MCB 517A: TFCB
Introduction to Tools for Computational Biology

Today’s objectives

After today’s class, you should be able to:

- Locate information relevant to the course materials and assessment
- Describe principles of reproducible computational methods
- Apply appropriate conventions for naming and organizing files and directories for computational projects

Introductions: Kate

- Research:
  - genomics, bioinformatics, evolutionary biology
  - plants, Drosophila, cancer (humans)
- Experience:
  - Primary programming tools are unix/bash and R
  - leader in non-profit teaching organization The Carpentries

Introductions

Katie Kistler (TA)

Trevor Bedford

Phil Bradley

Jesse Bloom

Erick Matsen

Gavin Ha

Arvind Rasi Subramaniam

Introductions: You!

- Name (including preferred form of address)
- Research interests

Course objectives

By the end of the course, you should be able to:

- Code in R, Python, and Unix/bash shell scripting using appropriate syntax and code convention
- Select appropriate tools to perform specific programming and data analysis tasks
- Apply good practices for computational research, including project organization and documentation
- Analyze common forms of data generated by molecular biology experiments including high throughput sequencing, flow cytometry, and 96-well plate readers.

Please complete the pre-class survey
(it should take less than 5 minutes)

Course materials (syllabus):
https://github.com/fredhutchio/tfcb_2019

Schedule

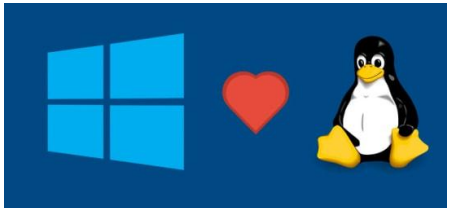
Git and GitHub

R statistical programming

Unix/bash command line

Python

Remote computing



Required software:

GitHub account (share username in survey), GitHub Desktop App (plus command line tools)

R and RStudio, plus extra packages

Windows Subsystem for Linux (not Mac)

Anaconda, plus extra libraries

Installation instructions here

Homework and grading

- Eight assignments: short answer questions and coding challenges (final assignment is capstone), each worth 10% of grade
- Participation in class sessions represents final 20% of grade
- Dates assigned and due available on GitHub syllabus
- Assignments available from, submitted through, and graded in GitHub classroom (discussed next class session)

Finding help

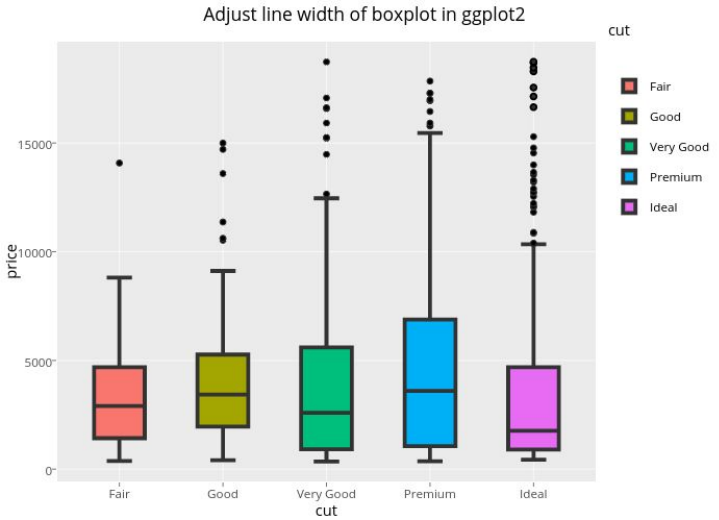
- We are working with open source tools this semester, so there are lots of resources to help you
- Homework you submit should be in your own words, with a citation (inline comment) of the online source or person that helped you

Putting this course in context

- This course is team taught AND a broad survey of tools and approaches
- Kate's job is to provide consistency and help synthesize information
- If you have questions or concerns, please talk to Kate
- Office hours (Kate and/or Katie): Tuesdays, 9am to noon in the Coop Lab (M1-B406), or by appointment (can be phone/Skype)

Reproducible Computational Biology


```
p Soccer Data 'soccer.txt" ######

afile into R ###########
able("soccer2002.txt", header=TRUE)

ze ####
occer$Freq)

an ####
leN)*(sum(soccer$Go

#### Poisson probability for X=1,
dpois(1, lambda=smean)

#### Poisson probabilities for X=
prob<-dpois(0:8, lambda=smean)
prob


#### Compute expected frequencies
efreq<-sampleN*prob
efreq
```



Questions you will be able to answer after this course:

What are the most common tools in computational biology?

How are biological data (from molecular biology research) represented?

What are appropriate methods for making computational work reproducible?

After this course, you will NOT be able to:

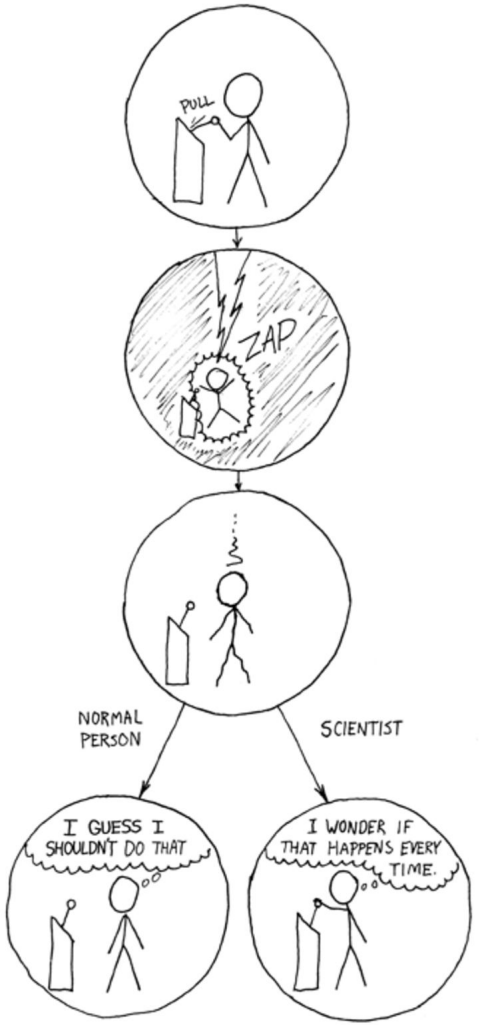
Use ALL of the tools your research will require

Know the best algorithm or analysis method for a specific research question

Code with expert-level skills (but with more work, this is possible!)

Reproducible computational methods

- *Reproducibility*: obtaining the same results multiple times
  - Confirm previously published scientific results
  - Automate large-scale data analysis projects
- *Transferability*: using methods multiple times
  - Among researchers
  - Among research questions



What does “reproducible” mean in the context of computational methods for biological research?

Good Enough Practices in Scientific Computing

1. **Data management**: saving both raw and intermediate forms, documenting all steps, creating tidy data amenable to analysis.
2. **Software**: writing, organizing, and sharing scripts and programs used in an analysis.
3. **Collaboration**: making it easy for existing and new collaborators to understand and contribute to a project.
4. **Project organization**: organizing the digital artifacts of a project to ease discovery and understanding.
5. **Tracking changes**: recording how various components of your project change over time.
6. **Manuscripts**: writing manuscripts in a way that leaves an audit trail and minimizes manual merging of conflicts.

Items in blue will be a focus of this class

Good Enough Practices: Project Organization

Put each project in its own directory, which is named after the project.

Divide work into projects based on overlap of data and code

```
|-- CITATION.txt  (how to reference project)
|-- README.md     (overview of project)
|-- LICENSE.txt   (how project/code can be used)
|-- requirements.txt  (dependencies for code to run)
|-- bin/
|-- data/
|-- doc/
|-- results/
|-- src/
```

Good Enough Practices: Project Organization

Put text documents associated with the project in the `doc` directory.

```
|-- doc/
|   |-- notebook.md  (electronic lab notebook)
|   |-- manuscript.md  (manuscript draft)
|   |-- changelog.txt  (record of how project
                        changed over time)
```

Good Enough Practices: Project Organization

Put raw data and metadata in a `data` directory

```
|-- data/
|   |-- data.csv  (raw data)
|   |-- README  (metadata, or information about data)
```

Good Enough Practices: Project Organization

Put files generated during cleanup and analysis in a `results` directory.

A separate `figures` directory is also useful.

```
|-- results/
|   |-- filtered_data.csv
|   |-- summarized_results.csv
|-- figures/
|   |-- scatterplot.csv
```

Good Enough Practices: Project Organization

Put project source code in the `src` directory.

Put external scripts or compiled programs in the `bin` directory.

```
|-- bin/
|   |-- labmates_script.py
|-- src/
|   |-- data_analysis.py
```

Good Enough Practices: Project Organization

Name all files to reflect their content or function.

- File suffixes matter!
- Avoid spaces: use underscores (or dashes, or capitalization)
- Avoid sequential numbers
- Avoid relative position in manuscript

```
|-- bin/
|   |-- labmates_script.py
|-- src/
|   |-- data_analysis.py
```

Good Enough Practices: Project Organization Summary

1. Put each project in its own directory, which is named after the project.
2. Put text documents associated with the project in the `doc` directory.
3. Put raw data and metadata in a `data` directory and files generated during cleanup and analysis in a `results` directory.
4. Put project source code in the `src` directory.
5. Put external scripts or compiled programs in the `bin` directory.
6. Name all files to reflect their content or function.

Different labs, companies, and open source projects likely have different guidelines!

Other resources for project organization

This class is a broad overview of skills and not project focused, so we will not necessarily apply best practices in project organization.

- A Quick Guide to Organizing Computational Biology Projects
- Fred Hutch project templates (currently for Python, others in development)
- Templates from other places:
  - Cookiecutter (many project types)
  - Shablona (Python, from UW eScience)
  - ProjectTemplate (R)
- Your advisor or employer

Summary

- This course focuses on data-driven computation using open-source scientific tools
- Course materials will reflect use of these tools
- Regardless of the extent to which you use computation in your research, learning the methods will make you a better scientist

Next time: Version control with Git!

For next time:

- Compete the pre-class survey
- Install all required software (and get a GitHub account!)
- Orient yourself to recommended reading materials