Package 'isolateR'

February 14, 2024

Type Package

Title Automated processing of Sanger sequencing data, taxonomic profiling, and generation of microbial strain libraries

Version 0.0.0.9003

Date 2023-11-20

Author Brendan Daisley, Sarah Vancuren, Dylan Brettingham, Jacob Wilde, Emma Allen-Vercoe.

Maintainer Brendan Daisley <bdaisley@uoguelph.ca>

Description isolateR aims to enhance microbial isolation workflows and support the identification of novel taxa. It addresses the challenges of manual Sanger sequencing data processing and limitations of conventional BLAST searches, crucial for identifying microorganisms and creating strain libraries. The package offers a streamlined three-step process that automates quality trimming Sanger sequence files, taxonomic classification via global alignment against type strain databases, and efficient strain library creation based on customizable sequence similarity thresholds. It features interactive HTML output tables for easy data exploration and optional tools for generating phylogenetic trees to visualize microbial diversity.

License GPL (>= 2)

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Suggests knitr,rmarkdown

VignetteBuilder knitr

biocViews

Imports ape, BiocManager, crosstalk, dataui, ggtree, htmltools, patchwork, LPSN, methods, msa, pander, R.utils, reactable, rentrez, sangeranalyseR, sangerseqR, scales, seqinr, shiny, stringr, svMisc, treeio, xmlconvert

Depends Biostrings,dplyr,reactablefmtr

Remotes timelyportfolio/dataui, glin/reactable, thomasp85/patchwork

Additional_repositories http://R-Forge.R-project.org

R topics documented:

`class-isoLIB` *isoLIB Class Object*

Description

S4 wrapper for `isoLIB` function. Access data via S4 slot functions.

Value

Returns an class-isoLIB object.

Slots

`input` Character string containing input directory information.

`strain_group` Character string containing list of group representative filenames.

`date` Character string containing run date from each of the input Sanger sequence .ab1 files ("YYYY_MM_DD" format).

`filename` Character string containing input filenames.

`phred_trim` Numeric string containing mean Phred scores after trimming.

`Ns_trim` Numeric string containing count of N's after trimming.

`length_trim` Numeric string containing sequence length after trimming.

`seqs_trim` Character string containing sequence after trimming.

`closest_match` Character string containing species + type strain no. of closest match from reference database.

`NCBI_acc` Character string containing NCBI accession number associated with closest match from reference database.

`ID` Numeric string containing containing pairwise similarity value for query vs database reference sequence. Calculation of ID is determined by isoTAX 'iddef' parameter (0-4, Default=2). See VSEARCH documentation for more details.

- (0) CD-HIT definition: (matching columns) / (shortest sequence length).
- (1) Edit distance: (matching columns) / (alignment length).
- (2) Edit distance excluding terminal gaps (default definition).
- (3) Marine Biological Lab definition counting each gap opening (internal or terminal) as a single mismatch, whether or not the gap was extended: 1.0- ((mismatches + gap openings)/(longest sequence length)).
- (4) BLAST definition, equivalent to –iddef 1 for global pairwise alignments.

`rank_phylum` Character string containing Phylum rank taxonomy

`rank_class` Character string containing Class rank taxonomy

`rank_order` Character string containing Order rank taxonomy

`rank_family` Character string containing Family rank taxonomy

`rank_genus` Character string containing Genus rank taxonomy

`rank_species` Character string containing Species rank taxonomy

`phylum_cutoff` Numeric string containing Phylum-level cutoff threshold

`class_cutoff` Numeric string containing Class-level cutoff threshold

`order_cutoff` Numeric string containing Order-level cutoff threshold

`family_cutoff` Numeric string containing Family-level cutoff threshold

`genus_cutoff` Numeric string containing Genus-level cutoff threshold

`species_cutoff` Numeric string containing Species-level cutoff threshold

See Also

`isoLIB`

`class-isoQC` *isoQC Class Object*

Description

S4 wrapper for `isoQC` function. Access data via S4 slot functions.

Value

Returns an class-isoQC object.

Slots

`date` Character string containing run date from each of the input Sanger sequence .ab1 files ("YYYY_MM_DD" format).

`filename` Character string containing input filenames.

`trim.start.pos` Numeric string containing trimming position start point.

`trim.end.pos` Numeric string containing trimming position end point.

`phred_spark_raw` List containing per nucleotide Phred score values for each sequence

`phred_raw` Numeric string containing mean Phred scores before trimming.

`phred_trim` Numeric string containing mean Phred scores after trimming.

`Ns_raw` Numeric string containing count of N's before trimming.

`Ns_trim` Numeric string containing count of N's after trimming.

`length_raw` Numeric string containing sequence length before trimming.

`length_trim` Numeric string containing sequence length after trimming.

`seqs_raw` Character string containing sequences before trimming.

`seqs_trim` Character string containing sequence after trimming.

`decision` Character string containing decision (PASS/FAIL) information based on `isoQC` 'min_phred_score' and 'min_length cutoffs'.

`input` Character string containing input directory information.

See Also

`isoQC`

`class-isoTAX` *isoTAX Class Object*

Description

S4 wrapper for `isoTAX` function. Access data via S4 slot functions.

Value

Returns an class-isoTAX object.

Slots

`input` Character string containing input directory information.

`warning` Character string containing list filenames of sequences that had poor alignment during taxonomic classification step.

`date` Character string containing run date from each of the input Sanger sequence .ab1 files ("YYYY_MM_DD" format).

`filename` Character string containing input filenames.

`phred_spark_raw` List containing per nucleotide Phred score values for each sequence

`phred_raw` Numeric string containing mean Phred scores before trimming.

`phred_trim` Numeric string containing mean Phred scores after trimming.

`Ns_raw` Numeric string containing count of N's before trimming.

`Ns_trim` Numeric string containing count of N's after trimming.

`length_raw` Numeric string containing sequence length before trimming.

`length_trim` Numeric string containing sequence length after trimming.

`seqs_raw` Character string containing sequences before trimming.

`seqs_trim` Character string containing sequence after trimming.

`closest_match` Character string containing species + type strain no. of closest match from reference database.

`NCBI_acc` Character string containing NCBI accession number associated with closest match from reference database.

`ID` Numeric string containing containing pairwise similarity value for query vs database reference sequence. Calculation of ID is determined by isoTAX 'iddef' parameter (0-4, Default=2). See VSEARCH documentation for more details.

- (0) CD-HIT definition: (matching columns) / (shortest sequence length).
- (1) Edit distance: (matching columns) / (alignment length).
- (2) Edit distance excluding terminal gaps (default definition).
- (3) Marine Biological Lab definition counting each gap opening (internal or terminal) as a single mismatch, whether or not the gap was extended: 1.0- ((mismatches + gap openings)/(longest sequence length)).
- (4) BLAST definition, equivalent to –iddef 1 for global pairwise alignments.

`rank_phylum` Character string containing Phylum rank taxonomy

`rank_class` Character string containing Class rank taxonomy

`rank_order` Character string containing Order rank taxonomy

`rank_family` Character string containing Family rank taxonomy

`rank_genus` Character string containing Genus rank taxonomy

`rank_species` Character string containing Species rank taxonomy

`phylum_cutoff` Numeric string containing Phylum-level cutoff threshold

`class_cutoff` Numeric string containing Class-level cutoff threshold

`order_cutoff` Numeric string containing Order-level cutoff threshold

`family_cutoff` Numeric string containing Family-level cutoff threshold

`genus_cutoff` Numeric string containing Genus-level cutoff threshold

`species_cutoff` Numeric string containing Species-level cutoff threshold

See Also

`isoTAX`

df_to_isoLIB *Convert isoLIB .CSV output to isoLIB class object*

Description

Helper function to convert isoLIB .CSV output to a class-isoLIB class object.

Usage

```
df_to_isoLIB(df)
```

Arguments

df	Dataframe in same format as .CSV output file from isoLIB step.

Value

Returns an S4 class-isoLIB object that can be used to generate interactive HTML output tables.

df_to_isoTAX *Convert isoTAX .CSV output to isoTAX class object*

Description

Helper function to convert isoTAX .CSV output to a class-isoTAX class object.

Usage

```
df_to_isoTAX(df)
```

Arguments

df	Dataframe in same format as .CSV output file from isoTAX step.

Value

Returns an S4 class-isoTAX object that can be used to generate interactive HTML output tables.

export_html *Export HTML for isoQC > isoTAX > isoLIB class objects*

Description

S4 wrapper functions to export interactive HTML tables from isoQC, isoTAX, or isoLIB class objects. Saves to HTML to current working directory and automatically opens.

Usage

```
## S4 method for signature 'isoQC'
export_html(obj)

## S4 method for signature 'isoTAX'
export_html(obj)

## S4 method for signature 'isoLIB'
export_html(obj)
```

Arguments

obj	An S4 class object generated from one of isoQC, isoTAX, or isoLIB steps

Value

HTML output file saved to working directory.

get_db *Download taxonomic reference database*

Description

This function donwloads taxonomic reference database and formats them for use.

Usage

```
get_db(db = "16S", force_update = FALSE)
```

Arguments

db	Database selection. One of "16S", "16S_arc", "18S", "ITS", or "cpn60"
force_update	Forces new datbases to be downloaded.

Value

Returns file path for database of interest

Examples

```
db.path <- get_db(db="16S", force_update=FALSE)
```

get_os *Determine user operating system.*

Description

Determines the type of operating system being used.

Usage

```
get_os()
```

Value

Returns sysname as one of windows/osx-mac/linux

Examples

```
#Example 1 on a Windows-based operating system
os.index <- get_os()
print(os.index)

#Example 2 on a Mac operating system
os.index <- get_os()
print(os.index)

#Example 3 on a Linux operating system
os.index <- get_os()
print(os.index)
```

get_sanger_date *get_sanger_date function*

Description

Helper function to automatically retrieve run date from Sanger sequencing .ab1 files.

Usage

```
get_sanger_date(file = NULL)
```

Arguments

file	The .ab1 file in from which to retrieve the date information. (Must be in S4 abif format)

Value

Returns date in "YYYY_MM_DD" format

Examples

```
#Path to the first listed .ab1 file in example directory
fpath <- file.path(system.file("extdata/abif_examples/rocket_salad", package = "isolateR"),
            list.files(system.file("extdata/abif_examples/rocket_salad", package = "isolateR"))[1])
#Read in the ab1 file to S4 format
ab1.S4 <- sangerseqR::read.abif(fpath)

#Retrieve date
get_sanger_date(ab1.S4)
```

`isoLIB` *Perform all commands in one step.*

Description

This function effectively wraps isoQC, isoTAX, and isoLIB steps into a single command for convenience. Input can be a single directory or a list of directories to process at once. If multiple directories are provided, the resultant libraries can be sequentially merged together by toggling the parameter 'merge=TRUE'. All other respective parameters from the wrapped functions can be passed through this command. . The The respective input parameters from the wrappred can be passed through this command with exception of the .creates a strain library by grouping closely related strains of interest based on sequence similarity. For adding new sequences to an already-established strain library, specify the .CSV file path of the older strain library using the 'old_lib_csv" parameter.

This function creates a strain library by grouping closely related strains of interest based on sequence similarity. For adding new sequences to an already-established strain library, specify the .CSV file path of the older strain library using the 'old_lib_csv" parameter.

Usage

```
isoALL(
  input = NULL,
  export_html = TRUE,
  export_csv = TRUE,
  export_fasta = TRUE,
  export_fasta_revcomp = FALSE,
  quick_search = FALSE,
  db = "16S",
  iddef = 2,
  phylum_cutoff = 75,
  class_cutoff = 78.5,
  order_cutoff = 82,
  family_cutoff = 86.5,
  genus_cutoff = 96.5,
  species_cutoff = 98.7,
  include_warnings = FALSE,
  strain_group_cutoff = 0.995,
  merge = FALSE
)

isoLIB(
```

```
  input = NULL,
  old_lib_csv = NULL,
  export_html = TRUE,
  export_csv = TRUE,
  include_warnings = FALSE,
  strain_group_cutoff = 0.995,
  phylum_cutoff = 75,
  class_cutoff = 78.5,
  order_cutoff = 82,
  family_cutoff = 86.5,
  genus_cutoff = 96.5,
  species_cutoff = 98.7
)
```

Arguments

input	Path of CSV output file from isoTAX step.
export_html	(Default=TRUE) Output the results as an HTML file
export_csv	(Default=TRUE) Output the results as a CSV file.
export_fasta	(Default=TRUE) Output the sequences in a FASTA file.
export_fasta_revcomp	(Default=FALSE) Output the sequences in reverse complement form in a fasta file. This is useful in cases where sequencing was done using the reverse primer and thus the orientation of input sequences needs reversing.
quick_search	(Default=FALSE) Whether or not to perform a comprehensive database search (i.e. optimal global alignment). If TRUE, performs quick search equivalent to setting VSEARCH parameters "–maxaccepts 100 –maxrejects 100". If FALSE, performs comprehensive search equivalent to setting VSEARCH parameters "–maxaccepts 0 –maxrejects 0"
db	(Default="16S") Select database option(s) including "16S" (for searching against the NCBI Refseq targeted loci 16S rRNA database), "ITS" (for searching against the NCBI Refseq targeted loci ITS database. For combined databases in cases where input sequences are dervied from bacteria and fungi, select "16S|ITS".
iddef	Set pairwise identity definition as per VSEARCH definitions (Default=2, and is recommended for highest taxonomic accuracy) (0) CD-HIT definition: (matching columns) / (shortest sequence length). (1) Edit distance: (matching columns) / (alignment length). (2) Edit distance excluding terminal gaps (default definition). (3) Marine Biological Lab definition counting each gap opening (internal or terminal) as a single mismatch, whether or not the gap was extended: 1.0-((mismatches + gap openings)/(longest sequence length)). (4) BLAST definition, equivalent to –iddef 1 for global pairwise alignments.
phylum_cutoff	Percent cutoff for phylum rank demarcation
class_cutoff	Percent cutoff for class rank demarcation
order_cutoff	Percent cutoff for order rank demarcation
family_cutoff	Percent cutoff for family rank demarcation
genus_cutoff	Percent cutoff for genus rank demarcation
species_cutoff	Percent cutoff for species rank demarcation

include_warnings
: (Default=FALSE) Whether or not to keep sequences with poor alignment warnings from Step 2 'isoTAX' function. Set TRUE to keep warning sequences, and FALSE to remove warning sequences.

strain_group_cutoff
: (Default=0.995) Similarity cutoff (0-1) for delineating between strain groups. 1 = 100% identical/0.995=0.5% difference/0.95=5.0% difference/etc.

old_lib_csv
: Optional: Path of CSV output isoLIB file or combined isoLIB file from previous run(s)

verbose
: (Default=FALSE) Output progress while script is running.

files_manual
: (Default=NULL) For testing purposes only. Specify a list of files to run as filenames without extensions, rather than the whole directory format. Primarily used for testing, use at your own risk.

exclude
: (Default=NULL) For testing purposes only. Excludes files of interest from input directory.

min_phred_score
: (Default=20) Do not accept trimmed sequences with a mean Phred score below this cutoff

min_length
: (Default=200) Do not accept trimmed sequences with sequence length below this number

sliding_window_cutoff
: (Default=NULL) Quality trimming parameter (M2) for wrapping SangerRead function in sangeranalyseR package. If NULL, implements auto cutoff for Phred score (recommended), otherwise set between 1-60.

sliding_window_size
: (Default=15) Quality trimming parameter (M2) for wrapping SangerRead function in sangeranalyseR package. Recommended range between 5-30.

date
: Set date "YYYY_MM_DD" format. If NULL, attempts to parse date from .ab1 file

Value

Returns a list of class-isoLIB class objects.

Returns an isoLIB class object. Default taxonomic cutoffs for phylum (75.0), class (78.5), order (82.0), family (86.5), genus (96.5), and species (98.7) demarcation are based on Yarza et al. 2014, Nature Reviews Microbiology (DOI:10.1038/nrmicro3330)

See Also

isoQC, isoTAX, isoLIB

isoTAX, isoLIB

Examples

```
#Set path to directory containing example .ab1 files
fpath1 <- system.file("extdata/abif_examples/rocket_salad", package = "isolateR")

#Run isoALL function with default settings
isoALL(input=fpath1)
#Set path to directory containing example .ab1 files
fpath1 <- system.file("extdata/abif_examples/rocket_salad", package = "isolateR")
```

```
#Step 1: Run isoQC function with default settings
isoQC.S4 <- isoQC(input=fpath1)

#Step 2: Run isoTAX function with default settings
fpath2 <- file.path(fpath1, "isolateR_output/01_isoQC_trimmed_sequences_PASS.csv")
isoTAX.S4 <- isoTAX(input=fpath2)

#Step 3: Run isoLIB function with default settings
fpath3 <- file.path(fpath1, "isolateR_output/02_isoTAX_results.csv")
isoLIB.S4 <- isoLIB(input=fpath3)

#Show summary statistics
isoLIB.S4
```

`isoQC` *Perform automated quality trimming of input .ab1 files*

Description

This function loads in ABIF files (.ab1 extension) and performs automatic quality trimming in batch mode.

Usage

```
isoQC(
  input = NULL,
  export_html = TRUE,
  export_csv = TRUE,
  export_fasta = TRUE,
  export_fasta_revcomp = FALSE,
  verbose = FALSE,
  exclude = NULL,
  min_phred_score = 20,
  min_length = 200,
  sliding_window_cutoff = NULL,
  sliding_window_size = 15,
  date = NULL,
  files_manual = NULL
)
```

Arguments

`input`	Path to directory with .ab1 files.
`export_html`	(Default=TRUE) Output the results as an HTML file
`export_csv`	(Default=TRUE) Output the results as a CSV file.
`export_fasta`	(Default=TRUE) Output the sequences in a FASTA file.
`export_fasta_revcomp`	(Default=FALSE) Output the sequences in reverse complement form in a fasta file. This is useful in cases where sequencing was done using the reverse primer and thus the orientation of input sequences needs reversing.

verbose (Default=FALSE) Output progress while script is running.

exclude (Default=NULL) For testing purposes only. Excludes files of interest from input directory.

min_phred_score (Default=20) Do not accept trimmed sequences with a mean Phred score below this cutoff

min_length (Default=200) Do not accept trimmed sequences with sequence length below this number

sliding_window_cutoff (Default=NULL) Quality trimming parameter (M2) for wrapping SangerRead function in sangeranalyseR package. If NULL, implements auto cutoff for Phred score (recommended), otherwise set between 1-60.

sliding_window_size (Default=15) Quality trimming parameter (M2) for wrapping SangerRead function in sangeranalyseR package. Recommended range between 5-30.

date Set date "YYYY_MM_DD" format. If NULL, attempts to parse date from .ab1 file

files_manual (Default=NULL) For testing purposes only. Specify a list of files to run as filenames without extensions, rather than the whole directory format. Primarily used for testing, use at your own risk.

Value

Returns quality trimmed Sanger sequences in FASTA format.

See Also

isoTAX, isoLIB

Examples

```
#Set path to directory containing example .ab1 files
fpath1 <- system.file("extdata/abif_examples/rocket_salad", package = "isolateR")

#Step 1: Run isoQC function with default settings
isoQC.S4 <- isoQC(input=fpath1)

#Show summary statistics
isoQC.S4
```

---

isoTAX *Classify taxonomy of sequences after quality trimming steps.*

Description

This function performs taxonomic classification steps by searching query Sanger sequences against specified database of interest. Takes CSV input files, extracts FASTA-formatted query sequences and performs global alignment against specified database of interest via Needleman-Wunsch algorithm by wrapping the –usearch_global command implemented in VSEARCH. Default taxonomic rank cutoffs for 16S rRNA gene sequences are based on Yarza et al. 2014, Nat Rev Microbiol.

Usage

```
isoTAX(
  input = NULL,
  export_html = TRUE,
  export_csv = TRUE,
  quick_search = TRUE,
  db = "16S",
  iddef = 2,
  phylum_cutoff = 75,
  class_cutoff = 78.5,
  order_cutoff = 82,
  family_cutoff = 86.5,
  genus_cutoff = 96.5,
  species_cutoff = 98.7
)
```

Arguments

input	Path of CSV output file from isoQC step.
export_html	(Default=TRUE) Output the results as an HTML file
export_csv	(Default=TRUE) Output the results as a CSV file.
quick_search	(Default=FALSE) Whether or not to perform a comprehensive database search (i.e. optimal global alignment). If TRUE, performs quick search equivalent to setting VSEARCH parameters "–maxaccepts 100 –maxrejects 100". If FALSE, performs comprehensive search equivalent to setting VSEARCH parameters "–maxaccepts 0 –maxrejects 0"
db	(Default="16S") Select database option(s) including "16S" (for searching against the NCBI Refseq targeted loci 16S rRNA database), "ITS" (for searching against the NCBI Refseq targeted loci ITS database. For combined databases in cases where input sequences are dervied from bacteria and fungi, select "16S|ITS".
iddef	Set pairwise identity definition as per VSEARCH definitions (Default=2, and is recommended for highest taxonomic accuracy) (0) CD-HIT definition: (matching columns) / (shortest sequence length). (1) Edit distance: (matching columns) / (alignment length). (2) Edit distance excluding terminal gaps (default definition). (3) Marine Biological Lab definition counting each gap opening (internal or terminal) as a single mismatch, whether or not the gap was extended: 1.0-((mismatches + gap openings)/(longest sequence length)). (4) BLAST definition, equivalent to –iddef 1 for global pairwise alignments.
phylum_cutoff	Percent cutoff for phylum rank demarcation
class_cutoff	Percent cutoff for class rank demarcation
order_cutoff	Percent cutoff for order rank demarcation
family_cutoff	Percent cutoff for family rank demarcation
genus_cutoff	Percent cutoff for genus rank demarcation
species_cutoff	Percent cutoff for species rank demarcation

Value

Returns taxonomic classification table of class isoTAX. Default taxonomic cutoffs for phylum (75.0), class (78.5), order (82.0), family (86.5), genus (96.5), and species (98.7) demarcation are based on Yarza et al. 2014, Nature Reviews Microbiology (DOI:10.1038/nrmicro3330)

See Also

isoQC, isoLIB, search_db

Examples

```
#Set path to directory containing example .ab1 files
fpath1 <- system.file("extdata/abif_examples/rocket_salad", package = "isolateR")

#Step 1: Run isoQC function with default settings
isoQC.S4 <- isoQC(input=fpath1)

#Step 2: Run isoTAX function with default settings
fpath2 <- file.path(fpath1, "isolateR_output/01_isoQC_trimmed_sequences_PASS.csv")
isoTAX.S4 <- isoTAX(input=fpath2)
#Show summary statistics
isoTAX.S4
```

---

make_fasta *Convert CSV file containing sequences to FASTA format*

Description

This function extracts sequences from a table in CSV format and converts them to FASTA format. Requires two columns, one with sequences and one with sequence names.

Usage

```
make_fasta(
  csv_file = NULL,
  col_names = "ID",
  col_seqs = "Sequence",
  output = "output.fasta"
)
```

Arguments

csv_file	Filename (or path and filename if not in working directory) of the table from which you would like to generate a FASTA file.
col_names	Column name with the unique names/identifiers. (Default="ID")
col_seqs	Column name with the sequences. (Default="Sequence")
output	Desired filename for output FASTA file (Default = "output.fasta")

Value

Returns sequences in FASTA format.

Examples

```
#Set path to directory containing example .ab1 files
fpath1 <- system.file("extdata/abif_examples/rocket_salad", package = "isolateR")

#Run isoQC function with default settings to generate CSV file
isoQC.S4 <- isoQC(input=fpath1)

#Set path of CSV output file from isoQC step
csv.path <- file.path(fpath1, "isolateR_output/01_isoQC_trimmed_sequences_PASS.csv")

#Run make_fasta function
make_fasta(csv_file= csv.path, col_names="filename", col_seqs="seqs_trim", output="output.fasta")
```

---

make_tree *Generate a phylogenetic tree from an isoLIB output file*

Description

This script will help the user make a simple phylogenetic tree from a strain library. It will allow the user to colour the tree by taxonomic rank only. See ggtree documentation for more information on customization options available.

Usage

```
make_tree(input = NULL)
```

Arguments

input	Full path to isoLIB strain library output file in .CSV format.

Value

Returns a ggtree class object

See Also

isoLIB

Examples

```
#Set path to directory containing example .ab1 files
fpath1 <- system.file("extdata/abif_examples/rocket_salad", package = "isolateR")

#Step 1: Run isoQC function with default settings
isoQC.S4 <- isoQC(input=fpath1)

#Step 2: Run isoTAX function with default settings
fpath2 <- file.path(fpath1, "isolateR_output/01_isoQC_trimmed_sequences_PASS.csv")
isoTAX.S4 <- isoTAX(input=fpath2)

#Step 3: Run isoLIB function with default settings
fpath3 <- file.path(fpath1, "isolateR_output/02_isoTAX_results.csv")
isoLIB.S4 <- isoLIB(input=fpath3)
```

```
#Step 4: Make a tree from isoLIB output CSV file
fpath4 <- file.path(fpath1, "isolateR_output/03_isoLIB_results.csv")
make_tree(input= fpath4)
```

`method-isoLIB` *setMethod functions for isoLIB*

Description

Initiation of isoLIB functions.

Usage

```
## S4 method for signature 'missing'
isoLIB(
  input = NULL,
  old_lib_csv = NULL,
  export_html = TRUE,
  export_csv = TRUE,
  include_warnings = FALSE,
  strain_group_cutoff = 0.995,
  phylum_cutoff = 75,
  class_cutoff = 78.5,
  order_cutoff = 82,
  family_cutoff = 86.5,
  genus_cutoff = 96.5,
  species_cutoff = 98.7
)
```

`method-isoQC` *setMethod functions for isoQC*

Description

Initiation of isoQC functions.

Usage

```
## S4 method for signature 'missing'
isoQC(
  input = NULL,
  export_html = TRUE,
  export_csv = TRUE,
  export_fasta = TRUE,
  export_fasta_revcomp = FALSE,
  verbose = FALSE,
  exclude = NULL,
  min_phred_score = 20,
  min_length = 200,
  sliding_window_cutoff = NULL,
```

```
  sliding_window_size = 15,
  date = NULL,
  files_manual = NULL
)
```

`method-isoTAX` *setMethod functions for isoTAX*

Description

Initiation of isoTAX functions.

Usage

```
## S4 method for signature 'missing'
isoTAX(
  input = NULL,
  export_html = TRUE,
  export_csv = TRUE,
  quick_search = TRUE,
  db = "16S",
  iddef = 2,
  phylum_cutoff = 75,
  class_cutoff = 78.5,
  order_cutoff = 82,
  family_cutoff = 86.5,
  genus_cutoff = 96.5,
  species_cutoff = 98.7
)
```

`S4_to_dataframe` *Converts S4 objects (isoQC, isoTAX, or isoLIB) to dataframe*

Description

Helper function to convert S4 class objects (`isoQC`, `isoTAX`, or `isoLIB`) to dataframe

Usage

```
S4_to_dataframe(obj)
```

Arguments

`obj`	S4 object generated from `isoQC`, `isoTAX`, or `isoLIB` steps

Value

Returns a dataframe containing sequence information in columns.

search_db *Perform global alignment pairwise identity search using VSEARCH and type strain database of interest.*

Description

Performs global alignment between FASTA-formatted query sequences and the specified database of interest. Uses the Needleman-Wunsch algorithm by wrapping the –usearch_global command implemented in VSEARCH.

Usage

```
search_db(
  query.path = NULL,
  uc.out = "VSEARCH_output.uc",
  b6.out = "VSEARCH_output.b6o",
  path = getwd(),
  quick_search = FALSE,
  db.path = NULL,
  db = NULL,
  keep_temp_files = FALSE,
  iddef = 2
)
```

Arguments

query.path	Path of FASTA-formatted query sequence file.
uc.out	Path of UC-formatted results output table.
b6.out	Path of blast6-formatted results output table.
path	Working path directory (Default is set to current working directory via 'getwd()'
quick_search	(Default=FALSE) Whether or not to perform a comprehensive database search (i.e. optimal global alignment). If TRUE, performs quick search equivalent to setting VSEARCH parameters "–maxaccepts 100 –maxrejects 100". If FALSE, performs comprehensive search equivalent to setting VSEARCH parameters "–maxaccepts 0 –maxrejects 0" Note: This option is provided for convenience and rough approximation of taxonomy only, set to FALSE for accurate % pairwise identity results.
db.path	Path of FASTA-formatted database sequence file. Ignored if 'database' parameter is set to anything other than NULL
db	Optional: Select any of the standard database option(s) including "16S" (for searching against the NCBI Refseq targeted loci 16S rRNA database), "ITS" (for searching against the NCBI Refseq targeted loci ITS database. For combined databases in cases where input sequences are dervied from bacteria and fungi, select "16S|ITS". Setting to anything other than NULL causes 'db.path' parameter to be ignored.
keep_temp_files	Toggle (TRUE/FALSE). If TRUE, temporary .uc and .b6o output files are kept from VSEARCH –uc and –blast6out commands, respectively. If FALSE, temporary files are removed.

iddef — Set pairwise identity definition as per VSEARCH definitions (Default=2, and is recommended for highest taxonomic accuracy) (0) CD-HIT definition: (matching columns) / (shortest sequence length). (1) Edit distance: (matching columns) / (alignment length). (2) Edit distance excluding terminal gaps (default definition). (3) Marine Biological Lab definition counting each gap opening (internal or terminal) as a single mismatch, whether or not the gap was extended: 1.0-((mismatches + gap openings)/(longest sequence length)). (4) BLAST definition, equivalent to –iddef 1 for global pairwise alignments.

Value

Returns a dataframe matching the UC-formatted output table from VSEARCH. Query sequences are automatically added to the final column. Summary of column information. See VSEARCH documentation for more details.

- V1 = Record type of hit (H) or no hit (N)
- V2 = Ordinal number of the target sequence (based on input order, starting from zero). Set to '*' for N.
- V3 = Sequence length. Set to '*' for N.
- V4 = Percentage of similarity with the target sequence. Set to '*' for N.
- V5 = Match orientation + or -. . Set to '.' for N.
- V6 = Not used, always set to zero for H, or '*' for N.
- V7 = Not used, always set to zero for H, or '*' for N.
- V8 = Compact representation of the pairwise alignment using the CIGAR format (Compact Idiosyncratic Gapped Alignment Report): M (match/mismatch), D (deletion) and I (insertion). The equal sign '=' indicates that the query is identical to the centroid sequence. Set to '*' for N.
- V9 = Label of the query sequence. Equivalent to 'filename' slot of isolateR class objects (e.g. isoQC, isoTAX, isoLIB).
- V10 = Label of the target centroid sequence. Set to '*' for N.

See Also

isoTAX

Examples

```
#Set path to directory containing example .ab1 files
fpath1 <- system.file("extdata/abif_examples/rocket_salad", package = "isolateR")

#Run isoQC function with default settings
isoQC.S4 <- isoQC(input=fpath1)

#Set path of CSV output file containing PASS sequences from isoQC step
fasta.path <- "01_isoQC_trimmed_sequences_PASS.fasta"

#Set paths
output.path <- file.path(fpath1, "isolateR_output")

#Run search_db function
uc.df <- search_db(query.path=fasta.path, path=output.path, quick_search=TRUE, db="16S")
```

```
#Inspect results
uc.df[1:10,1:10]
```

---

show *Generic show method for S4 class objects*

Description

Generic show method for S4 class objects.

Usage

```
## S4 method for signature 'isoQC'
show(object)

## S4 method for signature 'isoTAX'
show(object)

## S4 method for signature 'isoLIB'
show(object)
```

---

valid_tax_check *Validate species name via API client of LPSN*

Description

This function will determine if each species in a CSV file is validly published or not. Result file will be a CSV with the results appended to the input data. This function requires the user to have an LPSN API account setup. For more details and to register, see here: https://api.lpsn.dsmz.de/)

Usage

```
valid_tax_check(input = NULL, col_species = "species", export_csv = TRUE)
```

Arguments

input	CSV file path. Expects full path if CSV file is not in the current working directory.
col_species	Specify the column containing the binomial species names (e.g. "Akkermansia muciniphila")
export_csv	Toggle (TRUE/FALSE). Set TRUE to automatically write .CSV file of results to current directory. (Default=TRUE)

Value

Returns a dataframe containing

Index