ontpipeline2 Documentation

Yan Zhou

Jul 19, 2019

Contents

CHAPTER 1

Installation

1.1 Installation

1.1.1 Anaconda Installation

Installing on Linux https://docs.anaconda.com/anaconda/install/linux/

Note:

- Anaconda is installed in `/opt` directory .

1.1.2 JDK8 Installation[9]

1. Download source pakage from Oracle. https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html
2. Extract JDK8 files to the target folder.

```
sudo mkdir /usr/lib/jvm
sudo tar -zxvf jdk-8u211-linux-x64.tar.gz -C /usr/lib/jvm
```

3. Set environment variables for JDK8.

```
sudo nano ~/.bashrc
#set oracle jdk environment
export JAVA_HOME=/usr/lib/jvm/jdk-1.8.0_211
export JRE_HOME=${JAVA_HOME}/jre
export CLASSPATH=.:${JAVA_HOME}/lib:${JRE_HOME}/lib
export PATH=${JAVA_HOME}/bin:$PATH
```

(continues on next page)

[9] Ubuntu JDK 7 / JDK8 https://www.cnblogs.com/a2211009/p/4265225.html

(continued from previous page)

```
#make changes take effect immediately
source ~/.bashrc
```

4. Set JDK8 to jdk-1.8.0_211.

```
sudo update-alternatives --install /usr/bin/java java /usr/lib/jvm/jdk-1.8.0_211/bin/
↪java 300
sudo update-alternatives --install /usr/bin/javac javac /usr/lib/jvm/jdk-1.8.0_211/
↪bin/javac 300
sudo update-alternatives --install /usr/bin/jar jar /usr/lib/jvm/jdk-1.8.0_211/bin/
↪jar 300
sudo update-alternatives --install /usr/bin/javah javah /usr/lib/jvm/jdk-1.8.0_211/
↪bin/javah 300
sudo update-alternatives --install /usr/bin/javap javap /usr/lib/jvm/jdk-1.8.0_211/
↪bin/javap 300
#set path to jdk-1.8.0_211
sudo update-alternatives --config java
```

5. Test

```
java -version
# The following messages should be showed if it works.
java version "1.8.0_211"
Java(TM) SE Runtime Environment (build 1.8.0_211-b12)
Java HotSpot(TM) 64-Bit Server VM (build 25.211-b12, mixed mode)
```

1.1.3 Guppy Installation

Guppy is a data processing toolkit that contains the Oxford Nanopore Technologies' basecalling algorithms, and several bioinformatic post-processing features.[1]

```
wget https://mirror.oxfordnanoportal.com/software/analysis/ont-guppy-cpu_3.0.3_
↪linux64.tar.gz
tar -xf ont-guppy-cpu_3.0.3_linux64.tar.gz
sudo mv ont-guppy-cpu_3.0.3_linux64 /opt/ont-guppy-cpu_3.0.3
```

1.1.4 Porechop Installation

Porechop is a tool for finding and removing adapters from Oxford Nanopore reads.[2]

```
/opt/anaconda3/bin/conda create -n porechop
source /opt/anaconda3/bin/activate porechop
conda install -c bioconda porechop
conda deactivate
```

1.1.5 NanoStat Installation

NanoStat calculates various statistics from a long read sequencing dataset in fastq, bam or albacore sequencing summary format.[3]

[1] Guppy v3.0.3 Release https://community.nanoporetech.com/posts/guppy-3-0-release
[2] Porechop https://github.com/rrwick/Porechop
[3] NanoStat https://github.com/wdecoster/nanostat

```
/opt/anaconda3/bin/conda create -n nanostat
source /opt/anaconda3/bin/activate nanostat
conda install -c bioconda nanostat
conda deactivate
```

1.1.6 NanoFilt Installation

NanoFilt filters and trims long read sequencing data.[4]

```
/opt/anaconda3/bin/conda create -n nanofilt
source /opt/anaconda3/bin/activate nanofilt
conda install -c bioconda nanofilt
conda deactivate
```

1.1.7 Unicycler Installation

Unicycler is an assembly pipeline for bacterial genomes.[5]

```
/opt/anaconda3/bin/conda create -n unicycler
source /opt/anaconda3/bin/activate unicylcer
conda install -c bioconda unicycler
conda install -c bioconda bcftools # for .vcf file
conda deactivate
```

Warning:

- Change the memory setting in Pilon is necessary or it could be failed to start[10] .

1.1.8 BUSCO Installation

BUSCO v3 provides quantitative measures for the assessment of genome assembly, gene set, and transcriptome completeness, based on evolutionarily-informed expectations of gene content from near-universal single-copy orthologs selected from OrthoDB v9.[6]

```
/opt/anaconda3/bin/conda create -n busco
source /opt/anaconda3/bin/activate busco
conda install -c bioconda busco
conda deactivate
```

1.1.9 BWA Installation

BWA is a software package for mapping low-divergent sequences against a large reference genome.[7]

[4] NanoFilt https://github.com/wdecoster/nanofilt
[5] Unicycler https://github.com/rrwick/Unicycler
[10] Pilon step runs out of error https://github.com/rrwick/Unicycler/issues/147
[6] BUSCO v3 https://busco.ezlab.org
[7] BWA https://github.com/lh3/bwa

```
/opt/anaconda3/bin/conda create -n bwa
source /opt/anaconda3/bin/activate bwa
conda install -c bioconda bwa
conda deactivate
```

1.1.10 Seqtk Installation

Seqtk is a fast and lightweight tool for processing sequences in the FASTA or FASTQ format.[8]

```
/opt/anaconda3/bin/conda create -n seqtk
source /opt/anaconda3/bin/activate seqtk
conda install -c bioconda seqtk
conda deactivate
```

1.1.11 Trimmomatic Installation

Trimmomatic is a flexible read trimming tool for Illumina NGS data.[11]

```
/opt/anaconda3/bin/conda create -n trimmomatic
source /opt/anaconda3/bin/activate trimmomatic
conda install -c bioconda trimmomatic
conda deactivate
```

[8] Seqtk https://github.com/lh3/seqtk
[11] Trimmomatic http://www.usadellab.org/cms/?page=trimmomatic

CHAPTER 2

Input File Structure

2.1 Start from Base Calling

Start the pipeline from Base calling.

```
ONT_Reads_Directory/
├── HPz800_20180821_FAJ18422_MN17776_sequencing_run_VIM4_Janina_26844_read_11_ch_171_
↪strand.fast5
├── HPz800_20180821_FAJ18422_MN17776_sequencing_run_VIM4_Janina_26844_read_11_ch_203_
↪strand.fast5
├── HPz800_20180821_FAJ18422_MN17776_sequencing_run_VIM4_Janina_26844_read_15_ch_344_
↪strand.fast5
├── HPz800_20180821_FAJ18422_MN17776_sequencing_run_VIM4_Janina_26844_read_17_ch_249_
↪strand.fast5
├── HPz800_20180821_FAJ18422_MN17776_sequencing_run_VIM4_Janina_26844_read_19_ch_397_
↪strand.fast5
└── ......

Illumina_Reads_Directory/
├── Prefix01_HQ_1.fastq.gz
├── Prefix01_HQ_2.fastq.gz
├── Prefix02_HQ_1.fastq.gz
├── Prefix02_HQ_2.fastq.gz
├── Prefix03_HQ_1.fastq.gz
├── Prefix03_HQ_2.fastq.gz
└── ......
```

Note:

- Illumina reads files naming structure for each pair: Prefix_HQ_1.fastq.gz Prefix_HQ_2.fastq.gz
- If there is without "HQ" in the file name, these Illumina reads will be trimmed.
- "Prefix" is the sample name, each pair should has its own prefix.

- "*" means arbitrarily long characters.

Warning:

- Do not use underscore ("_") in the prefix.

2.2 Start from Demultiplexing

Start the pipeline from Demultiplexing.

```
ONT_Reads_Directory/
├── fastq_runid_50a6171cadcfb6b5cb2dae4e75a0ccc05b71e3d8_0.fastq
├── fastq_runid_50a6171cadcfb6b5cb2dae4e75a0ccc05b71e3d8_1.fastq
├── fastq_runid_50a6171cadcfb6b5cb2dae4e75a0ccc05b71e3d8_2.fastq
├── fastq_runid_50a6171cadcfb6b5cb2dae4e75a0ccc05b71e3d8_3.fastq
├── fastq_runid_50a6171cadcfb6b5cb2dae4e75a0ccc05b71e3d8_4.fastq
└── ......

Illumina_Reads_Directory/
├── Prefix01_HQ_1.fastq.gz
├── Prefix01_HQ_2.fastq.gz
├── Prefix02_HQ_1.fastq.gz
├── Prefix02_HQ_2.fastq.gz
├── Prefix03_HQ_1.fastq.gz
├── Prefix03_HQ_2.fastq.gz
└── ......
```

2.3 Start from Reads Filter

Start the pipeline from Reads Filter.

```
ONT_Reads_Directory/
├── Prefix01.fastq
├── Prefix02.fastq
├── Prefix03.fastq
├── Prefix04.fastq
├── Prefix05.fastq
└── ......

Illumina_Reads_Directory/
├── Prefix01_HQ_1.fastq.gz
├── Prefix01_HQ_2.fastq.gz
├── Prefix02_HQ_1.fastq.gz
├── Prefix02_HQ_2.fastq.gz
├── Prefix03_HQ_1.fastq.gz
├── Prefix03_HQ_2.fastq.gz
└── ......
```

2.4 Start from Assembly

Start the pipeline from Assembly.

```
ONT_Reads_Directory/
├── Prefix01.fastq
├── Prefix02.fastq
├── Prefix03.fastq
├── Prefix04.fastq
├── Prefix05.fastq
└── ......

Illumina_Reads_Directory/
├── Prefix01_HQ_1.fastq.gz
├── Prefix01_HQ_2.fastq.gz
├── Prefix02_HQ_1.fastq.gz
├── Prefix02_HQ_2.fastq.gz
├── Prefix03_HQ_1.fastq.gz
├── Prefix03_HQ_2.fastq.gz
└── ......
```

2.5 Start from Polishing

Start the pipeline from Polishing.

```
ONT_Reads_Directory/
├── Prefix01.fasta
├── Prefix02.fasta
├── Prefix03.fasta
├── Prefix04.fasta
├── Prefix05.fasta
└── ......

Illumina_Reads_Directory/
├── Prefix01_HQ_1.fastq.gz
├── Prefix01_HQ_2.fastq.gz
├── Prefix02_HQ_1.fastq.gz
├── Prefix02_HQ_2.fastq.gz
├── Prefix03_HQ_1.fastq.gz
├── Prefix03_HQ_2.fastq.gz
└── ......
```

2.6 Sample Sheet

Table 1: Sample Sheet

Sample	Barcode
example1	barcode01
example2	barcode02
example3	barcode03
example4	barcode04
example5	barcode05

Note:

- The type of sample sheet file is CSV (split cell contents by comma) or TSV (split cell contents by tab).
- The format of barcode name: barcodeXX ("barcode" can be any characters, but XX must be two digits: 01,02,03,. . . ,10,11,12,. . .)

CHAPTER 3

Output File Structure

```
Output_Directory/
  ├── Analysis_{timestamp}/
  |     ├── _Basecalled/
  |     ├── _Barcodes/
  |     |     ├── barcode01/
  |     |     ├── barcode02/
  |     |     ├── barcode03/
  |     |     ├── unclassified/
  |     |     ├── Prefix01.fastq
  |     |     ├── Prefix02.fastq
  |     |     └── Prefix03.fastq
  |     ├── _AdaperTrimmedFiles/
  |     |     ├── Prefix01_trimmed.fastq
  |     |     ├── Prefix02_trimmed.fastq
  |     |     └── Prefix03_trimmed.fastq
  |     ├── _FiltedFiles/
  |     |     ├── Prefix01_filted.fastq
  |     |     ├── Prefix02_filted.fastq
  |     |     └── Prefix03_filted.fastq
  |     ├── _StatFiles/
  |     |     ├── Prefix01_trimmed_stat.txt
  |     |     ├── Prefix02_trimmed_stat.txt
  |     |     ├── Prefix03_trimmed_stat.txt
  |     |     ├── Prefix01_filted_stat.txt
  |     |     ├── Prefix02_filted_stat.txt
  |     |     └── Prefix03_filted_stat.txt
  |     ├── Prefix01_Assembly/
  |     |     ├── ...
  |     |     └── assembly.fasta
  |     ├── Prefix02_Assembly/
  |     |     ├── ...
  |     |     └── assembly.fasta
  |     ├── Prefix03_Assembly/
  |     |     ├── ...
```

(continues on next page)

(continued from previous page)

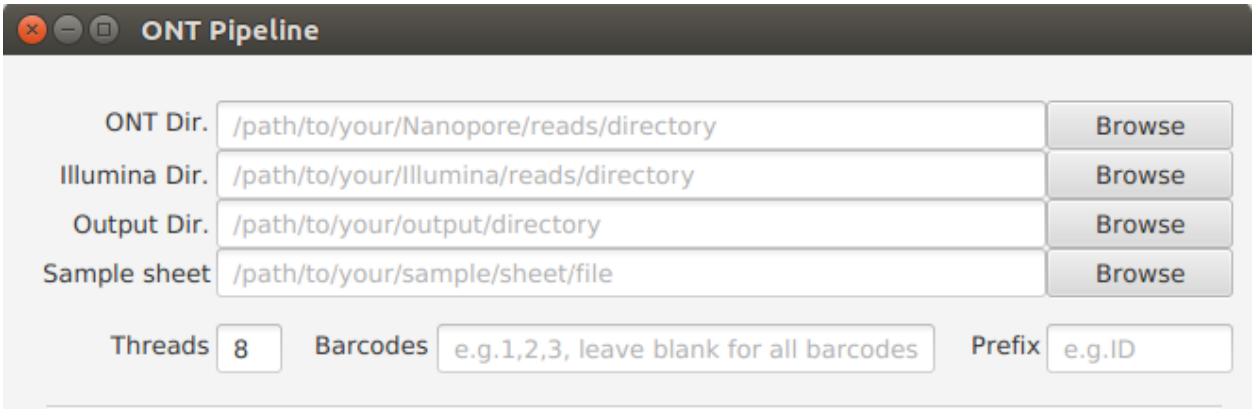
```
|     |     └── assembly.fasta
|     ├── Prefix01_Polishing/
|     |     ├── run_Prefix01_busco/
|     |     |     ├── ...
|     |     |     └── full_table_Prefix01_busco.tsv
|     |     ├── ...
|     |     └── pilon_1.fasta
|     ├── Prefix02_Polishing/
|     |     ├── run_Prefix02_busco/
|     |     |     ├── ...
|     |     |     └── full_table_Prefix02_busco.tsv
|     |     ├── ...
|     |     └── pilon_1.fasta
|     ├── Prefix03_Polishing/
|     |     ├── run_Prefix03_busco/
|     |     |     ├── ...
|     |     |     └── full_table_Prefix03_busco.tsv
|     |     ├── ...
|     |     └── pilon_1.fasta
|     └── _Logs/
|           ├── guppy_basecaller.log
|           ├── guppy_barcoder.log
|           ├── barcode_rename.log
|           ├── Prefix01_trimmed.log
|           ├── Prefix02_trimmed.log
|           ├── Prefix03_trimmed.log
|           ├── Prefix01_filted.log
|           ├── Prefix02_filted.log
|           ├── Prefix03_filted.log
|           ├── Prefix01_assembly.log
|           ├── Prefix02_assembly.log
|           ├── Prefix03_assembly.log
|           ├── Prefix01_polishing_1.log
|           ├── Prefix02_polishing_1.log
|           ├── Prefix03_polishing_1.log
|           ├── Prefix01_busco.log
|           ├── Prefix02_busco.log
|           └── Prefix03_busco.log
├── pipelineWithLoop_{timestamp}.pbs # Submitted PBS file.
└── userlog_{timestamp}.log # User given parameters.

/home/{$USER}/
├── Ont_Pipeline.e* # Error messages after the run.
└── Ont_Pipeline.o* # Output messages after the run.

/opt/ontpipeline/logs/
├── ...
└── {$USER}_error.log # Error messages if something wrong with the program.
```

CHAPTER 4

General Settings



4.1 ONT Dir. (Required)

Set the path to the Nanopore reads directory.

Note:

- **Example:** `/path/to/your/ONT/reads/directory`

4.2 Illumina Dir. (Optional/Required)

Set the path to the Illumina reads directory.

Note:

- **Example:** `/path/to/your/Illumina/reads/directory`
- Required if "hybrid-assembly" or/and "polishing" is/are used.

Warning:

- If the structure of Illumina reads filename is Prefix_{1,2}.fastq.gz (for example: ID40_1.fastq.gz, ID40_2.fastq), these Illumina reads will be trimmed.
- If the structure of Illumina reads filename is Prefix_HQ_{1,2}.fastq.gz (for example: ID40_HQ_1.fastq.gz, ID40_HQ_2.fastq), these Illumina reads will not be trimmed.

4.3 Output Dir. (Required)

Set the path to the output directory.

Note:

- **Example:** `/path/to/your/output/directory`

4.4 Sample sheet (Optional)

Set the path to the sample sheet file.

Note:

- The sample sheet file type must be CSV or TSV.

Warning:

- Underscore('_') is **not** allowed in the sample name.

4.5 Prefix (Optional)

Set the prefix for the Nanopore reads after demultiplexing.

Note:

- **Example:** ID .
- Default: barcode .

4.6 Threads (Required)

Set the number of threads/cpus to run the analysis.

Note:

- Default: 8.

4.7 Barcodes (Optional)

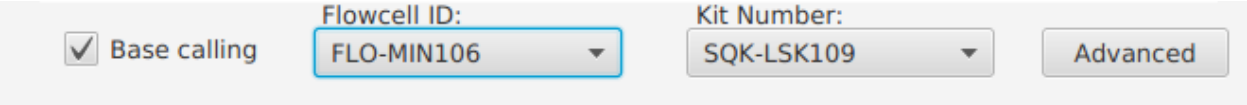
Set which barcodes will be analyzed. Put in the numbers and separate them with a comma.

Note:

- **Example:** 1,2,3,4
- If you want to analysis all barcodes, leave it blank.

CHAPTER 5

Base Calling Settings

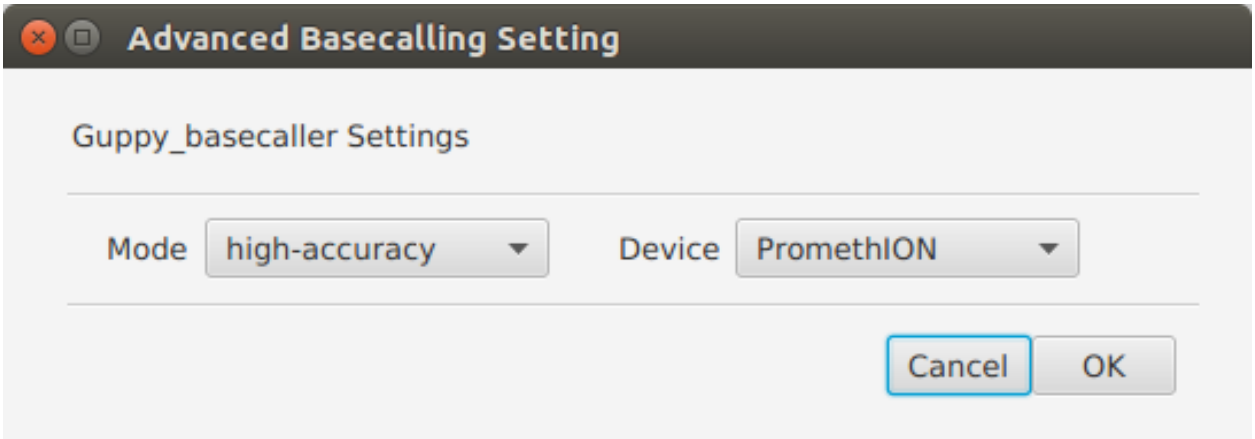


5.1 Flowcell ID[2] (Required)

Choose the Flowcell ID from the select list.

5.2 Kit Number[2] (Required)

Choose the kit number from the select list.



[2] How to configure Guppy parameters https://community.nanoporetech.com/protocols/Guppy-protocol-preRev/v/gpb_2003_v1_revg_14dec2018/how-to-configure-guppy-parameters

5.3 Mode (Required)

Set the Guppy base calling mode.

Note:

- Default: high-accuracy.

5.4 Device (Required)

Set the sequencing device.

Note:

- Default: PromethION.

5.5 cpu_threads_per_caller[1] (Default)

Note:

- Set value: 1.

5.6 records_per_fastq[2] (Default)

Note:

- Set value: 0.
- Use a single file (per worker, per run id).

5.7 recursive[2] (Default)

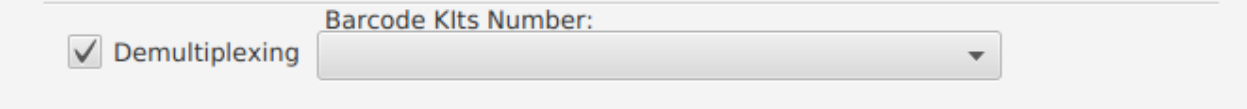
Note:

- Set value: search for input files recursively.

[1] Guppy v3.0.3 Release https://community.nanoporetech.com/posts/guppy-3-0-release

CHAPTER 6

Demultiplexing Settings



6.1 Barcode kit[1] (Optional)

Choose the barcode kit(s) from the list if used.

Note:

- If no barcode kit was used, leave it blank.
- Multiple selections possible.

6.2 records_per_fastq[1] (Default)

Note:

- Set value: 0.
- Use a single file (per worker, per run id).

[1] How to configure Guppy parameters https://community.nanoporetech.com/protocols/Guppy-protocol-preRev/v/gpb_2003_v1_revg_14dec2018/how-to-configure-guppy-parameters

6.3 recursive[1] (Default)

Note:

- Set value: search for input files recursively.

6.4 trim_barcodes[2] (Default)

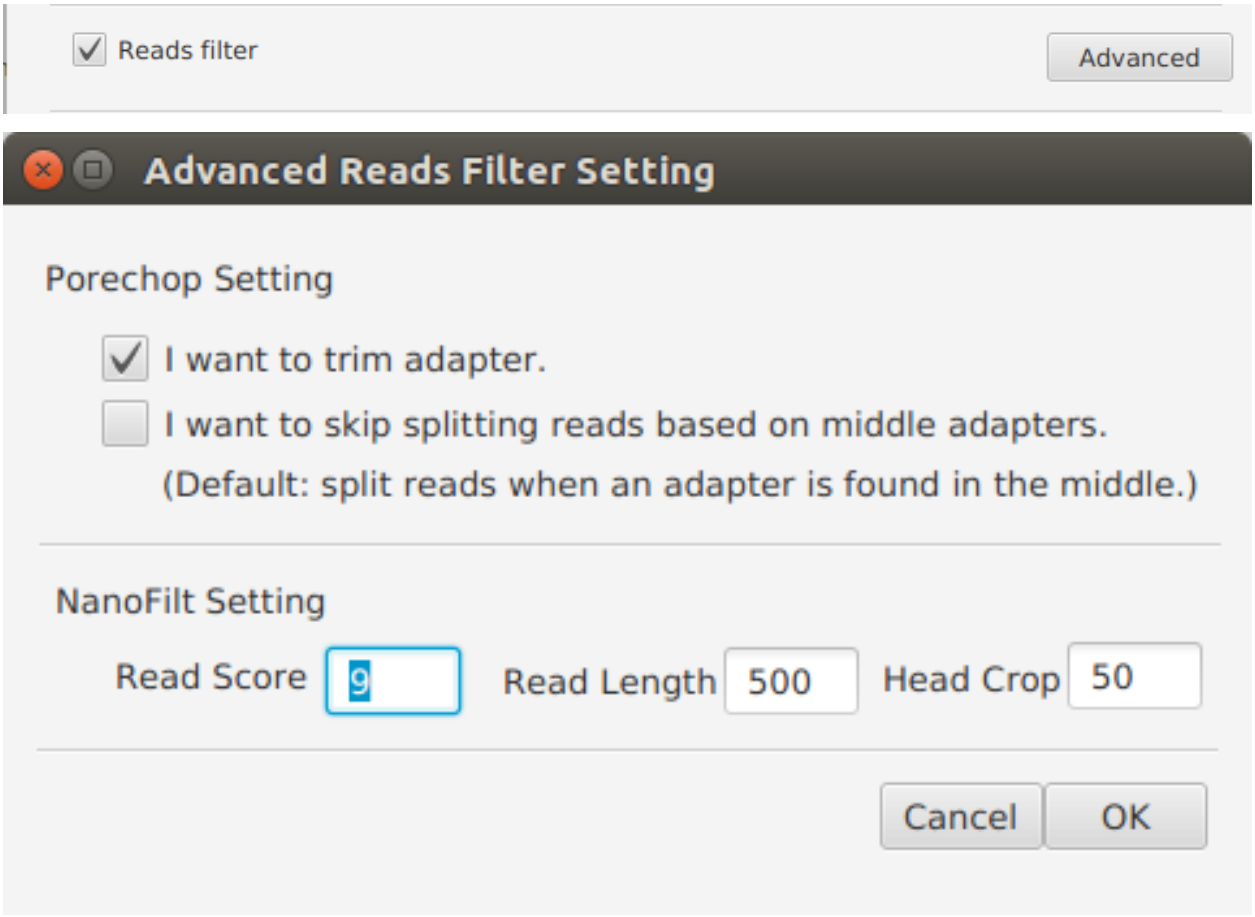
Note:

- Trim the barcodes from the output sequences in the FASTQ files.

[2] Guppy update (v3.1.5) https://community.nanoporetech.com/posts/guppy-update-v3-1-5

CHAPTER 7

Reads Filter Settings



7.1 Porechop Settings[1] (Optional)

Set Porechop options.

Note:

- Select "I want to trim adapter" if you want to use Porechop to trim adapters. Defaut: selected.
- Select "I want to skip splitting reads based on middle adapters" if you do not want to splt reads when an adapter is found in the middle. Default: not selected.

7.2 Read Score[2] (Required)

Set a minimum average read quality score to filter the reads.

Note:

- Default: 9.

7.3 Read Length [2] (Required)

Set a minimum read length to filter reads.

Note:

- Default: 500.

7.4 Head Crop[2] (Required)

Set n nucleotides to be trimmed from start of read.

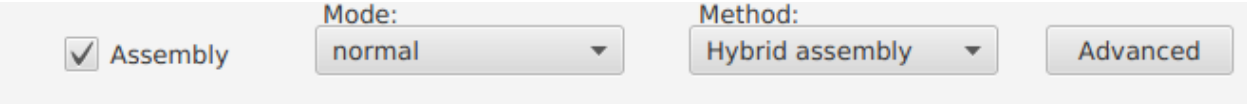
Note:

- Default: 50.

[1] Porechop https://github.com/rrwick/Porechop
[2] NanoFilt https://github.com/wdecoster/nanofilt

CHAPTER 8

Assembly Settings



8.1 Mode[1] (Required)

Choose an assembly mode.

Note:

- Conservative: Conservative mode is least likely to produce a complete assembly but has a very low risk of misassembly.
- Normal: Normal mode is intermediate regarding both completeness and misassembly risk.
- Bold: Bold mode is most likely to produce a complete assembly but carries greater risk of misassembly.
- Default: Normal.

8.2 Method[1] (Required)

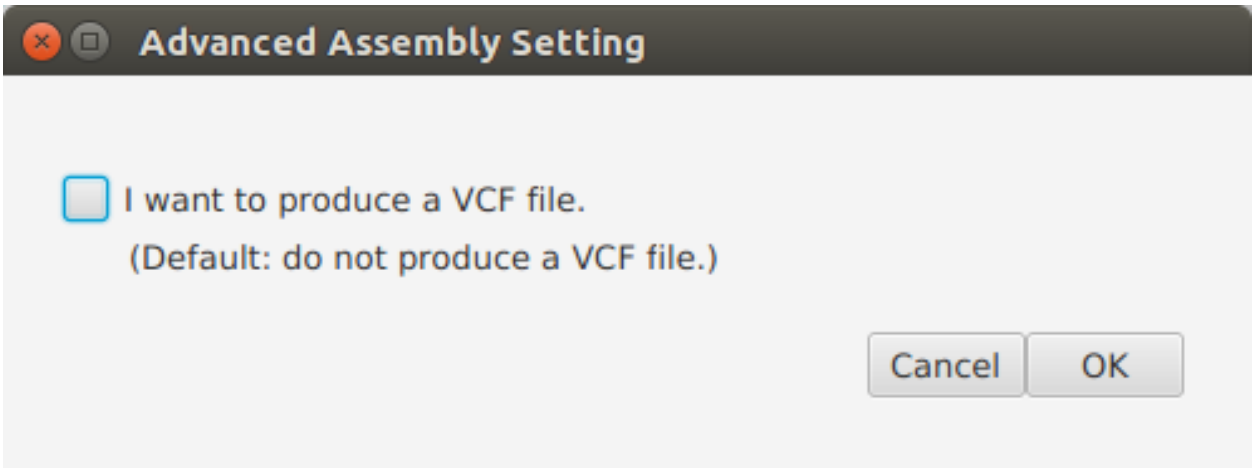
Choose an assembly method.

Note:

- Long-read-only assembly: Long-read-only assembly using only long reads.
- Hybrid assembly: Hybrid assembly using both Illumina reads and long reads.

[1] Unicycler https://github.com/rrwick/Unicycler

- Default: Hybrid assembly.



8.3 VCF[1] (Optional)

Produce a VCF by mapping the short reads to the final assembly if selected.

Note:

- Default: not selected.

8.4 Trimmomatic settings[2]

Trim Illumina reads when it is necessary.

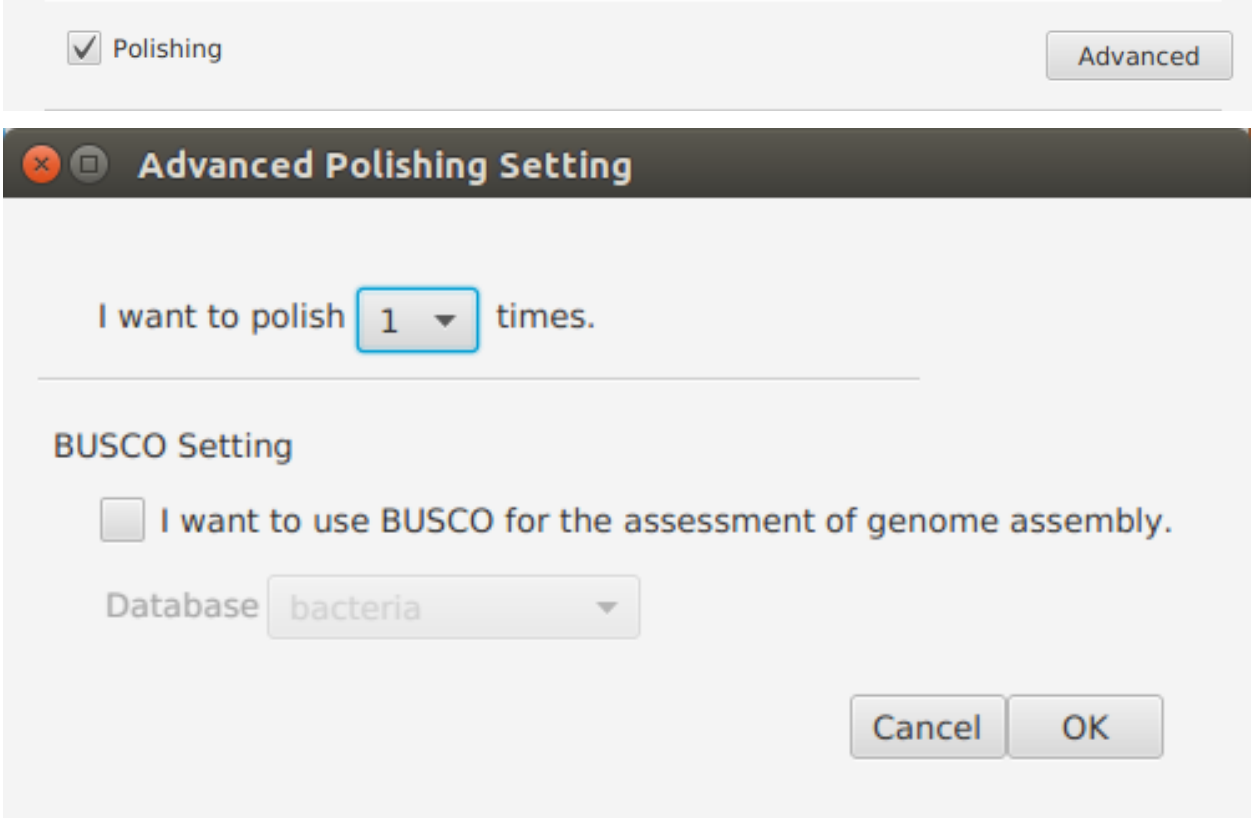
Note:

- Illumina reads will be trimmed in "Assembly" step only when all the following two conditions are satisfied:
  1. Hybrid assembly;
  2. Illumina reads filename contains no "HQ".
- Remove Illumina adapters provided in the NexteraPE-PE.fa file (provided). Initially Trimmomatic will look for seed matches (16 bases) allowing maximally 2 mismatches. These seeds will be extended and clipped if in the case of paired end reads a score of 30 is reached (about 50 bases), or in the case of single ended reads a score of 10, (about 17 bases).
- Remove leading low quality or N bases (below quality 3)
- Remove trailing low quality or N bases (below quality 3)
- Scan the read with a 4-base wide sliding window, cutting when the average quality per base drops below 15
- Drop reads which are less than 40 bases long after these steps

[2] Trimmomatic http://www.usadellab.org/cms/?page=trimmomatic

CHAPTER 9

Polishing Settings



9.1 Polishing times (Required)

Set polishing times.

Note:

- Default: 1.

9.2 BUSCO settings (Optional)

Set BUSCO options.

Note:

- Select "I want to use BUSCO for the assessement of genome assembly" if you want to use BUSCO. Default: not selected.
- Select a lineage dataset. Default: Bacteria.

9.3 Trimmomatic settings[1]

Trim Illumina reads when it is necessary.

Note:

- Illumina reads will be trimmed in "Polishing" step only when all the following two conditions are satisfied:
  1. No hybrid assembly or without assembly;
  2. Illumina reads filename contains no "HQ".
- Remove Illumina adapters provided in the NexteraPE-PE.fa file (provided). Initially Trimmomatic will look for seed matches (16 bases) allowing maximally 2 mismatches. These seeds will be extended and clipped if in the case of paired end reads a score of 30 is reached (about 50 bases), or in the case of single ended reads a score of 10, (about 17 bases).
- Remove leading low quality or N bases (below quality 3)
- Remove trailing low quality or N bases (below quality 3)
- Scan the read with a 4-base wide sliding window, cutting when the average quality per base drops below 15
- Drop reads which are less than 40 bases long after these steps

[1] Trimmomatic http://www.usadellab.org/cms/?page=trimmomatic

CHAPTER 10

FAQ

10.1 What bioinformatics tools are used?

- Guppy https://community.nanoporetech.com
- Porechop https://github.com/rrwick/Porechop
- NanoStat https://github.com/wdecoster/nanostat
- NanoFilt https://github.com/wdecoster/nanofilt
- Unicycler https://github.com/rrwick/Unicycler
- BUSCO https://busco.ezlab.org
- Seqtk https://github.com/lh3/seqtk