Informatics and Statistics for Molecular Biologists (MOLB 7900)

Jay Hesselberth

2019-04-22

Contents

Course Overview

Informatics and Statistics for Molecular Biologists (MOLB 7900) is offered at the University of Colorado School of Medicine and teaches students how to design and analyze common molecular biology experiments.

Syllabus

0.1 Course Overview

Informatics and Statistics for Molecular Biologists (MOLB 7900) teaches students to design and analyze experiments commonly used in molecular biology. The course is organized around the Central Dogma (DNA > RNA > Protein) wherein each block presents 2-3 experimental approaches. Each week, a new experiment is introduced with a discussion of appropriate design and statistical considerations. The remaining weeks' classes are devoted to digging into the analysis of a sample data set.

The course begins with a 4 week "boot camp" designed to get students familiar with and bring them up to speed on using software for shell programming and data analysis with R and Python. We also establish accounts on Github for problem set submission.

0.1.1 Block 0: Bootcamp

```
A: Shell programming
B: R Studio
C: Integration with Git
D: Python
```

0.1.2 Block 1: DNA

```
A: Chromatin accessibility
B: Motif finding
C: Variant Calling
```

0.1.3 Block 2: RNA

```
A: Quantitative PCR
B: Bulk mRNA-seq
C: RNA-protein interaction (CLIP)
```

0.1.4 Block 3: PROTEIN

```
A: Mass spectrometry (counts)
B: Densitometry of gel images
C: Fluorescent protein localization (Chad)
```

0.2 Course Objectives

- Cater to students of many backgrounds (more computational or more biological)
- Be able to formulate questions that are testable with computational techniques
- Understand the limitations of sequencing-based techniques
- Be fluent in statistical considerations for different approaches and design well-controlled experiments that can be analyzed using statistical tests
- Be fluent in command-line programming, scripting (Python) and data analysis / viz (R / R Studio)
- Understand the value of internet-based analysis tools (NCBI BLAST etc)
- Use reproducible software development approaches (Github) and dynamic documents (Rmarkdown)
- Independently conceive of and implement a soup to nuts reanalysis of an existing data set, which are presented to the class.

0.3 Class Schedule

Monday, Wednesday, and Friday, 1-2:30pm from Sept 10 to Dec 10.

Chapter 1

Bootcamp

This 4 week "boot camp" is designed to get students familiar with and bring them up to speed on using software for shell programming and data analysis with R and Python. We also establish accounts on Github for problem set submission.

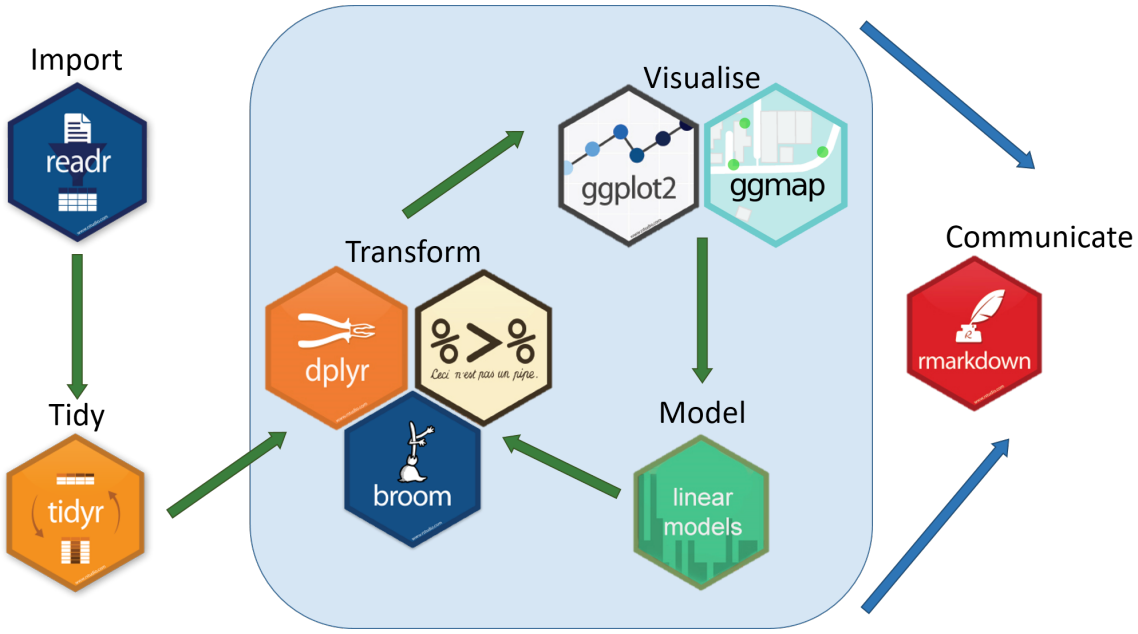
1.1 Shell programming

Students will become familiar with the operating in the Shell, which is closely related to command-line/terminal. The Shell is a program which runs other programs rather than doing calculations itself. Bash is the default shell on most modern implementations of Unix, Mac, and in most packages that provide Unix-like tools for Windows.

After this section, students will be able to navigate directories, create an organized directory structure for a project, and install and run software.

1.2 Rstudio

Students will learn to use Rstudio, which is an integrated development environment that makes it way easier to code in R. Within Rstudio, students will use Rmarkdown to produce high-quality reports, presentations, and documents in a highly reproducible manner. Students will become fluent in importing, processing, and transforming data using a collection of R packages designed for data science tidyverse.



After achieving exellence in basic data handling, students will be introduced to a broad range of commonly used statistical tests and an underlying conceptual framework for deciding the appropriate statistical test. We will emphasize concepts unique to genomic/big data. These statistical concepts and tests will be revisited during applied sections of the course in which specific technology and data are introduced and re-analyzed.

After this section, students will be able to import, process, transform, visualize data; use statistical tests to analyze basic tabular data; genarate html/pdf reports of their work.

1.3 Git

Modern scientists need to write code as part of their research. This code needs to be documented just as bench experiments need to be logged in a lab notebook. Version control systems like Git, and online hosting site, GitHub, are critical tools to address these important issues. These tools allow students to track iterative changes made to their code, revert to a specific previous version, and share their code with the broader scientific community. Altogether these tools are a cornerstone of reproducible research practices that is conveniently integrated within Rstudio and Rmarkdown.

After this section, students will be able to use git within Rstudio to easily version and share their code. They will be expected to turn in assignments using git in Rstudio.

1.4 Python

Python is an easy to learn language that combines the flexibility of bash along with the conveniences of higher level languages like R. This useful for solving problems for which software does not exist, which is important for students to learn in order to anticipate unmet needs and data from new technologies/assays. Many tools for the analysis of modern datasets are available as python packages, allowing the incorporation of pre-built analysis tools within custom, made-from-scratch, code. Furthermore, python's suitability as

a scripting language makes it a natural fit for APIs of other common analysis tools. For example, image analysis pipelines can be constructed in Fiji, allowing the batch processing of many files at once.

After this section, students will be able to write basic software and scripts that will enable them to derive meaning from the large datasets typical of modern biology.

Chapter 2

Block 1: DNA

Chapter 3

Block 2: RNA

Chapter 4

Block 3: Protein

Chapter 5

Exercises