Metagenomic read based processing

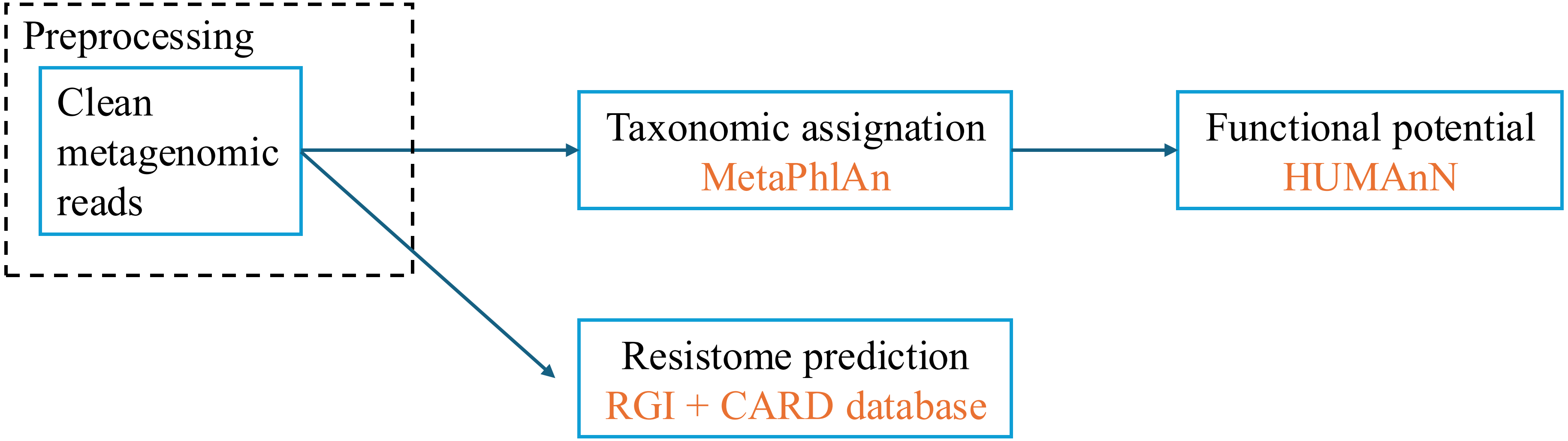
A pipeline of the Vonaesch Lab for the curnagl cluster

Simon Yersin

Introduction

- These bioinformatic scripts attempts to describe the steps to go from clean (adapters, low quality reads and host reads removed) reads to taxonomy tables and functional potential tables
- The pipeline include the following general steps:
  - MetaPhlAn taxonomic assignation
  - HUMAnN functional potential prediction
  - Prediction of resistome with RGI and CARD or GROOT
- Scripts for the HPC cluster

Pipeline for Read-based analysis



Preprocessing

Clean metagenomic reads

Taxonomic assignation
MetaPhlAn

Functional potential
HUMAnN

Resistome prediction
RGI + CARD database

PREPARATION

- We are assuming that the following steps have already been performed upon reception of the data:
  - Cleaning of the reads
  - Reads on the HPC cluster scratch
- If necessary, run 00_create_dir.sh to copy your reads into directories for each samples

MetaPhlAn – Taxonomy profiling

- MetaPhlAn is a computational tool for profiling the composition of microbial communities (Bacteria, Archaea and Eukaryotes) from metagenomic shotgun sequencing data (i.e. not 16S) with species-level.
- MetaPhlan4 is currently installed on the curnagl cluster with the ChocoPhlAn database : vJun23_CHOCOPhlAnSGB_202307
- Run MetaPhlAn4 using:
  - 📜01_metaphlan.sh
- MetaPhlAn4 produce 2 tables per samples:
  - Relative abundance
  - Absolute abundance (read count)

GTDB taxonomy and merging

- MetaPhlAn4 use the NCBI taxonomy, we can change the tables (relative abundance only) to the GTDB taxonomy using:
  - 📜 02_metaphlan_profile_gtdb.sh
- As MetaPhlAn4 produce one table per samples, we can merge them into a single table to produce one table in relative abundance, one table in absolute abundance and one table in relative abundance with the GTDB taxonomy using:
  - 📜 03_merge_profiles.sh
- These tables can then be imported in RStudio for taxonomic profile analysis

HUMAnN - Functional potential

- HUMAnN (v.3.9) is a method for efficiently and accurately profiling the abundance of microbial metabolic pathways and other molecular functions from metagenomic sequencing data.
- We can provide the MetaPhlAn profile to HUMAnN as we already did it in the previous step.
- Run HUMAnN using:
  - 01_humann.sh
- For each sample, humann will produce the following tables:
  - genefamilies.tsv : details the abundance of each gene family in the community
  - pathabundance.tsv details the abundance of each pathway in the community
  - pathcoverage
  - log

HUMAnN - Functional potential

- By default, HUMAnN will annotate the genes using the uniref90 database. The output file, genefamily.tsv, can be changed into other database annotations such as:
  - MetaCyc Reactions, KEGG Orthogroups, Pfam domains, Level-4 enzyme commission categories, EggNOG, Gene Ontology
  - Change uniref90 to other annotations using:

    📜 02_humann_regroup.sh
- Normalize the abundance output files (copy per millions and/or relative abundance), including the ones with changed annotations using:

  📜 03_humann_norm.sh
- Unpack pathways using:

  📜 04_humann_unpack.sh
- Merge the output files by annotations type using:

  📜 05_humann_merge.sh

HUMAnN3 output files:

Gene family file details the abundance of each gene family in the community. Gene family abundance at the community level is stratified to show the contributions from known and unknown species. Individual species' abundance contributions sum to the community total abundance. Gene family can be in different nomenclature (see next slide), by default in UniRef90. Gene families are part of pathways, for each sample, the unpack function break down the abundance of each gene family within a pathway.

Path abundance file details the abundance of each pathway in the community as a function of the abundances of the pathway's component reactions, with each reaction's abundance computed as the sum over abundances of genes catalyzing the reaction. By default, the pathways are in MetaCyc pathways definition.

Microbial metabolism analysis: :KOs, MetaCyc or EC

- KEGG Orthogroups (KOs): Functional orthologous groups, genes grouped based on functional similarity to link genes to biological functions and pathways.

  + Can be mapped to KEGG pathways

  - Limited curation, some genes are assigned to multiple function

- **MetaCyc** Reactions: Catalogue of metabolic reactions and pathways

  + Curated, comprehensive for bacteria and allow pathway reconstructions at finer levels, pathways abundance file in metacyc classification

  - Harder to interpret gene-level contribution since it focuses on reactions

- Level-4 enzyme commission (EC) categories: Standardized enzyme classification based on catalytic activity

  + Useful for looking at specific enzyme-catalyzed reactions

  - Doesn't provide full pathways, overlapping and ambiguous EC numbers

Protein function analysis: Uniref90 or Pfam

- UniRef90: protein sequences
- Pfam domains: Protein domains and families, defines funtional and structural protein domains.

Gene-level function analysis: GO or EggNOG

- EggNOG: Groups homologous genes into functional categories, predicts function of unknown genes based on evolutionary conservation
- Gene Ontology (GO): functional annotation of genes and proteins, standardized vocabulary for gene function across species (Three categories: Biological Process, Molecular Function and Cellular Component)

Resistome prediction – Option 1

- Resistance Gene Identifier (RGI) from the Comprehensive Antibiotic Resistance Database (CARD, a bioinformatic database of resistance genes, their products and associated phenotypes) can be used to predict resistome from protein and nucleotide data based on homology and SNP models.
- RGI can align short reads DNA sequence in FASTQ format using Bowtie2, BWA or KMA (recommended) against CARD's protein homolog models. For RGI bwt, fastq sequences can be aligned to the canonical curated CARD reference sequences associated with the Antibiotic Resistance Ontology (published sequences) or additionally to the *in silico* predicted allelic variants available in CARD's Resistome & Variants data set (recommended for non-clinical samples).
- Predict the resistome using:

rgi_bwt.sh

  - Use parameter --include_wildcard to include Resistome and Variants data

Resistome prediction – Option 2

- GROOT (Graphic Resistance out of metagenomes) is a tool to type Antibiotic Resistance Genes (ARGs) in metagenomic samples (a.k.a Resistome Profiling). Currently, the groot-db is installed and it is made by combining all sequences in ResFinder, ARG-annot and CARD databases (duplicates are removed and the sequences are then clustered at 90% identity).
- To run GROOT, we first need to filter out reads that are too small in length (under 31bp) using bbmap (installed in GROOT conda environment). Then, GROOT can profile the resistome. Finally, from A. Almeida scripts (https://github.com/alexmsalmeida/amr-profiler/tree/main), we produce two output tables. Profile the resistome in the samples using:

  groot.sh

  Be sure to have: parse_groot-report.py and res_classes.tsv
- Output:
  - amr_count.tsv: includes the gene found in the samples (coverage of the genes by reads >=95%), the read count and the reference gene length.
  - amr_classes.tsv: number of genes found that belong to each class of AMR.