Package: demovuln (via r-universe)

May 28, 2026

Type Package

Title Demographic Vulnerability Metrics for Matrix Population Models

Version 0.1.0

Description Simulates temporally structured perturbations in matrix population models and computes population reduction and integrated demographic vulnerability across perturbation regimes. Perturbations can be applied to adult survival, juvenile survival, fecundity, all demographic entries, or user-defined matrix elements. The package provides tools to simulate individual perturbation trajectories, evaluate perturbation grids, and summarize demographic vulnerability in structured populations.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 8.0.0

Suggests ggplot2, knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

URL https://github.com/agimenezromero/demovuln-r

BugReports https://github.com/agimenezromero/demovuln-r/issues

NeedsCompilation no

Author Àlex Giménez-Romero [aut, cre] (ORCID: <https://orcid.org/0000-0003-2796-6801>), Meritxell Genovart [aut] (ORCID: <https://orcid.org/0000-0003-2919-1288>)

Maintainer Àlex Giménez-Romero <alex.gimenez@csic.es>

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

Date/Publication 2026-05-28 17:05:56 UTC

RemoteUrl https://github.com/cran/demovuln

RemoteRef HEAD

RemoteSha 79985e8d7c6c56bb8e833c0e40019c5e48468a8d

Contents

`apply_perturbation` *Apply a proportional perturbation to a projection matrix*

Description

Apply a proportional perturbation to a projection matrix

Usage

```
apply_perturbation(
  model,
  target,
  magnitude,
  survival_affects_fecundity = TRUE,
  custom_mask = NULL
)
```

Arguments

`model`	A `demovuln_model` object or a numeric square projection matrix.
`target`	One of `"adult_survival"`, `"juvenile_survival"`, `"fecundity"`, `"all"`, or `"custom"`.
`magnitude`	Proportional reduction applied to the selected entries. Must lie in `[0, 1]`.
`survival_affects_fecundity`	Logical. If `TRUE`, survival perturbations scale whole source-stage columns, including fecundity entries.
`custom_mask`	Optional logical matrix used when `target = "custom"`.

Value

Perturbed projection matrix.

build_target_mask *Build a perturbation target mask*

Description

Return a logical matrix selecting the entries affected by a perturbation.

Usage

```
build_target_mask(
  model,
  target,
  survival_affects_fecundity = TRUE,
  custom_mask = NULL
)
```

Arguments

model	A demovuln_model object or a numeric square projection matrix.
target	One of "adult_survival", "juvenile_survival", "fecundity", "all", or "custom".
survival_affects_fecundity	Logical. If TRUE, survival perturbations scale whole source-stage columns, including fecundity entries.
custom_mask	Optional logical matrix used when target = "custom".

Value

Logical matrix with the same dimensions as the projection matrix.

compute_vulnerability *Compute integrated demographic vulnerability*

Description

Compute integrated demographic vulnerability

Usage

```
compute_vulnerability(table, column = "population_reduction")
```

Arguments

table	Data frame returned by run_grid().
column	Name of the column containing percent population reduction.

Value

Mean percent population reduction, ignoring missing values.

`dominant_eigenvalue` *Dominant eigenvalue of a projection matrix*

Description

Dominant eigenvalue of a projection matrix

Usage

```
dominant_eigenvalue(A)
```

Arguments

`A`	Numeric square projection matrix.

Value

The real part of the eigenvalue with largest modulus.

`grid_scenarios` *Enumerate perturbation-grid scenarios*

Description

Enumerate perturbation-grid scenarios

Usage

```
grid_scenarios(grid, skip_infeasible = TRUE)
```

Arguments

`grid`	A `demovuln_grid` object.
`skip_infeasible`	Logical. If `TRUE`, discard regimes where duration is greater than period.

Value

Data frame with columns `magnitude`, `duration`, `period`, and `feasible`.

`matrix_population_model` *Matrix population model*

Description

Create a matrix population model and define the demographic targets used by perturbation functions.

Usage

```
matrix_population_model(
  A,
  fecundity_mask = NULL,
  fecundity_rows = 1L,
  adult_stages = NULL,
  juvenile_stages = NULL,
  name = NULL
)
```

Arguments

`A`	Numeric square projection matrix. Columns are source stages at time `t` and rows are destination stages at time `t + 1`.
`fecundity_mask`	Optional logical matrix with the same dimensions as `A`, identifying fecundity entries.
`fecundity_rows`	Integer vector identifying rows interpreted as newborn or reproductive-output rows. Defaults to the first row.
`adult_stages`	Optional integer vector with source-stage columns interpreted as adult or reproductive stages.
`juvenile_stages`	Optional integer vector with source-stage columns interpreted as juvenile or pre-reproductive stages.
`name`	Optional model or species label.

Value

An object of class `demovuln_model`.

perturbation_grid *Perturbation grid*

Description

Perturbation grid

Usage

```
perturbation_grid(magnitudes, durations, periods)
```

Arguments

magnitudes	Numeric vector of proportional reductions.
durations	Integer vector of perturbation durations.
periods	Integer vector of perturbation periods.

Value

An object of class demovuln_grid.

perturbation_grid_from_frequencies *Build a perturbation grid from frequencies*

Description

Build a perturbation grid from frequencies

Usage

```
perturbation_grid_from_frequencies(
  magnitudes,
  durations,
  frequencies,
  generation_time = 1,
  rounding = c("nearest", "floor", "ceil")
)
```

Arguments

magnitudes	Numeric vector of proportional reductions.
durations	Integer vector of perturbation durations.
frequencies	Numeric vector of event frequencies, interpreted as events per generation_time projection intervals.
generation_time	Number of projection intervals corresponding to one reference generation or time unit.
rounding	One of "nearest", "floor", or "ceil".

Value

An object of class demovuln_grid.

---

population_reduction *Compute percent population reduction*

Description

Compute percent population reduction

Usage

```
population_reduction(final_population, baseline_final_population)
```

Arguments

final_population	Final population size under perturbed dynamics.
baseline_final_population	Final population size under unperturbed baseline dynamics.

Value

Percent population reduction relative to the baseline.

run_grid *Simulate a perturbation grid*

Description

Simulate a perturbation grid

Usage

```
run_grid(
  model,
  target,
  grid,
  t_max,
  recovery_steps = 0L,
  start = 0L,
  initial_state = NULL,
  normalize_by_lambda = TRUE,
  survival_affects_fecundity = TRUE,
  custom_mask = NULL,
  return_trajectories = FALSE,
  skip_infeasible = TRUE,
  force_during_recovery = FALSE
)
```

Arguments

model	A demovuln_model object or a numeric square projection matrix.
target	One of "adult_survival", "juvenile_survival", "fecundity", "all", or "custom".
grid	A demovuln_grid object.
t_max	Number of projection intervals in the perturbation-forcing window.
recovery_steps	Number of additional unperturbed projection intervals.
start	Projection interval at which the first perturbation event starts.
initial_state	Optional initial population vector.
normalize_by_lambda	Logical. If TRUE, normalize matrices by the dominant eigenvalue of the unperturbed projection matrix.
survival_affects_fecundity	Logical. If TRUE, survival perturbations scale whole source-stage columns, including fecundity entries.
custom_mask	Optional logical matrix used when target = "custom".
return_trajectories	Logical. If TRUE, store individual simulation outputs for all feasible scenarios.

skip_infeasible
: Logical. If TRUE, skip regimes where duration is greater than period. If FALSE, keep them in the output table with missing population reduction.

force_during_recovery
: Logical. If TRUE, scheduled perturbations continue during the recovery window.

Value

An object of class demovuln_grid_result.

---

simulate_dynamics *Simulate dynamics under a temporally structured perturbation*

Description

Simulate dynamics under a temporally structured perturbation

Usage

```
simulate_dynamics(
  model,
  target = "adult_survival",
  magnitude,
  duration,
  period,
  t_max,
  recovery_steps = 0L,
  start = 0L,
  initial_state = NULL,
  normalize_by_lambda = TRUE,
  survival_affects_fecundity = TRUE,
  custom_mask = NULL,
  return_stage_vectors = FALSE,
  force_during_recovery = FALSE
)
```

Arguments

model	A demovuln_model object or a numeric square projection matrix.
target	One of "adult_survival", "juvenile_survival", "fecundity", "all", or "custom".
magnitude	Proportional reduction applied to the selected entries.
duration	Number of consecutive projection intervals during which each perturbation event is active.
period	Number of projection intervals between perturbation onsets.
t_max	Number of projection intervals in the perturbation-forcing window.

`recovery_steps` Number of additional unperturbed projection intervals after the forcing window.

`start` Projection interval at which the first perturbation event starts. The default `0` means that forcing can begin at the first projection step.

`initial_state` Optional initial population vector. If omitted, the stable stage distribution of the unperturbed model is used.

`normalize_by_lambda` Logical. If `TRUE`, baseline and perturbed matrices are divided by the dominant eigenvalue of the unperturbed projection matrix.

`survival_affects_fecundity` Logical. If `TRUE`, survival perturbations scale whole source-stage columns, including fecundity entries.

`custom_mask` Optional logical matrix used when `target = "custom"`.

`return_stage_vectors` Logical. If `TRUE`, return the full stage-vector trajectories.

`force_during_recovery` Logical. If `TRUE`, scheduled perturbations continue during the recovery window. The default is `FALSE`.

Value

An object of class `demovuln_simulation`.

`stable_stage_distribution` *Stable stage distribution*

Description

Stable stage distribution

Usage

```
stable_stage_distribution(model)
```

Arguments

`model` A `demovuln_model` object or a numeric square projection matrix.

Value

Numeric vector normalized to sum to one.

Index