

# Package ‘concatipede’

October 12, 2022

**Title** Easy Concatenation of Fasta Sequences

**Version** 1.0.1

**Description** Concatenation of multiple sequence alignments based on a correspondence table that can be edited in Excel <[doi:10.5281/zenodo.5130603](https://doi.org/10.5281/zenodo.5130603)>.

**License** MIT + file LICENSE

**URL** <https://github.com/tardipede/concatipede>,  
<https://tardipede.github.io/concatipede/>

**BugReports** <https://github.com/tardipede/concatipede/issues>

**Encoding** UTF-8

**RoxygenNote** 7.1.1

**Imports** ape, dplyr, igraph, qualV, magrittr, tibble, readxl,  
stringdist, stringr, writexl

**Suggests** DT, knitr, rmarkdown, tidyverse, testthat (>= 3.0.0)

**VignetteBuilder** knitr, rmarkdown

**SystemRequirements** GNU make

**Config/testthat/edition** 3

**NeedsCompilation** no

**Author** Matteo Vecchi [aut, cre] (<<https://orcid.org/0000-0002-7995-6827>>),  
Mattieu Bruneaux [aut] (<<https://orcid.org/0000-0001-6997-192X>>)

**Maintainer** Matteo Vecchi <[matteo.vecchi15@gmail.com](mailto:matteo.vecchi15@gmail.com)>

**Repository** CRAN

**Date/Publication** 2021-08-06 18:10:05 UTC

## R topics documented:

auto_match_seqs . . . . .	2
concatipede . . . . .	3
concatipede_example . . . . .	4
concatipede_prepare . . . . .	5

find_fasta . . . . .	6
get_genbank_table . . . . .	7
read_xl . . . . .	8
rename_sequences . . . . .	8
write_aln . . . . .	9
write_xl . . . . .	10

<b>Index</b>	<b>11</b>
--------------	-----------

---

auto_match_seqs	<i>Build a template table with automatically matched sequence names</i>
-----------------	-------------------------------------------------------------------------

---

## Description

The algorithm used to match sequences across fasta files based on their names is outlined below.

## Usage

```
auto_match_seqs(x, method = "lv", xlsx)
```

## Arguments

x	A table (data frame or tibble) typically produced by <a href="#">concatipede_prepare</a> . It must be of the same format as a table returned by this function: a first column called "name" followed by one column per fasta file. Those columns have the name of their corresponding fasta file, and they contain the names of the sequences in this file, with one sequence name per cell. The number of rows in the number of sequences of the fasta file with the most sequences, and the columns for the other fasta files are filled with NA for padding.
method	Method for string distance calculation. See <code>?stringdist::stringdist-metrics</code> for details. Default is "lv".
xlsx	Optional, a path to use to save the output table as an Excel file.

## Details

Let's assume a situation with N fasta files, with each fasta file i having  $n_i$  sequence names. The problem of matching the names in the best possible way across the fasta files is similar to that of identifying homologous proteins across species, using e.g. reciprocal blast.

The algorithm steps are:

- For each pair of fasta files, identify matching names using a reciprocal match approach: two names match if and only if they are their reciprocal best match.
- Those matches across fasta files define a graph.
- We identify sub-graphs such that (i) they contain at most one sequence name per fasta file and (ii) all nodes in a given sub-graph are fully connected (i.e., they are all their best reciprocal matches across any pair of fasta files).

**Value**

A table (tibble) with the same columns as `x` and with sequence names automatically matched across fasta files. Sequence names which did not have a best reciprocal match in other fasta files are appended to the end of the table, so that the output table columns contain all the unique sequence names present in the corresponding column of the input table. The first column, "name", contains a suggested name for the row (not guaranteed to be unique). If a path was provided to the `xlsx` argument, an Excel file is saved and the table is returned invisibly.

**Examples**

```
xlsx_file <- concatipede_example("sequences-test-matching.xlsx")
xlsx_template <- readxl::read_xlsx(xlsx_file)
auto_match_seqs(xlsx_template)
## Not run:
  auto_match_seqs(xlsx_template, xlsx = "my-automatic-output.xlsx")

## End(Not run)
```

---

concatipede	<i>Concatenate alignments</i>
-------------	-------------------------------

---

**Description**

This function concatenate sequences from alignments present in the working directory based on a correspondence table and saves the output in a new directory

**Usage**

```
concatipede(
  df = NULL,
  filename = NULL,
  format = c("fasta", "nexus", "phylip"),
  dir,
  plotting = FALSE,
  out = NULL,
  remove.gaps = TRUE,
  write.outputs = TRUE,
  save.partitions = TRUE,
  excel.sheet = 1
)
```

**Arguments**

<code>df</code>	The user-defined correspondence table, as a data frame or equivalent. This is used only if no <code>filename</code> argument is provided.
<code>filename</code>	Filename of input correspondence table. Alternatively, if no <code>filename</code> is provided, the user can provide their own correspondence table as the <code>df</code> argument.

format	a string specifying in what formats you want the alignment
dir	Optional, path to the directory containing the fasta files. This argument has an effect only if fasta files names are taken from the columns of the df argument, and that df does not have an attribute dir_name itself. If no dir is provided and df does not have a dir_name attribute, the current working directory is used with a warning.
ploting	Logical, save a graphical representation of the alignment in pdf format. Default: FALSE.
out	specify outputs filenames
remove.gaps	Logical, remove gap only columns. Useful if not using all sequences in the alignments. Default: TRUE.
write.outputs	Logical, save concatenated alignment, partitions position table and graphical representation. If FALSE it overrides ploting. Default: TRUE.
save.partitions	Logical, save in the concatenated alignment directory a text file with partitions limits for the concatenated alignment. Default: TRUE.
excel.sheet	specify what sheet from the excel spreadsheet has to be read. Either a string (the name of a sheet), or an integer (the position of the sheet).

**Value**

The concatenated alignment (invisibly if out is not NULL).

**Examples**

```
dir <- system.file("extdata", package = "concatipede")
z <- concatipede(filename = paste0(dir, "/Macrobiotidae_seqnames.xlsx"), dir = dir,
                 write.outputs = FALSE)
z
```

---

concatipede\_example    *Get the path to one of concatipede example files*

---

**Description**

Several example files are shipped with the concatipede package. This function facilitates the access to those files.

**Usage**

```
concatipede_example(example_file = NULL)
```

**Arguments**

example\_file    Basename of the target example file. If NULL (the default), the basenames of the available files are listed.

## Details

**COI\_Macrobiotidae.fas** Example fasta file.

**ITS2\_Macrobiotidae.fas** Example fasta file.

**LSU\_Macrobiotidae.fas** Example fasta file.

**SSU\_Macrobiotidae.fas** Example fasta file.

**sequences-test-matching.xlsx** This is an Excel file (extension .xlsx) typically used to test or demonstrate the automatic matching capabilities of the concatipede package. This file represents the Excel template that could be produced by `concatipede_prepare` after detecting the fasta files present in a working directory.

**Macrobiotidae\_seqnames.xlsx** This is an Excel file (extension .xlsx) that contains the correspondence table that can be used to concatenate the sequences contained in the example fasta files COI\_Macrobiotidae.fas, ITS2\_Macrobiotidae.fas, LSU\_Macrobiotidae.fas, and SSU\_Macrobiotidae.fas.

## Value

The full path to access the example file, or a list of available example files if no `example_file` argument was provided.

## Examples

```
concatipede_example()
example <- concatipede_example("sequences-test-matching.xlsx")
if (requireNamespace("readxl")) {
  seqs <- readxl::read_xlsx(example)
  seqs
}
```

---

<code>concatipede_prepare</code>	<i>Load alignments and prepare template correspondence table for concatenate () function</i>
----------------------------------	----------------------------------------------------------------------------------------------

---

## Description

This function creates a template correspondence table that can also be saved in the working directory.

## Usage

```
concatipede_prepare(fasta_files, out = "seqnames", excel = TRUE, exclude)
```

**Arguments**

fasta_files	Optional, a vector of paths to the fasta files that should be merged. If this argument is missing, the function automatically detects and uses all the fasta files present in the working directory.
out	Optional, a filename for the correspondence table template to save (without extension). No file is saved if out is not provided. In all cases, the function also returns a tibble with the correspondence table template (invisibly if out is provided).
excel	Boolean, should the correspondence table template be saved in excel format? If FALSE, it is saved as a tab-separated text file instead. Default is TRUE. The correct file extension is automatically appended to the out argument. If out is missing, this argument has no effect.
exclude	If no fasta_files argument has been passed, fasta files matching the exclude pattern will be ignored by the function when it automatically detects fasta files in the working directory.

**Value**

A tibble with the correspondence table template (invisibly if an out argument was provided to save the table to a file).

**Examples**

```
dir <- system.file("extdata", package = "concatipede")
fasta_files <- find_fasta(dir)
z <- concatipede_prepare(fasta_files)
z
```

---

find_fasta	<i>Find fasta files present in a folder</i>
------------	---------------------------------------------

---

**Description**

Find fasta files present in a folder

**Usage**

```
find_fasta(dir, pattern = "\\fa$|\\fas$|\\fasta$", exclude)
```

**Arguments**

dir	Path to the directory which should be examined. If not provided, the current working directory (as returned by <code>getwd</code> ) is used.
pattern	Regular expression used by <code>list.files</code> to detect the fasta files. The default is to list all files ending in ".fa", ".fas", and ".fasta".
exclude	Optional regular expression used to exclude some filenames from the list of detected files.

**Value**

A vector with the full paths to the found files.

**Examples**

```
# Get the directory containing the package example files
dir <- system.file("extdata", package = "concatipede")
# List the fasta files containing in that directory
find_fasta(dir)
# Exclude some files
find_fasta(dir, exclude = "COI")
```

---

get_genbank_table	<i>Extract GenBank accession number from correspondence table</i>
-------------------	-------------------------------------------------------------------

---

**Description**

Extract GenBank accession number from correspondence table formatted with the same requirements for concatipede()

**Usage**

```
get_genbank_table(
  df = NULL,
  filename = NULL,
  writetable = FALSE,
  out = "",
  excel.sheet = 1
)
```

**Arguments**

df	The user-defined correspondence table, as a data frame or equivalent. This is used only if no filename argument is provided.
filename	Filename of input correspondence table. Alternatively, if no filename is provided, the user can provide their own correspondence table as the df argument.
writetable	if TRUE save the Genbank table as excel file in the working directory
out	if writetable == T, the name to be attached to the excel filename
excel.sheet	specify what sheet from the excel spreadsheet you wanna read. Either a string (the name of a sheet), or an integer (the position of the sheet).

**Value**

Table with GenBank accession numbers

---

read_xl	<i>Read an Excel file</i>
---------	---------------------------

---

### Description

This function loads a table from an Excel file.

### Usage

```
read_xl(path, sheet = 1)
```

### Arguments

path	The path to the Excel file to read.
sheet	Optional, the sheet to read (either a string with the name of the sheet or an integer with its position). Default: 1.

### Value

A tibble.

---

rename_sequences	<i>Rename sequences</i>
------------------	-------------------------

---

### Description

This function renames sequences in fasta files based on a correspondence table.

### Usage

```
rename_sequences(  
  fasta_files,  
  df = NULL,  
  filename = NULL,  
  marker_names = NULL,  
  out = NULL,  
  format = "fasta",  
  excel.sheet = 1,  
  unalign = FALSE,  
  exclude  
)
```



**Arguments**

fasta_files	Optional, a vector of paths to the fasta files that should be renamed. If this argument is missing, the function automatically detects and uses all the fasta files present in the working directory.
df	The user-defined correspondence table, as a data frame or equivalent. This is used only if no filename argument is provided.
filename	Filename of correspondence table. Alternatively, if no filename is provided, the user can provide their own correspondence table as the df argument.
marker_names	the name of the marker for each alignment to be appended at the end of the sequences names, in the same order as in the correspondence table
out	specify outputs filename
format	a string specifying in what formats you want the alignment. Can be "fasta", "phylip" and "nexus"
excel.sheet	specify what sheet from the excel spreadsheet you wanna read. Either a string (the name of a sheet), or an integer (the position of the sheet).
unalign	return unaligned fasta files as output
exclude	Optional regular expression used to exclude some filenames from the list of detected files.

**Value**

No return value, called for side effect of saving a correspondence table.

---

write_aln	<i>Writing alignments</i>
-----------	---------------------------

---

**Description**

Alignments can be saved in fasta, nexus, and phylip formats.

**Usage**

```
write_fasta(x, path)
```

```
write_nexus(x, path)
```

```
write_phylip(x, path)
```

**Arguments**

x	Alignment to save (an object of class DNABin).
path	Path of the file to be written, without file extension (the appropriate extension is added automatically, i.e. the path will be extended with ".fasta", ".nexus", or ".phy" depending on the file format used).

**Value**

The input x (invisibly).

**Examples**

```
## Not run:
# Path to an example alignment file
pkg_aln <- concatipede_example("COI_Macrobiotidae.fas")
# Load the alignment into the R session
aln <- ape::read.FASTA(pkg_aln)
# Write the alignment in various formats
# Note that the appropriate file extension is added by the writing functions.
write_fasta(aln, "my-alignment")
write_nexus(aln, "my-alignment")
write_phylip(aln, "my-alignment")

## End(Not run)
```

---

write\_xl

*Write an Excel file*

---

**Description**

This function writes an input table to an Excel file and returns its input (invisibly).

**Usage**

```
write_xl(x, path)
```

**Arguments**

x	A data frame or tibble to write to an Excel file.
path	The path to the Excel file to be written.

**Value**

The input table x, invisibly (so that the function can be part of a pipeline with the pipe operator).

# Index

`auto_match_seqs`, 2

`concatipede`, 3

`concatipede_example`, 4

`concatipede_prepare`, 2, 5, 5

`find_fasta`, 6

`get_genbank_table`, 7

`getwd`, 6

`list.files`, 6

`read_xl`, 8

`rename_sequences`, 8

`write_aln`, 9

`write_fasta(write_aln)`, 9

`write_nexus(write_aln)`, 9

`write_phylip(write_aln)`, 9

`write_xl`, 10