TIC-PIPELINE

Documentation

Version Alpha

Antonios Kioukis

June 22, 2021

Contents

1 Introduction

The importance of 16S rRNA gene amplicon profiles for understanding the influence of microbes in a variety of environments coupled with the steep reduction in sequencing costs led to a surge of microbial sequencing projects. The expanding crowd of scientists and clinicians wanting to make use of sequencing datasets can choose among a range of multipurpose software platforms, the use of which can be intimidating for non-expert users. Here we present the TIC-Pipeline, a mix of python, R, and bash tools that encode a series of well-documented choices for the downstream analysis of whole microbial studies, including the creation of ASVs, taxonomic assignments based on both known and novel taxonomic paths, a novel clustering method based on the taxonomic assignment and elementary graphing options. TIC-Pipeline is primarily a straightforward starting point for beginners, but can also be a framework for advanced users who can modify and expand the tool. As the community standards evolve, TIC-Pipeline will adapt to always represent the current state-of-the-art in microbial profiles analysis in the clear and comprehensive way allowed by the python language.

2 Organization

TIC-Pipeline is composed of 4 steps that can be run independently or as a set.

1. ASV-Creation
2. Taxonomy-Classification
3. Taxonomy-Informed-Clustering
4. Results_Processing

Running them in the given order simplifies the process as the outputs of each step are often the inputs of the next. There is also an extra folder where the original data is recommended to be placed to keep the analysis of one study in a compact and organized structure. Inside the Original-Data folder are contained the template files that served as basis for the TIC-Pipeline presentation that can be used for training purposes. Before running any script, please make sure you have read and fully understood the corresponding section of the documentation for each step.

3 Quick Reference

3.1 ASV-Creation

```
SAMPLES_PROCESS_STEP: Flag (YES/NO) enables the step.
USER_FASTQ_FOLDER: the full path of the FASTQ files.
TRIM_SCORE: the minimum quality score of the fastq read last position
     [3-20] [recommended 20]
MAXDIFF: This is the maximum mismatces during merging of reads
    allowed.
MINPCTID: Minimum \%identity of alignment. Default 90. Consider
    decreasing if you have long overlaps.
MINMERGELEN: 200 or 380 or 250 the minimun length allowed after
    pairing of reads.
MAXMERGELEN: 260 or 440 or 310 the maximun length allowed after
    pairing of reads.
FORWARD_TRIM: length of trimming at the forward side of a read
    [5-25][recommended 10].
REVERSE_TRIM: length of trimming at the reverse side of a read
    [5-25][recommended 10].
EXPECTED_ERROR_RATE: the maximum rate of expected errors allowed in
    the assemblied paired end reads 0.01 is 1\% .

# -----------------------------------

ASV_CREATION_STEP: Flag (YES/NO) enables the step.
MIN_ZOTU_SIZE: The minimum size of a ZOTU to be counted (smaller more
     sensitive but slower)
```

3.2 Taxonomy-Classification

```
ALIGNMENT_CLASSIFICATION_STEP:Flag (YES/NO) enables the step.
INPUT_FASTA_ALI_CLASS: Provide the full path of the TIC-Pipeline.
    Where the non-chimeric ZOTUs should be at 1.ASV-Creation/
    good_ZOTUS.fa
OUTPUT_FASTA_ALI_CLASS:Provide the full path of the TIC-Pipeline.
    Where the sina-aligned ZOTUs will be.
PDF_REGION_OUTPUT: The calculation of the SINA regions in order to
    view it and select which region you want to extract for the
    clustering process.

# -----------------------------------

EXTRACTION_STEP:Flag (YES/NO) enables the step.
INPUT_FASTA_EXTRACTION:/home/antonios/TIC-Pipeline/sina_output.fasta
EXTRACTION_REGION_START: start of the exraction region from the sina
    alignment. [0-49999].
EXTRACTION_REGION_END: end of the exraction region from the sina
    alignment. [0-49999].
```

```
EXTRACTION_REGION_LIMIT: the minimum number of bases that should be
   present in the sina aligned region selected, so the sequence is
   outputed. This number is multiplied by 0.80 to be more generous so
    keep this in mind.
OUTPUT_FASTA_EXTRACTION: Provide the full path of the extracted
   region ASVs with taxonomic information FASTA file.
```

3.3 Taxonomy-Informed-Clustering

```
TAXONOMIC_CLUSTERING_STEP: Flag (YES/NO) enables the step.
CLUSTERING_DIRECTORY: Full path directory to save the results of the
   taxonomic split and taxonomic clustering.
INPUT_FASTA_CLUSTERING: Full path directory of the FASTA with
   taxonomic classifications which will be used as input of the step.
FAMILY_IDENTITY: Percentage identity on the Family level. [0-
   GENERA_IDENTITY].
GENERA_IDENTITY: Percentage identity on the GENERA level. [
   FAMILY_IDENTITY-SPECIES_IDENTITY].
SPECIES_IDENTITY: Percentage identity on the GENERA level. [0-100].
```

3.4 Results_Processing

```
RESULTS_CREATION_STEP: Flag (YES/NO) enables the step.
OUTPUT_FOLDER: Full path of the directory where all results will be
   saved.
OUTPUT_ASV_FASTA_WITH_TAXONOMY: Name of the output FASTA file
   containing the ASVs along with their full taxonomic information.
OUTPUT_ASV_TABLE:Name of the output TAB file containing the ASVs
   along with their full taxonomic information and number of reads in
    each sample of the study.
```

4 Detailed Reference

4.1 ASV-Creation

This is the first step of the TIC-Pipeline.

4.2 Taxonomy-Classification

This is the second step of the TIC-Pipeline.

4.3 Taxonomically Informed Clustering

This is the third step of the TIC-Pipeline.

4.4 Results Processing

This is the forth and final step of the TIC-Pipeline.