Package ‘Rcompadre’

April 30, 2021

Type Package

Title Utilities for using the 'COM(P)ADRE' Matrix Model Database

Version 1.0.0


License GPL-3

URL https://github.com/jonesor/Rcompadre

BugReports https://github.com/jonesor/Rcompadre/issues

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

Depends R (>= 2.10)

Imports methods, popdemo, tibble

Suggests spelling, knitr, markdown, testthat, popbio, ggplot2, maps, dplyr, ggtern, covr, rcrossref

VignetteBuilder knitr

Collate 'CompadreDB.R' 'CompadreDB-Methods.R'
  'CompadreDB-Subsetting.R' 'CompadreDB-Tidyverse.R'
  'CompadreMat.R' 'ClassUnionMethods.R' 'cdb_check_species.R'
  'cdbCollapse.R' 'cdb_compare.R' 'cdb_fetch.R' 'cdb_flag.R'
  'cdb_flatten.R' 'cdb_id.R' 'cdb_id_stages.R' 'cdb_id_studies.R'
  'cdb_mean_matF.R' 'cdb_metadata.R' 'cdb_rbind.R'
  'cdb_unflatten.R' 'cdb_unnest.R' 'data.R' 'mpm_mean.R'
  'mpm_methods.R' 'string_representation.R' 'zzz.R'

Language en-GB

Config/testthat/parallel true

Config/testthat/edition 3

NeedsCompilation no
Author  Patrick Barks [aut] (<https://orcid.org/0000-0002-5947-8151>),
Danny Buss [aut],
Roberto Salguero-Gomez [aut] (<https://orcid.org/0000-0002-6085-4433>),
Iain Stott [aut] (<https://orcid.org/0000-0003-2724-7436>),
William K. Petry [aut] (<https://orcid.org/0000-0002-5230-5987>),
Tamora James [aut] (<https://orcid.org/0000-0003-1363-4742>),
Owen Jones [aut, cre] (<https://orcid.org/0000-0001-5720-4686>),
Julia Jones [aut] (<https://orcid.org/0000-0001-9223-1778>),
Gesa Römer [aut] (<https://orcid.org/0000-0002-4859-5870>),
Sam Levin [aut] (<https://orcid.org/0000-0002-3289-9925>)

Maintainer  Owen Jones <jones@biology.sdu.dk>

Repository  CRAN

Date/Publication  2021-04-30 12:40:02 UTC

R topics documented:

as_cdb ......................................................... 3
cdb_check_species ........................................... 3
cdbCollapse  .................................................. 4
cdb_compare .................................................... 6
cdb_fetch ....................................................... 7
cdb_flag ......................................................... 8
cdb_flatten .................................................... 9
cdb_id .......................................................... 10
cdb_id_stages ................................................. 11
cdb_id_studies ............................................... 12
cdb_mean_matF ................................................ 13
cdb_metadata .................................................. 14
cdb_rbind ...................................................... 15
cdb_unflatten ................................................ 16
cdb_unnest ..................................................... 17
Compadre ..................................................... 18
CompadreDB ................................................... 20
CompadreDB-Methods ......................................... 22
CompadreDB-Subsetting ...................................... 23
CompadreDB-Tidyverse ....................................... 24
CompadreLegacy .............................................. 26
CompadreMatrixMethods ..................................... 29
mpm_mean ...................................................... 31
mpm_methods ................................................... 32
string_representation ....................................... 34

Index  36
as_cdb

(Convert legacy COM(P)ADRE database object to CompadreDB)

Description

Convert a legacy COM(P)ADRE database object (of class 'list') to a CompadreDB object.

Usage

as_cdb(from)

Arguments

from

A legacy COM(P)ADRE database

Value

A CompadreDB object

Author(s)

Iain M. Stott

Examples

Compadre <- as_cdb(CompadreLegacy)

cdb_check_species

(Compadre)

Description

Takes a vector of species names and checks whether those species are represented within a CompadreDB object. It outputs either a data frame depicting the species of interest and whether they occur in the database (TRUE/FALSE), or, if return_db == TRUE, a CompadreDB object subset to the species of interest.

Usage

cdb_check_species(cdb, species, return_db = FALSE)
Arguments

cdb          A CompadreDB object
species      Character vector of binomial species names, with the genus and specific epithet
             separated by either an underscore or a space (e.g. c("Acipenser_fulvescens","Borrelia_burgdorferi")
return_db    Logical argument indicating whether a database should be returned

Value

If return_db == FALSE, returns a data frame with a column of species names and a column indi-
- cating whether a species occurs in the database. If return_db == TRUE, returns a subset of cdb
containing only those species within argument species.

Author(s)

Danny Buss <db50@cam.ac.uk>
Owen R. Jones <jones@biology.sdu.dk>
Rob Salguero-Gómez <rob.salguero@zoo.ox.ac.uk>
Patrick Barks <patrick.barks@gmail.com>

See Also

Other data checking: cdb_collapse(), cdb_compare(), cdb_flag(), mpm_methods

Examples

species <- c("Primula vulgaris", "Trillium ovatum", "Homo sapiens")
cdb_check_species(Compadre, species)
CompadreSubset <- cdb_check_species(Compadre, species, return_db = TRUE)

cdb_collapse          Collapse a COM(P)ADRE database by averaging matrices over levels
                       of one or more grouping variables

Description

Collapses a CompadreDB object by averaging matrices over levels of one or more grouping vari-
- ables (e.g. SpeciesAuthor, MatrixPopulation).

For a given study and species, a COM(P)ADRE database may contain multiple matrices, reflecting
different combinations of population, year, and/or treatment. Collapsing allows a user to obtain a
single ‘grand mean matrix’ for each group of interest (e.g. MatrixPopulation), and therefore limit
pseudoreplication.

All members of a group must have the same matrix dimension (consider adding MatrixDimension
as a grouping variable). All members of a group should have the same AnnualPeriodicity and matrix
stage class definitions (see cdb_id_stages). Note that Seasonal matrices should not be collapsed
using this method (they should be matrix-multiplied rather than averaged).
Usage

cdb_collapse(cdb, columns)

Arguments

cdb A CompadreDB object
columns Vector of grouping variables to collapse over (corresponding to columns within cdb)

Details

Will give a warning if members of any group do not all share the same AnnualPeriodicity or stage class definitions, or if cdb contains any rows with a MatrixComposite value of "Seasonal".

Prior to collapsing, columns of class 'factor' will be coerced to 'character', and any list-column apart from mat will be removed.

Within a group, rows of a given column are collapsed as follows:

- mat: matrices are averaged using mpm_mean, and stage class definitions are taken from the first group member
- MatrixComposite: return original value if N = 1, else return "Collapsed"
- Lat: re-calculated by averaging Lat column (if available)
- Lon: re-calculated by averaging Lon column (if available)
- SurvivalIssue: re-calculated from the collapsed mat (max(colSums(matU)))
- others: if all elements equal return that unique value, else paste together all unique values separated by "; "

Value

A CompadreDB object

Author(s)

Patrick M. Barks <patrick.barks@gmail.com>
Owen R. Jones <jones@biology.sdu.dk>

See Also

cdb_id_stages

Other data checking: cdb_check_species(), cdb_compare(), cdb_flag(), mpm_methods
Examples

# filter out Seasonal matrices
CompSub <- subset(Compadre, MatrixComposite != "Seasonal")

# add column identifying unique stage class definitions
CompSub$id_stage <- cdb_id_stages(CompSub, "MatrixClassOrganized")

# collapse
CompCollapse <- cdb_collapse(CompSub, columns = c("id_stage"))

cdb_compare

Compare two versions or subsets of a COM(P)ADRE database

Description

Prints a summary of the differences between two CompadreDB objects, including the number of species, studies, and matrices in each. If argument verbose == TRUE, additionally prints a list of the species and studies that are present in one database but not the other.

Usage

cdb_compare(cdb1, cdb2, verbose = FALSE)

Arguments

cdb1, cdb2 CompadreDB objects to compare

verbose Logical argument indicating whether or not to return lots of detail

Value

NULL. Output is printed rather than returned.

Author(s)

Owen R. Jones <jones@biology.sdu.dk>

See Also

Other data checking: cdb_check_species(), cdb_collapse(), cdb_flag(), mpm_methods

Examples

Compadre1 <- subset(Compadre, Continent == "Asia")
Compadre2 <- subset(Compadre, Continent == "Africa")

cdb_compare(Compadre1, Compadre2)
**Description**

Fetches the current version of a COM(P)ADRE database from https://compadre-db.org, or load any version stored in a local .RData file.

**Usage**

cdb_fetch(cdb)

**Arguments**

cdb Either "comadre" or "compadre" (case insensitive) to fetch the most recent database from https://compadre-db.org, or a path to an existing COMPADRE database (i.e. .RData file) stored on the local machine.

**Value**

A CompadreDB object

**Author(s)**

Patrick M. Barks <patrick.barks@gmail.com>

Owen R. Jones <jones@biology.sdu.dk>

**See Also**

Other data acquisition: cdb_metadata()

**Examples**

```r
## Not run:
#Download direct from the COMPADRE website
compadre <- cdb_fetch("comadre")

#using file path to downloaded data
compadre <- cdb_fetch("data/COMPADRE_v.5.0.1.RData")

## End(Not run)
```
**Description**

Adds columns to the data slot of a CompadreDB object that flag potential problems in the matrix population models. These columns can subsequently be used to subset the database by logical argument.

Optional checks include:

- `check_NA_A`: missing values in matA?
- `check_NA_U`: missing values in matU?
- `check_NA_F`: missing values in matF?
- `check_NA_C`: missing values in matC?
- `check_zero_U`: matU all zeros (including NA)?
- `check_singular_U`: matU singular (i.e. non-invertable)?
- `check_component_sum`: do matU/matF/matC components sum to matA (see Details)?
- `check_ergodic`: is matA ergodic (see `isErgodic`)?
- `check_irreducible`: is matA irreducible (see `isIrreducible`)?
- `checkPrimitive`: is matA primitive (see `isPrimitive`)?
- `check_surv_gte_1`: does matU contain values that are equal to or greater than 1?

**Usage**

```r
cdb_flag(cdb, checks = c("check_NA_A", "check_NA_U", "check_NA_F", "check_NA_C", "check_zero_U", "check_singular_U", "check_component_sum", "check_ergodic", "check_irreducible", "check_primitive", "check_surv_gte_1"))
```

**Arguments**

- **cdb** A CompadreDB object
- **checks** Character vector specifying which checks to run. Defaults to all, i.e. `c("check_NA_A", "check_NA_U", "check_NA_F", "check_NA_C", "check_zero_U", "check_singular_U", "check_component_sum", "check_ergodic", "check_irreducible", "check_primitive", "check_surv_gte_1")`

**Details**

For the flag `check_component_sum`, a value of NA will be returned if the matrix sum of matU, matF, and matC consists only of zeros and/or NA, indicating that the matrix has not been split.
Value

Returns `cdb` with extra columns appended to the data slot (columns have the same names as the corresponding elements of `checks`) to indicate (TRUE/FALSE) whether there are potential problems with the matrices corresponding to a given row of the data.

Author(s)

Owen Jones <jones@biology.sdu.dk>
Julia Jones <juliajones@biology.sdu.dk>
Roberto Salguero-Gomez <rob.salguero@zoo.ox.ac.uk>
Danny Buss <dlb50@cam.ac.uk>
Patrick Barks <patrick.barks@gmail.com>

See Also

Other data checking: `cdb_check_species()`, `cdb Collapse()`, `cdb_compare()`, `mpm_methods`

Examples

```r
CompadreFlag <- cdb_flag(Compadre)

# only check whether matA has missing values, and whether matA is ergodic
CompadreFlag <- cdb_flag(Compadre, checks = c("check_NA_A", "check_ergodic"))
```

---

cdb_flatten  

Convert a COM(P)ADRE database to a flat data frame with matrices and vectors stored in string representation

Description

Converts a CompadreDB object to a flat data frame by extracting the data slot, and splitting the mat column into separate columns for each component (matrices matA, matU, matF, matC, and vectors MatrixClassAuthor, and MatrixClassOrganized). The component matrices and vectors within the six new columns are stored in string format so that the database can be written to a flat file format such as csv (see `string_representation`).

Usage

cdb_flatten(cdb)

Arguments

cdb  

A CompadreDB object
A data frame based on the data slot of cdb, but with the column mat replaced by six separate columns (for matrices matA, matU, matF, matC, and vectors MatrixClassAuthor, and MatrixClassOrganized), whose elements are matrices or vectors in string representation.

Author(s)
Owen R. Jones <jones@biology.sdu.dk>
Patrick M. Barks <patrick.barks@gmail.com>

See Also
cdb_unflatten string_representation
Other data management: cdb_id_stages(), cdb_id_studies(), cdb_id(), cdb_mean_matF(), cdb_rbind(), cdb_unflatten(), cdb_unnest(), mpm_mean(), string_representation

Examples
CompadreFlat <- cdb_flatten(Compadre)

---

**Description**

Creates a vector of integer identifiers corresponding to the rows of a CompadreDB object, based on unique combinations of the elements in a given set of columns.

**Usage**
cdb_id(cdb, columns)

**Arguments**
cmdb A CompadreDB object
columns Vector of column names from which unique combinations should be identified

**Details**
Identifiers are assigned by pasting together the relevant columns, assigning factor levels based on alphabetical order, and then converting the factor levels to integers.

**Value**
Vector of integer identifiers corresponding to the rows of cdb, based on unique combinations of the elements in columns.
cdb_id_stages

Author(s)
Patrick Barks <patrick.barks@gmail.com>

See Also
Other data management: cdb_flatten(), cdb_id_stages(), cdb_id_studies(), cdb_mean_matF(), cdb_rbind(), cdb_unflatten(), cdb_unnest(), mpm_mean(), string_representation

Examples
cdb_id(Compadre, columns = c("SpeciesAuthor", "MatrixTreatment"))

---

cdb_id_stages
Create integer identifiers for a COM(P)ADRE database corresponding to unique combinations of species and matrix stage class definitions

Description
Creates a vector of integer identifiers corresponding to the rows of a CompadreDB object, based on unique combinations of the column 'SpeciesAccepted' and a list of matrix stage class definitions (either 'MatrixClassAuthor' or 'MatrixClassOrganized').

Usage
cdb_id_stages(cdb, stage_def = "MatrixClassAuthor")

Arguments
- **cdb**: A CompadreDB object
- **stage_def**: Whether to define matrix stage class based on "MatrixClassAuthor" or "MatrixClassOrganized" (see Details). Defaults to "MatrixClassAuthor".

Details
The vector 'MatrixClassOrganized' reflects standardized stage classes ('prop', 'active', or 'dorm'), whereas 'MatrixClassAuthor' reflects a description of the stage classes as defined by the original author (e.g. c('Seedling', 'Medium rosette', 'Large (2 rosettes)', 'Flowering')).

Because the 'MatrixClassAuthor' definitions are less standardized, they are more prone to typos that could lead to slight differences between stage descriptions of matrices that really do have the same stage classes (e.g. a set of matrices from a single study/species/population). Therefore, using 'MatrixClassAuthor' to define stage classes is potentially prone to mistakenly 'splitting' identifiers that should really be the same.

'MatrixClassOrganized' has the opposite problem. It's possible for two matrices from a given study to have the same stage definitions based on 'MatrixClassOrganized', but legitimately differ in stage definitions as defined by the author. Therefore, using 'MatrixClassAuthor' to define stage classes is potentially prone to mistakenly 'lumping' identifiers that should actually differ.
Because the majority of studies in COM(P)ADRE use a single set of stage definitions for all matrices, and typos are rare, results for the different stage definitions will usually be similar. Note, however, that the actual integers returned for the different stage definitions are likely to be very different (because they are based on alphabetical order).

Value

Vector of integer identifiers corresponding to the rows of cdb.

Author(s)

Patrick Barks <patrick.barks@gmail.com>

See Also

cdb_id

Other data management: cdb_flatten(), cdb_id_studies(), cdb_id(), cdb_mean_matF(), cdb_rbind(), cdb_unflatten(), cdb_unnest(), mpm_mean(), string_representation

Examples

cdb_id_stages(Compadre, stage_def = "MatrixClassOrganized")

cdb_id_studies(cdb, columns = c("Authors", "Journal", "YearPublication", "DOI.ISBN"))

Arguments

cdb A CompadreDB object

columns Vector of column names from which unique combinations should be identified. Defaults to c("Authors", "Journal", "YearPublication", "DOI.ISBN").
**Details**

Identifiers are assigned by pasting together the relevant columns, assigning factor levels based on alphabetical order, and then converting the factor levels to integers.

**Value**

Vector of integer study identifiers corresponding to the rows of `cdb`, based on unique combinations of the elements in `columns`.

**Author(s)**

Patrick Barks <patrick.barks@gmail.com>

**See Also**

`cdb_id`

Other data management: `cdb_flatten()`, `cdb_id_stages()`, `cdb_id()`, `cdb_mean_matF()`, `cdb_rbind()`, `cdb_unflatten()`, `cdb_unnest()`, `mpm_mean()`, `string_representation`

**Examples**

```r
Compadre$StudyID <- cdb_id_studies(Compadre)
```

---

**Description**

Takes a CompadreDB object and calculates a grand mean fecundity matrix for each unique population (a mean of all population-specific fecundity matrices, including fecundity matrices for which `MatrixComposite == 'Mean'`).

Populations are defined based on unique combinations of the columns `SpeciesAuthor`, `MatrixPopulation`, and `MatrixDimension`, (or optionally, a different set of columns supplied by the user).

The main purpose of this function is to identify stage classes that are potentially reproductive (i.e. the absence of fecundity in a given stage class and year does not necessarily indicate that the stage in question is non-reproductive).

**Usage**

```r
cdb_mean_matF(
  cdb,
  columns = c("SpeciesAuthor", "MatrixPopulation", "MatrixDimension")
)
```
Arguments

- **cdb**: A CompadreDB object
- **columns**: Vector of column names from which unique populations should be identified. Defaults to c("SpeciesAuthor", "MatrixPopulation", "MatrixDimension").

Value

Returns a list of matrices, representing the mean fecundity matrix associated with each row of the database.

Author(s)

- Owen R. Jones <jones@biology.sdu.dk>
- Danny Buss <dlb50@cam.ac.uk>
- Julia Jones <juliajones@biology.sdu.dk>
- Iain Stott <stott@biology.sdu.dk>
- Patrick Barks <patrick.barks@gmail.com>

See Also

Other data management: cdb_flatten(), cdb_id_stages(), cdb_id_studies(), cdb_id(), cdb_rbind(), cdb_unflatten(), cdb_unnest(), mpm_mean(), string_representation

Examples

```r
# print matF associated with row 16 of database
Compadre$mat[[16]]

# create list of meanMatFs
meanF <- cdb_mean_matF(Compadre)

# print meanMatF associated with row 16 of database
meanF[[16]]
```

---

cdb_metadata

**Extract metadata from a COM(P)ADRE database**

Description

Extract a tibble with only metadata information from a CompadreDB object, by dropping the matrix column "mat".

Usage

cdb_metadata(cdb)
**Arguments**

cdb  A CompadreDB object

**Details**

Transforms the large CompadreDB object into a tibble and drops the matrix column ("mat").

**Value**

Tibble with all metadata columns of cdb

**Author(s)**

Gesa Römer <gesa.roemer@gmail.com>

**See Also**

Other data acquisition: `cdb_fetch()`

**Examples**

Compadre_metadata <- cdb_metadata(Compadre)

cdb_rbind(cdb1, cdb2)

**Description**

Merges two CompadreDB objects via a row-bind of the data slots.

**Usage**

`cdb_rbind(cdb1, cdb2)`

**Arguments**

cdb1, cdb2  CompadreDB objects

**Value**

A CompadreDB object created by binding the rows of cdb1 and cdb2

**Author(s)**

Sam Levin <levisc8@gmail.com>

Owen R. Jones <jones@biology.sdu.dk>
See Also

Other data management: `cdb_flatten()`, `cdb_id_stages()`, `cdb_id_studies()`, `cdb_id()`, `cdb_mean_matF()`, `cdb_unflatten()`, `cdb_unnest()`, `mpm_mean()`, `string_representation`

Examples

```r
Compadre1 <- subset(Compadre, Continent == "Asia")
Compadre2 <- subset(Compadre, Continent == "Africa")

cdb_rbind(Compadre1, Compadre2)
```

---

### `cdb_unflatten`

**Reconstitute a flattened COM(P)ADRE database (created by `cdb_flatten`) as a CompadreDB object**

**Description**

Converts a flattened COM(P)ADRE database (created by `cdb_flatten`) back to the CompadreDB class

**Usage**

```r
cdb_unflatten(db)
```

**Arguments**

- `db`: A data frame created with `cdb_flatten`, with columns for matrices `matA`, `matU`, `matF`, `matC`, and vectors `MatrixClassAuthor`, and `MatrixClassOrganized` in string representation.

**Value**

A CompadreDB object. Because version details are lost when the database is flattened, the `Version` and `DateCreated` elements of the returned CompadreDB object will be NA.

**Author(s)**

Patrick M. Barks <patrick.barks@gmail.com>

**See Also**

- `cdb_flatten`, `string_representation`

Other data management: `cdb_flatten()`, `cdb_id_stages()`, `cdb_id_studies()`, `cdb_id()`, `cdb_mean_matF()`, `cdb_rbind()`, `cdb_unnest()`, `mpm_mean()`, `string_representation`
Unnest a COM(P)ADRE database by spreading the components of CompadreMat into separate list-columns

Unnests a CompadreDB object by spreading the components of CompadreMat into separate list-columns. Components that may be extracted include:

- `matA` (matrix)
- `matU` (matrix)
- `matF` (matrix)
- `matC` (matrix)
- `MatrixClassAuthor` (character vector)
- `MatrixClassOrganized` (character vector)
- `MatrixClassNumber` (integer vector)

Usage

```
cdb_unnest(
    cdb,
    components = c("matA", "matU", "matF", "matC", "MatrixClassAuthor", "MatrixClassOrganized", "MatrixClassNumber")
)
```

Arguments

- `cdb` A CompadreDB object
- `components` Character vector specifying which components to extract. Defaults to all, i.e. `c("matA", "matU", "matF", "matC", "MatrixClassAuthor", "MatrixClassOrganized", "MatrixClassNumber")`

Value

`cdb` with additional list-columns for each element of argument `components`

Author(s)

Patrick M. Barks <patrick.barks@gmail.com>
See Also

Other data management: `cdb_flatten()`, `cdb_id_stages()`, `cdb_id_studies()`, `cdb_id()`, `cdb_mean_matF()`, `cdb_rbind()`, `cdb_unflatten()`, `mpm_mean()`, `string_representation`

Examples

```r
# unnest all components
CompadreUnnest <- cdb_unnest(Compadre)

# unnest select components (matU and MatrixClassAuthor)
CompadreUnnest <- cdb_unnest(Compadre, c("matU", "MatrixClassAuthor"))
```

Description

Compadre (plant matrices) and Comadre (animal matrices) are subsamples of the COMPADRE Plant Matrix Database and COMADRE Animal Matrix Database, respectively, that are used for testing and examples. Each object is of class 'CompadreDB' and therefore has the following two slots: data and version.

For full documentation see the COMPADRE User Guide.

Usage

Compadre

Comadre

Format

Slot **data** - A tibble-style data frame with the following 48 columns:

- mat - A list of 'CompadreMat' objects, each with the following slots:
  - matA - A matrix population model
  - matU - The survival- and growth-related component of matA
  - matF - The sexual reproduction component of matA
  - matC - The clonal reproduction component of matA
  - matrixClass - A data frame with the following columns:
    - MatrixClassOrganized - Standardized stage class of the matrix population model
    - MatrixClassAuthor - Stage description from the original publication
    - MatrixClassNumber - Integer stage number
- SpeciesAuthor - Binomial species name given by the paper's author
• SpeciesAccepted - Accepted binomial species name taken from *The Plant List* or *Catalogue of Life*
• CommonName - Common name for species used in the publication
• Genus - Taxonomic genus that the accepted species belongs to
• Family - Family that the species belongs to
• Order - Order that the species belongs to
• Class - Class that the species belongs to
• Phylum - Phylum that the species belongs to
• Kingdom - Kingdom that the species belongs to
• OrganismType - Organism type (see COMPADRE User Guide for documentation)
• DicotMonoc - Whether the species is a dicot or monocot. Non-angiosperms are NA.
• AngioGymno - Whether the species is an angiosperm or gymnosperm. Non-plants are NA.
• Authors - Last name of all authors (separated with ",")
• Journal - Abbreviated journal title, or type of source document (e.g. "PhD thesis")
• YearPublication - Year of publication
• DOI.ISBN - Digital Object Identifier or International Standard Book Number codes to identify each publication
• AdditionalSource - Additional source(s) used to reconstruct the matrix or obtain additional metadata for the matrix (if applicable)
• StudyDuration - Number of years of observation in study (StudyEnd - StudyStart)
• StudyStart - Study start year
• StudyEnd - Study end year
• AnnualPeriodicity - Inverse of the length of the projection interval (in years)
• NumberPopulations - The number of study populations as defined by the authors. Within site replication of permanent plots is not defined as separate populations
• MatrixCriteriaSize - Indicates whether the matrix contains stages based on size. If so, indicates what that measure of size is
• MatrixCriteriaOntogeny - Indicates whether the matrix contains stages based on ontogenic/developmental stages
• MatrixCriteriaAge - Indicates whether the matrix contains stages based on age
• MatrixPopulation - Population name or definition of where the matrix was recorded, usually given by the author. See COMPADRE User Guide.
• Lat - Latitude in decimal degrees
• Lon - Longitude in decimal degrees
• Altitude - Altitude of study site (m above sea level)
• Country - 3-letter ISO country code for the country in which the study took place (multiple countries separated with ",")
• Continent - Continent on which study took place
• Ecoregion - Ecoregion in which study took place. See COMPADRE User Guide.
• StudiedSex - Whether study included only males ("M"), only females ("F"), or both sexes ("M/F")
• MatrixComposite - Indicates the type of matrix population model. Possible values are Individual, Mean, Pooled, and Seasonal. See COMPADRE User Guide.
• MatrixTreatment - Describes if a treatment was applied to the population or not. If yes, includes a brief description of the treatment. If not, Unmanipulated
• MatrixCaptivity - Whether species was studied in the wild (W), captivity (C), or captured from a wild population (CW)
• MatrixStartYear - First year of matrix
• MatrixStartSeason - First season of matrix as described by author (hemisphere-specific)
• MatrixStartMonth - First month of matrix
• MatrixEndYear - Final year of matrix
• MatrixEndSeason - Final season of matrix as described by author (hemisphere-specific)
• MatrixEndMonth - Final month of matrix
• MatrixSplit - Whether the matrix has been split into components U, F, and C ("Divided") or not ("Indivisible"). If not, elements of matU, matF, and matC are filled with NA
• MatrixFec - Whether fecundity was measured for the matrix model
• Observation - Additional observations recorded by database compilers
• MatrixDimension - Dimension of the A matrix
• SurvivalIssue - Denotes the maximum stage-specific survival value

Slot version - A list with the following elements:
• Version - The version number of the database
• DateCreated - The date that the .RData file was created
• Agreement - Link to the COMADRE license agreement

An object of class CompadreDB with 150 rows and 48 columns.
An object of class CompadreDB with 150 rows and 48 columns.
Usage

CompadreData(object)

## S4 method for signature 'CompadreDB'
CompadreData(object)

## S4 method for signature 'CompadreDB'
x$name

## S4 replacement method for signature 'CompadreDB'
x$name <- value

## S4 method for signature 'CompadreDB,ANY,missing'
x[[i, j, ...]]

## S4 replacement method for signature 'CompadreDB,ANY,missing'
x[[i, j]] <- value

VersionData(object)

## S4 method for signature 'CompadreDB'
VersionData(object)

Version(object)

## S4 method for signature 'CompadreDB'
Version(object)

DateCreated(object)

## S4 method for signature 'CompadreDB'
DateCreated(object)

Arguments

object A CompadreDB object
x A CompadreDB object
name The name of a column within x
value Vector of values to assign to the column
i, j elements to extract or replace (see \[.data.frame\)
... ignored

Slots

data A tibble-style data frame with a list-column of matrix population models (column mat) and a variety of other metadata columns.
version  A list with elements Version (database version number), DateCreated (date of version release), and Agreement (a url link to the User Agreement)

Author(s)

Iain M. Stott
Tamora D. James

See Also

CompadreDB-Methods CompadreDB-Subsetting

Examples

# extract entire 'data' slot
dat <- CompadreData(Compadre)

# access the date of database creation
DateCreated(Compadre)

# extract column SpeciesAccepted
Compadre$SpeciesAccepted

# create new list-column with stage-specific survival
Compadre$stage_survival <- lapply(Compadre$mat, function(x) colSums(x@matU))

Description

This page describes a variety of methods that can be used with CompadreDB objects, including common data frame operations (head, names, and merge), conversion methods (as.data.frame and as_tibble), and methods to calculate the number of species (NumberAcceptedSpecies), studies (NumberStudies), or matrices (NumberMatrices).

Usage

## S3 method for class 'CompadreDB'
as.data.frame(x, ...)

## S3 method for class 'CompadreDB'
as_tibble(x)

## S3 method for class 'CompadreDB'
head(x, n = 6L, ...)
CompadreDB-Subsetting

## S3 method for class 'CompadreDB'
tail(x, n = 6L, ...)
## S3 method for class 'CompadreDB'
names(x)
## S3 method for class 'CompadreDB'
dim(x)
## S3 method for class 'CompadreDB'
merge(x, y, ...)

NumberAcceptedSpecies(object)
## S4 method for signature 'CompadreDB'
NumberAcceptedSpecies(object)

NumberStudies(object)
## S4 method for signature 'CompadreDB'
NumberStudies(object)

NumberMatrices(object)
## S4 method for signature 'CompadreDB'
NumberMatrices(object)

### Arguments

- `x`, `object`  
  - A CompadreDB object
- `...`  
  - Additional arguments
- `n`  
  - The number of rows to extract
- `y`  
  - A data.frame to merge with `x`

### Value

No return value, called for side effects

---

**CompadreDB-Subsetting**  
*Subsetting CompadreDB objects*

### Description

CompadreDB objects can be subset just like a regular `data.frame`, using either `[` or `subset()`. Note, however, that the `mat` column will always be retained during subsetting, even if it is not included in the user’s column subset.
Usage

```r
## S4 method for signature 'CompadreDB,ANY,ANY,ANY'
x[i, j, ..., drop = FALSE]

## S3 method for class 'CompadreDB'
subset(x, subset, select, drop = FALSE, ...)
```

Arguments

- `x`: A CompadreDB object
- `i`: row indices (see `.data.frame`)
- `j`: column indices (see `.data.frame`)
- `...`: ignored
- `drop`: ignored
- `subset`: logical expression indicating which rows to keep
- `select`: expression indicating which columns to keep

Value

No return value, called for side effects

Examples

```r
# subset to the first 10 rows
Compadre[1:10,]

# subset to the species 'Echinacea angustifolia'
subset(Compadre, SpeciesAccepted == "Echinacea angustifolia")

# remove the column SurvivalIssue
Compadre[,names(Compadre) != "SurvivalIssue"]

# Not run:
# column selection doesn't include mat, but mat will still be returned with a
# along with a warning
subset(Compadre, select = c("SpeciesAccepted", "Authors"))

# End(Not run)
```

Description

CompadreDB methods for functions in `dplyr` and `ggplot2`. 
**Usage**

```r
fortify.CompadreDB(model, data, ...)
filter.CompadreDB(.data, ...)
slice.CompadreDB(.data, ...)
arrange.CompadreDB(.data, ...)
mutate.CompadreDB(.data, ...)
group_by.CompadreDB(.data, ..., add = FALSE)
ungroup.CompadreDB(x, ...)
summarize.CompadreDB(.data, ...)
summarise.CompadreDB(.data, ...)
select.CompadreDB(.data, ...)
rename.CompadreDB(.data, ...)
left_join.CompadreDB(
  x,
  y,
  by = NULL,
  copy = FALSE,
  suffix = c(".x", ".y"),
  ...
)
right_join.CompadreDB(
  x,
  y,
  by = NULL,
  copy = FALSE,
  suffix = c(".x", ".y"),
  ...
)
inner_join.CompadreDB(
  x,
  y,
  by = NULL,
  copy = FALSE,
  suffix = c(".x", ".y"),
  ...
)
```
full_join.CompadreDB(
    x,
    y,
    by = NULL,
    copy = FALSE,
    suffix = c(".x", ".y"),
    ...
)

Arguments

data see fortify
... other arguments
add Logical indicating whether to overwrite existing groups (FALSE) or add to any existing groups (TRUE)
x, model, .data A CompadreDB object
y see join
by see join
copy see join
suffix see join

Value

No return value, called for side effects

CompadreLegacy Subsample of a legacy version of the COMPADRE Plant Matrix Database for testing and examples

Description

CompadreLegacy is a subsample of the COMPADRE Plant Matrix Database in the legacy format (class 'list'), for use in testing and examples. For full documentation see the COMPADRE User Guide.

Usage

CompadreLegacy
Format

A list with four elements:

**metadata** - A data frame with the following 47 columns:

- SpeciesAuthor - Binomial species name given by the paper's author
- SpeciesAccepted - Accepted binomial species name taken from *The Plant List* or *Catalogue of Life*
- CommonName - Common name for species used in the publication
- Genus - Taxonomic genus that the accepted species belongs to
- Family - Family that the species belongs to
- Order - Order that the species belongs to
- Class - Class that the species belongs to
- Phylum - Phylum that the species belongs to
- Kingdom - Kingdom that the species belongs to
- OrganismType - Organism type (see COMPADRE User Guide for documentation)
- DicotMonoc - Whether the species is a dicot or monocot. Non-angiosperms are NA.
- AngioGymno - Whether the species is an angiosperm or gymnosperm. Non-plants are NA.
- Authors - Last name of all authors (separated with ";")
- Journal - Abbreviated journal title, or type of source document (e.g. "PhD thesis")
- YearPublication - Year of publication
- DOI.ISBN - Digital Object Identifier or International Standard Book Number codes to identify each publication
- AdditionalSource - Additional source(s) used to reconstruct the matrix or obtain additional metadata for the matrix (if applicable)
- StudyDuration - Number of years of observation in study (*StudyEnd* −*StudyStart*)
- StudyStart - Study start year
- StudyEnd - Study end year
- AnnualPeriodicity - Inverse of the length of the projection interval (in years)
- NumberPopulations - The number of study populations as defined by the authors. Within site replication of permanent plots is not defined as separate populations
- MatrixCriteriaSize - Indicates whether the matrix contains stages based on size. If so, indicates what that measure of size is
- MatrixCriteriaOntogeny - Indicates whether the matrix contains stages based on ontogenic/developmental stages
- MatrixCriteriaAge - Indicates whether the matrix contains stages based on age
- MatrixPopulation - Population name or definition of where the matrix was recorded, usually given by the author. See COMPADRE User Guide.
- Lat - Latitude in decimal degrees
- Lon - Longitude in decimal degrees
• Altitude - Altitude of study site (m above sea level)
• Country - 3-letter ISO country code for the country in which the study took place (multiple countries separated with ";")
• Continent - Continent on which study took place
• Ecoregion - Ecoregion in which study took place. See COMPADRE User Guide.
• StudiedSex - Whether study included only males ("M"), only females ("F"), or both sexes ("M/F")
• MatrixComposite - Indicates the type of matrix population model. Possible values are Individual, Mean, Pooled, and Seasonal. See COMPADRE User Guide.
• MatrixTreatment - Describes if a treatment was applied to the population or not. If yes, includes a brief description of the treatment. If not, Unmanipulated
• MatrixCaptivity - Whether species was studied in the wild (\textit{W}), captivity (\textit{C}), or captured from a wild population (\textit{CW})
• MatrixStartYear - First year of matrix
• MatrixStartSeason - First season of matrix as described by author (hemisphere-specific)
• MatrixStartMonth - First month of matrix
• MatrixEndYear - Final year of matrix
• MatrixEndSeason - Final season of matrix as described by author (hemisphere-specific)
• MatrixEndMonth - Final month of matrix
• MatrixSplit - Whether the matrix has been split into components \textit{U}, \textit{F}, and \textit{C} ("Divided") or not ("Indivisible"). If not, elements of matU, matF, and matC are filled with \textit{NA}
• MatrixFec - Whether fecundity was measured for the matrix model
• Observation - Additional observations recorded by database compilers
• survivalIssue - Denotes the maximum stage-specific survival value

\textbf{mat} - A list of population projection models, which are also in list format. Each list element contains four matrices:

• matA - A matrix population model
• matU - The survival- and growth-related component of matA
• matF - The sexual reproduction component of matA
• matC - The clonal reproduction component of matA

\textbf{matrixClass} - A list of data frames, each with the following columns:

• MatrixClassOrganized - Standardized stage class of the matrix population model
• MatrixClassAuthor - Stage description from the original publication
• MatrixClassNumber - Integer stage number

\textbf{version} - A list with the following elements:

• Version - The version number of the database
DateCreated - The date that the .RData file was created
NumberAcceptedSpecies - The number of accepted species in the original version
NumberStudies - The number of studies in the original version
NumberMatrices - The number of matrices in the original version
Agreement - Link to the COMADRE license agreement

Description
This page describes methods for accessing any matrix information from CompadreMat and CompadreDB objects.

Most methods for working with matrices are applicable to both CompadreMat and CompadreDB objects. These are described on this page (along with a couple) of methods applicable to only CompadreMat or CompadreDB objects.

Usage
matA(object)

## S4 method for signature 'CompadreMat'
matA(object)

## S4 method for signature 'CompadreDB'
matA(object)

## S4 method for signature 'list'
matA(object)

matU(object)

## S4 method for signature 'CompadreMat'
matU(object)

## S4 method for signature 'CompadreDB'
matU(object)

## S4 method for signature 'list'
matU(object)

matF(object)

## S4 method for signature 'CompadreMat'
matF(object)
## S4 method for signature 'CompadreDB'
matF(object)

## S4 method for signature 'list'
matF(object)

matC(object)

## S4 method for signature 'CompadreMat'
matC(object)

## S4 method for signature 'CompadreDB'
matC(object)

## S4 method for signature 'list'
matC(object)

matrixClass(object)

## S4 method for signature 'CompadreMat'
matrixClass(object)

## S4 method for signature 'CompadreDB'
matrixClass(object)

## S4 method for signature 'list'
matrixClass(object)

MatrixClassAuthor(object)

## S4 method for signature 'CompadreMat'
MatrixClassAuthor(object)

## S4 method for signature 'CompadreDB'
MatrixClassAuthor(object)

## S4 method for signature 'list'
MatrixClassAuthor(object)

MatrixClassOrganized(object)

## S4 method for signature 'CompadreMat'
MatrixClassOrganized(object)

## S4 method for signature 'CompadreDB'
MatrixClassOrganized(object)
## S4 method for signature 'list'
MatrixClassOrganized(object)

MatrixClassNumber(object)

## S4 method for signature 'CompadreMat'
MatrixClassNumber(object)

## S4 method for signature 'CompadreDB'
MatrixClassNumber(object)

## S4 method for signature 'list'
MatrixClassNumber(object)

### Arguments

- **object**: A CompadreDB object

### Slots

- **matA**: A matrix population model (i.e. a square projection matrix)
- **matU**: The survival component of a matrix population model (i.e. a square projection matrix reflecting survival-related transitions; e.g. progression, stasis, and retrogression)
- **matF**: The sexual component of a matrix population model (i.e. a square projection matrix reflecting transitions due to sexual reproduction)
- **matC**: The clonal component of a matrix population model (i.e. a square projection matrix reflecting transitions due to clonal reproduction)
- **matrixClass**: A data frame with columns MatrixClassOrganized (elements are "active", "prop", or "dorm"). MatrixClassAuthor (the matrix author's stage description), and MatrixClassNumber (integer stage number)

---

### Description

Calculates an element-wise mean over a list of matrices or CompadreMat objects of constant dimension.

### Usage

- `mat_mean(x, na.rm = FALSE)`
- `mpm_mean(x, na.rm = FALSE)`
Arguments

- **x**: List of matrices (mat_mean) or list of CompadreMat objects (mpm_mean), all of the same dimension.
- **na.rm**: Logical indicating whether missing values should be excluded (see Details). Defaults to FALSE.

Details

If `na.rm` == TRUE, missing values are ignored in the calculation of the mean matrix. If `na.rm` == TRUE and a given element is NA in every matrix within x, the value returned for that element will be 0.

Value

A matrix (mat_mean) or a CompadreMat object (mpm_mean).

Author(s)

Patrick Barks <patrick.barks@gmail.com>
Owen R. Jones <jones@biology.sdu.dk>

See Also

Other data management: cdb_flatten(), cdb_id_stages(), cdb_id_studies(), cdb_id(), cdb_mean_matF(), cdb_rbind(), cdb_unflatten(), cdb_unnest(), string_representation

Examples

```r
# there are four rows for species 'Haplopappus_radiatus' in Compadre
mpms <- Compadre_mat[Compadre$SpeciesAuthor == "Haplopappus_radiatus"]
mpm_mean(mpms)

# extract list of matA and take mean
mats <- matA(mpms)
mat_mean(mats)
```

### Description

Methods for extracting stage-class information from CompadreMat or CompadreDB objects, including whether the matrix population model includes one or more propagule stages (`mpm_has_prop`), dormant stages (`mpm_has_dorm`), or active stages (`mpm_has_active`), and the integer index of the first active stage class (`mpm_first_active`).

These methods will return a single value if passed a CompadreMat object, or a vector of values if passed a CompadreDB object (one value for every CompadreMat object within the column 'mat').
Usage

mpm_has_prop(object)

## S4 method for signature 'CompadreMat'
mpm_has_prop(object)

## S4 method for signature 'CompadreDB'
mpm_has_prop(object)

mpm_has_active(object)

## S4 method for signature 'CompadreMat'
mpm_has_active(object)

## S4 method for signature 'CompadreDB'
mpm_has_active(object)

mpm_has_dorm(object)

## S4 method for signature 'CompadreMat'
mpm_has_dorm(object)

## S4 method for signature 'CompadreDB'
mpm_has_dorm(object)

mpm_first_active(object)

## S4 method for signature 'CompadreMat'
mpm_first_active(object)

## S4 method for signature 'CompadreDB'
mpm_first_active(object)

Arguments

object A CompadreMat or CompadreDB object

Value

No return value, called for side effects

Author(s)

Patrick Barks <patrick.barks@gmail.com>

See Also

Other data checking: cdb_check_species(), cdbCollapse(), cdb_compare(), cdb_flag()
Examples

```r
# with CompadreMat object
mpm_has_prop(Compadre$mat[[1]])
mpm_has_active(Compadre$mat[[1]])
mpm_has_dorm(Compadre$mat[[1]])
mpm_first_active(Compadre$mat[[1]])

# with CompadreDB object
mpm_has_prop(Compadre)
mpm_has_active(Compadre)
mpm_has_dorm(Compadre)
mpm_first_active(Compadre)
```

string_representation

Convert vectors or square numeric matrices to and from string representation

Description

Functions to convert vectors or square numeric matrices to and from string representation, which is primarily useful for writing data frames with list-columns containing vectors or matrices to a flat file format such as csv.

String representations of vectors and matrices begin with an open bracket ("[") and end with a closed bracket ("]"). Matrix elements are separated with a space ("[0.2 0.3 0.1 0]") whereas vector elements are separate with two vertical bars ("[Seedling||Juvenile||Reproductive]").

Usage

```r
mat_to_string(mat)
vec_to_string(vec)
string_to_mat(mat_str)
string_to_vec(vec_str, numeric = FALSE)
```

Arguments

- `mat`: A square numeric matrix
- `vec`: A vector
- `mat_str`: A square numeric matrix in string representation
- `vec_str`: A vector in string representation
- `numeric`: Logical value indicating whether a string representation of a vector should be coerced to numeric (if FALSE remains character)
**Value**

A square numeric matrix (string_to_mat), vector (string_to_vec), or string (mat_to_string or vec_to_string).

**Author(s)**

Owen R. Jones <jones@biology.sdu.dk>
Patrick M. Barks <patrick.barks@gmail.com>

**See Also**

cdb_flatten cdb_unflatten

Other data management: cdb_flatten(), cdb_id_stages(), cdb_id_studies(), cdb_id(), cdb_mean_matF(), cdb_rbind(), cdb_unflatten(), cdb_unnest(), mpm_mean()

**Examples**

```r
mat_str <- "[3.3 5.2 6.1 0.1 NA 0.3 0.2 0.4 0.1]"
mat <- string_to_mat(mat_str)

vec1_str <- "[0.30||0.42||0.19||0.09]"
vec1 <- string_to_vec(vec1_str, numeric = TRUE)

vec2_str <- "[Seedling 1||Seedling 2||Juvenile||Reproductive]"
vec2 <- string_to_vec(vec2_str)

# convert back to string format
mat_to_string(mat)
vec_to_string(vec1)
vec_to_string(vec2)
```

```r
## Not run:
# non-square matrix
mat_str <- "[0.42 0.52 0.15 0.23 0.14]"
string_to_mat(mat_str)

## End(Not run)
```
Index

* data acquisition
  cdb_fetch, 7
  cdb_metadata, 14

* data checking
  cdb_check_species, 3
  cdb_collapse, 4
  cdb_compare, 6
  cdb_flag, 8
  mpm_methods, 32

* data management
  cdb_flatten, 9
  cdb_id, 10
  cdb_id_stages, 11
  cdb_id_studies, 12
  cdb_mean_matF, 13
  cdb_rbind, 15
  cdb_unflatten, 16
  cdb_unnest, 17
  mpm_mean, 31
  string_representation, 34

* datasets
  Compadre, 18
  CompadreLegacy, 26
  [,CompadreDB,ANY,ANY,ANY-method (CompadreDB-Subsetting), 23
  [.data.frame, 24
  [[,CompadreDB,ANY,missing-method (CompadreDB), 20
  [[,data.frame, 21
  [[<-CompadreDB,ANY,missing-method (CompadreDB), 20
  $,CompadreDB-method (CompadreDB), 20
  $<-,CompadreDB-method (CompadreDB), 20
  as_tibble.CompadreDB (CompadreDB-Methods), 22
  cdb_check_species, 3, 5, 6, 9, 33
  cdb_collapse, 4, 4, 6, 9, 33
  cdb_compare, 4, 5, 6, 9, 33
  cdb_fetch, 7, 15
  cdb_flag, 4–6, 8, 33
  cdb_flatten, 9, 11–14, 16, 18, 32, 35
  cdb_id, 10, 10, 12–14, 16, 18, 32, 35
  cdb_id_stages, 4, 5, 10, 11, 11, 13, 14, 16, 18, 32, 35
  cdb_id_studies, 10–12, 12, 14, 16, 18, 32, 35
  cdb_mean_matF, 10–13, 13, 16, 18, 32, 35
  cdb_metadata, 7, 14
  cdb_rbind, 10–14, 15, 16, 18, 32, 35
  cdb_unflatten, 10–14, 16, 16, 18, 32, 35
  cdb_unnest, 10–14, 16, 17, 32, 35
  Comadre (Compadre), 18
  Compadre, 18
  CompadreData (CompadreDB), 20
  CompadreData,CompadreDB-method (CompadreDB), 20
  CompadreDB, 20
  CompadreDB-Methods, 22
  CompadreDB-Subsetting, 23
  CompadreDB-Tidyverse, 24
  CompadreLegacy, 26
  CompadreMatrixMethods, 29
  DateCreated (CompadreDB), 20
  DateCreated,CompadreDB-method (CompadreDB), 20
  dim.CompadreDB (CompadreDB-Methods), 22
  dplyr, 24
  filter.CompadreDB (CompadreDB-Tidyverse), 24
  fortiy, 26
fortify.CompadreDB
(CompadreDB-Tidyverse), 24
full_join.CompadreDB
(CompadreDB-Tidyverse), 24
ggplot2, 24
group_by.CompadreDB
(CompadreDB-Tidyverse), 24
head.CompadreDB
(CompadreDB-Methods), 22
inner_join.CompadreDB
(CompadreDB-Tidyverse), 24
isErgodic, 8
isIrreducible, 8
isPrimitive, 8
join, 26
left_join.CompadreDB
(CompadreDB-Tidyverse), 24
mat_mean (mpm_mean), 31
mat_to_string (string_representation), 34
matA (CompadreMatrixMethods), 29
matA,CompadreDB=method
(CompadreMatrixMethods), 29
matA,CompadreMat=method
(CompadreMatrixMethods), 29
matA,list-method
(CompadreMatrixMethods), 29
matC (CompadreMatrixMethods), 29
matC,CompadreDB=method
(CompadreMatrixMethods), 29
matC,CompadreMat=method
(CompadreMatrixMethods), 29
matC,list-method
(CompadreMatrixMethods), 29
matF (CompadreMatrixMethods), 29
matF,CompadreDB=method
(CompadreMatrixMethods), 29
matF,CompadreMat=method
(CompadreMatrixMethods), 29
matF,list-method
(CompadreMatrixMethods), 29
matrixClass (CompadreMatrixMethods), 29
matrixClass,CompadreDB=method
(CompadreMatrixMethods), 29
matrixClass,CompadreMat=method
(CompadreMatrixMethods), 29
matrixClass,list-method
(CompadreMatrixMethods), 29
MatrixClassAuthor
(CompadreMatrixMethods), 29
MatrixClassAuthor,CompadreDB=method
(CompadreMatrixMethods), 29
MatrixClassAuthor,CompadreMat=method
(CompadreMatrixMethods), 29
MatrixClassAuthor,list-method
(CompadreMatrixMethods), 29
MatrixClassNumber
(CompadreMatrixMethods), 29
MatrixClassNumber,CompadreDB=method
(CompadreMatrixMethods), 29
MatrixClassNumber,CompadreMat=method
(CompadreMatrixMethods), 29
MatrixClassNumber,list-method
(CompadreMatrixMethods), 29
MatrixClassOrganized
(CompadreMatrixMethods), 29
MatrixClassOrganized,CompadreDB=method
(CompadreMatrixMethods), 29
MatrixClassOrganized,CompadreMat=method
(CompadreMatrixMethods), 29
MatrixClassOrganized,list-method
(CompadreMatrixMethods), 29
matU (CompadreMatrixMethods), 29
matU,CompadreDB=method
(CompadreMatrixMethods), 29
matU,CompadreMat=method
(CompadreMatrixMethods), 29
matU,list-method
(CompadreMatrixMethods), 29
merge.CompadreDB
(CompadreDB-Methods), 22
mpm_first_active (mpm_methods), 32
mpm_first_active,CompadreDB=method
(mpm_methods), 32
mpm_first_active,CompadreMat=method
(mpm_methods), 32
mpm_first_active,CompadreDB=method
(mpm_methods), 32
mpm_has_active (mpm_methods), 32
mpm_has_active,CompadreDB=method
(mpm_methods), 32
mpm_has_active,CompadreMat=method
(mpm_methods), 32
mpm_has_dorm (mpm_methods), 32
mpm_has_dorm, CompadreDB-method (mpm_methods), 32
mpm_has_dorm, CompadreMat-method (mpm_methods), 32
mpm_has_prop (mpm_methods), 32
mpm_has_prop, CompadreDB-method (mpm_methods), 32
mpm_has_prop, CompadreMat-method (mpm_methods), 32
mpm_mean, 5, 10–14, 16, 18, 31, 35
mpm_methods, 4–6, 9, 32
mutate.CompadreDB (CompadreDB-Tidyverse), 24
names.CompadreDB (CompadreDB-Methods), 22
NumberAcceptedSpecies (CompadreDB-Methods), 22
NumberAcceptedSpecies, CompadreDB-method (CompadreDB-Methods), 22
NumberMatrices (CompadreDB-Methods), 22
NumberMatrices, CompadreDB-method (CompadreDB-Methods), 22
NumberStudies (CompadreDB-Methods), 22
NumberStudies, CompadreDB-method (CompadreDB-Methods), 22
rename.CompadreDB (CompadreDB-Tidyverse), 24
right_join.CompadreDB (CompadreDB-Tidyverse), 24
select.CompadreDB (CompadreDB-Tidyverse), 24
slice.CompadreDB (CompadreDB-Tidyverse), 24
string_representation, 9–14, 16, 18, 32, 34
string_to_mat (string_representation), 34
string_to_vec (string_representation), 34
subset.CompadreDB (CompadreDB-Subsetting), 23
summarise.CompadreDB (CompadreDB-Tidyverse), 24
summarize.CompadreDB (CompadreDB-Tidyverse), 24
tail.CompadreDB (CompadreDB-Methods), 22