APR 04, 2023

DOI:
dx.doi.org/10.17504/protocols.io.36wgqjk43vk5/v1

Protocol Citation: Remi Denise 2023. Viruses identification in metagenomes . **protocols.io** https://dx.doi.org/10.17504/protocols.io.36wgqjk43vk5/v1

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

Created: Mar 28, 2023

Last Modified: Apr 04, 2023

PROTOCOL integer ID: 79566

Viruses identification in metagenomes

Remi Denise[1]

[1]APC Microbiome Ireland & School of Microbiology, University College Cork, Co. Cork, Ireland.

Remi Denise

ABSTRACT

This protocol shows how to use this virus identification workflow (doi: 10.5281/zenodo.7778392) and some manual curation criteria for viral sequence identification in metagenomes.

BEFORE START INSTRUCTIONS

This tutorial requires Unix OS with "conda" installed. If you cannot access any Unix OS, you can use virtual machines such as VirtualBox or Vagrant in Windows. If you do not "conda" installed, you can follow the instructions here.

Install dependencies and prep test data

1 **Install dependencies**

We need the following three tools for this SOP:

- snakemake (version >= 7.25.0)
- snakedeploy (version >= 0.8.6)

First lets install mamba:

```
conda install -c conda-forge  -n base mamba
```

Second lets create new conda environment using mamba for this tutorial:

```
mamba create -c bioconda -c conda-forge --name snakemake snakemake
snakedeploy
```

Note: When you install the environment you will only need to activate the environment to run the workflow no need to install it again

2 Deploy the workflow

To deploy the workflow you need first to activate the environment

```
conda activate snakemake
```

Then you will need to deploy the workflow depending on the release you want

```
# Path were you want to deploy the workflow
mkdir /path/to/where/you/want/to/deploy/the/workflow

# Command line to deploy the workflow in this folder
snakedeploy deploy-workflow
https://github.com/rdenise/detection_virus_metagenomes
/path/to/where/you/want/to/deploy/the/workflow --tag 0.0.1
```

3 Preparation of the config file

Now that the workflow is deployed. In the folder where you created, there is a file named config/config.yaml

The important things to change in the file is:

```
# path to contigs reads folder
reads_folder: Here you write the path of the reads folder

# identifier of the sample name in the reads files (e.g. _R if the
file is named sample1_R1.fastq.gz)
reads_identifier: Here you indicate what separate the name of the
sample and the sens of the read (e.g. "_R" if the file is name
sample1_R1.fastq.gz or "_" if the file is name sample1_1.fastq.gz)
```

If you want the workflow to start after the read assembly you need to indicate the folder of your assemble contigs

```
# path to contigs folder if you already done the assembly
assemble_contigs: path of the assemble contigs

# Exention of the contig file (e.g. fasta)
contigs_ext: extension of the fasta file (e.g. fasta, fa, fna...)
```

Note: If you want to start at the assembly steps, just let the value empty with a ""

For the databases for the virus identification only ICTV, refseq viral, IMG/VR and crassphage are mandatory. If you want to add more database add your database to the list in this format

```
name_of_the_database:
    path: path/to/the/database/fasta/nucleotide/file
```

Note: everything in the config file could be change and if you miss a value for some mandatory item, the workflow will raise an error

Run the workflow

4 **Activate the snakemake environment**

Before running the workflow make sure that the conda environment is activated

```
conda activate snakemake
```

And that you are in the folder where the workflow is deployed

5 **Run snakemake workflow**

Now everything is configure, you just have to run:

```
snakemake --cores 10 --use-conda
```

The workflow will do all the steps for you and install all the needed software.

Results

6 When the workflow is done running. You will have a output folder looking like that

```
[output_name]                          <- Main results folder
├── databases                          <- Folder containing
databases used in the analysis
│   ├── checkv_db                      <- CheckV database for viral
genome completeness and contamination assessment
│   ├── contigs                        <- Folder containing contig
FASTA files
│   └── reads_trimmed                  <- Folder containing trimmed
read FASTQ files
├── logs                               <- Folder containing log
files for each analysis step
├── processed_files                    <- Folder containing
processed files resulting from the analysis
│   ├── assemblies                     <- Folder containing
assembled contigs
│   ├── blast                          <- Folder containing BLAST
output files
│   ├── bowtie2                        <- Folder containing Bowtie2
output files
│   ├── checkv                         <- Folder containing CheckV
output files
│   ├── genomad                        <- Folder containing
downloaded GenomAD data
│   ├── otu                            <- Folder containing OTU
clustering output files
│   └── samtools                       <- Folder containing
Samtools output files
├── qc                                 <- Folder containing quality
control reports
│   ├── fastqc                         <- FastQC report files for
each input read file
│   ├── multiqc_report.trimmed_data    <- MultiQC report for
trimmed reads
│   └── multiqc_report.untrimmed_data  <- MultiQC report for
untrimmed reads
└── results                            <- Folder containing final
analysis results
    ├── bacphlip_out                   <- Output files for BacPhlip
viral protein prediction tool
    ├── iphop                          <- Output files for IPHOP
prophage prediction tool
    ├── taxonomy                       <- Folder containing
taxonomy assignment output files
    └── viral_contigs                  <- Folder containing viral
contigs identified in the analysis
```