MicrobiomeSequence

Alejandra

2024-04-21

#load required packages

```
library(dada2)
```

```
## Loading required package: Rcpp
```

```
library(Biostrings)
```

```
## Warning: package 'Biostrings' was built under R version 4.3.3
```

```
## Loading required package: BiocGenerics
```

```
##
## Attaching package: 'BiocGenerics'
```

```
## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs
```

```
## The following objects are masked from 'package:base':
##
##     anyDuplicated, aperm, append, as.data.frame, basename, cbind,
##     colnames, dirname, do.call, duplicated, eval, evalq, Filter, Find,
##     get, grep, grepl, intersect, is.unsorted, lapply, Map, mapply,
##     match, mget, order, paste, pmax, pmax.int, pmin, pmin.int,
##     Position, rank, rbind, Reduce, rownames, sapply, setdiff, sort,
##     table, tapply, union, unique, unsplit, which.max, which.min
```

```
## Loading required package: S4Vectors
```

```
## Loading required package: stats4
```

```
##
## Attaching package: 'S4Vectors'
```

```
## The following object is masked from 'package:utils':
##
##     findMatches
```

```
## The following objects are masked from 'package:base':
##
##     expand.grid, I, unname
```

```
## Loading required package: IRanges
```

```
## Loading required package: XVector
```

```
## Loading required package: GenomeInfoDb
```

```
## Warning: package 'GenomeInfoDb' was built under R version 4.3.3
```

```
##
## Attaching package: 'Biostrings'

## The following object is masked from 'package:base':
##
##     strsplit
```

```
library(ShortRead)
```

```
## Loading required package: BiocParallel

## Loading required package: Rsamtools

## Loading required package: GenomicRanges

## Loading required package: GenomicAlignments

## Loading required package: SummarizedExperiment

## Loading required package: MatrixGenerics

## Loading required package: matrixStats

##
## Attaching package: 'MatrixGenerics'

## The following objects are masked from 'package:matrixStats':
##
##     colAlls, colAnyNAs, colAnys, colAvgsPerRowSet, colCollapse,
##     colCounts, colCummaxs, colCummins, colCumprods, colCumsums,
##     colDiffs, colIQRDiffs, colIQRs, colLogSumExps, colMadDiffs,
##     colMads, colMaxs, colMeans2, colMedians, colMins, colOrderStats,
##     colProds, colQuantiles, colRanges, colRanks, colSdDiffs, colSds,
##     colSums2, colTabulates, colVarDiffs, colVars, colWeightedMads,
##     colWeightedMeans, colWeightedMedians, colWeightedSds,
##     colWeightedVars, rowAlls, rowAnyNAs, rowAnys, rowAvgsPerColSet,
##     rowCollapse, rowCounts, rowCummaxs, rowCummins, rowCumprods,
##     rowCumsums, rowDiffs, rowIQRDiffs, rowIQRs, rowLogSumExps,
##     rowMadDiffs, rowMads, rowMaxs, rowMeans2, rowMedians, rowMins,
##     rowOrderStats, rowProds, rowQuantiles, rowRanges, rowRanks,
##     rowSdDiffs, rowSds, rowSums2, rowTabulates, rowVarDiffs, rowVars,
##     rowWeightedMads, rowWeightedMeans, rowWeightedMedians,
##     rowWeightedSds, rowWeightedVars

## Loading required package: Biobase

## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.

##
## Attaching package: 'Biobase'

## The following object is masked from 'package:MatrixGenerics':
##
##     rowMedians

## The following objects are masked from 'package:matrixStats':
##
##     anyMissing, rowMedians
```

```
library(phyloseq)
```

```
##
## Attaching package: 'phyloseq'
## The following object is masked from 'package:SummarizedExperiment':
##
##     distance
## The following object is masked from 'package:Biobase':
##
##     sampleNames
## The following object is masked from 'package:GenomicRanges':
##
##     distance
## The following object is masked from 'package:IRanges':
##
##     distance
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
## The following object is masked from 'package:ShortRead':
##
##     id
## The following objects are masked from 'package:GenomicAlignments':
##
##     first, last
## The following object is masked from 'package:Biobase':
##
##     combine
## The following object is masked from 'package:matrixStats':
##
##     count
## The following objects are masked from 'package:GenomicRanges':
##
##     intersect, setdiff, union
## The following objects are masked from 'package:Biostrings':
##
##     collapse, intersect, setdiff, setequal, union
## The following object is masked from 'package:GenomeInfoDb':
##
##     intersect
## The following object is masked from 'package:XVector':
##
##     slice
## The following objects are masked from 'package:IRanges':
##
##     collapse, desc, intersect, setdiff, slice, union
```

```
## The following objects are masked from 'package:S4Vectors':
##
##     first, intersect, rename, setdiff, setequal, union

## The following objects are masked from 'package:BiocGenerics':
##
##     combine, intersect, setdiff, union

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(BiMiCo)
```

#load sequences

```
path <- "sequences"
list.files(path)
```

```
##  [1] "119_S106_L001_0_L001_R1_001.fastq.gz"
##  [2] "119_S106_L001_24_L001_R2_001.fastq.gz"
##  [3] "122_S207_L001_1_L001_R1_001.fastq.gz"
##  [4] "122_S207_L001_25_L001_R2_001.fastq.gz"
##  [5] "133_S265_L001_2_L001_R1_001.fastq.gz"
##  [6] "133_S265_L001_26_L001_R2_001.fastq.gz"
##  [7] "165_S230_L001_27_L001_R2_001.fastq.gz"
##  [8] "165_S230_L001_3_L001_R1_001.fastq.gz"
##  [9] "176_S154_L001_28_L001_R2_001.fastq.gz"
## [10] "176_S154_L001_4_L001_R1_001.fastq.gz"
## [11] "208_S177_L001_29_L001_R2_001.fastq.gz"
## [12] "208_S177_L001_5_L001_R1_001.fastq.gz"
## [13] "210_S336_L001_30_L001_R2_001.fastq.gz"
## [14] "210_S336_L001_6_L001_R1_001.fastq.gz"
## [15] "220_S155_L001_31_L001_R2_001.fastq.gz"
## [16] "220_S155_L001_7_L001_R1_001.fastq.gz"
## [17] "236_S241_L001_32_L001_R2_001.fastq.gz"
## [18] "236_S241_L001_8_L001_R1_001.fastq.gz"
## [19] "252_S179_L001_33_L001_R2_001.fastq.gz"
## [20] "252_S179_L001_9_L001_R1_001.fastq.gz"
## [21] "260_S178_L001_10_L001_R1_001.fastq.gz"
## [22] "260_S178_L001_34_L001_R2_001.fastq.gz"
## [23] "281_S130_L001_11_L001_R1_001.fastq.gz"
## [24] "281_S130_L001_35_L001_R2_001.fastq.gz"
## [25] "282_S217_L001_12_L001_R1_001.fastq.gz"
## [26] "282_S217_L001_36_L001_R2_001.fastq.gz"
## [27] "306_S120_L001_13_L001_R1_001.fastq.gz"
## [28] "306_S120_L001_37_L001_R2_001.fastq.gz"
## [29] "331_S131_L001_14_L001_R1_001.fastq.gz"
## [30] "331_S131_L001_38_L001_R2_001.fastq.gz"
## [31] "332_S105_L001_15_L001_R1_001.fastq.gz"
## [32] "332_S105_L001_39_L001_R2_001.fastq.gz"
## [33] "361_S168_L001_16_L001_R1_001.fastq.gz"
## [34] "361_S168_L001_40_L001_R2_001.fastq.gz"
```

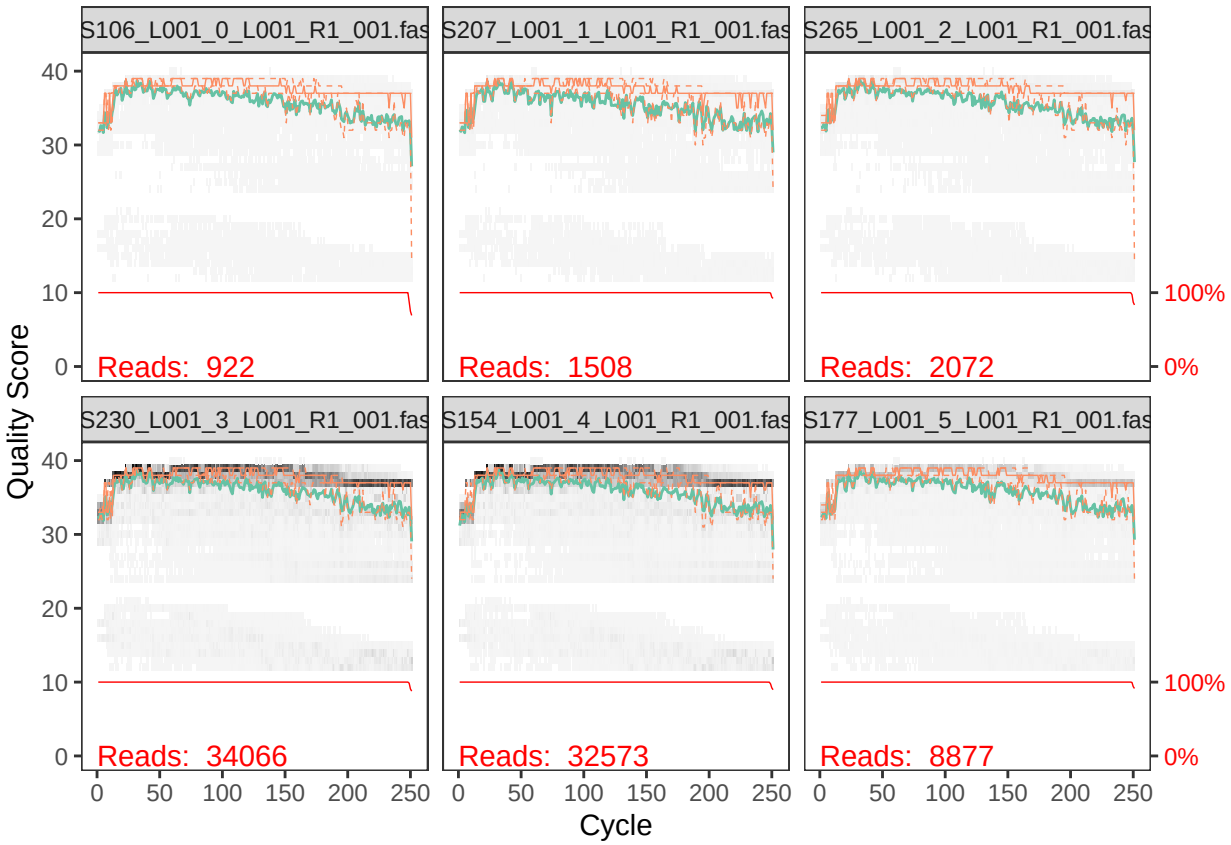
```
## [35] "368_S129_L001_17_L001_R1_001.fastq.gz"
## [36] "368_S129_L001_41_L001_R2_001.fastq.gz"
## [37] "41_S254_L001_18_L001_R1_001.fastq.gz"
## [38] "41_S254_L001_42_L001_R2_001.fastq.gz"
## [39] "50_S144_L001_19_L001_R1_001.fastq.gz"
## [40] "50_S144_L001_43_L001_R2_001.fastq.gz"
## [41] "57_S153_L001_20_L001_R1_001.fastq.gz"
## [42] "57_S153_L001_44_L001_R2_001.fastq.gz"
## [43] "72_S206_L001_21_L001_R1_001.fastq.gz"
## [44] "72_S206_L001_45_L001_R2_001.fastq.gz"
## [45] "90_S107_L001_22_L001_R1_001.fastq.gz"
## [46] "90_S107_L001_46_L001_R2_001.fastq.gz"
## [47] "94_S278_L001_23_L001_R1_001.fastq.gz"
## [48] "94_S278_L001_47_L001_R2_001.fastq.gz"
## [49] "filtered"
## [50] "MANIFEST"
## [51] "metadata.yml"
## [52] "RData"
```

#read file names

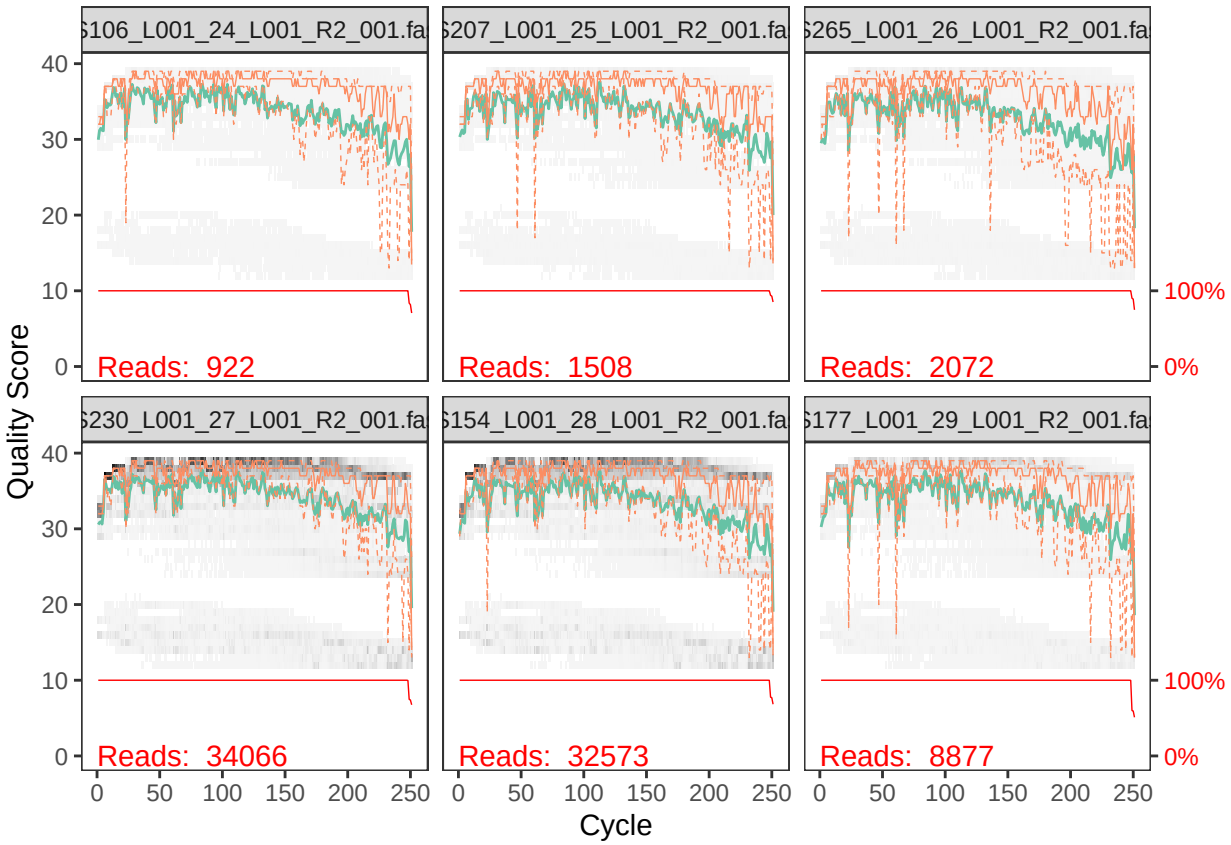
```
fnFs <- sort(list.files(path, pattern="_R1_001.fastq", full.names = TRUE))
fnRs <- sort(list.files(path, pattern="_R2_001.fastq", full.names = TRUE))
#extract file names
sample.names <- sapply(strsplit(basename(fnFs), "_"), `[`, 1)
```

#inspect file quality of forward and reverse reads

```
plotQualityProfile(fnFs[1:6])
```




```
plotQualityProfile(fnRs[1:6])
```



#filter and trim

```
#place filtered files in filtered/ subdirectory
filtFs <- file.path(path, "filtered", paste0(sample.names, "_F_filt.fastq.gz"))
filtRs <- file.path(path, "filtered", paste0(sample.names, "_R_filt.fastq.gz"))
names(filtFs) <- sample.names
names(filtRs) <- sample.names
out <- filterAndTrim(fnFs, filtFs, fnRs, filtRs, truncLen=c(130,130),
              maxN=0, maxEE=c(2,2), truncQ=2, rm.phix=TRUE,
              compress=TRUE, multithread=TRUE) # On Windows set multithread=FALSE
head(out)
```

```
##                                      reads.in reads.out
## 119_S106_L001_0_L001_R1_001.fastq.gz      922       904
## 122_S207_L001_1_L001_R1_001.fastq.gz     1508      1465
## 133_S265_L001_2_L001_R1_001.fastq.gz     2072      2024
## 165_S230_L001_3_L001_R1_001.fastq.gz    34066     33533
## 176_S154_L001_4_L001_R1_001.fastq.gz    32573     32157
## 208_S177_L001_5_L001_R1_001.fastq.gz     8877      8694
```

#learn error rates of reads

```
##learn error rates of forward and reverse reads
errF <- learnErrors(filtFs, multithread=TRUE)
```

```
## 45716970 total bases in 351669 reads from 24 samples will be used for learning the error rates.
```

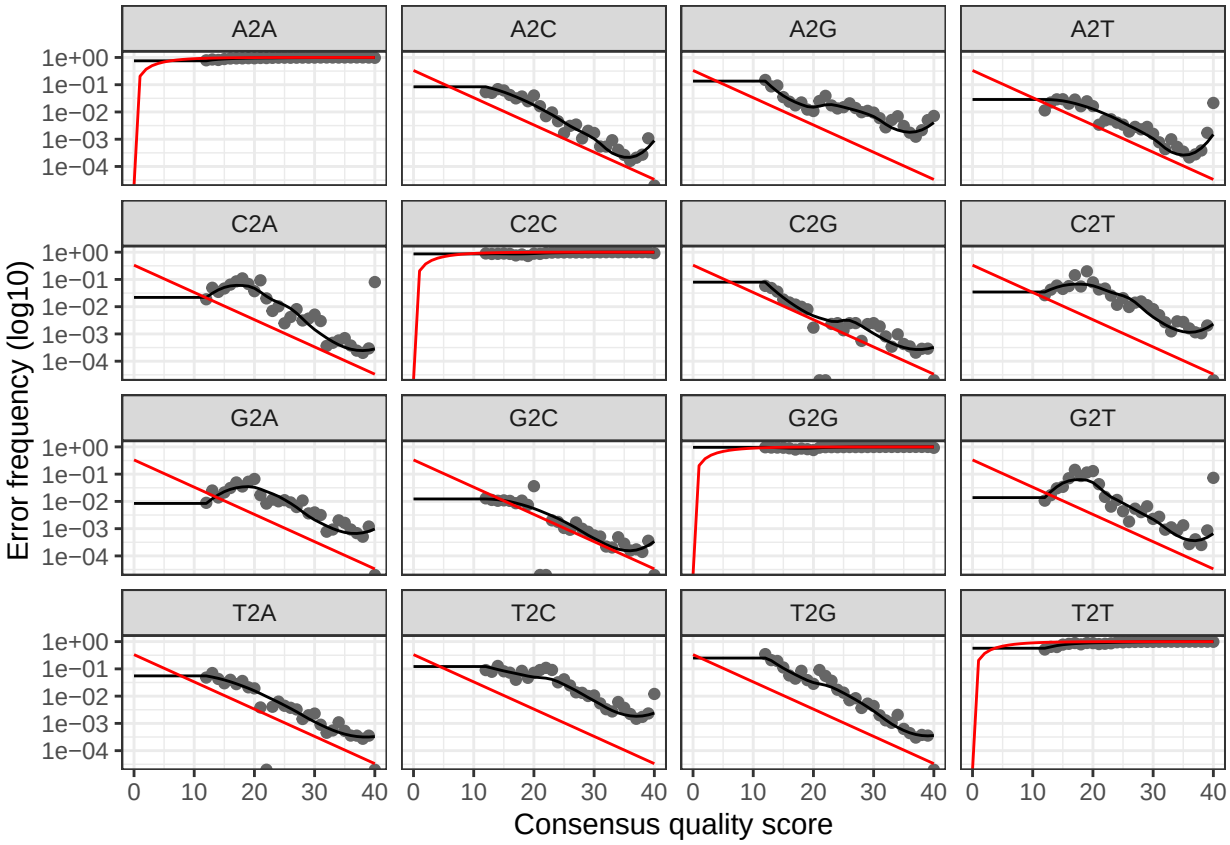
```
errR <- learnErrors(filtRs, multithread=TRUE)
```

```
## 45716970 total bases in 351669 reads from 24 samples will be used for learning the error rates.
```

#visulaize error rate

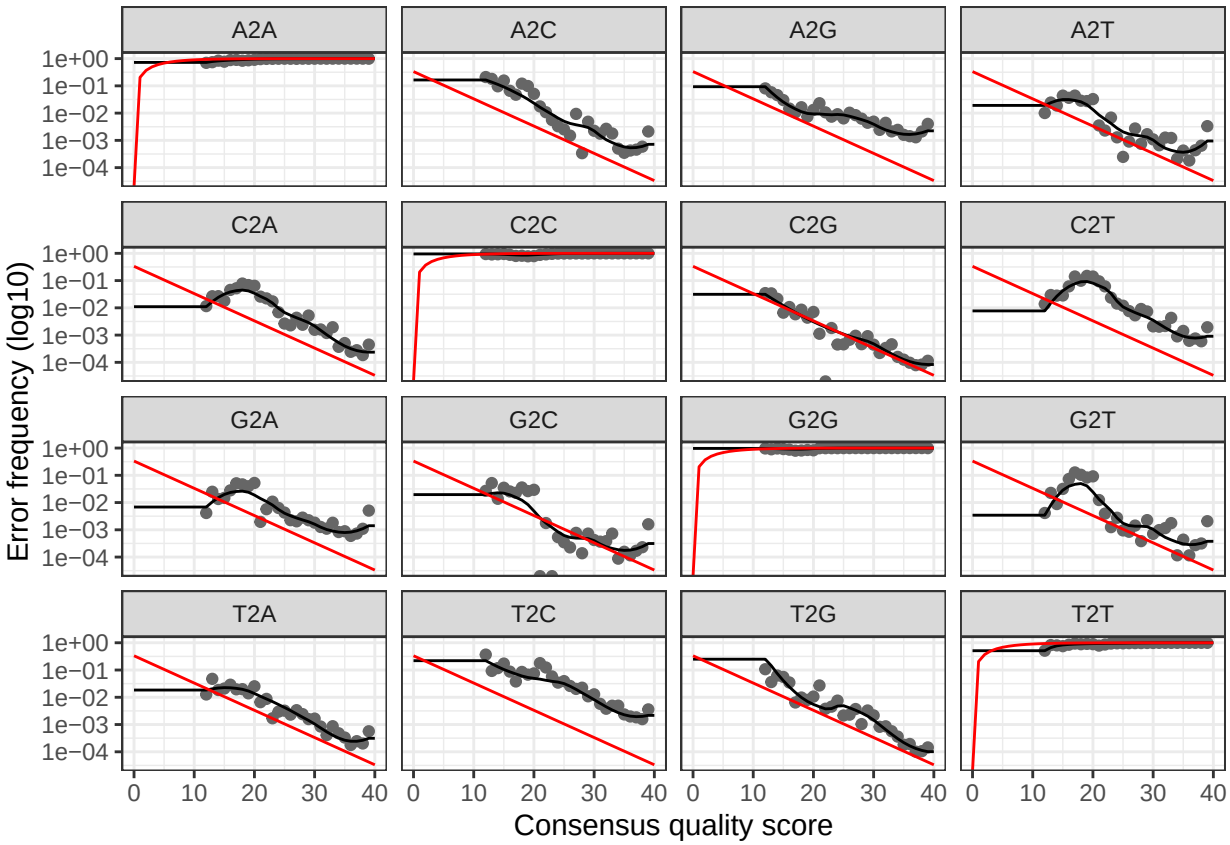
```
plotErrors(errF, nominalQ=TRUE)
```

```
## Warning in scale_y_log10(): log-10 transformation introduced infinite values.
## log-10 transformation introduced infinite values.
```




```
plotErrors(errR, nominalQ=TRUE)
```

```
## Warning in scale_y_log10(): log-10 transformation introduced infinite values.
## log-10 transformation introduced infinite values.
```



#will take reads and show how many sequences/species are in the sample

```
dadaFs <- dada(filtFs, err=errF, multithread=TRUE)
```

```
## Sample 1 - 904 reads in 268 unique sequences.
## Sample 2 - 1465 reads in 485 unique sequences.
## Sample 3 - 2024 reads in 558 unique sequences.
## Sample 4 - 33533 reads in 6505 unique sequences.
## Sample 5 - 32157 reads in 6320 unique sequences.
## Sample 6 - 8694 reads in 1825 unique sequences.
## Sample 7 - 5624 reads in 1412 unique sequences.
## Sample 8 - 45361 reads in 7892 unique sequences.
## Sample 9 - 31186 reads in 5343 unique sequences.
## Sample 10 - 790 reads in 225 unique sequences.
## Sample 11 - 3004 reads in 702 unique sequences.
## Sample 12 - 22709 reads in 4674 unique sequences.
## Sample 13 - 14077 reads in 2979 unique sequences.
## Sample 14 - 2956 reads in 754 unique sequences.
## Sample 15 - 8605 reads in 1976 unique sequences.
## Sample 16 - 1049 reads in 317 unique sequences.
## Sample 17 - 68637 reads in 10206 unique sequences.
## Sample 18 - 7564 reads in 1861 unique sequences.
## Sample 19 - 654 reads in 225 unique sequences.
## Sample 20 - 14783 reads in 2926 unique sequences.
## Sample 21 - 10037 reads in 2442 unique sequences.
## Sample 22 - 22048 reads in 3903 unique sequences.
## Sample 23 - 6519 reads in 1383 unique sequences.
```

```
## Sample 24 - 7289 reads in 1553 unique sequences.
```

```
dadaRs <- dada(filtRs, err=errR, multithread=TRUE)
```

```
## Sample 1 - 904 reads in 303 unique sequences.
## Sample 2 - 1465 reads in 551 unique sequences.
## Sample 3 - 2024 reads in 825 unique sequences.
## Sample 4 - 33533 reads in 8017 unique sequences.
## Sample 5 - 32157 reads in 9639 unique sequences.
## Sample 6 - 8694 reads in 2410 unique sequences.
## Sample 7 - 5624 reads in 1978 unique sequences.
## Sample 8 - 45361 reads in 13126 unique sequences.
## Sample 9 - 31186 reads in 8162 unique sequences.
## Sample 10 - 790 reads in 380 unique sequences.
## Sample 11 - 3004 reads in 948 unique sequences.
## Sample 12 - 22709 reads in 6037 unique sequences.
## Sample 13 - 14077 reads in 4279 unique sequences.
## Sample 14 - 2956 reads in 1066 unique sequences.
## Sample 15 - 8605 reads in 2673 unique sequences.
## Sample 16 - 1049 reads in 366 unique sequences.
## Sample 17 - 68637 reads in 15181 unique sequences.
## Sample 18 - 7564 reads in 2448 unique sequences.
## Sample 19 - 654 reads in 303 unique sequences.
## Sample 20 - 14783 reads in 4189 unique sequences.
## Sample 21 - 10037 reads in 3417 unique sequences.
## Sample 22 - 22048 reads in 4822 unique sequences.
## Sample 23 - 6519 reads in 2018 unique sequences.
## Sample 24 - 7289 reads in 2079 unique sequences.
```

#merge paired reads

```
mergers <- mergePairs(dadaFs, filtFs, dadaRs, filtRs, verbose=TRUE)
```

```
## 229 paired-reads (in 6 unique pairings) successfully merged out of 849 (in 52 pairings) input.
## 95 paired-reads (in 6 unique pairings) successfully merged out of 1356 (in 86 pairings) input.
## 404 paired-reads (in 17 unique pairings) successfully merged out of 1905 (in 119 pairings) input.
## 4458 paired-reads (in 59 unique pairings) successfully merged out of 32823 (in 786 pairings) input.
## 3470 paired-reads (in 41 unique pairings) successfully merged out of 31785 (in 527 pairings) input.
## 647 paired-reads (in 15 unique pairings) successfully merged out of 8557 (in 187 pairings) input.
## 261 paired-reads (in 6 unique pairings) successfully merged out of 5534 (in 153 pairings) input.
## 3681 paired-reads (in 26 unique pairings) successfully merged out of 44948 (in 530 pairings) input.
## 3038 paired-reads (in 32 unique pairings) successfully merged out of 30726 (in 473 pairings) input.
## 74 paired-reads (in 6 unique pairings) successfully merged out of 731 (in 52 pairings) input.
## 208 paired-reads (in 10 unique pairings) successfully merged out of 2923 (in 98 pairings) input.
## 1875 paired-reads (in 29 unique pairings) successfully merged out of 22483 (in 408 pairings) input.
## 2408 paired-reads (in 22 unique pairings) successfully merged out of 13808 (in 399 pairings) input.
## 640 paired-reads (in 16 unique pairings) successfully merged out of 2853 (in 117 pairings) input.
## 1136 paired-reads (in 22 unique pairings) successfully merged out of 8384 (in 228 pairings) input.
```

```
## 134 paired-reads (in 8 unique pairings) successfully merged out of 972 (in 59 pairings) input.
## 4026 paired-reads (in 34 unique pairings) successfully merged out of 68067 (in 608 pairings) input.
## 1490 paired-reads (in 20 unique pairings) successfully merged out of 7357 (in 303 pairings) input.
## 12 paired-reads (in 1 unique pairings) successfully merged out of 570 (in 28 pairings) input.
## 4468 paired-reads (in 41 unique pairings) successfully merged out of 14480 (in 332 pairings) input.
## 840 paired-reads (in 15 unique pairings) successfully merged out of 9795 (in 304 pairings) input.
## 1224 paired-reads (in 23 unique pairings) successfully merged out of 21768 (in 380 pairings) input.
## 2090 paired-reads (in 19 unique pairings) successfully merged out of 6335 (in 151 pairings) input.
## 655 paired-reads (in 16 unique pairings) successfully merged out of 7170 (in 172 pairings) input.
```

```
# Inspect the merger data.frame from the first sample
head(mergers[[1]])
```

```
##
## 1                 TACGTAAAAGACAAGTGTTATTCATCTTTAATAGGTTTAAAGGGTACCTAGACGGTATTATTAGCCCAAAAAAGGGTACGATT
## 2            CACAAGTAAGATTAGTGTTATTCATCTTTATTAGGTTTAAAGGGTACCTAGACGGCAAAAGCAACTTCTAAAAAGTATATCTTTGCTA
## 5                 TACGTAAAAGACAAGTGTTATTCATCTTTAATAGGTTTAAAGGGTACCTAGACGGTATTATTAGCCCAAAAAAGGGTACAATT
## 13           CACAAGTAAGATAAGTGTTATTCATCTTCATTAGGTTTAAGGGGTACCTAGACGGCATTTTTCACTTATTGAAAGAAAATAATTGCTA
## 18 TACGAGTAAGACTAGTGTTAGTCATCTTCATTAGGTTTAAAGGGTACCTAGACGGTATTTAGACCACAGTATAACACTGTTAGGTACATTAATACTAC
## 21           TACAAGTAAGACTAGTGTTATTCATCTTTATTAGGTTTAAAGGGTACCTAGACAGTATTTCTAGCCTCAAAAGGGAACAGATTTACTA
##    abundance forward reverse nmatch nmismatch nindel prefer accept
## 1         68       1       4     38         0      0      1   TRUE
## 2         63       2       3     33         0      0      1   TRUE
## 5         52      30       5     38         0      0      1   TRUE
## 13        22      11      12     33         0      0      1   TRUE
## 18        13      14      13     23         0      0      1   TRUE
## 21        11      16      20     33         0      0      1   TRUE
```

#construct sequence table to see how many sequences are present and length

```
seqtab <- makeSequenceTable(mergers)
dim(seqtab)
```

```
## [1]  24 221
```

```
# Inspect distribution of sequence lengths
table(nchar(getSequences(seqtab)))
```

```
##
## 180 201 203 204 215 216 220 221 222 223 224 225 226 227 228 229 231 235 236 237
##   1   1   2   1   1   1  14  31  18  35  19   7  12  35   5  10   2   1   2   4
## 238 239 240 243 244 247
##   1   1   2   1  13   1
```

#remove chimeras (two sperate reads that got smashed together)

```
seqtab.nochim <- removeBimeraDenovo(seqtab, method="consensus", multithread=TRUE, verbose=TRUE)
```

```
## Identified 4 bimeras out of 221 input sequences.
```

```
dim(seqtab.nochim)
```

```
## [1]  24 217
```

#track reads (which step lost reads)

```
getN <- function(x) sum(getUniques(x))
track <- cbind(out, sapply(dadaFs, getN), sapply(dadaRs, getN), sapply(mergers, getN), rowSums(seqtab.no
colnames(track) <- c("input", "filtered", "denoisedF", "denoisedR", "merged", "nonchim")
rownames(track) <- sample.names
head(track)
```

```
##     input filtered denoisedF denoisedR merged nonchim
## 119   922      904       856       876    229     229
## 122  1508     1465      1397      1403     95      95
## 133  2072     2024      1974      1941    404     404
## 165 34066    33533     33114     33168   4458    4458
## 176 32573    32157     32004     31915   3470    3465
## 208  8877     8694      8597      8627    647     647
```

#save setab.nochim as an R file

```
save(seqtab.nochim, file= "RData/seqtab.nochim.RData")
```

#load seqtab.nochim

```
load("RData/seqtab.nochim.RData")
```

#asign taxonomy

```
taxa <- assignTaxonomy(seqtab.nochim, "silva_nr99_v138.1_wSpecies_train_set.fa.gz", multithread=TRUE)
save(taxa, file = "RData/taxa.RData")
```

#load taxa and seqtab.nochim

```
load("RData/taxa.RData")
load("RData/seqtab.nochim.RData")
```