Amplicon sequencing pipeline using DADA2

Christoph Schmid, June 2017

last updated: 2019-02-24

General notes

- The scripts are written to be used from command line only. In order to run the scripts locally use the Rscript command, e.g. `Rscript input.R`. They can, however, also be uploaded to a HPC cluster using e.g. qsub.

- While sequencing adapters and other artificial DNA sequences have to be removed with other tools, the DADA2 pipeline is available from a **user interface**. The user interface is available in the same folder as the R scripts. It can be run by
  ```
  python3 /path/to/installation/folder/DADA2manager.py
  ```

  or, after adding the following to your .bashrc:
  ```
  alias dada="python3 /path/to/installation/folder/DADA2manager.py"
  ```

  simply type:
  ```
  dada
  ```

- The input / output of each script is described in detail in this document.

Installation instructions

To install the convenience scripts simply download them in a folder where you have permission to do so.

Install necessary third-party software (might require admin rights on your machine):

- for the GUI: PyPubSub
  ```
  pip install Pypubsub
  ```

- for the necessary R packages, execute the install.R script from the installation folder of the convenience scripts:
  ```
  Rscript install.R
  ```

- Local installations of DADA2:
  Local installations or different versions of DADA2 can be managed by adding them manually to the versionsDADA2.txt file. This file is created at the first start of the GUI. It contains three columns: version, path and status. For adding local installations of DADA2 add a new line with version number, the installation path of the local installation and a status (“stable” or “experimental”).

- for the taxonomic databases:
  - currently the following databases can be used:
    SILVA, RDP and GreenGenes for bacteria
    Unite for fungi
  - Create a subfolder “taxonomy” in your installation path.

- In the taxonomy folder, create a subfolder for each databases you wish to use. Use folder names
  “silva” for the SILVA database,
  “rdp” for the RDP database,
  “gg” for the GreenGenes database and
  “unite” for the Unit database.
- Download the database you wish to use from the DADA2 webpage and place it in the corresponding subfolder.
- The databases should now be recognised by the GUI and R scripts.

Removing adapters and primers

This step needs to be done with tools not provided here (yet?). Possible software for this task is:

- Adapterremoval
- cutadapt
- others as preferred.

Please refer to the documentation of these tools as well as the information provided in the DADA2 documentation.

IMPORTANT NOTE: When using the GUI, file names after adapter removal need to contain identifiers for the GUI to recognise them as samples. For forward reads this is “pair1”, for reverse reads “pair2”. The samples’ names should be separated from the remaining file name with underscores “_”.

DADA2 convenience scripts

From this point we will go further using the DADA2 R-scripts. We will go through them in order of use.

input.R

The first script that we need is called **input.R.** It can be used from the GUI or the command line. From command line it understands the following options:

`-i INPUT, --input INPUT`

- This specifies an input text file with the full path to all FASTQ files that were output from removing adapters. The script will automatically identify forward and reverse reads by “pair1” / “pair2” in the filename and ignore other files.

- in case all your files are in the same folder, you can create a path file quickly by typing the following when in the directory with the FASTQs:

  ```
  ls -d $PWD/* > paths.txt
  ```

- the paths.txt file is then used as the input file for input.R

`-o OUTPUT, --output OUTPUT`

- Specifies the output directory. It does not have to exist in advance and will be used for every output.

- output files are two text files, **selectedFilesF.txt** and **selectedFilesR.txt**, containing the full file paths for forward and reverse reads, respectively. These are necessary as input for the next script.

`-p PLOT, --plot PLOT`

- This specifies for how many samples quality plots should be produced. The plots will be saved in the /qualityPlots/ directory of the output path in png format. The default number of samples is 5.
- In case you want to turn off the quality plot generation, set -p 0.

`-V VERSION, --version VERSION`

- The DADA2 version to use. **By default the latest stable release of DADA2 is used**.
- All available versions must be saved in the **versionsDADA2.txt** file to be available to both the GUI and R scripts.

`--path /installation/path/of/scripts`

- The installation folder, where the convenience scripts are stored.

example:

```
Rscript /path/to/scripts/input.R -i /path/to/sequence/data/paths.txt -o
/path/to/sequence/data/input -p 10 -V 1.10.1 --path /path/to/scripts/
```

filtering.R

This script will filter your reads according to options set by you, the user. **This is the most crucial step of the pipeline, so spend some time here and do not trust default settings.** The script combines two steps of the pipeline. The filtering and trimming step and the dereplication step. This script is also available from the user interface.

The command line version understands the following options:

`-f FORWARD, --forward FORWARD`

- This is the path to the **selectedFilesF.txt** file output in the previous step.

`-r REVERSE, --reverse REVERSE`

- This is the path to the **selectedFilesR.txt** file output in the previous step.

`-x TRUNCRF, --truncRfwd TRUNCRF`

- The basepair at which the forward reads should be truncated.

`-y TRUNCRR, --truncRrev TRUNCRR`

- The basepair at which the reverse reads should be truncated.

`--truncLfwd TRUNCLF`

- How many basepairs to cut from the beginning of the read. Numbers lower than 10 will be replaced by 10.

`--truncLrev TRUNCKLR`

- How many basepairs to cut from the beginning of the read. Numbers lower than 10 will be replaced by 10.

`-o OUTPUT, --output OUTPUT`

- The output directory for the script.
- It will produce a folder ".../filtered" in the output directory, storing the **filtered FASTQs**.
- A tabular text file reporting the amount of reads per sample before and after filtering: **filterReport.txt**

`--minLenF, --minLenR, --maxLenF, --maxLenR`

- These specify the minimum and maximum length of forward and reverse reads.

- Be aware that if you supply only one value (either for forward or for reverse reads) this value will be applied to both forward and reverse reads. This is a shortcoming of the function definition of DADA2. As an example, if you want to have a maximum length for forward but not reverse reads use `Inf`, e.g. `--maxLenF 250 --maxLenR Inf`

`-e MAXERROR, --maxError MAXERROR`

- Set the maximum expected errors per sequence

`-q QUALITY, --quality QUALITY`

- Set the quality score that is used for cutting off bad quality bases [default 2]

`-c, --compress`

- If set, compression of filter output is omitted.

`-v, --verbose`

- If set, verbose output is turned off.

`-d, --derep`

- If set, dereplication of sequences is done.

- **Dereplication is not used in the newest version of the pipeline.** To save some time, let it turned off!

`-V VERSION, --version VERSION`

- The DADA2 version to use. **By default the latest stable release of DADA2 is used**.
- All available versions must be saved in the **versionsDADA2.txt** file to be available to both the GUI and R scripts.

`--path /installation/path/of/scripts`

- The installation folder, where the convenience scripts are stored.

examples:

```
Rscript /path/to/scripts/filtering.R -f
/path/to/sequence/data/input/selectedReadsF.txt -r
/path/to/sequence/data/input/selectedReadsR.txt -x 250 -y 200 -o
/path/to/sequence/data/filtering -e 5 --path /path/to/scripts/
```

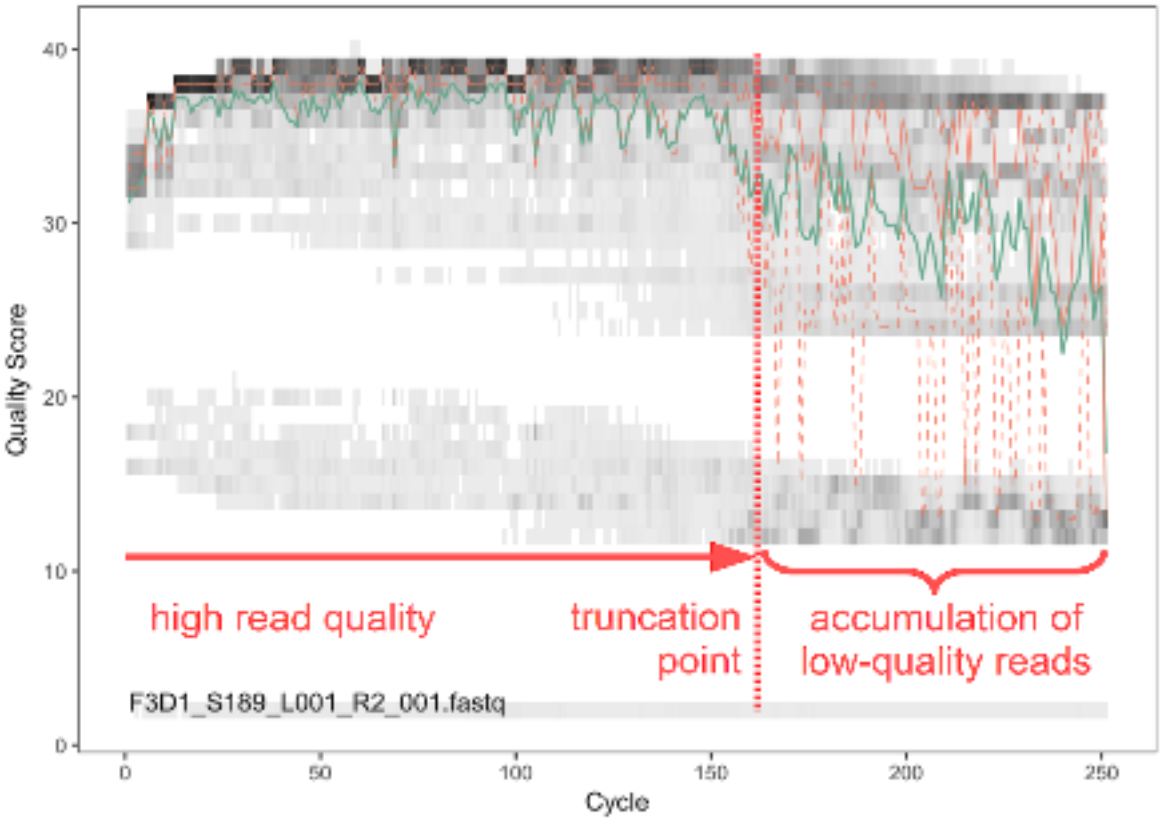
GUIDELINES FOR FILTERING

- Before you start, check the quality plots produced by input.R. **Check several plots from several samples** and try to find a common limit where to truncate the reads.

- Do the aforementioned for **forward and reverse reads separately**. You may (and probably will) have different lengths for forward and reverse reads due to worse read quality in reverse reads in Illumina sequencing.

- Remember that you need a **sufficient overlap** in order to be able to merge forward and reverse reads. The more overlap the better.

- An example quality plot is reproduced below. For this sample truncation at 160 bp seems a good idea. (The forward-reverse overlap here is almost 100 %.)



- In case too few reads pass with your chosen your cut-off points, you can also experiment with the maxError and quality options. However, bear in mind that setting these options less stringent will introduce more and more potential errors into your data.

- **A general note:** DADA2 deals quite well with low-quality reads, whereas a **too stringent filtering might influence the result fundamentally**. In test runs filter settings that let 95 % and more reads pass lead to satisfying results.

- In any case: **DOCUMENT YOUR FILTER SETTINGS CAREFULLY**

inference.R

This script will do the **de-replication of sequences, error modelling of reads and subsequent denoising of your data**. This is why we do not use the R script directly, but use qsub and an according shell script.

These computations will correct errors done by the sequencer and the amplicon sequence variants (ASVs) will be generated. As no further user input is required for further steps, the script will also merge paired-end reads, construct a raw sequence table and a chimera-cleaned sequence table.

For DADA2 versions ≥ 1.8.0 so called **pseudo-pooling** is available. This allows a **higher resolution for the detection of rare ASVs** by sharing sequence information between the samples. *However, it increases the computation time, since sample inference has to be repeated.*

The command line version understands the following options:

`-f FILTER`

- path to the filtered FASTQs output from the filtering.R script (".../filtered/")

`-p PLOT`

- This specifies for how many samples error model plots should be produced. The plots will be saved in the ".../errorPlots/" subdirectory of the output path in PNG format. The default number of samples is 5.

- In case you want to turn off the quality plot generation, set -p 0 (**NOT RECOMMENDED**)

- an error plot is reproduced below.

`-o OUTPUT`

- Specifies the output path for the script. Output files will be:
  - the error model plots in the ".../errorPlots" folder,
  - a backup file for merged Reads, **mergedReads.RData**,
  - a raw sequence table incl. chimeras, **seqTabRaw.RData**,
  - two files containing the sequence table after chimeras were removed: in .RData format (**seqTabClean.RData**) and in .CSV format (**seqTabClean_wo_taxonomy.csv**),
  - and a tabluar text file reporting the remaining amount of reads after each step: **readReport.txt**

`--seqtab`

- This stops before a sequence table is generated, i.e after merging the reads.

`--chimera`

- This stops the script before chimera removal is done.

`--pool REPEAT`

- Only available for DADA2 versions ≥ 1.8.0.
- If 0, pseudo-pooling is turned off.
- If > 0, the minimum prevalence (i.e. the amount of samples it appears in) a sequence must have in order to be considered for pooling.

`--concat`

- If set, the forward and reverse reads will be concatenated instead of merged by overlap. This is sometimes useful, when the reads are not sufficiently overlapping.

`-V VERSION, --version VERSION`

- The DADA2 version to use. **By default the latest stable release of DADA2 is used**.
- All available versions must be saved in the **versionsDADA2.txt** file to be available to both the GUI and R scripts.

`--path /installation/path/of/scripts`

- The installation folder, where the convenience scripts are stored.

example:

```
Rscript /path/to/scripts/inference.R -f
/path/to/sequence/data/filtering/filtered/ -p 20 -o
/path/to/sequence/data/denoising/ --path /path/to/scripts/
```

taxonomy.R

With this step we add taxonomic information to the sequence table generated by the previous step. This is much less time consuming than the denoising.

The command line version understands the following options:

`-i INPUT`

- path to the output file from the inference script: **seqTabClean.RData**

`-o OUTPUT`

- Specifies the output path for the script. Output files will be a backup file saving the taxonomy table, **taxonomyTable.RData**, and a CSV file with the final RSV table including taxonomic annotations, **seqTabClean_taxonomy.csv**.

`-d DATABASE`

- Specifies which database to use for taxonomic annotations. At the moment, we can choose from three databases for bacteria: `silva`, `rdp` or `gg`. For fungi we have the ITS database UNITE available: `unite`.

`--noPS`

- This option overrides the data to be saved in phyloseq format. By default, a phyloseq object is created and saved as **forPhyloseq.RData**. This file can be loaded in R and contains a phyloseq object with your sequencing data, the taxonomic information and the phylogenetic tree (if produced). However, you will need to supplement the Phyloseq object with your sample data using either Excel or R.

`-V VERSION, --version VERSION`

- The DADA2 version to use. **By default the latest stable release of DADA2 is used**.
- All available versions must be saved in the **versionsDADA2.txt** file to be available to both the GUI and R scripts.

`--path /installation/path/of/scripts`

- The installation folder, where the convenience scripts are stored.

example:

```
Rscript /path/to/scripts/taxonomy.R -i
/path/to/sequence/data/denoising/seqTabClean.RData -o
/path/to/sequence/data/taxonomy/ -d silva --path /path/to/scripts/
```