Package 'phylotypr'

February 21, 2025

Type Package

Title Classifying DNA Sequences to Taxonomic Groupings

Version 0.1.1

Description Classification based analysis of DNA sequences to taxonomic groupings. This package primarily implements Naive Bayesian Classifier from the Ribosomal Database Project. This approach has traditionally been used to classify 16S rRNA gene sequences to bacterial taxonomic outlines; however, it can be used for any type of gene sequence. The method was originally described by Wang, Garrity, Tiedje, and Cole in Applied and Environmental Microbiology 73(16):5261-7 <doi:10.1128/AEM.00062-07>. The package also provides functions to read in 'FASTA'-formatted sequence data.

License GPL-3

Encoding UTF-8

LazyData true

LazyDataCompression xz

URL https://github.com/mothur/phylotypr, https://mothur.org/phylotypr/

BugReports https://github.com/mothur/phylotypr/issues

RoxygenNote 7.3.2

Suggests knitr, rmarkdown, purrr, dplyr, testthat (>= 3.0.0)

Config/testthat/edition 3

Imports Rcpp, readr (>= 2.1.0), Rfast (>= 2.1.0), stats (>= 4.0.0), stringi (>= 1.8.0)

LinkingTo Rcpp

Depends R (>= 4.2)

VignetteBuilder knitr

NeedsCompilation yes

Author Pat Schloss [aut, cre, cph] (<https://orcid.org/0000-0002-6935-4275>)

Maintainer Pat Schloss <pschloss@umich.edu>

Repository CRAN

Date/Publication 2025-02-21 14:40:02 UTC

Contents

`build_kmer_database` *Build kmer database*

Description

Build kmer database for classifying 16S rRNA and other gene sequences to a genus when a kmer size is provided.

Usage

```
build_kmer_database(sequences, genera, kmer_size = 8)
```

Arguments

`sequences`	A vector of reference sequences for which we have genus-level taxonomic information in the same order as the value for genera.
`genera`	A character vector of genus-level taxonomic information for reference sequences in the same order as the value for sequences. Ideally, taxonomic information will be provided back to the domain level with each level separated by semicolons and no spaces.
`kmer_size`	An integer indicating the length of the nucleotide word to base our classification on (default = 8)

Value

A list object containing the genus level conditional probability (`conditional_prob`) of seeing each kmer in a given genus as well as the genus names (`genera`)

References

Wang Q, Garrity GM, Tiedje JM, Cole JR. Naive Bayesian classifier for rapid assignment of rRNA sequences into the new bacterial taxonomy. Appl Environ Microbiol. 2007 Aug;73(16):5261-7. doi:10.1128/AEM.00062-07 PMID: 17586664; PMCID: PMC1950982.

Examples

```
kmer_size <- 3
sequences <- c("ATGCGCTA", "ATGCGCTC", "ATGCGCTC")
genera <- c("A", "B", "B")

build_kmer_database(sequences, genera, kmer_size)
```

`classify_sequence` *Classify 16S rRNA gene sequence fragment*

Description

The `classify_sequence()` function implements the Wang et al. naive Bayesian classification algorithm for 16S rRNA gene sequences.

Usage

```
classify_sequence(
  unknown_sequence,
  database,
  kmer_size = 8,
  num_bootstraps = 100
)
```

Arguments

`unknown_sequence`
A character object representing a DNA sequence that needs to be classified

`database` A kmer database generated using `build_kmer_database`

`kmer_size` An integer value (default of 8) indicating the size of kmers to use for classifying sequences. Higher values use more RAM with potentially more specificity Lower values use less RAM with potentially less specificity. Benchmarking has found that the default of 8 provides the best specificity with the lowest possible memory requirement and fastest execution time.

`num_bootstraps` An integer value (default of 100). The value of `num_bootstraps` is the number of randomizations to perform where `1/kmer_size` of all kmers are sampled (without replacement) from `unknown_sequence`. Higher values will provide greater precision on the confidence score.

Value

A list object of two vectors. One vector (`taxonomy`) is the taxonomic assignment for each level. The second vector (`confidence`) is the percentage of `num_bootstraps` that the classifier gave the same classification at that level

References

Wang Q, Garrity GM, Tiedje JM, Cole JR. Naive Bayesian classifier for rapid assignment of rRNA sequences into the new bacterial taxonomy. Appl Environ Microbiol. 2007 Aug;73(16):5261-7. doi:10.1128/AEM.00062-07 PMID: 17586664; PMCID: PMC1950982.

Examples

```
kmer_size <- 3
sequences <- c("ATGCGCTA", "ATGCGCTC", "ATGCGCTC")
genera <- c("A", "B", "B")

db <- build_kmer_database(sequences, genera, kmer_size)
unknown_sequence <- "ATGCGCTC"

classify_sequence(
  unknown_sequence = unknown_sequence,
  database = db,
  kmer_size = kmer_size
)
```

`filter_taxonomy` *Filter taxonomy*

Description

The `filter_taxonomy()` function will filter a consensus taxonomy to remove any taxonomic levels where the confidence score is below a `min_confidence` level

Usage

```
filter_taxonomy(consensus, min_confidence = 80)
```

Arguments

`consensus`	A list object that contains two slots each with an equal sized vector. The `taxonomy` vector contains the classification at each taxonomic level and the `confidence` vector contains the percentage of bootstraps that had the specified classification
`min_confidence`	A double value between 0 and 100 (default = 80). The minimum percentage of bootstrap replicates that had the same classification. Any confidence score below this value will have the corresponding taxonomy removed

Value

A list object containing two equally sized vectors that are filtered to remove low confidence taxonomies. One vector, `taxonomy`, contains the taxonomy at each taxonomic level and the other vector, `confidence` contains the confidence score for that taxonomy. There will be no taxonomies or confidence scores below `min_confidence`

Examples

```
oscillospiraceae <- list(
  taxonomy = c(
    "Bacteria", "Bacillota", "Clostridia",
    "Eubacteriales", "Oscillospiraceae",
    "Flintibacter"
  ),
  confidence = c(100, 100, 99, 99, 98, 58)
)

filter_taxonomy(oscillospiraceae, min_confidence = 80)
```

---

phylotypr_example *Get path to phylotypr example*

Description

phylotypr comes bundled with some example files in its `inst/extdata` directory. This function make them easy to access.

Usage

```
phylotypr_example(path = NULL)
```

Arguments

`path`	Name of file. If `NULL`, the example files will be listed.

Value

A string indicating path to the file listed in `path`. If `NULL` is given then the return value is a vector of file names in the `extdata/` directory

Examples

```
phylotypr_example()
phylotypr_example("miseq_sop.fasta.gz")
```

print_taxonomy *Print taxonomy for an unknown sequence*

Description

The `print_taxonomy()` will output the consensus taxonomy for an unknown sequence with confidence scores for each taxonomic level and each taxonomic level separated by semi-colons

Usage

```
print_taxonomy(consensus, n_levels = 6)
```

Arguments

`consensus` A list object that contains two slots each with an equal sized vector. The `taxonomy` vector contains the classification at each taxonomic level and the `confidence` vector contains the percentage of bootstraps that had the specified classification

`n_levels` An integer indicating the number of taxonomic levels to expect. If the number of observed levels is less than this value, then missing levels will have "_unclassified" to the end of the last named classification

Value

A character string indicating the classification at each taxonomic level with the corresponding confidence in parentheses. Each taxonomic level is separated by a semi-colon

Examples

```
oscillospiraceae <- list(
  taxonomy = c(
    "Bacteria", "Bacillota", "Clostridia",
    "Eubacteriales", "Oscillospiraceae"
  ),
  confidence = c(100, 100, 99, 99, 98)
)

print_taxonomy(oscillospiraceae, n_levels = 6)
```

read_fasta *Read in a FASTA-formatted file containing DNA sequences*

Description

Given a standard FASTA-formatted file, `read_fasta` will read in the contents of the file and create a three column data frame with columns for the sequence id, the sequence itself, and any comments found in the header line for each sequence.

Usage

```
read_fasta(file, degap = TRUE)
```

Arguments

`file` Either a path to a file, a connection, or literal data (either a single string or a raw vector) containing DNA sequences in the standard FASTA format. There are no checks to determine whether the data are DNA or amino acid sequences.

Files ending in .gz, .bz2, .xz, or .zip will be automatically uncompressed. Files starting with `http://`, `https://`, `ftp://`, or `ftps://` will be automatically downloaded. Remote gz files can also be autom downloaded and decompressed.

`degap` Logical value (default = TRUE) Removes gap characters from sequences indicated by "." or "-"

Value

A data frame object with three columns. The `id` column will contain the non-space characters following the > in the header line of each sequence; the `sequence` column will contain the sequence; and the `comment` column will contain any text found after the first whitespace character on the header line.

Note

The sequences in the FASTA file can have line breaks within them and `read_fasta()` will put those separate lines into the same sequence

Examples

```
temp <- tempfile()
write(">seqA\nATGCATGC\n>seqB\nTACGTACG", file = temp)
write(">seqC\nTCCGATGC", file = temp, append = TRUE)
write(">seqD B.ceresus UW85\nTCCGATGC", file = temp, append = TRUE)
write(">seq4\tE. coli K12\tBacteria;Proteobacteria;\nTCCGATGC",
  file = temp,
  append = TRUE
)
write(">seq_4\tSalmonella LT2\tBacteria;Proteobacteria;\nTCCGATGC",
  file = temp, append = TRUE
)
write(">seqE B.ceresus UW123\nTCCGATGC\nTCCGATGC",
  file = temp,
  append = TRUE
)

sequence_df <- read_fasta(temp)
```

read_taxonomy *Read in taxonomy files*

Description

Read a mothur-formatted taxonomy file into R as a data frame

Usage

```
read_taxonomy(file)
```

Arguments

`file`	Either a path to a file, a connection, or literal data (either a single string or a raw vector) containing the sequence id and the taxonomy information for each sequence. Files ending in .gz, .bz2, .xz, or .zip will be automatically uncompressed. Files starting with `http://`, `https://`, `ftp://`, or `ftps://` will be automatically downloaded. Remote gz files can also be autom downloaded and decompressed.

Value

A data frame with two columns. The `id` column contains a name for each sequence and the `taxonomy` column, which contains the taxonomy for each sequence. The string in the `taxonomy` column is a series of taxonomic names separated by semi-colons. The string does not have a semicolon at the end of the sequence

Note

There are no checks to insure that each sequence has a unique id value. It is also assumed that each sequence has the same number of taxonomic levels represented in the second column of the input file.

Examples

```
temp <- tempfile()
write("seqA\tA;B;C;", file = temp)
write("seqB\tA;B; C;", file = temp, append = TRUE)
write("seqC\tA; B;C;", file = temp, append = TRUE)
write("seqD\tA;B;C", file = temp, append = TRUE)
write("seqE\tA;B; C", file = temp, append = TRUE)
write("seqF\tA; B;C", file = temp, append = TRUE)
write("seq G\tA;B;C;", file = temp, append = TRUE)

read_taxonomy(temp)
```

`trainset9_rdp` *RDP training set v9*

Description

The sequence and taxonomy data for the 10,049 sequences found in the Ribosomal Database Project's trainset9_032012 training set for use with the naive Bayesian classifier as implemented in the `{phylyotypr}` R package. Originally released by the RDP in September 2012. The `rdp` version contains the same sequences as provided by the official RDP version (9,665 bacterial and 384 archaeal). The `pds` version contains extra eukaryotic sequences including 119 chloroplasts and mitochondria (10,168 total sequences). See the mothur reference file page in "Sources" for more information. Be sure to see the mothur GitHub project where you can find the phylotyprrefdata package (https://github.com/mothur/phylotyprrefdata) for access to other taxonomic reference data.

Usage

```
trainset9_rdp

trainset9_pds
```

Format

A data frame with 3 columns. Each row represents a different sequence:

id Sequence accession identifier

sequence DNA sequence string

taxonomy Taxonomic string with each level separated with a ;

An object of class `data.frame` with 10169 rows and 3 columns.

Source

- mothur-formatted files
- RDP sourceforge page # nolint: line_length_linter

`write_fasta` *Write to a FASTA-formatted file*

Description

Writes a data frame containing id, sequence, and comment columns, `write_fasta` will write the data frame out to a standard FASTA-formatted file. The header will have a tab character between the sequence id and any comments. There won't be a tab if there's no comment for the sequence. All sequence data will be on a single line

Usage

```
write_fasta(data_frame, file = NULL)
```

Arguments

`data_frame` A data frame object with three columns. The `id` column will contain the non-space characters following the `>` in the header line of each sequence; the `sequence` column will contain the sequence; and the `comment` column will contain any text found after the first whitespace character on the header line. The `comment` column is optional.

`file` Either a path to a file, a connection, or literal data (either a single string or a raw vector) to write to a standard FASTA formatted file. There are no checks to determine whether the data are DNA or amino acid sequences.

Files ending in .gz, .bz2, .xz, or .zip will be automatically compressed. Files starting with `http://`, `https://`, `ftp://`, or `ftps://` will be automatically downloaded. Remote gz files can also be autom downloaded and decompressed.

If the value of `file` is `NULL` (default), the string will be written out to the screen

Value

Sequence data is either written out to the screen (`file = NULL`) or to a file.

Examples

```
df_d <- data.frame(
  id = c("seqA", "seqB", "seqC"),
  sequence = c("ATGCATGC", "ATGCATGA", "ATGCATGT"),
  comment = c("comment 1", "", "comment 3")
)

string_d <- write_fasta(df_d)
```

Index