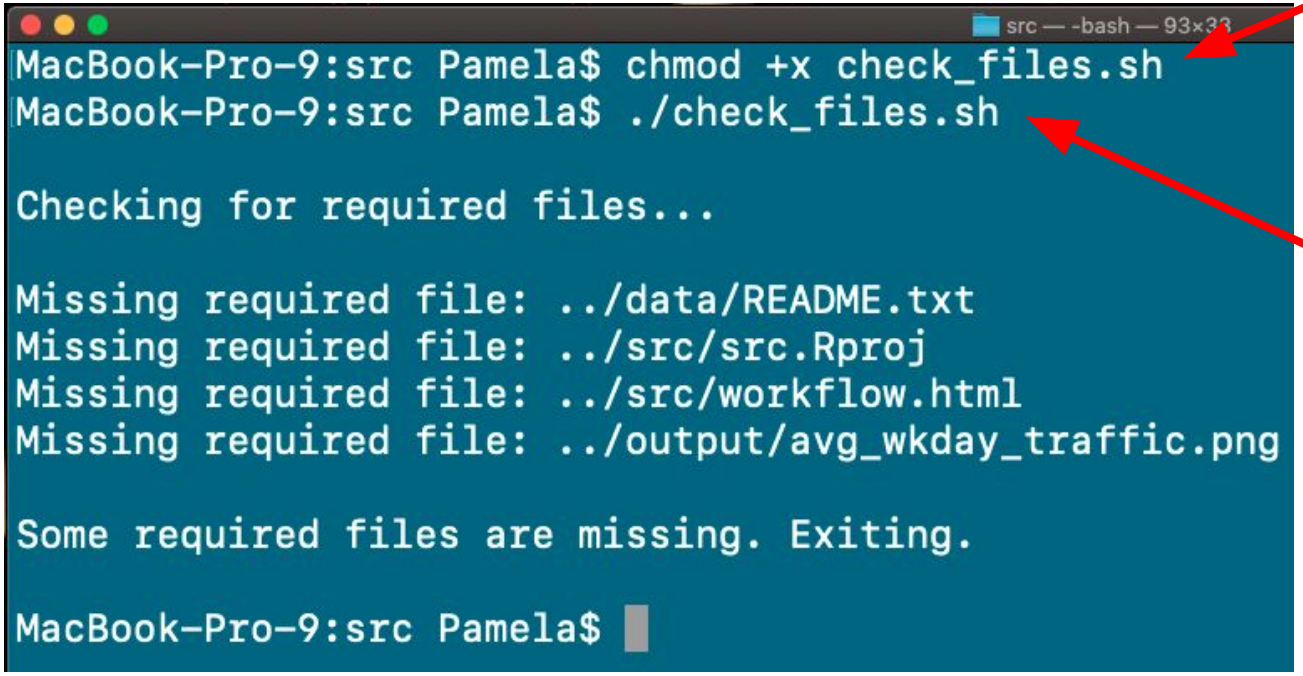
Common for genomic workflows using command line tools

Bash script in HW4: check_files.sh

src — vim check_files.sh — 93×33

```
#!/bin/bash


# This script checks for the existence of required files for HW4,
# and prints some basic information about each.
# You need to complete the section marked "TODO".


# "-e" means "process escape characters" e.g. render "\n" as a
# newline instead of literal "\n"
echo -e "\nChecking for required files...\n"


# Declare some string variables
# Note the "$" to refer to a variable that has already been declared
# Note how you combine strings into a concatenated string
dir_repo="../.."
dir_proj=".."
dir_data="$dir_proj/data"
dir_output="$dir_proj/output"
dir_src="$dir_proj/src"


# Create an array of names of the required files for the homework
# Note "\" means continue the command onto the next line, so this
# array declaration is a single command written across multiple lines
required_files=\
($dir_data/README.txt \
$dir_repo/.gitignore \
$dir_src/src.Rproj \
$dir_src/workflow.html \
$dir_output/avg_wkday_traffic.png)
"check_files.sh" 86L, 2291C
```

Running a bash script



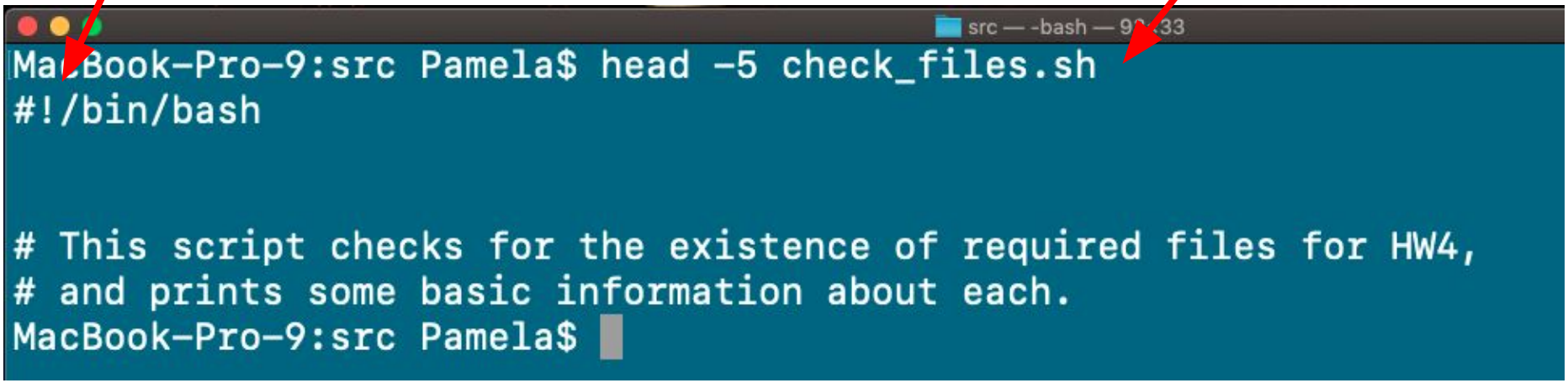
Make file executable by adding **+x** permissions (only need to do once)

Run script.
Note you need to supply the directory (“.”)

File extension and shebang

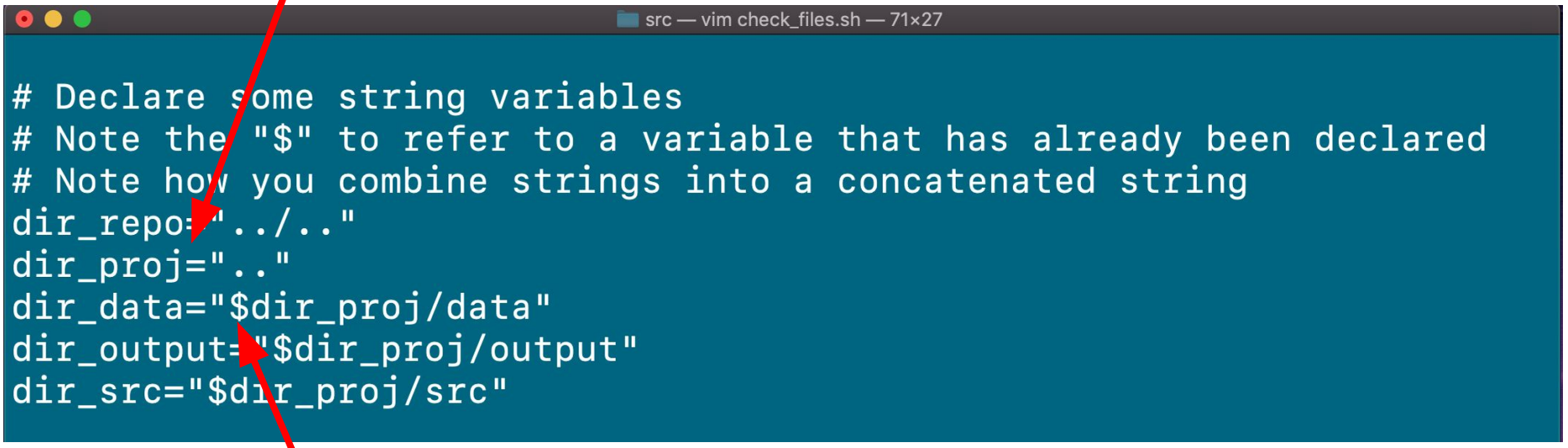
Shebang: **#!** (“hash bang”)
Instructs shell which interpreter to use.
Use **#!/bin/bash** in all Bash scripts.

File extension actually doesn’t matter except to remind us.
Shebang determines how file will be interpreted.



Variable declarations and references

Declare variable **dir_proj**

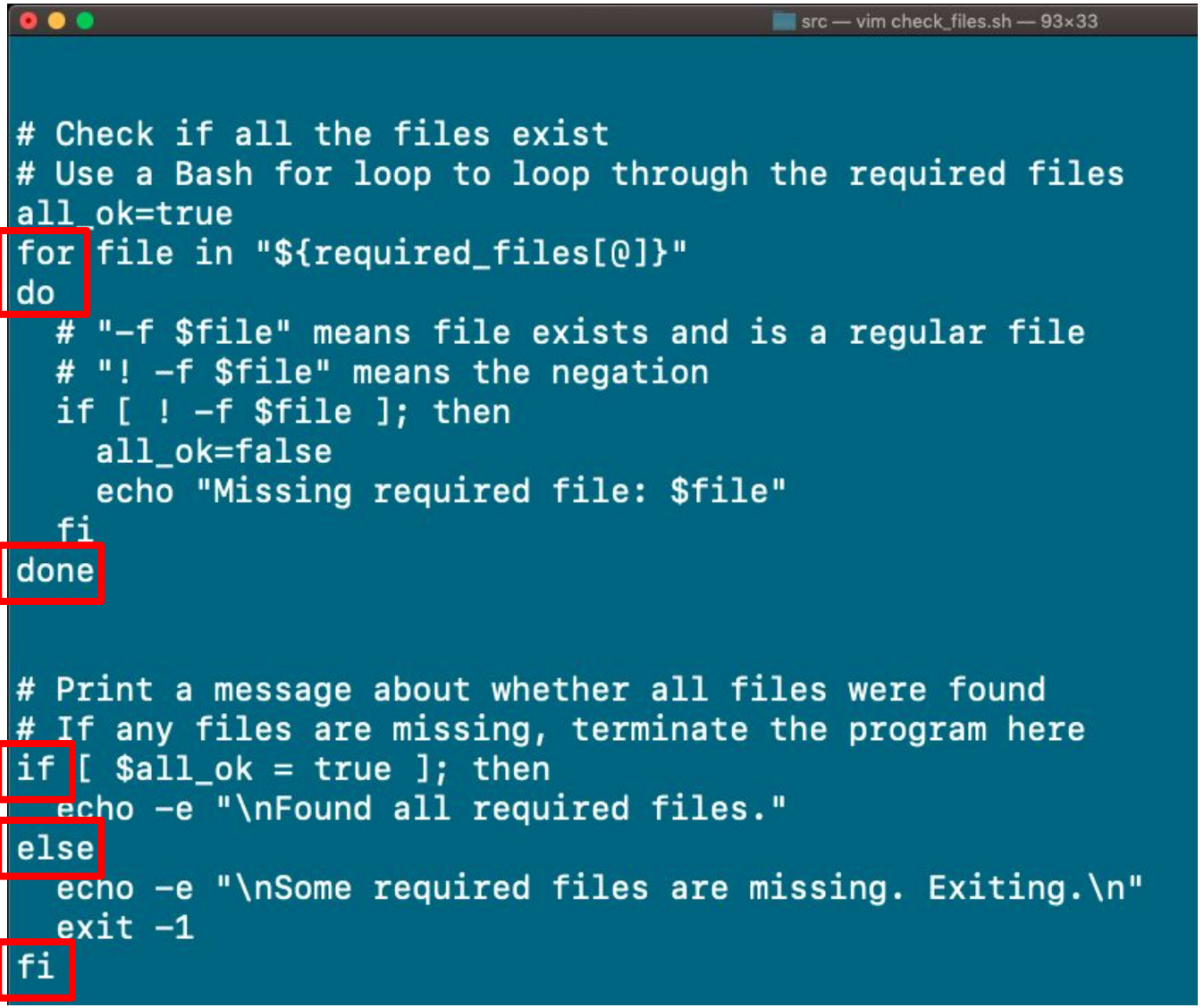


Reference **dir_proj** with **$**

Control structures

For loop

If/else statement



Line continuation

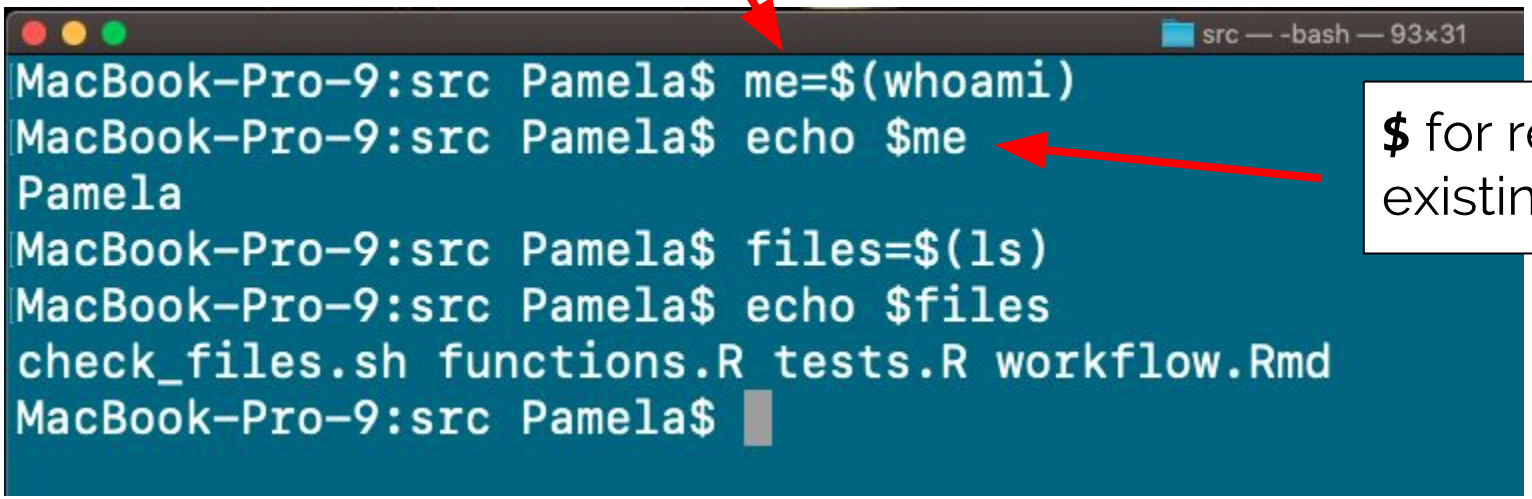
src — vim check_files.sh — 93×31

```
# Create an array of names of the required files for the homework
# Note "\" means continue the command onto the next line, so this
# array declaration is a single command written across multiple lines
required_files=\
($dir_data/README.txt \
$dir_repo/.gitignore \
$dir_src/src.Rproj \
$dir_src/workflow.html \
$dir_output/avg_wkday_traffic.png)
```

Backslash lets you continue a single command onto multiple lines

Capturing output of a command as a variable

Capture output of the **whoami** command in a variable called **me**

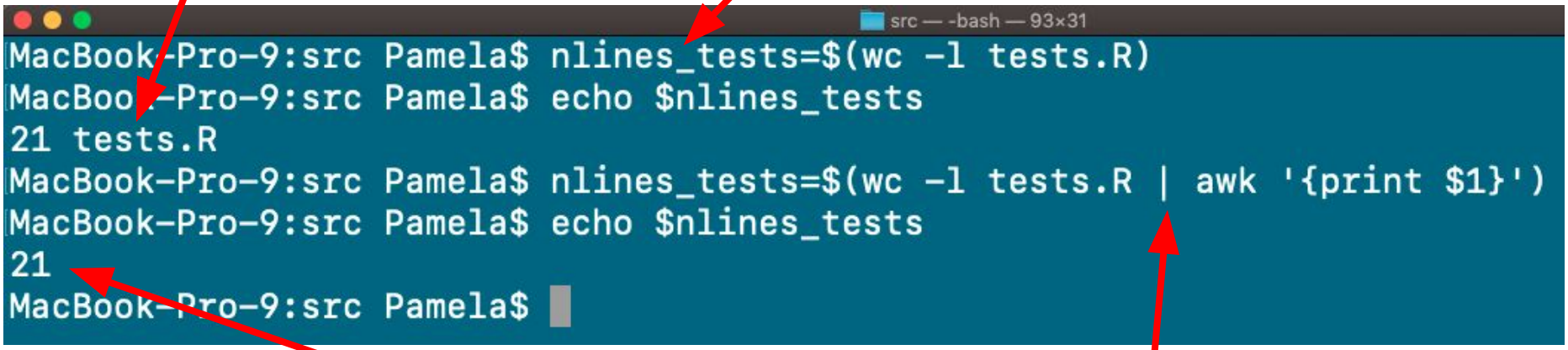


$ for references to existing variables

Pipes and awk

Try to capture number of lines in **`tests.R`** as a variable

Don't want to include the file name



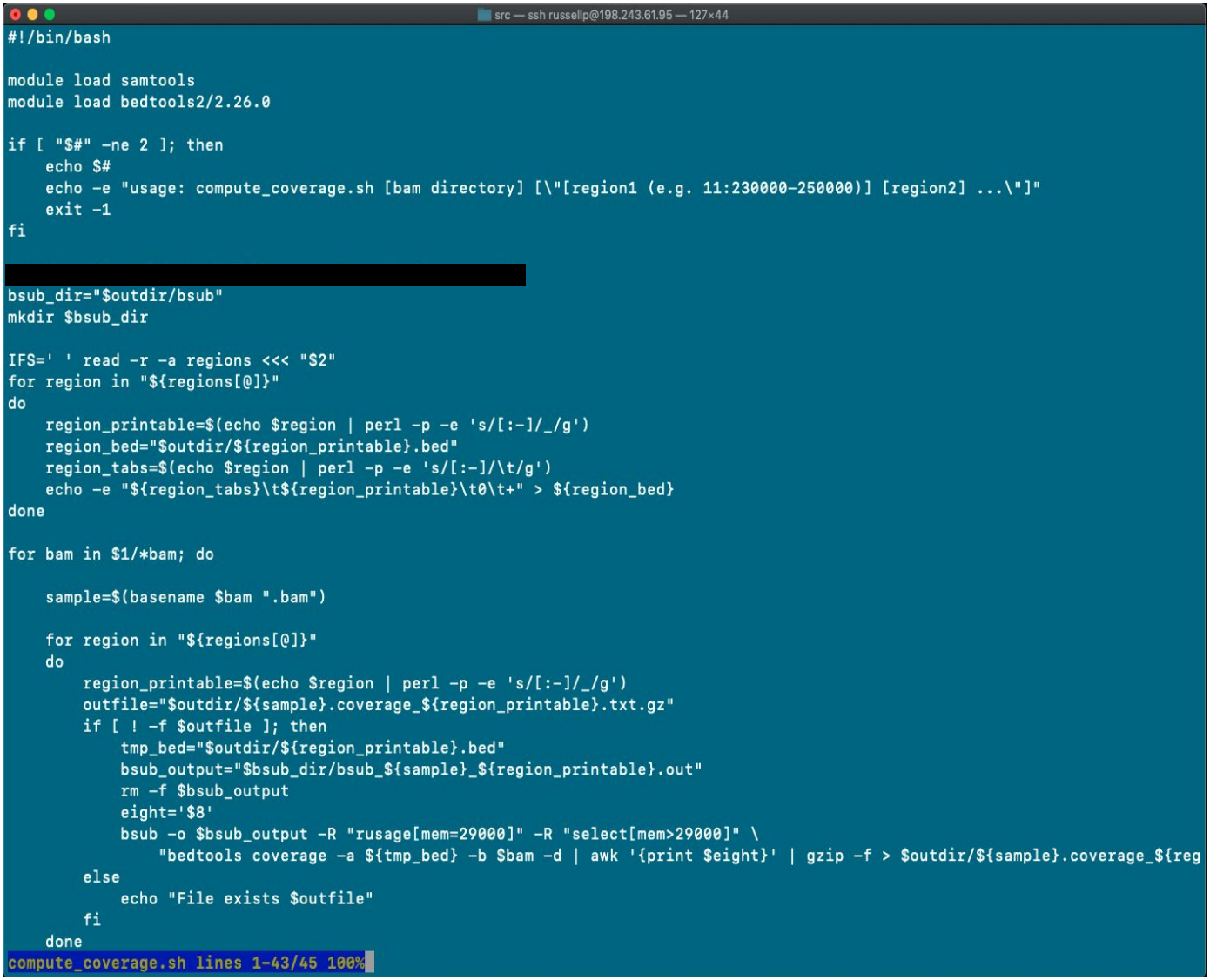
Pipe the original command to an **awk** command that prints the first field of each line of input

Bash scripts for genomics

Get arguments from command line

Do some text processing and write several output files

Loop through all .bam files in a directory and run a genomic program for each



Looping through files in a directory

- **$1** is a special variable holding the first command line argument, a directory in this case
- ***bam** means all files ending with “bam”

```
for bam in $1/*bam; do

    sample=$(basename $bam ".bam")

    for region in "${regions[@]}"
    do
        region_printable=$(echo $region | perl -p -e 's/[:-]/_/g'
        outfile="$outdir/${sample}.coverage_${region_printable}.t
        if [ ! -f $outfile ]; then
            tmp_bed="$outdir/${region_printable}.bed"
            bsub_output="$bsub_dir/bsub_${sample}_${region_printa
            rm -f $bsub_output
            eight='$8'
            bsub -o $bsub_output -R "rusage[mem=29000]" -R "selec
                "bedtools coverage -a ${tmp_bed} -b $bam -d | awk
        else
            echo "File exists $outfile"
        fi
    done

done
```

Why use Bash instead of an R script?

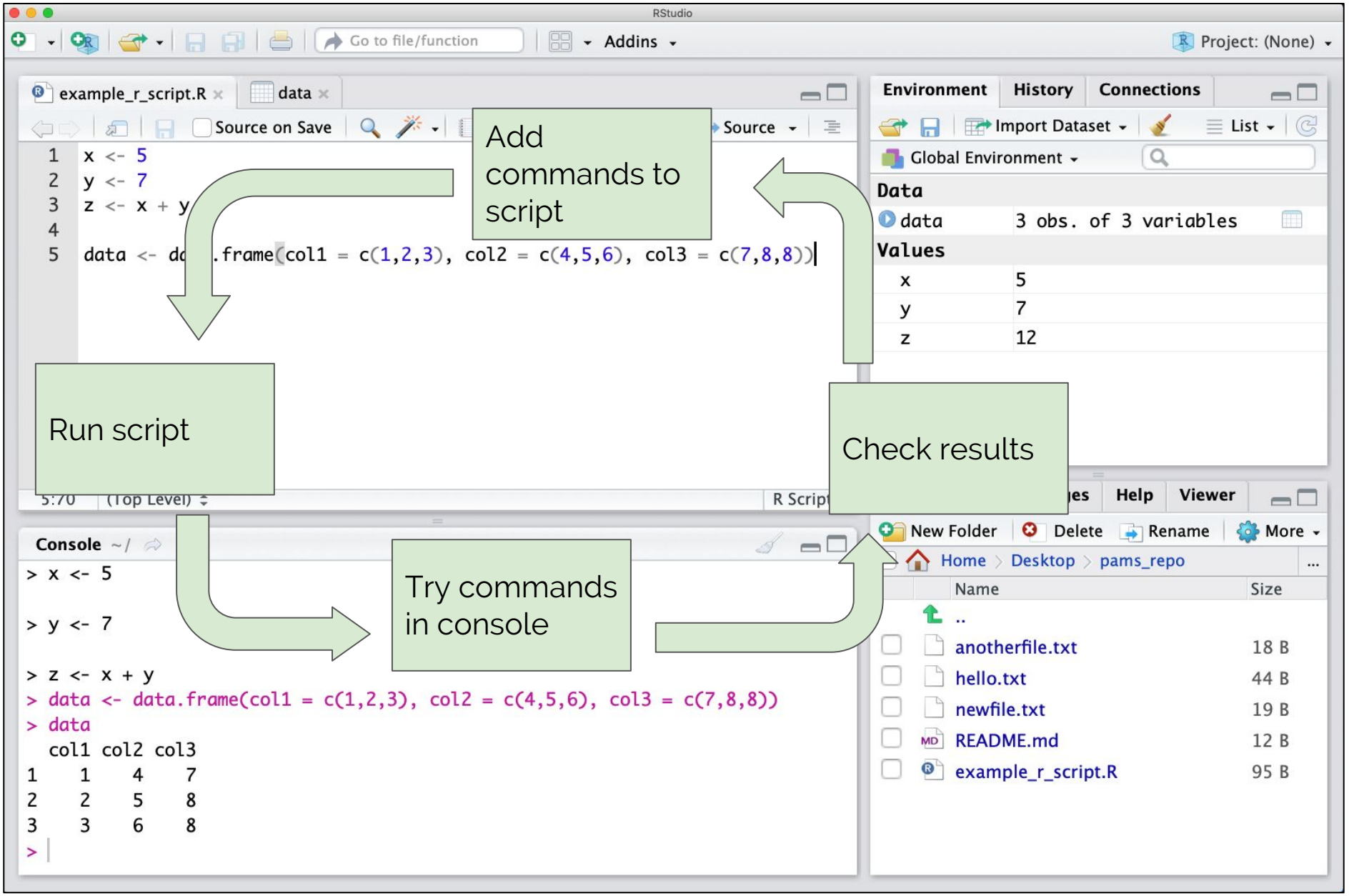
R:

- Has analogs of many Bash commands e.g. **`list.files()`**
- Theoretically can run any Linux command from R (see here)

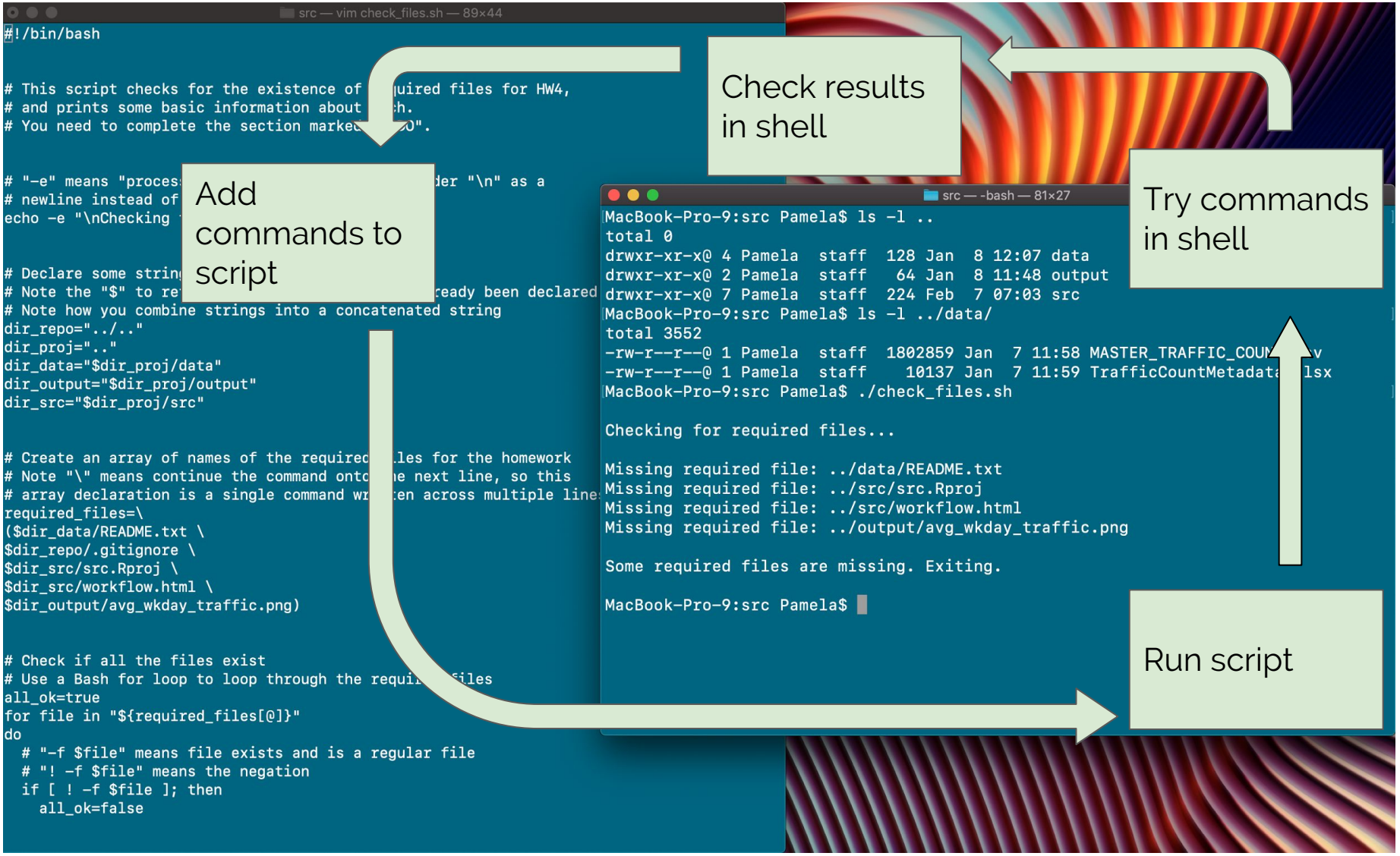
Bash script:

- Direct access to shell environment
  => Efficient development cycle
- Readability
- Integration of command line programs
- Job/process control for complex pipelines

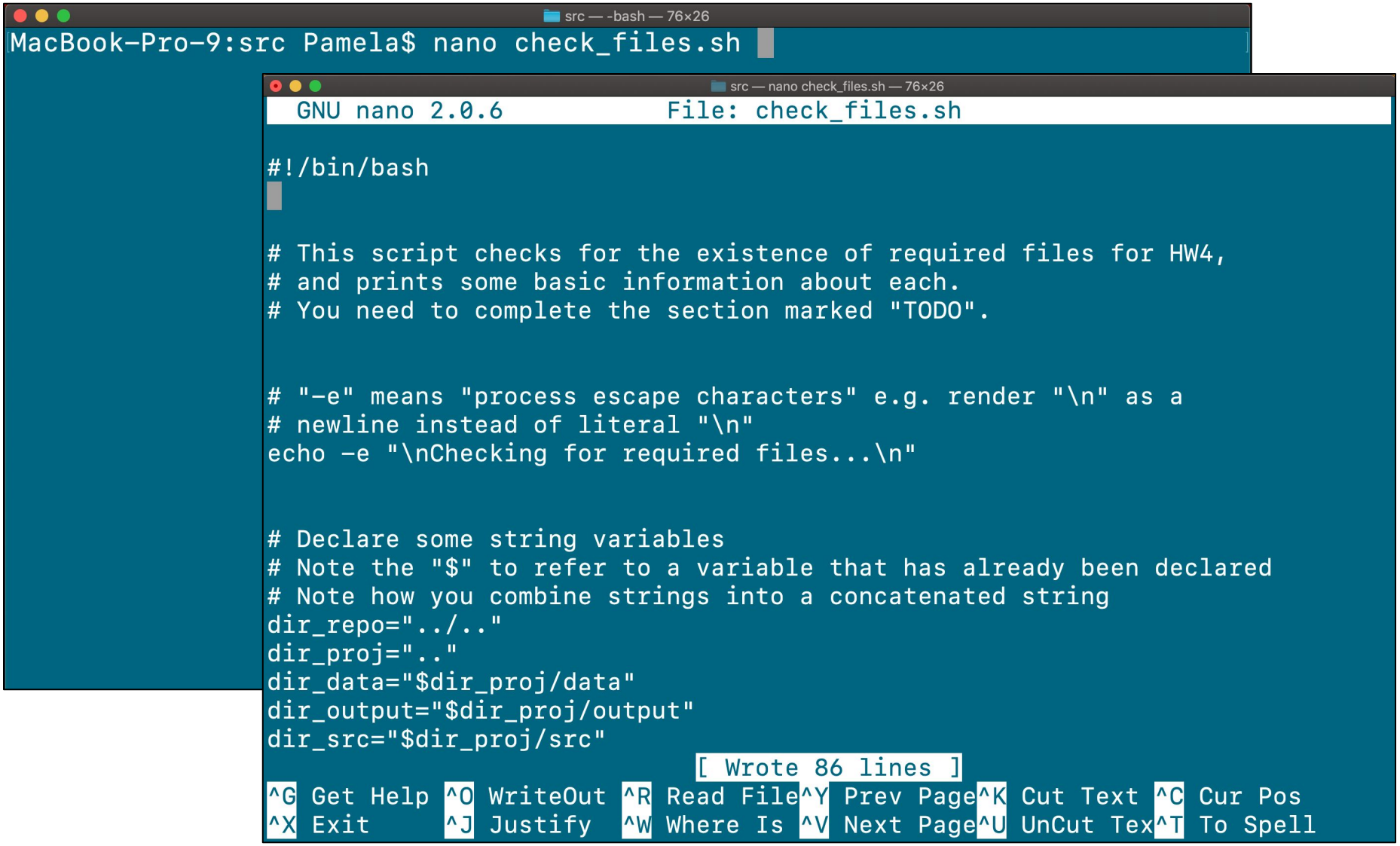
R development cycle (Lecture 3)



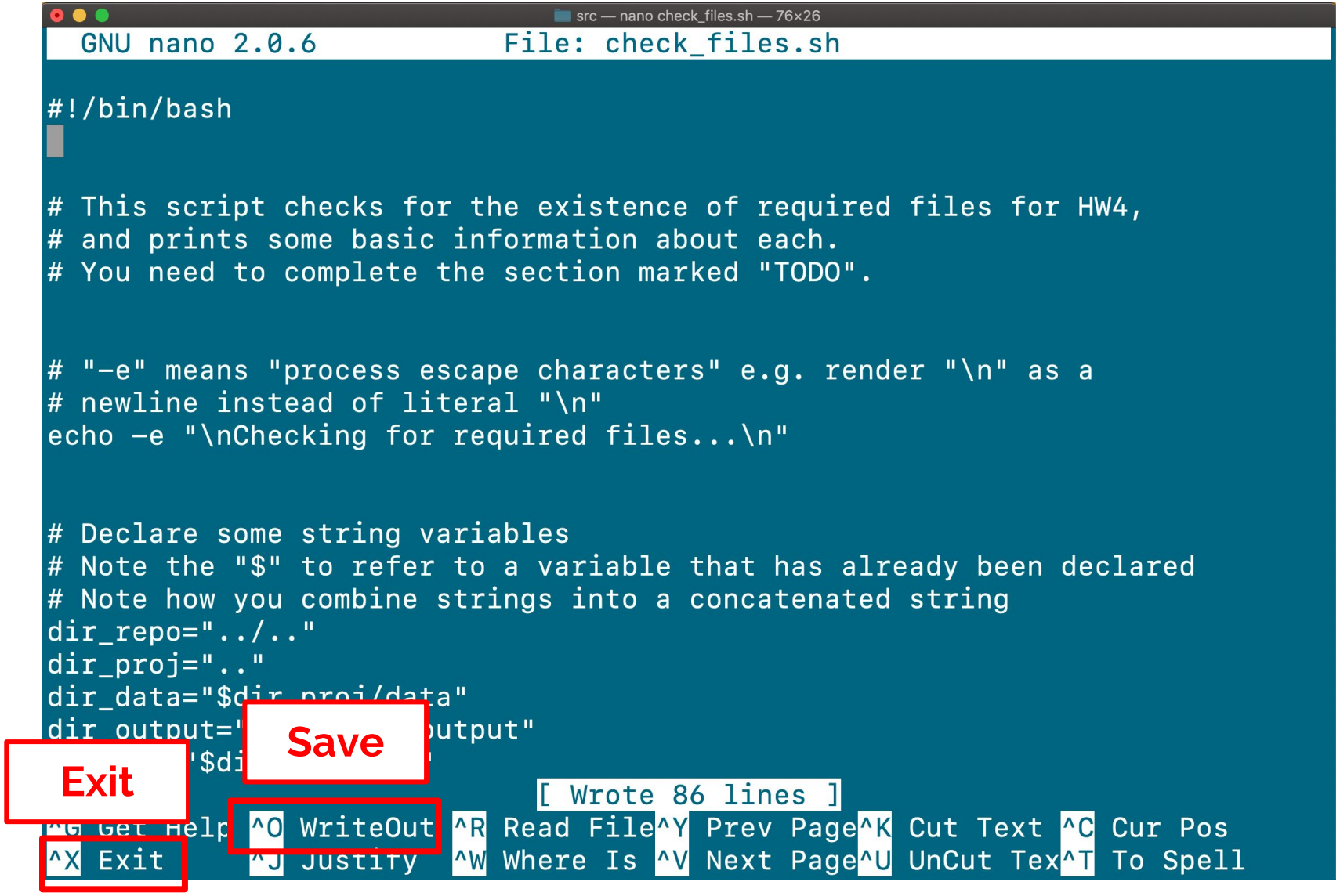
Bash development cycle



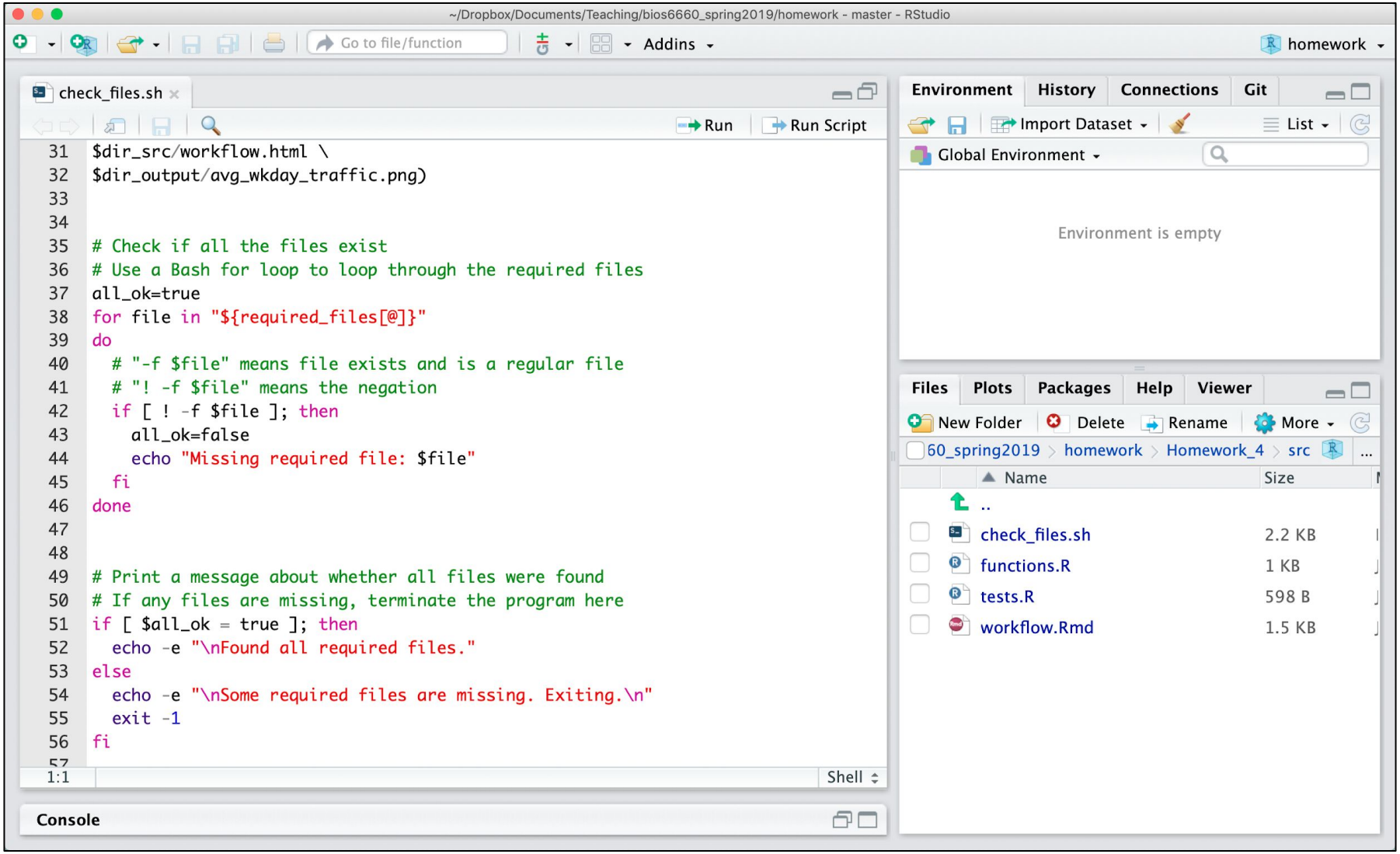
Editing a text file from the shell: Nano editor



Nano controls

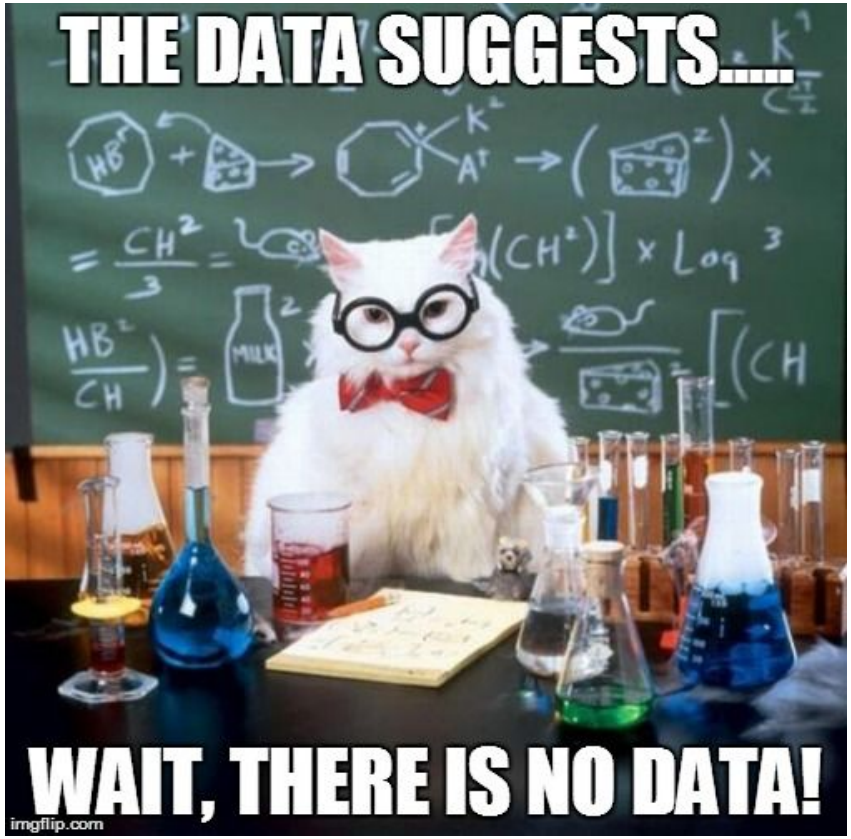


Graphical editors: convenient if you're working on your laptop



Data management

Lecture 8
BIOS 6660, Spring 2019
Instructor: Pam Russell



https://imgflip.com/i/ntnjb

Data: singular or plural?

Guardian style guide:

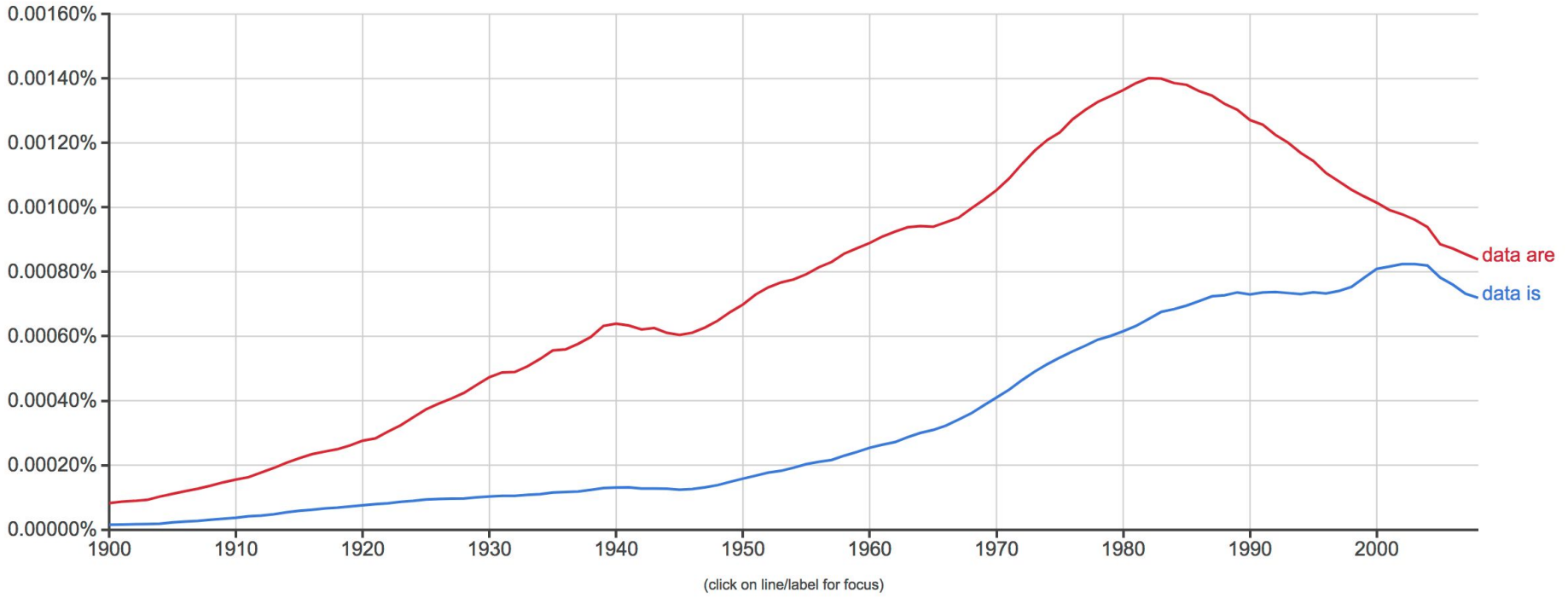
“Data takes a singular verb (like agenda), though strictly a plural; you come across datum, the singular of data, about as often as you hear about an agendum”

https://www.theguardian.com/guardian-observer-style-guide-d

Google Books Ngram Viewer

Graph these comma-separated phrases: data is,data are case-insensitive

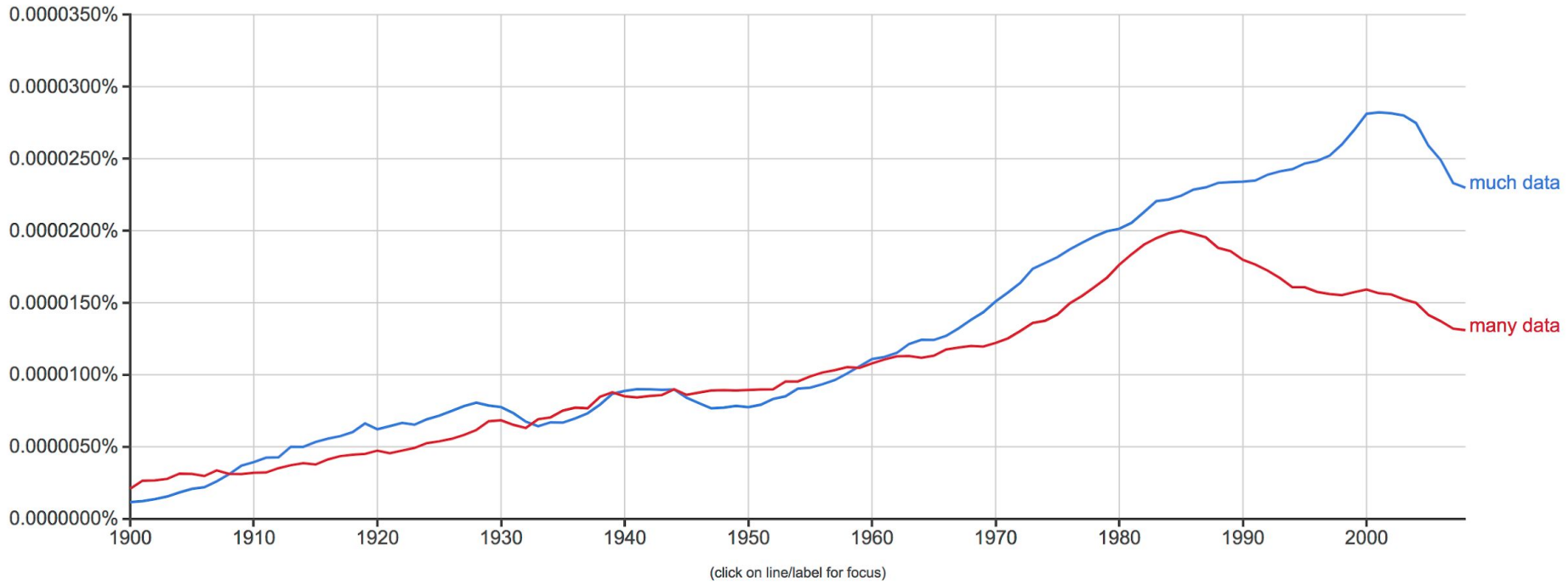
between 1900 and 2008 from the corpus English with smoothing of 3. Search lots of books

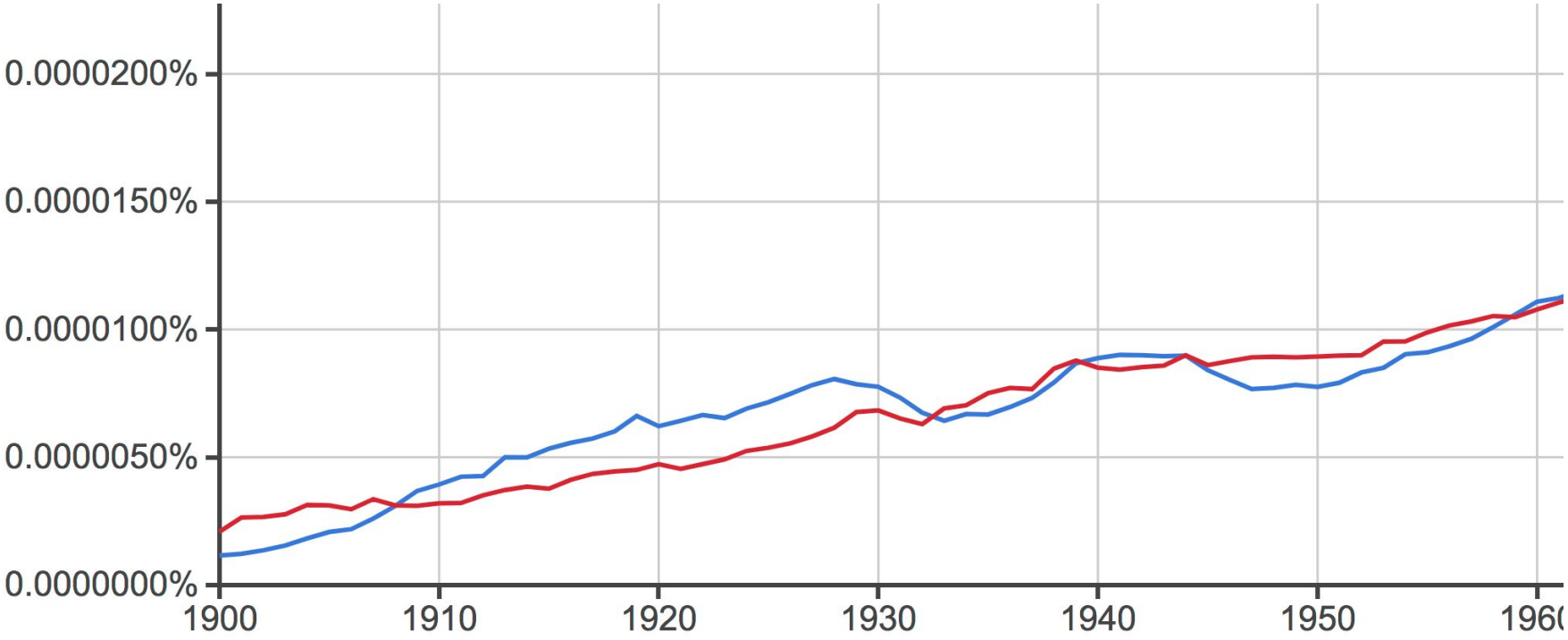


Google Books Ngram Viewer

Graph these comma-separated phrases: much data,many data case-insensitive

between 1900 and 2008 from the corpus English with smoothing of 3. Search lots of books





(click on line/label for focus)

Search in Google Books:

1900 - 1928	1929 - 1991	1992 - 1997	1998 - 2002	2003 - 2008
1900 - 1928	1929 - 1987	1988 - 1993	1994 - 2000	2001 - 2008

Run your own experiment! Raw data is available for download here.

What is data management?

Handling, documentation, sharing, and preservation of research data

Why data management?

For you:

- Data easier to find, use, analyze
- Get credit (citations) for your data
- Minimal effort to share data

For others:

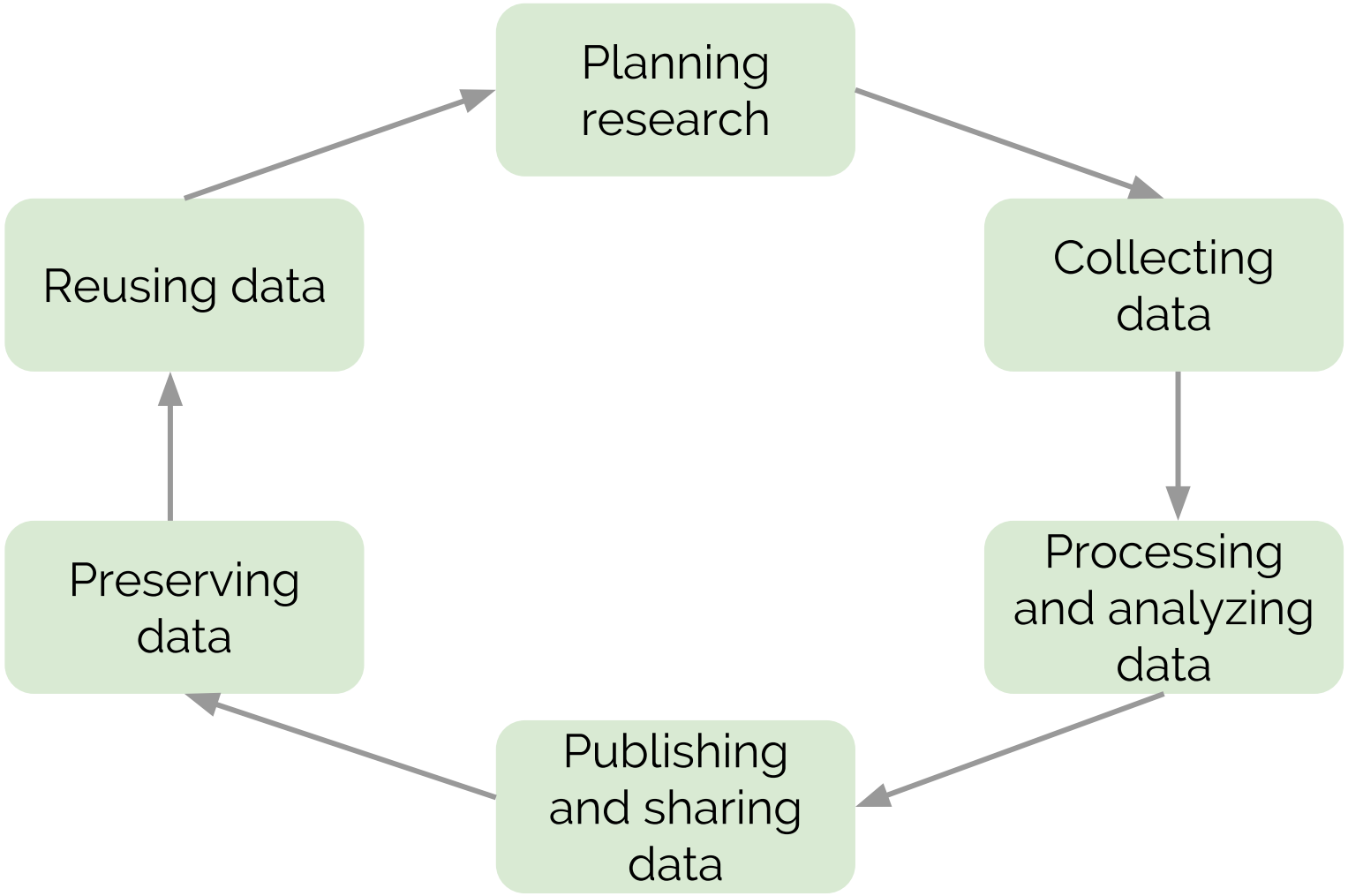
- Easier for collaborators to understand and use
- Other scientists can easily find and use

For society:

- Loss of data deprives society of benefits of research
- Journals and funders increasingly require data management and sharing

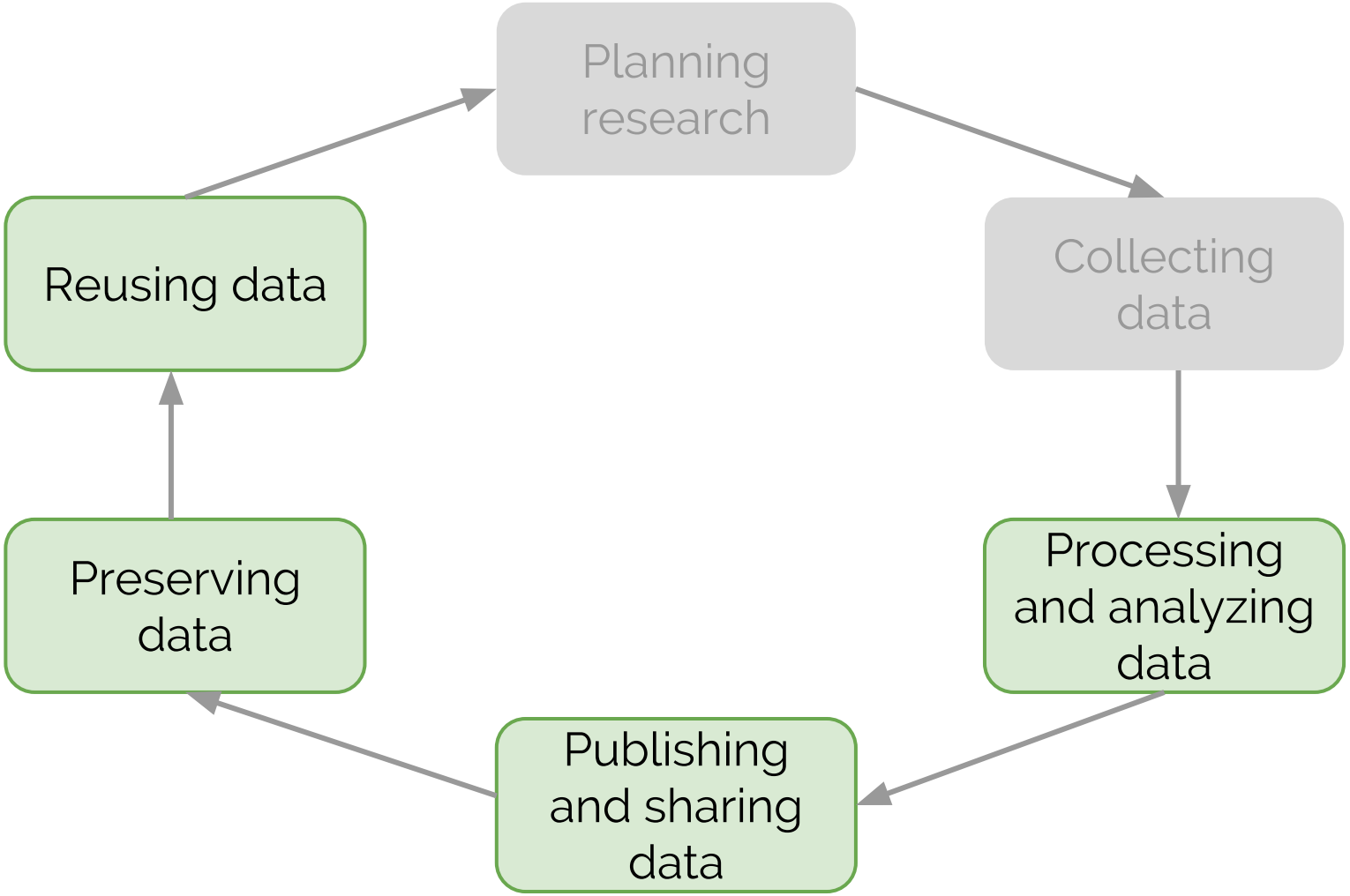
https://www.dataone.org/sites/all/documents/DataONE_BP_Primer_020212.pdf

Research data lifecycle



https://www.ukdataservice.ac.uk/manage-data/lifecycle

Research data lifecycle



Not covered

- Confidentiality
- Ethics

Issues with “anonymous” data: https://datascience.berkeley.edu/anonymous-data/

Data sharing

Making all underlying data available along with publication

In supplemental files, a data repository, or a field-specific database

Why share data?

For you:

- Get credit (citations) for your data
- Feel good about contributing to science

For others:

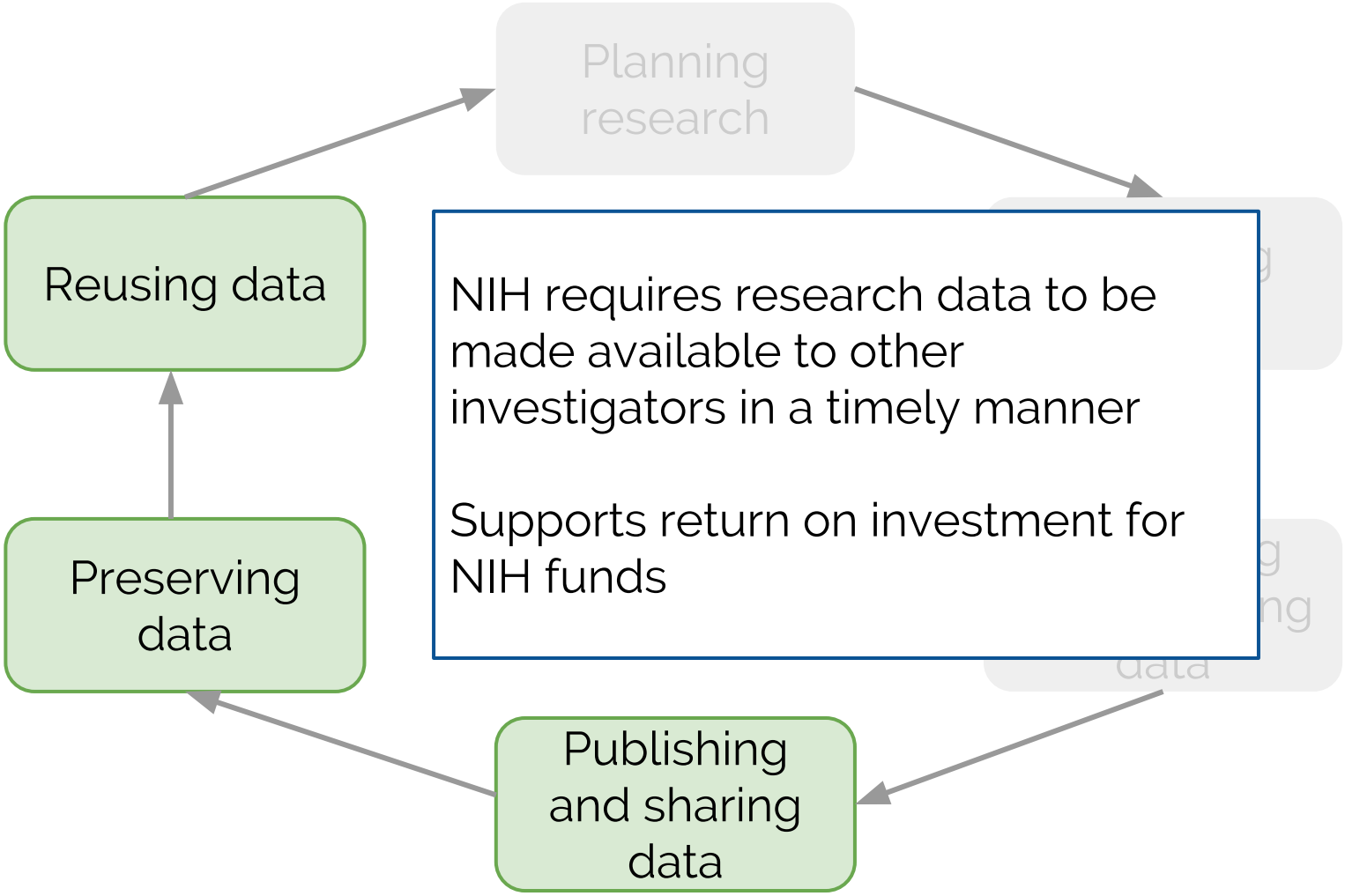
- Other scientists can easily find and use

For society:

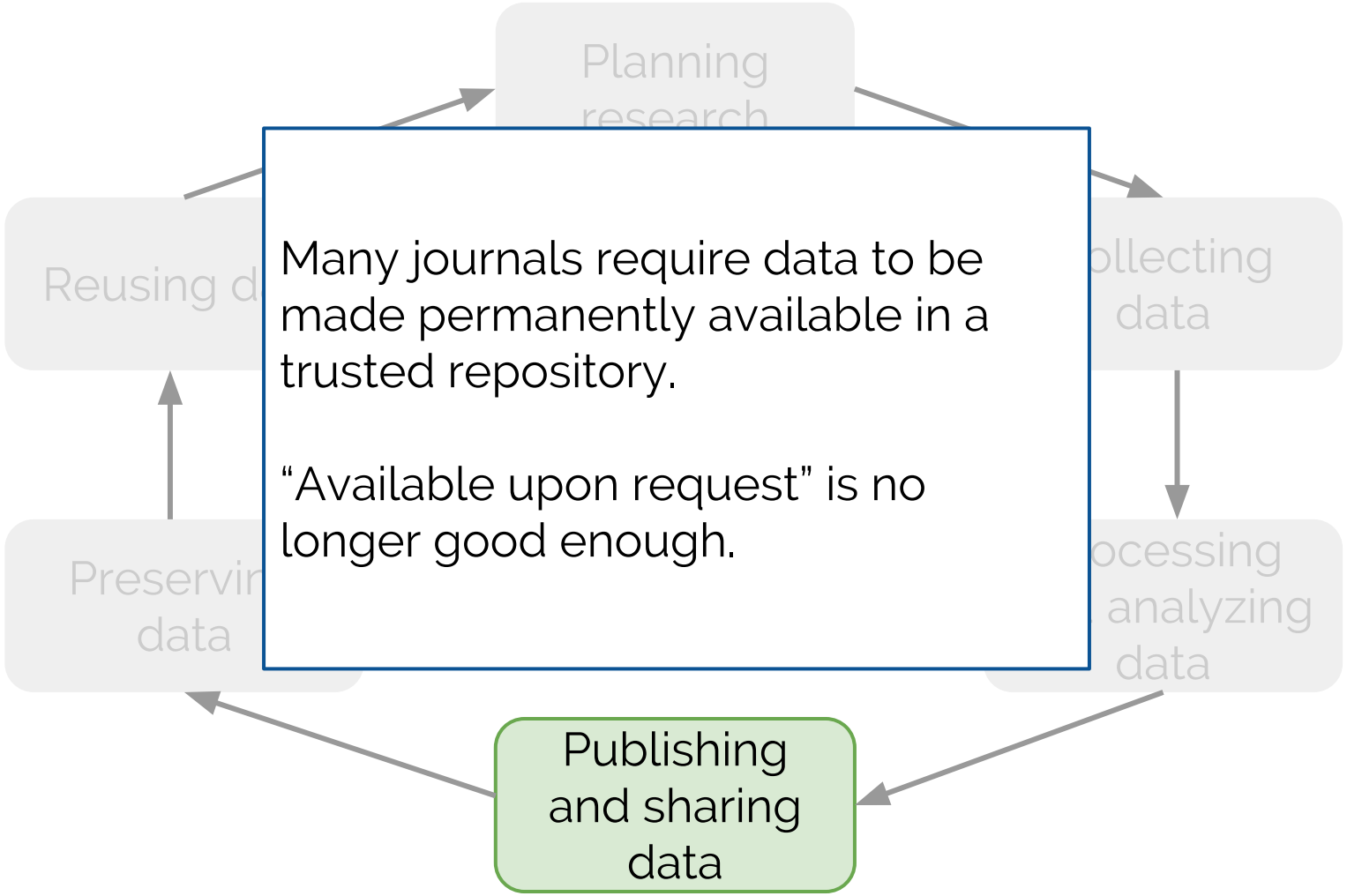
- Increased value from additional use of data
- Journal and funder requirements

https://www.dataone.org/sites/all/documents/DataONE_BP_Primer_020212.pdf

Funding agency requirements



Journal requirements



Data repositories

"A trusted digital repository is one whose mission is to provide reliable, long-term access to managed digital resources to its designated community, now and in the future."

- *Trusted Digital Repositories,* RLG-OCLC report

Types of repositories:

- General purpose
- Disciplinary
- Institutional

Data citation

"Citing research data in a manner similar to traditional scholarly works can help ensure proper attribution, improve reproducibility, improve discoverability, and help provide credit for data as a scholarly output."

- Princeton Research Data Management

https://makeameme.org/meme/if-you-dont-5b09d5
https://libguides.princeton.edu/c.php?g=102546&p=665869

Data citation

Repository provides persistent digital object identifier (DOI)

PANGAEA
Willmes, S et al. (2009): Onset dates of annual snowmelt on Antarctic sea ice in 2007/2008. doi:10.1594/PANGAEA.701380

Dryad
Kingsolver JG, Hoekstra HE, Hoekstra JM, Berrigan D, Vignieri SN, Hill CE, Hoang A, Gibert P, Beerli P (2001) Data from: The strength of phenotypic selection in natural populations. Dryad Digital Repository. doi:10.5061/dryad.166

Dataverse
Frederico Girosi; Gary King, 2006, 'Cause of Death Data', http://hdl.handle.net/1902.1/UOVMCPSWOL UNF:3:9JU+SmVyHgwRhAKclQ85Cg== IQSS Dataverse Network [Distributor] V3 [Version].

Figure 2: Data citation formats suggested by repositories

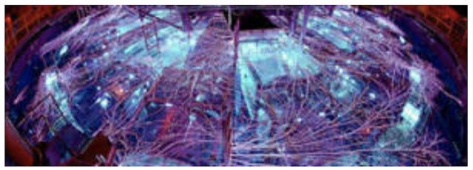
http://www.dcc.ac.uk/resources/how-guides/cite-datasets

Example: PLoS journals

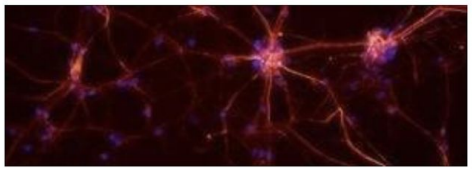
"PLOS is refining the definition, evaluation and recognition of scientific work, expanding toward a more Open Science ethos."

https://www.plos.org/innovation

Example: PLoS journals



PLOS encourages posting preprints to accelerate the dissemination of important results. Our partnership with bioRxiv makes posting life sciences preprints easy and convenient.



With their own experts at the helm, PLOS Channels provide scientific communities a single destination featuring curated research articles, commentary, blogs, news and discussions.

Ambra™, our journal and collections publishing platform and Aperta™, a submission platform developed here at PLOS, are available open source to the community under an MIT License.

Our partnership with Protocols.io enables authors to more effectively share methodological details about their research, supporting reproducability.

A forward-thinking Data Policy fosters reproducibility, transparency and accelerates discovery.

Operating in tandem, ORCID and CRediT ensure that authorship is properly attributed, and each individual authors' contributions are fully described.

https://www.plos.org/innovation

PLoS data policy

"PLOS journals require authors to make all data underlying the findings described in their manuscript fully available without restriction, with rare exception."

https://journals.plos.org/plosone/s/data-availability

Acceptable data sharing methods

Method	Notes
Data deposition	Best method, strongly recommended
Supporting information files	For smaller datasets
Available upon request or from a third party	Only if above methods are unethical or illegal

https://journals.plos.org/plosone/s/data-availability

Recommended data repositories

Cross-disciplinary repositories

- Dryad Digital Repository
- figshare
- Harvard Dataverse Network
- Open Science Framework
- Zenodo

Repositories by type

Biochemistry	**Neuroscience**	**Social Sciences**
Biomedical Sciences	**Omics**	**Structural Databases**
Marine Sciences	**Physical Sciences**	**Taxonomic & Species Diversity**
Model Organisms	**Sequencing**	**Unstructured and/or Large Data**

https://journals.plos.org/plosone/s/data-availability

The Official PLOS Blog

Boosting Open Science Hardware in an academic context: opportunities and challenges

Posted January 17, 2019 by PLOS in Guest blog, Open Science, Publishing

PLOS Board Appointments

Posted January 9, 2019 by Alison Mudditt in In the News, Open Access, Publishing

peer review science publishing discovery collaboration PLOS Journals impact diversity Open Access PLOS featured transparency Open Science advocacy researchers early career researcher preprints recognition bioRxiv science

PLOScasts

Episode List | About This Blog

Search This Blog

Welcome to PLOScast

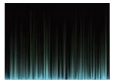
Here is the full list of PLOScasts, podcasts about scholarly publishing, the future of academia, and the changing experiences of scientists.

Explore PLOScast episodes:

- Episode 27: Ivan Oransky on the Value of Tracking Retractions
- Episode 26: How ECRs like Jessica Polka are reinventing science publishing
- Episode 25: Exploring the World of Medical Research: An interview featuring Lauren Maggio
- Episode 24: Check Your Stats: An interview featuring Michèle Nuijten
- Episode 23: Big Data in the Social Sciences: An interview featuring Ian Mulvany
- Episode 22: Building Taxonomies: An interview featuring Bob Kasenchak
- Episode 21: Why Open Research: An interview featuring Erin McKiernan
- Episode 20: Science Communication and Critique: An interview featuring Hilda Bastian
- Episode 19: Accessing Academic Sources on Wikipedia: An interview featuring Jake Orlowitz
- Episode 18: Science of Peer Review: An interview with Eamon Duede Part 2
- Episode 17: The Science of Science Part 1: An interview with Eamon Duede
- Episode 16: Open Source Science Communities: An interview with Abigail Cabunoc Mayes
- Special crossover episode with Open Science Radio
- Episode 15: The Power of Preprints: An interview with James Fraser

Popular Posts

PLOScast Highlights from 2017
posted on December 29, 2017



The History of Scientific...

File naming and organization

"File names should provide context and be easily understood by humans and computers, now and in the future."

- Meghan Frazer

https://acrl.ala.org/techconnect/post/an-elevator-pitch-for-file-naming-conventions/

MIT Libraries recommendations

- Spend time planning folder hierarchy and file naming conventions in the beginning of a project
- Establish appropriate folder hierarchy for the project
- Develop file naming scheme that includes important metadata e.g. [Date]_[Run]_[Sample]
- Consider sorting. Dates should be YYYYMMDD
- Provide a README (more on this next)

https://libraries.mit.edu/data-management/store/organize/

Metadata

Supporting documentation of the data

Without documentation, data can become unusable

Simplest way: keep in a README.txt file in the folder with the data

MIT Libraries recommendations: metadata attributes

- Name of project or dataset
- Who created it
- Key dates
- Subject or content
- Funders
- Any intellectual property rights
- Methodology (how data was generated)
- Explanation of directory and file naming structure

https://libraries.mit.edu/data-management/store/documentation/

File formats

Try not to use proprietary formats (e.g. Microsoft Office)

If you must, then also create a plain text version or other appropriate open format

GitHub and data

Never add protected data to GitHub, even a private repository

You might make the repo public and forget

Git makes it hard to permanently remove something from the history of a repo

Use .gitignore to keep files out of version control

https://dev.to/tschaka1904/show-me-your-gitignore--4415

Private alternatives to GitHub

GitLab is like GitHub but private and internal to organizations

Backups

Best way: institution's drive with regular backups

Don't store master copies on your laptop

External drives not recommended for long term (loss, bit rot)