

# Package ‘Rramas’

July 21, 2025

**Type** Package

**Title** Matrix Population Models

**Version** 0.1-8

**Date** 2025-04-30

**Depends** diagram

**Description** Analyzes and predicts from matrix population models (Caswell 2006) <[doi:10.1002/9781118445112.stat07481](https://doi.org/10.1002/9781118445112.stat07481)>.

**License** GPL (>= 2)

**LazyLoad** yes

**NeedsCompilation** no

**Author** Marcelino de la Cruz [aut, cre] (ORCID:  
<<https://orcid.org/0000-0002-9080-4525>>)

**Maintainer** Marcelino de la Cruz <[marcelino.delacruz@urjc.es](mailto:marcelino.delacruz@urjc.es)>

**Repository** CRAN

**Date/Publication** 2025-04-30 08:00:06 UTC

## Contents

as.tmatrix . . . . .	2
coryphanthaA . . . . .	3
decline . . . . .	4
projectn . . . . .	5

<b>Index</b>	<b>10</b>
--------------	-----------

---

as.tmatrix

*Population Transition Matrix*


---

## Description

Convert a matrix to transition matrix (class `tmatrix`) and display some of its properties.

## Usage

```
as.tmatrix(x, names.st = NULL, ...)
## S3 method for class 'tmatrix'
plot(x, ...)
## S3 method for class 'tmatrix'
print(x, ...)
## S3 method for class 'tmatrix'
summary(object, ...)
## S3 method for class 'summary.tmatrix'
print(x, ...)
```

## Arguments

<code>x</code>	For <code>as.tmatrix</code> , <code>x</code> is a square matrix,; for the <code>plot</code> , <code>print</code> and <code>summary</code> methods, <code>x</code> is a matrix of class <code>tmatrix</code> .
<code>object</code>	An object of class <code>tmatrix</code> , i.e, resulting from <code>as.tmatrix</code> .
<code>names.st</code>	A vector of names to rename the rows and columns of the transition matrix.
<code>...</code>	Other parameters passed to <code>print</code> and <code>plot</code> methods.

## Details

The purpose of `as.tmatrix` is to class a transition matrix as an object of class `tmatrix`, allowing the use of specific methods for transition matrices.

## Value

`as.tmatrix` returns a matrix of class `tmatrix`.

`summary.tmatrix` prints the main parameters of the transition matrix: the finite rate of increase ("lambda"), the stable stage distribution, the reproductive value and the sensitivities and elasticities matrices.

`plot.tmatrix` draw barplots of the stable stage distribution and the reproductive value and plot also a diagram of the life stage cycle.

## Author(s)

Marcelino de la Cruz Rot

**References**

- Akcakaya, H. R., Burgman, M. A. and Ginzburg L.R. 1999. *Applied Population Ecology*. Sinauer.
- Caswell, H. 2003. *Matrix Population Models: Construction, Analysis, and Interpretation*. Sinauer.

**Examples**

```
data(coryphanthaA)
coryphanthaA
coryphanthaA <- as.tmatrix(coryphanthaA)
summary(coryphanthaA)
plot(coryphanthaA)
```

---

coryphanthaA

*Transition Matrices of Three Coryphantha robbinsorum Populations*

---

**Description**

Transition matrices of three *Coryphantha robbinsorum* populations.

**Usage**

```
data(coryphanthaA)
```

**Format**

3x3 matrix.

**Source**

Although the original data are from Schmalzel et al (1995), these matrices have been transcribed from Gurevitch et al. (2002).

**References**

- Gurevitch, J., Scheiner, S. M. and Fox, G. A. 2002. *The Ecology of Plants*. Sinauer.
- Schmalzel, R.J., F.W. Reichenbacher, and S. Rutman. 1995. Demographic study of the rare *Coryphantha robbinsorum* (Cactaceae) in southeastern Arizona. *Madrono* 42:332-348.

**Examples**

```
data(coryphanthaA)
```

---

 decline

*Compute Declination and Explosion Probabilities*


---

### Description

Compute declination (or explosion) probabilities for a stage-structured population. From a vector of initial stage abundances and a transition matrix, `decline` and `explosion` compute respectively the probability that the population size falls below or surpasses some abundance thresholds during a given time interval.

### Usage

```
decline(rmas, bootsp = 1000)
explosion(rmas, bootsp = 1000)
## S3 method for class 'rmas.risk'
summary(object, q = c(0.025, 0.975), ...)
## S3 method for class 'summary.rmas.risk'
plot(x, ylim = NULL, col = NULL, xlab = NULL, ylab = NULL, main = NULL, ...)
```

### Arguments

<code>rmas</code>	An object of class <code>rmas</code> , i.e., resulting from <a href="#">projectn</a> .
<code>bootsp</code>	number of bootstrapped samples.
<code>object</code>	An object of class <code>rmas.risk</code> , i.e. resulting from <a href="#">decline</a> or <a href="#">explosion</a> .
<code>x</code>	An object of class <code>summary.rmas.risk</code> .
<code>q</code>	vector of quantiles to compute bootstrapped confidence intervals.
<code>ylim</code>	Vector with max and min values of the y (abundances) axis.
<code>col</code>	Color or vector of colors to draw the trajectories.
<code>xlab</code>	Label for the x-axis.
<code>ylab</code>	Label for the y-axis.
<code>main</code>	Text to appear as title.
<code>...</code>	Other parameters passed to <code>plot</code> and other methods.

### Details

Both `decline` and `explosion` require that some stochastic simulations for a given time interval had been previously constructed (using [projectn](#)). Using those simulations `decline` computes the probability of falling below some population threshold (and `explosion` the probability of surpassing it) as the ratio between the number of times that these threshold has been attained and the number of replications. The set of abundances in each time interval (in all the simulations) are bootstrapped (i.e. sampled with replacement) to build a confidence interval.

**Value**

Both `decline` and `explosion` return an object of class `rmas.rsik`, basically a list with the following elements

<code>cf.obs</code>	a data.frame with the evaluated thresholds and their probabilities.
<code>cf.boot</code>	a list of data.frames similar to <code>cf.obs</code> for each bootstrapped sample.
<code>abminbot</code>	a list of bootstrapped minimum (or maximum for explosion) abundances for each replica in the <code>rmas</code> object.
<code>main</code>	Text to appear as title when plotting the summary.

The methods `summary` and `plot.summary` summarize the results and print and plot probabilities and bootstrapped confidence interval of attaining a particular population threshold.

**Author(s)**

Marcelino de la Cruz Rot

**References**

Akcakaya, H. R., Burgman, M. A. and Ginzburg L.R. 1999. *Applied Population Ecology*. Sinauer.  
 Caswell, H. 2003. *Matrix Population Models: Construction, Analysis, and Interpretation*. Sinauer.

**Examples**

```
data(coryphanthaA)
coryphanthaA <- as.tmatrix(coryphanthaA)
#initial abundances:
v0 <- c(100,0,0)

# run 500 simulations of 20 years with demographic stochasticity:
simu20.ds <- projectn(v0=v0, mat=coryphanthaA, time = 20, estdem=TRUE, nrep=500)

# compute declination probabilities
simu20.ds.dec <- decline(simu20.ds)

summary(simu20.ds.dec)
```

---

projectn

*Demographic Projections*

---

**Description**

Make deterministic and stochastic demographic projections according to a transition matrix.

**Usage**

```

projectn(v0, mat, matsd = NULL, estamb = FALSE, estdem = FALSE,
         equalsign = TRUE, stmat=NULL, fecundity1=TRUE,
         nrep = 1, time = 10, management=NULL, round=TRUE)
project1(v0, mat, matsd=NULL, estamb=FALSE, estdem=FALSE,
         equalsign=TRUE, stmat=NULL, fecundity1=TRUE)
estambi(mat, matsd, equalsign)
estdemo(v0,mat, stmat=NULL, fecundity1=TRUE)
## S3 method for class 'rmas'
plot(x, sum = TRUE, mean=FALSE, type="l", harvest=FALSE, ...)
## S3 method for class 'rmas'
summary(object, stage=NULL, harvest=FALSE,...)
## S3 method for class 'summary.rmas'
plot(x, ylim=NULL, col=NULL, xlab=NULL, ylab=NULL, main=NULL,...)

```

**Arguments**

<code>v0</code>	Vector with the initial abundance of each stage.
<code>mat</code>	Transition matrix.
<code>matsd</code>	Matrix with the standard deviation of the probabilities in <code>mat</code> .
<code>estamb</code>	Logical. Should environmental stochasticity be considered to project the dynamics of the population?
<code>estdem</code>	Logical. Should demographic stochasticity be employed to project the dynamics of the population?
<code>equalsign</code>	Logical. Should the environmental deviations have all the same sign and magnitude? See details section.
<code>stmat</code>	Matrix indicating for each transition probability in <code>mat</code> which part (i.e. which proportion) should be considered resulting from fecundity (and the rest will be considered resulting from survival). See details.
<code>fecundity1</code>	Logical. Should the first row of <code>mat</code> be considered exclusively as fecundities? See details
<code>nrep</code>	Number of replications to evaluate the effects of stochasticity.
<code>time</code>	length of the demographic trajectory
<code>management</code>	Vector (or matrix) of management actions to be applied each time step.
<code>round</code>	Logical. Should the projections be rounded to the next integer each time step (i.e. consider finite individuals)?
<code>object</code>	An object of class <code>rmas</code> , i.e. resulting from <code>projectn</code> .
<code>x</code>	An object resulting from <code>projectn</code> or <code>summary.rmas</code> .
<code>stage</code>	Print only the trajectory of the stage called ...
<code>harvest</code>	Logical. Should the harvest history be summarized or plotted instead of the population one?
<code>sum</code>	Logical. If TRUE, print the trajectory of the whole population. If FALSE, print the individual trajectory of all stages.

mean	Logical. If TRUE, print the mean trajectory of all replications. If FALSE, print all the replicated trajectories.
type	Type of plot to represent the trajectories. By default, a line.
ylim	Vector with max and min values of the y (abundances) axis.
col	Color or vector of colors to draw the trajectories.
xlab	Label for the x-axis.
ylab	Label for the y-axis.
main	Text to appear as title.
...	Other parameters passed to plot and other methods.

## Details

The function `projectn` makes demographic projections by repeatedly calling (as long as the time argument) to `project1`. If no environmental or demographic stochasticity is required, `project1` will multiply the transition matrix `mat` by the vector of stage abundances.

If demographic stochasticity is required, `project1` will call to `estdemo` function, that for each time will assign abundances to each stage based in sampling from both a binomial and a poisson distribution. Sampling from `rbinom` with probability `mat[i, j]` will assign "survival" chances to each individual accounted for in the abundance vector, i.e. it would allow some of them to remain in the same stages (for transitions `mat[i, i]`) or to pass to another stage (for transitions `mat[i, j]`). Sampling from `rpois` with mean `mat[i, j]` will assign to each individual of stage `[j]` a random number of offspring of type `stage[i]`.

In the current implementation there are 3 options to generate demographic stochasticity. By default (`stmat=NULL` and `fecundity1=TRUE`) probabilities in the first row of the transition matrix (i.e. `mat[1, j]`) are assumed to represent **only** fecundities, i.e., they would not account for "survival" transitions from stage `[j]` to stage `[1]`, but only for newborns. This means that these probabilities will only be used as the mean for sampling from `rpois`. Transitions in rows others than first row will be assumed to represent "survival" transitions if its value is  $\leq 1$  and accordingly will be employed to sample from `rbinom`. Transitions  $> 1$  will be assumed to represent fecundities and will be employed to sample from `rpois`.

If `stmat=NULL` and `fecundity1=FALSE` transition probabilities in all the rows of the matrix `mat` are treated in the same way, i.e. probabilities  $\leq 1$  will be sampled from `rbinom` and probabilities  $> 1$  will be sampled from `rpois`.

If a `stmat` matrix (a matrix with values between 0 and 1) is provided, it will be used to divide transition probabilities `mat[i, j]` into fecundities (`mat[i, j] * stmat[i, j]`) and survival probabilities (`mat[i, j] - (mat[i, j] * stmat[i, j])`) and these matrices will be used to sample from `rbinom` and `rpois` respectively.

*The current implementation of `estdemo` assumes that reproduction takes place before "survival sorting", so even for individuals that wouldn't survive (according to its sampled binomial probability) offspring is computed (if appropriate) and accounted for.*

If environmental stochasticity is required (i.e., a `matsd` matrix is provided), `project1` will call to `estambi` function, that for each time will change `mat[i, j]` probabilities sampling from `rnorm` with mean=`mat[i, j]` and `sd = matsd[i, j]`. If `equalsign=TRUE` the random changes in all cells of `mat` will have the same sign and the same magnitude (relative to each individual `matsd[i, j]`). If `equalsign=FALSE` every transition probability will change independently.

If both environmental and demographic stochasticity are required project1 will call first to estambi and using the modified mat will call to estdemo.

If management is required, a vector or matrix of management actions should be provided. In the simplest case (i.e., a vector), each element in the vector will be interpreted as the management action that will be applied each time step to the corresponding stage. Positive and negative elements in the vector represent respectively the introduction or extraction of individuals from the corresponding stage. Elements whose absolute value is  $\geq 1$  will be interpreted as the introduction or extraction of exactly that number of individuals; absolute values  $< 1$  will be interpreted as the introduction or extraction of that proportion of individuals from the existing individuals in the corresponding stage. If "management" is a matrix, each time step the management actions represented by each column of the matrix will be applied sequentially, from first to last.

### Value

project1, estdemo and estambi return a vector of stage abundances with the same length that  $v_0$ . projectn return an object of class rmas, basically a list with four elements: vn, with length = (nrep) where each of its elements represents a replicate simulation and consist of a matrix of dimensions [length( $v_0$ ), time] representing the abundance of each stage at each time. If the simulation included management actions, harvest will be a list of length (nrep) where each of its elements represents the trajectory of harvest in a replicate simulation and consist of a matrix of dimensions [length( $v_0$ ), time] representing the number of individuals extracted of each stage at each time. The other two elements, mat and management, are respectively the transition matrix and the management matrix employed in the simulations. The plot method will draw the demographic trajectory of the population. By default (sum = TRUE, mean=FALSE) it will plot the abundance of the whole population (i.e. the the sum of abundances in each stage) vs. time. If nrep >1 it will plot together the trajectory of each replicated population. If ( sum = TRUE, mean=TRUE) it will plot the mean of all replicated populations. If (sum = FALSE, mean=TRUE) it wil plot the abundance (or the mean abundance in all the replications) of each stage vs. time.

The summary and plot.summary methods will print a table and draw a plot respectively with the maximum, mean + 1 sd, mean, mean - 1 sd and minimum values of population abundance in all the simulations.

### Author(s)

Marcelino de la Cruz Rot

### References

Akcakaya, H. R., Burgman, M. A. and Ginzburg L.R. 1999. *Applied Population Ecology*. Sinauer.  
Caswell, H. 2003. *Matrix Population Models: Construction, Analysis, and Interpretation*. Sinauer.

### Examples

```
data(coryphanthaA)
coryphanthaA <- as.tmatrix(coryphanthaA)

# run a deterministic simulation of 20 years from an initial state of
# 100 small juveniles:
v0 <- c(100,0,0)
```



```

simu20 <- projectn(v0=v0, mat=coryphanthaA, time = 20)
plot(simu20, sum=FALSE)
summary(simu20)

# run 100 simulations of 20 years with demographic stochasticity:
simu20.ds <- projectn(v0=v0, mat=coryphanthaA, time = 20, estdem=TRUE, nrep=100)
plot(simu20.ds)
summary(simu20.ds)

# run 100 simulations of 20 years with demographic stochasticity but
# assuming that the first row of the transition matrix represent both
# fecundity and survival, each with a 50% contribution

# first generate the stmat matrix:
stmat <- (coryphanthaA >0)
stmat <- stmat*c(0.5,0,0)
stmat

simu20.ds2 <- projectn(v0=v0, mat=coryphanthaA, time = 20, estdem=TRUE,
                      stmat=stmat, nrep=100)
plot(simu20.ds2)
summary(simu20.ds2)

# run 100 simulations of 20 years with both demographic and environmental
# stochasticity:
# first generate a sd matrix to describe environmental stochasticity:
sdenv <- coryphanthaA/20
sdenv

simu20.eds <- projectn(v0=v0, mat=coryphanthaA, matsd =sdenv, time = 20,
                      estdem=TRUE, estamb=TRUE, nrep=100)
plot(simu20.eds)
summary(simu20.eds)

# Example of management actions
# each time step, 10 individuals will be added to the first stage ,10 individuals
# will be added to the second stage, and 50 percent of the individuals in the
# third stage will be extracted

man <- c(10, 10, -0.5)
p1 <- projectn(v0 = c(100, 100,100), mat= coryphanthaA, management=man)

# summarize and plot population trajectory
summary(p1)

# summarizes and plots harvest history
summary(p1, harvest=TRUE)

```

# Index

## \* algebra

as.tmatrix, 2  
projectn, 5

## \* datasets

coryphanthaA, 3

## \* models

as.tmatrix, 2  
decline, 4  
projectn, 5

as.tmatrix, 2

coryphanthaA, 3

coryphanthaB (coryphanthaA), 3

coryphanthaC (coryphanthaA), 3

decline, 4, 4

estambi (projectn), 5

estdemo (projectn), 5

explosion, 4

explosion (decline), 4

plot.rmas (projectn), 5

plot.summary.rmas (projectn), 5

plot.summary.rmas.risk (decline), 4

plot.summary.tmatrix (as.tmatrix), 2

plot.tmatrix (as.tmatrix), 2

print.summary.tmatrix (as.tmatrix), 2

print.tmatrix (as.tmatrix), 2

project1 (projectn), 5

projectn, 4, 5

rbinom, 7

rnorm, 7

rpois, 7

summary.rmas (projectn), 5

summary.rmas.risk (decline), 4

summary.tmatrix (as.tmatrix), 2