Package 'eiopt2'

May 8, 2026

Type Package

Title Ecological Inference for RxC Tables via Nonlinear Quadratic Optimization

Version 0.1.1-6

Description Estimates RxC (R by C) vote transfer matrices (ecological contingency tables) from aggregate data by simultaneously minimizing Euclidean row-standardized unit-to-global distances.
Acknowledgements:
The authors wish to thank Generalitat Valenciana, Consellería de Educación, Cultura, Universidades y Empleo (grant CIAICO/2023/031) for supporting this research.

License GPL (>= 2)

Encoding UTF-8

Imports alabama, quadprog

Suggests ggplot2, scales

RoxygenNote 7.2.3

NeedsCompilation no

Author Jose M. Pavía [aut, cre] (ORCID: <https://orcid.org/0000-0002-0129-726X>), Fernández Victor [aut] (ORCID: <https://orcid.org/0000-0002-0595-516X>)

Maintainer Jose M. Pavía <jose.m.pavia@uv.es>

Repository CRAN

Date/Publication 2025-02-27 17:00:06 UTC

Contents

eiopt2 *Ecological Inference for RxC Tables via Simultaneous Minimization of Euclidean Row-Standardized Unit-to-Global Distances*

Description

Estimates RxC vote transfer matrices (ecological contingency tables) from aggregate data by simultaneously minimizing Euclidean row-standardized unit-to-global distances.

Usage

```
eiopt2(
  votes.election1,
  votes.election2,
  weights = "row",
 census.changes = c("adjust2", "raw", "simultaneous", "regular", "ordinary", "enriched",
    "adjust1", "semifull", "full", "fullreverse", "gold"),
  weights.init = "constant",
  eps = 1e-04,
  ilack.max = 1,
  trace = FALSE,
  kkt2.check = FALSE,
  ...
)
```

Arguments

votes.election1

data.frame (or matrix) of order NxR1 with the votes gained by (or the counts corresponding to) the R1 (row options) political options competing (available) on election 1 (or origin) in the N units considered. In general, the row margins of the N tables corresponding to the units.

votes.election2

data.frame (or matrix) of order NxC2 with the votes gained by (or the counts corresponding to) the C2 (column options) political options competing (available) on election 2 (or destination) in the N units considered. In general, the column margins of the N tables corresponding to the units.

weights

Weights to be used to ponder in the restricted minimization problem the distances between row-standardized proportions of individual units and the global row-standardized proportions. `weights` can be provided as a vector of length `N`, a matrix of order `NxR` (or order `NxR1`), or a character string from the set {`"constant"`, `"size"`, `"row"`}. When `weights` is a vector of length `N`, all the proportions of unit i are weighted by the ith coordinate of `weights`. For instance, `weights = "size"` assigns the number of votes in each unit as `weight`. When `weights` is a matrix of size `NxR`, the rth row of the ith unit receives as weight the (i,r) cell of the matrix. For instance, weights = "row" is equivalent to defining

weights as the matrix votes.election1. When weights = "constant" all the proportions are assigned equal weight. Default, "row".

census.changes A character string informing about the level of information available in votes.election1 and votes.election2 regarding new entries and exits of the election censuses between the two elections or indicating how their sum discrepancies should be handled. This argument allows eleven options; the eight options discussed in Pavia (2023) as well as two adjusting options and the mirror option of full. The options are: adjust2, raw, simultaneous, regular, ordinary, adjust1, enriched, semifull, full, fullreverse and gold. See **Details**. Default, adjust2.

weights.init Weights to be used to estimate the initial crude table of global proportions using quadratic programming. These weights ponder the unit residuals between the observed column margin votes and the expected column margin votes when the estimated global row-standardized proportions are applied to the observed row margins. The value of this argument is typically a character string chosen from the set {"constant", "size", "row"}. Default, weights.init = "constant". When weights.init = "constant", unit residuals are not weighted. When weights.init = "size", each unit residual is weighted with the number of eligible voters of the unit. When weights.init = "row", global row-standardized proportions are estimated using absolute number of votes without any weight. weights.init = "size" is used when weights.init is the vector defined in weights and weights.init = "row" when weights.init is the matrix defined in weights.

eps A positive real number indicating the tolerance for convergence of outer iterations of the barrier and/or augmented Lagrangian algorithm to be used for the function auglag of the package alabama on which this function relies on. Default, 1e-4.

ilack.max A positive integer number indicating the maximum number of outer iterations where no change in parameters is tolerated to be used for the function auglag of the package alabama on which this function relies on. The larger this number the most computational cost. Default, 1.

trace A TRUE/FALSE logical variable indicating whether information on outer iterations should be printed out. to be used for the function auglag of the package alabama on which this function relies on. If TRUE, at each outer iteration information is displayed on: (i) how well the equality constraints are satisfied, (ii) current parameter values, and (iii) current objective function value. Default, FALSE.

kkt2.check A TRUE/FALSE logical variable indicating whether the second-order Karush-Kuhn-Tucker conditions should be checked. Default is FALSE.

... Other arguments to be passed to the function. Not currently used.

Details

Description of the census.changes argument in more detail.

- adjust2: The default value. This is one of the simplest and the most popular solution for handling discrepancies between the sums of the margins (the total number of counts) of the first

and second elections. With this value the column-aggregations of the counts of the first election in `votes.election1` are proportionally adjusted to equal the aggregation of the counts in `votes.election2` of the second election. In this scenario, R is equal to R1 and C equal to C2.

- `raw`: This value defines a scenario with two elections elapsed at least some months where only the raw election data recorded in the N (territorial) units, in which the electoral space under study is divided, and census changes are not adjusted but estimated. Net entries and net exits are approached from the available information. In this scenario, net exits and net entries are estimated according to Pavia (2023). When both net entries and exits are no null, constraint (15) of Pavia (2022) applies: no transfer between entries and exits are allowed. In this scenario, R could be equal to R1 or R1 + 1 and C equal to C2 or C2 + 1.
- `simultaneous`: This is the value to be used in classical ecological inference problems, such as in ecological studies of social or racial voting, and in scenarios with two simultaneous elections. In this scenario, the sum by rows of `votes.election1` and `votes.election2` must coincide.
- `regular`: This value accounts for a scenario with two elections elapsed at least some months where (i) the column R1 of `votes.election1` corresponds to new (young) electors who have the right to vote for the first time, (ii) net exits and maybe other additional net entries are computed according to Pavia (2023). When both net entries and exits are no null, constraints (13) and (15) of Pavia (2023) apply. In this scenario, R could be equal to R1 or R1 + 1 and C equal to C2 or C2 + 1.
- `ordinary`: This value accounts for a scenario with two elections elapsed at least some months where (i) the column C1 of `votes.election2` corresponds to electors who died in the interperiod election, (ii) net entries and maybe other additional net exits are computed according to Pavia (2023). When both net entries and net exits are no null, constraints (14) and (15) of Pavia (2023) apply. In this scenario, R could be equal to R1 or R1 + 1 and C equal to C2 or C2 + 1.
- `enriched`: This value accounts for a scenario that somewhat combine `regular` and `ordinary` scenarios. It considers two elections elapsed at least some months where (i) the column R1 of `votes.election1` corresponds to new (young) electors who have the right to vote for the first time, (ii) the column C2 of `votes.election2` corresponds to electors who died in the interperiod election, (iii) other (net) entries and (net) exits are computed according to Pavia (2023). When both net entries and net exits are no null, constraints (12) to (15) of Pavia (2022) apply. In this scenario, R could be equal to R1 or R1 + 1 and C equal to C2 or C2 + 1.
- `adjust1`: This value accounts for a scenario with two elections elapsed at least some months where the census in each of the N polling units of the second election (the row-sums of `votes_election2`) are proportionally adjusted to match the corresponding census of the polling units in the first election (the row-sums of `votes_election1`).
- `semifull`: This value accounts for a scenario with two elections elapsed at least some months, where: (i) the column R1 = R of `votes.election1` totals new electors (young and immigrants) that have the right to vote for the first time in each polling unit and (ii) the column C2 = C of `votes.election2` corresponds to total exits of the census lists (due to death or emigration). In this scenario, the sum by rows of `votes.election1` and `votes.election2` must agree and constraint (15) of Pavia (2023) apply.
- `full`: This value accounts for a scenario with two elections elapsed at least some months, where R = R1, C = C2 and (i) the column R - 1 of `votes.election1` totals new (young) elec-

tors that have the right to vote for the first time, (ii) the column R of votes.election1 measures new immigrants that have the right to vote and (iii) the column C of votes.election2 corresponds to total exits of the census lists (due to death or emigration). In this scenario, the sum by rows of votes.election1 and votes.election2 must agree and constraints (13) and (15) of Pavia (2023) apply.

- fullreverse: This value is somehow the mirror version of full. It accounts for a scenario with two elections elapsed at least some months, where (i) the column R (= R1) of votes_election1 totals new electors (young and immigrants) that have the right to vote for the first time and (ii) total exits are separated out between exits due to emigration (column C - 1 of votes_election2) and deaths (column C of votes_election2). In this scenario, the sum by rows of votes_election1 and votes_election2 must agree and constraints (14) and (15) of Pavia (2023) apply.
- gold: This value accounts for a scenario similar to full, where R = R1, C = C2 and where (i) the column R - 1 of votes_election1 totals new young electors that have the right to vote for the first time, (ii) the column R of votes_election1 measures new immigrants that have the right to vote and total exits are separated out between (iii) exits due to emigration (column C - 1 of votes.election2) and (iv) deaths (column C of votes.election2). In this scenario, the sum by rows of votes.election1 and votes.election2 must agree. Constraints (12) to (15) of Pavia (2023) apply.

Value

A list with the following components

VTM — A matrix of order RxC with the estimated proportions of the row-standardized vote transitions from election 1 to election 2. In raw, regular, ordinary and enriched scenarios, this matrix includes the row and the column corresponding to net entries and net exits (when they are present).

VTM.votes — A matrix of order RxC with the estimated vote transfers from election 1 to election 2. In raw, regular, ordinary and enriched scenarios, this matrix includes the row and the column corresponding to net entries and net exits (when they are present).

VTM.units — An array of order RxCxN with the estimated proportions of the row-standardized vote transitions from election 1 to election 2 attained for each unit. In raw, regular, ordinary and enriched scenarios, each unit matrix includes the row and the column corresponding to net entries and net exits (when they are present).

VTM.votes.units — An array of order RxCxN with the estimated transfer of votes from election 1 to election 2 attained for each unit. In raw, regular, ordinary and enriched scenarios, each unit matrix includes the row and the column corresponding to net entries and net exits (when they are present).

VTM.initial.global — The matrix of order RxC obtained by aggregating across units the initial estimated matrix of row-standardized vote transitions from election 1 to election 2 used as starting points of the iterative process. In raw, regular, ordinary and enriched scenarios, this matrix includes the row and the column corresponding to net entries and net exits (when they are present).

`VTM.crude.global`
: The matrix of order RxC of estimated proportions for the row-standardized vote transitions from election 1 to election 2 in the whole space attained using quadratic programming. In `raw`, `regular`, `ordinary` and `enriched` scenarios, this matrix includes the row and the column corresponding to net entries and net exits (when they are present).

`VTM.initial.units`
: An array of order RxCxN with, by layer, the initial estimated matrices of row-standardized vote transitions from election 1 to election 2 used as starting points of the iterative process. In `raw`, `regular`, `ordinary` and `enriched` scenarios, each unit matrix includes the row and the column corresponding to net entries and net exits (when they are present).

`VTM.initial.votes.units`
: An array of order RxCxN with, by layer, the initial estimated matrices of vote transfers from election 1 to election 2. In `raw`, `regular`, `ordinary` and `enriched` scenarios, this matrix includes the row and the column corresponding to net entries and net exits (when they are present).

`iter`
: The number of iterations employed.

`inputs`
: A list containing all the objects with the values used as arguments by the function.

Author(s)

Jose M. Pavia <pavia@uv.es>

Victor Fernandez <victor.fernandez@uv.es>

Examples

```
votes1 <- structure(list(P1 = c(16L, 4L, 13L, 6L, 1L, 16L, 6L, 17L, 48L, 14L),
                         P2 = c(8L, 3L, 0L, 5L, 1L, 4L, 7L, 6L, 28L, 8L),
                         P3 = c(38L, 11L, 11L, 3L, 13L, 39L, 14L, 34L, 280L, 84L),
                         P4 = c(66L, 5L, 18L, 39L, 30L, 57L, 35L, 65L, 180L, 78L),
                         P5 = c(14L, 0L, 5L, 2L, 4L, 21L, 6L, 11L, 54L, 9L),
                         P6 = c(8L, 2L, 5L, 3L, 0L, 7L, 7L, 11L, 45L, 17L),
                         P7 = c(7L, 3L, 5L, 2L, 3L, 17L, 7L, 13L, 40L, 8L)),
                         row.names = c(NA, 10L), class = "data.frame")
votes2 <- structure(list(C1 = c(2L, 1L, 2L, 2L, 0L, 4L, 0L, 4L, 19L, 14L),
                         C2 = c(7L, 3L, 1L, 7L, 2L, 5L, 3L, 10L, 21L, 6L),
                         C3 = c(78L, 7L, 28L, 42L, 28L, 84L, 49L, 85L, 260L, 100L),
                         C4 = c(56L, 14L, 20L, 7L, 19L, 54L, 22L, 50L, 330L, 91L),
                         C5 = c(14L, 3L, 6L, 2L, 3L, 14L, 8L, 8L, 45L, 7L)),
                         row.names = c(NA, 10L), class = "data.frame")
example <- eiopt2(votes1, votes2)$VTM


votes1b <- structure(list(P1 = c(16L, 4L),
                          P2 = c(8L, 3L)),
                     row.names = c(NA, 2L),
                     class = "data.frame")
votes2b <- structure(list(C1 = c(10L, 10L),
```

```
                        C2 = c(7L, 4L)),
                   row.names = c(NA, 2L),
                   class = "data.frame")
example2 <- eiopt2(votes1b, votes2b)$VTM
```

---

plot.eiopt2 *Graphical representation of a RxC ecological inference (vote transfer) matrix*

Description

Plot method for objects obtained with eiopt2.

Usage

```
## S3 method for class 'eiopt2'
plot(
  x,
  margins = TRUE,
  digits = 2,
  row.names = NULL,
  col.names = NULL,
  size.numbers = 6,
  size.labels = 4,
  size.margins = 4,
  colour.cells = "steelblue4",
  colour.grid = "bisque3",
  alpha = 0.5,
  which = NULL,
  ...,
  show.plot = TRUE
)
```

Arguments

x	An object output of the **eiopt2** function.
margins	A TRUE/FALSE argument informing if the margins of the matrix should be displayed. Default, TRUE.
digits	Integer indicating the number of decimal places to be shown. Default, 2.
row.names	Names to be used for the rows of the matrix.
col.names	Names to be used for the columns of the matrix.
size.numbers	A reference number indicating the average font size to be used for the transfer numbers. Default, 6.
size.labels	A number indicating the font size to be used for labels. Default, 4.
size.margins	A number indicating the font size to be used for margin numbers. Default, 4.

colour.cells	Background base colour for cells.
colour.grid	Colour to be used for grid lines.
alpha	A [0,1] number of colour transparency.
which	A vector of integers informing the units for which the aggregate transfer matrix should be plotted. Default, NULL, the global matrix is shown.
...	Other arguments passed on to methods. Not currently used.
show.plot	A TRUE/FALSE indicating if the plot should be displayed as a side-effect. By default, TRUE.

Value

Invisibly returns the (ggplot) description of the plot, which is a list with components that contain the plot itself, the data, information about the scales, panels etc.

Note

ggplot2 is needed to be installed for this function to work.

Author(s)

Jose M. Pavia, <pavia@uv.es>

Examples

```
votes1 <- structure(list(P1 = c(16L, 4L, 13L, 6L, 1L, 16L, 6L, 17L, 48L, 14L),
                         P2 = c(8L, 3L, 0L, 5L, 1L, 4L, 7L, 6L, 28L, 8L),
                         P3 = c(38L, 11L, 11L, 3L, 13L, 39L, 14L, 34L, 280L, 84L),
                         P4 = c(66L, 5L, 18L, 39L, 30L, 57L, 35L, 65L, 180L, 78L),
                         P5 = c(14L, 0L, 5L, 2L, 4L, 21L, 6L, 11L, 54L, 9L),
                         P6 = c(8L, 2L, 5L, 3L, 0L, 7L, 7L, 11L, 45L, 17L),
                         P7 = c(7L, 3L, 5L, 2L, 3L, 17L, 7L, 13L, 40L, 8L)),
                         row.names = c(NA, 10L), class = "data.frame")
votes2 <- structure(list(C1 = c(2L, 1L, 2L, 2L, 0L, 4L, 0L, 4L, 19L, 14L),
                         C2 = c(7L, 3L, 1L, 7L, 2L, 5L, 3L, 10L, 21L, 6L),
                         C3 = c(78L, 7L, 28L, 42L, 28L, 84L, 49L, 85L, 260L, 100L),
                         C4 = c(56L, 14L, 20L, 7L, 19L, 54L, 22L, 50L, 330L, 91L),
                         C5 = c(14L, 3L, 6L, 2L, 3L, 14L, 8L, 8L, 45L, 7L)),
                         row.names = c(NA, 10L), class = "data.frame")
example <- eiopt2(votes1, votes2, method = "IPF")
p <- plot(example, show.plot = FALSE)
p


votes1b <- structure(list(P1 = c(16L, 4L),
                          P2 = c(8L, 3L)),
                     row.names = c(NA, 2L),
                     class = "data.frame")
votes2b <- structure(list(C1 = c(10L, 10L),
                          C2 = c(7L, 4L)),
                     row.names = c(NA, 2L),
```

```
                      class = "data.frame")
example2 <- eiopt2(votes1b, votes2b)$VTM
p2 <- plot(example2, show.plot = FALSE)
p2
```

---

print.eiopt2 *Print a summary of an output of the eiopt2 function*

Description

Print method for objects obtained with the eiopt2 function.

Usage

```
## S3 method for class 'eiopt2'
print(x, ..., margins = TRUE, digits = 2)
```

Arguments

x	An object output of the **eiopt2** function.
...	Other arguments passed on to methods. Not currently used.
margins	A TRUE/FALSE argument informing if the margins of the transition matrix should be displayed. Default, TRUE.
digits	Integer indicating the number of decimal places to be shown. Default, 2.

Value

No return value, called for side effects.

Author(s)

Jose M. Pavia, <pavia@uv.es>

Examples

```
votes1 <- structure(list(P1 = c(16L, 4L, 13L, 6L, 1L, 16L, 6L, 17L, 48L, 14L),
                         P2 = c(8L, 3L, 0L, 5L, 1L, 4L, 7L, 6L, 28L, 8L),
                         P3 = c(38L, 11L, 11L, 3L, 13L, 39L, 14L, 34L, 280L, 84L),
                         P4 = c(66L, 5L, 18L, 39L, 30L, 57L, 35L, 65L, 180L, 78L),
                         P5 = c(14L, 0L, 5L, 2L, 4L, 21L, 6L, 11L, 54L, 9L),
                         P6 = c(8L, 2L, 5L, 3L, 0L, 7L, 7L, 11L, 45L, 17L),
                         P7 = c(7L, 3L, 5L, 2L, 3L, 17L, 7L, 13L, 40L, 8L)),
                         row.names = c(NA, 10L), class = "data.frame")
votes2 <- structure(list(C1 = c(2L, 1L, 2L, 2L, 0L, 4L, 0L, 4L, 19L, 14L),
                         C2 = c(7L, 3L, 1L, 7L, 2L, 5L, 3L, 10L, 21L, 6L),
                         C3 = c(78L, 7L, 28L, 42L, 28L, 84L, 49L, 85L, 260L, 100L),
                         C4 = c(56L, 14L, 20L, 7L, 19L, 54L, 22L, 50L, 330L, 91L),
```

```
                        C5 = c(14L, 3L, 6L, 2L, 3L, 14L, 8L, 8L, 45L, 7L)),
                        row.names = c(NA, 10L), class = "data.frame")
example <- eiopt2(votes1, votes2, method = "IPF")
print(example, digits = 1, margins = TRUE)


votes1b <- structure(list(P1 = c(16L, 4L),
                          P2 = c(8L, 3L)),
                     row.names = c(NA, 2L),
                     class = "data.frame")
votes2b <- structure(list(C1 = c(10L, 10L),
                          C2 = c(7L, 4L)),
                     row.names = c(NA, 2L),
                     class = "data.frame")
example2 <- eiopt2(votes1b, votes2b)$VTM

print(example2, digits = 1, margins = TRUE)
```

`print.summary.eiopt2` *Print a summary of a summary.eiopt2 object*

Description

Print method for `summary.eiopt2` objects

Usage

```
## S3 method for class 'summary.eiopt2'
print(x, ..., margins = TRUE, digits = 2)
```

Arguments

`x`	An `summary.eiopt2` class object.
`...`	Other arguments passed on to methods. Not currently used.
`margins`	A TRUE/FALSE argument informing if the margins of the transition matrix should be displayed. Default, `TRUE`.
`digits`	Integer indicating the number of decimal places to be shown. Default, `2`.

Value

No return value, called for side effects.

summary.eiopt2 *Summarize an eiopt2 output object*

Description

Summary method for objects obtained with the eiopt2 function

Usage

```
## S3 method for class 'eiopt2'
summary(object, ...)
```

Arguments

object	An object output of the **eiopt2** function.
...	Other arguments passed on to methods. Not currently used.

Value

An object of class "summary.eiopt2". A list with four components:

prop.matrix	A matrix of order RxC with the estimated proportions of the row-standardized vote transitions from election 1 to election 2.
counts.matrix	A matrix of order RxC with the estimated vote transfers from election 1 to election 2.
row.margins	A vector of length R with aggregate observed distribution of votes in election 1.
col.margins	A vector of length C with aggregate observed distribution of votes in election 2.

Author(s)

Jose M. Pavia, <pavia@uv.es>

Examples

```
votes1 <- structure(list(P1 = c(16L, 4L, 13L, 6L, 1L, 16L, 6L, 17L, 48L, 14L),
                         P2 = c(8L, 3L, 0L, 5L, 1L, 4L, 7L, 6L, 28L, 8L),
                         P3 = c(38L, 11L, 11L, 3L, 13L, 39L, 14L, 34L, 280L, 84L),
                         P4 = c(66L, 5L, 18L, 39L, 30L, 57L, 35L, 65L, 180L, 78L),
                         P5 = c(14L, 0L, 5L, 2L, 4L, 21L, 6L, 11L, 54L, 9L),
                         P6 = c(8L, 2L, 5L, 3L, 0L, 7L, 7L, 11L, 45L, 17L),
                         P7 = c(7L, 3L, 5L, 2L, 3L, 17L, 7L, 13L, 40L, 8L)),
                         row.names = c(NA, 10L), class = "data.frame")
votes2 <- structure(list(C1 = c(2L, 1L, 2L, 2L, 0L, 4L, 0L, 4L, 19L, 14L),
                         C2 = c(7L, 3L, 1L, 7L, 2L, 5L, 3L, 10L, 21L, 6L),
                         C3 = c(78L, 7L, 28L, 42L, 28L, 84L, 49L, 85L, 260L, 100L),
                         C4 = c(56L, 14L, 20L, 7L, 19L, 54L, 22L, 50L, 330L, 91L),
                         C5 = c(14L, 3L, 6L, 2L, 3L, 14L, 8L, 8L, 45L, 7L)),
                         row.names = c(NA, 10L), class = "data.frame")
```

```
example <- eiopt2(votes1, votes2)
summary(example)


votes1b <- structure(list(P1 = c(16L, 4L),
                          P2 = c(8L, 3L)),
                     row.names = c(NA, 2L),
                     class = "data.frame")
votes2b <- structure(list(C1 = c(10L, 10L),
                          C2 = c(7L, 4L)),
                     row.names = c(NA, 2L),
                     class = "data.frame")
example2 <- eiopt2(votes1b, votes2b)$VTM

summary(example2)
```

Index