

Computational Notebooks

Amelia Harrison

With special thanks to Ryan Moore

April 19, 2022

Overview

- What is a computational notebook?
- What goes into a computational notebook?
- Notebook organization
 - Organization strategies
 - Tools to assist organization
- Notebook software
- Tips and recommendations

What is a computational notebook?

- Living document(s) that contains a record of all work
 - Experiments and their parameters
 - Explanations of experiments and decisions
 - Troubleshooting
 - Data analyses
 - Discussions of results

What do I put in my computational notebook?

Wet lab notebook

- Procedure
 - PCR, Western blot, cloning
- Equipment
 - Thermocycler, gel box
- Reagents/concentrations
 - SYBR Gold x1000, 50 μ L
- Errors and troubleshooting
 - Finding the reagent that contaminated your PCR

Dry lab notebook

- Procedure
 - Protein structure prediction
- Software (with version no.)
 - AlphaFold
- Commands and parameters
 - `--model_preset=monomer`
- Errors and troubleshooting
 - Finding the mistake in your code or command

What is **not** a computational notebook?

- Code or scripts
- Reports
- Documentation or tutorials
 - These are like protocols or SOPs

Organizing a computational notebook

- Two big questions:
 - How do I “order” a computational notebook?
 - Where and how do I store my notebook?
- Things to consider:
 - Will I be able to find this information in a week? A month? A year?
 - Will this flow of information always make sense?

Organizing a computational notebook

- Common methods of organization
 - Chronological (time-ordered)
 - Hierarchical or other project-based
 - Task-based

Chronologies by themselves are bad

- Humans do not always think linearly
- Computational work does not usually proceed cleanly enough
- Examples:
 - You have an job running that will not be finished for a week
 - A lab mate asks for help with a time-sensitive project
 - You have a poster session coming up

Hierarchies by themselves are bad

- Let's say you have 3 projects:
 - Cattle microbiome project
 - Oyster microbiome project
 - Ocean virus metagenome project
- You have decided to use a hierarchical organization system for your notes
- This is what you come up with...

Hierarchy example

- Cattle microbiome

- Assemblies
 - Read QC
 - Filter contaminants
 - Run assembler
 - Get assembly statistics
- Abundance
 - Predict ORFS
 - Map reads to ORFS
 - Map reads to contigs
 - Do magic calculations
- Make count table
- Ecological analysis
 - Alpha/beta diversity

- Oyster microbiome

- Assemblies
 - Read QC
 - Filter contaminants
 - Run assembler
 - Get assembly statistics
- Abundance
 - Predict ORFS
 - Map reads to ORFS
 - Map reads to contigs
 - Do magic calculations
- Make count table
- Ecological analysis
 - Alpha/beta diversity

- Ocean virus metagenome

- Assemblies
 - Read QC
 - Filter contaminants
 - Run assembler
 - Get assembly statistics
- Abundance
 - Predict ORFS
 - Map reads to ORFS
 - Map reads to contigs
 - Do magic calculations
- Make count table
- Ecological analysis
 - Alpha/beta diversity

Another hierarchy example

- Assemblies
 - Cattle assembly
 - Oyster assembly
 - Ocean assembly
 - Abundance
 - Cattle abundance
 - Oyster abundance
 - Ocean abundance
 - Make count table
 - Cattle table
 - Oyster table
 - Ocean table
 - Ecological stats
 - Cattle stats
 - Oyster stats
 - Ocean stats
- This hierarchy is task-based instead of project-based
 - Which structure makes more sense?

What is the alternative?

- Combine the different methods by connecting notes
 - Results in notes that are organized closer to the way a human thinks
- Two main ways to connect notes:
 - Linking
 - Tagging

Linking

- Method of directly linking notes to other resources
- From a note, you can link to:
 - Other notes
 - Files
 - Webpages
- Downside - links to local files are hard to manage by yourself

Tagging

- Method of categorizing notes
- Allows for a more “flattened” organization
- Often done with #
 - #virus_metagenome
 - #oyster_microbiome
 - #read_qc
- Downside - need software to manage and search tags or need to be good at searching with the command line

Example

- Cattle_microbiome_assembly.note
 - #cattle_microbiome #assembly
#read_qc #assembly_stats
 - [Next step: Abundance](#)
- Cattle_microbiome_abundance.note
 - #cattle_microbiome #orf_prediction
#read_mapping #magic_calculations
 - [Next step: Count table](#)
- Cattle microbiome
 - Assemblies
 - Read QC
 - Filter contaminants
 - Run assembler
 - Get assembly statistics
 - Abundance
 - Predict ORFS
 - Map reads to ORFS
 - Map reads to contigs
 - Do magic calculations
 - Make count table
 - Ecological analysis
 - Alpha/beta diversity

Software and formats for the best notes

- Pure Markdown (barebones):
 - Macdown (MacOS)
 - Typora (Linux, Windows)
 - StackEdit (Online/In-browser)
- Markdown based:
 - Obsidian (used in this workshop)
 - Zettlr
- Wiki based:
 - TiddlyWiki
- In-line analysis based (also use markdown):
 - R Notebooks/R Markdown
 - Jupyter Notebook
- Version control/git (advanced)

Tips

1. Do not put code in a word processor (e.g., Microsoft Word, Google Docs)
 - Word processors change whitespace characters
2. Use a notes manager or get good with the command line
 - (Most) humans are not designed for the level of organization you need
3. Don't worry about file names
 - There are only so many combinations of `project_name_step_name.note`

Recommendations

- Keep all of your notes in one folder
 - Looking for things becomes much harder when notes are scattered
 - If you feel the need, you can keep a readme with each project
- Keep a daily log
 - Use this as a place for narratives and to-do lists

Questions?