Whole genome sequencing analysis

Cedric Arisdakessian

2022-10-11

Minimum sequencing effort

How many reads do we need to sequence an organism with a relative abundance of 1%?

- Read length (Illumina): 300bp
- Genome size: 10Mbp
- Desired depth of coverage: 10X

$$\texttt{coverage} \approx \frac{\texttt{read\_length} \cdot n_{reads}}{\texttt{genome\_size}}$$

In our example, $n_{reads} = 0.3M$. Since relabund=1%, we need 100 times more reads $\Rightarrow$ 30M reads.

Note: For 16S, we need a lot less since our target region is small ($\approx 420bp$ for 16S V4 region)

Functional analysis

- Read-based: Classification of raw reads
- Assembly-based: Classification of contigs or bins
  - More accurate since sequences are longer
  - Identifies co-occurence of genes in the same genome
  - Only takes into account assembled reads
    Note: the assembly of low abundance taxa are usually very fragmented
    -> discarded when filtering short contigs
  - Detrimental if misassemblies or misbinning

Read based

Kraken2:

- Download and build database (time consuming and a bit buggy, but only done once)

```
# download
kraken2-build --download-taxonomy --use-ftp --standard \
  --db kraken2_db \

# build
kraken2-build --build \
  --db kraken2_db

# Time: can take multiple days
```

- Run kraken2 on raw reads

```
kraken2 \
  --paired --use-names \
  --report report.txt \
  --classified-out Met-1-09#.fq \
  --db kraken2_db \
  ~/data/KML/WGS/MET1-COBRE-MAUI/reads/Met-1-09_R*.fastq.gz \
  > kraken2.logs
```

Output:

- report (inspect with grep)
  format:
  https://github.com/DerrickWood/kraken2/wiki/Manual#sample-report-output-format
- reads
- logs

Contig based

WGS steps

- reads QC
- assembly
- binning
- **annotation**
  - taxonomic assignment
  - functional annotation =
    Gene calling (prodigal) + Gene/Protein annotation + Pathway enrichment analysis

WGS analysis with nf-core/mag

```
#!/bin/bash
#SBATCH --job-name=waimea-wgs
#SBATCH --partition=shared,exclusive
#SBATCH --cpus-per-task=1
#SBATCH --time=3-00:00:00
#SBATCH --mem=4G

module load lang/Anaconda3
. $(conda info --base)/etc/profile.d/conda.sh
conda activate nxf
module load tools/Singularity

nextflow run nf-core/mag -resume -profile mana \
    --input "$PWD/reads/waimea/*_R{1,2}.fastq.gz" \
    --outdir nf-mag-outputs \
    --busco_download_path "$PWD/db/busco-data" \
    --kraken2_db "$PWD/db/kraken2" \
    --skip_binning --skip_megahit
```

1. Retrieve contigs from nf-core/mag run (in Assembly/{ASSEMBLER})
2. Filter assembly (most contigs are <1kb)

```
# conda install -c bioconda seqtk
$ seqtk seq -L 10000 {path_to_contigs} > {output_name}.fasta
```

3. Make samplesheet.csv
4. Run nf-core/funcscan (revision: d8bd745)

```
nextflow run nf-core/funcscan -resume -profile docker \
  -r d8bd745 \
  --input samplesheet.csv \
  --outdir output-funcscan \
  --run_arg_screening \
  --arg_hamronization_summarizeformat interactive # or csv
```

Outputs of funcscan for ARG screening

https://nf-co.re/funcscan/dev/output