Package ‘taxodist’

May 6, 2026

Title Taxonomic Distance and Phylogenetic Lineage Computation

Version 0.3.0

Maintainer Rodrigo Fonseca Villa <`rodrigo03.villa@gmail.com`>

Description Computes phylogenetic distances between any two taxa using hierarchical lineage data retrieved from The Taxonomicon <`http://taxonomicon.taxonomy.nl`>, a comprehensive curated classification of all life based on Systema Naturae 2000 (Brands, 1989 <`http://taxonomicon.taxonomy.nl`>). Given any two taxon names, retrieves their full lineages, identifies the most recent common ancestor (MRCA), and computes a dissimilarity index based on lineage depth. Outputs native dist objects, enabling direct integration with the R statistical ecosystem for hierarchical clustering, principal coordinate analysis (PCoA), and multivariate ecological analyses. Supports individual distance queries, pairwise distance matrices, clade filtering, and lineage utilities.

Language en-US

URL `https://github.com/rodrigosqrt3/taxodist`

BugReports `https://github.com/rodrigosqrt3/taxodist/issues`

Encoding UTF-8

RoxygenNote 7.3.3

SystemRequirements internet access

Depends R (>= 4.1.0)

Imports httr (>= 1.4.0), rvest (>= 1.0.0), stringr (>= 1.4.0), purrr (>= 0.3.0), cli (>= 3.0.0), utils, stats, graphics

Suggests testthat (>= 3.0.0), mockery, knitr, rmarkdown, xml2, ape, vegan

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation no

Author Rodrigo Fonseca Villa [aut, cre] (ORCID: <`https://orcid.org/0009-0005-2938-2270`>)

Repository CRAN

Date/Publication 2026-05-05 22:30:02 UTC

Contents

`taxodist-package` *taxodist: Taxonomic Distance and Phylogenetic Lineage Computation*

Description

taxodist computes phylogenetic distances between any two taxa using hierarchical lineage data retrieved from The Taxonomicon (taxonomy.nl), a comprehensive curated classification of all life based on Systema Naturae 2000.

Core functions:

- `get_lineage()` — retrieve the full lineage of any taxon

- `taxo_distance()` — compute the tree metric distance between two taxa
- `mrca()` — find the most recent common ancestor
- `distance_matrix()` — compute all pairwise distances for a set of taxa
- `closest_relative()` — find the closest relative among candidates
- `compare_lineages()` — print a side-by-side lineage comparison
- `shared_clades()` — list clades shared between two taxa
- `is_member()` — test clade membership
- `filter_clade()` — filter taxa by clade membership
- `check_coverage()` — check Taxonomicon coverage for a list of taxa
- `lineage_depth()` — get the lineage depth of a taxon
- `clear_cache()` — clear the session lineage cache

Mathematical background:

The distance metric is based on the depth of the most recent common ancestor (MRCA):

$$d(A, B) = \frac{1}{\text{depth}(\text{MRCA}(A, B))}$$

The deeper the shared ancestor, the smaller the distance. This metric ensures that taxa sharing the same MRCA are always equidistant from any third taxon, regardless of lineage depth below the split — a key biological correctness property absent from Jaccard-based approaches.

Data source:

All lineage data is sourced from The Taxonomicon (taxonomy.nl), based on Systema Naturae 2000 by S.J. Brands (1989 onwards). Please cite this resource when using taxodist in published work.

Author(s)

Maintainer: Rodrigo Fonseca Villa <`rodrigo03.villa@gmail.com`> (ORCID)

References

Brands, S.J. (1989 onwards). Systema Naturae 2000. Amsterdam, The Netherlands. Retrieved from The Taxonomicon, `http://taxonomicon.taxonomy.nl`.

See Also

Useful links:

- `https://github.com/rodrigosqrt3/taxodist`
- Report bugs at `https://github.com/rodrigosqrt3/taxodist/issues`

`check_coverage` *Check whether a taxon is covered by The Taxonomicon*

Description

Queries The Taxonomicon for a taxon name and returns a logical indicating whether the taxon was found. Useful for pre-screening a list of names before running distance computations.

Usage

```
check_coverage(taxa, verbose = FALSE)
```

Arguments

`taxa`	A character vector of one or more taxon names.
`verbose`	Logical. If `TRUE`, prints progress messages. Default `FALSE`.

Value

A named logical vector. `TRUE` indicates the taxon was found, `FALSE` indicates it was not.

Examples

```
check_coverage(c("Tyrannosaurus", "Velociraptor", "Fakeosaurus"))
```

`clear_cache` *Clear the taxodist lineage cache*

Description

Clears all cached lineages stored in the current R session. Useful when you suspect cached data is stale or want to force fresh retrieval.

Usage

```
clear_cache()
```

Value

Invisibly returns `NULL`.

Examples

```
clear_cache()
```

`closest_relative` *Find the closest relative of a taxon among a set of candidates*

Description

Given a query taxon and a vector of candidate taxa, returns the candidate with the smallest phylogenetic distance to the query.

Usage

```
closest_relative(taxon, candidates, verbose = FALSE)
```

Arguments

`taxon`	A character string giving the query taxon name.
`candidates`	A character vector of candidate taxon names to compare against.
`verbose`	Logical. If `TRUE`, prints progress messages. Default `FALSE`.

Value

A data frame with columns `taxon` (candidate name) and `distance` (tree metric distance), sorted by distance ascending. Returns `NULL` if the query taxon cannot be found.

Examples

```
closest_relative("Tyrannosaurus",
  c("Velociraptor", "Triceratops", "Brachiosaurus", "Allosaurus"))
```

`compare_lineages` *Compare lineages of two taxa side by side*

Description

Prints the lineages of two taxa aligned at their most recent common ancestor, making the point of divergence easy to identify.

Usage

```
compare_lineages(taxon_a, taxon_b, verbose = FALSE)
```

Arguments

`taxon_a`	A character string giving the first taxon name.
`taxon_b`	A character string giving the second taxon name.
`verbose`	Logical. If `TRUE`, prints progress messages. Default `FALSE`.

Value

Invisibly returns a list with elements `lineage_a`, `lineage_b`, and `mrca_depth`.

Examples

```
compare_lineages("Tyrannosaurus", "Velociraptor")
compare_lineages("Tyrannosaurus", "Triceratops")
```

`distance_matrix` *Compute pairwise taxonomic distances for a set of taxa*

Description

Given a vector of taxon names, computes all pairwise phylogenetic distances and returns a symmetric distance matrix. Lineages are cached after first retrieval to minimise redundant network requests.

Usage

```
distance_matrix(taxa, verbose = FALSE, progress = TRUE)
```

Arguments

`taxa`	A character vector of taxon names.
`verbose`	Logical. If `TRUE`, prints progress for each pair. Default `FALSE`.
`progress`	Logical. If `TRUE`, shows a progress bar. Default `TRUE`.

Value

A symmetric numeric matrix of class `"dist"` containing pairwise distances. Row and column names are set to the input taxon names. Taxa that could not be found are included with `NA` distances.

See Also

`taxo_distance()`, `closest_relative()`

Examples

```
theropods <- c("Tyrannosaurus", "Velociraptor", "Spinosaurus",
               "Allosaurus", "Carnotaurus")
mat <- distance_matrix(theropods)
print(mat)
```

`filter_clade` *Filter a vector of taxa to those belonging to a given clade*

Description

Given a vector of taxon names and a clade name, returns only those taxa whose lineage includes the specified clade.

Usage

```
filter_clade(taxa, clade, verbose = FALSE)
```

Arguments

`taxa`	A character vector of taxon names.
`clade`	A character string giving the clade to filter by.
`verbose`	Logical. If `TRUE`, prints progress messages. Default `FALSE`.

Value

A character vector of taxa that are members of the specified clade.

Examples

```
taxa <- c("Tyrannosaurus", "Triceratops", "Velociraptor",
          "Brachiosaurus", "Homo")
filter_clade(taxa, "Theropoda")
filter_clade(taxa, "Dinosauria")
```

`get_lineage` *Retrieve the full taxonomic lineage of a taxon by name*

Description

A convenience wrapper that combines `get_taxonomicon_id()` and `get_lineage_by_id()` into a single call. Given a taxon name, returns its complete lineage from root to tip.

Usage

```
get_lineage(taxon, clean = TRUE, verbose = FALSE)
```

Arguments

`taxon`	A character string giving the taxon name.
`clean`	Logical. If `TRUE` (default), removes philosophical root nodes and cleans formatting markers.
`verbose`	Logical. If `TRUE`, prints progress messages. Default `FALSE`.

Value

A character vector of clade names ordered from root to tip, or `NULL` if the taxon cannot be found.

Examples

```
get_lineage("Tyrannosaurus")
get_lineage("Homo sapiens")
get_lineage("Quercus robur")
```

`get_lineage_by_id` *Retrieve the full taxonomic lineage of a taxon*

Description

Given a Taxonomicon numeric ID, retrieves and parses the complete hierarchical lineage from root (Natura) to the taxon itself. The lineage is returned as a character vector ordered from root to tip.

Usage

```
get_lineage_by_id(taxon_id, clean = TRUE, verbose = FALSE)
```

Arguments

`taxon_id`	A numeric or character string giving the Taxonomicon ID. Obtain this with `get_taxonomicon_id()`.
`clean`	Logical. If `TRUE` (default), removes philosophical root nodes above `Biota` (i.e., Natura, actualia, Mundus, naturalia) and strips dagger and superscript markers from names.
`verbose`	Logical. If `TRUE`, prints status messages. Default `FALSE`.

Details

Lineage data is sourced from The Taxonomicon, which is based on Systema Naturae 2000 (Brands, S.J., 1989 onwards). The depth of lineages in The Taxonomicon substantially exceeds that of other programmatic sources such as the Open Tree of Life, particularly for well-studied clades such as Dinosauria, where intermediate clades at the level of superfamilies, tribes, and named subclades are fully resolved.

Value

A character vector of clade names from root to tip, or `NULL` if retrieval fails.

See Also

`get_lineage()`, `taxo_distance()`

Examples

```
id <- get_taxonomicon_id("Tyrannosaurus")
lin <- get_lineage_by_id(id)
print(lin)
```

`get_taxonomicon_id` *Find the Taxonomicon ID for a taxon name*

Description

Queries The Taxonomicon (taxonomy.nl) to retrieve the internal numeric identifier for a given taxon name. The search filters out non-biological entities such as astronomical objects that may share the same name.

Usage

```
get_taxonomicon_id(taxon, verbose = FALSE)
```

Arguments

`taxon`	A character string giving the taxon name to search for. Typically a genus name (e.g., `"Tyrannosaurus"`) but species and higher ranks are also supported.
`verbose`	Logical. If `TRUE`, prints status messages during retrieval. Default is `FALSE`.

Details

The function queries the static search endpoint at `taxonomicon.taxonomy.nl/TaxonList.aspx` and parses the resulting HTML to extract the taxon ID from the hierarchy link. When multiple matches exist (e.g., a genus name shared with an astronomical object), biological entries are prioritised by filtering for entries annotated as dinosaur, reptile, archosaur, animal, plant, fungus, or bacterium.

Value

A character string containing the Taxonomicon numeric ID, or `NULL` if the taxon is not found.

See Also

`get_lineage()`, `taxo_distance()`

Examples

```
get_taxonomicon_id("Tyrannosaurus")   # returns "50841"
get_taxonomicon_id("Homo")
get_taxonomicon_id("Quercus")
```

`is_member` *Test whether one taxon is nested within another*

Description

Returns `TRUE` if `taxon` is a member of `clade` — i.e., if the clade name appears in the taxon's lineage.

Usage

```
is_member(taxon, clade, verbose = FALSE)
```

Arguments

`taxon`	A character string giving the taxon name to test.
`clade`	A character string giving the clade name to test membership in.
`verbose`	Logical. If `TRUE`, prints progress messages. Default `FALSE`.

Value

A logical value, or `NULL` if the taxon cannot be found.

Examples

```
is_member("Tyrannosaurus", "Theropoda")    # TRUE
is_member("Triceratops", "Theropoda")      # FALSE
is_member("Homo", "Amniota")               # TRUE
```

`lineage_depth` *Get the lineage depth of a taxon*

Description

Returns the number of nodes in the lineage of a taxon, from root to tip. This reflects how deeply nested the taxon is within the taxonomic hierarchy.

Usage

```
lineage_depth(taxon, verbose = FALSE)
```

Arguments

`taxon`	A character string giving the taxon name.
`verbose`	Logical. If `TRUE`, prints progress messages. Default `FALSE`.

Value

An integer giving the lineage depth, or `NULL` if the taxon cannot be found.

Examples

```
lineage_depth("Tyrannosaurus")   # deep - many intermediate clades
lineage_depth("Biota")           # shallow - near root
```

`load_cache` *Load a previously saved taxodist cache from disk*

Description

Restores lineage data saved with `save_cache()` into the current session cache, avoiding network requests for taxa already retrieved in a previous session.

Usage

```
load_cache(file)
```

Arguments

`file`	Path to an `.rds` file created by `save_cache()`.

Value

Invisibly returns `NULL`.

See Also

`save_cache()`, `clear_cache()`

Examples

```
tmp <- tempfile(fileext = ".rds")
save_cache(tmp)
load_cache(tmp)
unlink(tmp)
```

`mrca` *Compute the most recent common ancestor of two taxa*

Description

Retrieves the lineages of two taxa and returns the name of their most recent common ancestor (MRCA) — the deepest node shared by both lineages.

Usage

```
mrca(taxon_a, taxon_b, verbose = FALSE)
```

Arguments

`taxon_a`	A character string giving the first taxon name.
`taxon_b`	A character string giving the second taxon name.
`verbose`	Logical. If `TRUE`, prints progress messages. Default `FALSE`.

Value

A character string giving the name of the MRCA, or `NULL` if either taxon cannot be found or no common ancestor exists.

Examples

```
mrca("Tyrannosaurus", "Velociraptor")   # "Tyrannoraptora"
mrca("Tyrannosaurus", "Triceratops")    # "Dinosauria"
mrca("Tyrannosaurus", "Homo")           # "Amniota"
```

`plot.taxodist_cluster` *Plot method for taxodist_cluster objects*

Description

Plots the hierarchical clustering dendrogram produced by `taxo_cluster()`.

Usage

```
## S3 method for class 'taxodist_cluster'
plot(
  x,
  main = "Taxonomic Clustering",
  xlab = "",
  sub = paste("Method:", if (!is.null(x$hclust)) x$hclust$method else "None"),
  ...
)
```

Arguments

`x`	A `taxodist_cluster` object from `taxo_cluster()`.
`main`	Plot title. Default: `"Taxonomic Clustering"`.
`xlab`	X-axis label. Default: `""`.
`sub`	Subtitle. Default: clustering method used.
`...`	Additional arguments passed to `stats::plot.hclust()`.

Value

Invisibly returns `x`. Called for side effects (plotting).

Examples

```
taxa <- c("Tyrannosaurus", "Velociraptor", "Triceratops", "Brachiosaurus")
cl <- taxo_cluster(taxa)
plot(cl)
```

`plot.taxodist_ord` *Plot method for taxodist_ord objects*

Description

Plots the PCoA ordination produced by `taxo_ordinate()`.

Usage

```
## S3 method for class 'taxodist_ord'
plot(
  x,
  main = "Taxonomic Ordination (PCoA)",
  xlab = "PC1",
  ylab = "PC2",
  labels = if (!is.null(x$points)) rownames(x$points) else NULL,
  ...
)
```

Arguments

`x`	A `taxodist_ord` object from `taxo_ordinate()`.
`main`	Plot title. Default: `"Taxonomic Ordination (PCoA)"`.
`xlab`	X-axis label. Default: `"PC1"`.
`ylab`	Y-axis label. Default: `"PC2"`.
`labels`	Character vector of labels. Default: rownames of points.
`...`	Additional arguments passed to `graphics::text()`.

Value

Invisibly returns `x`. Called for side effects (plotting).

Examples

```
taxa <- c("Tyrannosaurus", "Velociraptor", "Triceratops", "Brachiosaurus")
ord <- taxo_ordinate(taxa)
plot(ord)
```

`print.taxodist_path` *Print method for taxodist_path objects*

Description

Print method for taxodist_path objects

Usage

```
## S3 method for class 'taxodist_path'
print(x, ...)
```

Arguments

`x`	A `taxodist_path` object from `taxo_path()`.
`...`	Additional arguments (ignored).

Value

Invisibly returns `x`. Called for side effects (printing).

`print.taxodist_result` *Print method for taxodist distance results*

Description

Print method for taxodist distance results

Usage

```
## S3 method for class 'taxodist_result'
print(x, ...)
```

Arguments

`x`	A list returned by `taxo_distance()`.
`...`	Additional arguments (ignored).

Value

Invisibly returns `x`. Called for side effects (printing).

`save_cache` *Save the taxodist lineage cache to disk*

Description

Serialises the current session cache to an `.rds` file so it can be restored in a future session with `load_cache()`. Useful for reproducibility and for avoiding repeated network requests.

Usage

```
save_cache(file)
```

Arguments

`file`	Path to the `.rds` file to write.

Value

Invisibly returns `NULL`.

See Also

`load_cache()`, `clear_cache()`

Examples

```
tmp <- tempfile(fileext = ".rds")
save_cache(tmp)
unlink(tmp)
```

`shared_clades` *List all clades shared between two taxa*

Description

Returns the vector of clade names forming the shared trunk of two taxa's lineages, from root down to (and including) their MRCA.

Usage

```
shared_clades(taxon_a, taxon_b, verbose = FALSE)
```

Arguments

`taxon_a`	A character string giving the first taxon name.
`taxon_b`	A character string giving the second taxon name.
`verbose`	Logical. If `TRUE`, prints progress messages. Default `FALSE`.

Value

A character vector of shared clade names ordered from root to MRCA, or `NULL` if either taxon cannot be found.

Examples

```
shared_clades("Tyrannosaurus", "Velociraptor")
shared_clades("Tyrannosaurus", "Homo")
```

`summary.taxodist_ord` *Summary method for taxodist_ord objects*

Description

Prints a summary of the PCoA ordination, including the goodness-of-fit and the proportion of variance explained by each principal coordinate.

Usage

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

Arguments

`object`	A `taxodist_ord` object from `taxo_ordinate`.
`...`	Additional arguments (ignored).

Value

Invisibly returns a data frame containing the eigenvalues and variance explained by each dimension.

`taxo_cluster` *Cluster taxa by taxonomic distance*

Description

Computes pairwise taxonomic distances and performs hierarchical clustering.

Usage

```
taxo_cluster(taxa, method = "average", ...)
```

Arguments

`taxa`	A character vector of taxon names, or a `dist` object from `distance_matrix()`.
`method`	Clustering method passed to `stats::hclust()`. Default `"average"` (UPGMA), which works well with taxonomic distances.
`...`	Additional arguments passed to `distance_matrix()` (e.g. `verbose`, `progress`).

Value

An object of class `"taxodist_cluster"` — a list with:

`hclust` The `stats::hclust()` result.

`dist` The underlying distance matrix.

See Also

`taxo_ordinate()`, `distance_matrix()`

Examples

```
taxa <- c("Tyrannosaurus", "Velociraptor", "Triceratops", "Brachiosaurus")
cl <- taxo_cluster(taxa)
if (!is.null(cl$hclust)) {
  plot(cl)
}
```

`taxo_distance` *Compute the phylogenetic distance between two taxa*

Description

Given two taxon names, retrieves their lineages from The Taxonomicon and computes a taxonomic distance based on the depth of their most recent common ancestor (MRCA):

Usage

```
taxo_distance(taxon_a, taxon_b, verbose = FALSE)
```

Arguments

`taxon_a`	A character string giving the first taxon name.
`taxon_b`	A character string giving the second taxon name.
`verbose`	Logical. If `TRUE`, prints progress messages. Default `FALSE`.

Details

$$d(A,B) = \frac{1}{\mathrm{depth}(\mathrm{MRCA}(A,B))}$$

The deeper the shared ancestor, the smaller (closer to zero) the distance. This metric ensures that taxa diverging at the same node are always equidistant from any third taxon, regardless of lineage depth differences below the split.

Value

A named list of class `"taxodist_result"` with the following elements:

`distance`	Numeric. The distance between the two taxa. Returns 0 if one taxon is an ancestor of the other.
`mrca`	Character. The name of the most recent common ancestor.
`mrca_depth`	Integer. The depth of the MRCA node.
`depth_a`	Integer. The lineage depth of taxon A.
`depth_b`	Integer. The lineage depth of taxon B.
`taxon_a`	Character. Name of the first taxon.
`taxon_b`	Character. Name of the second taxon.

Returns `NULL` if either taxon cannot be found.

References

Brands, S.J. (1989 onwards). Systema Naturae 2000. Amsterdam, The Netherlands. Retrieved from The Taxonomicon, `http://taxonomicon.taxonomy.nl`.

See Also

`mrca()`, `distance_matrix()`, `get_lineage()`

Examples

```
# Distance between two theropods
taxo_distance("Tyrannosaurus", "Velociraptor")

# Distance between very distantly related taxa
taxo_distance("Tyrannosaurus", "Quercus")

# Distance between two oviraptorid genera
taxo_distance("Nomingia", "Huanansaurus")
```

`taxo_heatmap` *Plot a taxonomic heatmap*

Description

Computes pairwise taxonomic distances and plots a heatmap with hierarchical clustering dendrograms on the margins. Darker/hotter colors typically represent smaller distances (closer relatives).

Usage

```
taxo_heatmap(taxa, ...)
```

Arguments

`taxa`	A character vector of taxon names, or a `dist` object from `distance_matrix`.
`...`	Additional arguments passed to `heatmap`.

Value

Invisibly returns the underlying `dist` object. Called primarily for its side effect (plotting).

Examples

```
taxa <- c("Tyrannosaurus", "Velociraptor", "Homo", "Panthera", "Quercus")
taxo_heatmap(taxa)
```

`taxo_ordinate` *Ordinate taxa in taxonomic distance space*

Description

Computes pairwise taxonomic distances and applies classical multidimensional scaling (PCoA) to project taxa into a low-dimensional space.

Usage

```
taxo_ordinate(taxa, k = 2, ...)
```

Arguments

`taxa`	A character vector of taxon names, or a `dist` object from `distance_matrix()`.
`k`	Number of dimensions. Default `2`.
`...`	Additional arguments passed to `distance_matrix()`.

Value

An object of class `"taxodist_ord"` — a list with:

`points` A matrix of coordinates (taxa x k dimensions).

`dist` The underlying distance matrix.

`GOF` Goodness-of-fit from`stats::cmdscale()`.

`eig` The eigenvalues computed during PCoA.

See Also

`taxo_cluster()`, `distance_matrix()`

Examples

```
taxa <- c("Tyrannosaurus", "Velociraptor", "Triceratops", "Brachiosaurus")
ord <- taxo_ordinate(taxa)
if (!is.null(ord$points)) {
  plot(ord$points, type = "n")
  text(ord$points, labels = rownames(ord$points))
}
```

`taxo_path` *Get the taxonomic path between two taxa*

Description

Returns the full node-by-node path from one taxon up to their most recent common ancestor (MRCA) and back down to the other taxon. The result is a data frame with one row per node, making it easy to inspect, filter, or pipe into other functions.

Usage

```
taxo_path(taxon_a, taxon_b, verbose = FALSE)
```

Arguments

`taxon_a`	A character string giving the first taxon name.
`taxon_b`	A character string giving the second taxon name.
`verbose`	Logical. If `TRUE`, prints progress messages. Default `FALSE`.

Value

A data frame of class `"taxodist_path"` with columns:

`node` Character. The clade or taxon name at this step.

`depth` Integer. The depth of this node in the full lineage of its side (or the shared lineage for the MRCA).

`direction` Character. One of `"a"` (ascending from taxon A to MRCA), `"mrca"` (the shared ancestor), or `"b"` (descending from MRCA to taxon B).

Returns `NULL` if either taxon cannot be found.

See Also

`mrca()`, `shared_clades()`, `compare_lineages()`

Examples

```
taxo_path("Tyrannosaurus", "Velociraptor")
taxo_path("Tyrannosaurus", "Homo")
```

`taxo_search` *Search The Taxonomicon for a taxon name*

Description

Queries The Taxonomicon database and returns a data frame of all available biological entries matching the search string. This is particularly useful for exploring homonyms, historical ranks, or taxonomic synonyms before computing distances.

Usage

```
taxo_search(taxon, verbose = FALSE)
```

Arguments

`taxon`	A character string giving the taxon name to search for.
`verbose`	Logical. If `TRUE`, prints status messages. Default `FALSE`.

Value

A data frame of class `"data.frame"` with columns:

`id` Character. The numeric Taxonomicon ID.

`name` Character. The full taxon description, including rank and author.

Returns `NULL` if no matches are found.

See Also

`get_lineage()`, `taxo_distance()`

Examples

```
taxo_search("Bacteria")
taxo_search("Nereis")
taxo_search("Tyrannosaurus")
```

Index