RDPutils Workflows for USEARCH Output

John Quensen

October 23, 2018

Introduction

Illumina's MiSeq sequencing platform is capable of generating much larger data sets than Roche's older 454 pyrosequencing method. Processing these large data sets using RDPTools' complete linkage clustering method is impractical because of the large computer memory and long run times required. The `UPARSE` pipeline implemented in `USEARCH` is an attractive alternative method because of its much greater speed and smaller memory requirements. `RDPutils` version 1.3 added several functions for importing `USEARCH/UPARSE` output as `phyloseq` otu and taxonomy tables. The otu sequences may also be imported as reference sequences.

`USEARCH` is available as a free 32-bit version and a paid 64-bit version. The 32-bit version has a 4 GB memory limit which prevents dereplicating and clustering data from larger experiments. Presently the latest 64-bit version available on MSU's HPCC is 8.1, so I have written the script below to be run with`USEARCH 8.1` with one exception. For comparison, I have included classification using the `sintax` command new beginning with `USEARCH` version 9. The memory requirement for classification is small enough that the 32-bit version of the program can be used. Beginning with merged and trimmed sequences from all samples catenated together into `all_samples.fasta`, an example script for processing 16S data on MSU's HPCC is:

```
#!/bin/bash

# Set paths.
usearch81=/mnt/research/rdp/public/thirdParty/usearch8.1.1831_i86linux64
usearch91=/mnt/home/quensenj/usearch9/usearch9.1.13_i86linux32
infernal_dir=/mnt/research/rdp/public/thirdParty/infernal-1.1/src
cm_model_dir=/mnt/research/rdp/public/fungene_pipeline/resources/RRNA_16S_BACTERIA
utax_rdp_16s=~/resources/utax_16s_ref.udb
sintax_rdp_16s=~/resources/sintax_rdp_16s.udb

# Load modules.
module load FastTree

# Dereplicate.
$usearch81 -derep_fulllength all_samples.fasta -sizeout -fastaout uniques.fa \
  -relabel Uniq

# Cluster the dereplicated sequences - otus_03.fa are the representative sequences.
$usearch81 -cluster_otus uniques.fa -minsize 2 -otus otus_03.fa -relabel Otu \
  -otu_radius_pct 3.0

# Make the otu table.
$usearch81 -usearch_global all_samples.fasta -db otus_03.fa -strand plus -id 0.97 \
  -otutabout otu_03_table_only.txt -biomout otu_03_table_only.json

# Make the taxonomy table.
$usearch81 -utax otus_03.fa -db $utax_rdp_16s -strand both \
  -utaxout utax_tax_table_03.txt  -utax_cutoff 0.8
```

```
# With USEARCH 8.1 it is possible to output the OTU table in biom format,
# and it is also possible to includes taxonomy with the otu table and biom file
# if taxonomy is first added to the representative sequences.
# Add taxonomy to the representative sequences.
$usearch81 -utax otus_03.fa -db  $utax_rdp_16s -strand both -fastaout otus_tax_03.fa \
  -utax_cutoff 0.8

# Make otu table with taxonomy and biom file with otu and taxonomy tables.
$usearch81 -usearch_global all_samples.fasta -db otus_tax_03.fa -strand plus -id 0.97 \
 -otutabout otu_03_tax_table.txt -biomout otu_03_tax_table.json

# With USEARCH 9.0 and later, taxonomy can also be assigned with the sintax function.
$usearch91 -sintax otus_03.fa -db $sintax_rdp_16s -strand both \
  -tabbedout sintax_tax_table.txt -sintax_cutoff 0.8

# Assign taxonomy with the RDP Classifier.
java -Xmx2g -jar $RDPTools_dir/classifier.jar classify -g 16srrna \
  -c 0.8 -f fixrank -o rdp_classified_03.txt otus_03.fa

# Align the representative sequences. AFA is aligned fasta format.
$infernal_dir/cmalign -g --noprob --outformat AFA --dnaout -o aligned_otus_03.fasta \
  $cm_model_dir/model.cm otus_03.fa

# Tree the aligned representatve sequences.
FastTree -nt -gtr < aligned_otus_03.fasta > usearch_03_tree.nwk
```

Importing OTU Tables

The OTU table can be imported in several ways. The file containing only the OTU table can be read in with the base `read.table` function, but the argument `comment.char = ""` must be included in order to ignore the leading `#` in the first line. It can then be converted to a `phyloseq otu_table` with `phyloseq`'s `otu_table` function.

```
suppressWarnings(suppressMessages(library(phyloseq)))
suppressWarnings(suppressMessages(library(RDPutils)))
suppressWarnings(suppressMessages(library(Biostrings)))
otu.file <- system.file("extdata", "otu_03_table_only.txt",
                        package="RDPutils")
otu <- read.table(file = otu.file, header = TRUE, row.names = 1,
                  sep = '\t', comment.char = "")
head(otu)
```

```
##       Sample_1 Sample_2 Sample_3 Sample_4
## Otu39        1        2        1        0
## Otu2         6        7       11       12
## Otu11        1        3        2        1
## Otu4         9        5        3        3
## Otu1        13       12        2       10
## Otu5        10        3       16       12
```

```
my_otu <- otu_table(otu, taxa_are_rows = TRUE, errorIfNULL = TRUE)
class(my_otu)
```

```
## [1] "otu_table"
```

```
## attr(,"package")
## [1] "phyloseq"
```

The same file can also be read in with the `RDPutils` function `import_otutab_taxa` function.

```
otu <- import_otutab_taxa(in_file = otu.file)
head(otu)
```

```
## OTU Table:          [6 taxa and 4 samples]
##                      taxa are rows
##       Sample_1 Sample_2 Sample_3 Sample_4
## Otu39        1        2        1        0
## Otu2         6        7       11       12
## Otu11        1        3        2        1
## Otu4         9        5        3        3
## Otu1        13       12        2       10
## Otu5        10        3       16       12
```

```
class(otu)
```

```
## [1] "otu_table"
## attr(,"package")
## [1] "phyloseq"
```

The `biom` file containing only the OTU table can be read in with `phyloseq`'s `import_biom` function provided a modified `parseFunction` is given. The `USEARCH` taxonomy fields should be broken on commas.

```
parse_taxonomy_usearch <- function (char.vec){
  parse_taxonomy_default(strsplit(char.vec, ",", TRUE)[[1]])
}


biom.file <- system.file("extdata", "otu_03_table_only.json",
                         package="RDPutils")

otu <- import_biom(BIOMfilename = biom.file, parseFunction = parse_taxonomy_usearch)
head(otu)
```

```
## OTU Table:          [6 taxa and 4 samples]
##                      taxa are rows
##       Sample_1 Sample_2 Sample_3 Sample_4
## Otu39        1        2        1        0
## Otu2         6        7       11       12
## Otu11        1        3        2        1
## Otu4         9        5        3        3
## Otu1        13       12        2       10
## Otu5        10        3       16       12
```

```
class(otu)
```

```
## [1] "otu_table"
## attr(,"package")
## [1] "phyloseq"
```

Importing Taxonomy Tables

RDP Classifier

Import the taxonomy table made with the RDP classifier with the `make_tax_table` function. It returns a `phyloseq tax_table object`. The confidence level is chosen on import. It is 0.5 by default.

```
rdp.class.file <- system.file("extdata", "rdp_classified_03.txt", package = "RDPutils")

rdp_tax <- make_tax_table(in_file = rdp.class.file, confidence = 0.8)
head(rdp_tax)
```

```
## Taxonomy Table:     [6 taxa by 6 taxonomic ranks]:
##       Domain     Phylum                   Class
## Otu1  "Bacteria" "Proteobacteria"         "Alphaproteobacteria"
## Otu10 "Bacteria" "unclassified_Bacteria" "unclassified_Bacteria"
## Otu11 "Bacteria" "Proteobacteria"         "Alphaproteobacteria"
## Otu12 "Bacteria" "Proteobacteria"         "Gammaproteobacteria"
## Otu13 "Bacteria" "Acidobacteria"          "Acidobacteria_Gp6"
## Otu14 "Bacteria" "unclassified_Bacteria" "unclassified_Bacteria"
##       Order                    Family
## Otu1  "Rhizobiales"            "Bradyrhizobiaceae"
## Otu10 "unclassified_Bacteria" "unclassified_Bacteria"
## Otu11 "Rhizobiales"            "unclassified_Rhizobiales"
## Otu12 "Xanthomonadales"        "Sinobacteraceae"
## Otu13 "Gp6"                    "Gp6"
## Otu14 "unclassified_Bacteria" "unclassified_Bacteria"
##       Genus
## Otu1  "Bradyrhizobium"
## Otu10 "unclassified_Bacteria"
## Otu11 "unclassified_Rhizobiales"
## Otu12 "unclassified_Sinobacteraceae"
## Otu13 "Gp6"
## Otu14 "unclassified_Bacteria"
```

```
rank_names(rdp_tax)
```

```
## [1] "Domain" "Phylum" "Class"  "Order"  "Family" "Genus"
```

```
taxa_names(rdp_tax)
```

```
##  [1] "Otu1"  "Otu10" "Otu11" "Otu12" "Otu13" "Otu14" "Otu15" "Otu16"
##  [9] "Otu17" "Otu18" "Otu19" "Otu2"  "Otu20" "Otu21" "Otu22" "Otu23"
## [17] "Otu24" "Otu25" "Otu26" "Otu27" "Otu28" "Otu29" "Otu3"  "Otu30"
## [25] "Otu31" "Otu32" "Otu33" "Otu34" "Otu35" "Otu36" "Otu37" "Otu38"
## [33] "Otu39" "Otu4"  "Otu40" "Otu41" "Otu42" "Otu43" "Otu44" "Otu45"
## [41] "Otu46" "Otu47" "Otu48" "Otu49" "Otu5"  "Otu50" "Otu51" "Otu52"
## [49] "Otu53" "Otu54" "Otu55" "Otu6"  "Otu7"  "Otu8"  "Otu9"
```

```
class(rdp_tax)
```

```
## [1] "taxonomyTable"
## attr(,"package")
## [1] "phyloseq"
```

UTAX

Taxonomy tables created with `USEARCH`'s utax function are imported with `import_utax_file`. The confidence level is chosen on import. It is 0.8 by default.

```
utax.table.file <- system.file("extdata", "utax_tax_table_03.txt", package = "RDPutils")

u_tax <- import_utax_file(in_file = utax.table.file, confidence = 0.8)
head(u_tax)
```

```
## Taxonomy Table:     [6 taxa by 6 taxonomic ranks]:
##      Domain       Phylum             Class
## Otu1 "d_Bacteria" "p_Proteobacteria" "c_Alphaproteobacteria"
## Otu2 "d_Bacteria" "uncl_d_Bacteria"  "uncl_d_Bacteria"
## Otu3 "d_Bacteria" "p_Proteobacteria" "c_Alphaproteobacteria"
## Otu4 "d_Bacteria" "uncl_d_Bacteria"  "uncl_d_Bacteria"
## Otu5 "d_Bacteria" "p_Acidobacteria"  "c_Acidobacteria_Gp6"
## Otu6 "d_Bacteria" "p_Acidobacteria"  "c_Acidobacteria_Gp6"
##      Order             Family                Genus
## Otu1 "o_Rhizobiales"   "f_Bradyrhizobiaceae" "g_Bradyrhizobium"
## Otu2 "uncl_d_Bacteria" "uncl_d_Bacteria"     "uncl_d_Bacteria"
## Otu3 "o_Rhizobiales"   "f_Hyphomicrobiaceae" "uncl_f_Hyphomicrobiaceae"
## Otu4 "uncl_d_Bacteria" "uncl_d_Bacteria"     "uncl_d_Bacteria"
## Otu5 "o_Gp6"           "uncl_o_Gp6"          "uncl_o_Gp6"
## Otu6 "o_Gp6"           "uncl_o_Gp6"          "uncl_o_Gp6"
```

```
class(u_tax)
```

```
## [1] "taxonomyTable"
## attr(,"package")
## [1] "phyloseq"
```

SINTAX

Taxonomy tables created with `USEARCH`'s sintax function are imported with `import_sintax_file`. The confidence level is chosen on import. It is 0.8 by default.

```
sintax.table.file <- system.file("extdata", "sintax_tax_table.txt", package = "RDPutils")

s_tax <- import_sintax_file(in_file = sintax.table.file, confidence = 0.8)
head(s_tax)
```

```
## Taxonomy Table:     [6 taxa by 6 taxonomic ranks]:
##      Domain       Phylum             Class
## Otu1 "d_Bacteria" "p_Proteobacteria" "c_Alphaproteobacteria"
## Otu2 "d_Bacteria" "uncl_d_Bacteria"  "uncl_d_Bacteria"
## Otu3 "d_Bacteria" "p_Proteobacteria" "c_Alphaproteobacteria"
## Otu4 "d_Bacteria" "uncl_d_Bacteria"  "uncl_d_Bacteria"
## Otu5 "d_Bacteria" "p_Acidobacteria"  "c_Acidobacteria_Gp6"
## Otu6 "d_Bacteria" "p_Acidobacteria"  "c_Acidobacteria_Gp6"
##      Order             Family                Genus
## Otu1 "o_Rhizobiales"   "f_Bradyrhizobiaceae" "g_Bradyrhizobium"
## Otu2 "uncl_d_Bacteria" "uncl_d_Bacteria"     "uncl_d_Bacteria"
## Otu3 "o_Rhizobiales"   "f_Hyphomicrobiaceae" "uncl_f_Hyphomicrobiaceae"
## Otu4 "uncl_d_Bacteria" "uncl_d_Bacteria"     "uncl_d_Bacteria"
```

```
## Otu5 "g_Gp6"           "uncl_g_Gp6"          "uncl_g_Gp6"
## Otu6 "g_Gp6"           "uncl_g_Gp6"          "uncl_g_Gp6"
```

```
class(s_tax)
```

```
## [1] "taxonomyTable"
## attr(,"package")
## [1] "phyloseq"
```

Importing Combined OTU & Taxonomy Tables

USEARCH offers options to produce files with combined OTU and taxonomy information in a tab-delimited text file and as a biom file, and RDPutils includes functions for importing these files as phyloseq objects with both OTU table and taxonomy tables. Confidences, however, cannot be chosen on import. That is, they cannot be altered from how they were created with the USEARCH command.

Import the tab-delimited otutab_taxa file:

```
otu.tab.tax.file <- system.file("extdata", "otu_03_tax_table.txt", package = "RDPutils")

otu_tax <- import_otutab_taxa(in_file = otu.tab.tax.file)
otu_tax
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 55 taxa and 4 samples ]
## tax_table()   Taxonomy Table:    [ 55 taxa by 6 taxonomic ranks ]
```

```
head(otu_table(otu_tax))
```

```
## OTU Table:          [6 taxa and 4 samples]
##                      taxa are rows
##       Sample_1 Sample_2 Sample_3 Sample_4
## Otu39        1        2        1        0
## Otu2         6        7       11       12
## Otu11        1        3        2        1
## Otu4         9        5        3        3
## Otu1        13       12        2       10
## Otu5        10        3       16       12
```

```
head(tax_table(otu_tax))
```

```
## Taxonomy Table:     [6 taxa by 6 taxonomic ranks]:
##       Domain       Phylum             Class
## Otu39 "d_Bacteria" "p_Proteobacteria" "c_Alphaproteobacteria"
## Otu2  "d_Bacteria" "uncl_d_Bacteria"  "uncl_d_Bacteria"
## Otu11 "d_Bacteria" "p_Proteobacteria" "c_Alphaproteobacteria"
## Otu4  "d_Bacteria" "uncl_d_Bacteria"  "uncl_d_Bacteria"
## Otu1  "d_Bacteria" "p_Proteobacteria" "c_Alphaproteobacteria"
## Otu5  "d_Bacteria" "p_Acidobacteria"  "c_Acidobacteria_Gp6"
##       Order             Family                Genus
## Otu39 "o_Rhizobiales"   "uncl_o_Rhizobiales"  "uncl_o_Rhizobiales"
## Otu2  "uncl_d_Bacteria" "uncl_d_Bacteria"     "uncl_d_Bacteria"
## Otu11 "o_Rhizobiales"   "uncl_o_Rhizobiales"  "uncl_o_Rhizobiales"
## Otu4  "uncl_d_Bacteria" "uncl_d_Bacteria"     "uncl_d_Bacteria"
## Otu1  "o_Rhizobiales"   "f_Bradyrhizobiaceae" "g_Bradyrhizobium"
## Otu5  "o_Gp6"           "uncl_o_Gp6"          "uncl_o_Gp6"
```

Import a `biom` file with both OTU and taxonomy tables.

```
biom.otu.tax.file <- system.file("extdata", "otu_03_tax_table.json", package="RDPutils")

biom_otu_tax <- import_biom(biom.otu.tax.file, parseFunction = parse_taxonomy_usearch)
biom_otu_tax
```

Including Sample Data, Reference Sequences, & Trees

For `USEARCH` processed data, reference sequences can always be included in an experiment level phyloseq object, as can sample data. Sample data text files are best created in a spreadsheet program and saved as a comma- or tab-delimited text file. The sample names in such a file must match exactly the sample names in the OTU table. Such a sample data file is read into `R` with the base `read.csv` or `read.table` functions and then converted to `phyloseq`'s `sample_data function`. If the gene of interest can be aligned and treed, as is the case with this 16S example, then the tree can also be included in the experiment level phyloseq object. This is not possible with ITS2 data because the sequences cannot be aligned.

An experiment-level `phyloseq` object is assembled from its component data with the `phyloseq` constructor, as below:

```
seq.file <- system.file("extdata", "otus_03.fa", package = "RDPutils")

my_seqs <- readDNAStringSet(seq.file, format = "fasta")
my_seqs
```

```
##   A DNAStringSet instance of length 55
##      width seq                                               names
##  [1]   330 TGCGTAGGCGGGTCTTTAAG...AGTACGGTCGCAAGATTAAA Otu1
##  [2]   332 CGCGTAGGCGGGATGGTAAG...AGTACGGCCGCAAGGTTGAA Otu2
##  [3]   330 CACGTAGGCGGATGTGCCAG...AGTACGGCCGCAAGGTTAAA Otu3
##  [4]   332 CGCGTAGGCGGGATGGCAAG...AGTACGGCCGCAAGGTTGAA Otu4
##  [5]   332 CTCGTAGGCGGCCAACTAAG...AGTACGGTCGCAAGGCTGAA Otu5
##  ...   ... ...
## [51]   330 AGTGTAGGTGGTTGTCCAAG...AGTACGGCCGCAAGGTTGAA Otu51
## [52]   332 CGCGTAGGCGGCTTGACAAG...AGTACGGTCGCAAGGCTGAA Otu52
## [53]   330 CGCGTAGGCGGCTTTGTAAG...AGTACGGTCGCAAGATTAAA Otu53
## [54]   330 CGCGTAGGCGGCTTATCAAG...AGTACGGTCGCAAGATTAAA Otu54
## [55]   335 CTCGTAGGCGGTTCAGCAAG...AGTACGGCCGCAAGGCTAAA Otu55
```

```
usearch.tree.file <- system.file("extdata", "usearch_03_tree.nwk", package = "RDPutils")
my_tree <- read_tree(usearch.tree.file, errorIfNULL = TRUE)
my_tree
```

```
##
## Phylogenetic tree with 55 tips and 53 internal nodes.
##
## Tip labels:
##  Otu19, Otu14, Otu20, Otu10, Otu40, Otu21, ...
## Node labels:
##  , 0.892, 0.881, 0.780, 0.988, 0.921, ...
##
## Unrooted; includes branch lengths.
```

```
sam.data.file <- system.file("extdata", "sam_data.txt", package = "RDPutils")
sam.data <- read.table(file = sam.data.file, header = TRUE, row.names = 1, sep = "\t")
my_sam <- sample_data(sam.data, errorIfNULL = TRUE)
```

```
expt <- phyloseq(my_otu, my_sam, s_tax, my_tree, my_seqs)
expt
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 55 taxa and 4 samples ]
## sample_data() Sample Data:       [ 4 samples by 4 sample variables ]
## tax_table()   Taxonomy Table:    [ 55 taxa by 6 taxonomic ranks ]
## phy_tree()    Phylogenetic Tree: [ 55 tips and 53 internal nodes ]
## refseq()      DNAStringSet:      [ 55 reference sequences ]
```

Substituting Taxonomy Tables

If as in the script above you have created several taxonomy tables, you may substitute one for another in the experiment level `phyloseq` object. If you do this after the taxa have been subset in some way, the substituting taxonomy table is automatically subset on substitution. To demonstrate, subset `expt` to include only the 20 more abundant OTUs and then substitute the present tax table created with `sintax` with the one created with the RDP classifier:

```
keep <- names(sort(taxa_sums(expt), decreasing = TRUE)[1:20])
expt.top.20 <- prune_taxa(keep, expt)
tax_table(expt.top.20) <- rdp_tax
expt.top.20
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 20 taxa and 4 samples ]
## sample_data() Sample Data:       [ 4 samples by 4 sample variables ]
## tax_table()   Taxonomy Table:    [ 20 taxa by 6 taxonomic ranks ]
## phy_tree()    Phylogenetic Tree: [ 20 tips and 18 internal nodes ]
## refseq()      DNAStringSet:      [ 20 reference sequences ]
```

It is obvious that the taxa have been subset. `expt` contains 55 taxa, while `expt.top.20` contains only 20. To confirm that the taxonomy tables have been substituted, examine the first few rows in `expt` and `expt.top.20` taxonomy tables:

```
head(tax_table(expt))
```

```
## Taxonomy Table:     [6 taxa by 6 taxonomic ranks]:
##       Domain       Phylum            Class
## Otu19 "d_Bacteria" "uncl_d_Bacteria" "uncl_d_Bacteria"
## Otu14 "d_Bacteria" "p_Acidobacteria" "c_Acidobacteria_Gp16"
## Otu20 "d_Bacteria" "p_Chloroflexi"   "c_Anaerolineae"
## Otu10 "d_Bacteria" "uncl_d_Bacteria" "uncl_d_Bacteria"
## Otu40 "d_Bacteria" "uncl_d_Bacteria" "uncl_d_Bacteria"
## Otu21 "d_Bacteria" "p_Acidobacteria" "c_Acidobacteria_Gp4"
##       Order                         Family
## Otu19 "uncl_d_Bacteria"             "uncl_d_Bacteria"
## Otu14 "uncl_c_Acidobacteria_Gp16" "uncl_c_Acidobacteria_Gp16"
## Otu20 "o_Anaerolineales"            "f_Anaerolineaceae"
## Otu10 "uncl_d_Bacteria"             "uncl_d_Bacteria"
## Otu40 "uncl_d_Bacteria"             "uncl_d_Bacteria"
## Otu21 "g_Gp4"                       "uncl_g_Gp4"
##       Genus
## Otu19 "uncl_d_Bacteria"
## Otu14 "g_Gp16"
```

```
## Otu20 "uncl_f_Anaerolineaceae"
## Otu10 "uncl_d_Bacteria"
## Otu40 "uncl_d_Bacteria"
## Otu21 "uncl_g_Gp4"
```

```
head(tax_table(expt.top.20))
```

```
## Taxonomy Table:     [6 taxa by 6 taxonomic ranks]:
##       Domain     Phylum                   Class
## Otu14 "Bacteria" "unclassified_Bacteria" "unclassified_Bacteria"
## Otu20 "Bacteria" "Chloroflexi"           "Anaerolineae"
## Otu21 "Bacteria" "Acidobacteria"         "Acidobacteria_Gp4"
## Otu43 "Bacteria" "Acidobacteria"         "Acidobacteria_Gp4"
## Otu5  "Bacteria" "Acidobacteria"         "Acidobacteria_Gp6"
## Otu6  "Bacteria" "Acidobacteria"         "Acidobacteria_Gp6"
##       Order                    Family
## Otu14 "unclassified_Bacteria" "unclassified_Bacteria"
## Otu20 "Anaerolineales"        "Anaerolineaceae"
## Otu21 "Gp4"                   "Gp4"
## Otu43 "Gp4"                   "Gp4"
## Otu5  "Gp6"                   "Gp6"
## Otu6  "Gp6"                   "Gp6"
##       Genus
## Otu14 "unclassified_Bacteria"
## Otu20 "unclassified_Anaerolineaceae"
## Otu21 "Gp4"
## Otu43 "Gp4"
## Otu5  "Gp6"
## Otu6  "Gp6"
```