Automated Analysis Pipeline for Bacterial Genome Sequencing Data

Final Assignment – Part A

Sajjad Khawari and Michael Riethmüller
Applied Sequence Analysis

Summer Term 2025
Instructor: Dr. Sandro Andreotti July 8, 2025

Contents

1 Introduction

In this project, we design a workflow for the analysis of bacterial whole-genome sequencing data. The data was generated using Illumina (short reads) and optionally also PacBio or Oxford Nanopore (long reads). Our aim is to build a flexible and reproducible pipeline using Snakemake, which automates common steps like assembly, annotation, and quality assessment.

The workflow also supports further downstream analyses such as MLST typing, screening for antibiotic resistance or virulence genes, and plasmid detection. A core genome analysis and phylogeny will allow for comparisons across the sequenced samples.

To make the pipeline as reusable as possible, we included options to activate or skip individual modules via a configuration file. We also allow the user to exclude low-quality samples from the phylogeny or to add an external genome as an outgroup.

In the following sections, we describe the main steps of the workflow and explain our choice of tools.

2 Workflow Overview

The proposed workflow consists of the following main modules:

1. **Preprocessing and Quality Control**
2. **Genome Assembly (short reads only or hybrid with long reads)**
3. **Assembly Quality Assessment**
4. **Genome Annotation**
5. **Downstream Analyses (MLST, resistance, virulence, plasmids)**
6. **Comparative Genomics (core genome, phylogeny)**
7. **Reporting and Aggregation**

While preprocessing, genome assembly, and assembly quality assessment are core components that are always executed, the modules for genome annotation, downstream analyses, comparative genomics, and reporting/aggregation are configurable and can be selectively enabled or disabled through the configuration file. This modular architecture, as illustrated in Figure 1, ensures flexibility for a wide range of project needs. Parallelization and resource management are handled by Snakemake, utilizing Conda environments to guarantee reproducibility and scalability.

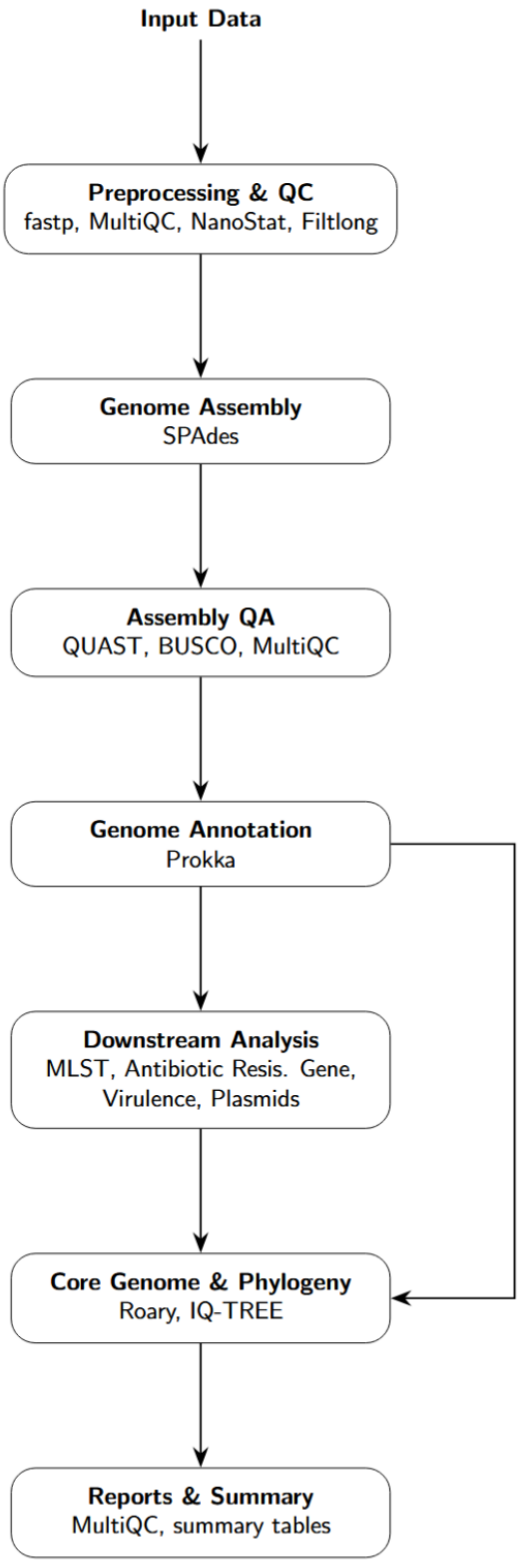

Figure 1: Overview of the bacterial genome analysis workflow. The pipeline includes preprocessing, assembly, quality control, annotation, optional downstream analyses, phylogenetic comparison, and reporting.

3 Detailed Steps and Software Selection

For each step of our workflow, we selected modern and efficient tools that are well-suited for bacterial genome analysis and are easily integrable with Snakemake. Below, we outline the main steps of the workflow and justify our tool choices for each.

3.1 Preprocessing and Quality Control

High-quality input data is essential for accurate downstream analysis.

- **Short Reads (Illumina):**
  - *Quality control:* **FastQC** [1] is employed to generate comprehensive quality reports, allowing visual inspection of sequencing quality, GC content, and adapter contamination.
  - *Adapter/quality trimming:* **fastp** [2] is an all-in-one FASTQ preprocessor known for its speed, comprehensive reporting, and ability to handle both paired-end and single-end reads.
  - *QC report aggregation:* **MultiQC** [3] aggregates the results from multiple samples and tools into a unified summary report for easy interpretation and documentation.
- **Long Reads (ONT/PacBio):**
  - *Quality control:* **NanoStat** [4] provides comprehensive summary statistics for long-read sequencing data, including read length distributions, N50, and quality metrics, facilitating rapid assessment of data quality prior to downstream analysis.
  - *Filtering:* **Filtlong** [5] is used to filter reads based on quality and length, ensuring that only high-quality reads are passed to the assembly step.

3.2 Genome Assembly

Accurate and contiguous genome assemblies are critical for downstream analyses. For this workflow, we selected **SPAdes** [6] as our sole assembly tool. SPAdes is highly performant, widely adopted for bacterial genome projects, and supports both short-read (Illumina) and hybrid assemblies (Illumina + long reads). Its hybrid mode enables the incorporation of Oxford Nanopore or PacBio data alongside Illumina reads, resulting in improved assembly contiguity and completeness.

3.3 Assembly Quality Assessment

Assessing the quality and completeness of genome assemblies is crucial for ensuring reliable downstream analyses and biological interpretation. Our workflow incorporates multiple complementary tools to evaluate assembly integrity:

- **Contiguity and Structural Metrics:** We utilize **QUAST** [7] to compute a suite of standard assembly statistics, including the number of contigs, N50, total assembly length, largest contig, GC content, and misassembly rates. These metrics provide a quantitative measure of assembly contiguity and structural correctness, allowing for comparison between different assemblies and parameter settings.

- **Completeness Assessment:** To assess the biological completeness of each assembly, we employ **BUSCO** [8]. BUSCO searches for the presence of near-universal, single-copy orthologous genes specific to the target lineage (e.g., bacteria). The results categorize these genes as complete, single-copy, duplicated, fragmented, or missing, providing an estimate of how much of the expected gene content is captured by the assembly.

- **Aggregated Reporting:** For ease of interpretation and project management, we use **MultiQC** [3] to collate and visualize the outputs from QUAST and BUSCO across all samples. MultiQC generates comprehensive summary reports, facilitating the identification of outliers and ensuring transparent quality control throughout the dataset.

This multifaceted approach gives confidence in the results used for annotation and comparative analyses by guaranteeing the structural and biological validity of assemblies.

3.4 Genome Annotation

Comprehensive genome annotation is essential for understanding the functional potential and biological characteristics of bacterial genomes. In our workflow, we employ **Prokka** [9], a widely adopted tool specifically designed for rapid, high-quality annotation of prokaryotic genomes. Prokka integrates multiple databases and analysis tools to perform both structural and functional annotation. For functional annotation, Prokka assigns putative gene functions by comparison to curated protein databases, such as UniProt, RefSeq, and Pfam. The tool also incorporates standard nomenclature and locus tags, ensuring the resulting annotations are consistent and suitable for downstream comparative analyses. Prokka outputs standard-compliant GFF files, which contain detailed information on gene coordinates and predicted functions. If a GTF format is required for downstream applications, it can be generated from the Prokka-produced GFF file using tools such as `gffread`.

3.5 Downstream Analyses

After genome assembly, a variety of downstream analyses can be performed to characterize genetic features relevant to epidemiology, antimicrobial resistance, virulence, and plasmid content. Each module described below is optional and can be enabled or disabled through the workflow's configuration file. All steps are optimized for contig-level (post-assembly) data, leveraging best-in-class tools readily integrated into Snakemake pipelines.

- **MLST (Multi-Locus Sequence Typing): mlst** [10] is used to determine the sequence type (ST) of each sample by scanning assembled contigs against the PubMLST database. This enables rapid, standardized epidemiological typing at the species level, facilitating comparisons across studies and public health surveillance. The workflow automatically detects the species (if not specified) and selects the corresponding allele scheme for typing.

- **Antibiotic Resistance Gene Detection: RGI** (Resistance Gene Identifier) [11] is run on assembled contigs to identify known antimicrobial resistance (AMR) genes using the Comprehensive Antibiotic Resistance Database (CARD). RGI provides detailed annotation, including the resistance mechanism, gene family, and predicted phenotype. As an alternative, **ABRicate** [12] can be configured to screen contigs against the CARD or ResFinder databases for a broader, rapid survey of AMR determinants.

- **Virulence Factor Identification: ABRicate** [12] is employed to search assembled contigs for virulence-associated genes using the Virulence Factors Database (VFDB). This step identifies putative virulence elements, aiding in the assessment of pathogenic potential and the investigation of outbreak strains.

- **Plasmid Screening:** To detect plasmid-borne sequences, **ABRicate** can be run against the PLSDB or PlasmidFinder databases, flagging contigs with high similarity to known plasmid replicons. Alternatively, machine learning-based tools like **PlasFlow** can be used for sequence-based plasmid prediction. These analyses are crucial for tracking the mobility of AMR or virulence genes and understanding horizontal gene transfer events.

Each analysis step produces standardized, tabular output files that summarize detected features per sample. Results can be aggregated for downstream comparative analyses or visualization. The modular nature of the pipeline allows seamless inclusion or exclusion of each analysis, and all tools are maintained with current databases to ensure up-to-date annotation.

3.6 Comparative Genomics and Phylogeny

For in-depth comparative genomics and phylogenetic reconstruction, the workflow integrates a set of established tools and flexible configuration options, enabling robust analysis across diverse bacterial datasets. The following modules are included:

- **Core Genome Clustering: Roary** [13] is employed to perform pan-genome analysis across all assembled genomes in the dataset. Roary rapidly clusters orthologous genes, distinguishing between core (present in all samples) and accessory (variable among samples) genome components. The workflow generates both the core gene alignment and presence/absence matrices, which can be further explored for gene content variation and epidemiological associations.

- **Phylogenetic Tree Construction: IQ-TREE** [14] is used to infer phylogenetic relationships based on the concatenated core genome alignment produced by Roary. IQ-TREE offers state-of-the-art model selection, ultrafast bootstrap analysis, and efficient maximum likelihood inference, producing robust phylogenies that reflect the evolutionary history of the sampled strains. Output includes Newick-formatted trees and branch support values, suitable for downstream visualization and annotation.
- **Blacklist Support:** To ensure analytical flexibility, users may specify a list of sample IDs to exclude from comparative analyses (e.g., low-quality assemblies, outlier strains, or contaminants). The workflow will automatically filter these samples from both the core genome clustering and subsequent phylogenetic analyses.
- **Outgroup Support:** For proper rooting and evolutionary context, users can provide an additional assembled genome to serve as an outgroup in the phylogenetic analysis. The workflow will include this assembly in the core genome alignment and ensure it is designated as the outgroup for tree rooting in IQ-TREE, allowing for clear inference of ancestor-descendant relationships.

All intermediate and final outputs—including gene presence/absence matrices, core genome alignments, and phylogenetic trees—are made available for downstream investigation. The modular design allows users to tailor the comparative genomics and phylogeny stage to their dataset, analytical goals, and quality considerations.

3.7 Reporting and Aggregation

- **Quality metrics: MultiQC** summarizes read and assembly QC as well as BUSCO results.
- **Tabular summary:** Results of MLST, resistance, virulence, and plasmid screening are aggregated into a single table or multi-sheet Excel file, using tools such as **pandas** or **openpyxl** in a custom script.
- **Logging:** Snakemake log files per rule provide full provenance tracking.

4 Workflow Implementation Details

- **Workflow manager:** Snakemake [15] for rule-based automation and parallelization.
- **Environment management:** Conda environments per tool, defined in `envs/`.
- **Configuration:** YAML config file specifying input samples, sequencing type, options for each analysis step, and blacklist/outgroup files.
- **Modularity:** Each module is implemented as an independent Snakemake rule or subworkflow. Steps can be (de)activated via config.

- **Reproducibility:** All software versions, parameters, and reference database versions are tracked in the workflow.
- **Scalability:** Snakemake will utilize available CPU cores and optionally cluster backends (e.g., SGE, SLURM).
- **Documentation:** The workflow will include usage instructions and a graphical DAG.

5 Conclusion

Our proposed Snakemake pipeline provides a comprehensive, reproducible, and extensible solution for bacterial genome assembly and analysis. It ensures high-quality results, streamlined reporting, and supports flexible comparative genomics, meeting both publication and future scalability requirements.

References

[1] Andrews, S. (2010). *FastQC: a quality control tool for high throughput sequence data.* https://www.bioinformatics.babraham.ac.uk/projects/fastqc/

[2] Chen, S. et al. (2018). fastp: an ultra-fast all-in-one FASTQ preprocessor. *Bioinformatics*, 34(17), i884-i890.

[3] Ewels, P. et al. (2016). MultiQC: summarize analysis results for multiple tools and samples in a single report. *Bioinformatics*, 32(19), 3047-3048.

[4] De Coster, W., D'Hert, S., Schultz, D. T., Cruts, M., & Van Broeckhoven, C. (2018). NanoPack: visualizing and processing long-read sequencing data. *Bioinformatics*, 34(15), 2666-2669.

[5] *Filtlong: quality filtering tool for long reads*, https://github.com/rrwick/Filtlong

[6] Bankevich, A. et al. (2012). SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing. *Journal of Computational Biology*, 19(5), 455-477.

[7] Gurevich, A. et al. (2013). QUAST: quality assessment tool for genome assemblies. *Bioinformatics*, 29(8), 1072-1075.

[8] Simão, F. A. et al. (2015). BUSCO: assessing genome assembly and annotation completeness with single-copy orthologs. *Bioinformatics*, 31(19), 3210-3212.

[9] Seemann, T. (2014). Prokka: rapid prokaryotic genome annotation. *Bioinformatics*, 30(14), 2068-2069.

[10] Seemann, T. (mlst), https://github.com/tseemann/mlst

[11] Jia, B. et al. (2016). CARD 2017: expansion and model-centric curation of the comprehensive antibiotic resistance database. *Nucleic Acids Research*, 45(D1), D566-D573.

[12] *ABRicate: mass screening of contigs for antimicrobial and virulence genes*, https://github.com/tseemann/abricate

[13] Page, A. J. et al. (2015). Roary: rapid large-scale prokaryote pan genome analysis. *Bioinformatics*, 31(22), 3691-3693.

[14] Nguyen, L.-T. et al. (2015). IQ-TREE: a fast and effective stochastic algorithm for estimating maximum-likelihood phylogenies. *Molecular Biology and Evolution*, 32(1), 268-274.

[15] Köster, J., Rahmann, S. (2012). Snakemake–a scalable bioinformatics workflow engine. *Bioinformatics*, 28(19), 2520-2522.