

Package: delimtools (via r-universe)

October 10, 2026

Type Package

Title Helper Functions for Species Delimitation Analysis

Version 0.3.5

Description Helpers functions to process, analyse, and visualize the output of single locus species delimitation methods.

License AGPL-3

Depends R (>= 4.2.0)

Imports ape (>= 5.0), cli, dplyr, glue, ggtree, grDevices, methods, purrr, rlang, stats, stringr, tidyr, tidyselect, tools, utils

Suggests forcats, furr, future, ggfun, ggplot2, gtools, here, knitr, patchwork, randomcoloR, readr, RColorBrewer, rmarkdown, scales, spider, tibble, tidytree, vctrs, withr

Encoding UTF-8

LazyData true

RoxygenNote 8.0.0

Roxygen list(markdown = TRUE)

NeedsCompilation yes

URL <https://github.com/legalLab/delimtools>,
<https://legallab.github.io/delimtools/>

BugReports <https://github.com/legalLab/delimtools/issues>

VignetteBuilder knitr

Config/pak/sysreqs
libcairo2-dev cmake libfontconfig1-dev libfreetype6-dev make libicu-dev libpng-dev libuv1-dev

Repository <https://legallab.r-universe.dev>

Date/Publication 2026-10-10 14:50:03 UTC

RemoteUrl <https://github.com/legallab/delimtools>

RemoteRef HEAD

RemoteSha 7233273cdeaffbd7aec5a01968545995a406df7c

Contents

delimtools-package	3
abgd	4
abgd_tbl	6
as_dwc	7
asap	9
asap_tbl	10
best_partition	12
bgmyc	13
bgmyc_tbl	15
check_delim	16
check_identifiers	17
clean_dna	18
collapse_others	19
confidence_intervals	20
delim_autoplot	23
delim_autoplot2	24
delim_brewer	26
delim_consensus	27
delim_join	28
delim_tbl	29
drop_sequences	30
dwc_terms	31
geophagus	32
geophagus_beast	32
geophagus_bootstraps	33
geophagus_delims	33
geophagus_info	34
geophagus_posterior	35
geophagus_raxml	35
get_delim_cols	36
get_dwc	37
gmyc	38
gmyc_tbl	38
hap_collapse	39
hap_unite	40
haplotype_tbl	41
locmin_tbl	42
match_ratio	43
min_brlen	44
morph_tbl	45
mptp	46
mptp_mcmc	48
mptp_support	50
mptp_tbl	50
report_delim	52
spec.list	53

<i>delimtools-package</i>	3
summary.bgmyc_fit	54
Index	55

Description

Helpers functions to process, analyse, and visualize the output of single locus species delimitation methods. For full functionality, please install suggested software at <https://legallab.github.io/delimtools/articles/install.html>.

Author(s)

Maintainer: Pedro Bittencourt <pedro.sennabittencourt@gmail.com> ([ORCID](#)) [copyright holder]

Authors:

- Pedro Bittencourt <pedro.sennabittencourt@gmail.com> ([ORCID](#)) [copyright holder]
- Rupert Collins <rupertcollins@gmail.com> ([ORCID](#)) [contributor, copyright holder]
- Tomas Hrbek <hrbek@evoamazon.net> ([ORCID](#)) [contributor]

Other contributors:

- Tomas Flouri (mPTP C engine (AGPL-3)) [contributor]
- Sarah Lutteropp (mPTP C engine (AGPL-3)) [contributor]
- Paschalia Kapli (mPTP C engine (AGPL-3)) [contributor]

See Also

Useful links:

- <https://github.com/legalLab/delimtools>
- <https://legallab.github.io/delimtools/>
- Report bugs at <https://github.com/legalLab/delimtools/issues>

abgd

Automatic Barcode Gap Discovery (ABGD)

Description

Delimits species using the original algorithm of Puillandre et al. (2012). All computation (distances, gap detection, partitioning and recursion) is performed by the original C code of G. Achaz via `.Call()`.

Usage

```
abgd(
  x,
  model = "simple",
  prior_min = 0.001,
  prior_max = 0.1,
  prior_steps = 10L,
  min_slope_increase = 1.5,
  ts_tv = 2
)
```

Arguments

<code>x</code>	Path to an aligned FASTA file, or an aligned DNAbin (ape) object. Distance matrices are not accepted; distances are calculated internally from the sequences. A DNAbin object is exported to a temporary FASTA file and processed by the original C code.
<code>model</code>	Distance model: "simple" p-distance with Laplace correction – equivalent to <code>-d 3</code> in the original ABGD. Use this to replicate <code>abgd -a -d 3</code> . "JC69" Jukes-Cantor (original default, <code>-d 1</code>). "K80" Kimura 2-parameter (<code>-d 0</code>). "TN93" Tamura-Nei (<code>-d 2</code> , <i>not implemented</i>).
<code>prior_min</code>	Minimum prior for intraspecific divergence (<code>-p</code>). Default: 0.001.
<code>prior_max</code>	Maximum prior for intraspecific divergence (<code>-P</code>). Default: 0.1.
<code>prior_steps</code>	Number of steps in $[p, P]$ (<code>-n</code>). Default: 10.
<code>min_slope_increase</code>	Minimum slope increase factor (<code>-X</code>). Default: 1.5.
<code>ts_tv</code>	Transition/transversion ratio for K80 (<code>-t</code>). Default: 2.0.

Value

A named list with the following elements:

`initial` A data.frame with one row per prior (initial partition): `prior`, `n_groups`, `partition`.

`recursive` A data.frame with one row per prior (recursive partition): `prior`, `n_groups`, `partition`.

`dist_matrix` Numeric distance matrix (nseq x nseq).

`taxa` Character vector of sequence names.

AI Disclaimer

This function was written with assistance of AI coding agent (Claude Code Sonnet 4.6). Correctness was validated by comparing against original software, on a suite of real sequencing datasets, and manual code review. All validation and output verification was made by the authors.

Source Code

N. Puillandre, A. Lambert, S. Brouillet, G. Achaz (ABGD C engine)

Author(s)

Pedro S. Bittencourt

References

Puillandre N, Lambert A, Brouillet S, Achaz G (2012). ABGD, Automatic Barcode Gap Discovery for primary species delimitation. *Molecular Ecology*, 21(8), 1864–1877. doi:[10.1111/j.1365-294X.2011.05239.x](https://doi.org/10.1111/j.1365-294X.2011.05239.x)

Examples

```
## Not run:
# Equivalent to: abgd -a -d 3 geophagus.fasta
result <- abgd("geophagus.fasta", model = "simple")

result$initial[, c("prior", "n_groups")]
result$recursive[, c("prior", "n_groups")]
result$recursive$partition[[1]] # partition for the first prior

# Input from a DNABin object
library(ape)
seqs <- read.dna("geophagus.fasta", format = "fasta")
result <- abgd(seqs, model = "simple")

## End(Not run)
```

abgd_tbl	<i>A Command-Line Interface for ABGD - Automatic Barcode Gap Discovery</i>
----------	--

Description

abgd_tbl() returns species partition hypothesis estimated by ABGD software (<https://bioinfo.mnhn.fr/abi/public/abgd/>).

Usage

```
abgd_tbl(
  infile,
  exe = NULL,
  haps = NULL,
  slope = 1.5,
  model = 3,
  outfolder = NULL,
  webserver = NULL,
  delimname = "abgd",
  type = c("recursive", "initial")
)
```

Arguments

infile	Path to a FASTA file, or an object of class "abgd" returned by abgd() . When an "abgd" object is supplied, the CLI path is skipped and best_partition() is called directly.
exe	Path to an ABGD executable. Ignored when infile is an "abgd" object.
haps	Optional. A vector of haplotypes to keep into the tbl_df .
slope	Numeric. Relative gap width (slope). Default to 1.5.
model	An integer specifying evolutionary model to be used. Available options are: • 0: Kimura-2P • 1: Jukes-Cantor (default) • 2: Tamura-Nei (<i>not implemented</i>) • 3: simple distance (p-distance)
outfolder	Path to output folder. Default to NULL. If not specified, a temporary location is used.
webserver	A .txt file containing ABGD results obtained from a webserver. Default to NULL.
delimname	Character. String to rename the delimitation method in the table. Default to 'abgd'.
type	Which ABGD pass to use when infile is an "abgd" object: "recursive" (default) or "initial". Ignored for CLI and webserver paths.

Details

abgd_tbl() relies on [system](#) to invoke ABGD software through a command-line interface. Hence, you must have the software available as an executable file on your system in order to use this function properly. abgd_tbl() saves all output files in outfolder and imports the first recursive partition file generated to Environment. Alternatively, abgd_tbl() can parse a .txt file obtained from a webserver such as (<https://bioinfo.mnhn.fr/abi/public/abgd/abgdweb.html>).

Value

an object of class [tbl_df](#)

Author(s)

Pedro S. Bittencourt, Tomas Hrbek

References

Puillandre N., Lambert A., Brouillet S., Achaz G. 2012. ABGD, Automatic Barcode Gap Discovery for primary species delimitation. *Molecular Ecology* 21(8):1864-77. doi:[10.1111/j.1365-294X.2011.05239.x](https://doi.org/10.1111/j.1365-294X.2011.05239.x)

Examples

```
#' # get path to fasta file
path_to_file <- system.file("extdata/geophagus.fasta", package = "delimtools")

# run ABGD
abgd_df <- try( abgd_tbl(
  infile = path_to_file,
  exe = "/usr/local/bin/abgd",
  model = 3,
  slope = 0.5,
  outfolder = NULL
)
)
# check
try(abgd_df)
```

as_dwc

Rename Columns using Darwin Core Standard Terms

Description

as_dwc() rename columns in a [tbl_df](#) using a vector of terms defined by Darwin Core Standard.

Usage

```
as_dwc(dwc, data, terms)
```

Arguments

dwc	a list of standard terms and definitions created using <code>get_dwc()</code> .
data	a <code>tbl_df</code> .
terms	a vector or list of terms to be used as replacement.

Details

`as_dwc()` will replace current column names by the ones defined in `terms`. For each column in `data`, Darwin Core equivalent terms must be informed in the same order by the user. If `terms` and column names do not match in length or if `terms` used are not found in Darwin Core standard, an error will be printed on Console.

Value

an object of class `tbl_df`.

Author(s)

Pedro S. Bittencourt, Rupert A. Collins.

Examples

```
# get dwc terms and definitions
dwc <- get_dwc(type = "simple")

# create a data frame with sample metadata
my_df <- tibble::tibble(
  species = c("sp1", "sp2", "sp3"),
  location = c("loc1", "loc2", "loc3"),
  voucher = c("M01", "M02", "M03"),
  collector = c("John", "Robert", "David")
)

# rename columns
as_dwc(dwc, my_df, terms = c("scientificName", "locality", "catalogNumber", "recordedBy"))
```

Description

Delimits species from aligned sequences using the algorithm of Puillandre, Brouillet & Achaz (2021). The original C code (`asap_core.c`, `asap_common.c`) is executed without modifications; only the Python/graphical dependencies are replaced.

Usage

```
asap(
  x,
  model = "simple",
  len_seq = NULL,
  replicates = 1000L,
  pvalue_threshold = 0.001,
  slope_weight = 0.1,
  score_weight = 0.5,
  pairwise.deletion = TRUE
)
```

Arguments

<code>x</code>	Path to an aligned FASTA file, or an aligned DNAbin (ape) object. Distance matrices are not accepted; distances are calculated internally from the sequences.
<code>model</code>	Distance model. "simple" (default) uses <code>Simple_Dist</code> , identical to the ASAP original default. Any model from <code>ape::dist.dna()</code> is also accepted.
<code>len_seq</code>	Sequence length for the coalescent simulations. Inferred from the alignment length when NULL (default).
<code>replicates</code>	Coalescent replicates. Default: 1000.
<code>pvalue_threshold</code>	P-value threshold. Default: 0.001.
<code>slope_weight</code>	Slope window weight. Default: 0.1.
<code>score_weight</code>	Weight of the p-value rank in the ASAP-score. Default: 0.5.
<code>pairwise.deletion</code>	Passed to <code>ape::dist.dna()</code> . Default: TRUE.

Value

A list of class "asap" with:

- `partitions` data.frame sorted by ascending ASAP-score.
- `best` List with the best partition (rank 1).
- `dist_matrix` Distance matrix used.
- `taxa` Sequence names.

AI Disclaimer

This function was written with assistance of AI coding agent (Claude Code Sonnet 4.6). Correctness was validated by comparing against original software, on a suite of real sequencing datasets, and manual code review. All validation and output

Source Code

N. Puillandre, A. Lambert, S. Brouillet, G. Achaz (ASAP C engine)

Author(s)

Pedro S. Bittencourt

References

Puillandre N, Brouillet S, Achaz G (2021). ASAP: assemble species by automatic partitions. *Molecular Ecology Resources*, 21(2), 609-620. doi:10.1111/17550998.13281

Examples

```
## Not run:
library(ape)
seqs <- read.dna("barcode.fasta", format = "fasta")
result <- asap(seqs)
result <- asap("barcode.fasta") # aligned FASTA path also accepted
print(result)
bp <- best_partition(result)
bp$n_groups # number of species
bp$partition # assignment for each sequence

## End(Not run)
```

asap_tbl

A Command-Line Interface for ASAP - Assemble Species by Automatic Partitioning

Description

asap_tbl() returns species partition hypothesis estimated by ASAP software (<https://bioinfo.mnhn.fr/abi/public/asap/>).

Usage

```
asap_tbl(
  infile,
  exe = NULL,
  haps = NULL,
  model = 3,
```

```

    outfolder = NULL,
    webserver = NULL,
    delimname = "asap",
    rank = NULL
  )

```

Arguments

infile	Path to a FASTA file, or an object of class "asap" returned by asap() . When an "asap" object is supplied, the CLI path is skipped and <code>best_partition()</code> is called directly.
exe	Path to an ASAP executable. Ignored when infile is an "asap" object.
haps	Optional. A vector of haplotypes to keep into the tbl_df .
model	An integer specifying evolutionary model to be used. Available options are: • 0: Kimura-2P • 1: Jukes-Cantor (default) • 2: Tamura-Nei • 3: simple distance (p-distance)
outfolder	Path to output folder. Default to NULL. If not specified, a temporary location is used.
webserver	A .csv file containing ASAP results obtained from a webserver. Default to NULL.
delimname	Character. String to rename the delimitation method in the table. Default to 'asap'.
rank	Integer. Which ranked partition to return when infile is an "asap" object (default NULL selects rank 1 / best ASAP score). Ignored for CLI and webserver paths.

Details

`asap_tbl()` relies on [system](#) to invoke ASAP software through a command-line interface. Hence, you must have the software available as an executable file on your system in order to use this function properly. `asap_tbl()` saves all output files in `outfolder` and imports the first partition file generated to Environment. Alternatively, `asap_tbl()` can parse a .csv file obtained from webserver such as (<https://bioinfo.mnhn.fr/abi/public/asap/asapweb.html>).

Value

an object of class [tbl_df](#)

Author(s)

Pedro S. Bittencourt, Tomas Hrbek

References

Puillandre N., Brouillet S., Achaz G. 2021. ASAP: assemble species by automatic partitioning. *Molecular Ecology Resources* 21:609–620. doi:[10.1111/17550998.13281](https://doi.org/10.1111/17550998.13281)

Examples

```
#' # get path to fasta file
path_to_file <- system.file("extdata/geophagus.fasta", package = "delimtools")

# run ASAP
asap_df <- try(asap_tbl(infile = path_to_file, exe= "/usr/local/bin/asap", model= 3))

# check
try(asap_df)
```

best_partition	Select the Best Partition from a Delimitation Result
----------------	--

Description

Generic function to extract the most informative partition from an ABGD or ASAP result object.

Usage

```
best_partition(x, ...)

## S3 method for class 'abgd'
best_partition(x, pass = c("recursive", "initial"), ...)

## S3 method for class 'asap'
best_partition(x, rank = NULL, ...)
```

Arguments

x	A delimitation result object ("abgd" or "asap").
...	Additional arguments passed to methods.
pass	Which pass to use: "recursive" (default) or "initial".
rank	Integer. Rank of the partition to return (1 = best ASAP score). If NULL (default), returns rank 1.

Value

A named list describing the selected partition. See method documentation for details.

Examples

```
## Not run:
result <- abgd("geophagus.fasta", model = "simple")
best <- best_partition(result)
best$n_groups

## End(Not run)
```

bgmyc

*Public API Run Bayesian GMYC species delimitation***Description**

Runs a Bayesian implementation of the General Mixed Yule-Coalescent (bGMYC) model using the embedded C GMYC engine. Likelihood evaluations are performed in C (via `C_gmyc_loglik`), eliminating the dependency on the external **bGMYC** package.

Usage

```
bgmyc(
  tree,
  mcmc = 11000L,
  burnin = 1000L,
  thinning = 100L,
  py1 = 0,
  py2 = 2,
  pc1 = 0,
  pc2 = 2,
  t1 = 2L,
  t2 = NULL,
  scale = c(20, 10, 5),
  start = c(1, 0.5, 50),
  ppcutoff = 0.05,
  quiet = FALSE
)
```

Arguments

<code>tree</code>	An ultrametric, fully binary phylo object.
<code>mcmc</code>	Integer. Total MCMC steps (default 11 000).
<code>burnin</code>	Integer. Steps discarded as burn-in (default 1 000).
<code>thinning</code>	Integer. Thinning interval; every thinning-th post-burnin step is retained (default 100).
<code>py1, py2</code>	Numeric. Prior bounds for the Yule rate-change exponent (default 0 and 2).
<code>pc1, pc2</code>	Numeric. Prior bounds for the coalescent rate-change exponent (default 0 and 2).

t1, t2	Integer. Prior bounds for the threshold parameter (number of species). Defaults: t1 = 2, t2 = NULL (auto: Nnode - 1).
scale	Numeric vector of length 3. Proposal scale parameters for py, pc, and t (default c(20, 10, 5)).
start	Numeric vector of length 3. Starting values for py, pc, and t (default c(1, 0.5, 50)).
ppcutoff	Numeric. Posterior co-occurrence probability threshold used to compute the point-estimate \$assignments. Tip pairs with posterior probability of co-occurrence $\geq$ ppcutoff are merged into the same species (default 0.05). Can be recomputed later via bgmyc_tbl .
quiet	Logical. Suppress progress output (default FALSE).

Details

The Metropolis-within-Gibbs sampler follows Reid & Carstens (2012): a gamma proposal for py and pc, and a Gaussian random walk for the integer threshold t. The prior is uniform on $[py1, py2] * [pc1, pc2] * \{t1, \dots, t2\}$.

Value

An object of class "bgmyc_fit", a named list with:

par Matrix of retained MCMC samples (rows = samples, columns py, pc, t, loglik).
 assignments Data frame with columns labels (tip labels) and bgmyc (integer species index), giving the point-estimate partition at ppcutoff.
 probmat Numeric matrix ($n_tips * n_tips$) of posterior co-occurrence probabilities. Use with [bgmyc_tbl](#) to recompute the partition at any cutoff.
 accept_rates Named numeric vector of per-parameter acceptance rates (py, pc, t).
 mcmc, burnin, thinning Integer scalars.
 tree The input phylo object.

References

Reid N.M., Carstens B.C. 2012. Phylogenetic estimation error can decrease the accuracy of species delimitation: a Bayesian implementation of the general mixed Yule-coalescent model. *BMC Evolutionary Biology* 12, 196.

See Also

[bgmyc_tbl](#), [gmyc](#), [mptp_mcmc](#)

Examples

```
## Not run:
library(ape)
phy <- as.phylo(geophagus_beast)
result <- bgmyc(phy, mcmc = 11000, burnin = 1000, thinning = 100,
               start = c(1, 0.5, 30))
```

```
summary(result)
head(result$assignments)
bgmyc_tbl(result, ppcutoff = 0.95)

## End(Not run)
```

bgmyc_tbl

*Turns bGMYC Results Into a Tibble***Description**

bgmyc_tbl() processes output from `bgmyc()` (class "bgmyc_fit") or `bGMYC::bgmyc.singlephy()` (class "singlebgmyc") into an object of class `tbl_df`.

For "bgmyc_fit" objects the partition is derived from the stored posterior co-occurrence matrix (`$probmata`) by grouping tip pairs whose probability of co-occurrence exceeds `ppcutoff` (union-find). This lets you explore different thresholds without re-running the MCMC.

For legacy "singlebgmyc" objects the original **bGMYC** functions `spec.probmata()` and `bgmyc.point()` are reimplemented natively, so the **bGMYC** package is not required.

Usage

```
bgmyc_tbl(bgmyc_res, ppcutoff = 0.05, delimname = "bgmyc")
```

Arguments

bgmyc_res	Output from <code>bgmyc()</code> or <code>bGMYC::bgmyc.singlephy()</code> .
ppcutoff	Posterior co-occurrence probability threshold. Tip pairs with posterior probability of being conspecific $\geq$ <code>ppcutoff</code> are merged into the same species. Default 0.05.
delimname	Character. Column name for the species index in the returned tibble. Default "bgmyc".

Value

An object of class `tbl_df` with columns labels (tip labels, in tree-tip order) and the species index named by `delimname`.

Author(s)

Pedro S. Bittencourt

Source

Reid N.M., Carstens B.C. 2012. Phylogenetic estimation error can decrease the accuracy of species delimitation: a Bayesian implementation of the general mixed Yule-coalescent model. *BMC Evolutionary Biology* 12 (196).

See Also

`bgmyc()`, `gmyc_tbl()`

Examples

```
# bgmyc (no external package required)
result <- try( bgmyc(ape::as.phylo(geophagus_beast),
  mcmc = 11000, burnin = 1000, thinning = 100,
  start = c(1, 0.5, 30), quiet = TRUE
))
bgmyc_df <- try( bgmyc_tbl(result, ppcutoff = 0.05) )
try(bgmyc_df)

# legacy `singlebgmyc` object, as produced by the original bGMYC package
# (bGMYC itself is not required to build this example): `par` holds
# post-burnin (py, pc, t) draws, and `mrca[[t]]` lists the MRCA node
# offsets (APE node number - Ntip) merged into one species at threshold t.
tr <- ape::as.phylo(geophagus_beast)
mrca_node <- ape::getMRCA(tr, tr$tip.label[1:2]) - ape::Ntip(tr)
legacy_res <- structure(
  list(tree = tr, par = cbind(py = 1, pc = 1, t = 1L), mrca = list(mrca_node)),
  class = "singlebgmyc"
)
bgmyc_df2 <- try( bgmyc_tbl(legacy_res, ppcutoff = 0.5) )
try(bgmyc_df2)
```

check_delim

Checks If Two or More Species Delimitation Outputs are (Nearly) Equal

Description

`check_delim()` checks if two or more species delimitation outputs have differences in its dimensions, labels, and values.

Usage

```
check_delim(list)
```

Arguments

`list` a [list](#) containing two or more species delimitation outputs to check.

Details

check_delim() will check if two or more species delimitation outputs have different dimensions (rows, columns), if labels are the same or if there are any duplicated or absent labels, and if there are any NA values or if partitions were set using non numeric values. If TRUE for any of the cases listed above, check_delim() will return an error.

Value

A single logical value, TRUE or FALSE.

Author(s)

Pedro S. Bittencourt, Rupert A. Collins.

Examples

```
# create dummy delimitation outputs
delim_1 <- tibble::tibble(
  labels = paste0("seq", 1:10),
  method_A = c(rep(1, 5), rep(2, 5))
)

delim_2 <- tibble::tibble(
  labels = paste0("seq", 1:10),
  method_B = c(rep(1, 3), rep(2, 2), rep(3, 5))
)

delim_3 <- tibble::tibble(
  labels = paste0("seq", 1:10),
  method_C = c(rep(1, 3), rep(2, 2), rep(3, 3), rep(4, 2))
)

# check outputs
check_delim(list(delim_1, delim_2, delim_3))
```

check_identifiers	<i>Checks for Differences Between Identifiers in Metadata and DNA Sequence Files</i>
-------------------	--

Description

check_identifiers() checks for differences between identifiers in metadata and DNA sequence files.

Usage

```
check_identifiers(dna, identifier, data)
```

Arguments

dna a [DNABin](#) object.
 identifier column in data which contains sequence identifiers.
 data an object of class [tbl_df](#) containing sequence metadata.

Details

`check_identifiers()` is a helper function to check for inconsistencies between identifiers in meta-data and DNA sequence files. It performs three checks, in order:

1. every identifier in `dna` must be present in `data`; any missing identifiers are reported and abort the function.
2. `dna` must not contain duplicated identifiers; any duplicates are reported and abort the function.
3. `data` must not contain duplicated identifiers. Duplicated identifiers that also occur in `dna` are reported and abort the function, since they make matching sequences to metadata ambiguous. Duplicated identifiers that do not occur in `dna` are reported as a warning only, since they do not affect matching.

Value

Invisibly returns TRUE if all checks pass.

Author(s)

Pedro S. Bittencourt, Rupert A. Collins.

Examples

```
check_identifiers(geophagus, "gbAccession", geophagus_info)
```

clean_dna

Removes Gaps, Ambiguities and Missing Data from DNA Sequences

Description

`clean_dna()` removes all character not a valid ACTG base from a [DNABin](#) object.

Usage

```
clean_dna(dna, verbose = TRUE)
```

Arguments

dna an object of class [DNABin](#).
 verbose logical. Returns a warning if any sequence contains non ACTG bases.

Details

`clean_dna()` detects and removes any non ACTG bases from alignment. This includes: "N", "-", "?", "R", "Y", etc. If `verbose = TRUE`, returns a warning if these characters are inside the sequences, i.e, are not alignment padding chars at the ends.

Value

an object of class [DNABin](#).

Author(s)

Rupert A. Collins

Examples

```
geo_clean <- clean_dna(geophagus)
```

collapse_others	<i>Summarise Haplotype Metadata Down to One Row</i>
-----------------	---

Description

`collapse_others()` returns a [tbl_df](#) summarising all unique haplotype frequencies, duplicates and selected metadata into a single row.

Usage

```
collapse_others(data, hap_tbl, labels, cols)
```

Arguments

<code>data</code>	An object of class tbl_df containing sequence metadata.
<code>hap_tbl</code>	Output from haplotype_tbl .
<code>labels</code>	Column name which contains sequence names.
<code>cols</code>	A character vector of variables to collapse.

Details

`collapse_others()` is a helper function to summarise metadata along with [haplotype_tbl](#). For any given `cols`, `collapse_others()` flattens its content by unique haplotypes and its duplicates in `hap_tbl`.

Value

an object of class [tbl_df](#).

Author(s)

Pedro S. Bittencourt, Rupert A. Collins.

Examples

```
# summarise haplotypes
hap_tbl <- haplotype_tbl(geophagus)

# summarise country
others_df <- collapse_others(geophagus_info, hap_tbl, "gbAccession", "country")
```

confidence_intervals *Confidence Intervals for Species Delimitations Methods*

Description

These functions compute confidence intervals for various species delimitation methods, including GMYC, bGMYP, Local Minima, and mPTP.

Usage

```
gmyc_ci(tr, posterior, method = "single", interval = c(0, 5))

bgmyc_ci(
  tr,
  posterior,
  ppcutoff = 0.05,
  mcmc,
  burnin,
  thinning,
  py1 = 0,
  py2 = 2,
  pc1 = 0,
  pc2 = 2,
  t1 = 2,
  t2 = 51,
  scale = c(20, 10, 5),
  start = c(1, 0.5, 50)
)

locmin_ci(dna, block = 1, reps = 100, threshold = 0.01, haps = NULL, ...)

mptp_ci(
  infile,
  bootstraps,
  exe = NULL,
```

```

    outfolder = NULL,
    method = c("multi", "single"),
    minbren = 1e-04,
    webserver = NULL
)

```

Arguments

<code>tr</code>	A phylo object (ape); the maximum-likelihood (point-estimate) tree.
<code>posterior</code>	Trees from posterior. An object of class multiphylo .
<code>method</code>	Which algorithm for Maximum Likelihood point-estimate to be used. Available options are: • single Single-rate PTP model. It assumes that every species evolved with the same rate. • multi Multi-rate mPTP model. It assumes that all species have different evolutionary rates.
<code>interval</code>	Optimisation interval for the null model scaling exponent; see gmyc .
<code>ppcutoff</code>	Posterior probability threshold for clustering samples into species partitions. See bgmyc_tbl() for details. Default to 0.05.
<code>mcmc</code>	Integer. Total MCMC steps (default 11 000).
<code>burnin</code>	Integer. Steps discarded as burn-in (default 1 000).
<code>thinning</code>	Integer. Thinning interval; every thinning-th post-burnin step is retained (default 100).
<code>py1, py2</code>	Numeric. Prior bounds for the Yule rate-change exponent (default 0 and 2).
<code>pc1, pc2</code>	Numeric. Prior bounds for the coalescent rate-change exponent (default 0 and 2).
<code>t1, t2</code>	Integer. Prior bounds for the threshold parameter (number of species). Defaults: <code>t1 = 2</code> , <code>t2 = NULL</code> (auto: <code>Nnode - 1</code>).
<code>scale</code>	Numeric vector of length 3. Proposal scale parameters for <code>py</code> , <code>pc</code> , and <code>t</code> (default <code>c(20, 10, 5)</code>).
<code>start</code>	Numeric vector of length 3. Starting values for <code>py</code> , <code>pc</code> , and <code>t</code> (default <code>c(1, 0.5, 50)</code>).
<code>dna</code>	an object of class DNABin .
<code>block</code>	integer. Number of columns to be resampled together. Default to 1.
<code>reps</code>	Number of bootstrap replicates. Default to 100.
<code>threshold</code>	Distance cutoff for clustering. Default of 0.01. See localMinima for details.
<code>haps</code>	Optional. A vector of haplotypes to keep into the tbl_df .
<code>...</code>	Further arguments to be passed to dist.dna .
<code>infile</code>	Path to tree file in Newick format, or an object of class <code>"mptp_ml"</code> or <code>"mptp_mcmc"</code> returned by mptp or mptp_mcmc . When an <code>mptp_ml</code> object is supplied the remaining arguments are ignored and the assignments are returned directly.
<code>bootstraps</code>	Bootstrap trees. An object of class multiphylo .

exe	Path to an mPTP executable.
outfolder	Path to output folder. Default to NULL. If not specified, a temporary location is used.
minbrlen	Numeric. Branch lengths smaller or equal to the value provided are ignored from computations. Default to 0.0001. Use min_brlen for fine tuning.
webserver	A .txt file containing mPTP results obtained from a webserver. Default to NULL.

Details

Both `gmyc_ci` and `bgmyc_ci` can take a very long time to process, depending on how many posterior trees are provided. As an alternative, these analyses can be sped up significantly by running in parallel using [plan](#).

Value

A vector containing the number of species partitions in `tr`, `dna` or `infile` followed by the number of partitions in posterior, reps or bootstraps.

Author(s)

Pedro S. Bittencourt, Rupert A. Collins.

Examples

```
# gmyc confidence intervals

# compute values using multisession mode
{
  try( future::plan("multisession") )

  gmyc_res <- try( gmyc_ci(ape::as.phylo(geophagus_beast), geophagus_posterior) )

  # reset future parameters
  try( future::plan("sequential") )
}

# plot distribution
try(plot(density(gmyc_res)))

# tabulate
try( tibble::tibble(
  method = "gmyc",
  point_estimate = gmyc_res[1],
  CI_95 = as.integer(quantile(gmyc_res[-1], probs = c(0.025, 0.975))) |>
    stringr::str_flatten(collapse = "-"),
  CI_mean = as.integer(mean(gmyc_res[-1])),
  CI_median = as.integer(stats::median(gmyc_res[-1]))
)
)
```

delim_autoplot

*Plot Phylogenetic Trees With Species Delimitation Partitions***Description**

delim_autoplot() returns a phylogenetic tree plotted using ggtree alongside with a customized tile plot using [geom_tile](#) combined by [wrap_plots](#).

Usage

```
delim_autoplot(
  delim,
  tr,
  consensus = TRUE,
  n_match = NULL,
  delim_order = NULL,
  tbl_labs = NULL,
  col_vec = NULL,
  hexpand = 0.1,
  widths = c(0.5, 0.2)
)
```

Arguments

delim	Output from delim_join .
tr	A treedata object. Both phylogram and ultrametric trees are supported.
consensus	Logical. Should the majority-vote consensus to be estimated?
n_match	An Integer. If consensus = TRUE, threshold for majority-vote calculations. See delim_consensus for details.
delim_order	A character vector of species delimitation names ordered by user. Default to NULL.
tbl_labs	A tbl_df of customized labels for tree plotting. The first column must match tip labels of the tr object, while the second column should have customized labels.
col_vec	A color vector for species delimitation partitions. See delim_brewer for customized color palette options.
hexpand	Numeric. Expand xlim of tree by a ratio of x axis range. Useful if tiplabels become truncated when plotting. Default to 0.1.
widths	A numeric vector containing the relative widths of the tree and species delimitation bars. See wrap_plots for details. Defaults to c(0.5, 0.2).

Details

`delim_autoplot()` is a wrapper for tree plotting with associated data implemented using `ggtree`, `ggplot2`, and `patchwork`. If `consensus = TRUE`, a consensus bar will be plotted next to the species delimitation plot, summarizing partitions across samples. If no consensus is reached, an "X" will be plotted instead.

Value

A patchwork object.

Author(s)

Pedro S. Bittencourt, Rupert A. Collins.

Examples

```
# view partitions using an ultrametric tree
p <- delim_autoplot(geophagus_delims, geophagus_beast)
p

# view partitions using a phylogram
p1 <- delim_autoplot(geophagus_delims, geophagus_raxml)
```

`delim_autoplot2`*Plot Phylogenetic Trees With Species Delimitation Partitions*

Description

`delim_autoplot2()` returns a phylogenetic tree plotted using `ggtree` alongside with a customized tile plot using [geom_tile](#) combined by [wrap_plots](#).

Usage

```
delim_autoplot2(
  delim,
  tr,
  consensus = TRUE,
  n_match = NULL,
  delim_order = NULL,
  tbl_labs,
  species,
  hexpand = 0.1,
  widths = c(0.5, 0.2)
)
```

Arguments

delim	Output from delim_join .
tr	A treedata object. Both phylogram and ultrametric trees are supported.
consensus	Logical. Should the majority-vote consensus to be estimated?
n_match	An Integer. If consensus = TRUE, threshold for majority-vote calculations. See delim_consensus for details.
delim_order	A character vector of species delimitation names ordered by user. Default to NULL.
tbl_labs	A tbl_df of customized labels for tree plotting. The first column must match tip labels of the tr object, while the second column should have customized labels.
species	column name in tbl_labs which contains species names for each tip of the tree.
hexpand	Numeric. Expand xlim of tree by a ratio of x axis range. Useful if tiplabels become truncated when plotting. Default to 0.1.
widths	A numeric vector containing the relative widths of the tree and species delimitation bars. See wrap_plots for details. Defaults to c(0.5, 0.2).

Details

delim_autoplot2() is a wrapper for tree plotting with associated data implemented using ggtree, ggplot2, and patchwork. If consensus = TRUE, a consensus bar will be plotted next to the species delimitation plot, summarizing partitions across samples. If no consensus is reached, an "X" will be plotted instead. This function is a modified version of [delim_autoplot](#) which plots species partitions using a black and grey color scheme.

Value

A patchwork object.

Author(s)

Pedro S. Bittencourt, Rupert A. Collins.

Examples

```
# create labels
labs <- geophagus_info |> dplyr::select(gbAccession, scientificName)

# view partitions using an ultrametric tree
p <- delim_autoplot2(geophagus_delims,
  geophagus_beast,
  tbl_labs = labs,
  species = "scientificName"
)
p

# view partitions using a phylogram
p1 <- delim_autoplot2(geophagus_delims,
```

```

    geophagus_raxml,
    tbl_labs = labs,
    species = "scientificName"
  )

```

delim_brewer

*Customize Delimitation Colors***Description**

delim_brewer() returns a set of colors created by interpolating or using color palettes from [RColorBrewer](#), [viridisLite](#) or [randomcoloR](#).

Usage

```
delim_brewer(delim, package = NULL, palette = NULL, seed = NULL)
```

Arguments

delim	Output from delim_join .
package	Package which contains color palettes. Available options are "RColorBrewer", "viridisLite" or "randomcoloR".
palette	A palette name. brewer.pal for RColorBrewer or viridis for viridisLite options.
seed	Integer. Number to initialize random number generator.

Details

delim_brewer() interpolates over a color palette and returns a vector of random colors whose length is equal to the sum of unique species delimitation partitions in delim. For reproducibility, make sure to provide a seed. If not provided, [Sys.time](#) will be used as seed instead. One should also try different seeds to get best color combinations for plotting.

Value

A character vector of hexadecimal color codes.

Author(s)

Rupert A. Collins, Pedro S. Bittencourt

Examples

```

# create a vector of colors
cols <- delim_brewer(geophagus_delims, package = "randomcoloR")

```

delim_consensus	<i>Estimate a Majority-Vote Consensus</i>
-----------------	---

Description

`delim_consensus()` estimates a majority-vote consensus over the output of [delim_join](#) in a row-wise manner.

Usage

```
delim_consensus(delim, n_match = NULL)
```

Arguments

<code>delim</code>	Output from delim_join .
<code>n_match</code>	An integer. Threshold for Majority-Vote calculations. If not specified, returns a warning and the threshold will be defined as <code>ceiling(ncol(delim[, -1])/2)</code> .

Details

`delim_consensus()` iterates row-by-row, counting the number of matching species partition names across all species delimitations methods in [delim_join](#) output. If the sum of identical partition names is greater or equal `n_match`, the consensus column will be filled with its partition name. Otherwise, consensus column will be filled with [NA](#).

Value

an object of class [tbl_df](#).

Author(s)

Pedro S. Bittencourt

Examples

```
# estimate a majority vote consensus
delim_consensus <- delim_consensus(geophagus_delims, n_match= 5)

# check
delim_consensus
```

delim_join

*Join Multiple Species Delimitation Methods Outputs***Description**

delim_join() returns a [tbl_df](#) of species delimitation outputs whose partitions are consistent across different methods.

Usage

```
delim_join(delim, return = c("both", "df", "removed"))
```

Arguments

delim	A list or data.frame of multiple species delimitation methods outputs.
return	Which type of output to be returned if there are missing values within any of the species partitions used as input for analysis. Default to "df".

Details

delim_join() is a helper function to join multiple lists or columns of species delimitation outputs into a single [tbl_df](#) while keeping consistent identifications across multiple methods. Species delimitation outputs are in general a list or data frame of sample labels and its species partitions (Species 1, Species 2, etc.). These partition names may be or not the same across two or more methods. delim_join() standardizes partition names across two or more species delimitation outputs while keeping its underlying structure intact.

Value

an object of class [tbl_df](#).

Author(s)

Pedro S. Bittencourt, Rupert A. Collins.

Examples

```
## run GMYC
gmyc_res <- try( gmyc(ape::as.phylo(geophagus_beast), method = "single") )

# create a tibble
gmyc_df <- try( gmyc_tbl(gmyc_res) )

## run bGMYC
bgmyc_res <- try( bgmyc(ape::as.phylo(geophagus_beast),
  mcmc = 11000,
  burnin = 1000,
```

```

    thinning = 100,
    t1 = 2,
    t2 = ape::Ntip(ape::as.phylo(geophagus_beast)),
    start = c(1, 0.5, 50)
  )
)
# create a tibble
bgmyc_df <- try( bgmyc_tbl(bgmyc_res, ppcutoff = 0.05) )

## LocMin

# create a distance matrix
mat <- try( ape::dist.dna(geophagus, model = "raw", pairwise.deletion = TRUE) )

# estimate local minima from `mat`
locmin_res <- try( spider::localMinima(mat) )

# create a tibble
locmin_df <- try( locmin_tbl(mat,
  threshold = locmin_res$localMinima[1],
  haps = ape::as.phylo(geophagus_beast)$tip.label
)
)
# join delimitations
all_delims <- try( delim_join(list(gmyc_df, bgmyc_df, locmin_df)) )

# check
try(all_delims)

```

delim_tbl

*Generating a Delimitation Hypothesis Table***Description**

delim_tbl() returns species partition hypothesis estimated from a prior delimitation hypothesis supplied by the user.

Usage

```
delim_tbl(labels, sppVector, delimname = NULL)
```

Arguments

labels	Vector of unique sequence ID labels.
sppVector	Vector of corresponding morphological species delimitation groups.
delimname	Character. User needs to supply string to name the delimitation method in the table. Default to NA.

Details

`delim_tbl()` uses information in a species name vector to label each unique sample with a number corresponding to this name.

Value

an object of class `tbl_df`.

Author(s)

Rupert A. Collins

Examples

```
# create a tibble
delim_df <- delim_tbl(
  labels = geophagus_info$gbAccession,
  sppVector = geophagus_info$scientificName,
  delimname = "SNAPPER"
)

# check
delim_df
```

drop_sequences

Remove Sequences of a DNABin list object

Description

`drop_sequences()` removes sequences of a FASTA file by its names.

Usage

```
drop_sequences(dna, identifier, drop = TRUE)
```

Arguments

<code>dna</code>	a DNABin list object.
<code>identifier</code>	a character vector containing sequence names.
<code>drop</code>	Logical. If TRUE, sequence names in <code>identifier</code> will be dropped from <code>dna</code> . If FALSE, sequence names absent in <code>identifier</code> will be dropped instead.

Details

`drop_sequences()` relies on exact match between sequence names within a fasta file and `identifier` argument.

Value

an object of class [DNABin](#).

Author(s)

Pedro S. Bittencourt

Examples

```
# Create a vector of sequence names to drop or keep.
identifier <- names(geophagus)[1:3]

# Remove sequences listed in identifier
drop_sequences(geophagus, identifier, drop = TRUE)

# Remove sequences not listed in identifier
drop_sequences(geophagus, identifier, drop = FALSE)
```

dwc_terms

Print Darwin Core Terms, Definitions and Examples as Bullet Lists

Description

`dwc_terms()` checks a vector or list of terms and return definitions and examples for each one of them.

Usage

```
dwc_terms(dwc, terms)
```

Arguments

dwc	a list of standard terms and definitions created using get_dwc .
terms	a vector or list of terms to check.

Details

For each term in a vector or list, `dwc_terms` will return a bullet list containing the term, followed by its definition and examples.

Value

a bullet list.

Author(s)

Pedro S. Bittencourt, Rupert A. Collins.

Examples

```
dwc <- get_dwc(type= "simple")
dwc_terms(dwc, c("genus", "scientificName"))
```

geophagus

Cytochrome C Oxidase Sequences of Geophagus Eartheaters

Description

This is a set of 354 sequences of the mitochondrial gene cytochrome c oxidase subunit I (COI-5P) of the eartheaters of the *Geophagus sensu stricto* species group downloaded from GenBank. Most of these sequences are from the data analysed by Ximenes et al. (2021).

Usage

```
geophagus
```

Format

An object of class [DNABin](#)

Source

Ximenes AM, Bittencourt PS, Machado VN, Hrbek T, Farias IP. 2021. Mapping the hidden diversity of the *Geophagus sensu stricto* species group (Cichlidae: Geophagini) from the Amazon basin. PeerJ 9:e12443.

geophagus_beast

Geophagus Eartheaters Ultrametric Tree

Description

This is a Maximum Clade Credibility (MCC) tree containing unique haplotypes from [geophagus](#) estimated using BEAST2 v2.6.7. Unique haplotypes were select using [hap_collapse](#).

Usage

```
geophagus_beast
```

Format

An object of class [treedata](#).

geophagus_bootstraps *Geophagus Eartheaters Bootstrap Trees*

Description

This is a set of 100 Maximum Likelihood trees sampled from bootstrap trees used to estimate [geophagus_raxml](#) using RAxML-NG v. 1.1.0-master. Meant to be used for [confidence_intervals](#) estimation.

Usage

geophagus_bootstraps

Format

An object of class [multiphylo](#)

geophagus_delims *Geophagus Eartheaters Species Partitions*

Description

This is a data frame containing species delimitation partitions for all the 137 unique haplotypes of [geophagus](#) generated using functions contained in this package. Use [report_delim](#) to check number of lineages per method.

Usage

geophagus_delims

Format

A dataframe with 137 rows and 9 columns:

labels Unique haplotype labels
abgd species partitions for ABGD method
asap species partitions for ASAP method
bgmyc species partitions for bGMYC method
gmyc species partitions for GMYC method
locmin species partitions for locmin method
morph species partitions following NCBI taxonomy
mptp species partitions for mPTP method
ptp species partitions for PTP method

geophagus_info

Geophagus Earthearts Associated Metadata

Description

This is the associated metadata for the 354 sequences of the mitochondrial gene cytochrome c oxidase subunit I (COI-5P) of the *Geophagus sensu stricto* species group downloaded from GenBank and stored in [geophagus](#).

Usage

```
geophagus_info
```

Format

A data frame with 354 rows and 19 columns:

scientificName scientific name

scientificNameGenBank scientific name following NCBI taxonomy

class class

order order

family family

genus genus

dbid NCBI Nucleotide Database internal ID

gbAccession NCBI Nucleotide Database accession number

gene Gene acronym

length Sequence length in base pairs (bp)

organelle Organelle from which gene was sequenced

catalogNumber An identifier for the record within a data set or collection

country Name of the Country followed by sampling locality (when available)

publishedAs Title of the article which generated the sequences

publishedIn Journal which published the article

publishedBy A person, group, or organization responsible for depositing the sequence

date Date published

decimalLatitude Latitude in decimal degrees

decimalLongitude Longitude in decimal degrees

geophagus_posterior	<i>Geophagus Eartheaters Posterior Trees</i>
---------------------	--

Description

This is a set of 100 ultrametric trees sampled from the posterior trees used to estimate [geophagus_beast](#) using BEAST2 v2.6.7. Meant to be used for [confidence_intervals](#) estimation.

Usage

```
geophagus_posterior
```

Format

An object of class [multiphylo](#)

geophagus_raxml	<i>Geophagus Eartheaters Phylogram</i>
-----------------	--

Description

This is a Maximum Likelihood Estimation Tree containing unique haplotypes from [geophagus](#) estimated using RAxML-NG v. 1.1.0-master. Unique haplotypes were select using [hap_collapse](#).

Usage

```
geophagus_raxml
```

Format

An object of class [treedata](#).

get_delim_cols

*Extract Labels and Colors from Species Delimitation Partitions***Description**

get_delim_cols() returns a [tbl_df](#) format containing extracted and processed data from [delim_autoplot](#).

Usage

```
get_delim_cols(p, delimname = NULL, hap_tbl = NULL)
```

Arguments

p	Output from delim_autoplot .
delimname	A character vector of species delimitation names (optional). If provided, the function filters the data to only include rows matching such terms. Default to NULL.
hap_tbl	output from haplotype_tbl (optional). If provided, the function will annotate color and fill data for collapsed haplotypes. Default to NULL.

Details

get_delim_cols() is a convenience function to extract labels, species partitions, color and fill data from the output of [delim_autoplot](#) in a [tbl_df](#) format. It is best used when combined with haplotype information from [haplotype_tbl](#) or when combined with other metadata, such as GPS coordinates for map plotting.

Value

an object of class [tbl_df](#).

Author(s)

Pedro S. Bittencourt.

Examples

```
# plot using autoplot
p <- delim_autoplot(geophagus_delims, geophagus_beast)

# view
p

# get haplotypes
hap_tbl <- haplotype_tbl(geophagus)

# extract colors for consensus
get_delim_cols(p, delimname= "consensus", hap_tbl= hap_tbl)
```

`get_dwc`*Get Darwin Core Terms and Definitions*

Description

`get_dwc()` returns a list of standardized terms and definitions used by the Darwin Core Maintenance Interest Group <https://dwc.tdwg.org/>.

Usage

```
get_dwc(type)
```

Arguments

`type` Which type of distribution files to download. Available options are:

- `simple` Simple Darwin Core Terms.
- `all` All Darwin Core Terms.

Details

`get_dwc()` reads Darwin Core distribution documents and terms from Github repository <https://github.com/tdwg/dwc> directly into Environment. This function will return a list containing the most recent accepted terms as a vector and a `tbl_df` containing terms, definitions, examples and details about each one of them.

Value

a list.

Author(s)

Pedro S. Bittencourt, Rupert A. Collins

Examples

```
dwc <- get_dwc(type= "simple")
```

gmyc	<i>Run the GMYC Species Delimitation Analysis</i>
------	---

Description

Run the GMYC Species Delimitation Analysis

Usage

```
gmyc(tree, method = "single", interval = c(0, 5), quiet = TRUE, tol = 1e-06)
```

Arguments

tree	A "phylo" object (ape).
method	Analysis method: only "single" is implemented.
interval	Optimisation interval for the null model scaling exponent (used only for the pure-R fallback; C engine always searches [0, 5]).
quiet	Suppress per-threshold progress output (default TRUE).
tol	Ultrametricity tolerance (default 1e-6).

Value

A named list of class "gmyc", with the same structure as `splits::gmyc()`: `method`, `likelihood` (per-threshold vector), `parameters`, `entity`, `cluster`, `MRCA`, `threshold.time`, and `tree`.

References

Fujisawa & Barraclough (2013). *Syst. Biol.* 62(5), 707–724.

gmyc_tbl	<i>Turns GMYC Results Into a Tibble</i>
----------	---

Description

`gmyc_tbl()` processes output from `gmyc()` into an object of class `tbl_df`.

Usage

```
gmyc_tbl(gmyc_res, delimname = "gmyc")
```

Arguments

gmyc_res	Output from <code>gmyc()</code> .
delimname	Character. String to rename the delimitation method in the table. Default to 'gmyc'.

Details

[gmyc\(\)](#) optimizes genetic clusters, and [spec.list\(\)](#) clusters samples into species partitions from the result. [gmyc_tbl\(\)](#) turns these results into a tibble which matches the output from [bgmyc_tbl](#) and [locmin_tbl](#).

Value

An object of class [tbl_df](#).

Author(s)

Pedro S. Bittencourt, Tomas Hrbek

Source

Pons J., Barraclough T. G., Gomez-Zurita J., Cardoso A., Duran D. P., Hazell S., Kamoun S., Sumlin W. D., Vogler A. P. 2006. Sequence-based species delimitation for the DNA taxonomy of undescribed insects. *Systematic Biology*. 55:595-609.

Monaghan M. T., Wild R., Elliot M., Fujisawa T., Balke M., Inward D. J. G., Lees D. C., Ranaivosolo R., Eggleton P., Barraclough T. G., Vogler A. P. 2009. Accelerated species inventory on Madagascar using coalescent-based models of species delineation. *Systematic Biology*. 58:298-311.

Fujisawa T., Barraclough T. G. 2013. Delimiting Species Using Single-Locus Data and the Generalized Mixed Yule Coalescent Approach: A Revised Method and Evaluation on Simulated Data Sets. *Systematic Biology*. 62(5):707–724.

Examples

```
# run GMYC
gmyc_res <- try( gmyc(ape::as.phylo(geophagus_beast)) )

# create a tibble
gmyc_df <- try( gmyc_tbl(gmyc_res) )

# check
try(gmyc_df)
```

hap_collapse

Removes Duplicated Sequences from Alignment

Description

`hap_collapse()` collapses haplotypes from a [DNABin](#) object, keeping unique haplotypes only.

Usage

```
hap_collapse(dna, clean = TRUE, collapseSubstrings = TRUE, verbose = TRUE)
```

Arguments

dna A [DNABin](#) object.

clean logical. Whether to remove or not remove non ACTG bases from alignment.

collapseSubstrings logical. Whether to collapse or not collapse shorter but identical sequences.

verbose logical. Returns a warning if any sequence contains non ACTG bases. See [clean_dna](#) for details.

Details

hap_collapse() collapses a [DNABin](#) object, keeping unique haplotypes only. If clean = TRUE, the function will call [clean_dna](#) to remove any non ACTG bases from alignment prior to collapsing haplotypes. If clean = FALSE, the function will treat data as it is, and will not remove any bases. If collapseSubstrings = TRUE, the function will consider shorter but identical sequences as the same haplotype and collapse them, returning the longest sequence. If collapseSubstrings = FALSE, the function will consider shorter but identical sequences as different haplotypes and will keep them.

Value

A [DNABin](#) object.

Author(s)

Rupert A. Collins

Examples

```
# collapse into unique haplotypes, including shorter sequences
hap_collapse(geophagus, clean = TRUE, collapseSubstrings = TRUE)

# collapse into unique haplotypes keeping shorter sequences
hap_collapse(geophagus, clean = TRUE, collapseSubstrings = FALSE)
```

hap_unite

Unite Haplotype Summaries with Species Delimitation Outputs

Description

hap_unite() returns a single [tbl_df](#) combining all results from [haplotype_tbl](#) or [collapse_others](#) with results from [delim_join](#) or [delim_consensus](#).

Usage

```
hap_unite(hap_tbl, delim)
```

Arguments

hap_tbl output from [haplotype_tbl](#) or [collapse_others](#).
delim output from [delim_join](#) or [delim_consensus](#).

Details

Many functions in this package relies on the usage of unique haplotypes due to known issues when using identical or duplicated sequences for species delimitation analysis. Thus, these outputs will very often refer only to unique haplotypes within a given dataset, which can be determined by using functions like [hap_collapse](#). Assuming that a duplicated or identical sequence should share the same properties as the first sequence of the group has, `hap_unite()` combines the output of [haplotype_tbl](#) with the output of [delim_join](#). Alternatively, one may use [collapse_others](#) and [delim_consensus](#) as well. This output may be used for downstream analysis or to determine in which cluster a given sequence belongs.

Value

an object of class [tbl_df](#).

Author(s)

Pedro S. Bittencourt

Examples

```
# get haplotype table
hap_tbl <- haplotype_tbl(geophagus)

# unite
hap_unite(hap_tbl, geophagus_delims)
```

haplotype_tbl	<i>Summarise Haplotypes Down to One Row</i>
---------------	---

Description

`haplotype_tbl()` returns a [tbl_df](#) summarising all unique haplotype frequencies and duplicates into a single row.

Usage

```
haplotype_tbl(dna, clean = TRUE, collapseSubstrings = TRUE, verbose = TRUE)
```

Arguments

dna an object of class [DNABin](#).
clean logical. Whether to remove or not remove non ACTG bases from alignment.
collapseSubstrings logical. Whether to collapse or not collapse shorter but identical sequences.
verbose logical. Returns a warning if any sequence contains non ACTG bases. See [clean_dna](#) for details.

Details

haplotype_tbl() uses a combination of [clean_dna](#) and [hap_collapse](#) to summarise haplotypes into a tibble. Each row of the tibble has an unique haplotype, its frequency and all its collapsed duplicates in a flattened string.

Value

an object of class [tbl_df](#).

Author(s)

Rupert A. Collins, Pedro S. Bittencourt.

Examples

```
# get haplotype table
haplotype_tbl(geophagus)
```

locmin_tbl

Turns Local Minima Results into a Tibble

Description

locmin_tbl() processes output from [tclust](#) into an object of class [tbl_df](#).

Usage

```
locmin_tbl(distobj, threshold = 0.01, haps = NULL, delimname = "locmin")
```

Arguments

distobj A distance object (usually from [dist.dna](#)).
threshold Distance cutoff for clustering. Default of 0.01. See [localMinima](#) for details.
haps Optional. A vector of haplotypes to keep into the [tbl_df](#).
delimname Character. String to rename the delimitation method in the table. Default to 'locmin'.

Details

spider package uses [localMinima](#) to determine possible thresholds for any distance matrix and [tclust](#) to cluster samples within a given threshold into species partitions. `locmin_tbl()` turns these inputs into a tibble which matches the output from [gmyc_tbl](#) and [bgmyc_tbl](#).

Value

An object of class `tbl_df`.

Author(s)

Samuel Brown.

Source

Brown S.D.J., Collins R.A., Boyer S., Lefort M.-C., Malumbres-Olarte J., Vink C.J., Cruickshank, R.H. 2012. Spider: An R package for the analysis of species identity and evolution, with particular reference to DNA barcoding. *Molecular Ecology Resources*, 12: 562-565.

Examples

```
# create a distance matrix
mat <- ape::dist.dna(geophagus, model = "raw", pairwise.deletion = TRUE)

# run Local Minima
locmin_res <- spider::localMinima(mat)

# create a tibble
locmin_df <- locmin_tbl(mat,
                        threshold = locmin_res$localMinima[1],
                        haps = ape::as.phylo(geophagus_beast)$tip.label)

# check
locmin_df
```

match_ratio	<i>Compute Agreement Between Alternative Species Delimitation Partitions</i>
-------------	--

Description

`match_ratio()` uses the Match Ratio statistic of Ahrens et al. (2014) to compute agreement between all pairs of species delimitation partitions in [delim_join](#) output.

Usage

```
match_ratio(delim, sorted = TRUE)
```

Arguments

`delim` Output from [delim_join](#).
`sorted` If output should be sorted by match ratio values. Default to TRUE.

Details

`match_ratio()` iterates between all species delimitation partitions in [delim_join](#) output and returns a [tbl_df](#) containing the following columns:

- `pairs` pairs of species delimitation methods analyzed.
- `delim_1` number of species partitions in method 1.
- `delim_2` number of species partitions in method 2.
- `n_match` number of identical species partitions in methods 1 and 2.
- `match_ratio` match ratio statistic, where 0 indicates no agreement between pairs of species delimitation partitions and 1 indicates complete agreement between them.

Value

an object of class [tbl_df](#).

Author(s)

Pedro S. Bittencourt

Source

Ahrens D., Fujisawa T., Krammer H. J., Eberle J., Fabrizi S., Vogler A. P. 2016. Rarity and Incomplete Sampling in DNA-Based Species Delimitation. *Systematic Biology* 65 (3): 478-494. [doi:10.1093/sysbio/syw002](https://doi.org/10.1093/sysbio/syw002)

Examples

```
# estimate match ratio statistics
match_ratio(geophagus_delims)
```

<code>min_brlen</code>	<i>A function to report the smallest tip-to-tip distances in a phylogenetic tree</i>
------------------------	--

Description

`min_brlen()` returns a table of smallest tip-to-tip distances in a phylogenetic tree.

Usage

```
min_brlen(tree, n = 5, verbose = TRUE)
```

Arguments

tree	A path to tree file in Newick format, or a phylogenetic tree object of class phylo .
n	Number of distances to report (default = 5).
verbose	Logical of whether to print the result to screen (default = TRUE).

Details

`min_brlen()` tabulates the smallest tip-to-tip distances in a phylogenetic tree using [cophenetic.phylo](#) and prints a table to screen. This is useful when excluding identical or near-identical haplotypes using the `'-minbr'` parameter in `mPTP`.

Value

an object of class [tbl_df](#)

Author(s)

Rupert A. Collins

Examples

```
# estimate minimum branch length from raxml tree
min_brlen(ape::as.phylo(geophagus_raxml), n = 5)
```

morph_tbl

Generating a Morphological Delimitation Table

Description

`morph_tbl()` returns species partition hypothesis estimated from a prior taxonomic identifications supplied by the user.

Usage

```
morph_tbl(labels, sppVector, delimname = "morph")
```

Arguments

labels	Vector of unique sequence ID labels.
sppVector	Vector of corresponding morphological species delimitation groups.
delimname	Character. String to rename the delimitation method in the table. Default to <code>'morph'</code> .

Details

`morph_tbl()` uses information in a species name vector to label each unique sample with a number corresponding to this name.

Value

an object of class `tbl_df`.

Author(s)

Rupert A. Collins

Examples

```
# create a tibble
morph_df <- morph_tbl(
  labels = geophagus_info$gbAccession,
  sppVector = geophagus_info$scientificName
)

# check
morph_df
```

mptp

Run mPTP species delimitation (Maximum Likelihood)

Description

Calls the mPTP C engine directly via `.Call()` — no external process, no file parsing. Results are returned as a structured R object.

Usage

```
mptp(
  tree,
  method = c("multi", "single"),
  outgroup = NULL,
  outgroup_crop = FALSE,
  pvalue = 0.001,
  minbr = 1e-04,
  seed = as.integer(Sys.time()),
  quiet = FALSE
)
```

Arguments

tree	A path to a Newick tree file (character) or an <code>read.tree</code> phylo object. Rooted and unrooted trees are both accepted.
method	Coalescent rate model: "multi" (default, recommended) or "single".
outgroup	Character. Name of the outgroup taxon used for rooting an unrooted tree (optional; if omitted, the longest tip branch is used).

outgroup_crop	Logical. If TRUE, removes the outgroup after rooting (default FALSE).
pvalue	Numeric. Significance threshold for the likelihood ratio test (default 0.001).
minbr	Numeric. Minimum branch length; edges shorter than this are excluded from the analysis (default 0.0001).
seed	Integer. Random seed (default: current time).
quiet	Logical. Suppress C-level progress messages (default FALSE).

Value

An object of class "mptp_ml", a list with:

n_species Integer. Number of delimited species.
method Character. "multi" or "single".
null_logl Numeric. Null-model log-likelihood.
best_logl Numeric. Best log-likelihood under mPTP.
pvalue Numeric. LRT p-value.
lrt_passed Logical. Whether the LRT rejected the null.
edge_count Integer. Edges longer than minbr.
total_edges Integer. Total edges in the tree.
assignments Data frame with columns labels and mptp (integer index; ptp when method = "single").

See Also

[mptp_mcmc](#), [mptp_tbl](#)

Examples

```
## Not run:
result <- mptp("tree.nwk", outgroup = "Outgroup1",
               outgroup_crop = TRUE)

print(result)
head(result$assignments)

library(ape)
phy <- read.tree("tree.nwk")
result <- mptp(phy)

## End(Not run)
```

mptp_mcmc

*Run mPTP species delimitation (MCMC)***Description**

Calls the mPTP MCMC engine directly via `.Call()`. Returns ML species assignments plus per-node posterior support values.

Usage

```
mptp_mcmc(
  tree,
  method = c("multi", "single"),
  mcmc_steps = 1000000L,
  mcmc_sample = 1000L,
  mcmc_burnin = 100000L,
  mcmc_runs = 1L,
  mcmc_credible = 0.95,
  mcmc_startnull = FALSE,
  mcmc_startrandom = FALSE,
  mcmc_startml = FALSE,
  outgroup = NULL,
  outgroup_crop = FALSE,
  pvalue = 0.001,
  minbr = 1e-04,
  seed = as.integer(Sys.time()),
  quiet = FALSE
)
```

Arguments

<code>tree</code>	A path to a Newick tree file (character) or a phylo object from ape . See mptp for details.
<code>method</code>	Coalescent rate model: "multi" (default) or "single".
<code>mcmc_steps</code>	Integer. Total MCMC steps per run (default 1,000,000).
<code>mcmc_sample</code>	Integer. Sampling frequency (default 1,000).
<code>mcmc_burnin</code>	Integer. Steps discarded as burn-in (default 100,000).
<code>mcmc_runs</code>	Integer. Number of independent runs (default 1).
<code>mcmc_credible</code>	Numeric. Credible interval (default 0.95).
<code>mcmc_startnull</code>	Logical. Start from the null (one-species) model.
<code>mcmc_startrandom</code>	Logical. Start from a random delimitation.
<code>mcmc_startml</code>	Logical. Start from the ML delimitation.
<code>outgroup</code>	Character. Outgroup taxon name for rooting (optional).

outgroup_crop	Logical. Remove outgroup after rooting (default FALSE).
pvalue	Numeric. LRT significance threshold (default 0.001).
minbr	Numeric. Minimum branch length threshold (default 0.0001).
seed	Integer. Random seed (default: current time).
quiet	Logical. Suppress C-level progress messages (default FALSE).

Value

An object of classes `c("mptp_mcmc", "mptp_ml")`, a list with all fields of "mptp_ml" plus:

`support` Data frame with columns `node_label` and `support` (0–1); posterior probability of each inner node being a coalescent root.

`mcmc_steps` Integer.

`mcmc_sample` Integer.

`mcmc_burnin` Integer.

`mcmc_runs` Integer.

`seed` Integer. Actual seed used.

See Also

[mptp](#), [mptp_tbl](#)

Examples

```
## Not run:
result <- mptp_mcmc(
  "tree.nwk",
  mcmc_steps   = 5000000L,
  mcmc_burnin  = 500000L,
  mcmc_runs    = 3L,
  seed         = 42L,
  outgroup     = "Outgroup1",
  outgroup_crop = TRUE
)
print(result)
head(result$support)

## End(Not run)
```

mptp_support	<i>Extract MCMC support values</i>
--------------	------------------------------------

Description

Extract MCMC support values

Usage

```
mptp_support(x)
```

Arguments

x An object of class "mptp_mcmc".

Value

A data frame with columns node_label and support.

mptp_tbl	<i>A Command-Line Interface for mPTP - multi-rate Poisson Tree Processes</i>
----------	--

Description

mptp_tbl() returns species partition hypothesis estimated by mPTP software <https://github.com/Pas-Kapli/mptp>.

Usage

```
mptp_tbl(
  infile,
  exe = NULL,
  outfolder = NULL,
  method = c("multi", "single"),
  minbrlen = 1e-04,
  webserver = NULL,
  delimname = "mptp"
)
```

Arguments

infile	Path to tree file in Newick format, or an object of class "mptp_ml" or "mptp_mcmc" returned by mptp or mptp_mcmc . When an mptp_ml object is supplied the remaining arguments are ignored and the assignments are returned directly.
exe	Path to an mPTP executable.
outfolder	Path to output folder. Default to NULL. If not specified, a temporary location is used.
method	Which algorithm for Maximum Likelihood point-estimate to be used. Available options are: • single Single-rate PTP model. It assumes that every species evolved with the same rate. • multi Multi-rate mPTP model. It assumes that all species have different evolutionary rates.
minbrlen	Numeric. Branch lengths smaller or equal to the value provided are ignored from computations. Default to 0.0001. Use min_brlen for fine tuning.
webserver	A .txt file containing mPTP results obtained from a webserver. Default to NULL.
delimname	Character. String to rename the delimitation method in the table. Default to 'mptp'.

Details

mptp_tbl() relies on [system](#) to invoke mPTP software through a command-line interface. Hence, you must have the software available as an executable file on your system in order to use this function properly. mptp_tbl() saves all output files in outfolder and imports the results generated to Environment. If an outfolder is not provided by the user, then a temporary location is used. Alternatively, mptp_tbl() can parse a file obtained from webserver such as <https://mptp.h-its.org/>.

Value

an object of class [tbl_df](#)

Author(s)

Paschalia Kapli, Sarah Lutteropp, Jiajie Zhang, Kassian Kobert, Pavlos Pavlides, Alexandros Stamatakis, Tomáš Flouri.

Source

Kapli T., Lutteropp S., Zhang J., Kobert K., Pavlidis P., Stamatakis A., Flouri T. 2016. Multi-rate Poisson tree processes for single-locus species delimitation under maximum likelihood and Markov chain Monte Carlo. *Bioinformatics* 33(11):1630-1638.

Examples

```
# get path to phylogram
path_to_file <- system.file("extdata/geophagus_raxml.nwk", package = "delimtools")

# run mPTP in single threshold mode (PTP)
ptp_df <- try( mptp_tbl(
  infile = path_to_file,
  exe = "/usr/local/bin/mptp",
  method = "single",
  minbrlen = 0.0001,
  delimname = "ptp",
  outfolder = NULL
)
)
# check
ptp_df

# run mPTP in multi threshold mode (mPTP)
mptp_df <- try( mptp_tbl(
  infile = path_to_file,
  exe = "/usr/local/bin/mptp",
  method = "single",
  minbrlen = 0.0001,
  delimname = "mptp",
  outfolder = NULL
)
)
# check
try(mptp_df)
```

report_delim	<i>Report Unique Species Partitions</i>
--------------	---

Description

report_delim() reports the number of unique species partitions in delim.

Usage

```
report_delim(delim, verbose = TRUE)
```

Arguments

delim	Output from any *_tbl() (e.g. gmyc_tbl), delim_join or delim_consensus .
verbose	Logical. If TRUE, returns a message and a tabulated summary of delim.

Details

For each column in `delim`, `report_delim()` will calculate the number of unique partitions and print them to Console. If `delim` is an output from `*_tbl()`, `report_delim()` will get unique species partitions using `vec_unique_count`. If `delim` is an output from `delim_join` or `delim_consensus`, values are summarized by using `n_distinct` with `na.rm = TRUE`. This is to prevent any columns with NA values to be interpreted as species partitions.

Value

an object of class `tbl_df`.

Author(s)

Rupert A. Collins, Pedro S. Bittencourt

Examples

```
# report geophagus delimitations
report_delim(geophagus_delims)
```

<code>spec.list</code>	<i>Get species list from a delimitation result</i>
------------------------	--

Description

Returns a character vector of tip labels, grouped by inferred species.

Usage

```
spec.list(x, ...)

## S3 method for class 'gmyc'
spec.list(x, second.peak = FALSE, ...)
```

Arguments

<code>x</code>	A fitted delimitation object (e.g. <code>gmyc</code>).
<code>...</code>	Additional arguments passed to methods.
<code>second.peak</code>	Logical. If TRUE, use the second likelihood peak instead of the global maximum. Default: FALSE.

Value

A named character vector mapping tips to species identifiers.

summary.bgmyc_fit	<i>Summary of a Bayesian GMYC (bGMYC) delimitation</i>
-------------------	--

Description

Prints a summary of the MCMC run and species delimitation results from a bgmyc_fit object.

Usage

```
## S3 method for class 'bgmyc_fit'  
summary(object, ...)
```

Arguments

object	An object of class bgmyc_fit, as returned by bgmyc() (or equivalent fitting function).
...	Additional arguments, currently ignored (included for S3 method consistency).

Value

Invisibly returns object.

Index

* datasets

- geophagus, [32](#)
- geophagus_beast, [32](#)
- geophagus_bootstraps, [33](#)
- geophagus_delims, [33](#)
- geophagus_info, [34](#)
- geophagus_posterior, [35](#)
- geophagus_raxml, [35](#)

abgd, [4](#)
abgd(), [6](#)
abgd_tbl, [6](#)
as_dwc, [7](#)
asap, [9](#)
asap(), [11](#)
asap_tbl, [10](#)

best_partition, [12](#)
bgmyc, [13](#)
bgmyc(), [15](#), [16](#)
bgmyc_ci (confidence_intervals), [20](#)
bgmyc_tbl, [14](#), [15](#), [39](#), [43](#)
bgmyc_tbl(), [21](#)
brewer.pal, [26](#)

check_delim, [16](#)
check_identifiers, [17](#)
clean_dna, [18](#), [40](#), [42](#)
collapse_others, [19](#), [40](#), [41](#)
confidence_intervals, [20](#), [33](#), [35](#)
cophenetic.phylo, [45](#)

data.frame, [28](#)
delim_autoplot, [23](#), [25](#), [36](#)
delim_autoplot2, [24](#)
delim_brewer, [23](#), [26](#)
delim_consensus, [23](#), [25](#), [27](#), [40](#), [41](#), [52](#), [53](#)
delim_join, [23](#), [25–27](#), [28](#), [40](#), [41](#), [43](#), [44](#), [52](#), [53](#)
delim_tbl, [29](#)

delimtools (delimtools-package), [3](#)
delimtools-package, [3](#)
dist.dna, [21](#), [42](#)
DNABin, [18](#), [19](#), [21](#), [30–32](#), [39](#), [40](#), [42](#)
drop_sequences, [30](#)
dwc_terms, [31](#)

geom_tile, [23](#), [24](#)
geophagus, [32](#), [32–35](#)
geophagus_beast, [32](#), [35](#)
geophagus_bootstraps, [33](#)
geophagus_delims, [33](#)
geophagus_info, [34](#)
geophagus_posterior, [35](#)
geophagus_raxml, [33](#), [35](#)
get_delim_cols, [36](#)
get_dwc, [31](#), [37](#)
get_dwc(), [8](#)
gmyc, [14](#), [21](#), [38](#)
gmyc(), [38](#), [39](#)
gmyc_ci (confidence_intervals), [20](#)
gmyc_tbl, [38](#), [43](#), [52](#)
gmyc_tbl(), [16](#)

hap_collapse, [32](#), [35](#), [39](#), [41](#), [42](#)
hap_unite, [40](#)
haplotype_tbl, [19](#), [36](#), [40](#), [41](#), [41](#)

list, [16](#), [28](#)
localMinima, [21](#), [42](#), [43](#)
locmin_ci (confidence_intervals), [20](#)
locmin_tbl, [39](#), [42](#)

match_ratio, [43](#)
min_brlen, [22](#), [44](#), [51](#)
morph_tbl, [45](#)
mptp, [21](#), [46](#), [48](#), [49](#), [51](#)
mptp_ci (confidence_intervals), [20](#)
mptp_mcmc, [14](#), [21](#), [47](#), [48](#), [51](#)
mptp_support, [50](#)

mptp_tbl, [47](#), [49](#), [50](#)
multiphylo, [21](#), [33](#), [35](#)

n_distinct, [53](#)
NA, [27](#)

phylo, [13](#), [45](#)
plan, [22](#)

randomcoloR, [26](#)
RColorBrewer, [26](#)
read.tree, [46](#)
report_delim, [33](#), [52](#)

spec.list, [53](#)
spec.list(), [39](#)
summary.bgmyc_fit, [54](#)
Sys.time, [26](#)
system, [7](#), [11](#), [51](#)

tbl_df, [6–8](#), [11](#), [15](#), [18](#), [19](#), [21](#), [23](#), [25](#), [27](#), [28](#),
[30](#), [36–46](#), [51](#), [53](#)
tclust, [42](#), [43](#)
treedata, [23](#), [25](#), [32](#), [35](#)

vec_unique_count, [53](#)
viridis, [26](#)
viridisLite, [26](#)

wrap_plots, [23–25](#)