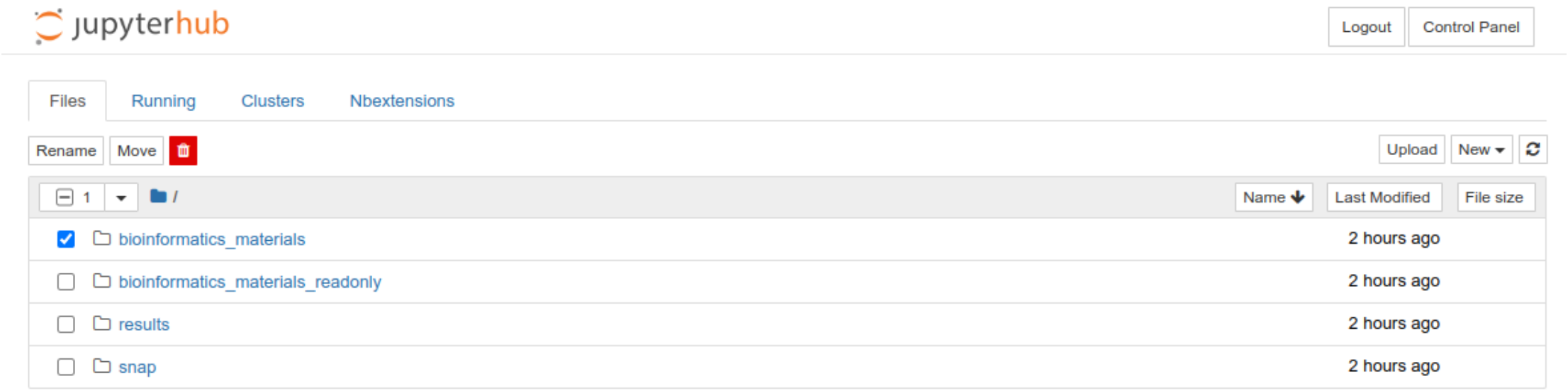
Jupyter 101

Basic usage of the jupyterhub and FAQ

Main menu

1: your files and directories. Click to open directories, notebooks and other files.
2: Here you see your current path. You can also navigate back to your home folder by clicking on the folder icon.
3: Here you can see your running terminals and notebooks, and resume your work if you closed a browser tab.

Main menu



We trust you to NOT rename and delete provided files or folders on your own!

(Unless we ask you to do it in a tutorial.)

Notebook view

Code blocks and markdown.

In some tutorials, Code is to be executed in the terminal, sometimes in the cell itself (Ctrl-Enter).

Answer fields

You can fill in your answer here.

Solution buttons

If you are stuck with a question, you can reveal the solution to continue working. Give yourself a chance to solve the questions by yourself, to optimize your learning outcome.

Save the notebook (and your progress) from time to time

jupyterhub improvedTutorial1 (read only)

Logout Control Panel

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

Run Markdown

Memory: 159.1 MB

Tutorial 1: Linux and command line part 1, O-notation

Introduction

This tutorial gives you a brief overview of working on the Linux command line, an essential tool for any computational biologist. Learning the command line is sometimes challenging and takes some effort. It's not that it's so hard, but rather it's so vast. However, unlike many other computer skills, knowledge of the command line is long lasting since it has proved successful over the years. In order to solve the following tasks you may need some of the most common shell commands listed on the last page of this tutorial (see Table 1). Use the Linux manual pages, referred to as man pages, to learn about what commands do and how parameters alter their behaviour. Let's start ...

1

For this Tutorial, we ask you to use the terminal. Open the terminal via the dashboard (return to it by right-clicking "jupyterhub" in the top left corner, then select "open link in new tab") -> new -> Other: Terminal

Note: The code cells in this Jupyter notebook will not work, so please don't run them.

Files and directories

After opening a terminal, let's move into the directory to work in:

In [ ]:
```
cd ~/bioinformatics_materials/Tutorial1and2
```

2

Decompress the file exercise_1_2.zip in your current directory.

In [ ]:
```
unzip ~/bioinformatics_materials_readonly/Tutorial1and2/exercise_1_2.zip
```

Then change directory from your current directory to the directory exercise_1_2 by typing

In [ ]:
```
cd exercise_1_2
```

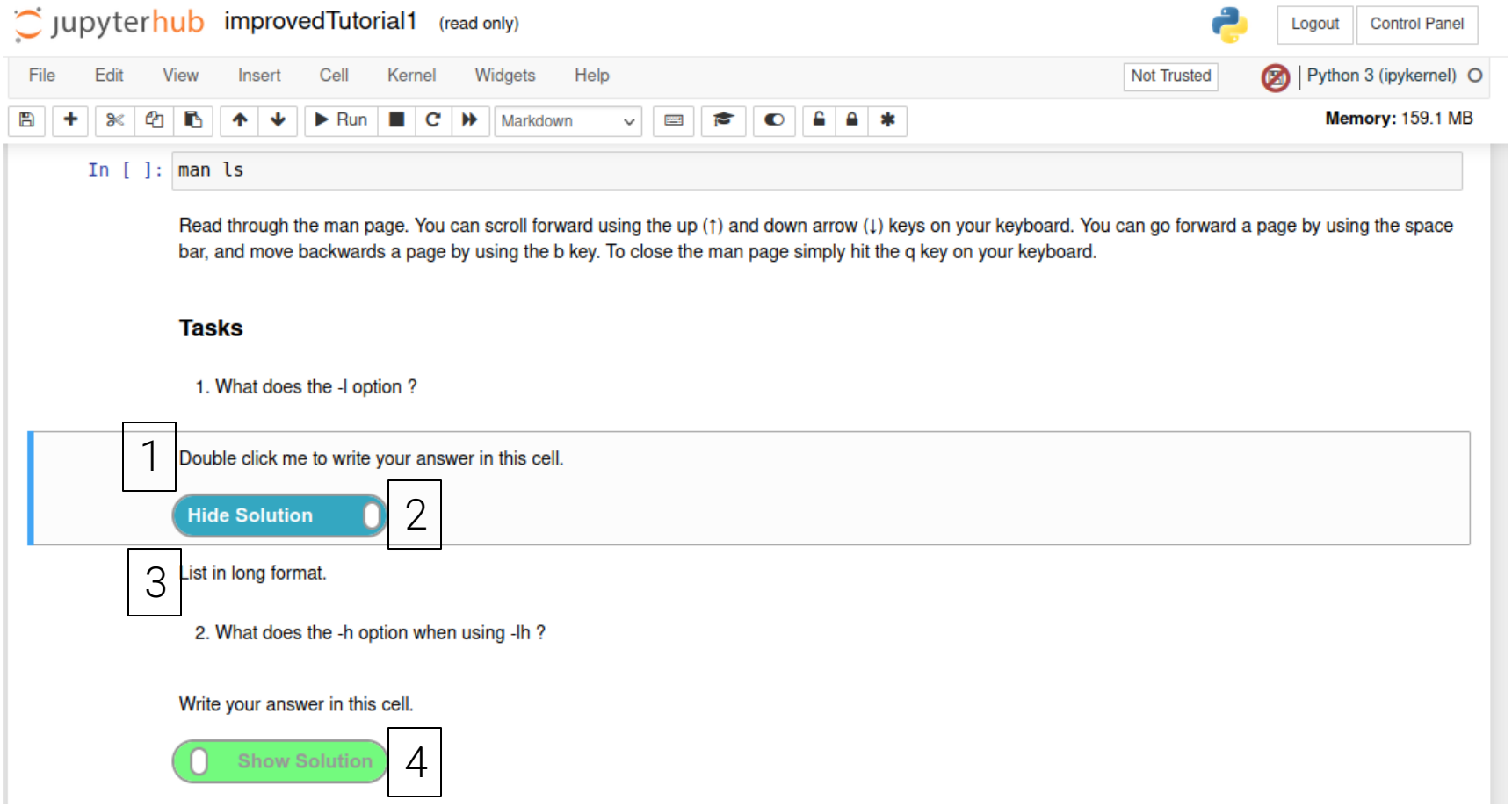
Find the full path to where you are by typing

In [ ]:
```
pwd
```

In [ ]:
```
ls
```

1: In some tutorials, we ask you to use the terminal for executing the code, in some tutorials the code is directly executed in the notebook.
2: This is a code cell, to be executed in the terminal.

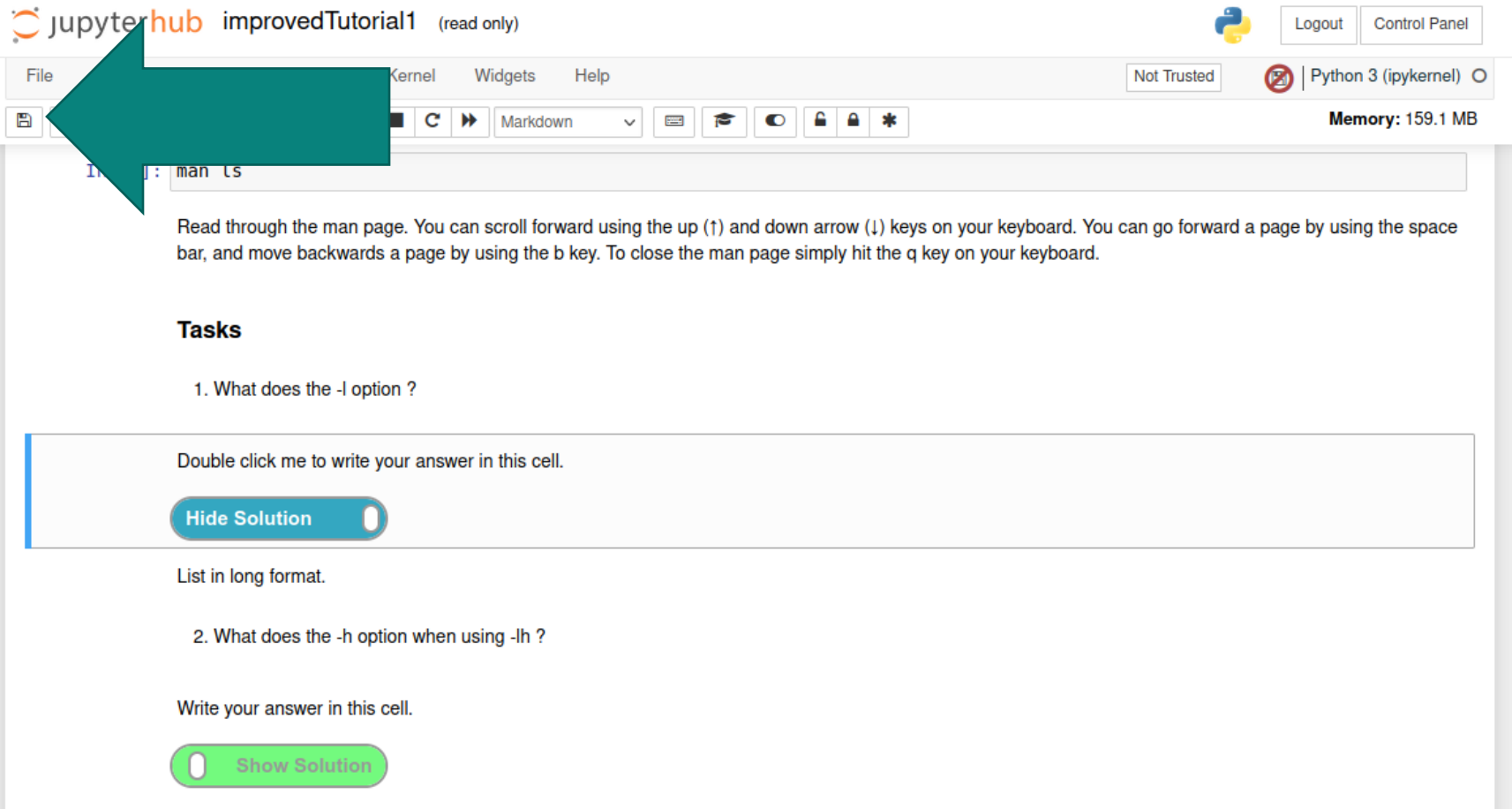
Notebook view - Tasks



There will be tasks in the tutorials for you to work on during the hands-on tutorial.
1: You can double click answer fields to enter your answers. The solutions will be discussed with your tutor at the end of each tutorial.
2-4: You can hide (2) and show (4) the solutions (3) to the tasks, so you can always continue if you don't know the answer or to check the solution.

Note: At the end of each tutorial, you will have opportunity to ask for explanations of the tasks. Also, do not look at the solutions immediately, because you will learn the most from solving problems yourself.

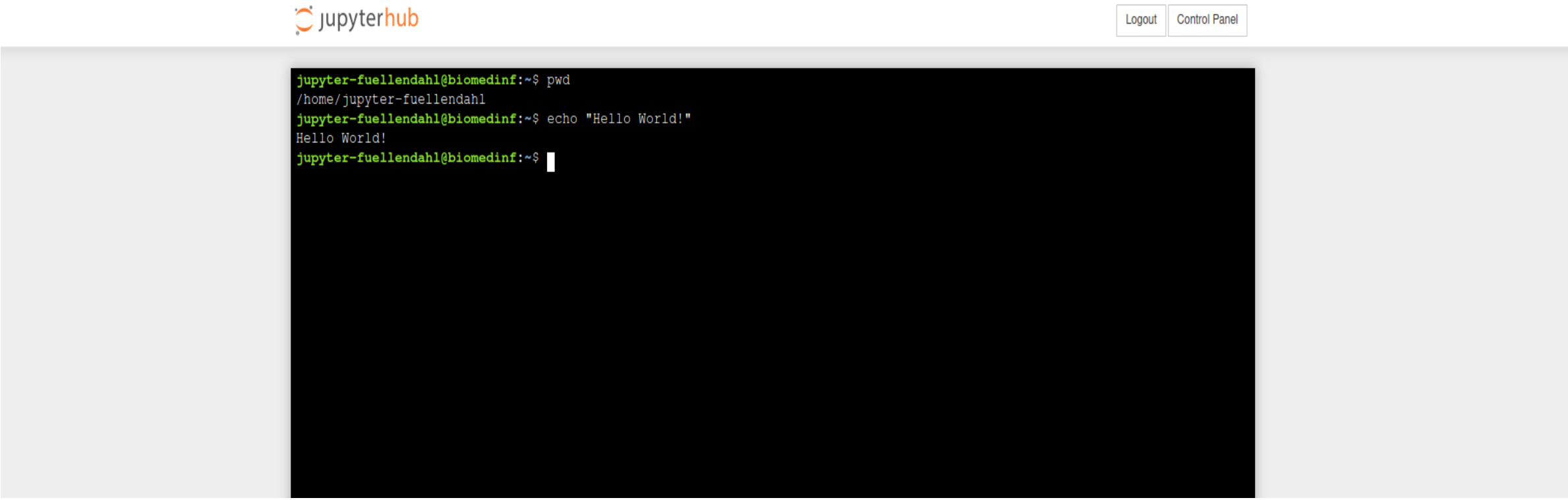
Notebook view - Tasks



Save your progress regularly and when you finish your work by pressing the Floppy disk icon!

Terminal view

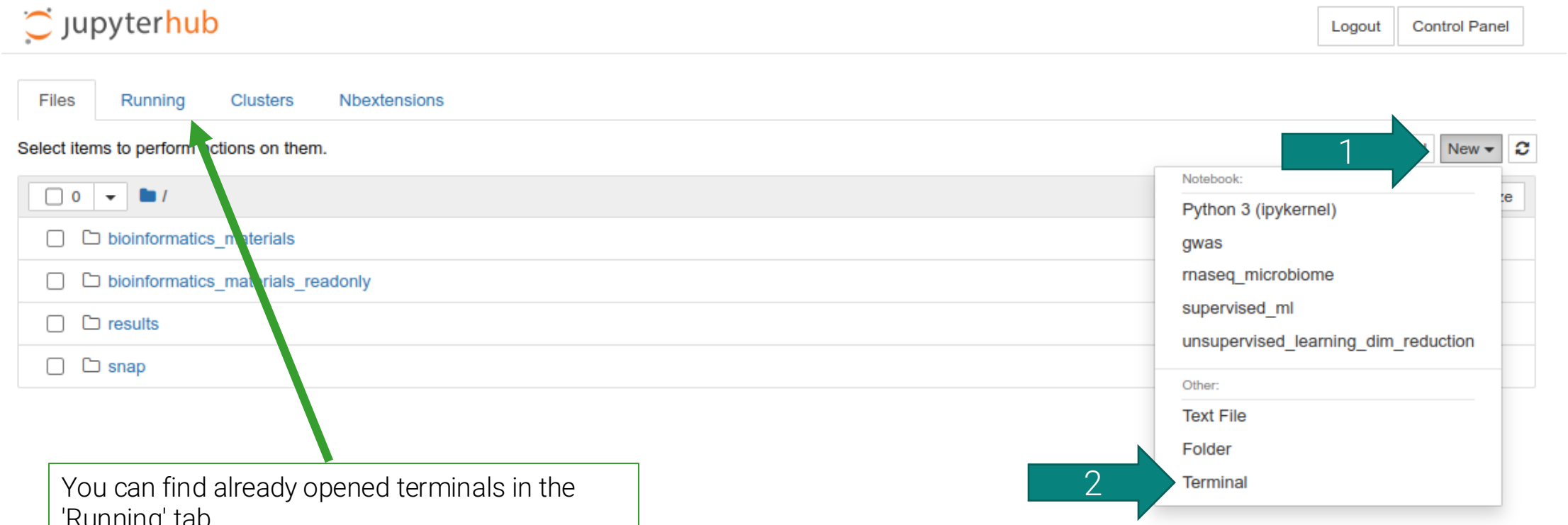
Here you can execute commands like you would on any other linux machine, this is just a web interface to a real terminal.
If you delete a file (rm filename.txt) or overwrite it, it's gone. There is no trash bin and no way to restore the deleted file!



FAQ

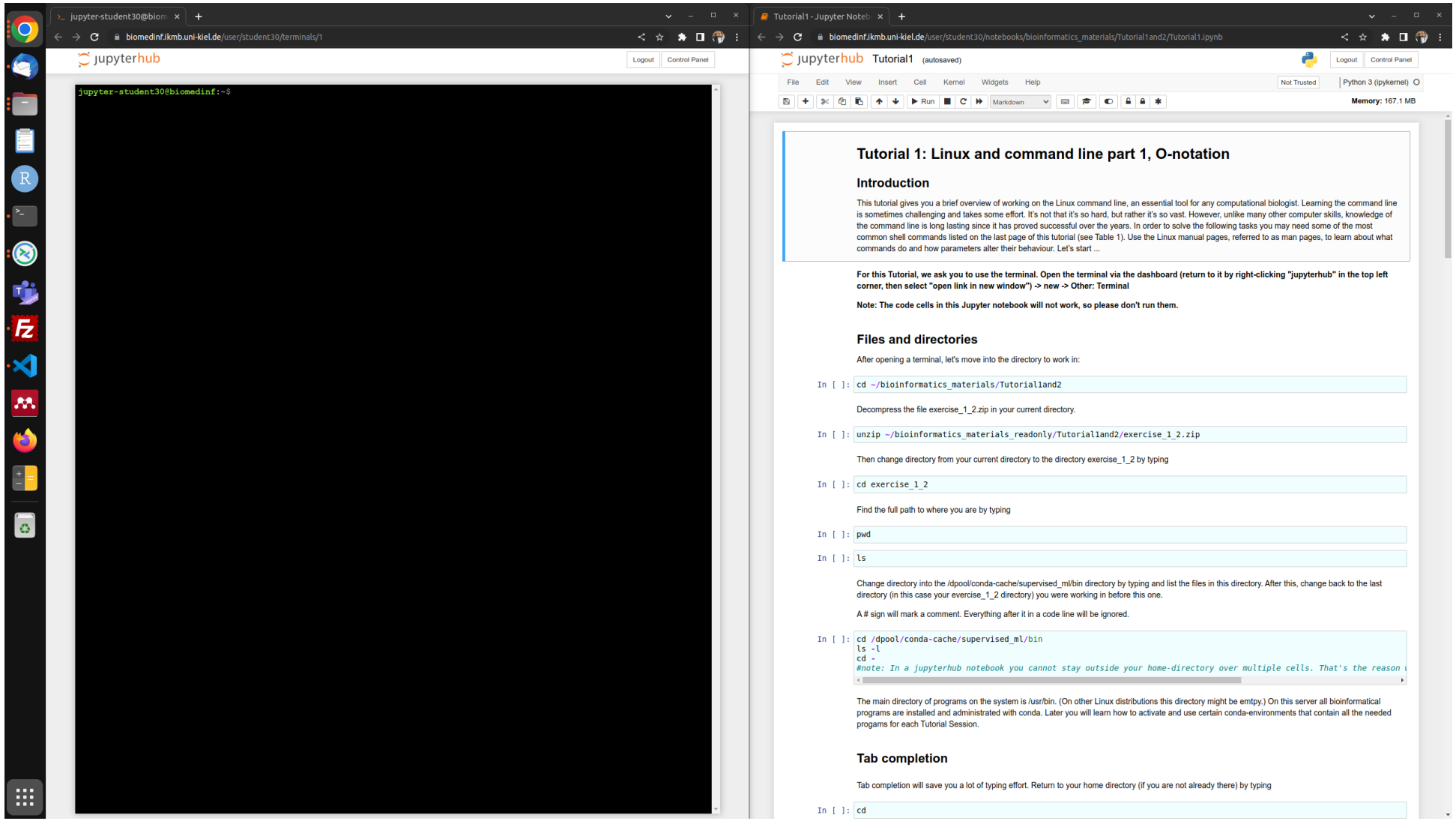
How do I open the terminal in jupyter?
What do I do if I accidentally deleted a file in my home directory?
I forgot my password.
How can I change my password?
A Notebook does not execute commands; I get only error messages.

How do I open the terminal in jupyter?



You can find already opened terminals in the 'Running' tab

Pro tip: Work with tabs!



Obtained results are precious!

Be careful not to delete your own results files in the Results folder by accident. We cannot restore them! (No "trash bin" available).

I forgot my password.

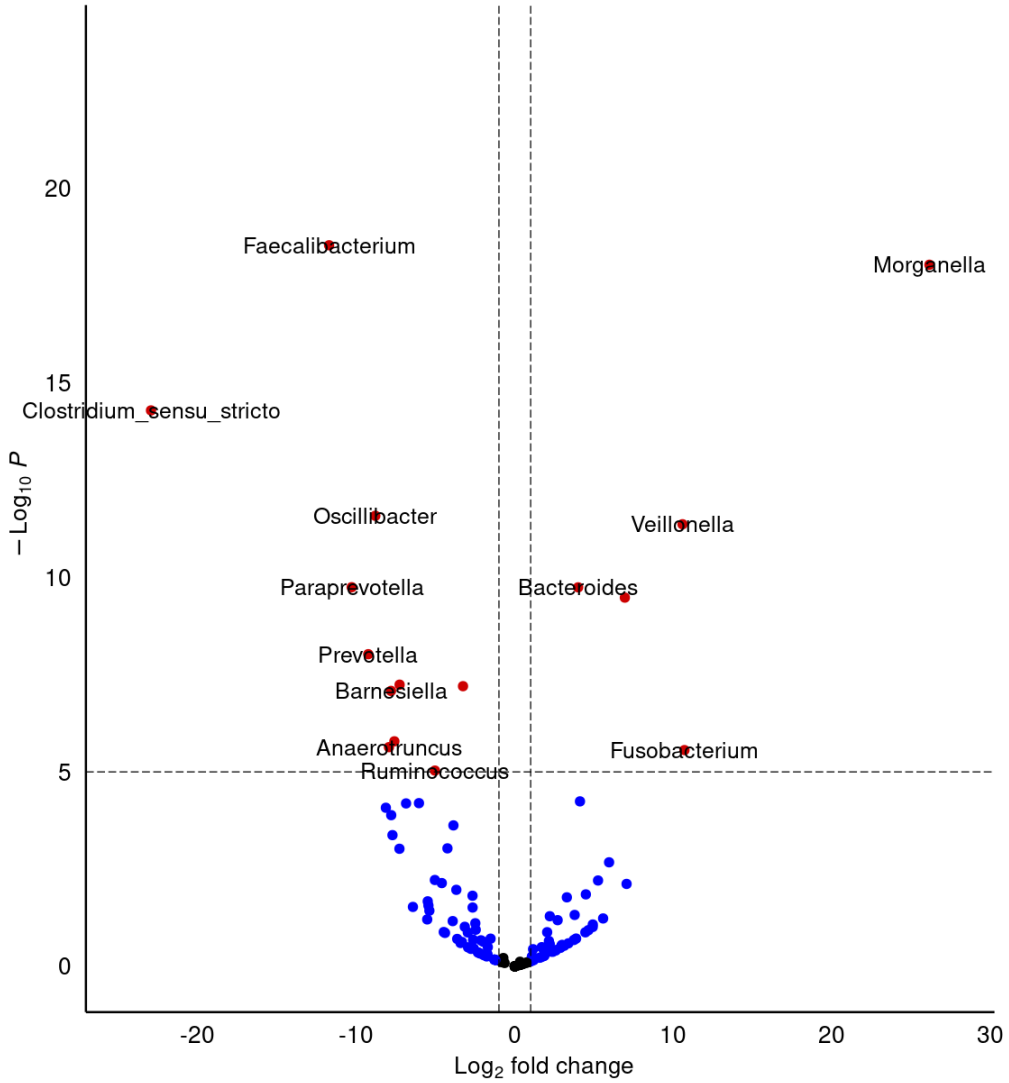
If you still have your student ID, we can reset your password. During the course, you can ask a tutor. However, this takes time (from your hands-on time) and effort. Please make sure to remember your password from now on.

A Notebook does not execute commands; I get only error messages.

1. In some tutorials (e.g. the last one) some code cells are meant to be executed in the terminal. Copy-Paste **(or better type it yourself)** them to a terminal and execute them in order.
2. Please note: The code cells may fail or give wrong results if they are not executed in the given order.
3. Make sure that the correct kernel 'kmc_workshop' is loaded.
4. Check that you did not accidentally modify code blocks.
5. If you checked all this, close and halt the notebook, then rerun it from the beginning to the part where the problem occurred.
6. If that does not help, ask a tutor during the workshop

Differential abundance analysis of 16S data

Eike Matthias Wacker



IKMB
Institute of Clinical Molecular Biology

Overview

- Quality checks
- Visualisation of abundance table
- Data normalization
- Data transformation
- Differential abundance analysis:
  - Non-parametric statistical test
  - DESeq2
  - MaAsLin2
- Comparison of results

Quality control

Phyloseq object QC:

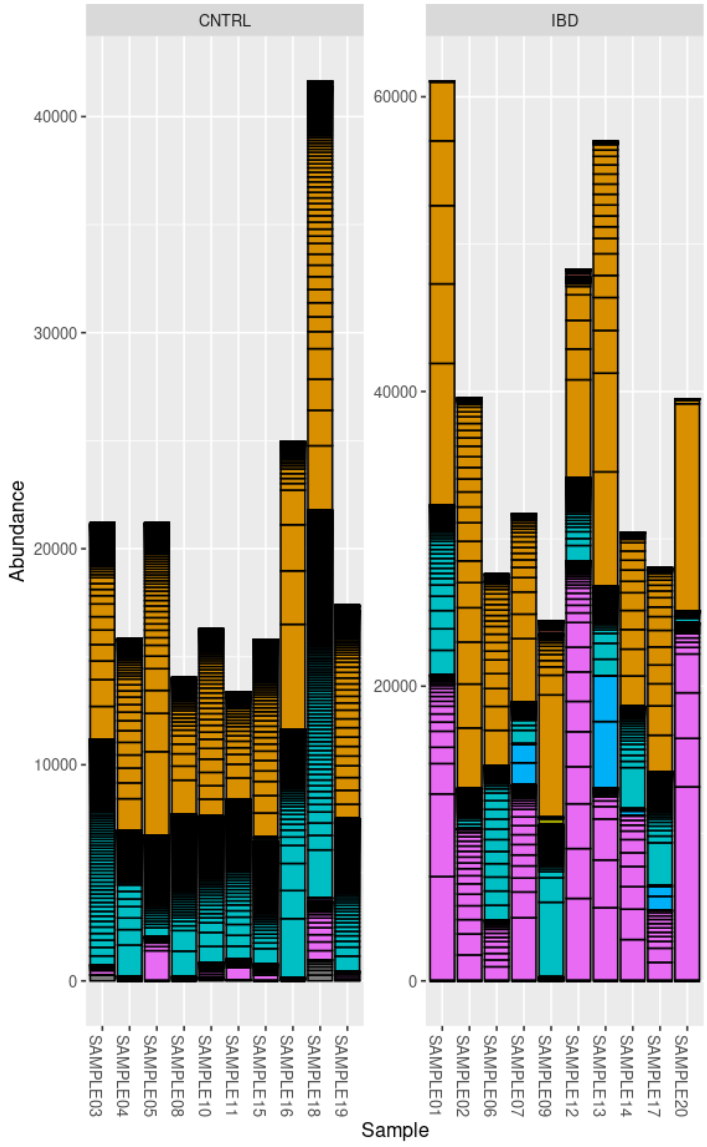
- Removal of samples with low read counts
- Removal of samples with only one taxa present

- Metadata checks!
  - Removal of samples with missing metadata
  - Not suited metadata
  - Mistakes in metadata

➡ **We skip the metadata check today!**

Data visualisation

	SAMPLE01	SAMPLE20	SAMPLE13	SAMPLE07	SAMPLE14
Actinobacteria	5	35	144	13	73
Bacteroidetes	28776	14338	30089	12742	11735
Candidatus_Saccharibacteria	0	0	0	0	0
Cyanobacteria/Chloroplast	0	0	0	7	0
Elusimicrobia	0	0	0	0	0
Firmicutes	11532	824	6059	2842	6929
Fusobacteria	0	0	7606	2781	455
Lentisphaerae	0	0	0	0	0
Proteobacteria	20753	24296	13089	13303	11233
Verrucomicrobia	0	0	0	0	0
Unknown	0	0	0	0	0

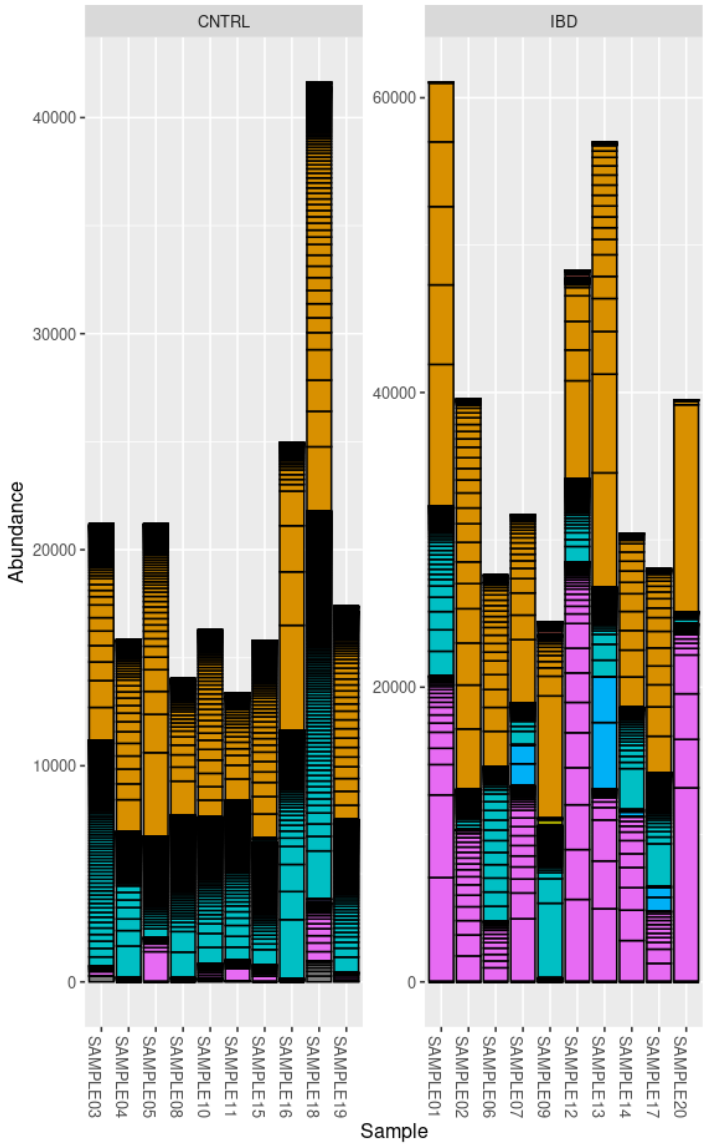


Data visualisation

Always visualize your data before doing time-consuming analysis!

Things to check:

- Does the data look as expected?
- Batch effects?
- Outliers?



Data normalization

- Rare taxas
- Read counts differences between samples

	SAMPLE01	SAMPLE20	SAMPLE13	SAMPLE07	SAMPLE14
Actinobacteria	5	35	144	13	73
Bacteroidetes	28776	14338	30089	12742	11735
Candidatus_Saccharibacteria	0	0	0	0	0
Cyanobacteria/Chloroplast	0	0	0	7	0
Elusimicrobia	0	0	0	0	0
Firmicutes	11532	824	6059	2842	6929
Fusobacteria	0	0	7606	2781	455
Lentisphaerae	0	0	0	0	0
Proteobacteria	20753	24296	13089	13303	11233
Verrucomicrobia	0	0	0	0	0
Unknown	0	0	0	0	0

Rare taxa

Removal of taxa with rare abundance to:

- Expected to be uninformative
- Speed up analysis
- Increase statistical power (correcting for many tests reduces statistical power)

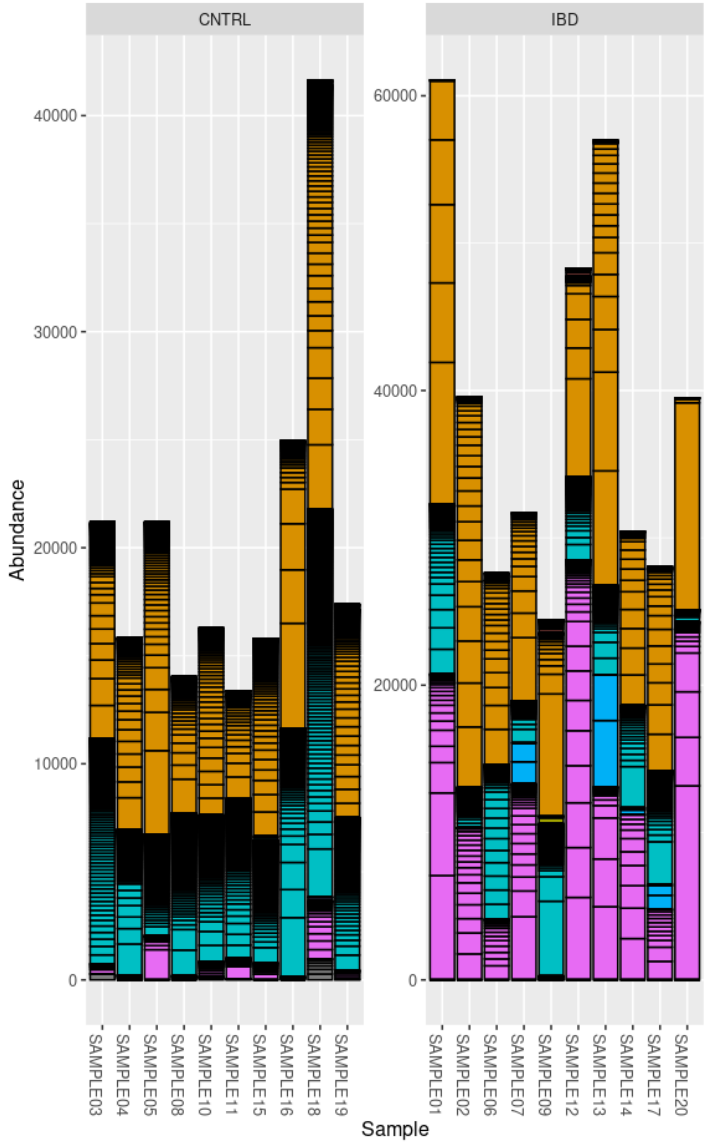
Problem: It can increase false positives

Advice: Use hard cut-offs for the prevalence and abundance of taxa across all samples, don't filter within groups

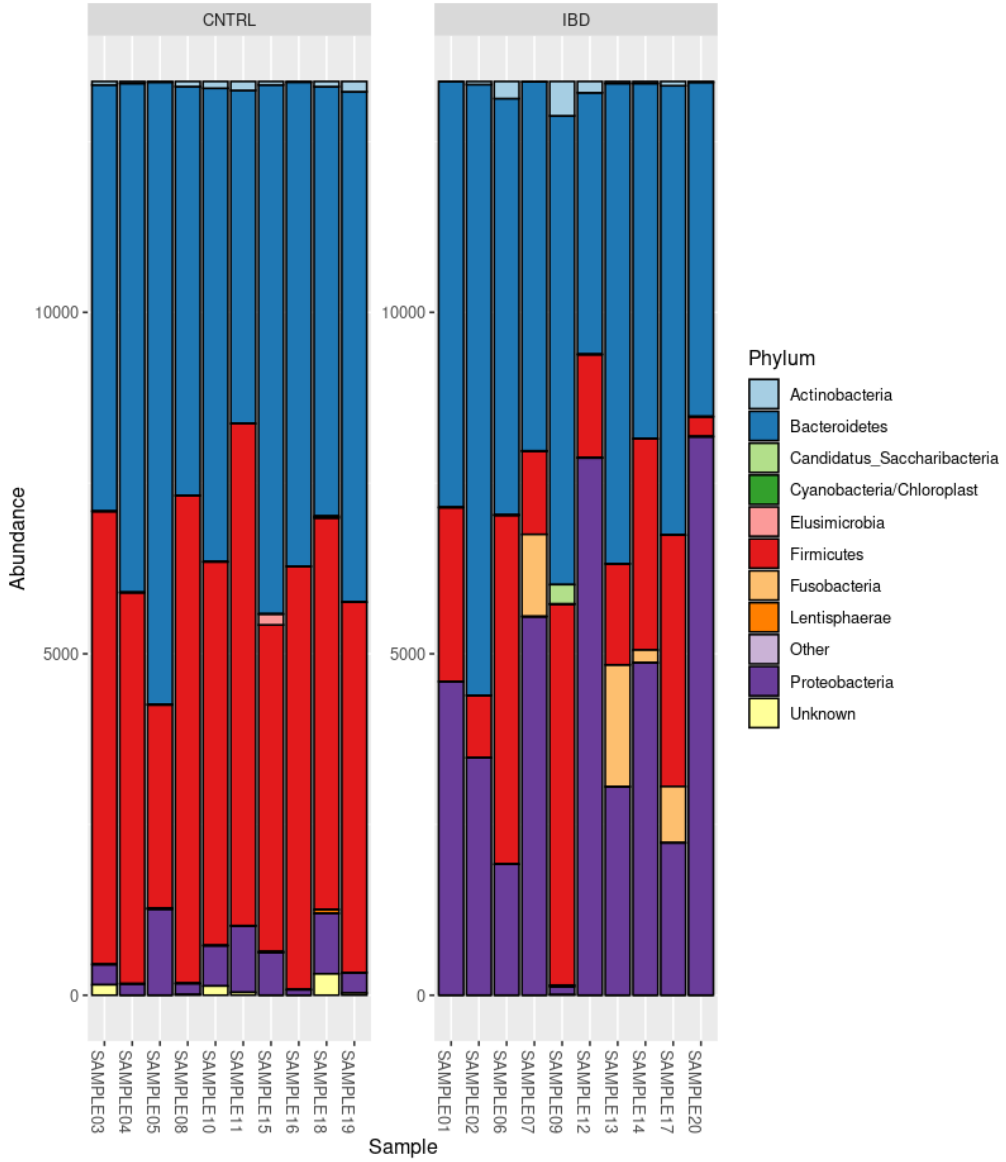
Possible filter: >10% of samples contain the taxa with >5% of relative abundance

	SAMPLE01	SAMPLE20	SAMPLE13	SAMPLE07	SAMPLE14
Actinobacteria	5	35	144	13	73
Bacteroidetes	28776	14338	30089	12742	11735
Candidatus_Saccharibacteria	0	0	0	0	0
Cyanobacteria/Chloroplast	0	0	0	7	0
Elusimicrobia	0	0	0	0	0
Firmicutes	11532	824	6059	2842	6929
Fusobacteria	0	0	7606	2781	455
Lentisphaerae	0	0	0	0	0
Proteobacteria	20753	24296	13089	13303	11233
Verrucomicrobia	0	0	0	0	0
Unknown	0	0	0	0	0

Rarefaction



Rarefaction



What is different between the phenotypes?

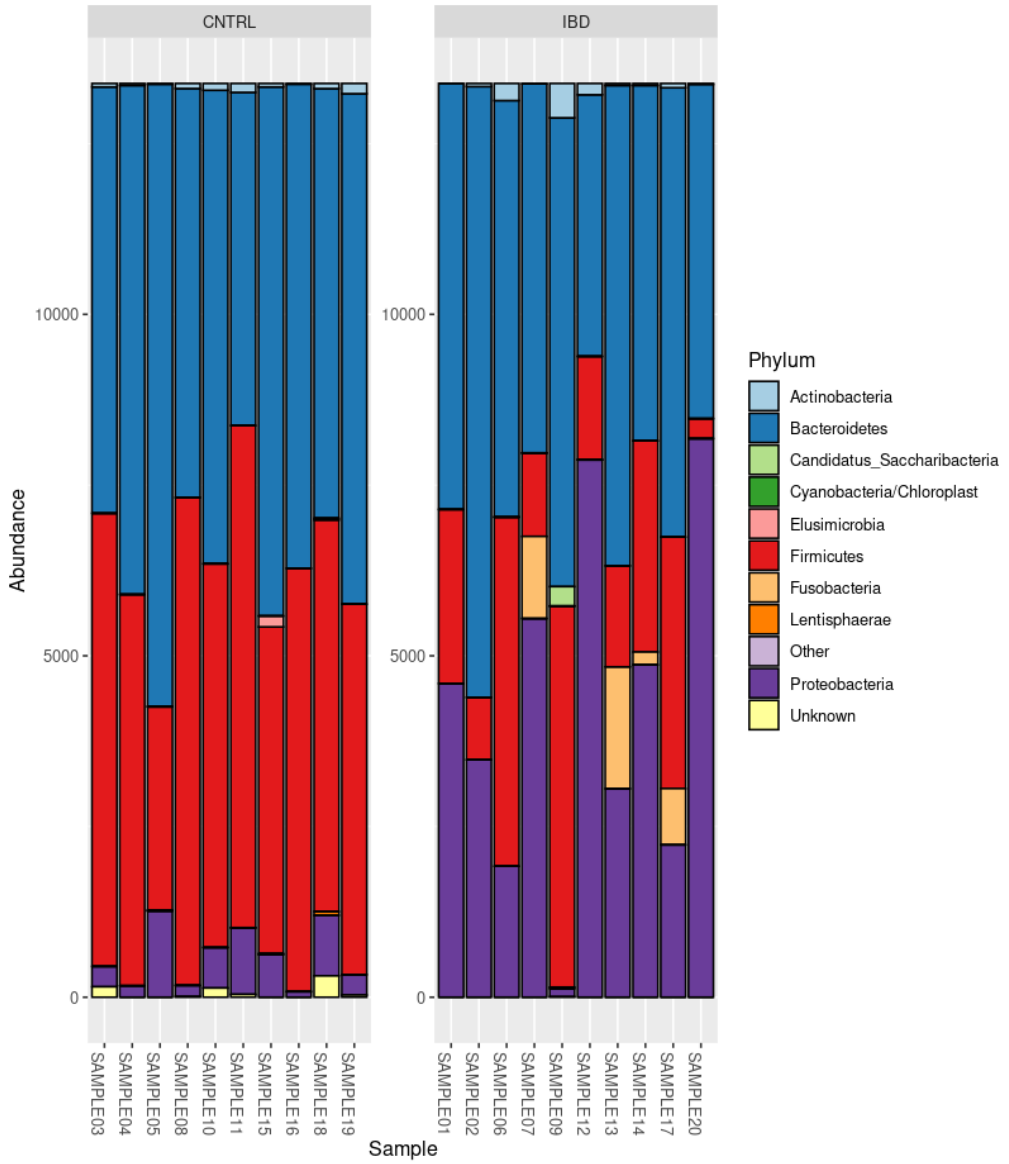
Statistical analysis

What is different between the phenotypes?

- Firmicutes ⬇
- Proteobacteria ⬆
- Actinobacteria ⬆ ⬇ **?**

- The other taxa?

<u>We need statistics to make clear statements!</u>



Transformation

Sequencing data is compositional

- Always represents relative abundance
- Each features abundance is depending on the abundance of other features

Limited to methods of compositional data analysis

The centered log-ratio (CLR)

- uses the geometric mean of the read counts of all taxa within a sample as a reference for that sample and applies log transformation
- Addition of a pseudocount (+1) to tackle 0 count values
- Compare log fold changes in this ratio between samples

$$clr(x) = \left[log \frac{x_1}{g(x)}, \dots, log \frac{x_D}{g(x)} \right]$$

g(x) is the geometric mean of x

Statistical tests

- We will apply Wilcoxon rank-sum test to CLR-transformed abundances on each taxa
- Wilcoxon is non-parametric: No normal distribution in each taxa needed
- By chance we will find low p-values when testing many times – Correction is needed!
- Applying multiple testing correction:
  - Bonferroni
  - FDR control/Benjamini Hochberg

DESeq2

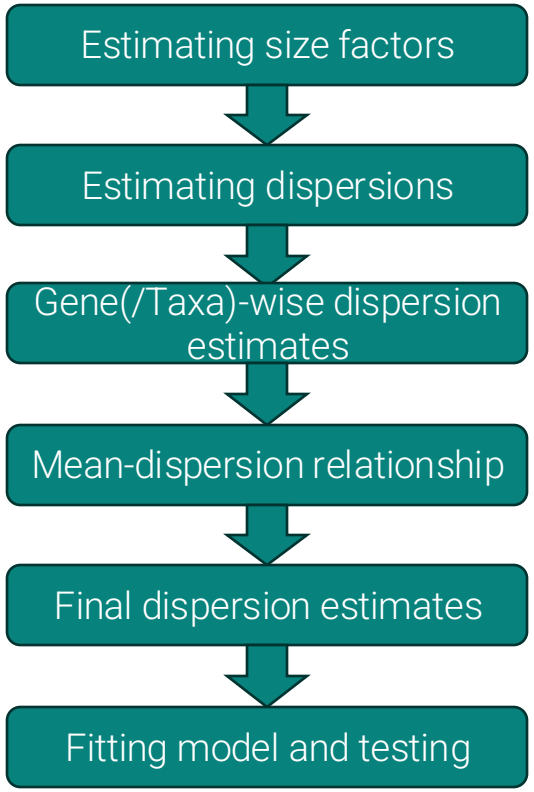
DESeq2 was developed for RNAseq analysis

Input are the raw counts!

Counts are normalized based on a per taxa normalization factor and their dispersion

Counts are fitted to a generalized linear model with negative binomial distribution

DESeq2 uses multiple steps to do this, but it can do all of them automatically



MaAsLin2 (Microbiome Multivariable Association with Linear Models)

Toolbox for association of metadata to omics data

- Uses a linear mixed model
- Offers a broad range of transformation and normalization methods
- Input are the raw counts!
- Normalization:
  - Total Sum Scaling (TSS),
  - CLR,
  - Trimmed Mean M-values (TMM),
  - Cumulative Sum Scaling (CSS)
  - NONE
- Transformation: LOG, LOGIT, arcsin square root (AST), NONE
- Creates publication ready plots

https://huttenhower.sph.harvard.edu/tools

Summary

- What taxa <u>significantly</u> differ in relative abundance between sample groupings?
- Transformations are recommended:
  - Centered log-ratio (CLR) transformation
  - Tool specific transformations
- Tools can make your job easier

HANDS ON PART

Script:
~/kmc_workshop/scripts/5_differential_abundance.ipynb

Metagenomics – A short overview

Eike Matthias Wacker

IKMB
Institute of Clinical Molecular Biology

Introduction

Also called shotgun metagenomics

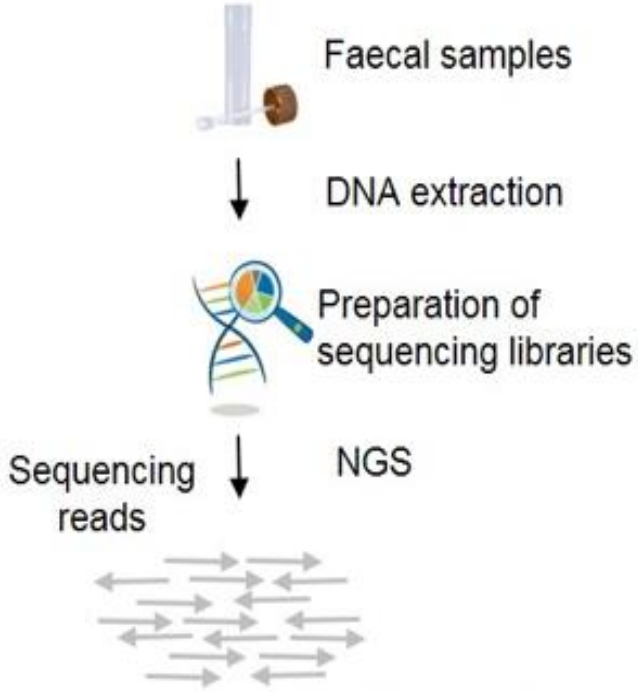
Sequencing of all available DNA (incl. host DNA) without targeting specific (bacteria-specific) marker genes

Current predominant technology is the Illumina platform (short reads) with reads < 300 bp

Long-read sequencing is currently emerging:

Nanopore: 10-100 kb per read

PacBio: 10–25 kb per read



Introduction

Host read removal important step for privacy compliant analysis

Faecal samples contain less than 10% of host DNA but saliva, nasal cavity, skin and vaginal samples are dominated by host DNA

nature microbiology

Article

https://doi.org/10.1038/s41564-023-01381-3

Reconstruction of the personal information from human genome reads in gut metagenome sequencing data

Received: 6 April 2022

Accepted: 12 April 2023

Published online: 15 May 2023

Check for updates

Yoshihiko Tomofuji[1,2,3], Kyuto Sonehara[1,2,4], Toshihiro Kishikawa[1,5,6], Yuichi Maeda[2,7,8], Kotaro Ogawa[9], Shuhei Kawabata[10], Takuro Nii[7,8], Tatsusada Okuno[9], Eri Oguro-Igashira[7,8], Makoto Kinoshita[9], Masatoshi Takagaki[10], Kenichi Yamamoto[1,11,12], Takashi Kurakawa[8], Mayu Yagita-Sakamaki[7,8], Akiko Hosokawa[9,13], Daisuke Motooka[2,14], Yuki Matsumoto[14], Hidetoshi Matsuoka[15], Maiko Yoshimura[15], Shiro Ohshima[15], Shota Nakamura[2,14,16], Hidenori Inohara[5], Haruhiko Kishima[10], Hideki Mochizuki[9], Kiyoshi Takeda[8,16,17], Atsushi Kumanogoh[2,7,18] & Yukinori Okada[1,2,3,4,12,16]

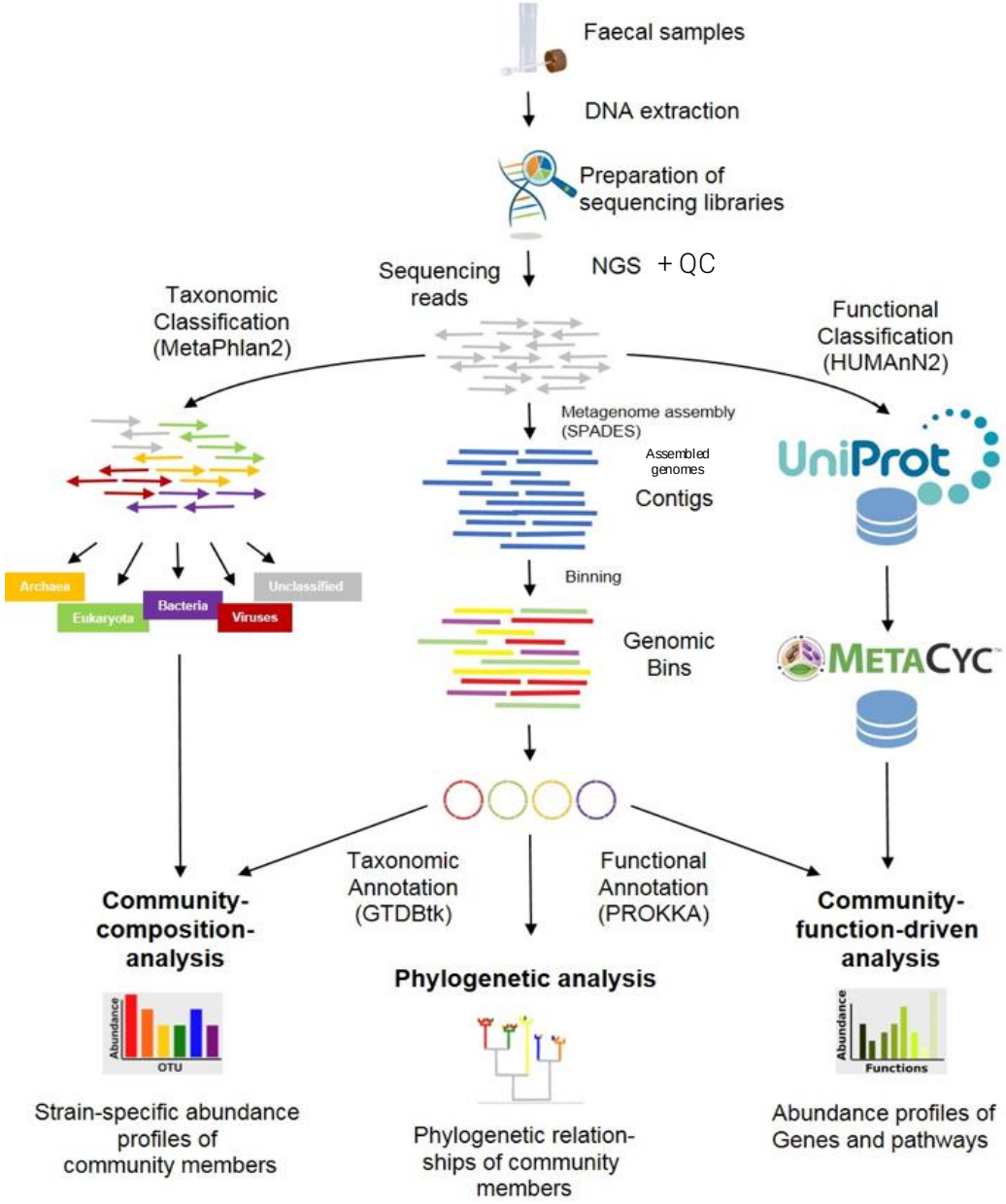
Processing Overview

QC:

- Trimming (read quality)
- Sequencing artefacts removal
- Host reads removal?

Possible analysis:

- Taxonomic abundance
  - → Strain analysis
- Functional classification
- Gene content analysis
- Virus/Phage identification
- Denovo metagenomic genome assembly



Taxonomic Abundance estimation

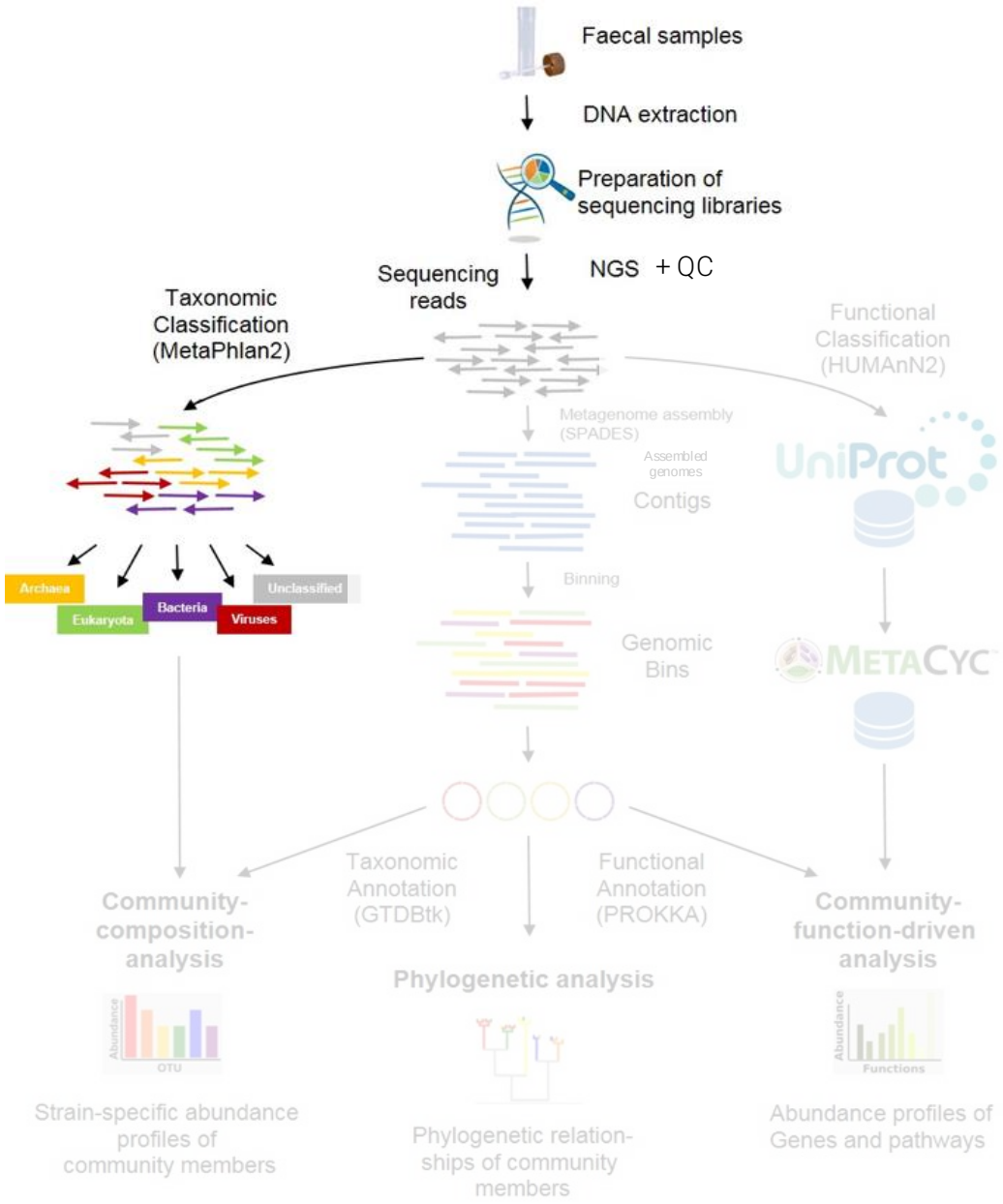
From short reads to abundance table

Tools:

- MetaPhlAn
- Sylph
- Salmon
- Kraken2
- DIAMOND
- …

Reference based -> database needed

No denovo findings possible



Functional classification

From short reads to pathway and gene abundances

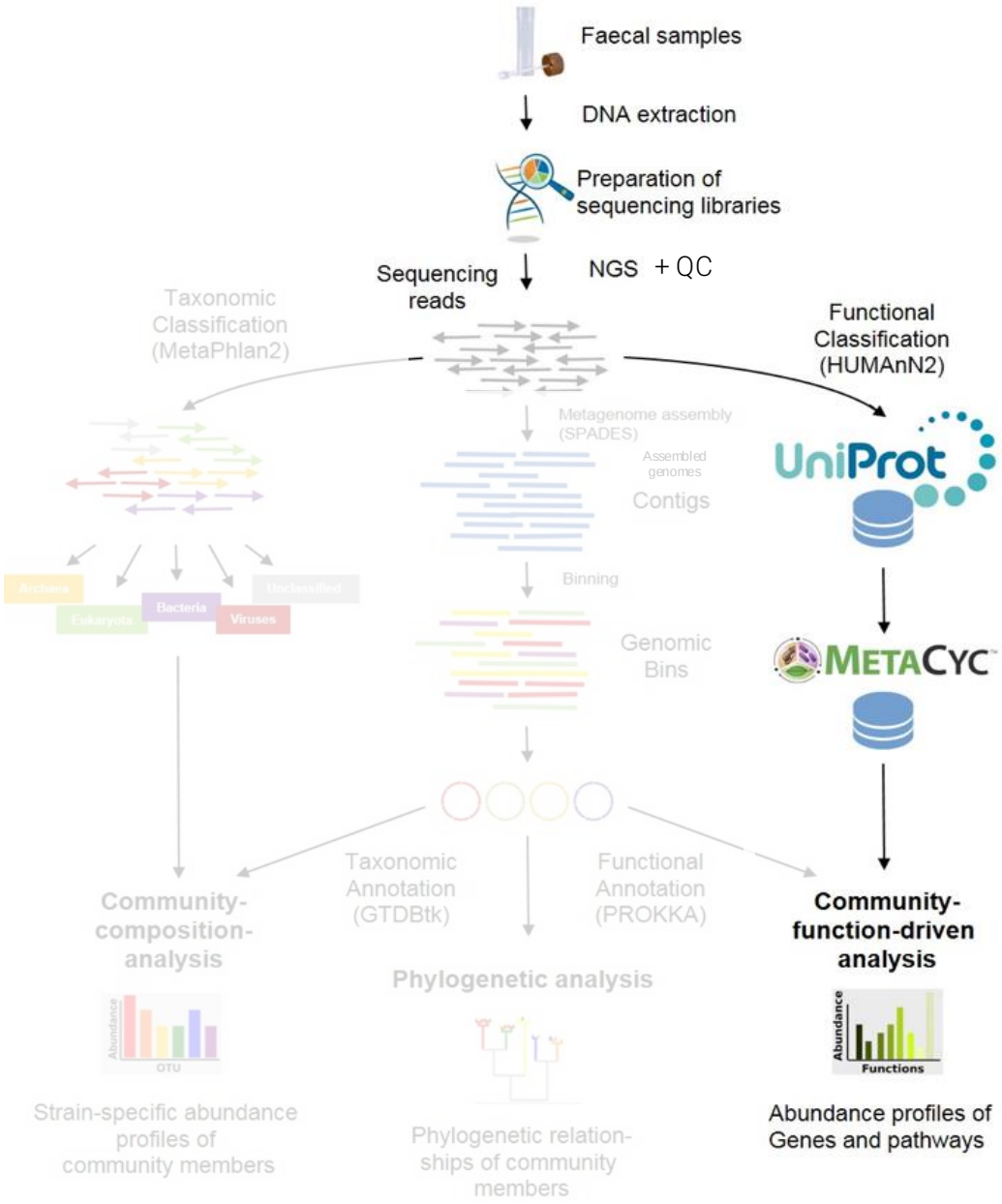
Non-assembly based tools:

- HUMAnN
- DIAMOND
- …

Reference based -> database needed
No denovo findings possible

Assembly approach:

- assemble contigs
- predict genes from contigs



Genome assembly

Overlapping sequencing reads will be assembled to longer fragments (Contigs)

Contigs will then binned by:

Contig coverage

GC content

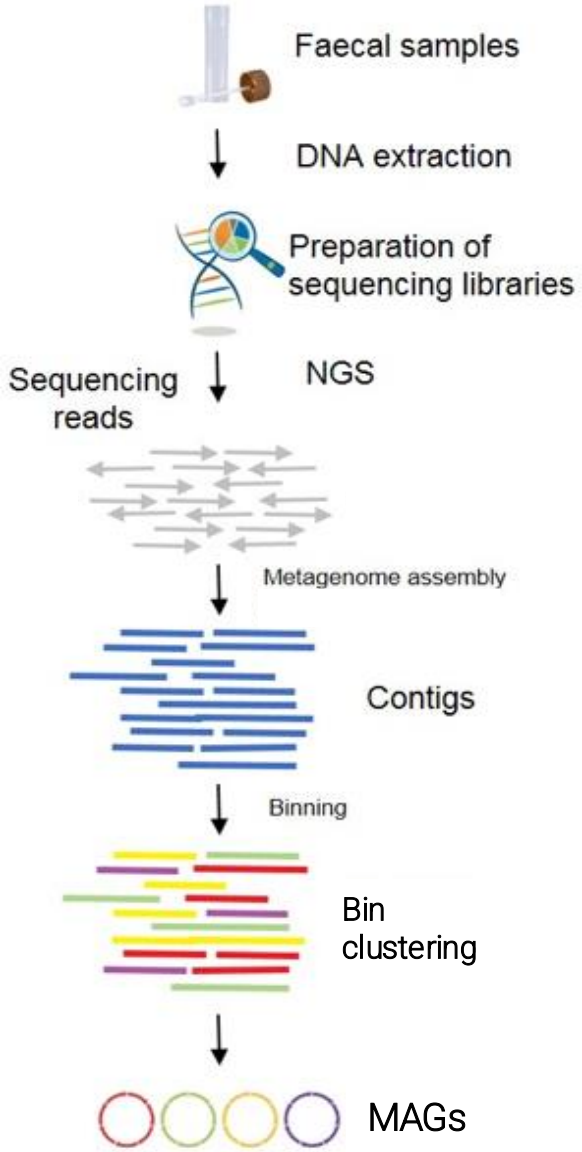
Based on different clustering and ML methods

Bins can be scored based on single-copy gene content (genes like 16S rRNA) and taxonomically defined

Wide range of tools available for all these steps:

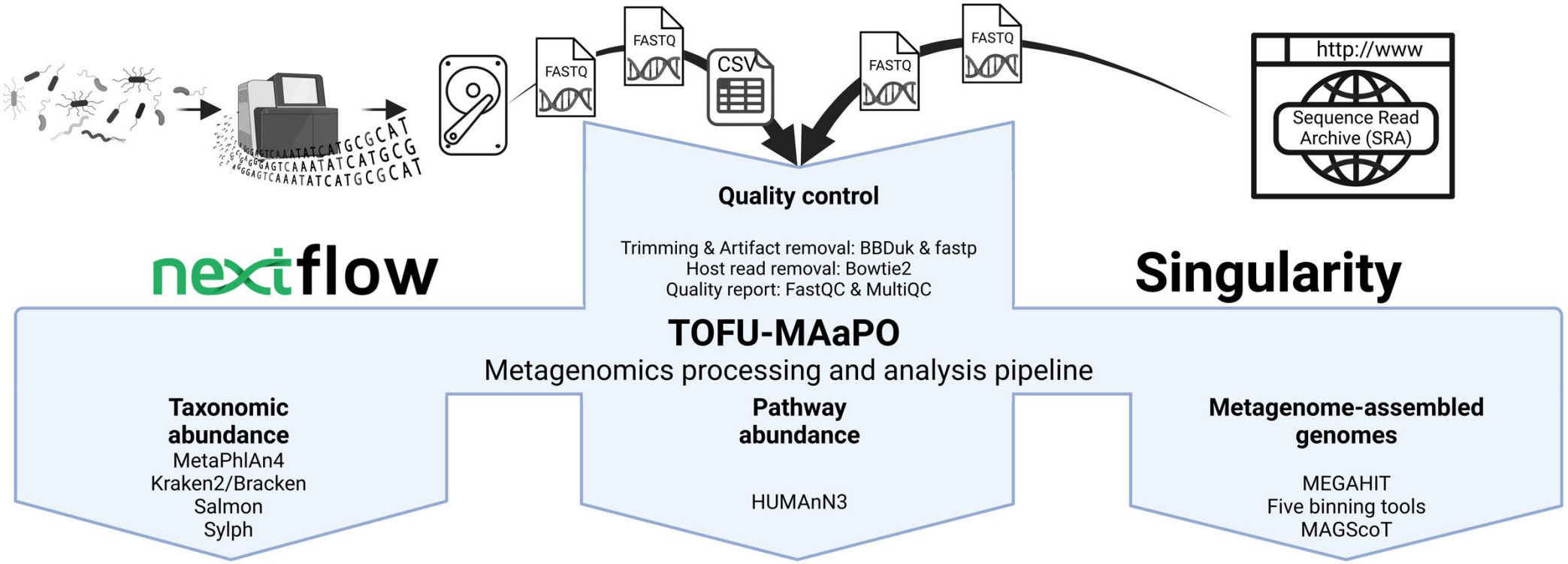
Assembler -> **Binning Tool(s)** (-> **Refinement**) -> **Assessment**

Megahit/Spades -> Metabat2/Concoct/vamb/... (-> DAS-Tool/MAGScoT) -> checkm/gtdb-tk



TOFU-MAaPO

Single command pipeline written in Nextflow, using Docker containers for scalable, reproducable processing and analysis of public and locally available metagenomic short reads.



github.com/ikmb/TOFU-MAaPO

Project example

HANDS ON PART

Script: ~/kmc_workshop/scripts/metagenome_outlook.ipynb