TAXONOMIC COMPOSITION AND DIVERSITY OF BACTERIA AND ARCHAEA WITH QIIME 2

Pipeline steps:

A. Preparing the data
B. Demultiplexing of paired-end reads
C. Denoising of poor quality reads
D. Merging data (optional)
E. Taxonomy classification
F. Excluding samples (optional)
G. Diversity analysis

Considerations:

- This pipeline is made for processing multiplexed paired-end reads from the V4 16S rRNA gene.
- This pipeline assumes you have at least a basic knowledge of bioinformatics or programming.
- QIIME 2 needs to be installed and activated from Anaconda on your computer. Use the following steps to guide you (this pipeline was made with QIIME 2 version 2020.11): https://docs.qiime2.org/2020.11/install/native/
- If you have never used QIIME 2 before, I recommend you to look over some QIIME 2 concepts and steps to understand some basics of this pipeline: https://docs.qiime2.org/2020.11/tutorials/overview/, Also, some QIIME 2 tutorials: https://docs.qiime2.org/2020.11/tutorials/. QIIME 2 forum for questions: https://forum.qiime2.org/.

Input files (needed files):

- **A set or sets of multiplexed paired-end read files** (forward, reverse, and barcode files).

  NOTE: This pipeline is fixed to work with either a single folder (a single set of raw data corresponding to a single sequencing run) or multiple folders (sets of raw data of several sequencing runs, each set must be contained in a

different folder but within the same directory). Also, you don't need to modify the name of your folders with this pipeline.

- **A single or multiple metadata files** (depending on the number of folders you will work with) that you must create because it will be required by QIIME 2. These files contain important information about the samples.
- The ***"classifier.qza"*** **file,** needed for the taxonomy classification step. This file is a classifier based on the Silva database version 138 (515f-806r). It is included on https://github.com/mdelacuba/pipeline-qiime-2.
- The **seven scripts** ***"folders.sh"***, ***"q2-demux.sh"***, ***"q2-denoise.sh"***, ***"q2-merge.sh"***, ***"q2-taxonomy.sh"***, ***"q2-exclude.sh"*** and ***"q2-diversity.sh"*** included on https://github.com/mdelacuba/pipeline-qiime-2/tree/main/src. Each script will carry out each step along the pipeline.

  NOTE: If you are already a QIIME 2 user, I recommend you glance at the scripts' content to be aware of what these include. Every QIIME2 command in the scripts is headed by a description that may help you to understand the steps that will be executed.

How building the metadata files required by QIIME 2:

NOTE: Create a metadata file for each folder you will work with. Each metadata file consists of a matrix (rows and columns) of information about your samples in a given folder.

1. MANDATORY FILE(S): Within an empty text file or Excel, create a table where the first and second columns must be headed by the labels "sample-id" and "barcode-sequence", respectively. Thus, the rows of the first column must indicate the identification of your samples, and the rows of the second column must indicate the corresponding barcode sequence of each sample. These columns are indispensable to QIIME 2 for identifying each sample and barcode sequence, respectively, therefore it is mandatory to set up these columns. Moreover, you can include more information about the samples in the next columns for making later comparisons among groups of samples if that is the case.

e.g., if you have 3 folders, the content of your metadata files should look like this:

(for the first folder)

	A	B	C	D
1	sample-id	barcode-sequence	depth	Other-info
2	S1	GCCCAAGTTCAC	5	info
3	S2	GCGCCGAATCTT	10	per sample
4	S3	ATAAAGAGGAGG	10	you
5	S4	ATCCCAGCATGC	5	want
6	S5	GCTTCCAGACAA	10	to include
7				

(for the second folder)

	A	B	C	D
1	sample-id	barcode-sequence	depth	Other-info
2	S6	TTCACCTGTATC	10	Info
3	S7	CTCCAGGTCATG	12	to include
4				

(for the third folder)

	A	B	C	D
1	sample-id	barcode-sequence	depth	Other-info
2	S8	CAGGATTCGTAC	5	more
3	S9	GTGGCCTACTAC	12	info
4	S10	TTCCCTTCTCCG	10	included
5				

The name of each file must be saved according to the name of the folder as:

metadata-*foldername*.tsv

e.g., if you have 3 folders, you must create 3 metadata files:

metadata-*foldername1*.tsv

metadata-*foldername2*.tsv

metadata-*foldername3*.tsv

2. OPTIONAL FILE: If you are working with multiple folders and you will want to merge the data after the denoising step, create an additional and single metadata file compiling the sample information of all folders (remember that sample IDs and barcodes must not be repeated within each column).

   The content of this metadata file must look like this:

	A	B	C	D
1	sample-id	barcode-sequence	depth	Other-info
2	S1	GCCCAAGTTCAC	5	info
3	S2	GCGCCGAATCTT	10	per
4	S3	ATAAAGAGGAGG	10	sample
5	S4	ATCCCAGCATGC	5	you
6	S5	GCTTCCAGACAA	10	want
7	S6	TTCACCTGTATC	10	to
8	S7	CTCCAGGTCATG	12	include
9	S8	CAGGATTCGTAC	5	in
10	S9	GTGGCCTACTAC	12	this
11	S10	TTCCCTTCTCCG	10	column
12				

   This file must be saved as:

   metadata-all.tsv

3. OPTIONAL FILE: If you will want to filter specific samples after the taxonomy classification step, create an additional metadata file with only the sample(s) you want to exclude (it is no problem if the samples are from the same or different folders).

   The content of this metadata file must look like this:

	A	B	C	D
1	sample-id	barcode-sequence	depth	Other-info
2	S2	GCGCCGAATCTT	10	info
3	S3	ATAAAGAGGAGG	10	included
4				

   This file must be saved as:

   metadata-exc.tsv

Output files (resulting files):

- Typical and default QIIME 2 output files for each step and each folder during the script execution.
- The “denoise.sh” file containing the commands needed for a customizable denoising step (it will work as an input file too).
- ”*info-*” files (.txt) containing the names of the QIIME 2 output files that were correctly generated and saved for each step.
- “error-” files (.txt) containing information about possible errors produced in any QIIME 2 output file for each step.

PIPELINE

IMPORTANT: Remember to locate all the folders you will work with in the same directory. The metadata file(s), the “*classifier.qza”* file, and the seven scripts must also be located within that directory. Also, remember to activate QIIME 2 through Anaconda as:

```
conda activate qiime2-2020.11
```

A. Preparing the data:

Before starting, multiplexed paired-end read files as name_R1_001.fastq, name_R2_001.fastq, and name_I1_001.fastq files must be renamed and compressed to "forward.fastq.gz", "reverse.fastq.gz" and “barcodes.fastq.gz”, respectively (do this into each folder you will work with). It is recommendable to keep a backup of the files with their original name in another directory you want.

For declaring and loading the folders you will work with for all next steps, execute the script “folders.sh” as:

bash folders.sh or ./folders.sh

The program will ask you for the folder name(s) you will work with, you must only write it/them as the program will indicate to you (separated by spaces if there are

multiple folders). In this way, your folders will be recognized and loaded for all the next steps. If you made a mistake, you must just execute the script again.

Expected output files: none

B. Demultiplexing paired-end reads:

From now on, we will work with QIIME 2.

For the demultiplexing step, execute the script “q2-demux.sh” as:

bash q2-demux.sh or ./q2-demux.sh

Expected output files:

- ➔ emp-paired-end-sequences-*foldername*.qza
- ➔ demux-full-*foldername*.qza
- ➔ demux-details-*foldername*.qza
- ➔ demux-full-*foldername*.qzv
- ➔ info-demux.txt
- ➔ error-demux.txt (IMPORTANT: Check out this file for any error related to QIIME 2. If it is empty, it means there are no errors)

NOTE: All “.qzv” output files could be visualized on a browser page by executing the following command (this applies to the rest steps too):

qiime tools view *filename*.qzv

The output file(s) “demux-full-*foldername*.qza” is the one that will display the quality control graphics of the demultiplexed reads.

C. Denoising poor quality reads

NOTE: DADA2 is used as the denoising method in this pipeline.

IMPORTANT: After deciding which parameters you want to use to denoise poor quality reads, open the output file “denoise.sh”. As the parameters for the denoising step depend on your analysis of the quality control graphics, you must

replace the words “REPLACE” in the commands within this file with the parameters you want to fix for this step, and save the changes. This file will contain the sets of commands needed for denoising the reads of each folder independently, thereby, notice the name of the folder is the corresponding to the parameters you are entering. Also, you can delete the lines containing parameters you don’t need/want to fix for denoising. If you don’t know how to fix the DADA2 parameters within QIIME 2, please check over this document:

https://docs.qiime2.org/2020.11/plugins/available/dada2/denoise-paired/

Example:

(Before replacing)

```
#!/bin/bash
#As the parameters for the denoising step depend on your own analysis of the quality control grap
 this file with the parameters you want to fix for this step. This file contain the sets of comman
 thereby, notice the name of the folder which the commands will be executed for is the correct one
 2, please check over this document: https://docs.qiime2.org/2020.11/plugins/available/dada2/denoi
#Also, you can delete the lines containing parameters you don’t need/want to fix for denoising.

qiime dada2 denoise-paired \
--i-demultiplexed-seqs ./demux-full-foldername1.qza \
--p-trunc-len-f REPLACE \
--p-trunc-len-r REPLACE \
--p-trim-left-f REPLACE \
--p-trim-left-r REPLACE \
--p-trunc-q REPLACE \
--p-chimera-method consensus \
--p-n-threads REPLACE \
--o-table ./dada2_table-foldername1.qza \
--o-representative-sequences ./dada2_rep_seqs-foldername1.qza \
--o-denoising-stats ./dada2_stats-foldername1.qza >> info-denoise.txt 2>> error-denoise.txt


qiime dada2 denoise-paired \
--i-demultiplexed-seqs ./demux-full-foldername2.qza \
--p-trunc-len-f REPLACE \
--p-trunc-len-r REPLACE \
--p-trim-left-f REPLACE \
--p-trim-left-r REPLACE \
--p-trunc-q REPLACE \
--p-chimera-method consensus \
--p-n-threads REPLACE \
--o-table ./dada2_table-foldername2.qza \
--o-representative-sequences ./dada2_rep_seqs-foldername2.qza \
--o-denoising-stats ./dada2_stats-foldername2.qza >> info-denoise.txt 2>> error-denoise.txt


```

(After replacing)

```
#!/bin/bash
#As the parameters for the denoising step depend on your own analysis of the quality control grap
 this file with the parameters you want to fix for this step. This file contain the sets of comman
 thereby, notice the name of the folder which the commands will be executed for is the correct one
 2, please check over this document: https://docs.qiime2.org/2020.11/plugins/available/dada2/denoi
#Also, you can delete the lines containing parameters you don't need/want to fix for denoising.

qiime dada2 denoise-paired \
--i-demultiplexed-seqs ./demux-full-foldername1.qza \
--p-trunc-len-f 145 \
--p-trunc-len-r 145 \
--p-trunc-q 2 \
--p-chimera-method consensus \
--p-n-threads 8 \
--o-table ./dada2_table-foldername1.qza \
--o-representative-sequences ./dada2_rep_seqs-foldername1.qza \
--o-denoising-stats ./dada2_stats-foldername1.qza >> info-denoise.txt 2>> error-denoise.txt


qiime dada2 denoise-paired \
--i-demultiplexed-seqs ./demux-full-foldername2.qza \
--p-trunc-len-f 150 \
--p-trunc-len-r 150 \
--p-trim-left-f 10 \
--p-trim-left-r 13 \
--p-trunc-q 2 \
--p-chimera-method consensus \
--p-n-threads 8 \
--o-table ./dada2_table-foldername2.qza \
--o-representative-sequences ./dada2_rep_seqs-foldername2.qza \
--o-denoising-stats ./dada2_stats-foldername2.qza >> info-denoise.txt 2>> error-denoise.txt


```

Then, for the denoising step, execute the script “q2-denoise.sh” as:

bash q2-denoise.sh or ./q2-denoise.sh

Expected output files:

- ➔ dada2_table-*foldername*.qza
- ➔ dada2_rep_seqs-*foldername*.qza
- ➔ dada2_stats-*foldername*.qza
- ➔ dada2_table-*foldername*.qzv
- ➔ dada2_rep_seqs-*foldername*.qzv
- ➔ dada2_stats-*foldername*.qzv
- ➔ dada2_summary-*foldername*.qzv
- ➔ info-denoise.txt
- ➔ error-denoise.txt (IMPORTANT: Check out this file for any error related to QIIME 2. If it is empty, it means there is no errors)

D. Merging data (optional)

In the case you are working with multiple folders, you might want to merge the denoised data of different files, like tables and representative sequences, to a

single table and file of non-redundant representative sequences for posterior analyses. If it is not the case, omit this step.

For the merging step, execute the script “q2-merge.sh” as:

bash q2-merge.sh or ./q2-merge.sh

Expected output files:

- ➔ dada2_table-all.qza
- ➔ dada2_rep_seqs-all.qza
- ➔ dada2_table-all.qzv
- ➔ dada2_rep_seqs-all.qzv
- ➔ dada2_summary-all.qzv
- ➔ info-merge.txt
- ➔ error-merge.txt (IMPORTANT: Check out this file for any error related to QIIME 2. If it is empty, it means there is no errors)

E. Taxonomy classification:

NOTE: In this step, the Silva database version 138 (515f-806r) is used.

For the taxonomy classification step, execute the script “q2-taxonomy.sh” along with the percentage of confidence you want to work with (from 0 to 1) as:

bash q2-taxonomy.sh *%number* or ./q2-taxonomy.sh *%number*

(e.g.: ./q2-taxonomy.sh 0.8)

The execution of “q2-taxonomy.sh” will also make a filtering-features step as default after the taxonomy assignment. This consists of filtering representative sequences that were wrongly labeled as “mitochondria” and/or “chloroplast”, and/or sequences unresolved to at least phylum, labeled as “Unassigned”. You will obtain output files before and after this filtering step.

In addition, a merging step of tables containing the taxa and the sequences will also be carried out.

Expected output files:

- ➔ taxonomy80-*name*.qza
- ➔ taxonomy80-*name*.qzv
- ➔ taxa_barplot80-*name*.qzv
- ➔ taxa_barplot80-*name*-filt.qzv
- ➔ table_taxa-*name*-filt.qza
- ➔ summary-*name*-filt.qzv
- ➔ rep_seqs-*name*-filt.qza
- ➔ rep_seqs-*name*-filt.qzv
- ➔ merged-taxa-table-*name*.qzv
- ➔ merged-taxa-table-*name*-filt.qzv
- ➔ info-taxonomy.txt
- ➔ error-taxonomy.txt (IMPORTANT: Check out this file for any error related to QIIME 2. If it is empty, it means there are no errors)

F. Excluding samples (optional):

In the case you want to filter specific samples, this step will exclude the samples in the "metadata-exc.tsv" file to the results of taxonomy classification, filtering of features, merging of tables, and for the next analyses. If it is not the case, omit this step.

If you previously executed the merging samples step, now, for excluding samples execute the script "q2-exclude" as:

bash q2-exclude.sh or ./q2-exclude.sh

If you don't merged samples previously, execute the script "q2-exclude.sh" along with the name of the folder(s) where the samples you want to exclude are (separated by spaces) as:

bash q2-exclude.sh *foldername1 foldername2* or

./q2-exclude.sh foldername1 foldername2

In the case you did not merge samples before the taxonomy classification, the program will ask you whether you have a folder(s) where samples were not excluded. If it is the case, you must only write the name(s) as the program will indicate to you (separated by spaces).

Expected output files:

- ➔ table_taxa-exc-*name*.qza
- ➔ summary-exc-*name*.qzv
- ➔ summary-exc-*name*.qzv
- ➔ table_taxa-exc-filt-*name*.qza
- ➔ summary-exc-filt-*name*.qzv
- ➔ taxa_barplot80-exc-filt-*name*.qzv
- ➔ rep_seqs-exc-*name*.qza
- ➔ rep_seqs-exc-filt-*name*.qza
- ➔ rep_seqs-exc-filt-*name*.qzv
- ➔ merged-taxa-table-exc-*name*.qzv
- ➔ merged-taxa-table-exc-filt-*name*.qzv
- ➔ info-exclude.txt
- ➔ error-exclude.txt (IMPORTANT: Check out this file for any error related to QIIME 2. If it is empty, it means there is no errors)

G. Diversity analysis:

NOTE: In this step, MAFFT and FastTree are used for the alignment and phylogenetic tree construction, respectively.

For making a diversity analysis, execute the script “q2-diversity.sh” along with the number of threads you want to work with the diversity analysis, and the number of the maximum depth which you want to use for the rarefaction curves (separated by spaces), as:

bash q2-diversity.sh *n°threads max-depth* or

./q2-diversity.sh *n°threads max-depth*

(e.g.: ./q2-diversity.sh 8 10000)

Expected output files:

- ➔ ASVs-exc-filt-*name*.qza
- ➔ ASVs-exc-filt-*name*.qzv
- ➔ aligned-*name*.qza
- ➔ masked-aligned-*name*.qza
- ➔ unrooted-tree-*name*.qza
- ➔ rooted-tree-*name*.qza
- ➔ rarefaction_curves-exc-filt-*name*.qzv
- ➔ diversity-metrics-*name* (folder)
- ➔ info-diversity.txt
- ➔ error-diversity.txt (IMPORTANT: Check out this file for any error related to QIIME 2. If it is empty, it means there are no errors)

NOTE: If the exclusion of samples was not used, the output file names will omit the "-exc-" label.

Final consideration:

If you have any error in an "error-.txt" file during any step, you can solve it either by following the own QIIME 2 error information or reviewing answers for similar errors in the QIIME 2 forum (https://forum.qiime2.org/). Once you identified the solution, remove the respective "error-.txt" file and run the last executed script again. Check out the "error-.txt" file until getting no errors to continue with the next pipeline steps.