Package: mpmaggregate (via r-universe)

May 24, 2026

Type Package

Title Aggregate Matrix Population Models

Version 0.2.6

Date 2026-05-23

Author Richard A. Hinrichsen [aut, cre] (<`https://orcid.org/0000-0003-0761-3005`>), Roberto Salguero-Gomez [aut] (<`https://orcid.org/0000-0002-6085-4433`>), Hiroyuki Yokomizo [aut] (<`https://orcid.org/0000-0001-9814-8465`>)

Maintainer Richard A. Hinrichsen <`r.hinrichsen.ecology@gmail.com`>

Description Aggregates matrix population models (MPMs) in both the lambda (stable growth rate) and R0 (net reproductive rate) frameworks, including standard and elasticity-consistent aggregators. Standard aggregation in the lambda framework maintains consistent lambda and stable stage distribution, while standard aggregation in the R0 framework maintains consistent R0 and cohort stable stage distribution. Elasticity-consistent aggregators maintain these same consistencies with respect to the chosen framework and additionally preserve consistent reproductive values in the lambda framework and cohort reproductive values in the R0 framework. Aggregation can take the form of general-to-general MPM (mpm_aggregate) or Leslie-to-Leslie MPM (leslie_aggregate).

License MIT + file LICENSE

Encoding UTF-8

Imports expm

Suggests testthat (>= 3.0.0), roxygen2, knitr, rmarkdown, Rcompadre, kableExtra, rphylopic, collidr, png

Config/testthat/edition 3

VignetteBuilder knitr

Depends R (>= 3.5)

LazyData true

Config/roxygen2/version 8.0.0

Repository https://hinrich62.r-universe.dev

Date/Publication 2026-05-24 01:47:48 UTC

RemoteUrl https://github.com/hinrich62/mpmaggregate

RemoteRef HEAD

RemoteSha 04661045b125c53f473f28d7eca97c717f5f616e

Contents

`mpmaggregate-package` *mpmaggregate: Aggregation of matrix population models*

Description

mpmaggregate aggregates matrix population models (MPMs) under both the λ (stable population growth rate) and R_0 (net reproductive rate) frameworks, using either standard or elasticity-consistent aggregation.

Details

The package provides methods for aggregating matrix population models while preserving key demographic quantities under either the λ or R_0 framework. Aggregation can be performed using either standard or elasticity-consistent criteria, depending on whether reproductive values are to be preserved in addition to growth rates.

- **framework = "lambda"**: aggregation targets the dominant eigenvalue λ and its associated eigenvectors.
- **framework = "R0"**: aggregation targets the dominant eigenvalue R_0 of a next-generation matrix derived from reproductive and survival components.
- **criterion = "standard"**: preserves λ and the stable stage distribution (λ framework) or R_0 and the cohort stable stage distribution (R_0 framework).
- **criterion = "elasticity"**: additionally preserves reproductive values (λ) or cohort reproductive values (R_0 framework).

Aggregation is implemented using effective left and right operators, P_{eff} and Q_{eff}, which map an original matrix M (dimension n) to an aggregated matrix M_{agg} (dimension m):

$$M_{\mathrm{agg}} = P_{\mathrm{eff}}\, M\, Q_{\mathrm{eff}}.$$

These operators are constructed from the partition matrix P and weights that depend on the selected framework and criterion.

Aggregation can be performed in two primary settings:

- General-to-general matrix population model aggregation via `mpm_aggregate`. Groupings are supplied by the user as a list of stage indices (see `mpm_partition`).
- Leslie-to-Leslie matrix population model aggregation via `leslie_aggregate`, which aggregates age classes into cohort blocks and may apply a disaggregation step when required.

Effectiveness of aggregation measures the quality of an aggregation by how closely the aggregated model reproduces the behavior of the original model under the chosen framework and criterion.

The package assumes that inputs are nonnegative matrices and (when required for computing Perron eigenvectors used in aggregation) that the population projection matrix is irreducible.

Main functions

- `mpm_aggregate`: general-to-general MPM aggregation using user-defined groups of stages.
- `leslie_aggregate`: Leslie-to-Leslie MPM aggregation into `ngroups` contiguous age classes.
- `generation_time`: return the generation time in either the lambda framework or R0 (cohort-based) framework.
- `mpm_elasticity`: compute elasticities of either the dominant eigenvalue λ (λ framework) or the net reproductive rate R_0 (R_0 framework) with respect to the entries of the population projection matrix `matA`.

Utility functions

- `mpm_partition`: convert a user-supplied grouping list into a partitioning matrix.
- `spectral_radius`: compute the spectral radius (dominant eigenvalue modulus) of a square matrix.
- `dominant_eigen`: return the dominant eigenvalue and its associated stable stage distribution and reproductive values for a population projection matrix.
- `leslie_dominant_eigen`: return the dominant eigenvalue and its associated stable age distribution and reproductive values for a Leslie population projection matrix.

- `is_leslie`: check whether the input matrix is a Leslie population projection matrix.
- `leslie_disaggregate`: disaggregate a Leslie population projection model so that its dimensionality is compatible with that of the aggregated model.
- `stable_stage`: return the stable stage distribution (w) associated with the dominant eigenvalue, scaled so that `sum(w) = 1`.
- `reproductive_values`: return the reproductive values (v) associated with the dominant eigenvalue, scaled so that `sum(v * w) = 1`.
- `leslie_stable_age`: return the stable age distribution (w) associated with the dominant eigenvalue of a Leslie matrix, scaled so that `sum(w) = 1`.
- `leslie_reproductive_values`: return the reproductive values (v) associated with the dominant eigenvalue of a Leslie matrix, scaled so that `sum(v * w) = 1`.

Author(s)

Maintainer: Richard A. Hinrichsen `<r.hinrichsen.ecology@gmail.com>` (ORCID)

Authors:

- Richard A. Hinrichsen `<r.hinrichsen.ecology@gmail.com>` (ORCID)
- Roberto Salguero-Gomez `<rob.salguero@zoo.ox.ac.uk>` (ORCID)
- Hiroyuki Yokomizo `<hiroyuki.yokomizo@nies.go.jp>` (ORCID)

References

Bienvenu, F., Akçay, E., Legendre, S. & McCandlish, D.M. (2017). The genealogical decomposition of a matrix population model with applications to the aggregation of stages. *Theoretical Population Biology*, 115, 69-80. doi:10.1016/j.tpb.2017.04.002

Bienvenu, F. & Legendre, S. (2015). A new approach to the generation time in matrix population models. *The American Naturalist*, 185(6), 834-843. doi:10.1086/681104

Caswell, H. (2001). *Matrix population models: construction, analysis and interpretation* (2nd ed.). Sinauer.

Hinrichsen, R. A. (2023). Aggregation of Leslie matrix models with application to ten diverse animal species. *Population Ecology*, 65(3), 146-166. doi:10.1002/1438390X.12149

Hinrichsen, R. A., Yokomizo, H., & Salguero-Gómez, R. (2026). From theory to application: Elasticity-consistent aggregation of Leslie matrix population models for comparative demography. *bioRxiv*, preprint. doi:10.64898/2026.02.04.703802

Salguero-Gómez, R., & Gamelon, M., eds. (2021). *Demographic methods across the tree of life*. Oxford University Press.

Salguero-Gómez, R. & Plotkin, J. B. (2010). Matrix dimensions bias demographic inferences: implications for comparative plant demography. *The American Naturalist*, 176, 710-722. doi:10.1086/657044

See Also

`mpm_aggregate`, `leslie_aggregate`

`dominant_eigen` *Dominant eigen-elements of a population projection matrix*

Description

Returns the dominant eigenvalue and its associated right/left eigenvectors (stable stage distribution `w` and reproductive values `v`) for a square population projection matrix `A`.

Usage

```
dominant_eigen(A, tol = 1e-12, ensure_positive = TRUE)
```

Arguments

`A`	A square numeric matrix.
`tol`	Numeric tolerance passed to `stable_stage` and `reproductive_values`.
`ensure_positive`	Logical. Passed to `stable_stage` and `reproductive_values`. If `TRUE`, attempts to orient eigenvectors so entries are mostly positive by flipping sign if needed.

Details

The output is normalized using the conventions in `stable_stage` and `reproductive_values`:

- `w` is scaled so that `sum(w) = 1`
- `v` is scaled so that `sum(v * w) = 1`

Value

A named list with elements:

- `lambda`: spectral radius of `A`
- `w`: stable stage distribution (right eigenvector), `sum(w)=1`
- `v`: reproductive values (left eigenvector), `sum(v*w)=1`

References

Caswell, H. (2001). *Matrix population models: construction, analysis and interpretation* (2nd ed.). Sinauer.

See Also

`spectral_radius`, `stable_stage`, `reproductive_values`

Examples

```
A <- matrix(c(
  0,   1,   2,
  0.5, 0,   0,
  0,   0.8, 0.9
), nrow = 3, byrow = TRUE)

dom <- dominant_eigen(A)
dom$lambda
sum(dom$w)
sum(dom$v * dom$w)
```

`generation_time` *Generation time from a matrix population model*

Description

Returns a generation time for either the λ or R_0 framework.

Usage

```
generation_time(
  matF,
  matU,
  matC = NULL,
  framework = c("lambda", "R0"),
  tol = 1e-12
)
```

Arguments

`matF`	A square fecundity/sexual reproduction matrix (numeric matrix). Must not be `NULL`.
`matU`	A square survival/growth/transition matrix (numeric matrix). Must not be `NULL`.
`matC`	Optional square clonal reproduction matrix (numeric matrix), same dimensions as `matF` and `matU`. Default `NULL`.
`framework`	Character scalar. Must be either `"lambda"` or `"R0"`.
`tol`	Numeric tolerance used for nonnegativity checks and near-zero screening.

Details

The full projection matrix `matA` is constructed internally as `matA` = `matU` + `matF` (+ `matC`) and is not accepted as an input because `matA` alone is not sufficient to define the required decomposition.

Eigen-elements (e.g., λ, R_0, stable stage distribution, and reproductive values) are obtained via the function `dominant_eigen`.

If `matC` is supplied, the reproductive matrix is defined as `matR` = `matF` + `matC`. Otherwise, `matR` = `matF`.

This function performs input validation, constructs `matA` and `matR`, and sets up the framework-specific branch for computing generation time.

Value

A list with elements:

- `framework`: the chosen framework (`"lambda"` or `"R0"`).
- `generation_time`: generation time (numeric scalar).

References

Bienvenu, F. & Legendre, S. (2015). A new approach to the generation time in matrix population models. *The American Naturalist*, 185(6), 834-843. doi:10.1086/681104

Ellner, S. P. (2018). Generation time in structured populations. *The American Naturalist*, 192(1), 105-110. doi:10.1086/697539

See Also

`dominant_eigen`, `spectral_radius`.

Examples

```
matU <- matrix(c(0.2, 0.0,
                 0.3, 0.4), nrow = 2, byrow = TRUE)
matF <- matrix(c(0.0, 1.2,
                 0.0, 0.0), nrow = 2, byrow = TRUE)
generation_time(matF = matF, matU = matU, framework = "lambda")
generation_time(matF = matF, matU = matU, framework = "R0")
```

`is_leslie` *Test whether a matrix is a Leslie matrix*

Description

Checks whether a matrix `A` has Leslie form: nonzero entries are allowed only in the first row (fertility rates) and on the subdiagonal (survival probabilities). Survival probabilities must be in $(0, 1]$ (within tolerance), fertility rates must be nonnegative, and at least one fertility rate must be positive.

Usage

```
is_leslie(A, tol = sqrt(.Machine$double.eps))
```

Arguments

`A`	A numeric square matrix. A length-1 positive numeric will be coerced to a 1x1 matrix.
`tol`	Numeric tolerance used for structural comparisons and bounds checks.

Details

A 1x1 matrix with a positive entry is treated as a valid (degenerate) Leslie matrix.

Value

Logical; `TRUE` if `A` is a Leslie matrix, otherwise `FALSE`.

Examples

```
L <- matrix(c(
  0, 2, 1,
  0.6, 0, 0,
  0, 0.7, 0
), nrow = 3, byrow = TRUE)
is_leslie(L)
```

`leslie_aggregate` *Aggregate Leslie-to-Leslie matrix population model*

Description

Aggregates a Leslie matrix into a smaller Leslie matrix with `ngroups` contiguous age classes.

Usage

```
leslie_aggregate(
  matA,
  ngroups,
  framework = c("lambda", "R0"),
  criterion = c("standard", "elasticity"),
  tol = 1e-12,
  ...
)
```

Arguments

`matA`	A square numeric matrix. Must be finite, nonnegative, and satisfy the structural constraints of a Leslie matrix (checked internally).
`ngroups`	Positive integer giving the number of aggregated age groups (the dimension m of the aggregated model). Must satisfy `ngroups <= n`.

`framework`	Character scalar; either `"lambda"` or `"R0"`. Determines whether aggregation is based on the projection matrix A or an R_0-based reference matrix built from the implied U and reproductive component of the Leslie matrix.
`criterion`	Character scalar; either `"standard"` for standard aggregation or `"elasticity"` for elasticity-consistent aggregation.
`tol`	Numeric tolerance used in positivity/zero-mass checks and balancing calculations for the elasticity-consistent case.
`...`	Reserved for future use.

Details

This function assumes the input is a valid Leslie matrix and checks the Leslie structure internally. If the original dimensionality n is not an integer multiple of m, a disaggregation step is applied internally so that the expanded dimensionality is divisible by m before aggregation proceeds.

Aggregation can be performed under two frameworks (`"lambda"` or `"R0"`) and two criteria (`"standard"` or `"elasticity"`), which determine how age classes are weighted during aggregation.

Irreducibility requirement: The input Leslie matrix `matA` must be irreducible. This function checks irreducibility of `matA` and will fail if it is reducible. (Any internal disaggregation preserves irreducibility.)

The Leslie matrix is internally decomposed into a reproductive component R (the first row) and a survival component U (the subdiagonal and any survival terms implied by the Leslie structure). For `framework` = `"R0"`, the next-generation matrix $K = R(I - U)^{-1}$ is formed and its dominant eigenvalue is used as R_0.

Aggregation proceeds on a k-step matrix where $k = n/m$, with n the dimension of the input Leslie matrix and $m =$ `ngroups`. The function returns an $m \times m$ k-step Leslie matrix (stored as `matAk_agg`) describing transitions among the aggregated age classes.

Effectiveness measures how closely the aggregated model reproduces the behavior of the original model under the chosen `framework` and `criterion`. When `effectiveness` is high (close to 1), applying the original model and then aggregating gives nearly the same result as aggregating first and then applying the aggregated model.

The returned element `effectiveness` is computed for both criteria:

- For `criterion` = `"standard"`, effectiveness is computed using weights based on the reference stable age distribution `w`.
- For `criterion` = `"elasticity"`, effectiveness is computed after balancing transformations (with weights based on `w * v`).

Value

A named list with elements:

`framework`	The matched framework used (`"lambda"` or `"R0"`).
`criterion`	The matched criterion used (`"standard"` or `"elasticity"`).
`matAk_agg`	Aggregated k-step Leslie matrix ($m \times m$).
`effectiveness`	A numeric effectiveness measure for the aggregation (definition depends on `criterion`).

References

Hinrichsen, R. A. (2023). Aggregation of Leslie matrix models with application to ten diverse animal species. *Population Ecology*, 65(3), 146-166. doi:10.1002/1438390X.12149

Hinrichsen, R. A., Yokomizo, H., & Salguero-Gómez, R. (2026). From theory to application: Elasticity-consistent aggregation of Leslie matrix population models for comparative demography. *bioRxiv*, preprint. doi:10.64898/2026.02.04.703802

Examples

```
# A simple 3x3 Leslie matrix (fertility rates in first row; survival probabilities on subdiagonal)
A <- matrix(c(
  0.0, 1.2, 1.8,
  0.5, 0.0, 0.0,
  0.0, 0.7, 0.0
), nrow = 3, byrow = TRUE)

# Aggregate to 2 age groups
res_std <- leslie_aggregate(
  matA = A,
  ngroups = 2,
  framework = "lambda",
  criterion = "standard"
)
res_std$matAk_agg
res_std$effectiveness

res_el <- leslie_aggregate(
  matA = A,
  ngroups = 2,
  framework = "lambda",
  criterion = "elasticity"
)
res_el$matAk_agg
res_el$effectiveness
```

`leslie_disaggregate` *Disaggregate a Leslie matrix population model to a compatible dimensionality*

Description

Expands an $n \times n$ Leslie matrix `A` to an $n_{new} \times n_{new}$ Leslie matrix, where $n_{new} = \text{LCM}(n, m)$.

Usage

```
leslie_disaggregate(A, m)
```

Arguments

`A`	A Leslie matrix (checked with `is_leslie`). A length-1 positive numeric is treated as a 1x1 Leslie matrix.
`m`	Target aggregated dimensionality (positive integer).

Details

Disaggregation is required when aggregating a Leslie model to dimensionality `m` whenever `m` does not divide `n` evenly. The expanded matrix introduces a finer age structure that is compatible with both dimensionalities.

Each original age class is subdivided into n_{new}/n sub-classes. Fertility rates from original age class i are placed at the end of the corresponding block, deterministic aging within blocks is represented by ones on the subdiagonal, and original survival probabilities are inserted at block boundaries.

Value

A Leslie matrix of dimensionality $n_{new} \times n_{new}$, where $n_{new} = \mathrm{LCM}(n, m)$.

References

Hinrichsen, R. A. (2023). Aggregation of Leslie matrix models with application to ten diverse animal species. *Population Ecology*, 65(3), 146-166. doi:10.1002/1438390X.12149

Hinrichsen, R. A., Yokomizo, H., & Salguero-Gómez, R. (2026). From theory to application: Elasticity-consistent aggregation of Leslie matrix population models for comparative demography. *bioRxiv*, preprint. doi:10.64898/2026.02.04.703802

Examples

```
L <- matrix(c(
  0, 2, 1,
  0.6, 0, 0,
  0, 0.7, 0
), nrow = 3, byrow = TRUE)
leslie_disaggregate(L, m = 2)
```

`leslie_dominant_eigen` *Dominant eigen-elements of a Leslie matrix*

Description

Returns the dominant eigenvalue (λ) and the stable age distribution `w` and reproductive values `v` for a Leslie matrix `A`.

Usage

```
leslie_dominant_eigen(A)
```

Arguments

`A`	A Leslie matrix (checked with `is_leslie`). A length-1 positive numeric will be coerced to a 1x1 matrix.

Details

Normalization: The output is normalized using the conventions in `leslie_stable_age` and `leslie_reproductive_values`

- `w` is scaled so that `sum(w) = 1`
- `v` is scaled so that `sum(v * w) = 1`

Value

A named list with elements:

- `lambda`: spectral radius of `A`
- `w`: stable age distribution, `sum(w)=1`
- `v`: reproductive values, `sum(v*w)=1`

References

Caswell, H. (2001). *Matrix population models: construction, analysis and interpretation* (2nd ed.). Sinauer.

Demetrius, L. (1974). Demographic parameters and natural selection. *Proceedings of the National Academy of Sciences*, 71(12), 4645-4647. doi:10.1073/pnas.71.12.4645

See Also

`spectral_radius`, `leslie_stable_age`, `leslie_reproductive_values`

Examples

```
leslie_dominant_eigen(1)

L <- matrix(c(
  0, 2, 1,
  0.6, 0, 0,
  0, 0.7, 0
), nrow = 3, byrow = TRUE)
dom <- leslie_dominant_eigen(L)
dom$lambda
sum(dom$w)
sum(dom$v * dom$w)
```

`leslie_reproductive_values` *Reproductive values for a Leslie matrix*

Description

Computes the reproductive value vector `v` for a Leslie matrix `A`, scaled so that `sum(v * w) = 1`, where `w` is the stable age distribution returned by `leslie_stable_age`.

Usage

```
leslie_reproductive_values(A)
```

Arguments

`A`	A Leslie matrix (checked with `is_leslie`). A length-1 positive numeric will be coerced to a 1x1 matrix.

Details

This implementation follows Demetrius (1974) equation expressed in terms of the stable age distribution and fertility rates.

A length-1 positive numeric is treated as a 1x1 Leslie matrix and returns `1`.

Value

A numeric vector of length `nrow(A)` giving reproductive values, normalized so that `sum(v * w) = 1`.

References

Caswell, H. (2001). *Matrix population models: construction, analysis and interpretation* (2nd ed.). Sinauer.

Demetrius, L. (1974). Demographic parameters and natural selection. *Proceedings of the National Academy of Sciences*, 71(12), 4645-4647. doi:10.1073/pnas.71.12.4645

Examples

```
leslie_reproductive_values(1)

L <- matrix(c(
  0, 2, 1,
  0.6, 0, 0,
  0, 0.7, 0
), nrow = 3, byrow = TRUE)
v <- leslie_reproductive_values(L)
w <- leslie_stable_age(L)
sum(v * w)  # should be 1
```

`leslie_stable_age` *Stable age distribution for a Leslie matrix*

Description

Computes the stable age distribution for a Leslie matrix `A` using the standard recursion based on subdiagonal survival probabilities and the dominant eigenvalue. The output is scaled so that `sum(w) = 1`.

Usage

```
leslie_stable_age(A)
```

Arguments

`A`	A Leslie matrix (checked with `is_leslie`). A length-1 positive numeric will be coerced to a 1x1 matrix.

Details

A length-1 positive numeric is treated as a 1x1 Leslie matrix and returns `1`.

Value

A numeric vector of length `nrow(A)` giving the stable age distribution, scaled so that `sum(w) = 1`.

References

Caswell, H. (2001). *Matrix population models: construction, analysis and interpretation* (2nd ed.). Sinauer.

Demetrius, L. (1974). Demographic parameters and natural selection. *Proceedings of the National Academy of Sciences*, 71(12), 4645-4647. doi:10.1073/pnas.71.12.4645

Examples

```
leslie_stable_age(1)

L <- matrix(c(
  0, 2, 1,
  0.6, 0, 0,
  0, 0.7, 0
), nrow = 3, byrow = TRUE)
leslie_stable_age(L)
```

`matA_elephant1` *Coarse-scale Asian elephant projection matrix*

Description

A female-only projection matrix for an Asian elephant population, included with the package for examples and vignettes. The matrix has dimension 12 × 12 and a projection interval of 5 years.

Usage

```
matA_elephant1
```

Format

A numeric matrix with 12 rows and 12 columns.

Details

Population/model ID: 249273 — a coarser model with a 5-year projection interval for the Nagarhole National Park elephant population in India (Chelliah et al., 2013).

The matrix was originally retrieved from the COMADRE Animal Matrix Database using the ‘Rcompadre‘ package and bundled with this package so examples and vignettes can run without requiring internet access.

Source

COMADRE Animal Matrix Database, MatrixID 249273.

References

Chelliah, K., Bukka, H., & Sukumar, R. (2013). Modeling harvest rates and numbers from age and sex ratios: a demonstration for elephant populations. *Biological Conservation*, 165, 54–61. doi:10.1016/j.biocon.2013.05.008

`matA_elephant2` *Fine-scale Asian elephant projection matrix*

Description

A female-only projection matrix for an Asian elephant population, included with the package for examples and vignettes. The matrix has dimension 60 × 60 and a projection interval of 1 year.

Usage

```
matA_elephant2
```

Format

A numeric matrix with 60 rows and 60 columns.

Details

Population/model ID: 249274 — a fine-grained model with a 1-year projection interval for the Periyar Reserve elephant population in India (Goswami et al., 2014).

The matrix was originally retrieved from the COMADRE Animal Matrix Database using the ‘Rcompadre‘ package and bundled with this package so examples and vignettes can run without requiring internet access. The final entry was zeroed out so that the matrix is a true Leslie matrix.

Source

COMADRE Animal Matrix Database, MatrixID 249274.

References

Goswami, V. R., Vasudev, D., & Oli, M. K. (2014). The importance of conflict-induced mortality for conservation planning in areas of human–elephant co-occurrence. *Biological Conservation*, 176, 191–198. doi:10.1016/j.biocon.2014.05.026

`mpm_aggregate` *Aggregate a general-to-general matrix population model*

Description

Aggregates one or more matrix population model (MPM) components from an $n \times n$ model to an $m \times m$ model by combining stages into user-defined groups.

Usage

```
mpm_aggregate(
  matA = NULL,
  matU = NULL,
  matF = NULL,
  matC = NULL,
  groups,
  framework = c("lambda", "R0"),
  criterion = c("standard", "elasticity"),
  tol = 1e-12,
  ...
)
```

Arguments

`matA`	Optional projection matrix A (square, finite, nonnegative). If `NULL`, A is constructed as $U + (F + C)$.
`matU`	Optional survival-transition matrix U (square, finite, nonnegative). Required when `framework = "R0"` and/or when `matA` is `NULL`.
`matF`	Optional fecundity matrix F (square, finite, nonnegative). Interpreted as part of reproduction $R = F + C$.
`matC`	Optional clonal reproduction matrix C (square, finite, nonnegative). Treated as reproduction and combined with `matF` as $R = F + C$.
`groups`	A non-empty list of integer vectors specifying aggregation groups. Each stage in `1:n` must appear exactly once across the list.
`framework`	Character scalar; either `"lambda"` or `"R0"`. Determines whether aggregation is based on the projection matrix A or an R_0-based reference matrix.
`criterion`	Character scalar; either `"standard"` for standard aggregation or `"elasticity"` for elasticity-consistent aggregation.
`tol`	Numeric tolerance used in positivity/zero-mass checks and balancing calculations for the elasticity-consistent case.
`...`	Reserved for future use.

Details

Groupings are supplied by `groups`, a list of integer vectors specifying how original stages (1,. . . ,n) are combined; see `mpm_partition`. Each original stage must appear exactly once.

Aggregation can be performed under two frameworks (`"lambda"` or `"R0"`) and two criteria (`"standard"` or `"elasticity"`), which determine how stages are weighted during aggregation.

Clonal reproduction `matC` is treated as reproductive output. When both `matF` and `matC` are supplied, the effective reproductive matrix $R = F + C$ is used internally (e.g., in the `"R0"` framework) but is not returned. The aggregated effective reproduction can be recovered as `matF_agg + matC_agg`.

Irreducibility requirement: The effective projection matrix A must be nonnegative, square, and irreducible. This function enforces irreducibility of A as either `matA` (if supplied) or $A = U + (F + C)$ otherwise, and will fail if reducible.

When `framework = "R0"`, this function requires `matU` and at least one of `matF` or `matC` in order to form the effective reproductive matrix $R = F + C$ and the next generation matrix $K = R(I - U)^{-1}$. The matrix R is used internally for computing reference quantities in the `"R0"` framework but is not returned. The aggregated effective reproduction can be obtained as `matF_agg + matC_agg` when both components are supplied.

All returned matrices are aggregated using the same weighting rules implied by the selected `framework` and `criterion`.

Effectiveness measures how closely the aggregated model reproduces the behavior of the original model under the chosen `framework` and `criterion`. When `effectiveness` is high (close to 1), applying the original model and then aggregating gives nearly the same result as aggregating first and then applying the aggregated model.

The returned element `effectiveness` is computed for both criteria:

- For `criterion = "standard"`, effectiveness is computed using weights based on the reference stable stage distribution `w`.
- For `criterion = "elasticity"`, effectiveness is computed after balancing transformations (with weights based on `w * v`).

Value

A named list with elements:

`framework` The matched framework used (`"lambda"` or `"R0"`).

`criterion` The matched criterion used (`"standard"` or `"elasticity"`).

`matA_agg` Aggregated projection matrix A_{agg}.

`matU_agg` Aggregated survival-transition matrix U_{agg}, or `NULL` if `matU` was not supplied.

`matF_agg` Aggregated fecundity matrix F_{agg}, or `NULL` if `matF` was not supplied.

`matC_agg` Aggregated clonal reproduction matrix C_{agg}, or `NULL` if `matC` was not supplied.

`effectiveness` Numeric effectiveness measure for the aggregation (definition depends on `criterion`).

References

Bienvenu, F., Akcay, E., Legendre, S. and McCandlish, D.M. (2017). The genealogical decomposition of a matrix population model with applications to the aggregation of stages. *Theoretical Population Biology*, 115, 69-80. doi:10.1016/j.tpb.2017.04.002

Hooley, D. E. (2000). Collapsed matrices with (almost) the same eigenstuff. *The College Mathematics Journal*, 31(4), 297-299. doi:10.1080/07468342.2000.11974162

Salguero-Gomez, R. & Plotkin, J. B. (2010). Matrix dimensions bias demographic inferences: implications for comparative plant demography. *The American Naturalist*, 176, 710-722. doi:10.1086/657044

Examples

```
# Example aggregation of a 3x3 projection matrix to 2x2 using groups:
# group 1 = stage 1; group 2 = stages 2 and 3.
A <- matrix(c(
  0.0, 1.0, 2.0,
  0.5, 0.0, 0.0,
  0.0, 0.8, 0.9
), nrow = 3, byrow = TRUE)

res_std <- mpm_aggregate(
  matA = A,
  groups = list(c(1), c(2, 3)),
  framework = "lambda",
  criterion = "standard"
)
res_std$matA_agg
res_std$effectiveness

res_el <- mpm_aggregate(
```

```
  matA = A,
  groups = list(c(1), c(2, 3)),
  framework = "lambda",
  criterion = "elasticity"
)
res_el$matA_agg
res_el$effectiveness
```

`mpm_elasticity` *Elasticity of λ or R_0 with respect to entries of* `matA`

Description

Compute elasticities of either the dominant eigenvalue λ (λ framework) or the net reproductive rate R_0 (R_0 framework) with respect to the entries of the population projection matrix `matA`.

Usage

```
mpm_elasticity(
  matA = NULL,
  matF = NULL,
  matU = NULL,
  matC = NULL,
  framework = c("lambda", "R0"),
  normalize = TRUE,
  tol = 1e-12
)
```

Arguments

`matA`	A square projection matrix (numeric matrix). Required when `framework = "lambda"`. Must be `NULL` when `framework = "R0"`.
`matF`	A square fecundity/sexual reproduction matrix (numeric matrix). Required when `framework = "R0"`. Must be `NULL` when `framework = "lambda"`.
`matU`	A square survival/growth/transition matrix (numeric matrix). Required when `framework = "R0"`. Must be `NULL` when `framework = "lambda"`.
`matC`	Optional square clonal reproduction matrix (numeric matrix). Only allowed when `framework = "R0"`.
`framework`	Character scalar. Must be either `"lambda"` or `"R0"`.
`normalize`	Logical. Must be either `TRUE` or `FALSE`. This is relevant only when `framework = "R0"`. When true, the elasticity of `"R0"` with respect to the entries of `"matA"` are scaled to sum to 1.
`tol`	Numeric tolerance used for nonnegativity checks. Default `1e-12`.

Details

The function enforces a strict separation of inputs by framework:

- `framework = "lambda"`: `matA` must be provided and `matF`, `matU`, and `matC` must be `NULL`.
- `framework = "R0"`: `matF` and `matU` must be provided, `matC` is optional, and `matA` must be `NULL`. The projection matrix is constructed internally as `matA = matR + matU`, where `matR = matF (+ matC)`.

Irreducibility of the resulting `matA` is enforced using the internal helper `.check_irreducible_hj` (defined elsewhere in the package).

Eigen-elements (e.g., dominant eigenvalue, left and right eigenvectors) are obtained via the function `dominant_eigen`.

Lambda framework: Elasticity is the elasticity of λ with respect to the entries of `matA`.

R0 framework: Elasticity is the elasticity of R_0 with respect to the entries of the internally constructed `matA`.

Value

A list with elements:

- `framework`: the chosen framework.
- `elasticity`: a matrix of elasticities with the same dimensionality as `matA`.

References

Caswell, H. (2001). *Matrix population models: construction, analysis and interpretation* (2nd ed.). Sinauer.

See Also

`dominant_eigen`, `spectral_radius`

Examples

```
## Lambda framework: matA provided directly
matA <- matrix(
  c(0.2, 1.2,
    0.3, 0.4),
  nrow = 2, byrow = TRUE
)
out_lambda <- mpm_elasticity(matA = matA, framework = "lambda")
str(out_lambda)

## R0 framework: matA constructed from matF and matU
matU <- matrix(
  c(0.2, 0.0,
    0.3, 0.4),
  nrow = 2, byrow = TRUE
)
matF <- matrix(
```

```
    c(0.0, 1.2,
      0.0, 0.0),
    nrow = 2, byrow = TRUE
  )
  out_R0 <- mpm_elasticity(matF = matF, matU = matU, framework = "R0")
  str(out_R0)
```

`mpm_partition` *Create a partitioning matrix for MPM aggregation*

Description

Builds a 0/1 partitioning matrix P that maps an n-stage model to an m-stage aggregated model by summing specified original stages within each aggregated stage.

Usage

```
mpm_partition(groups, n = NULL)
```

Arguments

`groups`	A list of integer vectors giving stage indices for each aggregated group.
`n`	Optional integer. Number of original stages. If `NULL`, inferred as `max(unlist(groups))`. Supplying `n` can help detect errors when some stage indices are missing from `groups`, since all indices `1:n` must appear exactly once.

Details

The grouping is provided as a list of integer vectors. For example, `groups = list(c(1), c(2:3))` defines $m = 2$ aggregated stages: the first contains original stage 1, and the second contains original stages 2 and 3.

Each original stage index `1:n` must appear exactly once across `groups`.

Value

A numeric matrix P of dimensionality `m x n` with entries 0 or 1.

Examples

```
g <- list(c(1), c(2:3))
P <- mpm_partition(g, n = 3)
P
```

`reproductive_values` *Compute reproductive values*

Description

Returns the reproductive value vector `v` for a square nonegative matrix `A`, defined as the left eigenvector associated with the dominant eigenvalue (spectral radius). The vector is normalized in the usual demographic way so that `sum(v * w) = 1`, where `w` is the stable stage distribution returned by `stable_stage`.

Usage

```
reproductive_values(A, tol = 1e-12, ensure_positive = TRUE)
```

Arguments

`A`	A square numeric nonnegative matrix.
`tol`	Numeric tolerance used for normalization/positivity checks.
`ensure_positive`	Logical. If `TRUE`, attempts to orient the eigenvector so its entries are mostly positive by flipping its sign if necessary.

Details

For matrix population models with nonegative irreducible matrix, reproductive values are strictly positive (up to scaling).

Value

A numeric vector of length `nrow(A)` giving reproductive values, scaled so that `sum(v * w) = 1`.

References

Caswell, H. (2001). *Matrix population models: construction, analysis and interpretation* (2nd ed.). Sinauer.

Examples

```
#irreducible example
A <- matrix(c(
  0, 1,
  0.5, 0
), nrow = 2, byrow = TRUE)

w <- stable_stage(A)
v <- reproductive_values(A)
v
sum(v * w)  # should be 1
```

`spectral_radius` *Compute the spectral radius of a matrix*

Description

Returns the spectral radius of a square matrix, defined as the maximum modulus of its eigenvalues:

$$\rho(A) = \max |\lambda_i|$$

Usage

```
spectral_radius(A)
```

Arguments

`A`	A square numeric matrix.

Details

This function is primarily intended for use with matrix population models, where the spectral radius corresponds to the stable growth rate (i.e., λ) when the governing matrix is square and nonnegative. It is also used to calculate net reproductive rate R_0 from a next generation matrix.

Value

A single numeric value giving the spectral radius of `A`.

References

Horn, R. A. and Johnson, C. R. (2013). *Matrix analysis*. Cambridge University Press.

Examples

```
A <- matrix(c(
  0, 1,
  0.5, 0
), nrow = 2, byrow = TRUE)

spectral_radius(A)
```

`stable_stage` *Compute the stable stage distribution*

Description

Returns the stable stage distribution for a square nonnegative matrix `A`, defined as the right eigenvector associated with the dominant eigenvalue. The returned vector `w` is normalized so that `sum(w) = 1`.

Usage

```
stable_stage(A, tol = 1e-12, ensure_positive = TRUE)
```

Arguments

`A`	A square numeric matrix.
`tol`	Numeric tolerance used for positivity/normalization checks.
`ensure_positive`	Logical. If `TRUE`, attempts to orient the eigenvector so its entries are mostly positive by flipping its sign if necessary.

Details

For matrix population models with nonegative irreducible matrix, stable stage densities are strictly positive.

Value

A numeric vector of length `nrow(A)` giving the stable stage distribution, normalized to sum to 1.

References

Caswell, H. (2001). *Matrix population models: construction, analysis and interpretation* (2nd ed.). Sinauer.

Examples

```
#irreducible example
A <- matrix(c(
  0, 1,
  0.5, 0
), nrow = 2, byrow = TRUE)

w <- stable_stage(A)
w
sum(w)
```

Index