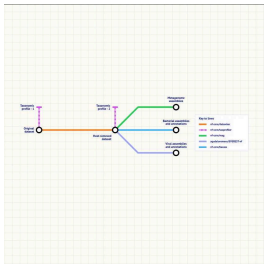
May 12, 2025 Version 2

Metagenome Processing in Clinical Setting for Respiratory Pathogens V.2



DOI
https://dx.doi.org/10.17504/protocols.io.36wgq6rdolk5/v2

Abhinav Sharma[1], Jose A Caparros-Martin[2], Patricia Agudelo-Romero[3]

[1]DSI-NRF Centre of Excellence for Biomedical Tuberculosis Research, SAMRC Centre for Tuberculosis Research, Division of Molecular Biology and Human Genetics, Faculty of Medicine and Health Sciences, Stellenbosch University, 7505, Cape Town, Western Cape, South Africa.;
[2]Wal-Yan Respiratory Research Centre, The Kids Research Institute Australia, 6009, Perth, Western Australia, Australia AND Curtin Health Innovation Research Institute (CHIRI), Curtin University, 6102, Perth, Western Australia, Australia.;
[3]Wal-Yan Respiratory Research Centre, The Kids Research Institute Australia, 6009, Perth, Western Australia, Australia AND Australian Research Council Centre of Excellence in Plant Energy Biology, School of Molecular Sciences, The University of Western Australia, 6009, Perth, Western Australia, Australia. AND European Virus Bioinformatics Center, Friedrich-Schiller-Universitat Jena, 07737, Thuringia, Germany.

Abhinav Sharma
Stellenbosch University

DOI: https://dx.doi.org/10.17504/protocols.io.36wgq6rdolk5/v2

Protocol Citation: Abhinav Sharma, Jose A Caparros-Martin, Patricia Agudelo-Romero 2025. Metagenome Processing in Clinical Setting for Respiratory Pathogens. **protocols.io** https://dx.doi.org/10.17504/protocols.io.36wgq6rdolk5/v2Version created by **Abhinav Sharma**

Protocol status: Working
We use this protocol and it's working

Created: May 12, 2025

Last Modified: May 12, 2025

Protocol Integer ID: 218080

Keywords: Bioinformatics, Metagenomics, Viruses, respiratory pathogens metagenomic bioinformatics pipeline, metagenomic bioinformatics pipeline, metagenomic dataset, metagenomic, simultaneous profiling of diverse microbial taxa, metagenomic research, critical gap in metagenomic, producing comprehensive microbial profile, reproducibility of metagenomic research, comprehensive microbial profile, metagenome processing, metagenome processing in clinical setting, full spectrum of microbial community, respiratory pathogen, diverse microbial taxa, sequencing data, microbial community, extracting meaningful biological insight, specific microbial group, meaningful biological insights from the vast amount, bioinformatics, accurate removal of human dna read, microbial study, host dna removal, pathogen detection, sensitivity of pathogen detection, bacteria, human dna read, simultaneous profiling, including bacteria, significant computational resource

Funders Acknowledgements:
Google Cloud Education Program grant
Grant ID: na
Branchi Foundation
Grant ID: na

Abstract

Metagenomic bioinformatics pipelines are essential for extracting meaningful biological insights from the vast amounts of sequencing data generated in microbial studies. However, current approaches face several challenges that limit their utility and accessibility.

A major hurdle is the efficient and accurate removal of human DNA reads from metagenomic datasets, a critical step for improving the sensitivity of pathogen detection. This process often demands significant computational resources and can become a bottleneck in large-scale analyses. In addition, most existing pipelines are designed to focus on specific microbial groups, making it difficult to comprehensively analyse the full spectrum of microbial communities, including bacteria and viruses, from a single sample. This lack of unified, end-to-end solutions limits both the scope and reproducibility of metagenomic research.

To address these limitations, we developed a comprehensive protocol that streamlines host DNA removal and enables the simultaneous profiling of diverse microbial taxa from a single FASTQ file. This workflow consolidates existing pipelines within a unified framework, optimising computational efficiency while producing comprehensive microbial profiles. Our approach bridges a critical gap in metagenomics by supporting both clinical and environmental research needs in a scalable and conscious manner.

Attachments

SVG

protocol-v1.svg

32KB

Guidelines

This protocol was created as a guide for new users. We strongly recommend exploring your data and reading the manuals for the software we present before designing your own sequence data pipeline.

Materials

The following 5 softwares (bioinformatics pipelines) were used in this protocol:

1.

Software

NAME

nf-core/mag

REPOSITORY

https://github.com/nf-core/mag/

2.

Software

NAME

nf-core/detaxizer

REPOSITORY

https://github.com/nf-core/detaxizer

3.

Software

NAME

nf-core/taxprofiler

REPOSITORY

https://github.com/nf-core/taxprofiler

4.

Software	
p_agudeloromero/EVEREST-nf	NAME
https://github.com/agudeloromero/everest_nf	REPOSITORY

5.

Software	
nf-core/bacass	NAME
https://github.com/nf-core/bacass	REPOSITORY

Safety warnings

1. **[DATA INTEGRITY]** To preserve computational resources and avoid downstream errors in a multi-pipeline analysis, we strongly recommend verifying the integrity of your FASTQ files using MD5 checksums for each individual file prior to analysis.

2. **[INSTALLATION]** Perform a pre-flight check of your Java installation. Ensure that a Long-Term Support (LTS) version is being used rather than an internal or development build, which may cause unexpected issues. Refer to the official Nextflow documentation for guidance: https://www.nextflow.io/docs/latest/install.html

3. **[SAMPLESHEETS]** If your dataset includes samples from two or more batches, please ensure that sample names in the first column of the CSV samplesheet are unique across all batches. This avoids potential naming collisions during processing.

Before start

1. If you are working in an HPC environment, we recommend first downloading all required pipelines locally. Use either (i) the nf-core CLI tool for nf-core pipelines or (ii) the `nextflow pull` ther pipelines to ensure that the correct versions are pinned and reproducible.
2. Since public internet access is often restricted to the login node on HPC systems, we also advise downloading all necessary databases for the individual pipelines in advance of running any analysis.
3. Ensure that each pipeline is correctly configured for your HPC infrastructure. This can be done through a customised `nextflow.config` file or by using an institutional profile from the **nf-core/configs** repository.

Taxonomic profiling of reads (nf-core/taxprofiler)

1 Prepare the samplesheets for (i) FASTQ files as well as (ii) databases

Note

sample,run_accession,instrument_platform,fastq_1,fastq_2,fasta
SAMPLE2,SAMPLE2,ILLUMINA,/home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/SAMPLE2_R1_filtered.fastq.gz,/home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/SAMPLE2_R2_filtered.fastq.gz,
V3,V3,ILLUMINA,/home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/V3_R1_filtered.fastq.gz,/home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/V3_R2_filtered.fastq.gz,

For this inital profiling of sequences, we do not conduct any preprocessing (reads filtering, trimming, or QC) since we wish to have an overview of the current state of the sequences. If you deem it necessary to conduct some preprocessing for your samples, please refer to the relevant parameters of nf-core/taxprofiler pipeline https://nf-co.re/taxprofiler/1.2.0/parameters.

Set the appropriate parameters for the dataset via the db_params column.

Note

tool,db_name,db_params,db_path
centrifuge,centrifuge-db,,/home/opc/db_dir/centrifuge_p_compressed+h+v
kraken2,kraken2-db,--memory-mapping,/home/opc/db_dir/minikraken_8GB_20200312

2 Prepare the parameters YAML file with the appropriate options.

Note

input: /home/opc/samplesheet.taxprofiler.csv
databases : '/home/opc/samplesheet.dbs.taxprofiler.csv'

run_kraken2: true
kraken2_save_reads: true
run_centrifuge: true
run_krona: true

run_profile_standardization: true

save_untarred_databases: true

3 Initiate the pipeline using the following command

Command

nexflow run nf-core/taxprofiler

```
nextflow run nf-core/taxprofiler \
                -name oci-project-t-taxprofiler \
                -profile docker \
                -c taxprofiler.config \
                -with-tower \
                --outdir /home/opc/results/taxprofiler/oci-project-t-
taxprofiler \
                -params-file /home/opc/params.taxprofiler.yml \
                -work-dir /home/opc/work/oci-project-t-taxprofiler \
                -revision 1.2.0 \
                -resume
```

Removal of host reads (nf-core/detaxizer)

4 Prepare the samplesheets with the following format

Note

sample,short_reads_fastq_1,short_reads_fastq_2,long_reads_fastq_1
SAMPLE9,/home/opc/dataset/CAP11520/CAP11520A9/SAMPLE9_S15_L001_R1_001.fastq.gz,/home/opc/dataset/CAP11520/CAP11520A9/SAMPLE9_S15_L001_R2_001.fastq.gz,
V1,/home/opc/dataset/CAP11520/CAP11520A16/V1_S14_L001_R1_001.fastq.gz,/home/opc/dataset/CAP11520/CAP11520A16/V1_S14_L001_R2_001.fastq.gz,

5 Prepare the parameters YAML file with the appropriate options.

Note

input: /home/opc/samplesheet.detaxizer.csv

classification_bbduk: true
classification_kraken2: true
classification_kraken2_post_filtering: true
enable_filter: true

kraken2db: "/home/opc/db_dir/k2_standard_16gb_20240904.tar.gz"

save_output_fastqs_removed: false
save_output_fastqs_filtered: true
output_removed_reads: true
genome: "GRCh38"
fasta_bbduk: "/home/opc/db_dir/genome_GRCh38.fa"

generate_downstream_samplesheets: true
generate_pipeline_samplesheets: 'mag,taxprofiler'

6 Initiate the pipeline using the following command

Command

nextflow run nf-core/detaxizer

```
nextflow run nf-core/detaxizer \
                -name oci-ol8-project-t-detaxizer-originalfastqs \
                -profile docker \
                -c detaxizer.config \
                -with-tower \
                --outdir /home/opc/results/detaxizer/oci-ol8-aerial-
t-detaxizer-originalfastqs \
                -params-file /home/opc/params.detaxizer.yml \
                -work-dir /home/opc/work/oci-ol8-aerial-t-detaxizer-
originalfastqs \
                -revision 1.1.0 \
                -resume
```

Confirmation of host-read removal (nf-core/taxprofiler)

7 Prepare the samplesheets for (i) FASTQ files as well as (ii) databases

Note

sample,run_accession,instrument_platform,fastq_1,fastq_2,fasta
SAMPLE2,SAMPLE2,ILLUMINA,/home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/SAMPLE2_R1_filtered.fastq.gz,/home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/SAMPLE2_R2_filtered.fastq.gz,
V3,V3,ILLUMINA,/home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/V3_R1_filtered.fastq.gz,/home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/V3_R2_filtered.fastq.gz,

Note

tool,db_name,db_params,db_path
centrifuge,centrifuge-db,,/home/opc/db_dir/centrifuge_p_compressed+h+v
kraken2,kraken2-db,--memory-mapping,/home/opc/db_dir/minikraken_8GB_20200312

8 Prepare the parameters file (YAML format) with the appropriate options.

Note

input: /home/opc/samplesheet.taxprofiler.csv
databases : '/home/opc/samplesheet.dbs.taxprofiler.csv'

run_kraken2: true
kraken2_save_reads: true
run_centrifuge: true
run_krona: true

run_profile_standardization: true

save_untarred_databases: true

9 Initiate the pipeline using the following command

Command

nextflow run nf-core/taxprofiler

```
nextflow run nf-core/taxprofiler \
                -name oci-project-t-taxprofiler-afterdetaxizer \
                -profile docker \
                -c taxprofiler.config \
                -with-tower \
                --outdir /home/opc/results/taxprofiler/oci-project-t-
taxprofiler-afterdetaxizer \
                -params-file /home/opc/params.taxprofiler.yml \
                -work-dir /home/opc/work/oci-project-t-taxprofiler-
afterdetaxizer \
                -revision 1.2.0 \
                -resume
```

10 Once this step is complete, it is possible to initiate the following two steps in parallel, if sufficient computational resources are available.

go to step #11

go to step #14

go to step #17

Generation of assemblies and bins (nf-core/mag)

11 Prepare the samplesheet file (CSV format) using the output of nf-core/detaxizer after host-reads removal.

Note

sample,group,short_reads_1,short_reads_2,long_reads
SAMPLE2,0,/home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/SAMPLE2_R1_filtered.fastq.gz,/home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/SAMPLE2_R2_filtered.fastq.gz,
V3,0,/home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/V3_R1_filtered.fastq.gz,/home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/V3_R2_filtered.fastq.gz,

12 Prepare the parameters file (YAML format) with the appropriate options.

Note

input: /home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/downstream_samplesheets/mag-pe.csv

run_virus_identification : false

binqc_tool: checkm2
skip_megahit: false
skip_maxbin: false
skip_spades: false
skip_spadeshybrid: true
skip_quast: false
skip_metaeuk : true
skip_prokka : true
ancient_dna : false
skip_metabat2: false
skip_concoct: false

cat_db: /home/opc/db_dir/CAT_prepare_20210107.tar.gz
kraken_db: /home/opc/db_dir/k2_standard_16gb_20240904.tar.gz

13 Initiate the pipeline using the following command

Command

nextflow run nf-core/mag

```
nextflow run nf-core/mag \
                -name oci-project-t-mag-afterdetaxizer \
                -profile docker \
                -c mag.config \
                -with-tower \
                --outdir /home/opc/results/mag/oci-project-t-mag-
afterdetaxizer \
                -params-file /home/opc/params.mag.yml \
                -work-dir /home/opc/work/oci-project-t-mag-
afterdetaxizer \
                -revision 3.3.0 \
                -resume
```

Identification of viral taxons (p_agudeloremero/EVEREST-nf)

14 Prepare the samplesheet using the output of nf-core/detaxizer after host-reads removal.

Note

sample,fastq_1,fastq_2
SAMPLE9,/home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/SAMPLE9_R1_filtered.fastq.gz,/home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/SAMPLE9_R2_filtered.fastq.gz
V10,/home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/V10_R1_filtered.fastq.gz,/home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/V10_R2_filtered.fastq.gz

15 Prepare the parameters YAML file, with the appropriate options.

Note

input: /home/opc/results/detaxizer/oci-ol8-project-t-detaxizer-originalfastqs-28187/filter/filtered/samplesheet.everest.csv
rnaseq: false

DATABASES
pharokka_db: /home/opc/db_dir/everest/DB_pharokka/
diamond_db_aa: /home/opc/db_dir/everest/DB_aa_diamond/viral_aa_diamond.dmnd
adaptor: /home/opc/db_dir/everest/adaptors/adaptors.fa
baltimore_db: /home/opc/db_dir/everest/DB_Baltimore/Baltimore_group_ICTV_2021_Generated_16052022_v2.txt
checkv_db: /home/opc/db_dir/everest/DB_checkv/checkv-db-v1.1
mmseq_viral_db_aa: /home/opc/db_dir/everest/DB_MMSEQ2_aa
mmseq_viral_db_aa_ref_name: "viral.aa.fnaDB"
mmseq_viral_db_nt: /home/opc/db_dir/everest/DB_MMSEQ2_nt
tax_aa: /home/opc/db_dir/everest/TAX_aa
tax_nt: /home/opc/db_dir/everest/TAX_nt
virsorter_db: /home/opc/db_dir/everest/DB_virsorter

REFERENCES

genome: /home/opc/db_dir/everest/genome/male.hg19.fasta
transcriptome: /home/opc/db_dir/everest/genome/Homo_sapiens.GRCh38.cdna.all.fa
fasta: /home/opc/db_dir/everest/genome/Homo_sapiens.GRCh38.cdna.all.fa

16 Initiate the pipeline using the following command

Command

nextflow run agudeloromero/everest_nf

```
nextflow run agudeloromero/everest_nf
                -c /home/opc/EVEREST/everest.config \
                -name oci-project-t-everest \
                -profile docker \
                -with-tower \
                --outdir /home/opc/EVERSET/results/everest-nf/oci-
aerial-t-everest-26858 \
                -params-file /home/opc/EVEREST/params.data_test.yaml
\
                -r develop \
                -work-dir /home/opc/EVEREST/work/ \
                -resume \
                -latest
```

Bacterial assembly and annotation (nf-core/bacass)

17 Prepare the samplesheet using the output of nf-core/detaxizer after host-reads removal.

Note

ID R1 R2 LongFastQ Fast5 GenomeSize
shortreads ./data/S1_R1.fastq.gz ./data/S1_R2.fastq.gz NA NA NA
longreads NA NA ./data/S1_long_fastq.gz ./data/FAST5 2.8m
shortNlong ./data/S1_R1.fastq.gz ./data/S1_R2.fastq.gz ./data/S1_long_fastq.gz ./data/FAST5 2.8m

Bacterial assembly and annotation (nf-core/bacass)

18 Prepare the parameters YAML file, with the appropriate options.

Note

input: /home/p_agudeloromero/myscratch/microbiome/PROJECT_T/samplesheet.bacass.tsv
kraken2db: /home/setonix/p_agudeloromero/db_dir/k2_standard_16gb_20240904.tar.gz

19 Initiate the pipeline using the following command

Command

nextflow run nf-core/bacass

```
nextflow run /home/p_agudeloromero/mysoftware/nf-core-
bacass_2.3.1/2_3_1 \
                -name setonix-project-t-bacass \
                -c /home/p_agudeloromero/bacass.config \
                -profile test,pawsey_setonix \
                -params-file
/home/p_agudeloromero/params.bacass.yaml \
                -work-dir
\/scratch/pawsey0876/p_agudeloromero/deleteme/work/setonix-project-t-
bacass \
                --outdir
/scratch/pawsey0876/p_agudeloromero/deleteme/results/setonix-aerial-t-
bacass \
                -with-tower \
                -resume
```

Protocol references

Di Tommaso, P., Chatzou, M., Floden, E. *et al.* Nextflow enables reproducible computational workflows. *Nat Biotechnol* **35**, 316–319 (2017). https://doi.org/10.1038/nbt.3820

Ewels, P.A., Peltzer, A., Fillinger, S. *et al.* The nf-core framework for community-curated bioinformatics pipelines. *Nat Biotechnol* **38**, 276–278 (2020). https://doi.org/10.1038/s41587-020-0439-x

James A. Fellows Yates, Sofia Stamouli, Lili Andersson-Li, Moritz E. Beber, nf-core bot, Lauri Mesilaakso, Thomas A. Christensen II, Mahwash Jamy, JIANHONG OU, Maxime Borry, Rafal Stepien, Håkon Kaspersen, Joon Klaps, Husen M. Umer, Matthias Hörtenhuber, Robert Syme, Alex Hübner, Ian Light, Maxime U Garcia, & Zandra Fagernäs. (2024). nf-core/taxprofiler: v1.2.0 - Bouncy Basenji [2024-10-03] (1.2.0). Zenodo. https://doi.org/10.5281/zenodo.13885203

James A. Fellows Yates, Sabrina Krakau, Daniel Straub, Hadrien Gourlé, Jim Downie, Maxime Borry, nf-core bot, Antonia Schuster, PhilPalmer, Daniel Lundin, Carson J Miller, Gregory Sprenger, willros, Alex Hübner, Maxime U Garcia, Alexander Ramos, Phil Ewels, Gisela Gabernet, Till E., ... Michael L Heuer. (2024). nf-core/mag: mag 3.1.0 - Pink Panda (3.1.0). Zenodo. https://doi.org/10.5281/zenodo.13889068

Jannik Seidel, Jannik, nf-core bot, Joon Klaps, Friederike Hanssen, Matthias Hörtenhuber, Phil Ewels, & Júlia Mir Pedrol. (2024). nf-core/detaxizer: detaxizer 1.1.0 - Kombjuudr (1.1.0). Zenodo. https://doi.org/10.5281/zenodo.14056601

Agudelo-Romero, P. & Sharma, A. (2025). A Nextflow-Based Automated Pipeline for Viral Assembly and Characterisation (EVEREST). Zenodo. https://doi.org/10.5281/zenodo.14963685

Daniel VM, Alexander Peltzer, Daniel Straub, nf-core bot, Florian Wuennemann, Maxime U Garcia, Ben Taylor, Angel Angelov, Phil Ewels, Y. Zhou, Adam Talbot, Harshil Patel, Matthias Hörtenhuber, Ramon Rivera, & Kevin Menden. (2024). nf-core/bacass: v2.3.1 (2.3.1). Zenodo. https://doi.org/10.5281/zenodo.14180424