THE 4th VIETNAM SCHOOL OF BIOLOGY (VSOB-4)

Metagenome Analysis in One Health Practice – From 16S to Shotgun

September 03rd-06th, 2025, ICISE, Quy Nhon, Vietnam

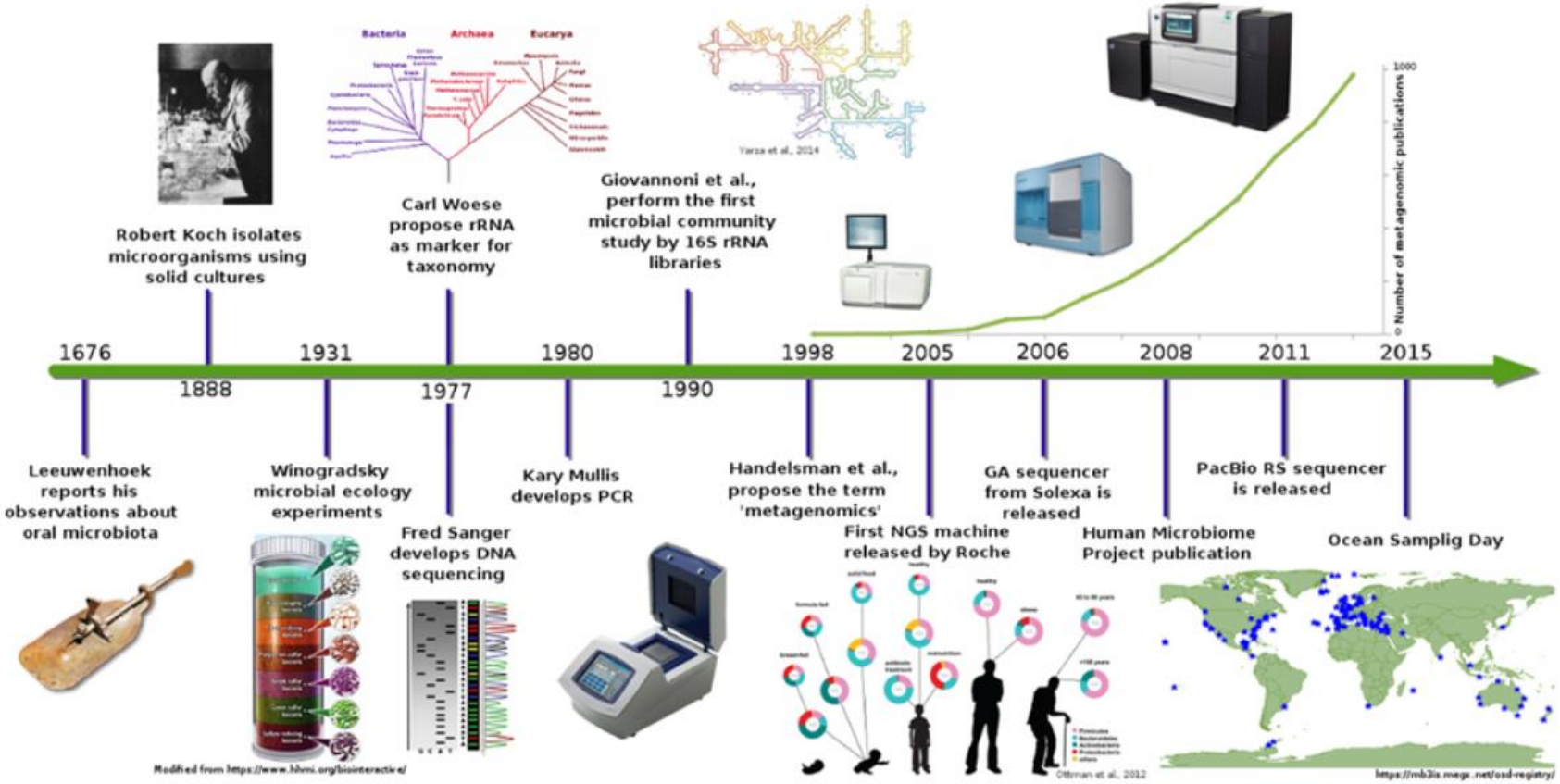
Lecture 1
Shotgun metagenome analysis

Lecturer: Pham Quang Huy, Phd candidate

Introduction

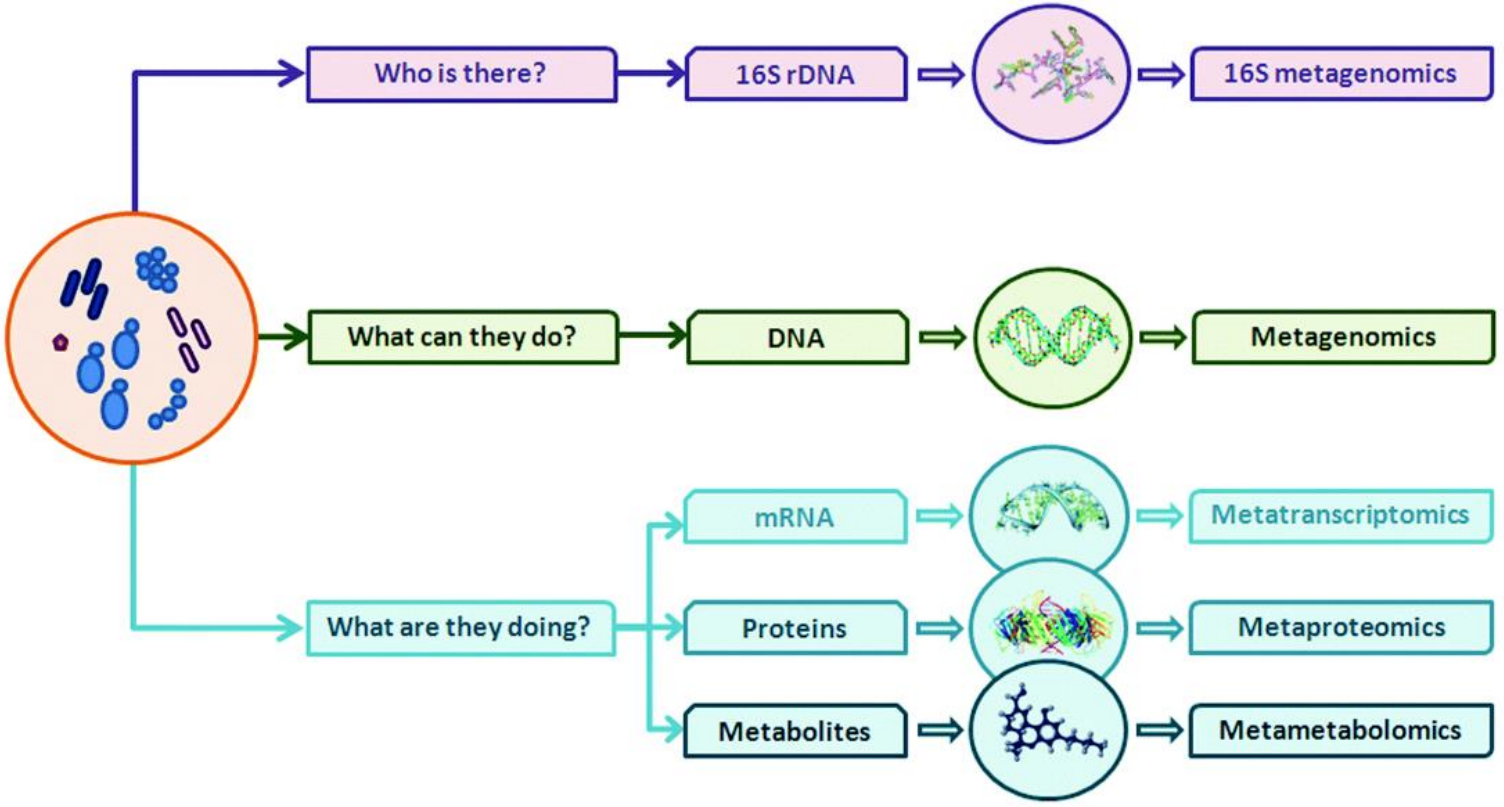
The road to metagenomics: from microbiology to DNA sequencing technologies and bioinformatics (2015) [1]

Introduction



The road to metagenomics: from microbiology to DNA sequencing technologies and bioinformatics (2015) [1]

Introduction



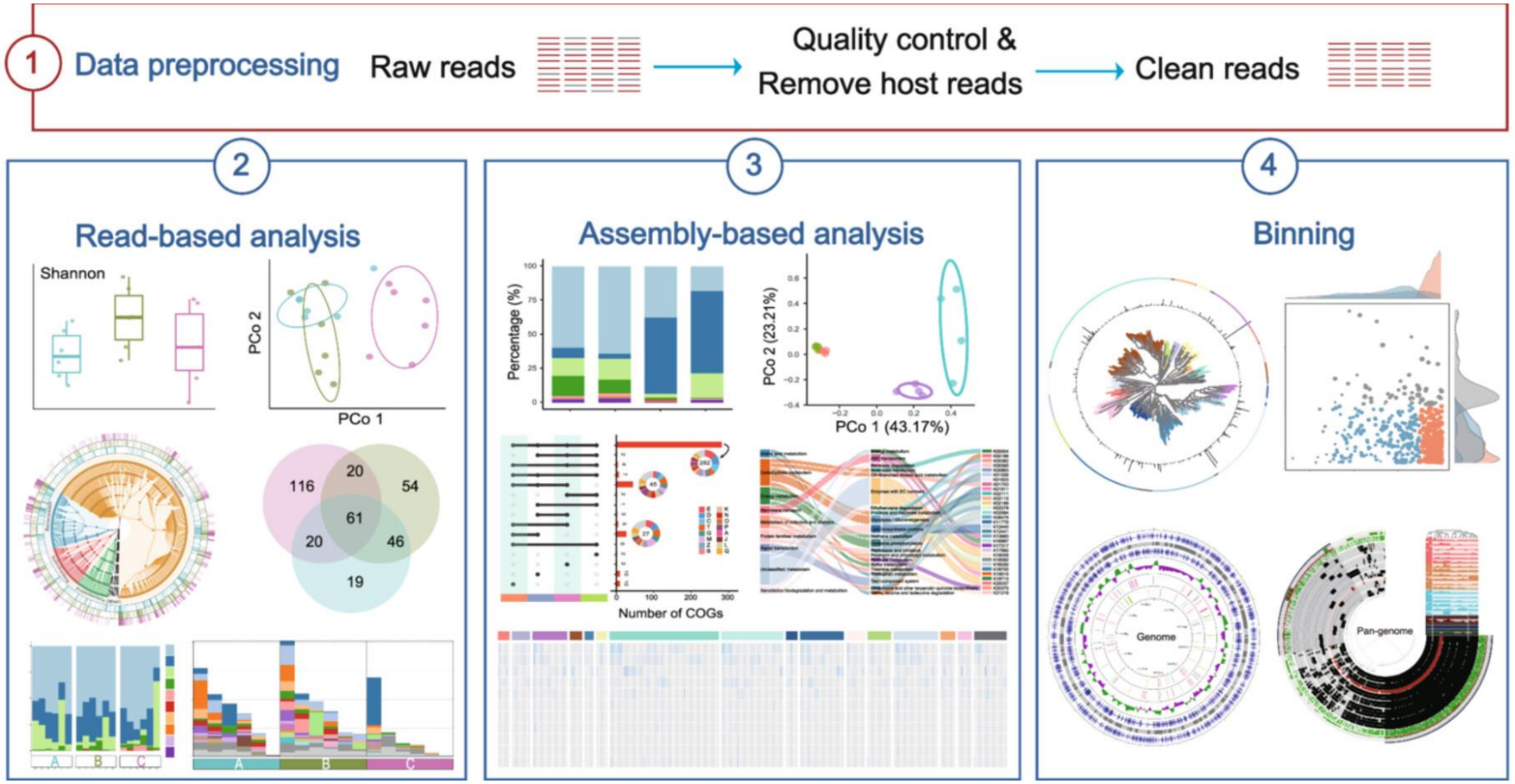
Introduction

Table 1. Metataxonomics, metagenomics and meta-transcriptomics strategies

Technique	Advantages and challenges	Main applications
Metataxonomics using amplicon sequencing of the 16S or 18S rRNA gene or ITS	+ Fast and cost-effective identification of a wide variety of bacteria and eukaryotes – Does not capture gene content other than the targeted genes – Amplification bias – Viruses cannot be captured	* Profiling of what is present * Microbial ecology * rRNA-based phylogeny
Metagenomics using random shotgun sequencing of DNA or RNA	+ No amplification bias + Detects bacteria, archaea, viruses and eukaryotes + Enables *de novo* assembly of genomes – Requires high read count – Many reads may be from host – Requires reference genomes for classification	* Profiling of what is present across all domains * Functional genome analyses * Phylogeny * Detection of pathogens
Meta-transcriptomics using sequencing of mRNA	+ Identifies active genes and pathways – mRNA is unstable – Multiple purification and amplification steps can lead to more noise	* Transcriptional profiling of what is active

Breitwieser, Lu and Salzberg (2019)

Introduction



Introduction

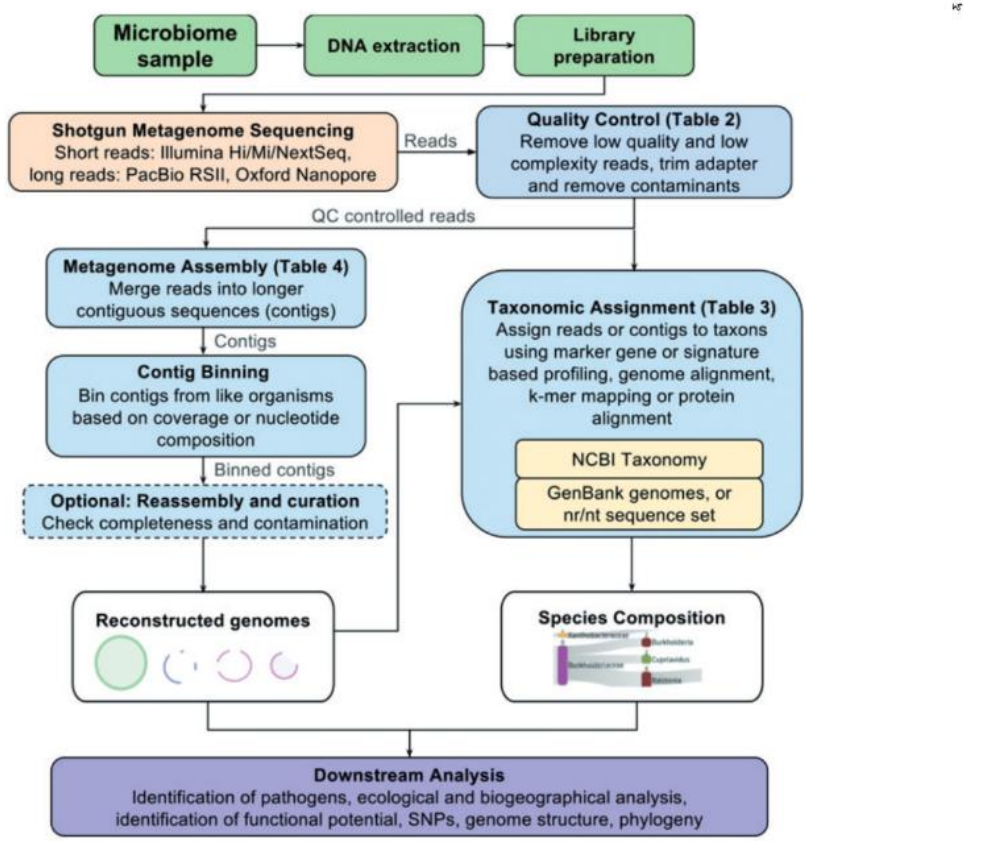
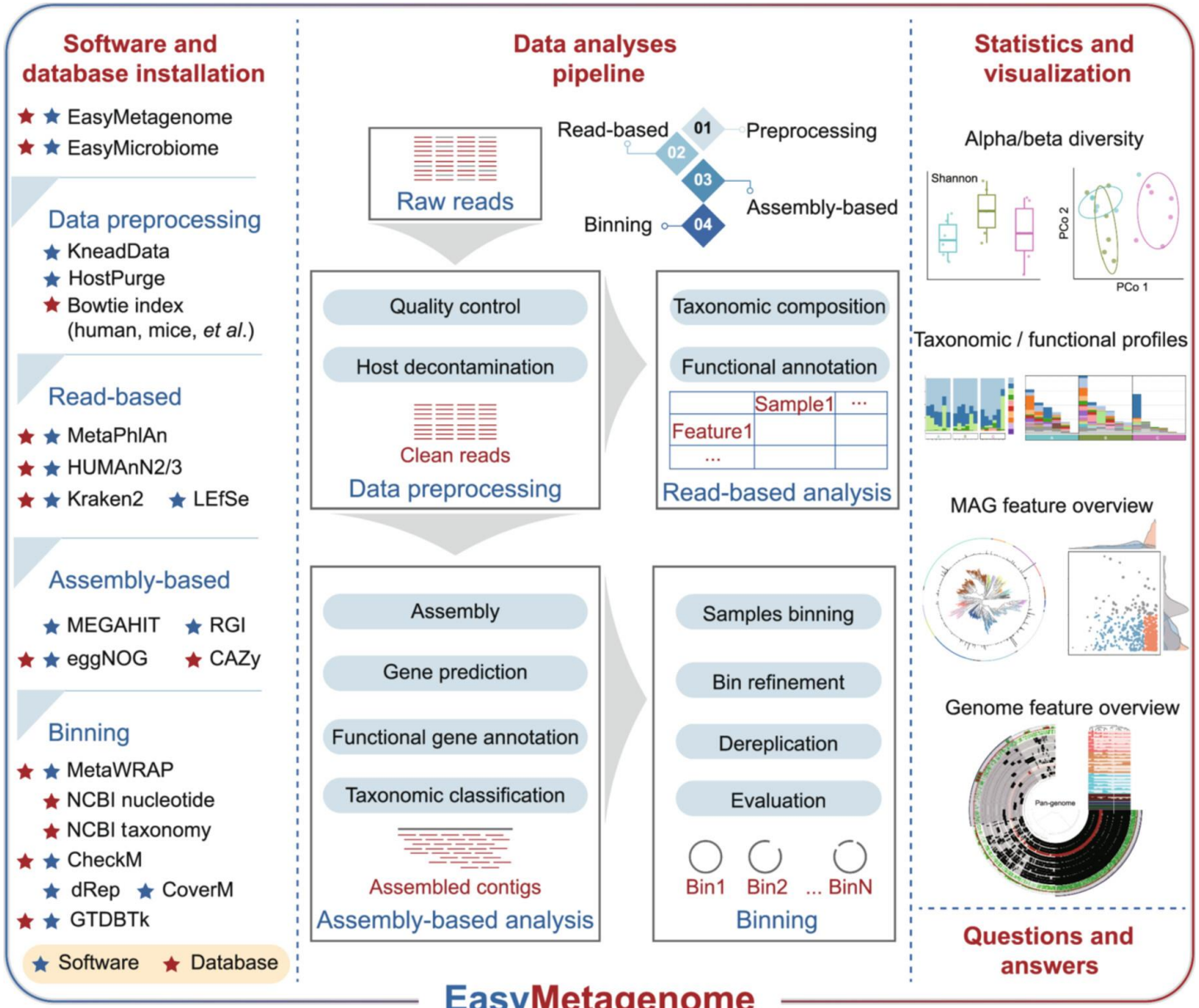


Figure 1. Common analysis procedures for metagenomics data. Note that the order of some of the analysis steps can be shuffled. For example, reads might be binned before assembly or before taxonomic assignment, so that the downstream algorithms can work only with a subset of the data.

Breitwieser, Lu and Salzberg (2019)

Introduction

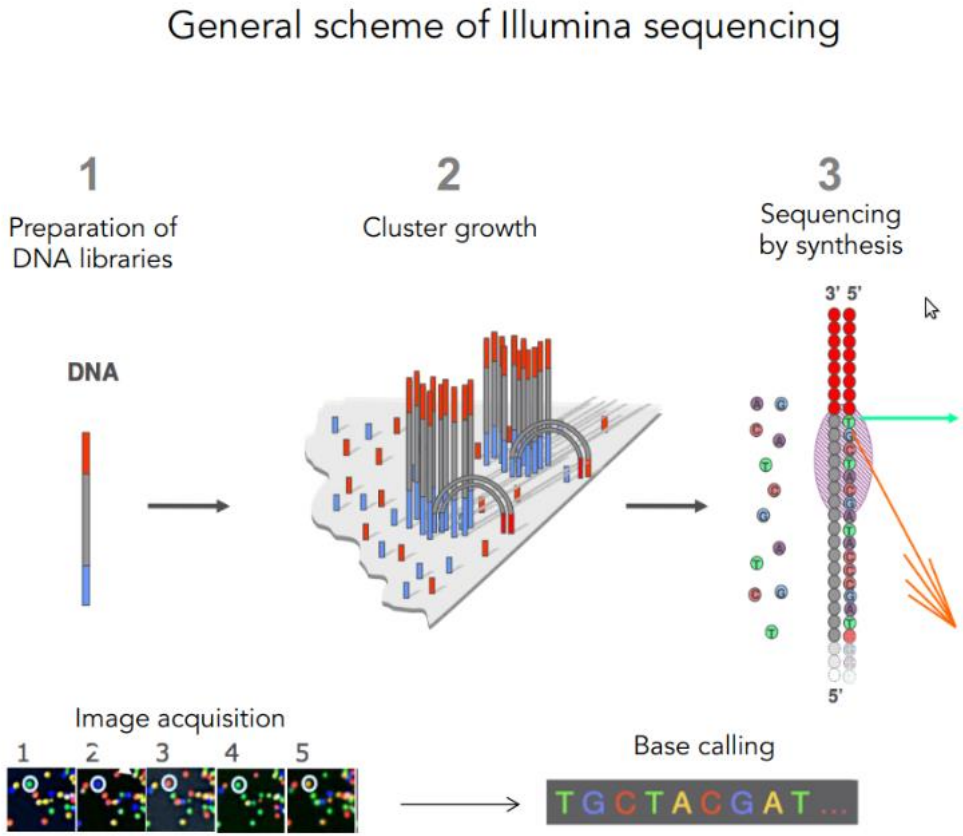


Breitwieser, Lu and Salzberg (2019)

Reminders about sequencing

The road to metagenomics: from microbiology to DNA sequencing technologies and bioinformatics (2015) [1]

Sequencing Vocabulary



Sequencing Vocabulary

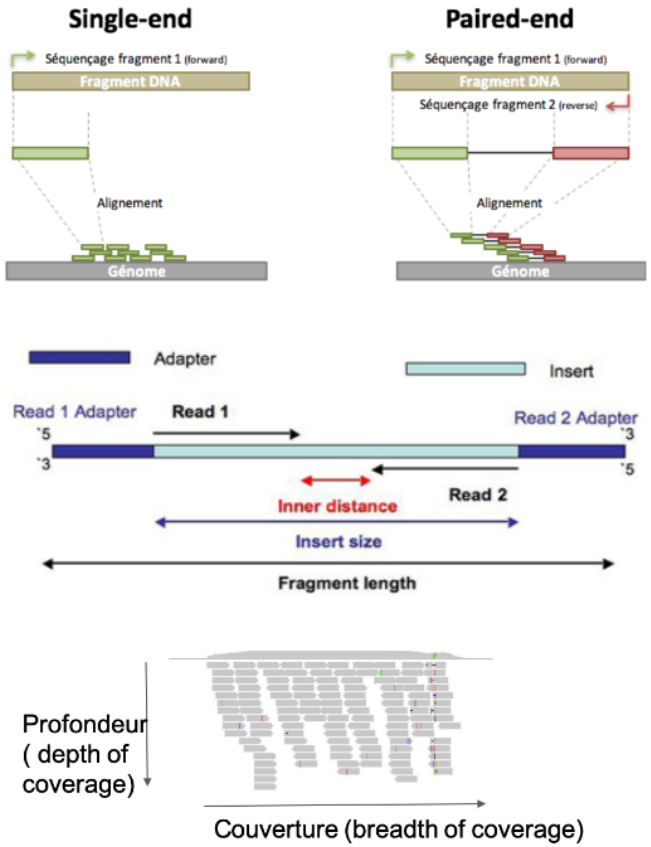
Read : piece of sequenced DNA

DNA fragment = 1 or more reads depending on whether the sequencing is single end or paird-end

Insert = Fragment size

Depth = $N * L/G$
N = number of reads,
L = size,
G = genome size

Coverage = % of genome covered



FASTQ syntax

The FASTQ format is the de facto standard by which all sequencing instruments represent data. It may be thought of as a variant of the FASTA format that allows it to associate a quality measure to each sequence base: **FASTA with QUALITIES**.

FASTQ syntax

The FASTQ format consists of 4 sections:

A FASTA-like header, but instead of the > symbol it uses the @ symbol. This is followed by an ID and more optional text, similar to the FASTA headers.
The second section contains the measured sequence (typically on a single line), but it may be wrapped until the + sign starts the next section.
The third section is marked by the + sign and may be optionally followed by the same sequence id and header as the first section
The last line encodes the quality values for the sequence in section 2, and must be of the same length as section 2.

FASTQ syntax

Example

```
@SEQ_ID
GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTGTTCAACTCACAGTTT
+
!''*((((***+))%%%++)(%%%%).1***-+*''))**55CCF>>>>>>CCCCCCC65
```

FASTQ quality

Each character represents a numerical value: a so-called Phred score, encoded via a single letter encoding.

```
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHI
|    |    |    |    |    |    |    |    |
0....5...10...15...20...25...30...35...40
|    |    |    |    |    |    |    |    |
worst.................................best
```

The numbers represent the error probabilities via the formula:

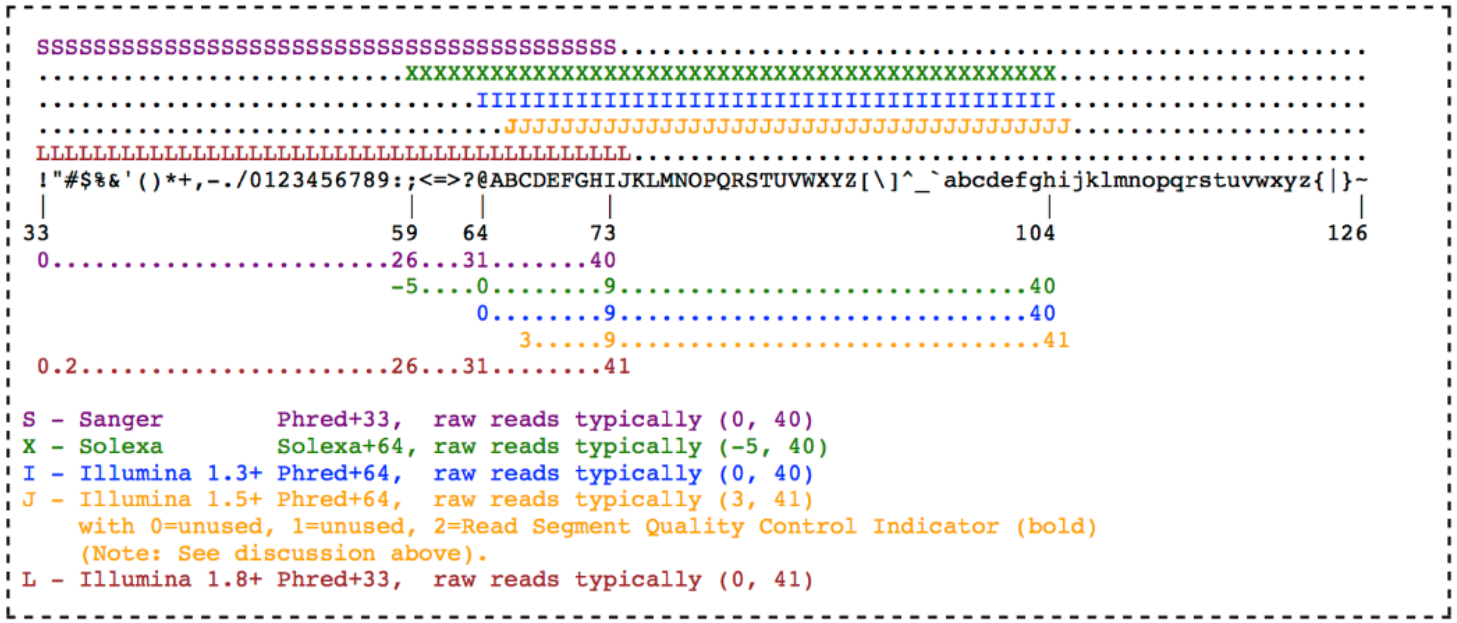
$$Error = 10^{-P/10}$$

It is basically summarized as:

- P=0 means 1/1 (100% probability of error)
- P=10 means 1/10 (10% probability of error)
- P=20 means 1/100 (1% probability of error)
- P=30 means 1/1000 (0.1% probability of error)

FASTQ quality encoding specificities

There was a time when instrumentation makers could not decide at what character to start the scale. The **current standard** shown above is the so-called Sanger (+33) format where the ASCII codes are shifted by 33. There is the so-called +64 format that starts close to where the other scale ends.



FASTQ Header informations

Information is often encoded in the “free” text section of a FASTQ file.

`@EAS139:136:FC706VJ:2:2104:15343:197393 1:Y:18:ATCACG` contains the following information:

- `EAS139`: the unique instrument name
- `136`: the run id
- `FC706VJ`: the flowcell id
- `2`: flowcell lane
- `2104`: tile number within the flowcell lane
- `15343`: ‘x’-coordinate of the cluster within the tile
- `197393`: ‘y’-coordinate of the cluster within the tile
- `1`: the member of a pair, 1 or 2 (paired-end or mate-pair reads only)
- `Y`: Y if the read is filtered, N otherwise
- `18`: 0 when none of the control bits are on, otherwise it is an even number
- `ATCACG`: index sequence

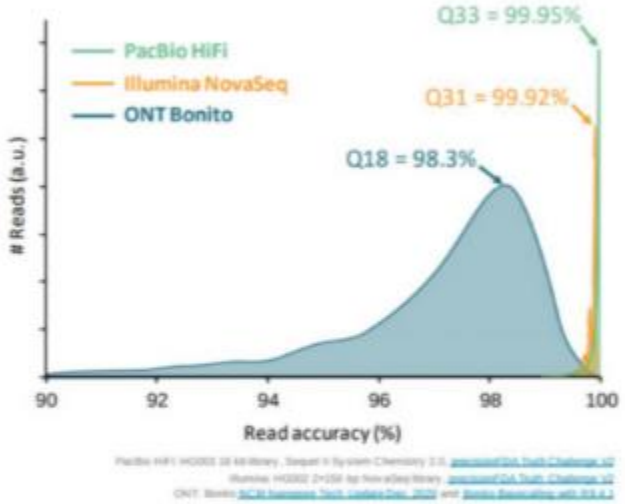
FASTQ compression

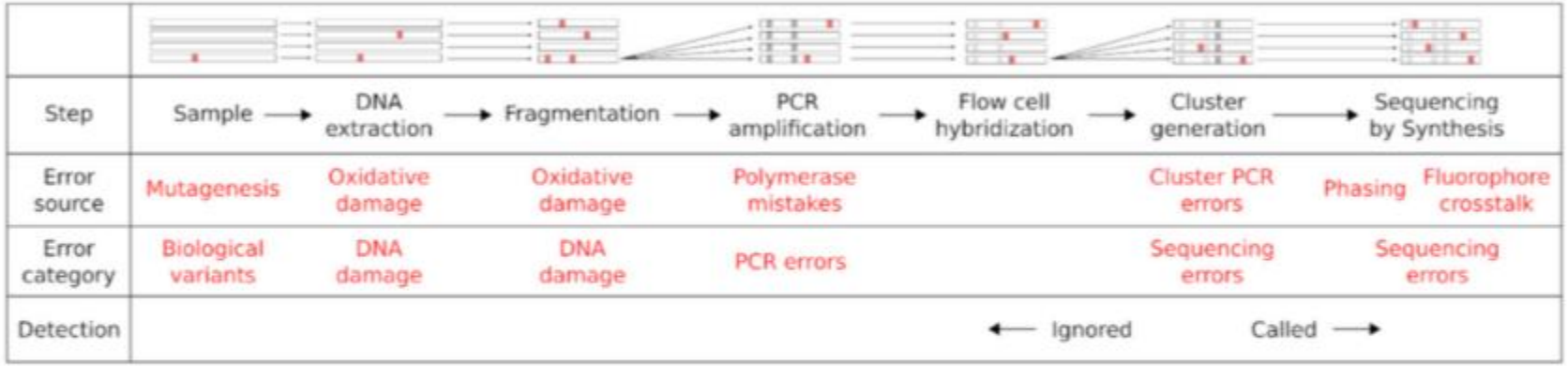
- Compression is essential to deal with FASTQ files (reduce disk storage)
- extension: file.fastq.gz
- Tools are (almost all) able to deal with compressed files

Quality control

- One of the most easy step in bioinformatics …
- … but one of the most important
- check if everything is ok
- Indicates if/how to clean reads
- Shows possible sequencing problems
- The results must be interpreted in relation to what has been sequenced

Reads are not perfect





Step	Sample →	DNA extraction →	Fragmentation →	PCR amplification →	Flow cell hybridization →	Cluster generation →	Sequencing by Synthesis
Error source	Mutagenesis	Oxidative damage	Oxidative damage	Polymerase mistakes		Cluster PCR errors	Phasing, Fluorophore crosstalk
Error category	Biological variants	DNA damage	DNA damage	PCR errors		Sequencing errors	Sequencing errors
Detection					← Ignored	Called →	

Why QC'ing your reads?

Try to answer to (not always) simple questions:

- Are data conform to the expected level of performance?
  - Size / Number of reads / Quality
- Residual presence of adapters or indexes?
- (Un)expected techincal biases?
- (Un)expected biological biases?

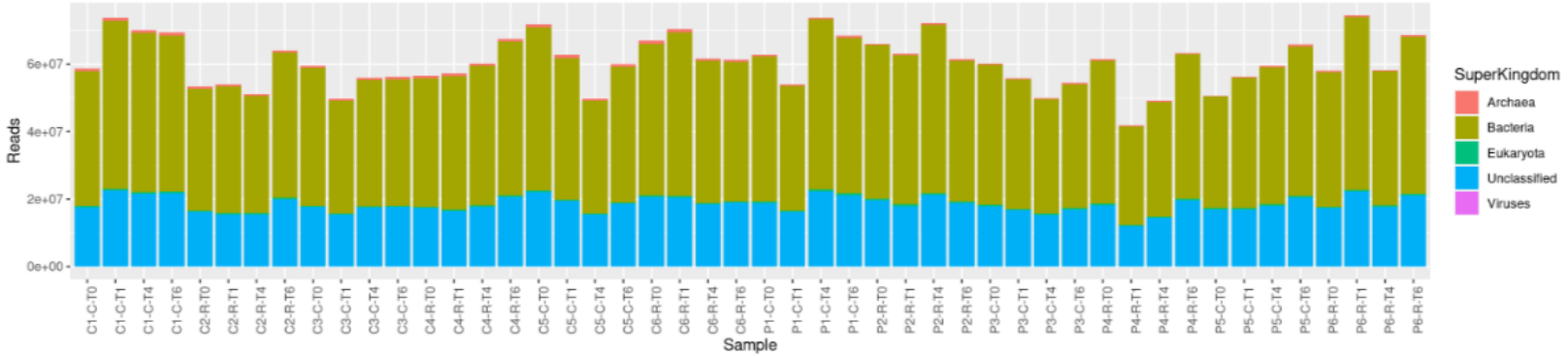
Taxonomic affiliation

The road to metagenomics: from microbiology to DNA sequencing technologies and bioinformatics (2015) [1]

Taxonomic affiliation

Definition

Taxonomic assignment in the context of bioinformatics involves the computational identification and classification of organisms into their taxonomic groups using various data sources, such as DNA sequences, protein sequences, or other molecular markers. This process typically utilizes algorithms and computational tools to compare sequences against reference databases or phylogenetic trees, allowing for accurate identification and classification of organisms at different taxonomic levels.



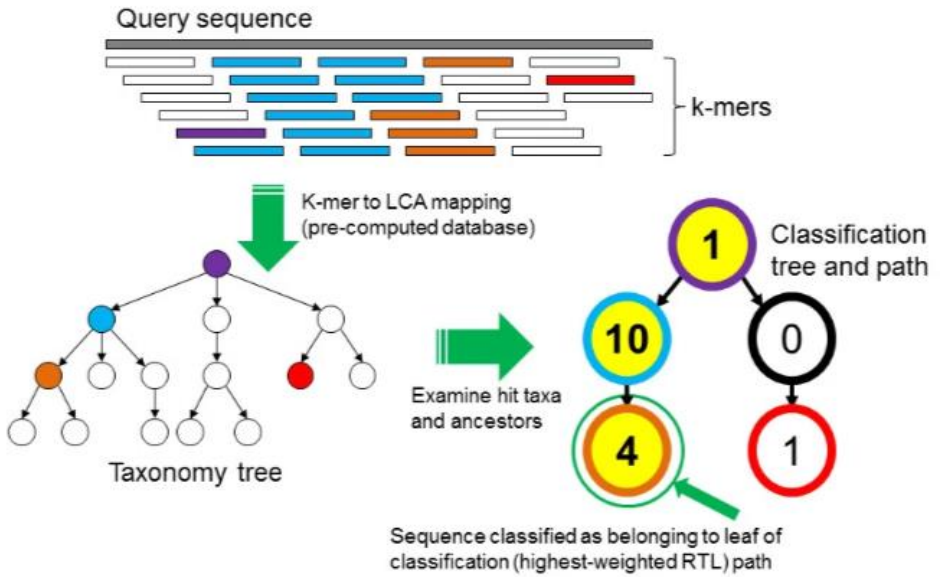
Kraken

Kraken [5] is a very popular taxonomic affiliation tool. It is very fast and accurate. Kraken examines the k-mers (~35 bp) within the query sequence, searches for them in the database, looks for where these are placed within the taxonomy tree inside the database, makes the classification with the most probable position, then maps k-mers to the lowest common ancestor (LCA) of all genomes known to contain the given k-mer.

Method:

1. Chop genomes into k-mers and link to a taxonomic id.
2. Chop reads into k-mers and search for exact hits in database
3. Search for highest-wighted root-to-leaf paths and assign the taxonomic id of the lowest node to read

Kraken



The Kraken sequence classification algorithm. To classify a sequence, each *k*-mer in the sequence is mapped to the lowest common ancestor (LCA) of the genomes that contain that *k*-mer in a database. The taxa associated with the sequence's *k*-mers, as well as the taxa's ancestors, form a pruned subtree of the general taxonomy tree, which is used for classification. In the classification tree, each node has a weight equal to the number of *k*-mers in the sequence associated with the node's taxon. Each root-to-leaf (RTL) path in the classification tree is scored by adding all weights in the path, and the maximal RTL path in the classification tree is the classification path (nodes highlighted in yellow). The leaf of this classification path (the orange, leftmost leaf in the classification tree) is the classification used for the query sequence.

Kaiju

Kaiju [6] is an equivalent of Kraken, but with some particularities:

- Database of proteic sequences
- Supposed to be more sensitive
- Translate reads in all six reading frames, split at stop codons
- Use BWT and FM-index table

Kaiju

Option	Description	Sequences*	RAM in GB (makedb)*
`refseq`	Completely assembled and annotated reference genomes of Archaea, Bacteria, and viruses from the NCBI RefSeq database.	63M	43 (55)
`progenomes`	Representative set of genomes from the proGenomes database and viruses from the NCBI RefSeq database.	41.8M	30 (35)
`viruses`	Only viruses from the NCBI RefSeq database.	0.37M	0.3 (0.3)
`plasmids`	Plasmid sequences from the NCBI RefSeq database.	2M	1.3 (2)
`fungi`	Fungi sequences from the NCBI RefSeq database.	3.2M	3 (4)
`nr`	Subset of NCBI BLAST *nr* database containing all proteins belonging to Archaea, Bacteria and Viruses.	196M	105 (175)
`nr_euk`	Like option `-s nr` and additionally include proteins from fungi and microbial eukaryotes, see taxon list in `bin/kaiju-taxonlistEuk.tsv`.	213M	117 (194)
`mar`	Protein sequences from all Mar databases. Subsets can be chosen by `mar_ref`, `mar_db`, or `mar_mag`.	32.6M	21 (27)
`rvdb`	Protein sequences from RVDB-prot	4.6M	4 (149)

Taxonomic classification caveats

- Databanks
- K-mer choice (sensitivity / specificity)
- Allow a “fast” overview of your data
  - Contaminants?
  - Host reads?
  - Classification rate

Some taxonomic affiliation tools

tool	migale	comments
Kraken2	✔	the reference, fast and efficient
Kaiju	✔	protein level
Bracken	✔	Bayesian Reestimation of Abundance with Kraken
Centrifuge	✔	indexing scheme based on the Burrows-Wheeler transform (BWT) and the Ferragina-Manzini (FM) index, optimized specifically for the metagenomic classification problem
MetaPhlAn3	✔	MetaPhlAn relies on unique clade-specific marker genes identified from ~17,000 reference genomes (~13,500 bacterial and archaeal, ~3,500 viral, and ~110 eukaryotic)

Reads cleaning

The road to metagenomics: from microbiology to DNA sequencing technologies and bioinformatics (2015) [1]

Quality/Length/Adapters

- Detect and remove sequencing adapters present in the FASTQ files
- Filter / trim reads according to quality (seen with FastQC)

rRNA removal

- Mandatory step if you want to skip rRNA reads

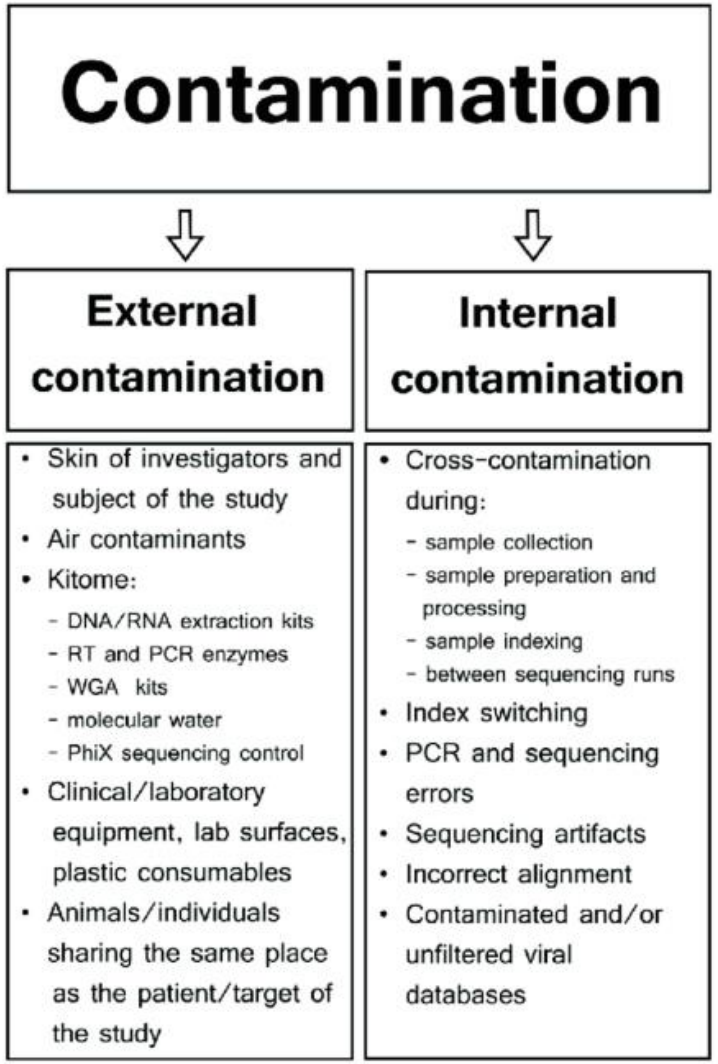
Contamination

Definition

Contamination corresponds to the presence of DNA that does not come from the sample studied.

- Recognizing contamination, followed by appropriate decontamination, should be a critical first step for all microbiome analyses.
- Skipping this step can easily result in confounded results and false conclusions.

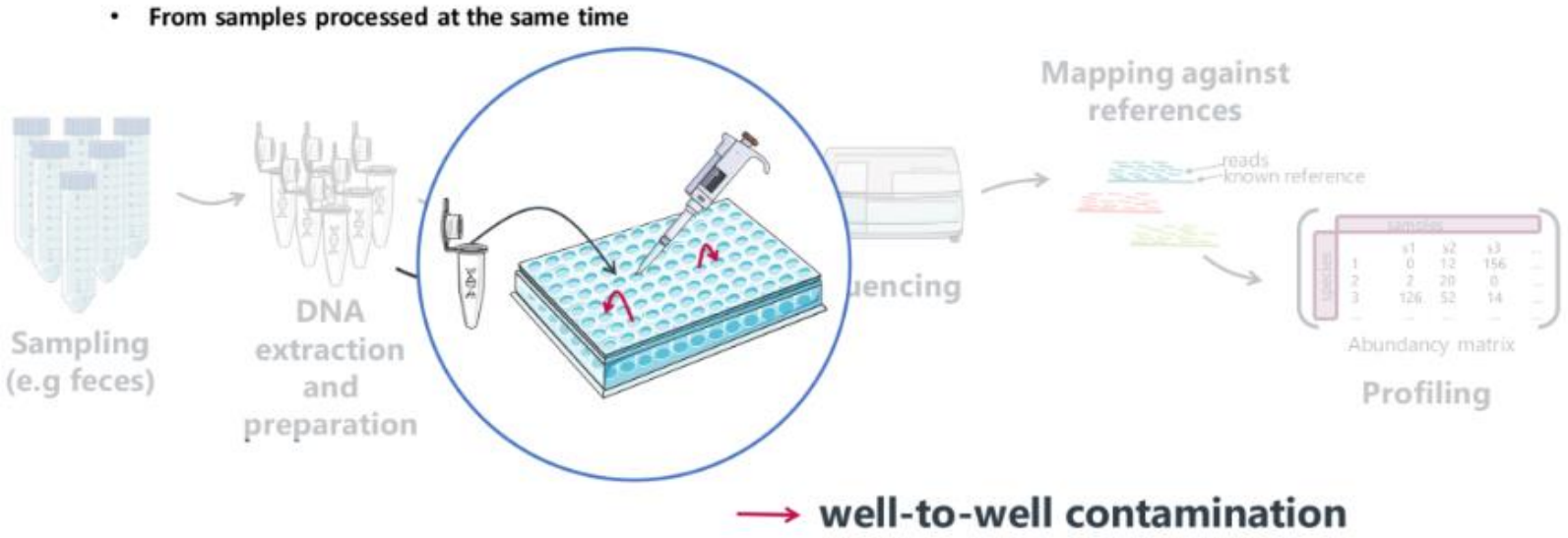
Contamination origins



External contaminants

- negative controls (i.e., blank reagent controls) during sample collection, preparation, and/or sequencing
- in silico detection (taxonomic affiliation, mapping…)

Well-to-well contamination



- Use blanks as negative controls [8]
- Study the sharing of strains between samples [9]
- Compare abundance profiles (CroCoDeEL, in dev.)

Cleaning tools

Reads cleaning

tool	migale	comments
fastp [10]	✔	all in one
pear [11]	✔	for merging reads
sickle [12]	✔	adaptative trimming

Decontamination tools

tool	migale	comments
kneaddata [13]	✔	remove rRNA reads
sortmerna [14]	✔	remove rRNA reads, slow…
HoCoRT [15]	✔	choice between several aligners, easy to use

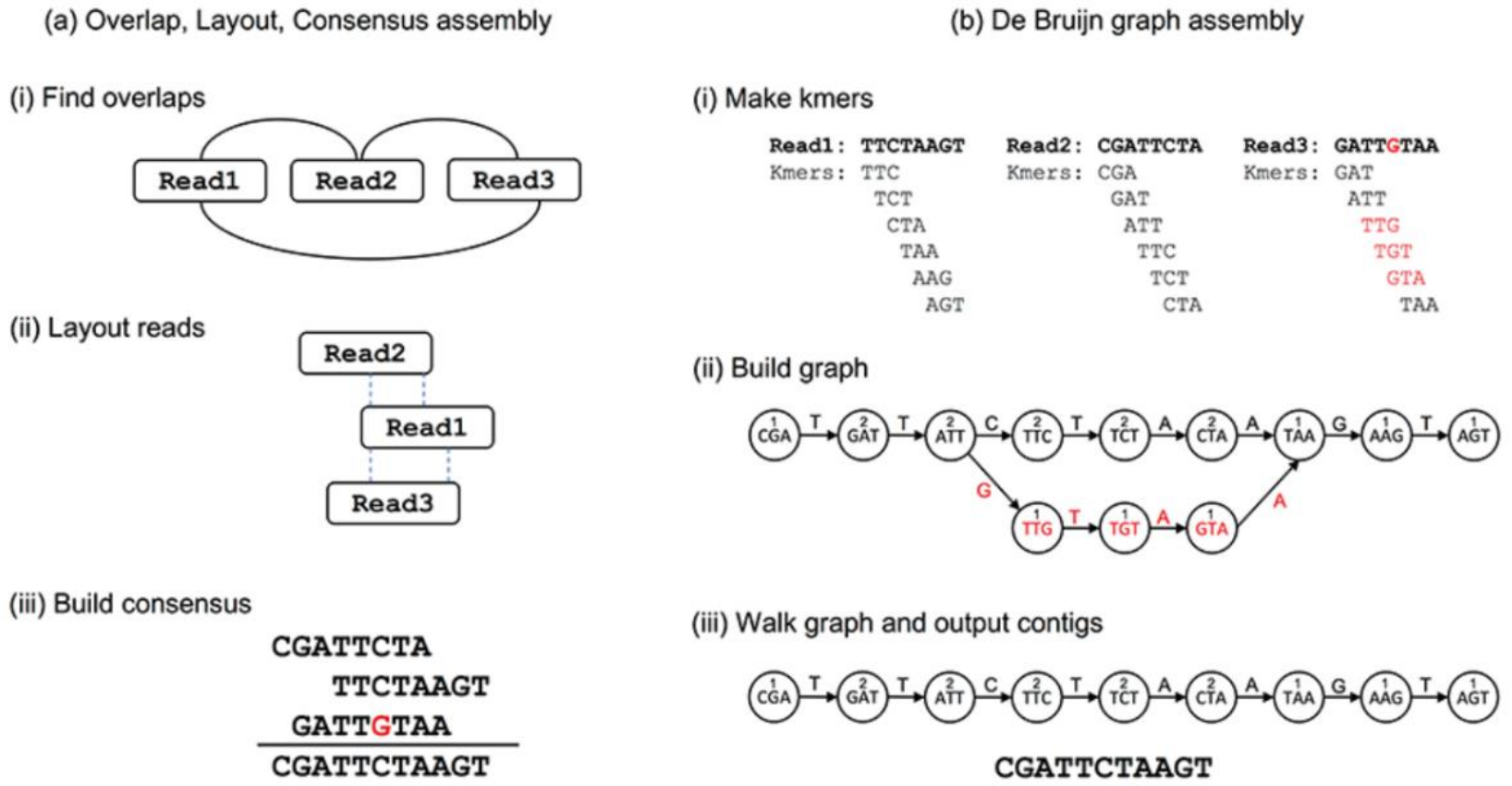
Metagenomics assembly

The road to metagenomics: from microbiology to DNA sequencing technologies and bioinformatics (2015) [1]

Objectives

- Reconstruct genes and organisms from complex mixtures
- Dealing with the ecosystem's heterogeneity, multiple genomes at varying levels of abundance
- Limiting the reconstruction of chimeras

Assembly strategies



Co-assembly?

- Usefull in some cases:
  - differences in coverage between samples

Pros of co-assembly	Cons of co-assembly
More data	Higher computational overhead
Better/longer assemblies	Risk of shattering the assembly
Access to lower abundant organisms	Risk of increased contamination

Co-assembly

In these cases, co-assembly is reasonable if:

- Same samples
- Same sampling event
- Longitudinal sampling of the same site
- Related samples

If it is not the case, individual assembly should be prefered. In this case, an extra step of de-replication should be used

Tools

- Generic tool with a meta option : SPAdes and metaSPAdes [16]
- Tools requiring less memory : MEGAHIT [17]
- The historical tool allowing many parameters : Velvet (and MetaVelvet)
- A (not so) recent benchmark of short reads metagenome assemblers. [18]
- Long read / Hybrid assemblies use different algorithms and strategies and are still a research question.

Assessment of assembly quality

After assembly, we use MetaQUAST [19] to evaluate and compare metagenome assemblies.

What MetaQUAST does :

- De novo metagenomic assembly evaluation
- [Optionally] identify reference genomes from the content of the assembly
- Reference-based evaluation
- Filtering so-called misassemblies based on read mapping
- Report and visulaisation

De novo metrics

Evaluation of the assembly based on:

- Number of contigs greater than a given threshold (0, 1kb, …)
- Total / thresholded assembly size
- Largest contig size
- N50 : the sequence length of the shortest contig at 50% of the total assembly length, equivalent to a median of contig lengths. (N75 idem, for 75%)
- L50 : the number of contigs at 50% of the total assembly length. (L75 idem, for 75%)

Reference-based metrics

- Metrics based on the comparison with reference genomes.
- Reference genomes are given by the user or automatically constitued by MetaQuast based on comparison of rRNA genes content of the assembly and a reference database (Silva).
- Complete genomes are then automatically downloaded.

Reference-based metrics

For each given reference genome, based on an alignement of all contigs on it :

- Duplication rate
- Percent genome complete
- NGA50 : equivalent of N50 but with the aligned block of the contigs
- “Misassemblies” : breakpoint of alignement in a contigs.
- An individual Quast report and an alignement visualisation.

Counts on assembly

- We used only a portion of the reads for assembly.
- For further analyses it is necessary to map all the reads on the contigs to obtain counts per features (genes/contigs…)
- We will use BWA [20]
  - Firstly, we build an index.
  - Secondly, reads are aligned.
  - We can use samtools [21] and bedtools [22] to manipulate SAM/BAM files.

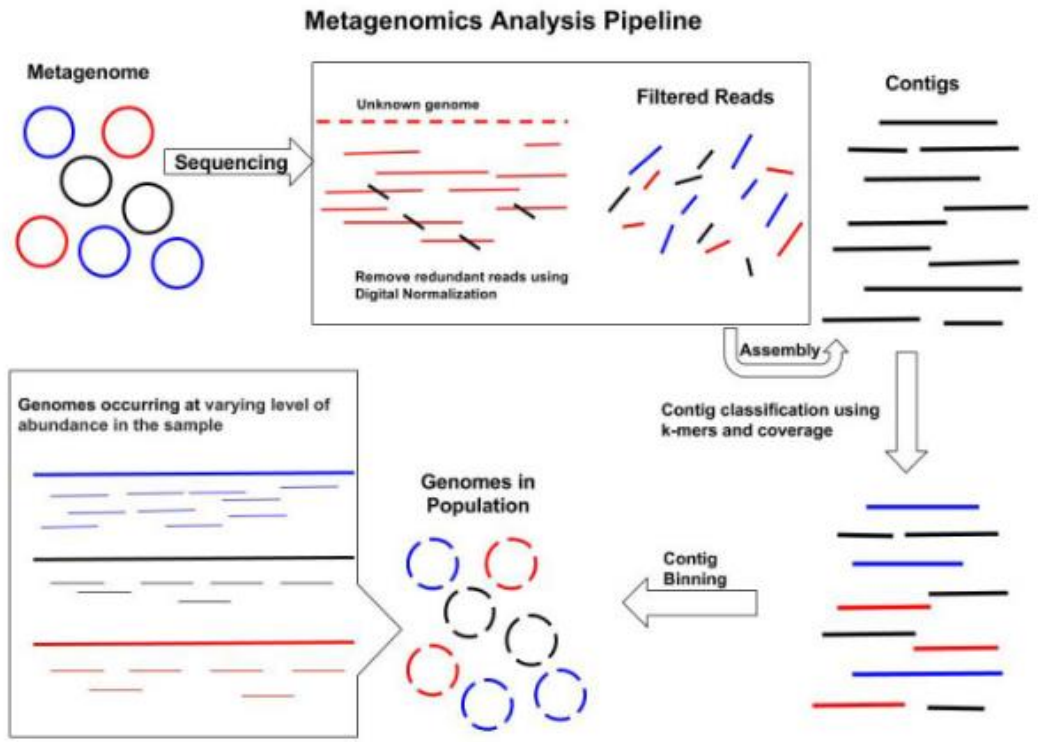
Binning

The road to metagenomics: from microbiology to DNA sequencing technologies and bioinformatics (2015) [1]

Objectives

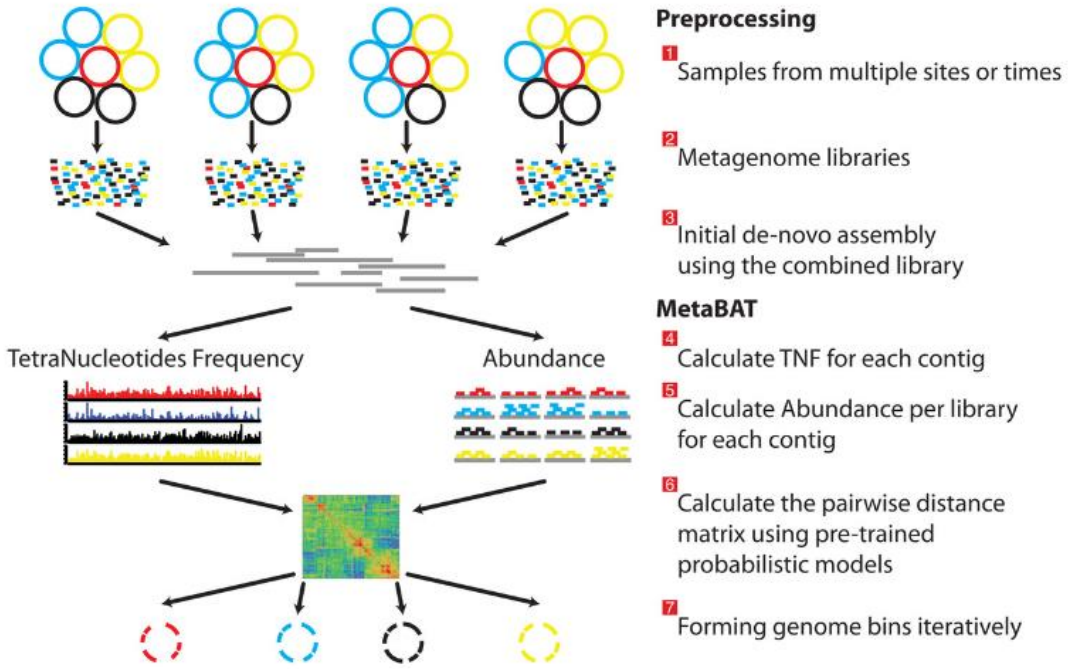
Binning is a good compromise when the assembly of whole genomes is not feasible.

Similar contigs are grouped together.



Approach

- MetaBAT [23] is a tool for reconstructing genomes from complex microbial communities.



Bins evaluation

For the evaluation of bins, we will use *completeness* and *contamination* estimated by CheckM [24]

- Use of collocated sets of genes that are ubiquitous and single-copy within a phylogenetic lineage.
- Among a set of tools in CheckM we will use the `checkm2 predict` workflow which only mandatory requires a directory of genome bins.

Annotation

The road to metagenomics: from microbiology to DNA sequencing technologies and bioinformatics (2015) [1]

Objective

- Syntaxic annotation (gene prediction)
- Functionnal annotation (function prediction for protein coding genes)
- Prokka [25] is a is a software tool to annotate bacterial, archaeal and viral genomes (*quite*) **quickly** and produce standards-compliant output files.
- Prokka *automatically* annotate a complete bacterial genome in ~15mn.
- Prokka will not replace expert annotation but gives you an homogenoeus procedure for annotation of conserved genes familly

Genes prediction (1)

- Gene prediction in complete prokaryotic genomes isn't as such a problem.
  - Most errors / difference in **start** position
- Efficient gene predictors are available (bactgeneSHOW, prodigal, glimmerHMM,…).
- Most of them uses HMM (*Hidden Markov Models*) models to predict the gene structure

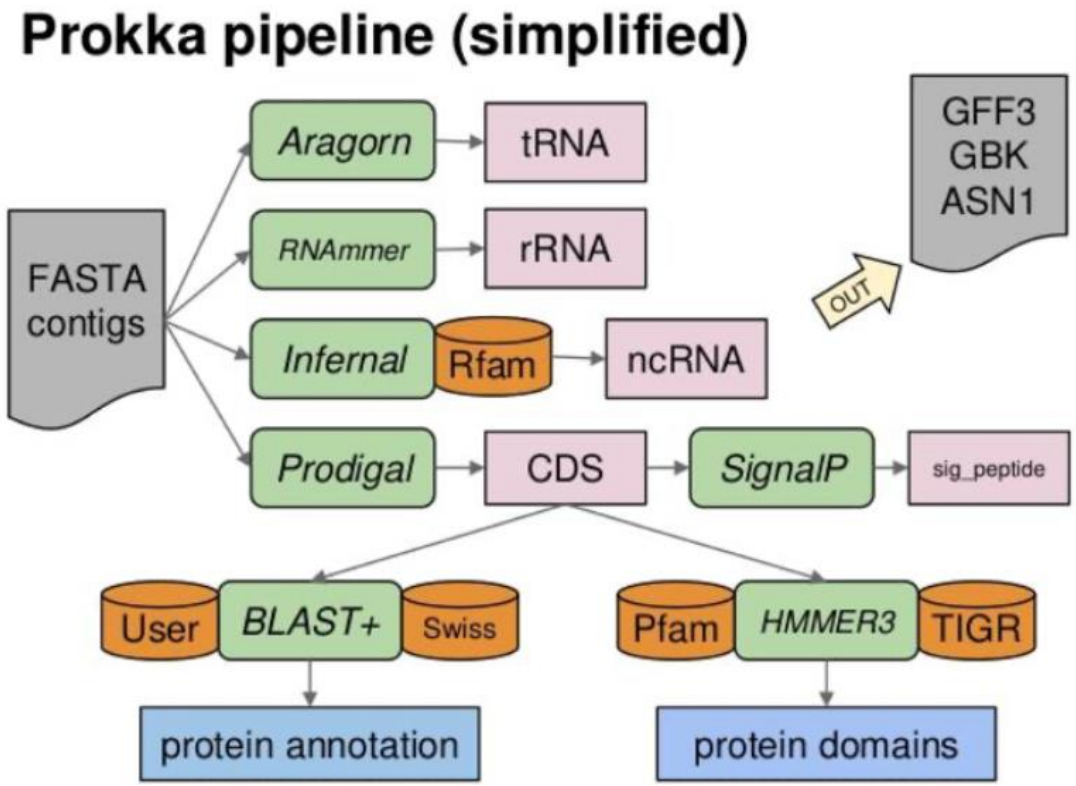
Genes prediction (2)

- Gene prediction on metagenomes is difficult due to :
  - assembly fragmentation
  - assembly errors, frameshift, chimeras,…
  - different species in the same sample that could/should lead to use different models
  - a mix of viral, bacterial and eukaryotic sequences.
- Prodigal (with the `-meta` parameter) and fraggenescan have good enough results on metagenomic contigs.
- Caution to partial genes in the following analysis !

Prokka pipeline

- Coding gene prediction with Prodigal [26]
- tRna; rRna gene prediction with Aragorn, Barnap, RNAmmer (optionnal)
- Functionnal annotation based on similarity search with a threshold (1e-6) and hierarchically against :
  - [Optionnal] a given proteome (`--proteins` parameter)
  - ISFinder for transposases, not entire IS
  - NCBI Bacterial Antimicrobial Resistance Reference Gene Database for Antimicrobial Resistance Genes.
  - UniprotKB/Swissprot **curated** proteins with evidence that (i) from Bacteria (or Archaea or Viruses); (ii) not be “Fragment” entries; and (iii) have an evidence level (“PE”) of 2 or lower, which corresponds to experimental mRNA or proteomics evidence.
  - Domain and motifs (hmmsearch) :
    - Pfam && HAMAP

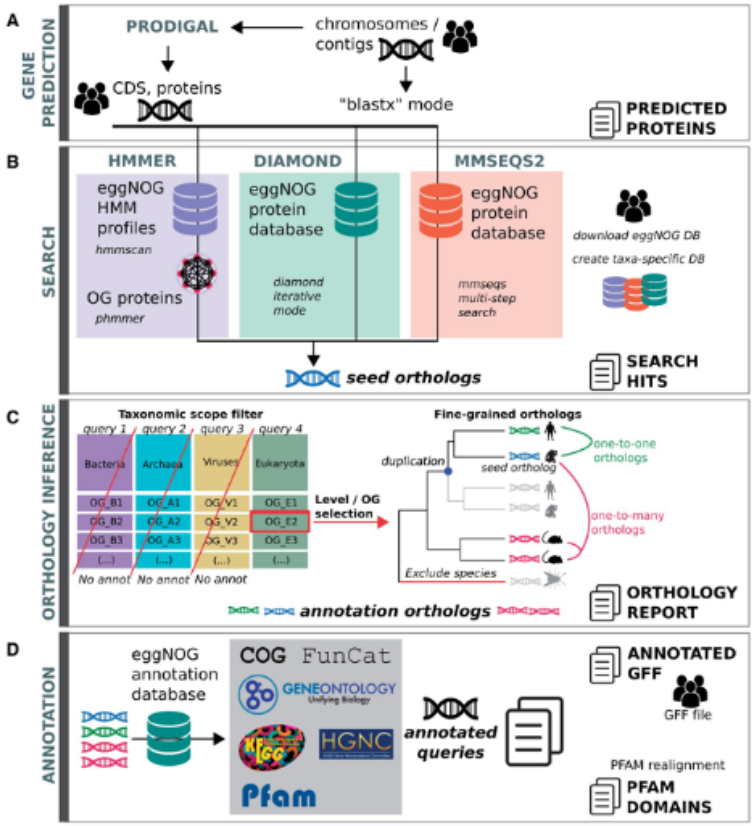
Prokka pipeline



EggNogg Mapper

- eggNOG-mapper [27] is a tool for fast functional annotation of novel sequences. It uses precomputed orthologous groups and phylogenies from the eggNOG database to transfer functional information from fine-grained orthologs only.
- *eggNog* database :
  - 5090 organisms, 2502 viruses.
  - 4.4 M orthologous groups annotated with COG category, Gene Ontolgy, EC number, Kegg orthologs and pathways, CAZy, PFAMs

EggNogg Mapper workflow



- eggNOG uses hmmsearch to search against HMM eggNOG database OR Diamond / MMSeqs2 to search against eggNOG protein database
- Refine first hit using a list of precomputed orthologs.
- Assign one ortholog
- Functionnal annotation transfer using this ortholog

Other options for functionnal annotation

- **Diamond** [28] is a sequence aligner for protein (equivalent blastp) and translated DNA (equivalent tblastx) searches, designed for high performance analysis of big sequence data. Diamond is 100x to 20,000x the speed of BLAST.
  - Diamond could be used to query against any given databank.
- **ghostKOALA** [29, Online]
- **KOALA** (KEGG Orthology And Links Annotation) is KEGG's internal annotation tool for K number assignment.

Other options for functionnal annotation (2)

- **cd-hit** [30] is a software for clustering protein sequences. It can be used to downsize the number of lines in the in the gene count matrix.

Biostatistics

The road to metagenomics: from microbiology to DNA sequencing technologies and bioinformatics (2015) [1]

What to do after bioinformatics

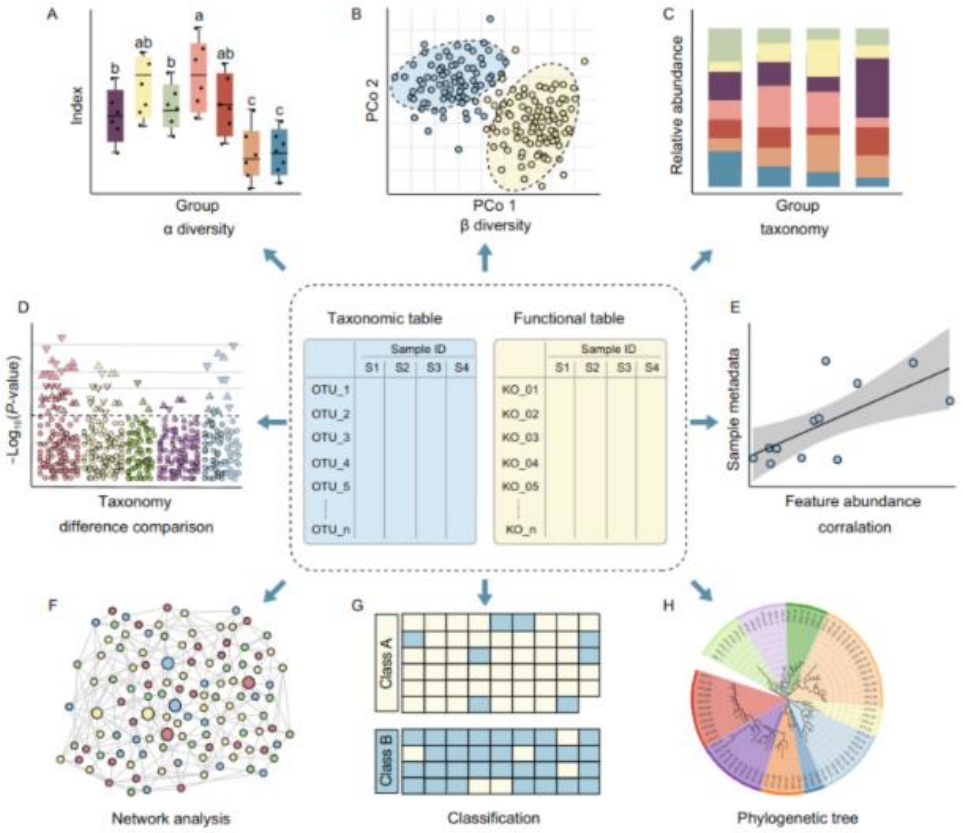


Figure 3. Overview of statistical and visualization methods for feature tables. Downstream analysis of microbiome feature tables, including alpha/beta-diversity (A/B), taxonomic composition (C), difference comparison (D), correlation analysis (E), network analysis (F), classification of machine learning (G), and phylogenetic tree (H). Please see Table 2 for more details.

Automatization

The road to metagenomics: from microbiology to DNA sequencing technologies and bioinformatics (2015) [1]

Snakemake workflow

We have developed a workflow that allows us to automate all these analyses.

- developed with snakemake
- executable on MIGALE

```
{
    "SAMPLES": ["mock"],
    "NORMALIZATION": true,
    "SORTMERNA": true,
    "ASSEMBLER": "metaspades",
    "CONTIGS_LEN": 1000,
    "PROTEINS-PREDICTOR": "prodigal"
}
```

- GitLab

Other tools

- `Pear` : merge paired-end
- `SimkaMin` [32] : fast and resource frugal de novo comparative metagenomics
- `Linclus` (MMseqs2) [33] : Clustering huge protein sequence sets in linear time.
- `PLASS`[34] : Protein-Level ASSembler
- …

Anvi'o: integrated multi-omics at scale

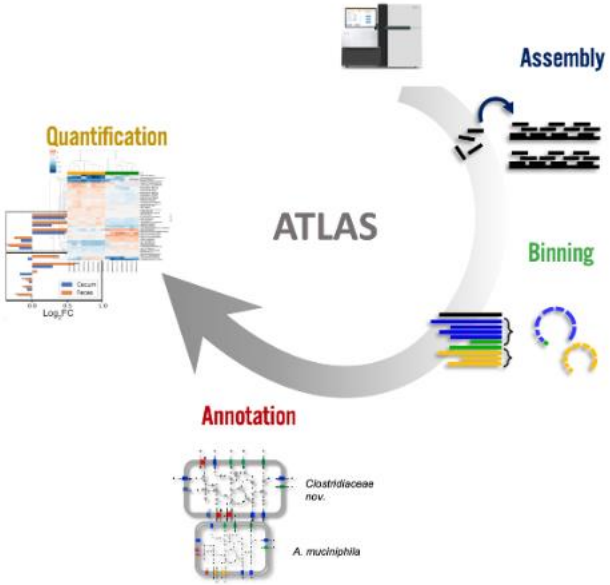
- Anvi'o [36] is an open-source, community-driven analysis and visualization platform for microbial 'omics.
- With this tutorial, starting from a metagenomic assembly, you will :
  - Process your contigs,
  - Profile your metagenomic samples and merge them,
  - Visualize your data, identify and/or refine genome bins interactively, and create summaries of your results.

Metagenome atlas

- Metagenome-atlas [37] is an *easy-to-use* metagenomic pipeline based on `snakemake`.
- It handles all steps of analysis :

```
mamba install -y -c bioconda -c conda-forge metagenome-atlas
atlas init --db-dir databases path/to/fastq/files
atlas run all
```

Metagenome atlas



1 Quality Control
- PCR duplicates removal
- Quality trimming
- Host removal
- Common contaminant removal

➢ **QC reads**

2 Assembly
- Error correction
- Paired-end merging
- Assembly (metaSpades/megahit)
- Post-filtering

➢ **High-quality Scaffolds**

3 Genomic Binning
- Binning (metabat, maxbin2)
- Quality Assessment (checkM)
- Bin refining (DAS Tool)
- Dereplication (dRep)
- Quantification
- Robust taxonomic classification (CAT)

➢ **Genomes**
➢ **Abundances**

4 Annotation
- Gene prediction (prodigal)
- Cluster redundant genes (linclust/ cd-hit)
- Annotation (eggNOG)

➢ **Comparable gene catalog**

Automatization - Galaxy

The road to metagenomics: from microbiology to DNA sequencing technologies and bioinformatics (2015) [1]

What is galaxy

Galaxy

Data Intensive *analysis* for everyone

- Versatile and reproducible workflows
- **Web** platform
- **Open source** under Academic Free License
- Developed at Penn State, Johns Hopkins, OHSU and Cleveland Clinic with substantial outside contributions

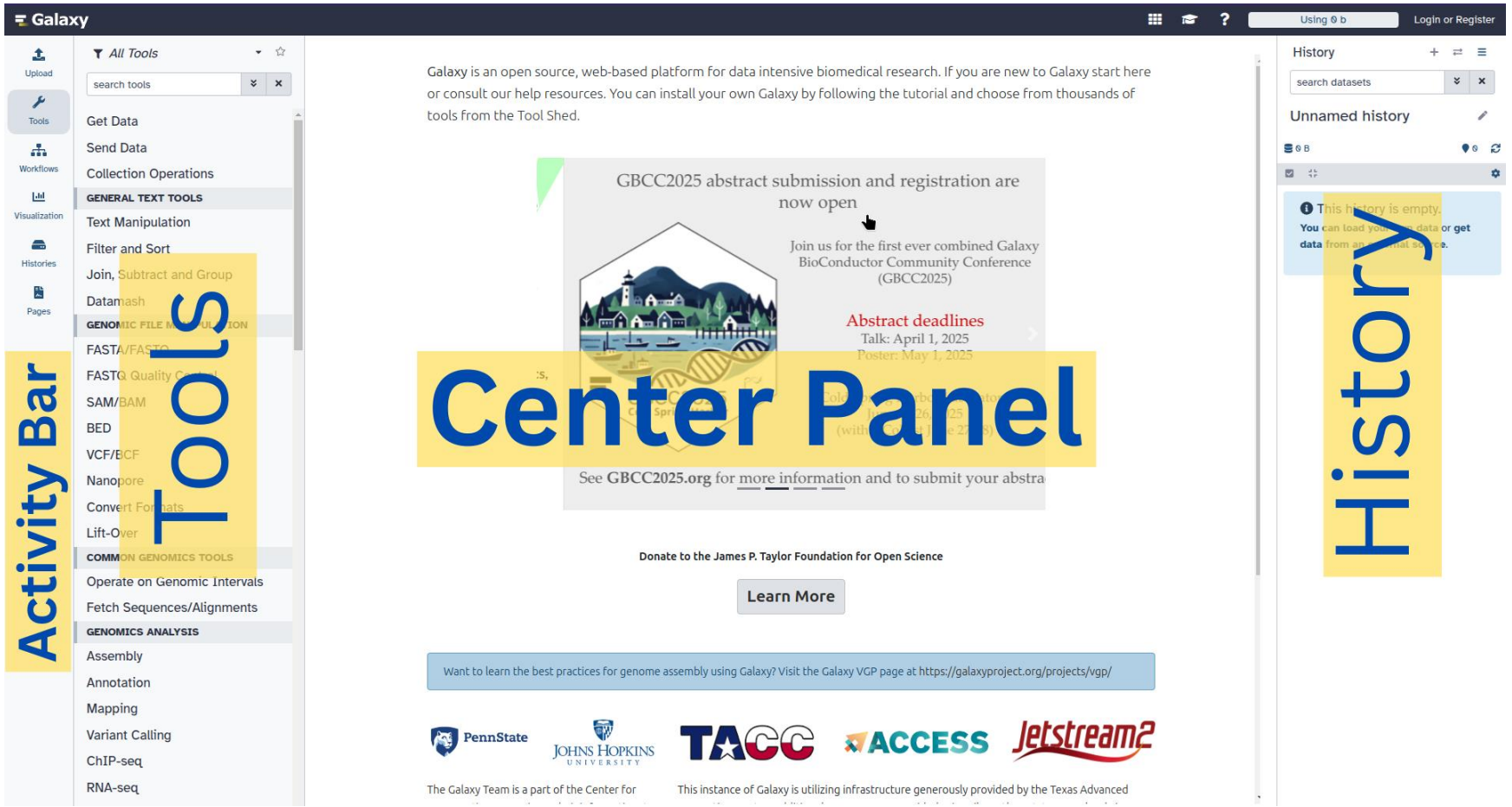
Core value

- **Accessibility**
  - Users without programming experience can easily upload/retrieve data, run complex tools and workflows, and visualize data
- **Reproducibility**
  - Galaxy captures information so that any user can understand and repeat a complete computational analysis
- **Transparency**
  - Users can share or publish their analyses (histories, workflows, visualizations)
  - Pages: online Methods for your paper

Galaxy growth

- **More than 10,300 ready to use tools for users**
- **More than 13,000 citations**
- **More than 170 public Galaxy resources**
  - **130+ public servers, many more non-public**
  - **Both general-purpose and domain-specific**

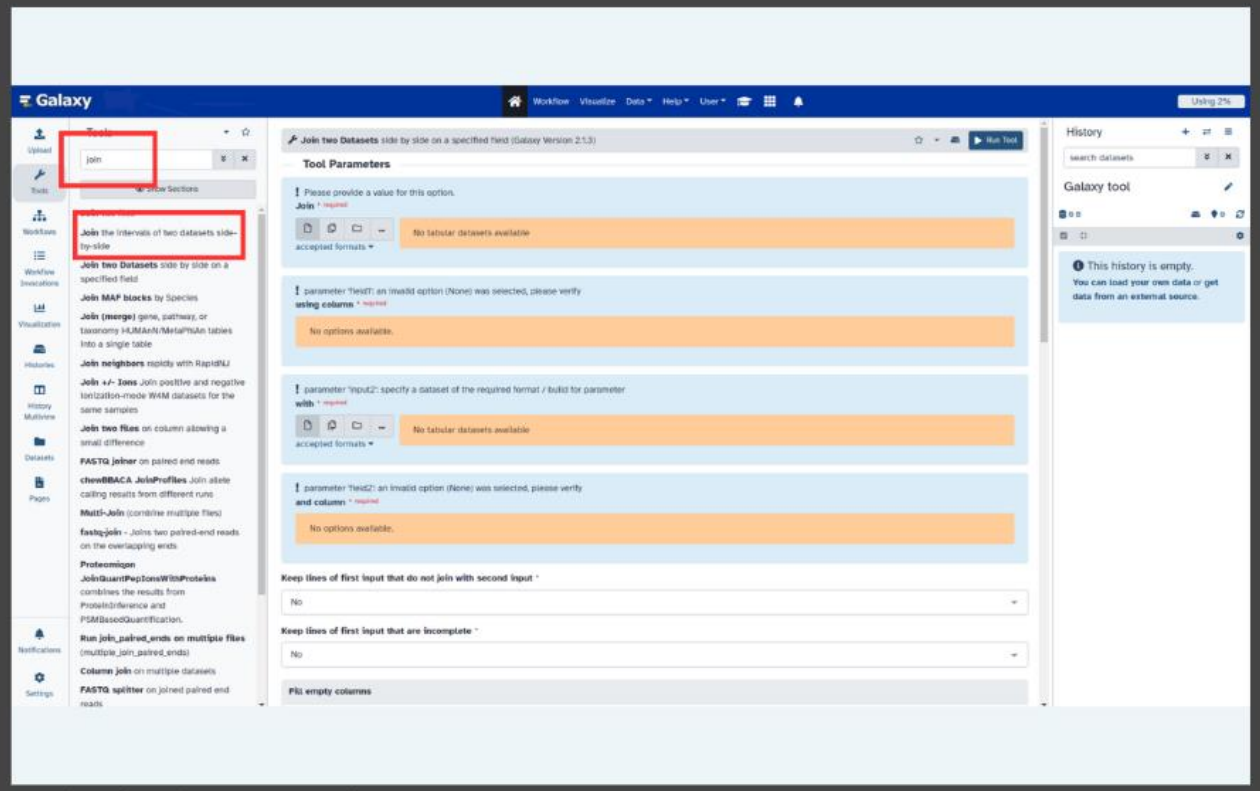
Main galaxy interface



Top menu

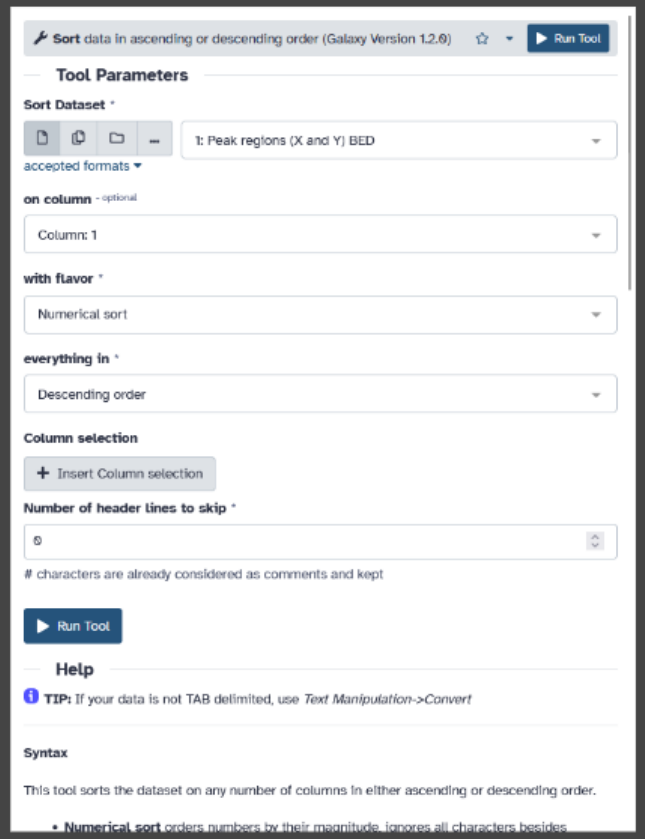
Link	Usage
(or *Analyze Data*)	go back to the homepage
Workflow	access existing workflows or create new one using the editable diagrammatic pipeline
Visualize	create new visualisations and launch Interactive Environments
Shared Data	access data libraries, histories, workflows, visualizations and pages shared with you
Help	links to Galaxy Help Forum (Q&A), Galaxy Community Hub (Wiki), and Interactive Tours
User	your preferences and saved histories, datasets, pages and visualizations

Tools



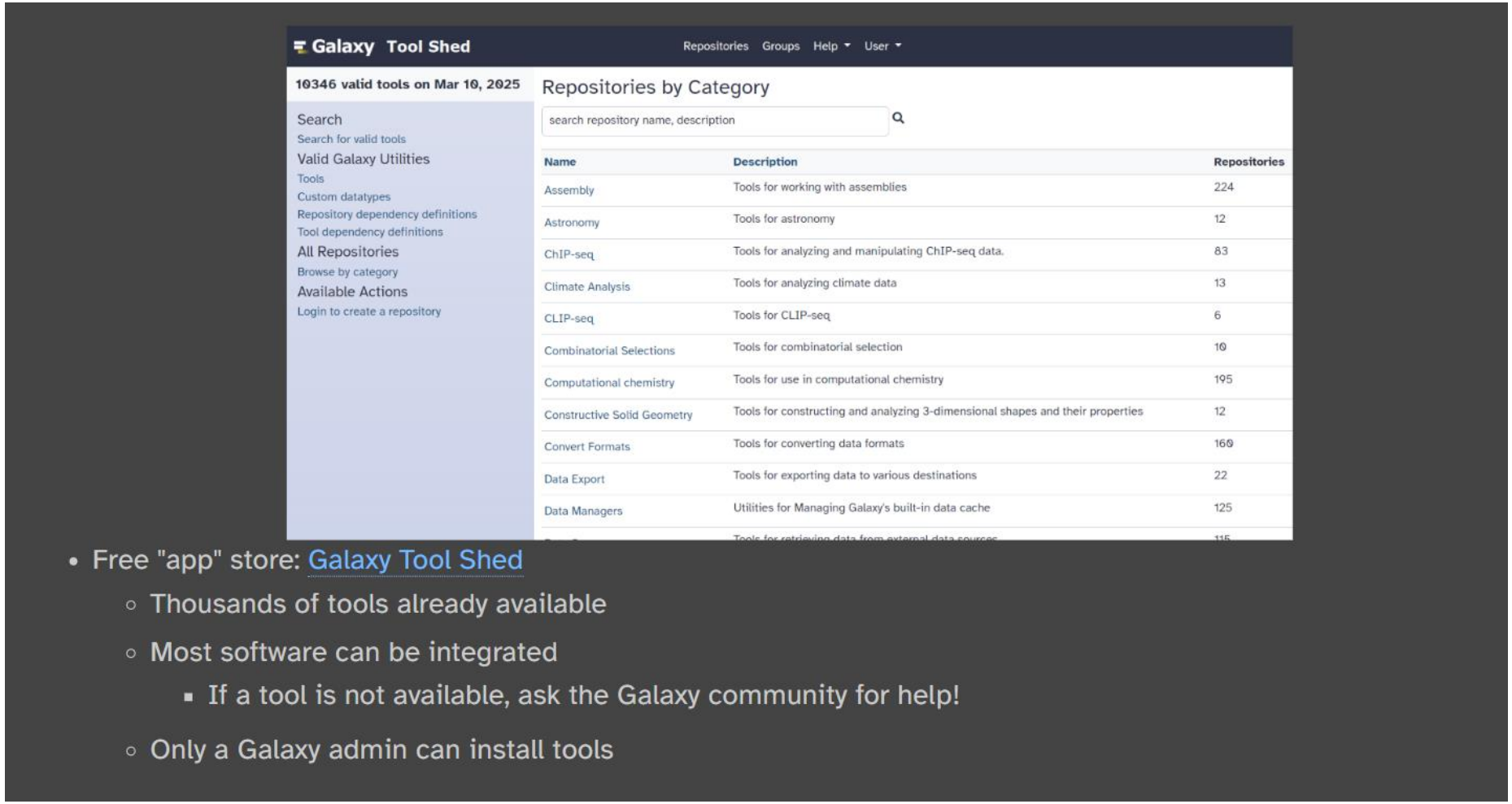
- The tool search helps in finding a tool in a crowded toolbox

Tool interface



- A tool form contains:
  - input datasets and parameters
  - help, citations, metadata
  - a `Run Tool` button to start a job, which will add some output datasets to the history
- New tool versions can be installed without removing old ones to ensure reproducibility

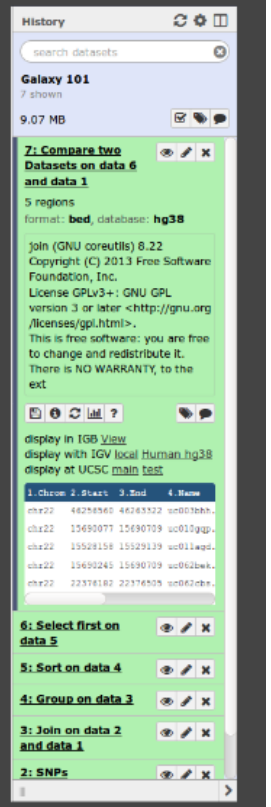
Tool Shed



- Free "app" store: Galaxy Tool Shed
  - Thousands of tools already available
  - Most software can be integrated
    - If a tool is not available, ask the Galaxy community for help!
  - Only a Galaxy admin can install tools

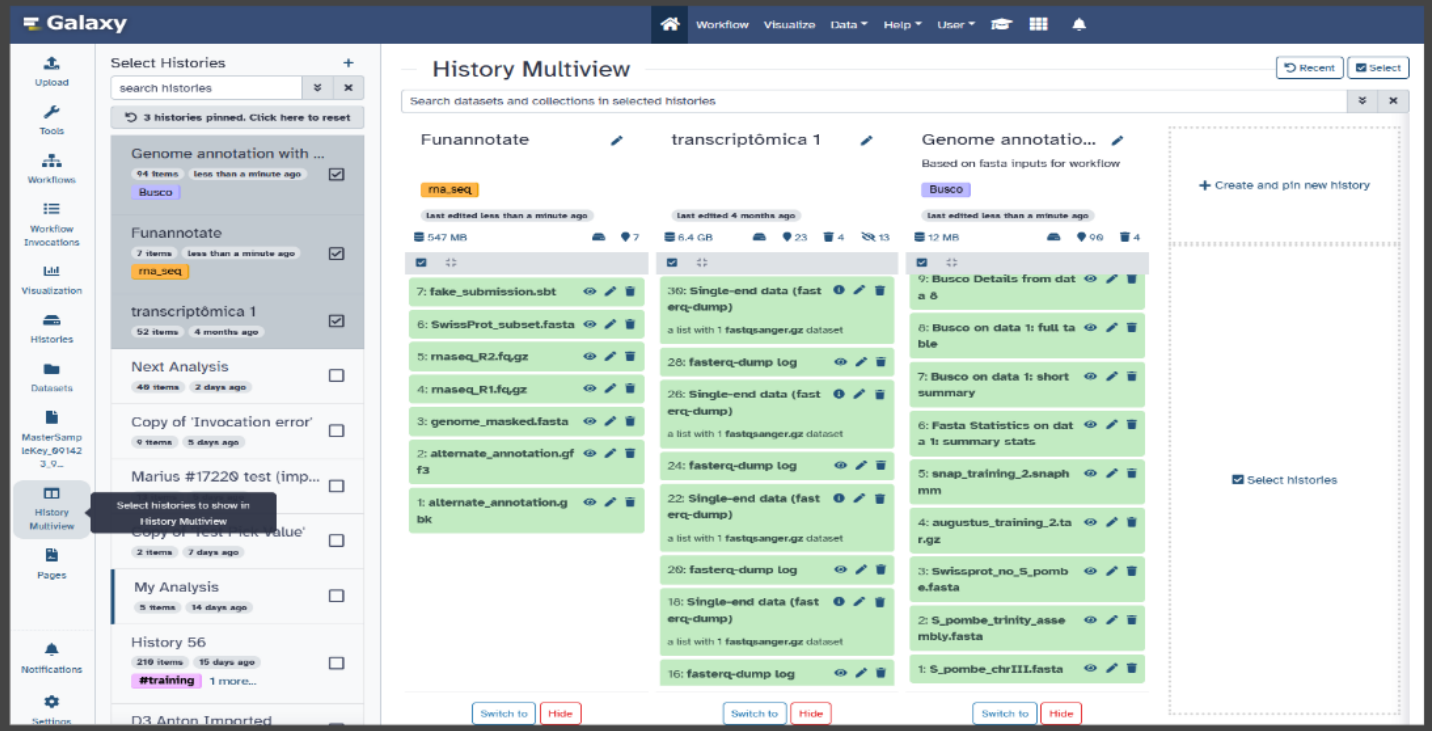
History

- Location of all analyses
  - collects all datasets produced by tools
  - collects all operations performed on the data
- For each dataset (the heart of Galaxy's reproducibility), the history tracks
  - name, format, size, creation time, datatype-specific metadata
  - tool id, version, inputs, parameters
  - standard output (`stdout`) and error (`stderr`)
  - state (waiting, running, success, failed)
  - hidden, deleted, purged



Multiple histories

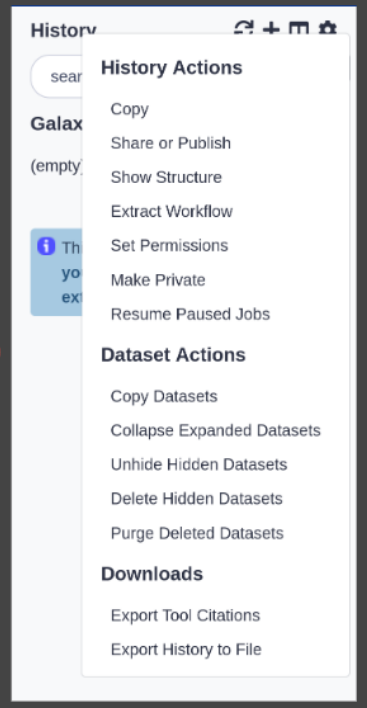
- You can have as many histories as you want
  - each history should correspond to a **different analysis**
  - and should have a meaningful **name**

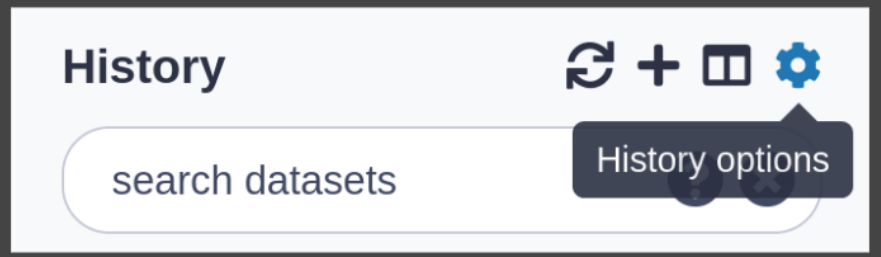


History options menu

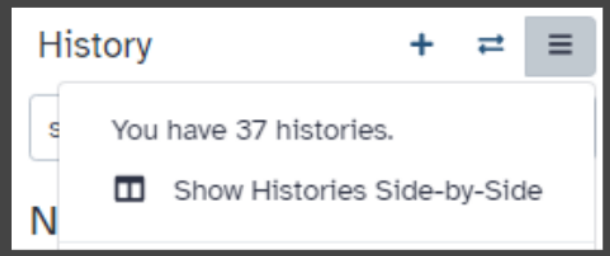
History behavior is controlled by the *History options*

(gear icon)





- *Create new history* (+ icon) will **not** make your current history disappear
- To see all of your histories, use the history switcher



- *Copy Datasets* from one history to another and save disk space for your quota

Importing data

- Copy/paste some text
- Upload files from your local computer
- Upload data from an internet URL
- Upload data from online databases: UCSC, BioMart, ENCODE, modENCODE, Flymine etc.
- Import from Shared Data (libraries, histories, pages)
- Upload data from FTP
- See Getting data into Galaxy

Datatypes

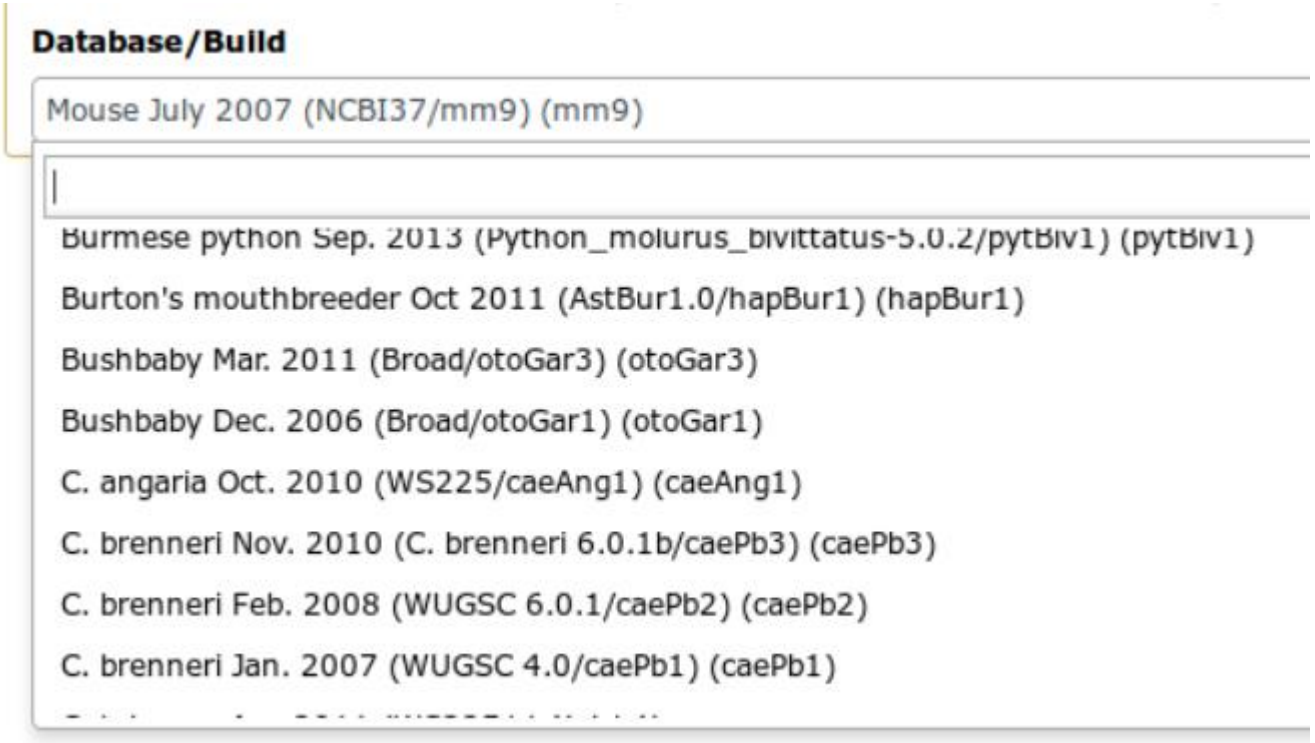
- Tools only accept input datasets with the appropriate datatypes
- When uploading a dataset, its datatype can be either:
  - automatically detected
  - assigned by the user
- Datasets produced by a tool have their datatype assigned by the tool
- To change the datatype of a dataset, either:
  - *Edit attributes* and *Datatypes* (if original wrong), or
  - *Edit attributes* and *Convert*

Reference datasets

- Example: reference Genome
- Genome build specifies which genome assembly a dataset is associated with
  - e.g. mm10, hg38...
- Can be assigned by a tool or by the user
- Users can create custom genome builds
- New builds can be added by the admin

Reference datasets

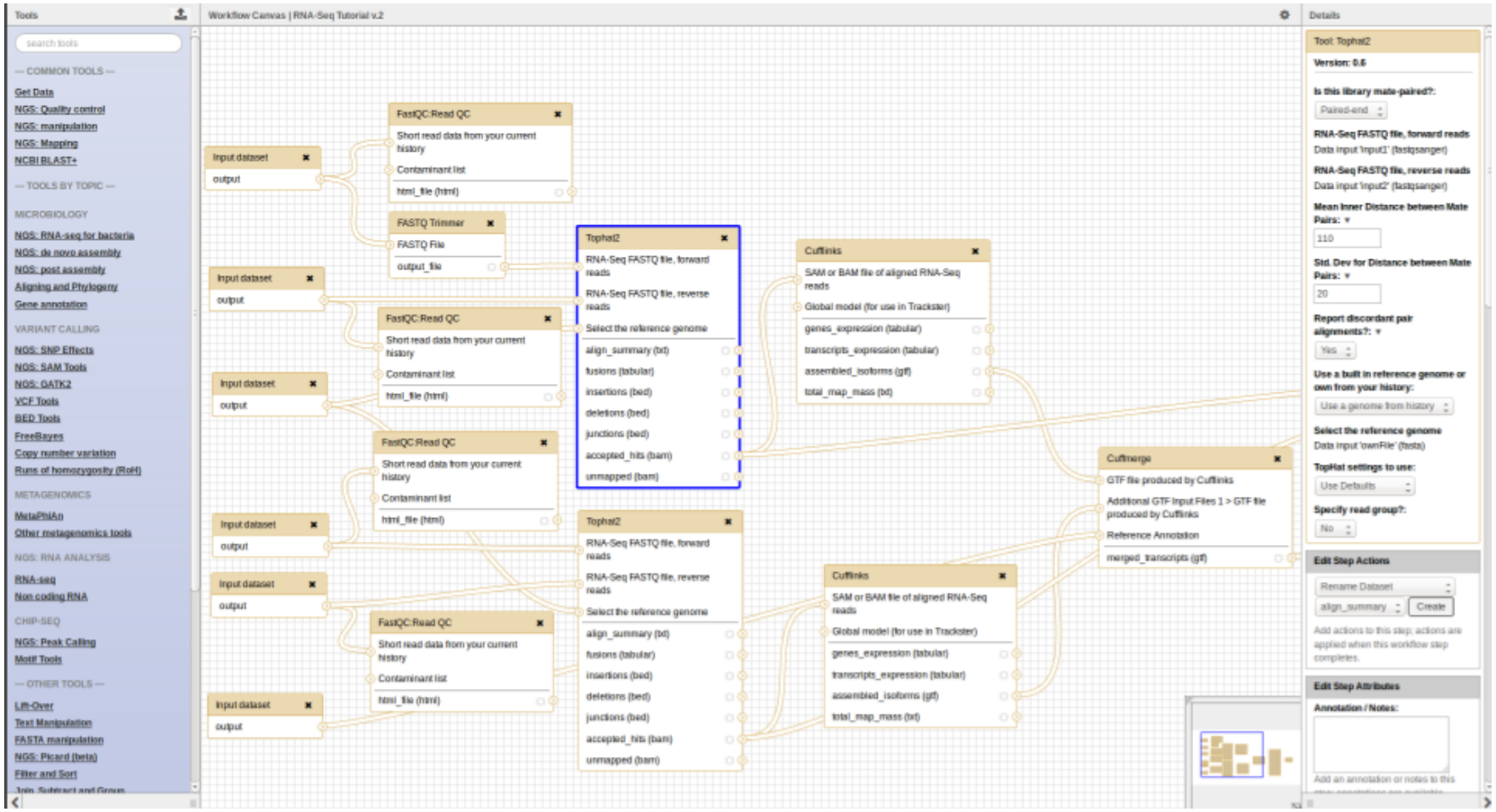
- Example: reference Genom
- Genome build specifies whi associated with
  - e.g. mm10, hg38...
- Can be assigned by a tool c
- Users can create custom ge
- New builds can be added b



Workflow Editor

- **Extracted** from a history
- **Built manually** by adding and configuring tools using the canvas
- **Imported** using an existing shared workflow

Workflow Editor



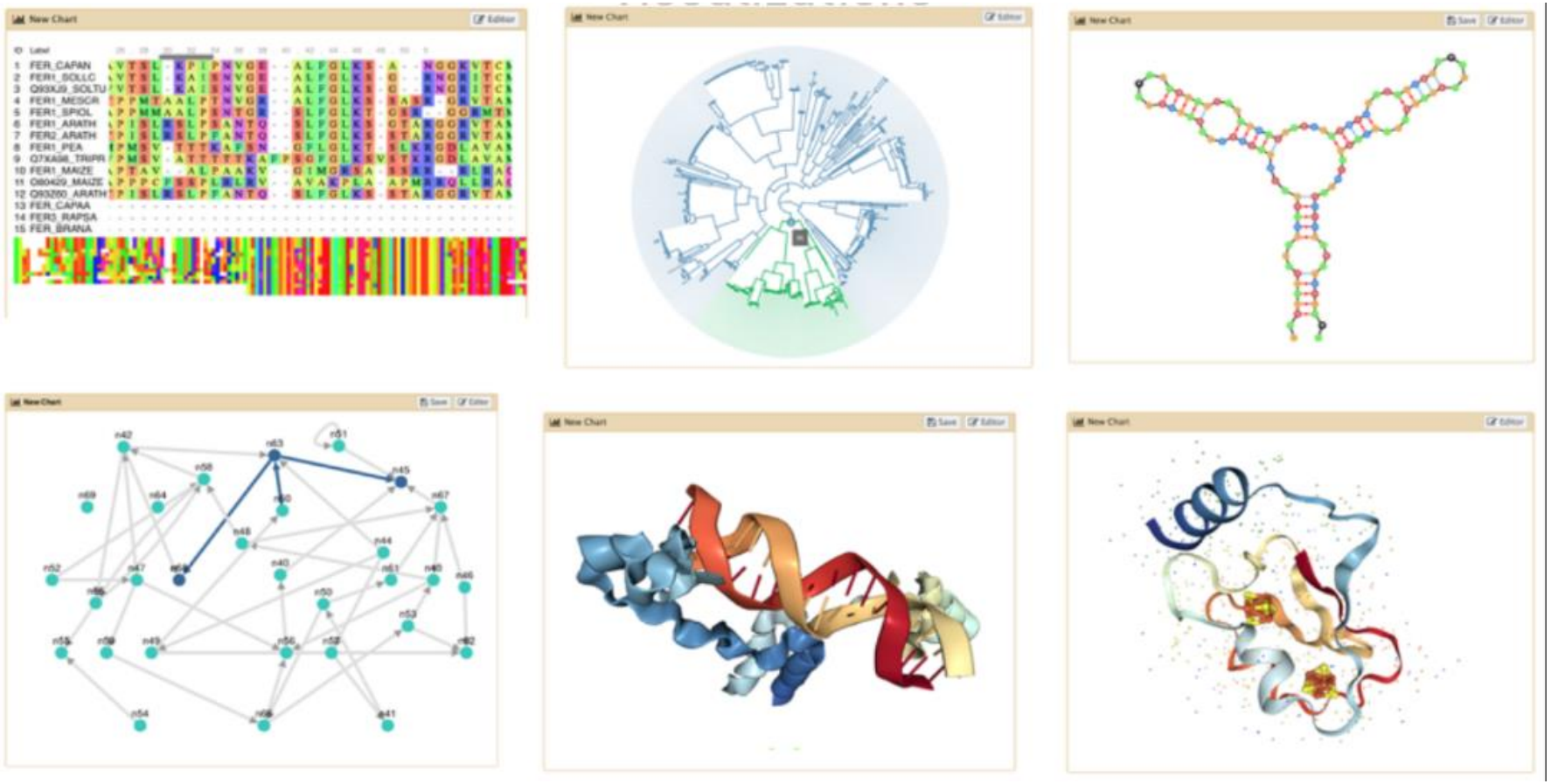
Why would you want to create workflows?

- **Re-run** the same analysis on different input data sets
- **Change parameters** before re-running a similar analysis
- Make use of the workflow job **scheduling**
  - jobs are submitted as soon as their inputs are ready
- Create **sub**-workflows: a workflow inside another workflow
- **Share** workflows for publication and with the community

Visualizations

- Datatypes know what tools can be used to visualize datasets:
  - Sequencing data has a button for visualizing in IGV
  - Tabular data will prompt you to build charts
  - Protein data can be seen in a 3D viewer
- Interactive environments: Jupyter, RStudio, etc

Visualizations



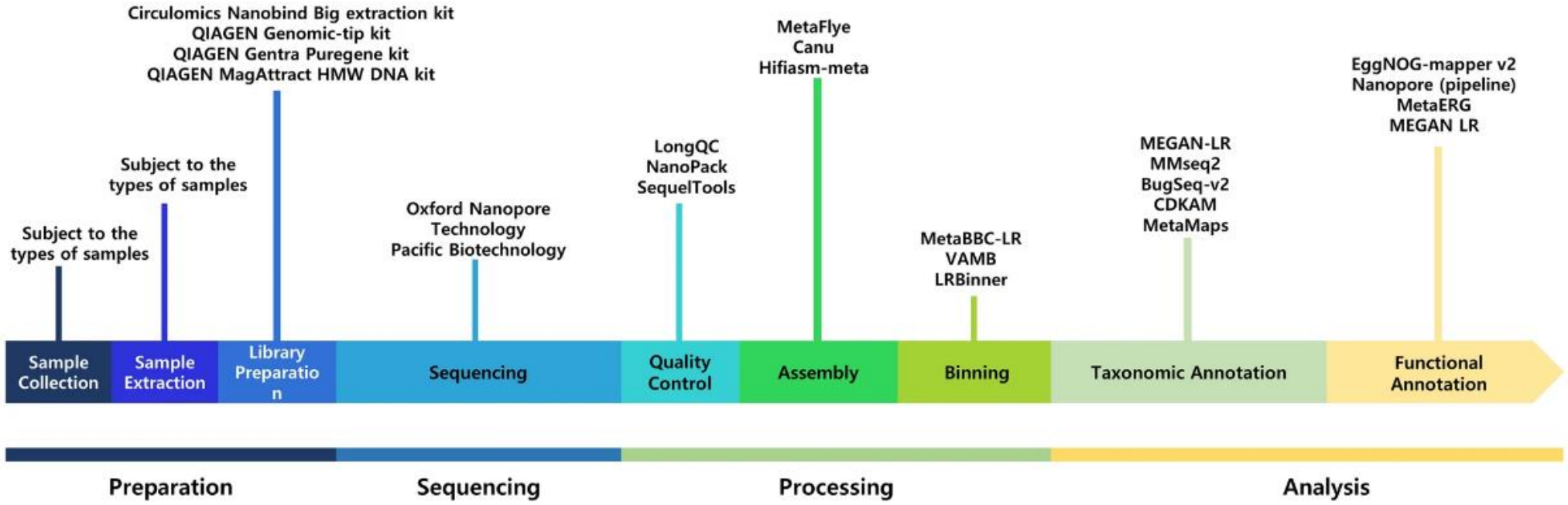
Sharing data

- Share everything you do in Galaxy - histories, workflows, and visualizations
  - Directly using a Galaxy account's email addresses on the same instance
  - Using a web link, with anyone who knows the link
  - Using a web link and publishing it to make it accessible to everyone from the *Shared Data* menu
- See Sharing your History in Galaxy

Appendix

The road to metagenomics: from microbiology to DNA sequencing technologies and bioinformatics (2015) [1]

Long reads metagenomics



Long reads metagenomics

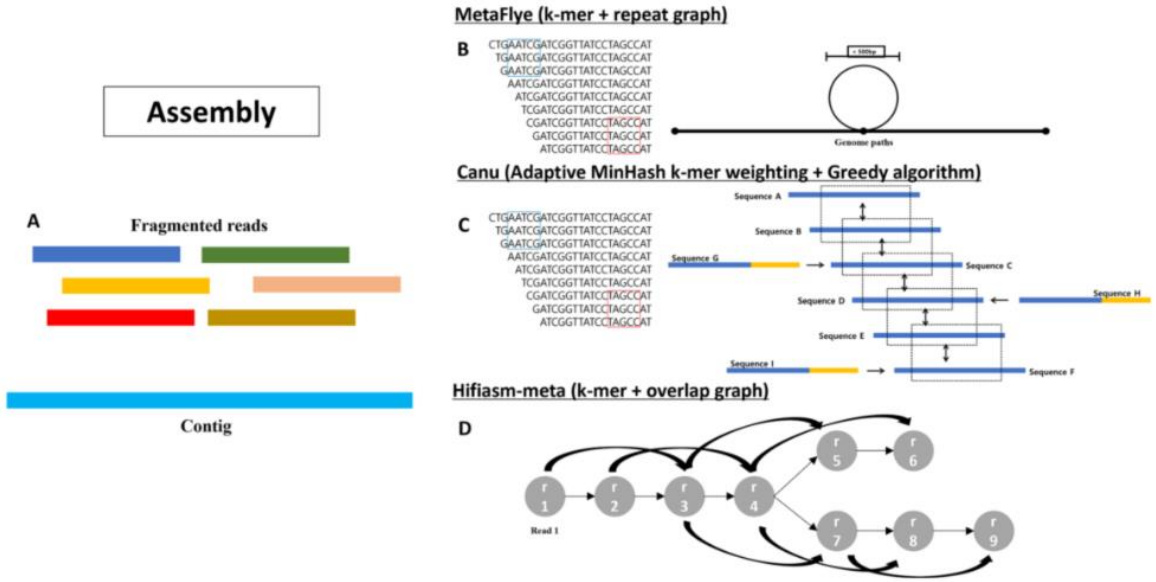


Fig. 3 Overview of assembly algorithm. **A** Fragmented reads are overlaid and merged to reconstruct microbial genomes, with the longer reads enhancing the connections between each fragment. **B** MetaFlye, a LRS metagenome assembler, uses k-mer and repeat detection algorithm, which is particularly useful when detecting repeats inside a bubble. **C** Canu utilizes a form of k-mer algorithm known as adaptive MinHash k-mer weighting, in combination with an altered version of the greedy best overlap graph (BOG) algorithm. The greedy BOG algorithm, initially developed by Miller and colleagues, serves as the foundation for constructing the graph in Canu. Sequences with mutual best overlap are indicated with arrows going both ways. However, sequences G, H, and I have one-sided arrows, meaning that the best overlap regions are not mutual and are not included as part of the best overlap. **D** Hifiasm-meta utilizes k-mers to query reads and construct an overlap graph while retaining reads with rare k-mers, which typically correspond to low abundance sequences

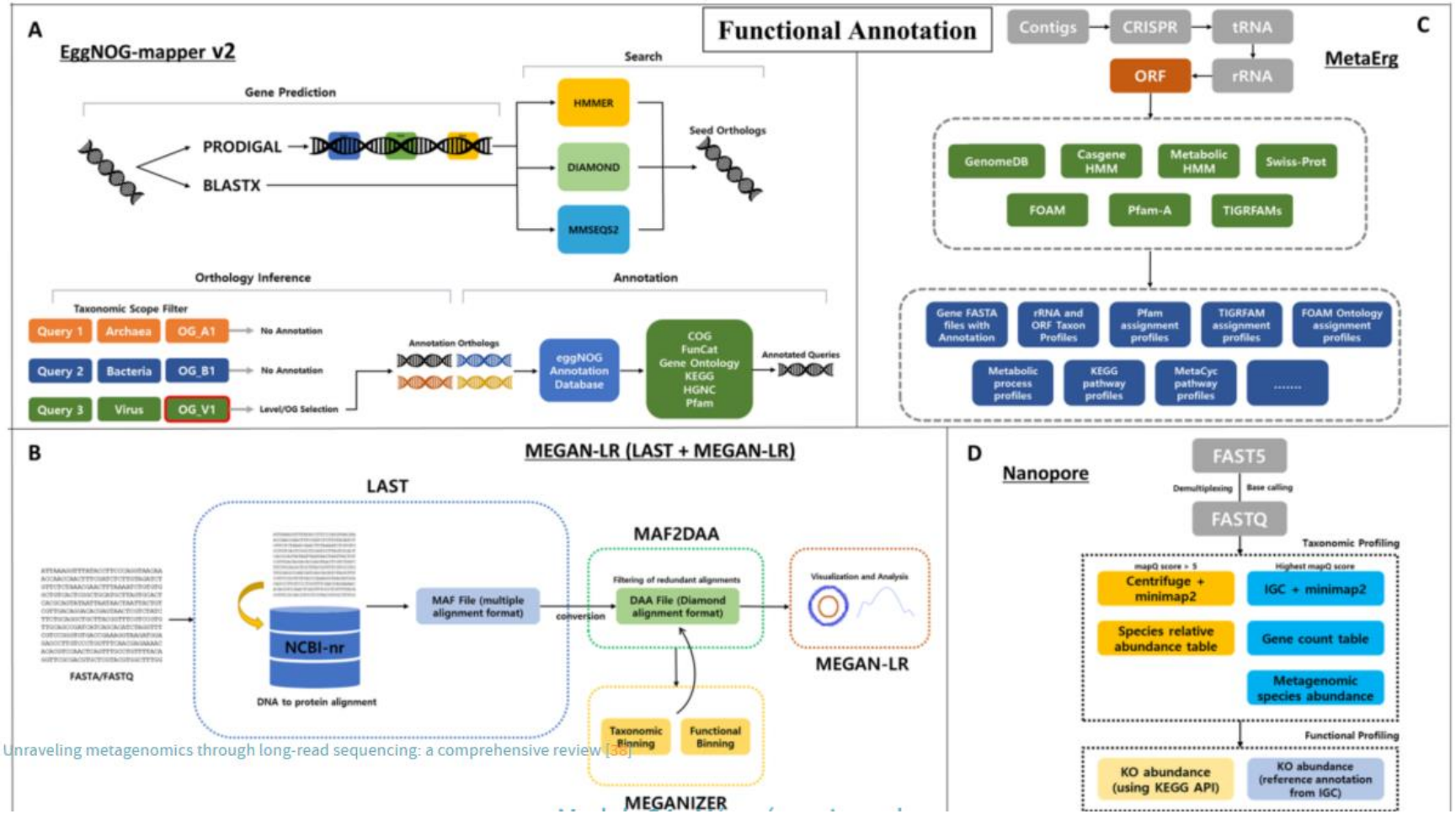
Long reads metagenomics

Taxonomic classifiers

Table 1 Taxonomic classifiers used for long reads

Name	Algorithm	Reference database	Developed year	LRS affinity
MEGAN-LR [65]	Translation + LCA algorithm	NCBI nt	2018	ONT
	Nucleotide + LCA algorithm	NCBI nt		ONT
MMseq2 [66]	Translation + LCA algorithm	NCBI nt	2017	PacBio
BugSeq-v2 [68]	Nucleotide + LCA algorithm	NCBI nt	2021	ONT
CDKAM [69]	Approximate matching + kmer	NCBI nt	2019	ONT
MetaMaps [70]	Approximate mapping + EM	Miniseq + H	2016	PacBio

Long reads metagenomics

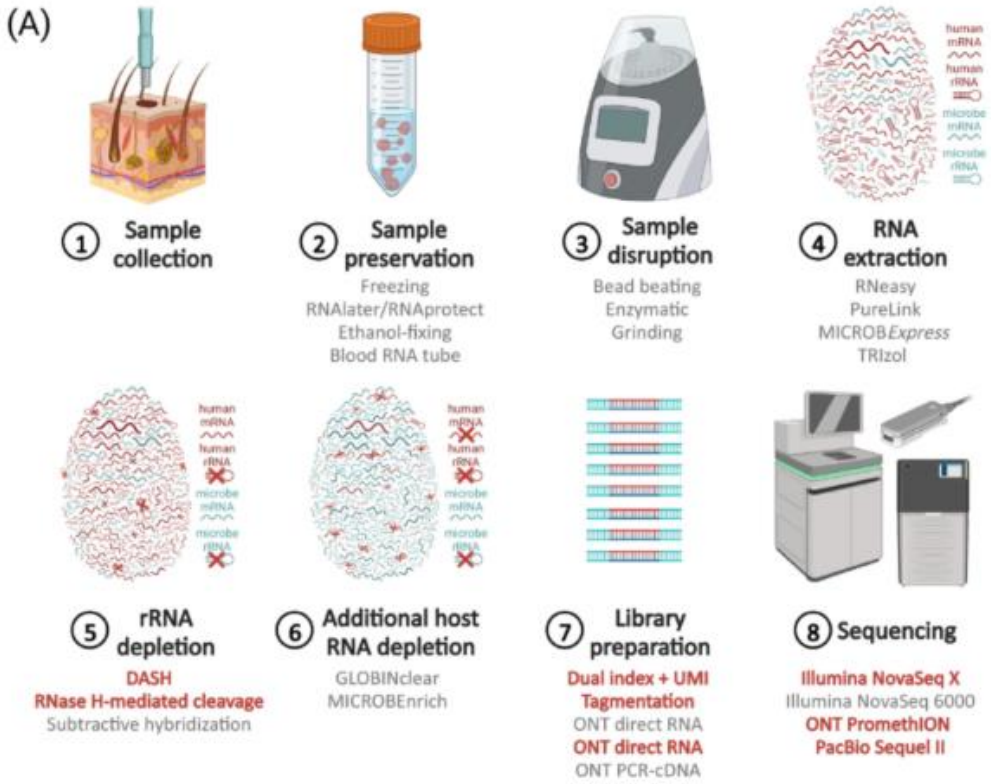


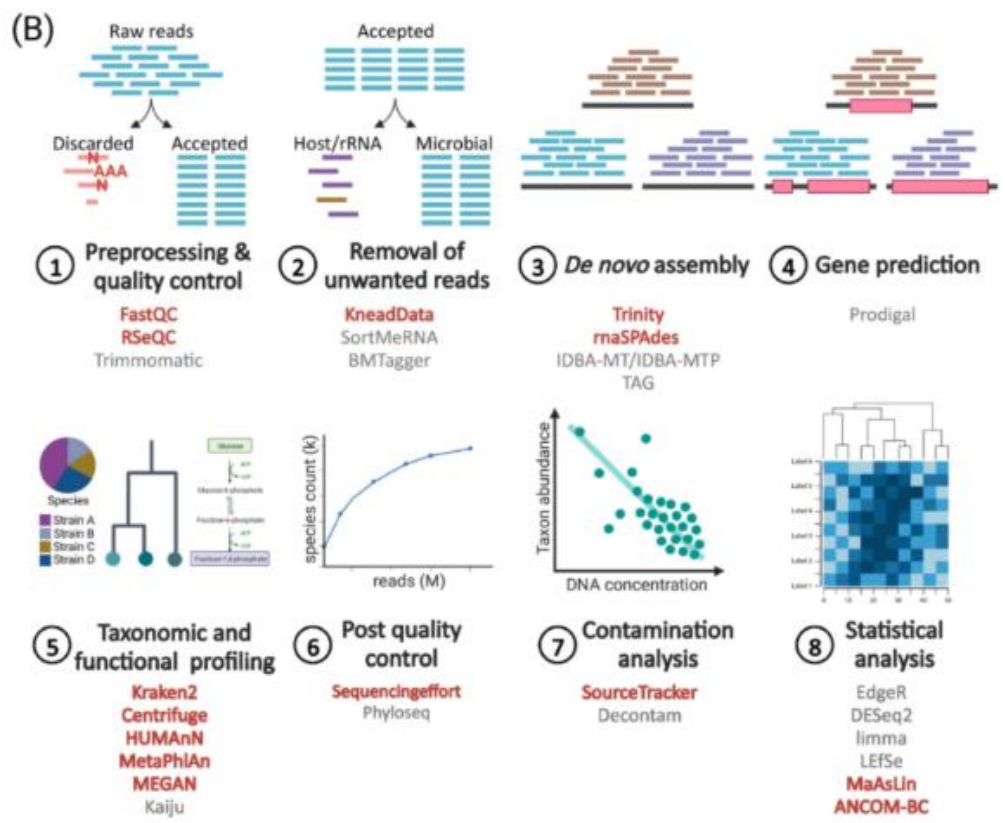
Unraveling metagenomics through long-read sequencing: a comprehensive review [38]

Long reads metagenomics

- Long read data can be used to improve assembly
- Bottlenecks :
  - DNA extraction (?)
  - cost of data generation
  - sequencing errors
- State of the art pipeline for assembly :
  - standalone long read assembly (ex: MetaFLYE [39])
  - optional error correction with short reads

Metatranscriptomics

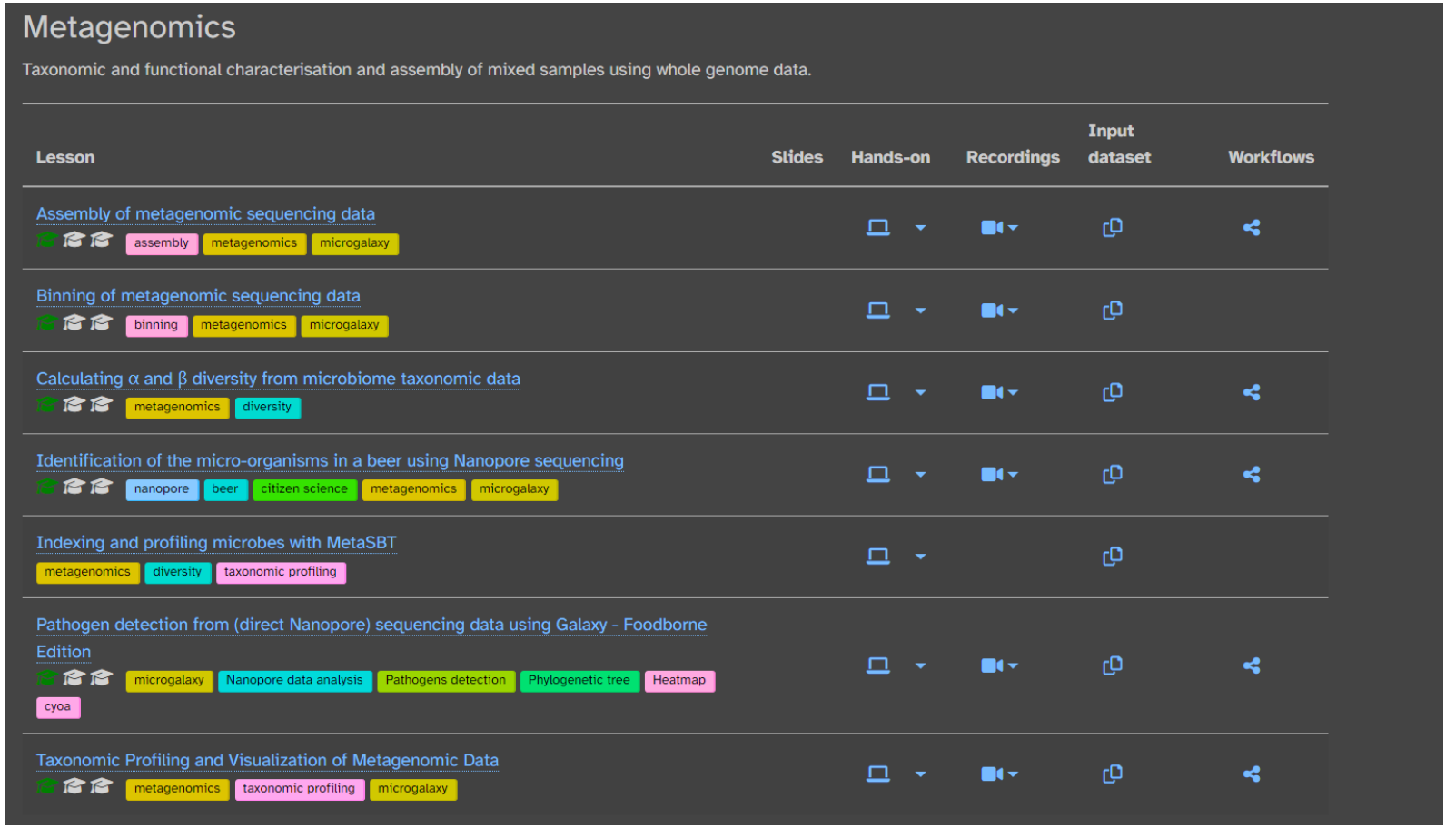




Hand-on training

- https://training.galaxyproject.org/training-material/topics/microbiome/

Hand-on training



Metagenomics

Taxonomic and functional characterisation and assembly of mixed samples using whole genome data.

Lesson	Slides	Hands-on	Recordings	Input dataset	Workflows
Assembly of metagenomic sequencing data assembly metagenomics microgalaxy					
Binning of metagenomic sequencing data binning metagenomics microgalaxy					
Calculating α and β diversity from microbiome taxonomic data metagenomics diversity					
Identification of the micro-organisms in a beer using Nanopore sequencing nanopore beer citizen science metagenomics microgalaxy					
Indexing and profiling microbes with MetaSBT metagenomics diversity taxonomic profiling					
Pathogen detection from (direct Nanopore) sequencing data using Galaxy - Foodborne Edition microgalaxy Nanopore data analysis Pathogens detection Phylogenetic tree Heatmap cyoa					
Taxonomic Profiling and Visualization of Metagenomic Data metagenomics taxonomic profiling microgalaxy					

Take home message

- Shotgun metagenomics is still an ongoing active bioinformatics research field
- Numerous software dedicated to assembly, binning, functional annotation are actively developed
- Depending on the ecosystem , one can have different approaches :
  - mapping on a reference database
  - assembly and mapping
- The biological question must determine the analysis

Thank you