nf-core/viralmetagenome: A Novel Pipeline for Untargeted Viral Genome Reconstruction

Joon Klaps[1,*], Philippe Lemey[1], nf-core community[2], Liana Eleni Kafetzopoulou[1]

[1]Rega Institute for Medical Research, Department of Microbiology, Immunology and Transplantation, KU Leuven, Belgium

[2]A full list of contributors can be found at https://nf-co.re/community.

*Correspondence: joon.klaps@kuleuven.be

July 21, 2025

Abstract

Motivation: Eukaryotic viruses present significant challenges for genome reconstruction and variant analysis due to their extensive diversity and potential genome segmentation. While de novo assembly followed by reference database matching and scaffolding is a commonly used approach, the manual execution of this workflow is extremely time-consuming, particularly due to the extensive reference curation required. Here, we address the critical need for an automated, scalable pipeline that can efficiently handle viral metagenomic analysis without manual intervention.

Results: We present nf-core/viralmetagenome, a comprehensive viral metagenomic pipeline for untargeted genome reconstruction and variant analysis of eukaryotic DNA and RNA viruses. Viralmetagenome is implemented as a Nextflow workflow that processes short-read metagenomic samples to automatically detect and assemble viral genomes, while also performing variant analysis. The pipeline features automated reference selection, consensus quality control metrics, comprehensive documentation, and seamless integration with containerization technologies, including Docker and Singularity. We demonstrate the utility and accuracy of our approach through validation on both simulated and real datasets, showing robust performance across diverse viral families in metagenomic samples.

Availability: nf-core/viralmetagenome is freely available at https://github.com/nf-core/viralmetagenome with comprehensive documentation at https://nf-co.re/viralmetagenome

Contact: joon.klaps@kuleuven.be

Supplementary information: Supplementary data are available at https://github.com/Joon-Klaps/nf-core-viralmetagenome-manuscript online.

Keywords: viralmetagenome, bioinformatic pipeline, nextflow, viral metagenomics, viral assembly, viral variant analysis

1 Introduction

Reconstructing viral genomes from metagenomic sequencing data presents considerable computational challenges, particularly for viruses exhibiting extensive genetic diversity [1, 2, 3]. This diversity is further compounded by segmented genomes in families like influenza, rotavirus, and bunyaviruses, where individual segments can evolve under distinct selective pressures and reassort, contributing to a complex landscape for genome reconstruction. While pipelines often target specific viruses [4], accurate and complete genome reconstruction of samples with unknown references typically requires manual curation of contigs and reference matching [5, 6, 7], making it impractical for large-scale studies or rapid outbreak response.

To address these limitations, we developed nf-core/viralmetagenome, a comprehensive pipeline specifically designed for untargeted viral genome reconstruction. The pipeline is developed using Nextflow [8] within the nf-core framework [9], ensuring reproducibility through containerization with Docker [10] and Singularity [11], and enabling portability across computational platforms such as local desktops, high-performance clusters and cloud environments.

2 Pipeline Description

Nf-core/viralmetagenome implements an automated workflow performing *de novo* assembly, reference matching, and iterative refinement to reconstruct viral genomes without prior target knowledge. The pipeline consists of five stages: read preprocessing, metagenomic diversity assessment, contig assembly and scaffolding, iterative consensus refinement with variant analysis, and quality control (Figure 1). Tool details are in Supplementary Table 1. Multiple options accommodate established workflows. Source code: https://github.com/nf-core/viralmetagenome.

3 Implementation

Nf-core/viralmetagenome requires only Nextflow and a container system (Docker, Singularity, or Conda):

```
nextflow run nf-core/viralmetagenome \
  -profile docker \
  --input samplesheet.csv \
  --output results
```

Input is provided via sample sheets (CSV, TSV, YAML, or JSON) containing sample names and FASTQ paths. The pipeline supports single/paired-end data, UMIs, and sequencing run merging.

3.1 Read preprocessing

The preprocessing module performs quality control using FastQC and adapter trimming with Fastp [12] (default) or Trimmomatic [13]. UMI deduplication uses HUMID [14] and UMI-tools [15].

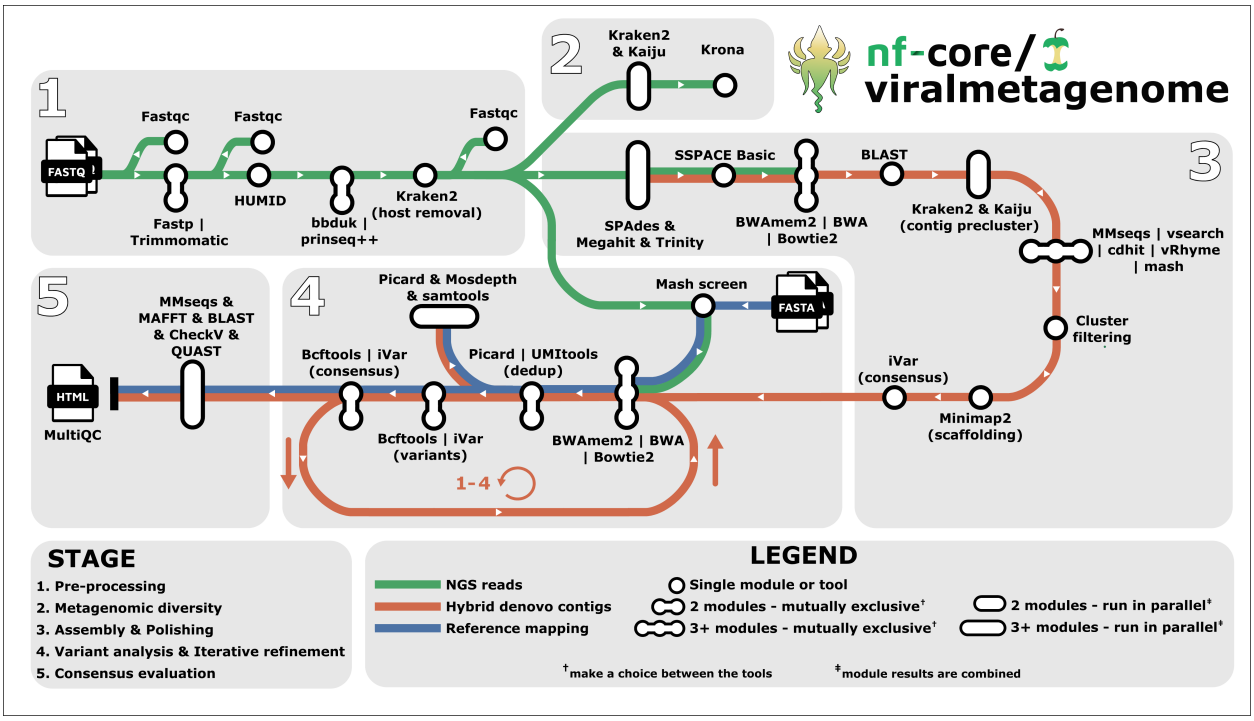

Figure 1: Visual overview of the nf-core/viralmetagenome pipeline for untargeted viral genome reconstruction. nf-core/viralmetagenome processes short-read FASTQ files through optionally read pre-processing (adapter removal, quality filtering, host removal), metagenomic diversity assessment, *de novo* assembly with multiple assemblers, scaffolding with automated reference identification and contig taxonomy-guided clustering, and iterative consensus refinement through read mapping and variant calling. Quality control metrics, assembly statistics, and coverage data are integrated into interactive MultiQC reports and standardised overview tables for downstream analysis.

Multiple sequencing runs are merged after trimming by specifying group identifiers in the sample sheet. Complexity filtering with BBduk [16] or prinseq++ [17] removes repetitive sequences. Host removal uses Kraken2 [18] against user-specified databases (default: human genome subset).

3.2 Metagenomic diversity assessment

Taxonomic classification of preprocessed reads is performed using two complementary approaches - Kaiju [19] and Kraken2 [18] - to maximise detection sensitivity across diverse viral families. Results from both classifiers are visualised using Krona [20].

3.3 *De novo* assembly and clustering

The assembly workflow implements multi-assembler approaches with clustering and scaffolding. *De novo* assembly uses SPAdes [3] (RNAviral mode), MEGAHIT [21], or Trinity [22], capitalizing on distinct algorithmic strengths. Optional contig extension uses SSPACE Basic [23].

Reference identification uses BLASTn [24] against the Reference Viral Database [25], retaining top five hits for taxonomy-guided clustering.

Clustering employs two stages: taxonomic pre-clustering using Kraken2 [18] and Kaiju [19], followed by nucleotide similarity clustering with CD-HIT-EST [26], VSEARCH [27], MMseqs2 [28], vRhyme [29], or Mash [30].

As an optional filtering step of contig clusters, after assembly and extension, reads can be mapped to all contigs using BWAmem2 [31] (default), BWA [32], or Bowtie2 [33]. Clusters are filtered based on the cumulated percentage of reads mapped to the contigs of a cluster. By filtering clusters, low-coverage assemblies can be identified that likely represent assembly artefacts.

For the final scaffolding step, all cluster members are mapped to the cluster representative or centroid using Minimap2 [34], followed by consensus calling with iVar [35] to generate reference-assisted assemblies. Regions with zero coverage depth can optionally be represented by the reference genome to produce a more complete scaffold genome for consensus calling.

3.4 Iterative consensus refinement and variant calling

The consensus module supports external reference-based analysis and scaffold refinement. Users can provide references via `-mapping_constraints`; when multiple genomes are provided, the most similar is selected using Mash [30].

Scaffold refinement performs up to 4 iterative cycles (default 2). Each iteration maps reads using BWAmem2 [31], BWA [32], or Bowtie2 [33], followed by variant calling with BCFtools [36] or iVar [35]. BCFtools outperforms iVar in precision and recall [37], while iVar detects more low-frequency variants, increasing false positives but reducing false negatives.

Optional deduplication can be performed using Picard or when UMI's are available with UMI-tools [15]. Mapping statistics are generated using samtools (flagstat, idxstats, stats) [36], Picard CollectMultipleMetrics [38], and coverage statistics with mosdepth [39].

3.5 Consensus Quality Control

Quality control employs CheckV [40] for completeness estimates, BLASTn [24] for reference similarity, and MMseqs2 [28] against Virosaurus [41] for annotation. MAFFT [42] compares consensus progression. Metrics are compiled into MultiQC reports [43] and overview tables.

4 Applications

We validated nf-core/viralmetagenome using simulated HIV-1 coinfections (80-99

Reference selection significantly affects accuracy: divergent references introduce up to 187 nucleotide differences compared to similar references (Figure 2). The pipeline addresses this by automatically selecting appropriate references or using cluster representatives when no suitable external reference exists.

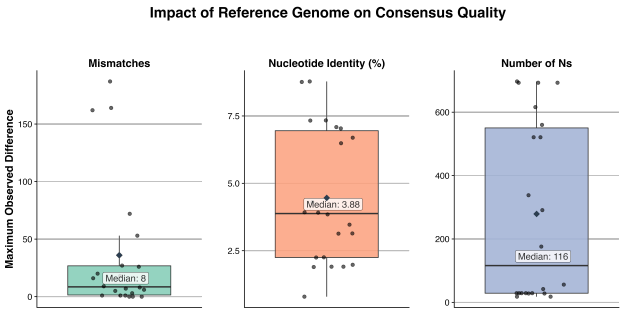

Figure 2: Boxplot of maximum observed differences between consensus sequences generated with different reference genomes during scaffolding. Mean highlighted by a diamond.

Processing 28 human viral samples required 412 CPU hours and 79GB RAM maximum. Performance correlates with database quality—comprehensive, up-to-date databases like Reference Viral Database [25] and Virosaurus [41] improve results. The pipeline targets eukaryotic viruses; bacteriophage analysis requires specialized tools like VIRify [44].

5 Conclusion

nf-core/viralmetagenome provides an automated, scalable solution for untargeted viral genome reconstruction, successfully automating the traditionally manual process through integrated *de novo* assembly, reference selection, and iterative refinement.

Our validation demonstrates broad applicability across eukaryotic viral families while ensuring reproducibility and cross-platform deployment. As viral surveillance increasingly relies on metagenomic sequencing, automated pipelines like nf-core/viralmetagenome are essential for timely pathogen identification, making viral genome reconstruction accessible without extensive bioinformatics expertise.

Acknowledgments

J.K, P.L. and L.E.K acknowledge support from the Research Foundation - Flanders (Fonds voor Wetenschappelijk Onderzoek – Vlaanderen, G005323N and G051322N, 1SH2V24N, 12X9222N).

Author Contributions

J.K. designed and implemented the pipeline, performed validation analyses, and wrote the manuscript. P.L. and L.E.K. supervised the project and provided critical feedback. The nf-core community contributed to maintaining the pipeline. All authors reviewed and approved the final manuscript.

Data Availability

The nf-core/viralmetagenome pipeline is freely available at https://github.com/nf-core/viralmetagenome. The raw data and analysis code is available on https://github.com/Joon-Klaps/nf-core-viralmetagenome-manuscript/.

Conflict of Interest

The authors declare no competing interests.

References

[1] Jasmijn A Baaijens, Amal Zine El Aabidine, Eric Rivals, and Alexander Schönhuth. De novo assembly of viral quasispecies using overlap graphs. 27:835–848.

[2] Zhi-Luo Deng, Akshay Dhingra, Adrian Fritz, Jasper Götting, Philipp C Münch, Lars Steinbrück, Thomas F Schulz, Tina Ganzenmüller, and Alice C McHardy. Evaluating assembly and variant calling software for strain-resolved analysis of large DNA viruses. 22.

[3] Dmitry Meleshko, Iman Hajirasouliha, and Anton Korobeynikov. coronaSPAdes: from biosynthetic gene clusters to RNA viral assemblies.

[4] Samuel S Shepard, Sarah Meno, Justin Bahl, Malania M Wilson, John Barnes, and Elizabeth Neuhaus. Viral deep sequencing needs an adaptive approach: IRMA, the iterative refinement meta-assembler. 17:708.

[5] Christopher Tomkins-Tinch, Simon Ye, Hayden Metsky, Irwin Jungreis, Rachel Sealfon, Xiao Yang, Kristian Andersen, Michael Lin, and Daniel Park. broadinstitute/viral-ngs: v1.14.0.

[6] Jutte J C de Vries, Julianne R Brown, Nicole Fischer, Igor A Sidorov, Sofia Morfopoulou, Jiabin Huang, Bas B Oude Munnink, Arzu Sayiner, Alihan Bulgurcu, Christophe Rodriguez, Guillaume Gricourt, Els Keyaerts, Leen Beller, Claudia Bachofen, Jakub Kubacki, Cordey

Samuel, Laubscher Florian, Schmitz Dennis, Martin Beer, Dirk Hoeper, Michael Huber, Verena Kufner, Maryam Zaheri, Aitana Lebrand, Anna Papa, Sander van Boheemen, Aloys C M Kroes, Judith Breuer, F Xavier Lopez-Labrador, and Eric C J Claas. Benchmark of thirteen bioinformatic pipelines for metagenomic virus diagnostics using datasets from clinical samples. 141:104908.

[7] Yiqiao Li, Mariana Polychronopoulou, Ine Boonen, Antonios Fikatas, Sophie Gryseels, Anne Laudisoit, Joelle Gouy de Bellocq, Bram Vrancken, Gkikas Magiorkinis, Philippe Lemey, and Magda Bletsa. Evaluation of metatranscriptomic sequencing protocols to obtain full-length RNA virus genomes from mammalian tissues. 20:e0324537.

[8] Paolo Di Tommaso, Maria Chatzou, Evan W Floden, Pablo Prieto Barja, Emilio Palumbo, and Cedric Notredame. Nextflow enables reproducible computational workflows. 35:316–319.

[9] Philip A Ewels, Alexander Peltzer, Sven Fillinger, Harshil Patel, Johannes Alneberg, Andreas Wilm, Maxime Ulysse Garcia, Paolo Di Tommaso, and Sven Nahnsen. The nf-core framework for community-curated bioinformatics pipelines. 38:276–278.

[10] D Merkel. Docker: lightweight linux containers for consistent development and deployment. 2014:2.

[11] Gregory M Kurtzer, Vanessa Sochat, and Michael W Bauer. Singularity: Scientific containers for mobility of compute. 12:e0177459.

[12] Shifu Chen, Yanqing Zhou, Yaru Chen, and Jia Gu. fastp: an ultra-fast all-in-one FASTQ preprocessor. 34:i884–i890.

[13] Anthony M Bolger, Marc Lohse, and Bjoern Usadel. Trimmomatic: a flexible trimmer for illumina sequence data. 30:2114–2120.

[14] Jeroen F J Laros. HUMID: HUMID: reference free FastQ deduplication.

[15] Tom Smith, Andreas Heger, and Ian Sudbery. UMI-tools: modeling sequencing errors in unique molecular identifiers to improve quantification accuracy. 27:491–499.

[16] Brian Bushnell. BBMap.

[17] Vito Adrian Cantu, Jeffrey Sadural, and Robert Edwards. PRINSEQ++, a multi-threaded tool for fast and efficient quality control and preprocessing of sequencing datasets.

[18] Derrick E Wood, Jennifer Lu, and Ben Langmead. Improved metagenomic analysis with kraken 2. 20:257.

[19] Peter Menzel, Kim Lee Ng, and Anders Krogh. Fast and sensitive taxonomic classification for metagenomics with kaiju. 7:11257.

[20] Brian D Ondov, Nicholas H Bergman, and Adam M Phillippy. Interactive metagenomic visualization in a web browser. 12:385.

[21] Dinghua Li, Ruibang Luo, Chi-Man Liu, Chi-Ming Leung, Hing-Fung Ting, Kunihiko Sadakane, Hiroshi Yamashita, and Tak-Wah Lam. MEGAHIT v1.0: A fast and scalable metagenome assembler driven by advanced methodologies and community practices. 102:3–11.

[22] Manfred G Grabherr, Brian J Haas, Moran Yassour, Joshua Z Levin, Dawn A Thompson, Ido Amit, Xian Adiconis, Lin Fan, Raktima Raychowdhury, Qiandong Zeng, Zehua Chen, Evan Mauceli, Nir Hacohen, Andreas Gnirke, Nicholas Rhind, Federica di Palma, Bruce W Birren, Chad Nusbaum, Kerstin Lindblad-Toh, Nir Friedman, and Aviv Regev. Full-length transcriptome assembly from RNA-seq data without a reference genome. 29:644–652.

[23] Marten Boetzer, Christiaan V Henkel, Hans J Jansen, Derek Butler, and Walter Pirovano. Scaffolding pre-assembled contigs using SSPACE. 27:578–579.

[24] S F Altschul, W Gish, W Miller, E W Myers, and D J Lipman. Basic local alignment search tool. 215:403–410.

[25] Norman Goodacre, Aisha Aljanahi, Subhiksha Nandakumar, Mike Mikailov, and Arifa S Khan. A reference viral database (RVDB) to enhance bioinformatics analysis of high-throughput sequencing for novel virus detection. 3.

[26] Weizhong Li and Adam Godzik. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. 22:1658–1659.

[27] Torbjørn Rognes, Tomáš Flouri, Ben Nichols, Christopher Quince, and Frédéric Mahé. VSEARCH: a versatile open source tool for metagenomics. 4:e2584.

[28] Martin Steinegger and Johannes Söding. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. 35:1026–1028.

[29] Kristopher Kieft, Alyssa Adams, Rauf Salamzade, Lindsay Kalan, and Karthik Anantharaman. vRhyme enables binning of viral genomes from metagenomes. 50:e83.

[30] Brian D Ondov, Gabriel J Starrett, Anna Sappington, Aleksandra Kostic, Sergey Koren, Christopher B Buck, and Adam M Phillippy. Mash screen: high-throughput sequence containment estimation for genome discovery. 20:232.

[31] Md Vasimuddin, Sanchit Misra, Heng Li, and Srinivas Aluru. Efficient architecture-aware acceleration of BWA-MEM for multicore systems. In *2019 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 314–324. IEEE.

[32] Heng Li. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM.

[33] Ben Langmead, Christopher Wilks, Valentin Antonescu, and Rone Charles. Scaling read aligners to hundreds of threads on general-purpose processors. 35:421–432.

[34] Heng Li. Minimap2: pairwise alignment for nucleotide sequences. 34:3094–3100.

[35] Nathan D Grubaugh, Karthik Gangavarapu, Joshua Quick, Nathaniel L Matteson, Jaqueline Goes De Jesus, Bradley J Main, Amanda L Tan, Lauren M Paul, Doug E Brackney, Saran Grewal, Nikos Gurfield, Koen K A Van Rompay, Sharon Isern, Scott F Michael, Lark L Coffey, Nicholas J Loman, and Kristian G Andersen. An amplicon-based sequencing framework for accurately measuring intrahost virus diversity using PrimalSeq and iVar. 20:8.

[36] Petr Danecek, James K Bonfield, Jennifer Liddle, John Marshall, Valeriu Ohan, Martin O Pollard, Andrew Whitwham, Thomas Keane, Shane A McCarthy, Robert M Davies, and Heng Li. Twelve years of SAMtools and BCFtools. 10.

[37] Irene Bassano, Vinoy K Ramachandran, Mohammad S Khalifa, Chris J Lilley, Mathew R Brown, Ronny van Aerle, Hubert Denise, William Rowe, Airey George, Edward Cairns, Claudia Wierzbicki, Natalie D Pickwell, Myles Wilson, Matthew Carlile, Nadine Holmes, Alexander Payne, Matthew Loose, Terry A Burke, Steve Paterson, Matthew J Wade, and Jasmine M S Grimsley. Evaluation of variant calling algorithms for wastewater-based epidemiology using mixed populations of SARS-CoV-2 variants in synthetic and wastewater samples.

[38] Broad Institute. Picard toolkit.

[39] Brent S Pedersen and Aaron R Quinlan. Mosdepth: quick coverage calculation for genomes and exomes. 34:867–868.

[40] Stephen Nayfach, Antonio Pedro Camargo, Frederik Schulz, Emiley Eloe-Fadrosh, Simon Roux, and Nikos C Kyrpides. CheckV assesses the quality and completeness of metagenome-assembled viral genomes. 39:578–585.

[41] Anne Gleizes, Florian Laubscher, Nicolas Guex, Christian Iseli, Thomas Junier, Samuel Cordey, Jacques Fellay, Ioannis Xenarios, Laurent Kaiser, and Philippe Le Mercier. Virosaurus a reference to explore and capture virus genetic diversity. 12.

[42] Kazutaka Katoh, Kazuharu Misawa, Kei-Ichi Kuma, and Takashi Miyata. MAFFT: a novel method for rapid multiple sequence alignment based on fast fourier transform. 30:3059–3066.

[43] Philip Ewels, Måns Magnusson, Sverker Lundin, and Max Käller. MultiQC: summarize analysis results for multiple tools and samples in a single report. 32:3047–3048.

[44] Guillermo Rangel-Pineros, Alexandre Almeida, Martin Beracochea, Ekaterina Sakharova, Manja Marz, Alejandro Reyes Muñoz, Martin Hölzer, and Robert D Finn. VIRify: an integrated detection, annotation and taxonomic classification pipeline using virus-specific protein profile hidden markov models. page 2022.08.22.504484.