

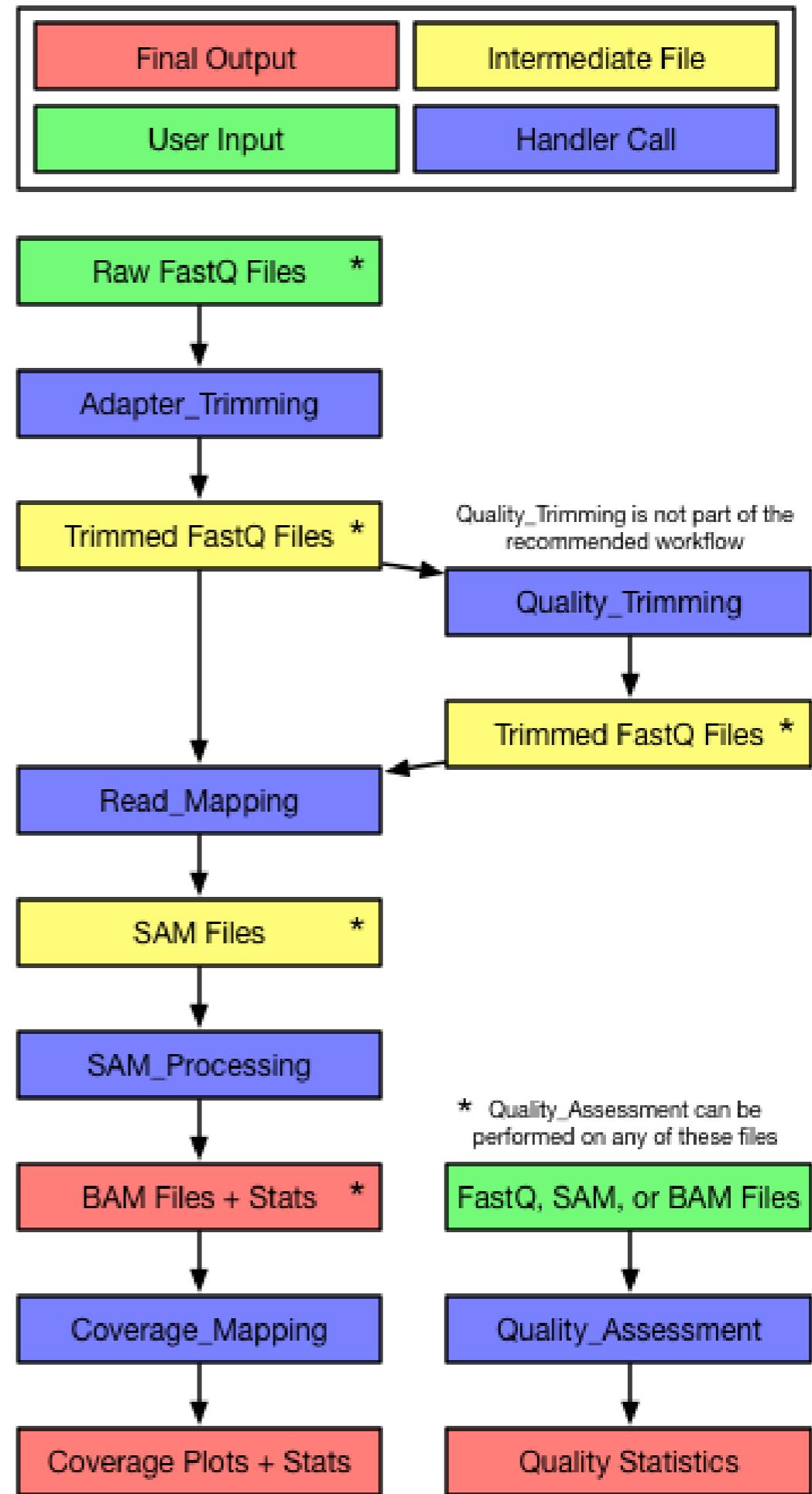


Reproducible Computational based Research

To infinity and beyond

Outline - getting to reproducibility

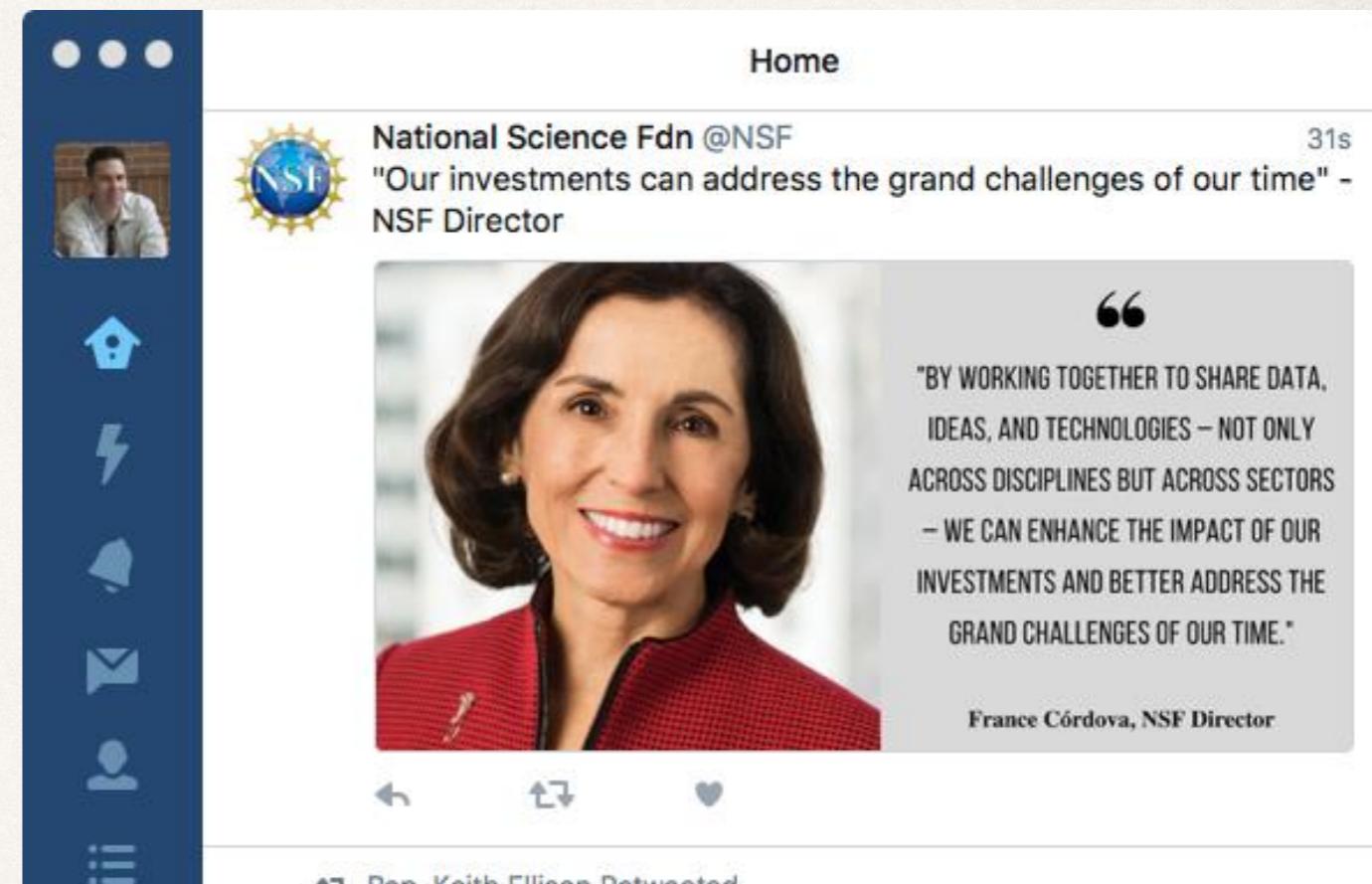
- ❖ What does reproducibility mean?
- ❖ Should I design my data analysis as code?
- ❖ What is data & what is metadata?
- ❖ Should I keep a digital notebook?



Inspirations for talk

- ✿ Yosef Cohen - [Statistics and Data with R](#)
- ✿ Vince Buffalo - [Bioinformatics Data Skills](#)
- ✿ Karl Broman - [Initial steps toward reproducible research](#)
- ✿ Greg Wilson, Titus Brown, et al. - [Best practices for scientific computing](#)
- ✿ Every dissertation, manuscript, or paper I read!

NSF Statement



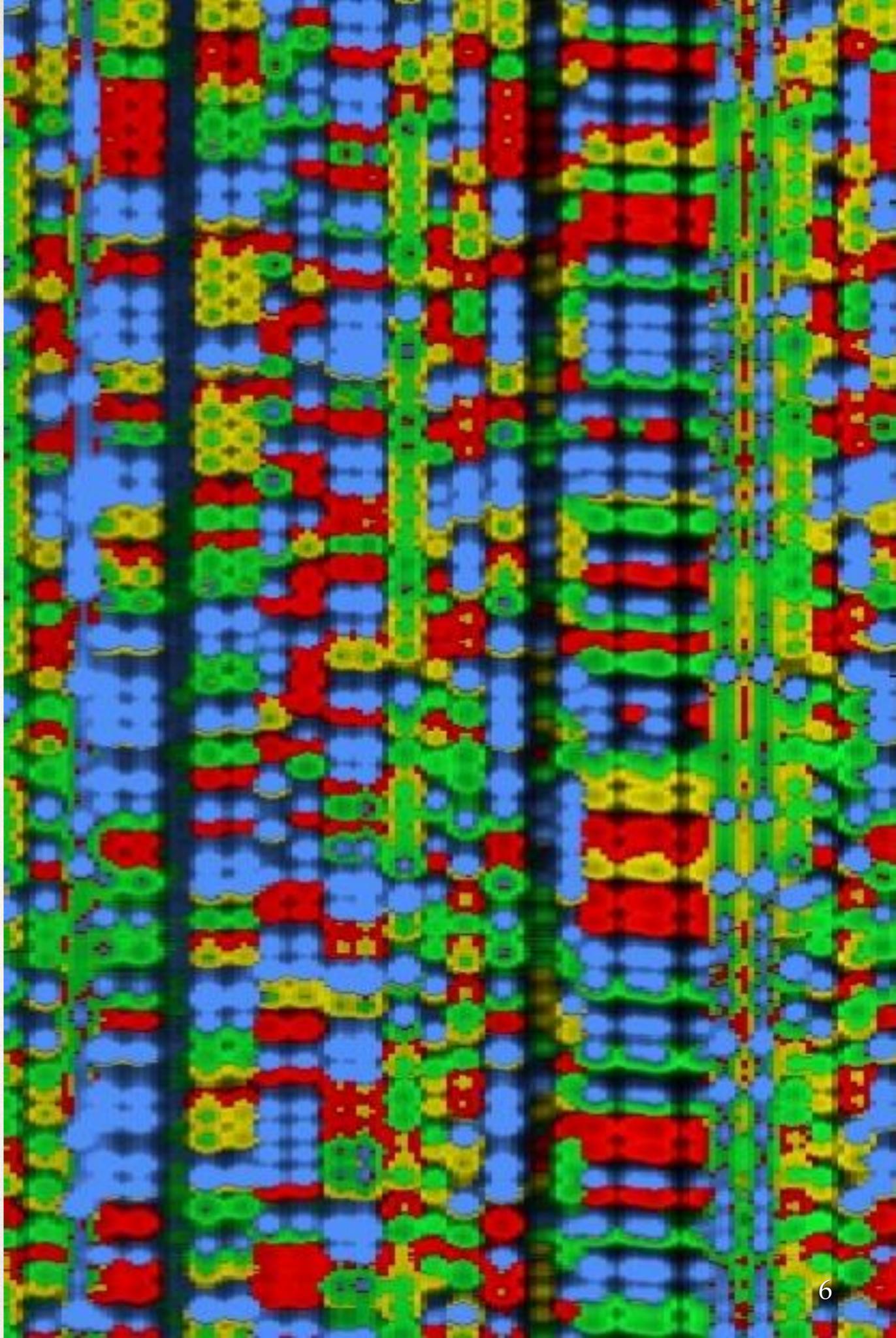
What does reproducible mean?

- ❖ Person “skilled in the art” can reproduce results
- ❖ Or collect new data and get to similar results



Why should research be reproducible?

- ❖ Public investment requires reliable results
- ❖ Research training is NOT intended to teach a boutique set of skills only an artisan can replicate
- ❖ Work is replicated and updated; current human genome is version 38



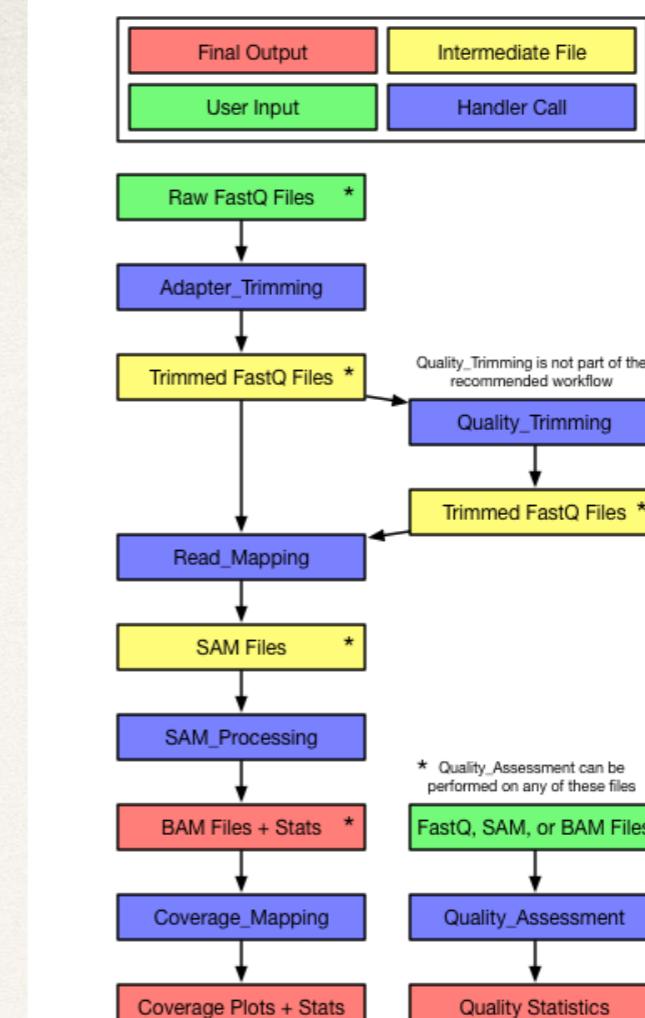
Why this does [not] apply to me?

- ✿ Whenever possible approaches and methods should be “generalizable”
- ✿ “I work on a highly specialized system”
- ✿ “Croak length in subterranean Iberian frogs”

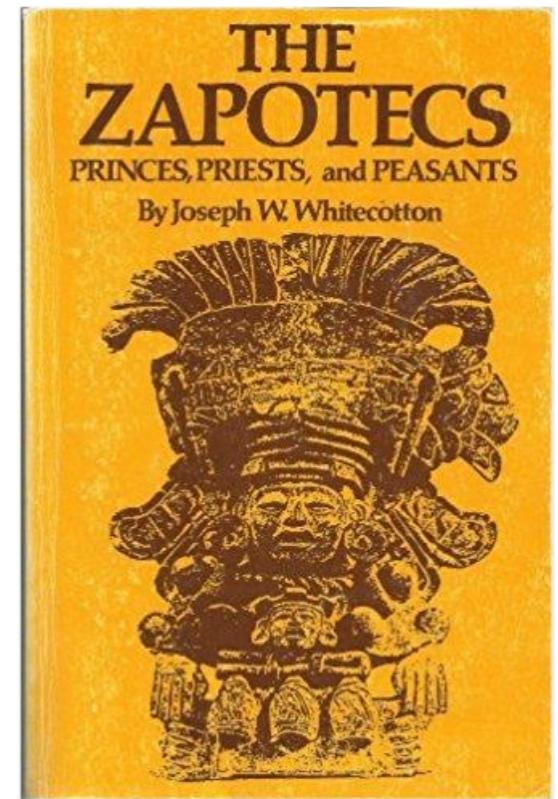


Should I design my analysis as code?

- ❖ Easier said than done
- ❖ Goal should be self-contained code that executes analyses
 - ❖ Hand-editing any file or intermediate step is a no-no
 - ❖ Primarily because you will have to do it repeatedly, and could make mistakes

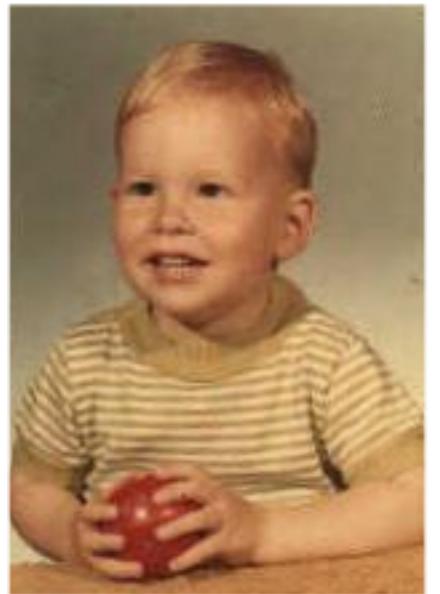


```
Adapter_Trimming.sh
1 #!/bin/bash
2
3 # This script performs adapter trimming
4 # on a series of FASTQ samples using scythe.
5 # Please install scythe before use.
6
7 set -o pipefail
8
9 # What are the dependencies for Adapter_Trimming?
10 declare -a Adapter_Trimming_Dependencies=(scythe parallel)
11
12 # A function to perform the trimming
13 # Adapted from Tom Kono
14 function trimAdapters() {
15     # Set the arguments for the trimming
16     local sample="$1" # What sample are we working with?
17     local out="$2" # Outdirectory
18     local adapters="$3" # Adapter file
19     local prior="$4" # Prior
20     local platform="$5" # Quality encoding platform
21     local forwardNaming="$6" # What is the forward naming scheme?
22     local reverseNaming="$7" # What is the reverse naming scheme?
23     # Check to see if we have a forward or reverse sample
24     if [[ ! -z "${forwardNaming}" && ${echo "$sample"} | grep "${forwardNaming}" ]] # Is this a forward sample?
25     then # If yes
26         local name="${basename ${sample} ${forwardNaming}}_Forward # Make the name say forward
27     elif [[ ! -z "${reverseNaming}" && ${echo "$sample"} | grep "${reverseNaming}" ]] # Is this the reverse sample?
28     then # If yes
29         local name="${basename ${sample} ${reverseNaming}}_Reverse # Make the name say reverse
30     else # If this is neither
31         local name="${basename ${sample} | cut -f 1 -d '.'}_Single # Make the name say single
32     fi
33     # Make the outdirectory
34     mkdir -p "$out"
35     # Is our sample compressed?
36     if [[ ${echo ${sample} | rev | cut -f 1 -d '.' | rev} == "gz" ]] # Is this gzipped?
37     then # If so
38         local toTrim="${out}/${name}_PIPE" # Make a name for the pipe which will be passed to the trimmer
39         rm -f "${toTrim}" # Remove any pipes with the same name
40         mknod "${toTrim}" # Make the pipe
41         gzip -cd "${sample}" > "${toTrim}" & # Uncompress our sample to the pipe
42     elif [[ ${echo ${sample} | rev | cut -f 1 -d '.' | rev} == "bz2" ]] # Is this bzipped?
43     then # If so
44         local toTrim="${out}/${name}_PIPE" # Make a name for the pipe which will be passed to the trimmer
45         rm -f "${toTrim}" # Remove any pipes with the same name
46         mknod "${toTrim}" # Make the pipe
47         bzip2 -cd "${sample}" > "${toTrim}" & # Uncompress our sample to the pipe
48     else # Otherwise
49         local toTrim="${sample}" # Our name will be the sample itself
50     fi
}
13 errors, Line 1, Column 1
Space
```



“You all grew up in Latin America, you just don’t know it yet.”

*–Joseph Whitecotton - anthropologist - to U. of Oklahoma
undergrads*

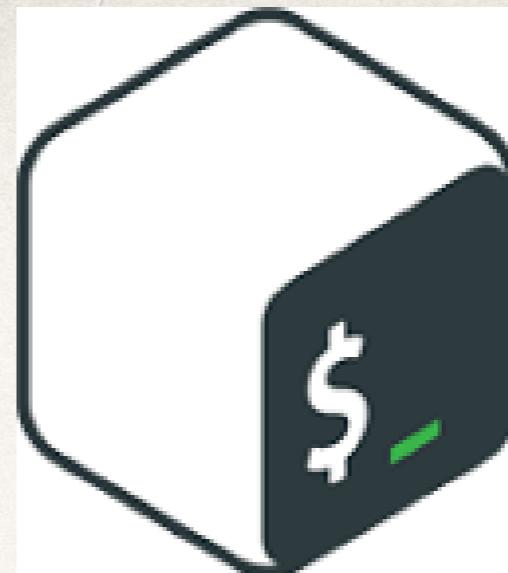


“You are all computational biologists, you just don’t know it yet.”

–Peter Morrell - biologist - to seminar audience at U. of Minnesota

Transition to executable analysis

- ✿ Often includes code in Perl, Python, R, BASH, C++ or a combination of languages
- ✿ Elegance of design and simplicity of interpretation increase with experience



BASH
THE BOURNE-AGAIN SHELL

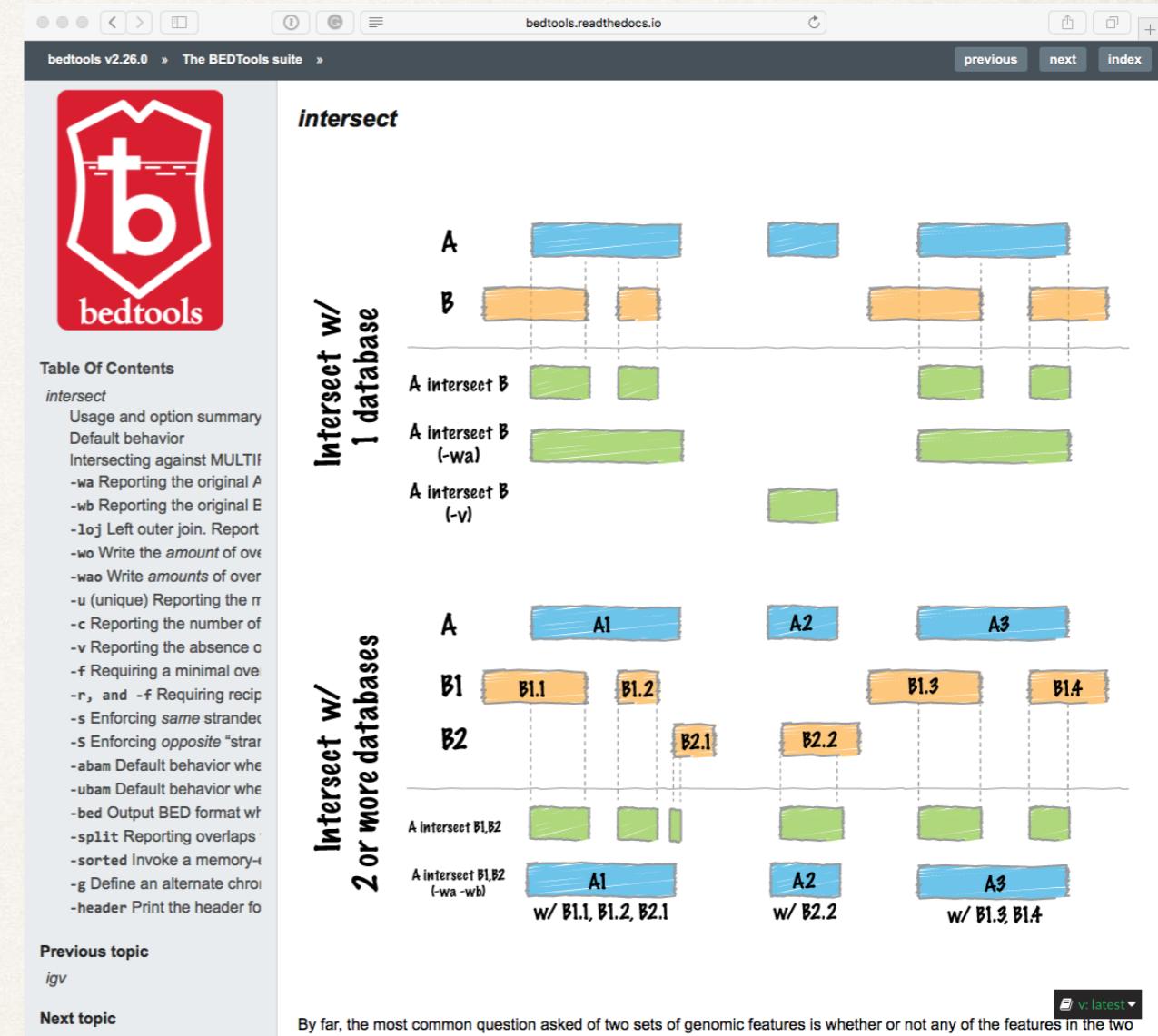


Reproducibility starts with good tools that are well vetted

- ✿ “I wrote my own sequence read mapper” - bad idea
- ✿ Unless you are a computer science grad student
- ✿ Even then, bad idea because there are dozens already
- ✿ Testing of these tools is crowd sourced

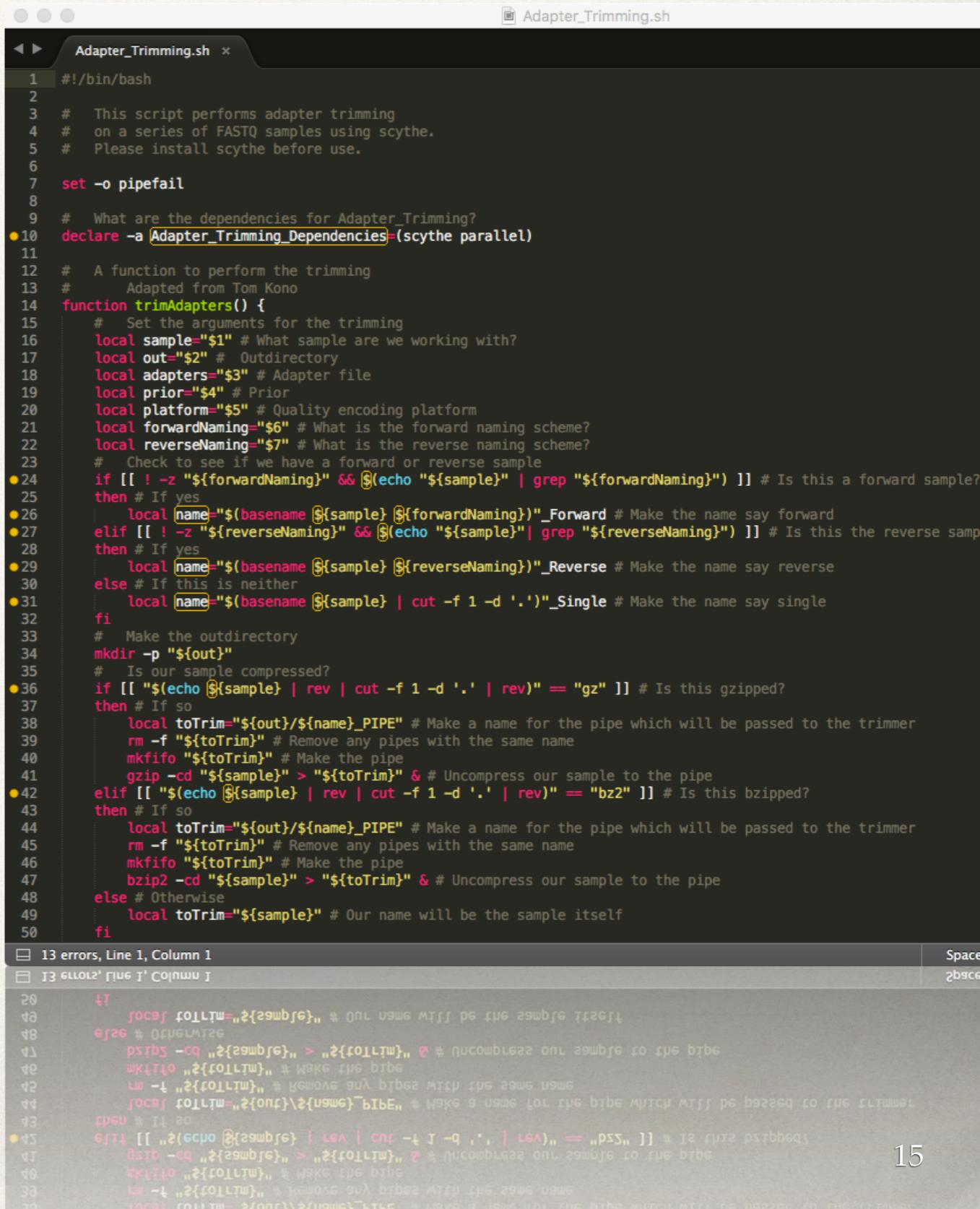
Power tools

- ✿ Primary tools for most analyses are developed by professional programmers and statisticians
- ✿ Use them whenever possible!
 - ✿ BWA - sequence read mapping
 - ✿ R/QTL - biparental QTL mapping
 - ✿ Bedtools - for genome slicing and dicing



Good code is well commented

- ❖ Comments should explain the “why” of a step taken in code
- ❖ Should be sufficient for anyone “skilled in the art”
- ❖ Avoid “!@##\$!@#*%” in comments; code eventually becomes public!



```
Adapter_Trimming.sh
#!/bin/bash

# This script performs adapter trimming
# on a series of FASTQ samples using scythe.
# Please install scythe before use.

set -o pipefail

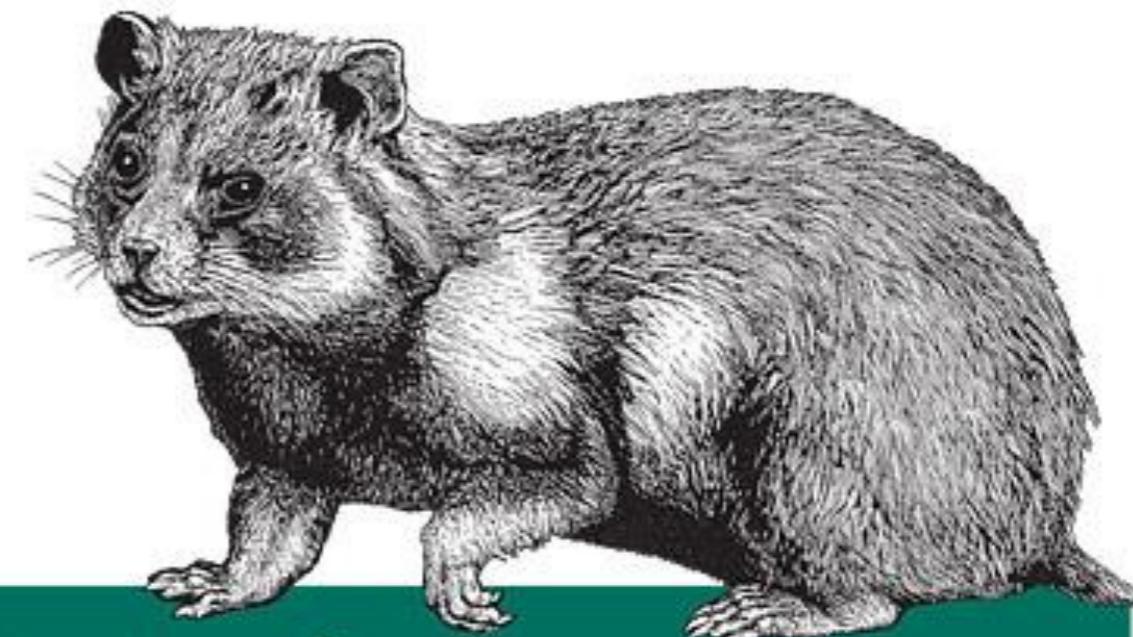
# What are the dependencies for Adapter_Trimming?
declare -a Adapter_Trimming_Dependencies=(scythe parallel)

# A function to perform the trimming
# Adapted from Tom Kono
function trimAdapters() {
    # Set the arguments for the trimming
    local sample="$1" # What sample are we working with?
    local out="$2" # Outdirectory
    local adapters="$3" # Adapter file
    local prior="$4" # Prior
    local platform="$5" # Quality encoding platform
    local forwardNaming="$6" # What is the forward naming scheme?
    local reverseNaming="$7" # What is the reverse naming scheme?
    # Check to see if we have a forward or reverse sample
    if [[ ! -z "${forwardNaming}" && $(echo "${sample}" | grep "${forwardNaming}") ]] # Is this a forward sample?
    then # If yes
        local name="${basename ${sample} ${forwardNaming}}_Forward # Make the name say forward
    elif [[ ! -z "${reverseNaming}" && $(echo "${sample}" | grep "${reverseNaming}") ]] # Is this the reverse sample?
    then # If yes
        local name="${basename ${sample} ${reverseNaming}}_Reverse # Make the name say reverse
    else # If this is neither
        local name="${sample} | cut -f 1 -d '.'}_Single # Make the name say single
    fi
    # Make the outdirectory
    mkdir -p "$out"
    # Is our sample compressed?
    if [[ "$(echo ${sample} | rev | cut -f 1 -d '.' | rev)" == "gz" ]] # Is this gzipped?
    then # If so
        local toTrim="${out}/${name}_PIPE" # Make a name for the pipe which will be passed to the trimmer
        rm -f "${toTrim}" # Remove any pipes with the same name
        mknod "${toTrim}" # Make the pipe
        gzip -cd "${sample}" > "${toTrim}" & # Uncompress our sample to the pipe
    elif [[ "$(echo ${sample} | rev | cut -f 1 -d '.' | rev)" == "bz2" ]] # Is this bzipped?
    then # If so
        local toTrim="${out}/${name}_PIPE" # Make a name for the pipe which will be passed to the trimmer
        rm -f "${toTrim}" # Remove any pipes with the same name
        mknod "${toTrim}" # Make the pipe
        bzip2 -cd "${sample}" > "${toTrim}" & # Uncompress our sample to the pipe
    else # Otherwise
        local toTrim="${sample}" # Our name will be the sample itself
    fi
}

13 errors, Line 1, Column 1
13 errors, Line 1, Column 1
```

If you only learn one language - BASH

- ❖ Unix shell (BASH) scripting provides access to pro tools
- ❖ Necessary to use a supercomputer
- ❖ Bioinformatics Data Skills - Chapters 3 & 7



Bioinformatics Data Skills

REPRODUCIBLE AND ROBUST RESEARCH WITH OPEN SOURCE TOOLS

Example shell script

- ✿ Avoid hard coding file paths in the middle of a script
- ✿ Hard coding nearly eliminates reproducibility
- ✿ Set paths as variable names at the top of the script
- ✿ WORKSHOP=/panfs/roc/groups/9/morrellp/pmorrell/Workshop

```
#!/bin/sh

#PBS -l mem=128gb, nodes=1:ppn=8, walltime=72:00:00
#PBS -m abe
#PBS -M useremail@umn.edu
#PBS -q mesabi

#+ Required for Java
module load java/jdk1.8.0_45

#+ Path to the GATK
GATK=/panfs/roc/groups/9/morrellp/shared/Software/GATK-3.6/GenomeAnalysisTK.jar
#+ Build the sample list
SAMPLE_LIST=/panfs/roc/groups/9/morrellp/shared/Projects/Barley_NAM_Parents/sequence_handling/SAM_
Processing/Picard/Finished/180_bam.list
#+ The output targets file
RTC_OUT=/panfs/roc/groups/9/morrellp/shared/Projects/Barley_NAM_Parents/SNP_calling/realign.
intervals
#+ The reference sequence
REF=/panfs/roc/groups/9/morrellp/shared/References/Reference_Sequences/Barley/Morex/barley_RefSeq_
v1.0/barley_pseudomolecules_parts.fa

#+ Put them into a format that will be accepted by the GATK command line
GATK_IN=()
for s in "${SAMPLE_LIST[@]}"
do
    GATK_IN+=("-I $s")
done

#+ JAVA OPTIONS
#+ -Xmx[amount] : use [amount] of memory.
#+ -jar <file> : execute <file>, which is a jar file

#+ GATK OPTIONS
#+ -T RealignerTargetCreator
#+   Create a list of regions to realign
#+ -L Regions
#+   Operate only in the genomic intervals specified in this regions file
#+ -nct <int>
#+   Use <int> CPU cores
#+   NOTE: This option can make performance WORSE if the system is
#+   IO-limited, and not compute-limited.
#+ -R <FASTA file>
#+   Where the reference sequence is stored
#+ -I <BAM file>
#+   The BAM file for which to create intervals
#+ -known <file>
#+   <file> contains known INDELS. VCF or BED format
#+ -o <file>
#+   Write the intervals to this file
#+   NOTE: This file must end in .list, .intervals, or .interval_list

export _JAVA_OPTIONS="-Xmx127g -Djava.io.tmpdir=${HOME}/tmp"
java -jar ${GATK}\
-T RealignerTargetCreator\
-nt 1\
-R ${REF}\
${GATK_IN[@]}\
-o ${RTC_OUT}
```

What is a concurrent versioning system?

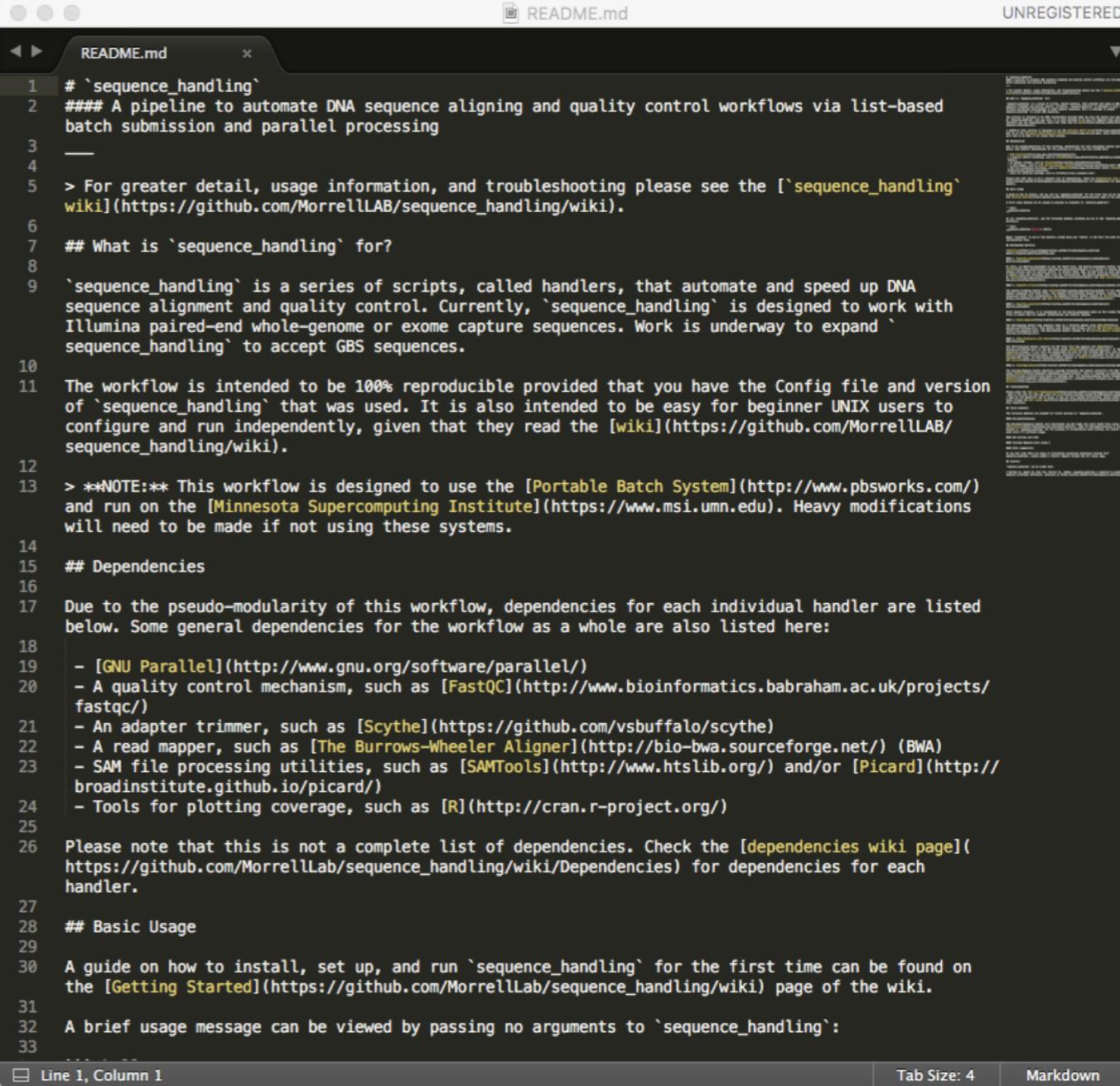
- ✿ Track files and changes over versions of the files
- ✿ Why is it relevant for my system?
- ✿ Remember, “I work on croak length of subterranean frogs”
- ✿ Rerunning analysis during manuscript revision, on a similar project years later, or by another group analyzing rooster crows

Git & Github are preferred options

- ✿ Git repositories can be local or shared on the internet
- ✿ Flaky WiFi and poor internet connectivity are not friendly for remote work
- ✿ Github stores repositories, can be accessed through a web browser or cloned to another computer (or server account)
- ✿ Alternatives include Bitbucket and UMN Github

What types of files belong in a repository?

- ❖ Plain text (with Markdown) is preferred format
- ❖ Code, readme files, workflow diagrams all belong in a repository – example
[https://github.com/MorrellLAB/sequence handling](https://github.com/MorrellLAB/sequence_handling)
- ❖ Data generally belong elsewhere



The screenshot shows a GitHub repository page with a file named 'README.md'. The file content is as follows:

```
1 # `sequence_handling`
2 ##### A pipeline to automate DNA sequence aligning and quality control workflows via list-based
batch submission and parallel processing
3
4
5 > For greater detail, usage information, and troubleshooting please see the [`sequence_handling`]
wiki](https://github.com/MorrellLAB/sequence_handling/wiki).
6
7 ## What is `sequence_handling` for?
8
9 `sequence_handling` is a series of scripts, called handlers, that automate and speed up DNA
sequence alignment and quality control. Currently, `sequence_handling` is designed to work with
Illumina paired-end whole-genome or exome capture sequences. Work is underway to expand `sequence_handling` to accept GBS sequences.
10
11 The workflow is intended to be 100% reproducible provided that you have the Config file and version
of `sequence_handling` that was used. It is also intended to be easy for beginner UNIX users to
configure and run independently, given that they read the [wiki](https://github.com/MorrellLAB/
sequence_handling/wiki).
12 > **NOTE:** This workflow is designed to use the [Portable Batch System](http://www.pbsworks.com/)
and run on the [Minnesota Supercomputing Institute](https://www.msi.umn.edu). Heavy modifications
will need to be made if not using these systems.
13
14 ## Dependencies
15
16 Due to the pseudo-modularity of this workflow, dependencies for each individual handler are listed
below. Some general dependencies for the workflow as a whole are also listed here:
17
18 - [GNU Parallel](http://www.gnu.org/software/parallel/)
19 - A quality control mechanism, such as [FastQC](http://www.bioinformatics.babraham.ac.uk/projects/
fastqc/)
20 - An adapter trimmer, such as [Scythe](https://github.com/vsbuffalo/scythe)
21 - A read mapper, such as [The Burrows-Wheeler Aligner](http://bio-bwa.sourceforge.net/) (BWA)
22 - SAM file processing utilities, such as [SAMTools](http://www.htslib.org/) and/or [Picard](http://
broadinstitute.github.io/picard/)
23 - Tools for plotting coverage, such as [R](http://cran.r-project.org/)
24
25
26 Please note that this is not a complete list of dependencies. Check the [dependencies wiki page](
https://github.com/MorrellLab/sequence_handling/wiki/Dependencies) for dependencies for each
handler.
27
28 ## Basic Usage
29
30 A guide on how to install, set up, and run `sequence_handling` for the first time can be found on
the [Getting Started](https://github.com/MorrellLab/sequence_handling/wiki) page of the wiki.
31
32 A brief usage message can be viewed by passing no arguments to `sequence_handling`:
33
```

“Code base” repository

- ❖ A Git/Github repository used across multiple projects
- ❖ Should be generalizable and extensible
- ❖ Changes relatively slowly
- ❖ Are often public during development

The screenshot shows a GitHub repository page for 'MorrellLAB / sequence_handling'. The repository has 449 commits, 3 branches, and 1 release. It was last updated 4 days ago. The repository description is: "A series of scripts to automate sequence workflows". The repository is forked from 'pmorrell/sequence_handling'. The repository page includes a list of recent commits, a 'sequence_handling' section with a description, and a link to the 'sequence_handling' wiki.

A series of scripts to automate sequence workflows

449 commits 3 branches 1 release 5 contributors

This branch is 442 commits ahead of pmorrell:master.

Aerin13 Bug fixes .github Handlers HelperScripts .Sequence_Handling_Workflow.png .gitignore Config README.md sequence_handling README.md

Update and clean up the issue template Add Genotype_GVCFs Add Haplotype_Caller.sh Update workflow diagram Fix output directory issue with Quality_Assessment Add Genotype_GVCFs Update README.md Bug fixes

Latest commit 05910f5 4 days ago 2 months ago 5 days ago a month ago 2 months ago 5 days ago a month ago 4 days ago

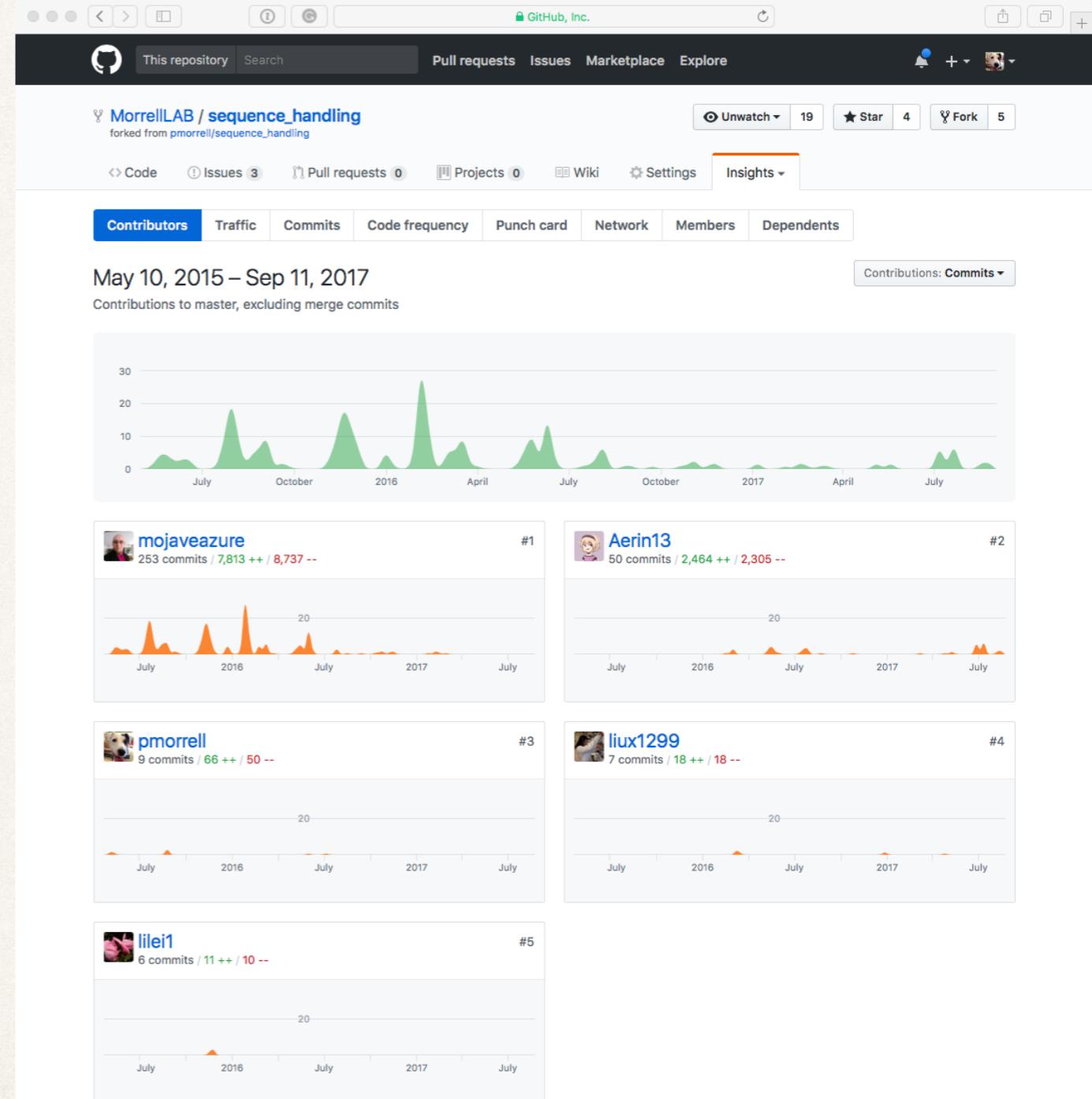
sequence_handling

A pipeline to automate DNA sequence aligning and quality control workflows via list-based batch submission and parallel processing

For greater detail, usage information, and troubleshooting please see the [sequence_handling wiki](#).

“Code base” repository

- ❖ Multiple users contribute
- ❖ Active contributors change over time



Project repository

- ❖ Git/Github repository with files for a specific project
- ❖ Can be specific challenges in a data set
- ❖ Changes often and quickly as a project develops
- ❖ Are often private until project is published

The screenshot shows a GitHub repository page for 'MorrellLAB / Barley_NAM_Parents'. The repository is private, has 33 commits, 1 branch, 0 releases, and 2 contributors. The latest commit was made 11 days ago. The README.md file contains instructions for processing scripts and documentation for NAM parents, mentioning sample metadata and raw sample downloading. It also lists several Excel files used for matching capture names with well numbers and indices.

Processing scripts and documentation for the 2-row and 6-row NAM parents

Aerin13 committed on GitHub Update README.md

SNP_calling Updates to PostFiltering

genotyping Add PLINK files

raw_samples Add contents of Dropbox

renaming Updated entire repository to reflect recent work

sequence_handling Add quality summary

README.md Update README.md

Barley_NAM_Parents

Processing scripts and documentation for the 2-row and 6-row NAM parents

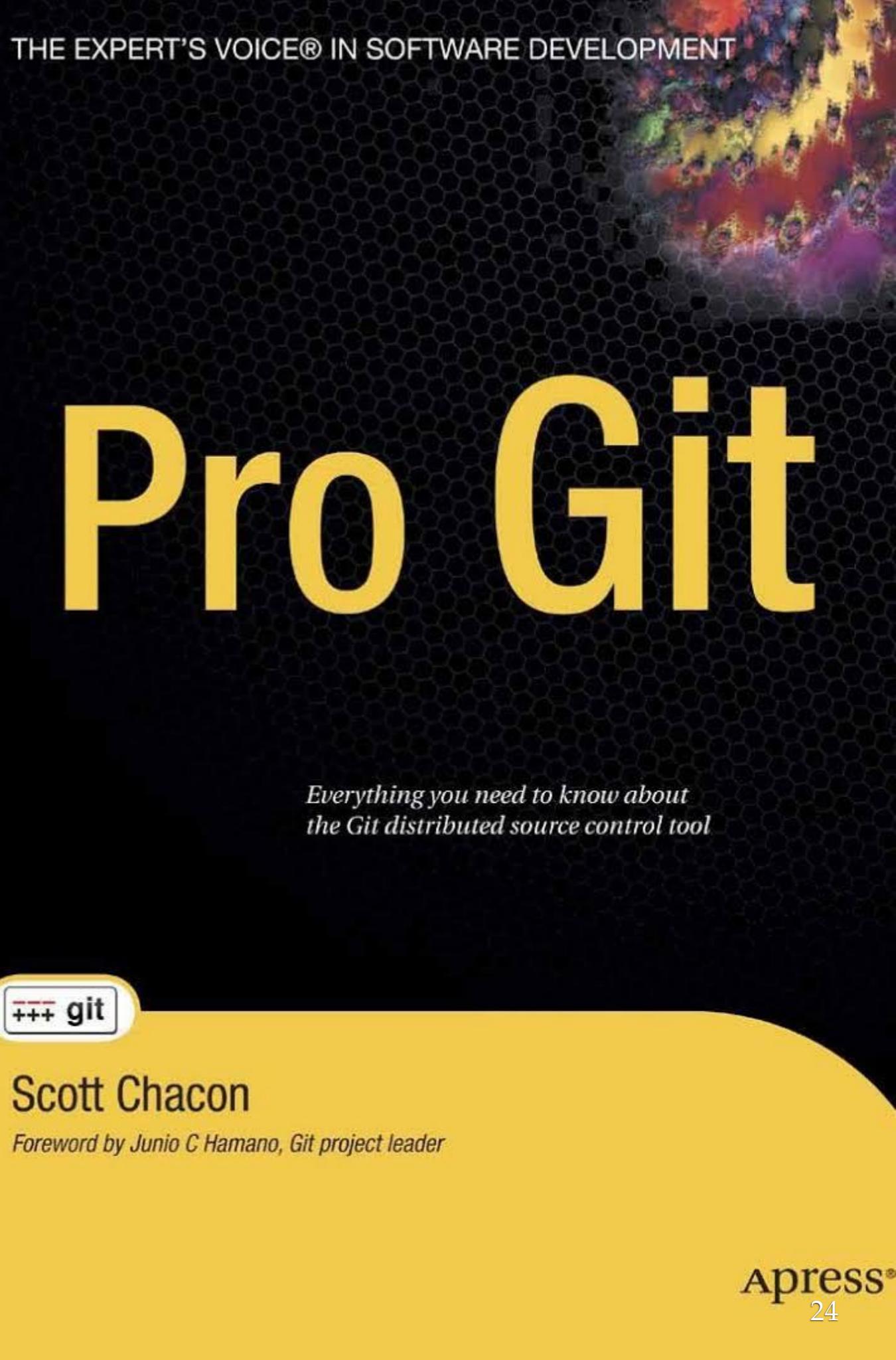
Sample Metadata and Raw Sample Downloading

Sample metadata is located in spreadsheets under [raw_samples](#).

- [Barley_Sequence_Index_info.xlsx](#) : Matches capture name (ex: BSC14) with well number and index.
- [BSC_Summary_072715_v2.xlsx](#) : Matches capture name (ex: BSC14) with well number and index.
- [Information_of_remaining_26_parents.xlsx](#) : Matches capture name with sample ID and index for 26 samples which were sequenced at a later date.

Getting started in Git/Github

- ❖ Bioinformatics Data Skills - Chapter 5
- ❖ Slides are available from Does[0]compute?
- ❖ Pro Git, a free book - 1st 3 chapters and chapter 6 are really useful



Example from my maize research

- ⌘ Began research using maize reference genome V2 , at completion of research V4 was released.
- ⌘ Original analysis of gene ontologies used V2 of genome with AgriGO. AgriGO updated to V2 in 2017.
- ⌘ Coauthors want to submit paper in 2 days, and my computer is being serviced so no access to data files and notes.
- ⌘ What would have been a better strategy?



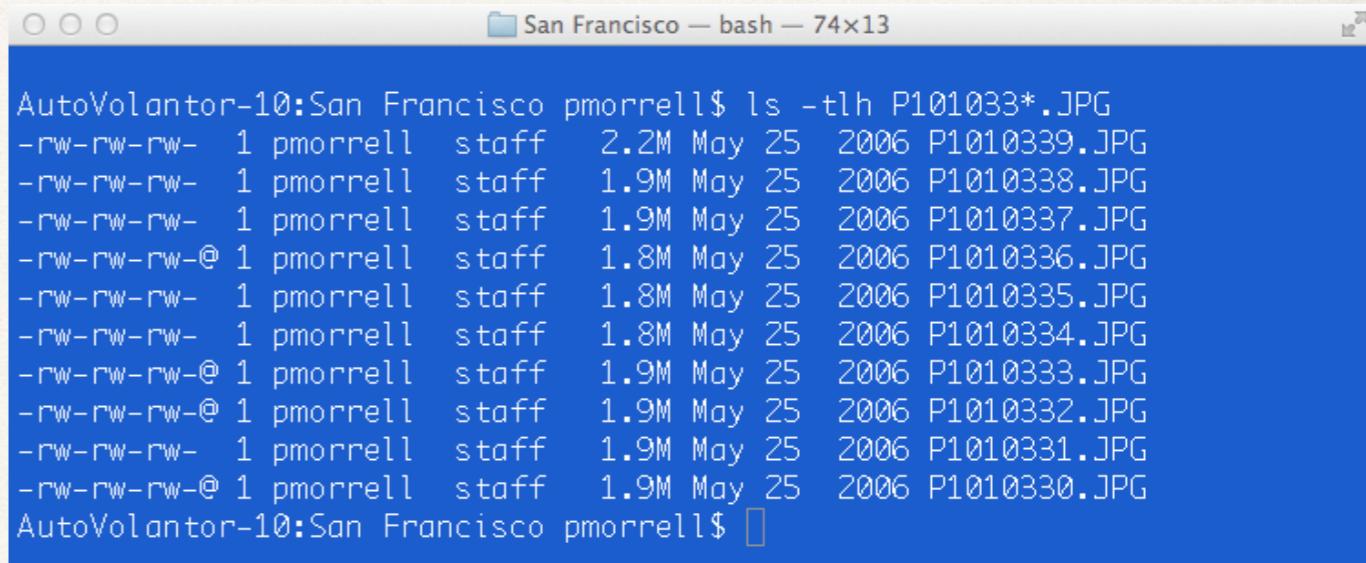
When to make the first commit?

- ❖ Create a repository as the first step to starting a project
- ❖ Create a README and jot down goals of project and expectations for analysis
- ❖ Commit code as soon as you have a few lines that are executable; even if only “Hello World!”



What is data, what is metadata?

- ❖ Actual information; e.g., the contents of a photo
- ❖ Metadata includes description of the file containing the photo



The screenshot shows a terminal window titled "San Francisco — bash — 74x13". The command "ls -tlh P101033*.JPG" is run, listing ten JPEG files from May 25, 2006. The files are sorted by size and name. The output is as follows:

```
AutoVolantor-10:San Francisco pmorrell$ ls -tlh P101033*.JPG
-rw-rw-rw- 1 pmorrell staff 2.2M May 25 2006 P1010339.JPG
-rw-rw-rw- 1 pmorrell staff 1.9M May 25 2006 P1010338.JPG
-rw-rw-rw- 1 pmorrell staff 1.9M May 25 2006 P1010337.JPG
-rw-rw-rw-@ 1 pmorrell staff 1.8M May 25 2006 P1010336.JPG
-rw-rw-rw- 1 pmorrell staff 1.8M May 25 2006 P1010335.JPG
-rw-rw-rw- 1 pmorrell staff 1.8M May 25 2006 P1010334.JPG
-rw-rw-rw-@ 1 pmorrell staff 1.9M May 25 2006 P1010333.JPG
-rw-rw-rw-@ 1 pmorrell staff 1.9M May 25 2006 P1010332.JPG
-rw-rw-rw- 1 pmorrell staff 1.9M May 25 2006 P1010331.JPG
-rw-rw-rw-@ 1 pmorrell staff 1.9M May 25 2006 P1010330.JPG
AutoVolantor-10:San Francisco pmorrell$
```

File Metadata

- ❖ All files have a physical location, size, dates
- ❖ Photos have extended attributes - EXIF data
- ❖ Geolocation - spatial metadata
- ❖ Doesn't include - accelerometer or inertial navigation info

P1010330.JPG Info

P1010330.JPG 2 MB
Modified: May 25, 2006 1:27 PM

► Spotlight Comments:

▼ General:

Kind: JPEG image
Size: 1,990,189 bytes (2 MB on disk)
Where: /Users/pmorrell/Pictures/iPhoto Library/Masters/2006/San Francisco
Created: Thursday, May 25, 2006 1:27 PM
Modified: Thursday, May 25, 2006 1:27 PM
Label:

Stationery pad
 Locked

▼ More Info:

Dimensions: 2560 x 1920
Device make: Panasonic
Device model: DMC-FZ20
Color space: RGB
Color profile: Camera RGB Profile
Focal length: 24.5
Alpha channel: No
Red eye: No
F number: 5.6
Exposure program: 2
Exposure time: 1/500

► Name & Extension:

► Open with:

▼ Preview:



► Sharing & Permissions:

```
AutoVolantor-10:San Francisco pmorrell$ mdls P1010330.JPG
kMDItemAcquisitionMake      = "Panasonic"
kMDItemAcquisitionModel    = "DMC-FZ20"
kMDItemBitsPerSample        = 32
kMDItemColorSpace           = "RGB"
kMDItemContentCreationDate  = 2006-05-25 18:27:12 +0000
kMDItemContentModificationDate = 2006-05-25 18:27:12 +0000
kMDItemContentType           = "public.jpeg"
kMDItemContentTypeTree       = (
    "public.jpeg",
    "public.image",
    "public.data",
    "public.item",
    "public.content"
)
kMDItemCreator               = "Ver1.0 "
kMDItemDisplayName           = "P1010330.JPG"
kMDItemEXIFVersion           = "2.2"
kMDItemExposureMode          = 0
kMDItemExposureProgram       = 2
kMDItemExposureTimeSeconds   = 0.002
kMDItemFlashOnOff            = 0
kMDItemFNumber                = 5.599999904632568
kMDItemFocalLength           = 24.5
kMDItemFSContentChangeDate   = 2006-05-25 18:27:12 +0000
kMDItemFSCreationDate        = 2006-05-25 18:27:12 +0000
kMDItemFSCreatorCode         = ""
kMDItemFSFinderFlags          = 0
kMDItemFSHasCustomIcon       = 0
kMDItemFSInvisible            = 0
kMDItemFSIsExtensionHidden   = 0
kMDItemFSIsStationery        = 0
kMDItemFSLabel                 = 0
kMDItemFSName                  = "P1010330.JPG"
kMDItemFSNodeCount             = 1990189
kMDItemFSOwnerGroupID         = 20
kMDItemFSOwnerUserID          = 502
kMDItemFSSize                  = 1990189
kMDItemFSTypeCode              = ""
kMDItemHasAlphaChannel        = 0
kMDItemIsApplicationManaged   = 1
kMDItemISOSpeed                = 80
kMDItemKind                    = "JPEG image"
kMDItemLogicalSize              = 1990189
kMDItemOrientation              = 0
kMDItemPhysicalSize             = 1990656
kMDItemPixelCount                = 4915200
kMDItemPixelHeight              = 1920
kMDItemPixelWidth                = 2560
kMDItemProfileName              = "Camera RGB Profile"
kMDItemRedEyeOnOff              = 0
kMDItemResolutionHeightDPI     = 72
kMDItemResolutionWidthDPI      = 72
kMDItemSupportFileType          = (
    iPhotoPreservedOriginal
```

iPhoto

1,699 of 3,912

DMC-FZ20 AWB

2560 x 1920 2.0 MB JPEG

ISO 80 | 24.5 mm | 0 EV | f/5.6 | 1/500

Baker Beach ★★★★★

May 23, 2006 4:09:00 AM

Golden Gate

Faces

Add a face...

Nova Scotia

Alaska, 6/05

Santa Monica...

Nova Scotia

San Francisco

Spring in Palm...

Wheat Diversity

Sayulita

Early Photos

Belo & Rio

Peter Morrell

Peter Morrell

Peter Morrell

Zoom

Baker Beach

Mt Tamalpais State Park Marin City

Golden Gate National Recreation Area

San Francisco

Davidson, Mount 280

POWERED BY Google Map data ©2012 Google Terms of Use

The image shows a wide-angle photograph of the Golden Gate Bridge from Baker Beach. The bridge's iconic red towers and suspension cables are visible against a clear blue sky. In the foreground, the sandy beach of Baker Beach meets the ocean, with small waves crashing onto the shore. A few people are scattered across the beach. The background features rolling hills and mountains under a bright sun. The iPhoto interface is visible around the image, including a sidebar with categories like 'LIBRARY' and 'RECENT', and a top bar with camera settings and a map overlay.

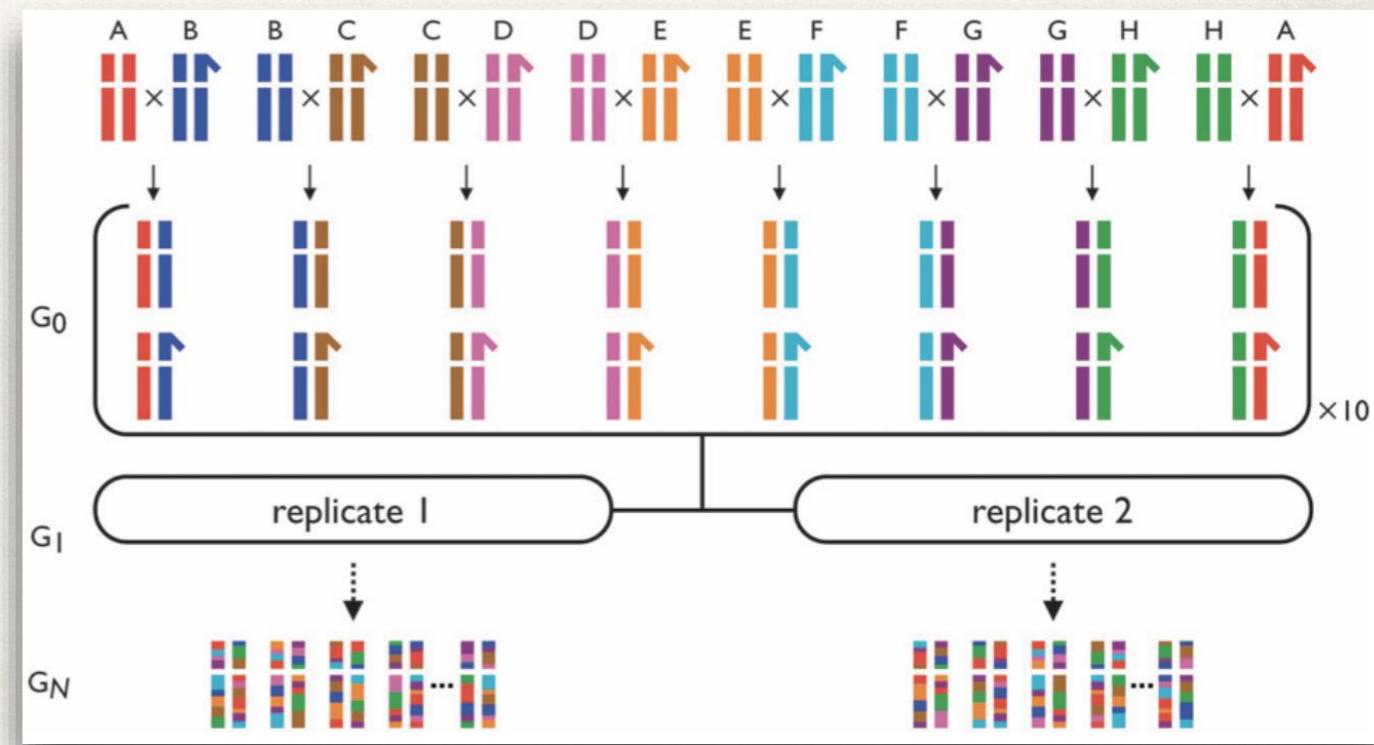
Metadata for RNASeq

- ❖ Tissue, growth stage, genotype, time of day, light intensity



Hypothetical Study

- ✿ Barley 8 parent MAGIC population
- ✿ Reporting on population creation, sequence and genetic mapping
- ✿ Recombination rate variation as a phenotype
- ✿ Genetics or PLoS Genetics manuscript



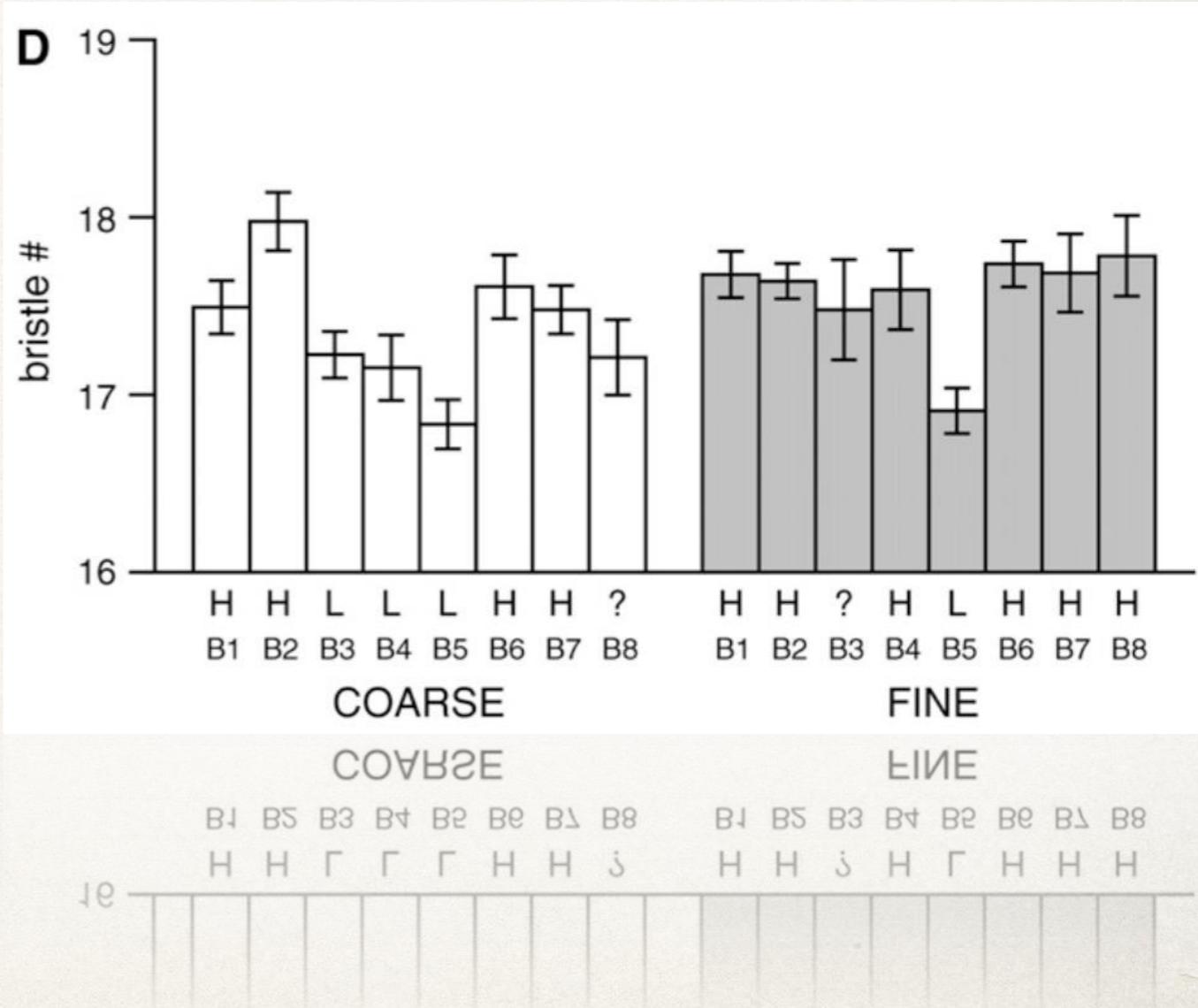
What data & metadata should we collect?

What data & metadata should we collect?

- ✿ Nanopore / Illumina sequencing of each parent
- ✿ Skim sequencing of progeny
- ✿ Recombination events per chromosomal segment for progeny

Primary manuscript

- ✿ Figures & tables
- ✿ Names of parental lines
- ✿ Sequence depth
- ✿ Crossover frequency
- ✿ Location, number, & effect of QTL discovered



Supplementary material

- ❖ Pedigree of parental lines
- ❖ Number of variants from barley Morex reference for each parent
- ❖ Proportion of variants that are coding, noncoding, deleterious, etc.

Table 1. Mean Numbers of SNPs in Various Classes.

Species	Diff. from Ref.	Noncoding	Syn.	Nonsyn.	Nonsense
Barley	162,954 (51,231.34)	115,456 (41,065.22)	15,591 (5,691.81)	12,351 (4,492.53)	77 (33.13)
Soybean	82,840 (56,780.03)	44,704 (29,477.65)	14,167 (8,161.21)	18,695 (11,289.72)	540 (345.05)

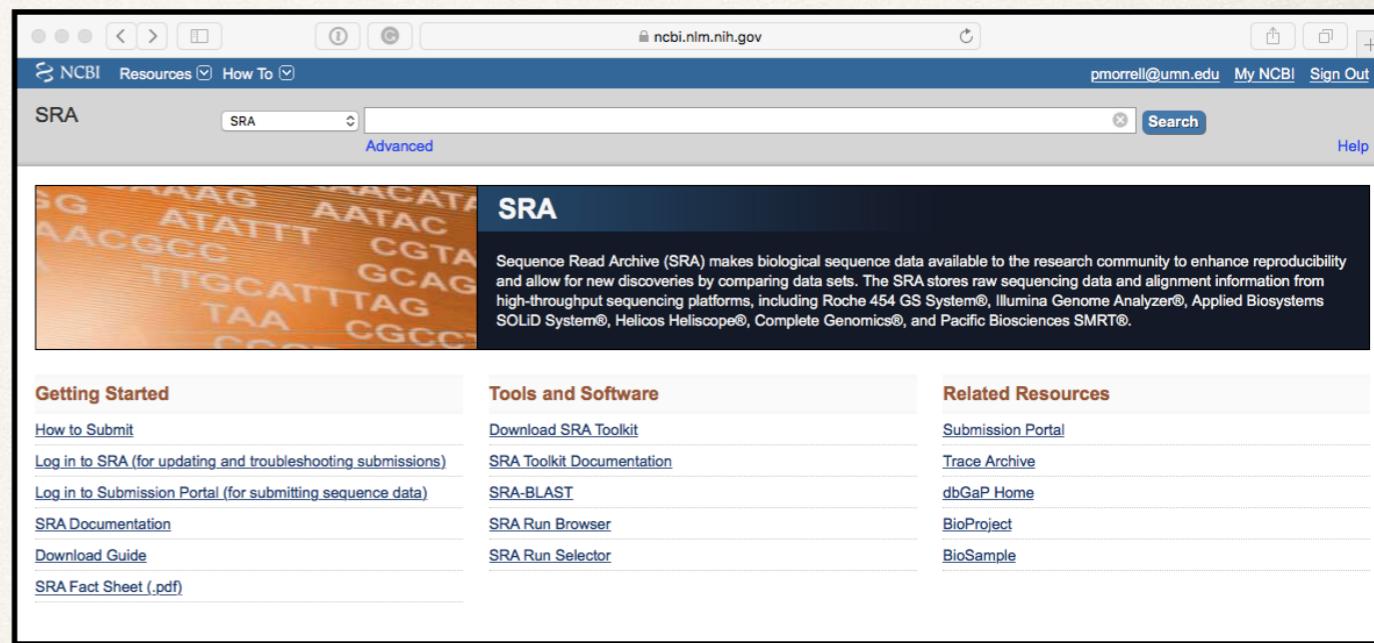
Syn., Synonymous; Nonsyn., Nonsynonymous. Numbers are mean (SD)

Data Storage

- * Git/Github not a location for data storage

Public archive

- ❖ Sequence Read Archive (SRA)
- ❖ Stores nucleotide sequence & quality values (FASTQ files)
- ❖ FASTQ for skim sequencing of progeny



DRUM

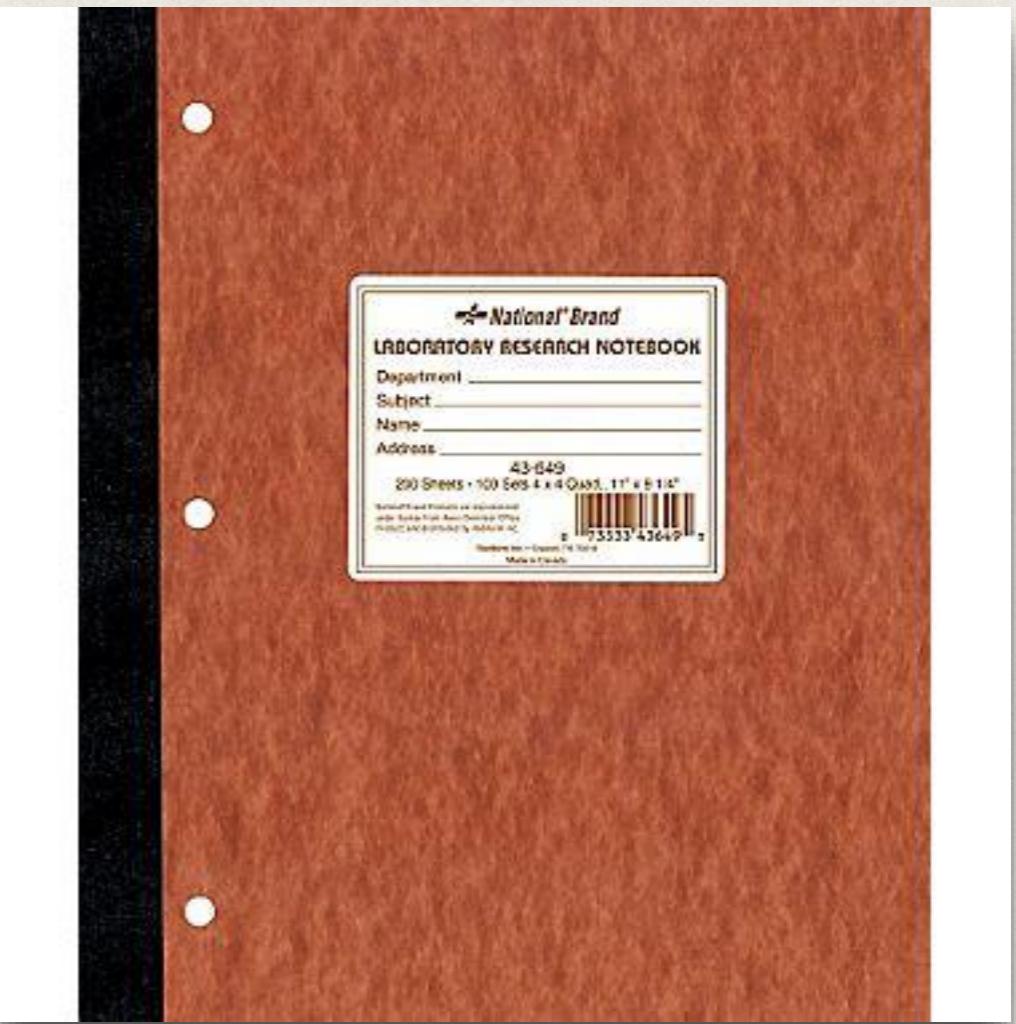
- ✿ Data needed to reproduce research
- ✿ Variant Call Format (VCF) - millions of SNPs from 8 barley parents
- ✿ Detailed variant annotation for each parent
- ✿ Get DOI with permanent URL to data

The screenshot shows the homepage of the DRUM (Data Repository for University of Minnesota) website. At the top, there's a navigation bar with the University of Minnesota logo, the text "LIBRARIES digital conservancy", and links for "Search", "Browse", "Help", and "Sign in". Below the header, the main title "Data Repository for U of M" is displayed. A search bar with the placeholder "Search the Data Repository" and a "Go" button is present. The main content area features a banner for "The Data Repository for University of Minnesota (DRUM)". It includes a brief description of DRUM as a publicly available collection of digital research data, instructions for submission, and a large yellow "Upload to the Data Repository" button. To the right, a sidebar highlights "DRUM receives Data Seal of Approval" with a red seal and a link to learn more. At the bottom, there are three columns: "How to Upload", "Features", and "Our Services".

How to Upload	Features	Our Services
1. Prepare Data Data should be free of identifying or sensitive information and include adequate documentation. Not sure? Contact us for help!	Flexible Access Options Choose to make your data immediately accessible to everyone, or moderate access to your data upon request.	Data Management Plan Assistance We offer personalized assistance for drafting

Digital notebook

- ❖ Store goals for analysis, output, code snippets
- ❖ Entries should contain as much metadata as possible

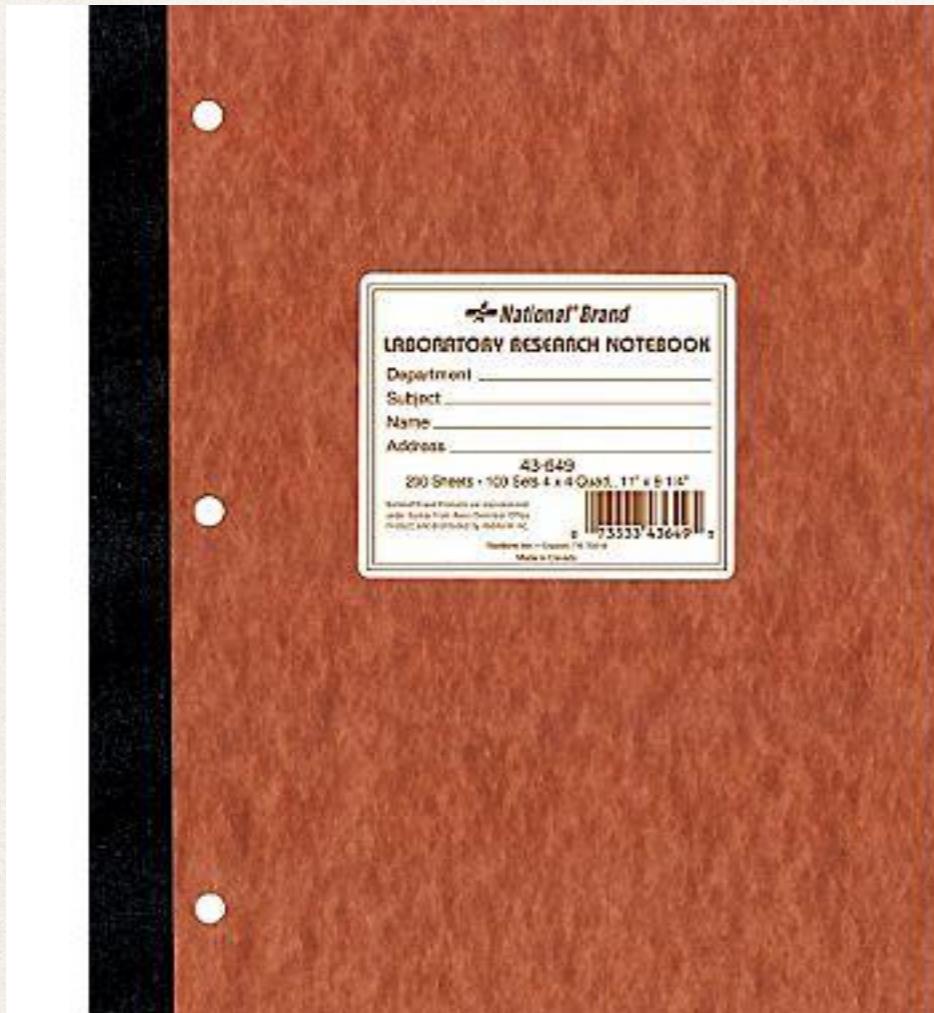


The screenshot shows a digital notebook application interface. The main area displays a list of tasks and notes, many of which are highlighted in red. One note contains a command-line script for PLINK:

```
# load the current version of plink
pmorrell@labq048:~/Workshop/PLINK_Mendel% module load plink/1.90b4.1
# To extract Mendel errors that are clearly not due to missing data
#$5 = 5 (print $0) MS1053063.mendel | grep -v '\*' | awk '{print $4}' | sort | uniq
# The following SNPs are not reverse complement errors
11_21174 C/C x C/C -> T/T
11_11040 A/A x A/A -> G/G
11_20826 T/T x T/T -> C/C
12_31509 T/T x T/T -> C/C
# All of these SNPs have issues, those not also in list have variants that differs in parents and progeny
# awk '$5 = 5 (print $0)' MS1053063.mendel | grep -v '\*' | awk '{print $4}' | sort | uniq
11_10041
11_10169
11_10288
11_10253 - het
11_10676
11_10724
11_10793
11_10926
11_10974
11_11016
11_11040
11_20092
11_20422
11_20521
11_20826
11_20944 - het
11_21038
11_21174
11_21272
11_21280
11_21521
12_30223 - het
12_30573
12_30920
12_31224
12_31262
12_31509
# Print lines that don't contain any missing data
awk '!/\*/ {print $0}' MS1152053.mendel
```

The sidebar on the right provides detailed metadata for the selected note:

- Topic: Mendel errors
- Date: 7/15/2017, 2:36 PM
- Tags: genotype
- Status: Completed
- Priority: None
- Due: 9/11/2017, 9:46 AM
- Rating: ⚡ ⚡ ⚡ ⚡ ⚡
- Created: July 15, 2017 at 2:36 PM
- Modified: Today at 9:45 AM
- Time Edited: 6 minutes 25 seconds
- Size: 2 KB
- Editable: ✓
- Flagged: □
- Icon: ⓘ
- Mood: ☺
- Word Goal: Inherit
- Label: No Label
- Background: Inherit
- Blog: Inherit
- Link:
- Location: 44.986 -93.183
- Time Zone: America/Chicago
- Related Files:



onene.com

Microsoft

Download Sign In Buy Office

Ideas take shape in OneNote

Download the app

Easily move from Evernote to OneNote and sync your notes on all devices, for free. Find out how

marinersoftware.com

Login | View Cart

MARINER SOFTWARE

Home About Blog Press Products Download eStore Support Affiliates My Account

MacJournal

Buy Now Download User Guide What's New & Upgrade

Available for Mac (Sierra Compatible)

What is MacJournal?

It's journal keeping for the 21st century. Instead of paper and pen, it is your journal on your Mac. Unlike other journal applications, MacJournal is packed with features, setting it apart. You can create as many journals as you'd like: for work, home, family, or friends. You can upload your entries to your blog. Record a message or a child's first word. It is one place where you can put everything. With MacJournal you can document any life event with all the sounds, images – even video – that comes with it.

You can't do that with a notebook.

Mendel errors — Work 2017

A

Topic: Mendel errors

Styles Spacing Lists

0 2 4 6 8 10 12 14 16 18 20 22 24 26

```

# load the current version of plink
pmorrell@labqi048 [~/Workshop/PLINK_Mendel] % module load plink/1.90b4.1

# To extract Mendel errors that are clearly not due to missing data
awk '$5 = 5 {print $0}' MS10S3063.mendel | grep -v '\*'

# The following SNPs are not reverse complement errors
11_21174    C/C x C/C -> T/T
11_11040    A/A x A/A -> G/G
11_20826    T/T x T/T -> C/C
12_31509    T/T x T/T -> C/C

# All of these SNPs have issues, those not also in list have variants that differ in parents and progeny
# awk '$5 = 5 {print $0}' MS10S3063.mendel | grep -v '\*' | awk '{print $4}' | sort | uniq
11_10041
11_10169
11_10208
11_10253 - het
11_10676
11_10724
11_10793
11_10926
11_10974
11_11016
11_11040
11_20092
11_20422
11_20521
11_20826
11_20944 - het
11_21038
11_21174
11_21272
11_21280
11_21521
12_30223 - het
12_30573
12_30920
12_31224
12_31262
12_31509

# Print lines that don't contain any missing data
awk '!/\*/ {print $0}' MS11S2053.mendel

```

Search i

Topic: Mendel errors
Date: 7/15/2017, 2:36 PM
Tags: genotype
Annotation:
Status: Completed
Priority: None
Due: 9/11/2017, 9:46 AM
Rating:
Created: July 15, 2017 at 2:36 PM
Modified: Today at 9:45 AM
Time Edited: 6 minutes 25 seconds
Size: 2 KB
 Editable Flagged
Icon:
Mood:
Word Goal: Inherit
Label: No Label
Background: Inherit
Blog: Inherit
Link:
Location: 44.986 -93.183
Time Zone: America/Chicago
Related Files:

+ 152 words 970 characters 125%

Work 2017 > Mendel errors

Why work like this?



When you can work like this!

Take home message

With disorder it is difficult to replicate your results, whereas order and documentation allows you and others to confirm your conclusions



Broader Impacts

Morrell Lab – University of Minnesota

Does[0]Compute? Lab Meetings People Publications Research Resources

Does[0]Compute?

Does[0]compute? (pronounced “Does naught compute?”), is intended to foster a discussion about best practices in computational biology. Starting **May 16th**, the discussion group meets in **213 Borlaug Hall** every other **Tuesday** from **2:00 PM - 3:00 PM**. Visit the DoesNaughtCompute Github for presentation slides, code, and sample data sets. Please bring a computer to use during the discussion. For more information please sign up for the Google group, Biocomputing Discussion Group.

Subscribe to a [calendar](#) of Does[0]compute? meetings. (Mac only)

Meeting Schedule

Morrell Lab – University of Minnesota

Morrell Lab
pmorrell@umn.edu

MorrellLab
 PeterLMorrell

Evolutionary Genetics and Plant Evolution
Last Updated: 2017-09-07

