<https://github.com/ErnakovichLab/dada2_ernakovichlab>

Preliminary checklist (Part 0)

1. Amplicon length:

16S: ~400-430 bp (mean: 418 bp)

ITS: ~230-430 bp (mean: 261 bp)

2. Read length:

PE250

(16S) before trimming primers: 251 bp

(16S) after trimming primers: forward 231-234 bp, reverse 227-230 bp

* 231+227 = 458 bp
* longest amplicons are 430 bp
* that’s around 30 bp overlap -> don’t truncate sequences

3. sequencing platform: NovaSeq

4. Databases

16S: Silva

ITS: UNITE

On my laptop:

**Setup (part1):**

-step 2 was omitted because “module” does not exist and is not relevant here

-step 3 note: folder is not called dada2\_ernakovichlab but dada2\_ernakovichlab-main

-in the part “if you are running it on your own computer”:

1. tutorial was downloaded before

2. tutorial data was not downloaded because I’ll try with my own data

3. cutadapt was not installed because conda environment was created from the .yml file

4. databases (Silva&UNITE) not downloaded yet, because I don’t know which version

-part about installing packages: omitted because I use the conda environment

**Setup (part2):**

-loading libraries: package versions are slightly different (more recent) than in the tutorial Rmd

-cutadapt: nothing changed, even though I’m not on Premise.

-system2 command gave output “4.1”, so I guess cutadapt is present via the conda environment

-change: data.fp <- “/home/roland/DADA/ernakovich\_test/raw\_sequences”

(contains sequences from the comparison of 4 tomatoes from fam.exp, with 3 reps, subsampled to 50k reads) ..+note that the filenames have been truncated from the beginning, and I1 & I2 files are already removed from the folder

-change: project.fp <- “/home/roland/DADA/ernakovich\_test/project\_folder”

**On server:**

Modifying scripts: only the .R ones, not .Rmd.

Note: there was no problem with “multithread=TRUE”, unlike when I ran it on my laptop.

Deleted install script (not needed because conda environment): 00\_install\_dada2\_tutorial\_16S.R

Script 01\_:

-changed: fnFs command: pattern changed to “\_R1” (rather than “R1\_”)

-changed: fnRs command: pattern changed to “\_R2” (rather than “R2\_”)

-changed primers:

FWD <- “CCTACGGGNGGCWGCAG” #341F

REV <- “GACTACHVGGGTATCTAATCC” #805R

-changed cutadapt arguments (11/8/2022):

-added “--discard-untrimmed”

-added “-e 0.05”, “--overlap 10” (0.05% mismatch allowed = 1 base ; minimum overlap of 10 bases)

Script 03\_:

-changed filterAndTrim function: removed truncLen argument (Note: the overlap is only like 30 bp (for 16S, similar for the longest ITS amplicons), so I don’t truncate by length. Pedro also doesn’t.)

-changed filterAndTrim (11/8/2022): maxEE=c(1,1)

Script 04\_:

-changed 2 times to: “gsub(“\_16S\_R1.fastq.gz”, “”, sample.names)” (with R2 instead of R1 in the second case)

Note: for ITS this will be different, as some files are “-ITS\_R1.fastq.gz” and others are “\_ITS\_R1.fastq.gz”. maybe the best is to change all to “\_ITS\_R1.fastq.gz”.

-changed “nbases” to 10^9 on all occurrences (10 times) (this will learn the error models with 10 times more bases, it was set to 10^8) (changed on 11/8/2022)

Script 06\_:

-changed the path to Silva database

-output of: 100\*sum(seqtab.nochim)/sum(seqtab) .. -> 88.5% (outdated)

-change (because of filename extensions):

mutate(Sample = gsub("(\\\_R1\\\_)(.{1,})(\\.fastq\\.gz)","",rownames(.))) %>%

.. was changed to:

mutate(Sample = gsub("\_16S\_R1.fastq.gz","",rownames(.))) %>%

-note: Pedro included an ASV-filter in script 06\_, because in his case there were too many ASVs for R to handle. For me this was not necessary, the script ran fine on “bork.bioinformatics.nl” (48 cores, 754G RAM). Final number of ASVs is 254k for 16S.

-change: Pedro also made another change, of which I’m not yet sure if I might need it (might get an error later when importing stuff in R?):

# PEDRO'S alteration, table based on the object taxonomy and not object tax (object tax does not have ASV numbers. this is an issue in rownames eing changed into a column named " Row.names" at some point of the script.

write.table(select(taxonomy, -ASV, -ASV\_ID), file = paste0(table.fp, "/tax\_final.txt"),

sep = "\t", row.names = TRUE, col.names = NA)

Roland: also export a separate tax table with ASV ID in case it’s needed later. modified code to the following (only output filename changed):

write.table(select(taxonomy, -ASV, -ASV\_ID), file = paste0(table.fp, "/tax\_final\_with\_ASV\_ID.txt"),

sep = "\t", row.names = TRUE, col.names = NA)