Package 'DyadRatios'

May 11, 2026

Type Package

Title Dyad Ratios Algorithm for Latent Variable Estimation

Version 2.1

Date 2026-05-11

Description Implements the Dyad Ratios algorithm for estimating latent variables from time-series survey data. The algorithm estimates a latent mood dimension (or two dimensions) from a set of issue opinion series. Supports annual, quarterly, monthly, and daily aggregation intervals, optional exponential smoothing, and up to two latent dimensions. Input data can be provided as a data frame or read from delimited text files. Based on Stimson's 'MCalc' C++ program. See Stimson (2018) <doi:10.1177/0759106318761614> for more details.

License GPL-3

Encoding UTF-8

Depends R (>= 4.1.0)

Imports Rcpp (>= 1.0.0)

LinkingTo Rcpp

VignetteBuilder knitr

RoxygenNote 7.3.3

Suggests testthat (>= 3.0.0), ggplot2, knitr, rmarkdown, dplyr, lubridate, rio, tidyr

NeedsCompilation yes

Author James Stimson [aut] (Original C++ implementation), Dave Armstrong [cre, aut]

Maintainer Dave Armstrong <davearmstrong.ps@gmail.com>

Repository CRAN

Date/Publication 2026-05-11 19:00:14 UTC

Contents

dyadratios-package . . . 2
boot_dr . . . 3
extract . . . 5
get_mood . . . 8
jennings . . . 9
plot.boot_dr . . . 9
plot.extract . . . 10
print.extract . . . 11
summary.extract . . . 11

Index **12**

`dyadratios-package` *dyadratios: Dyad Ratios Algorithm for Latent Public Opinion Estimation*

Description

Implements the Dyad Ratios algorithm (Stimson 1991) for estimating latent public mood from a collection of time-series survey marginals. The computationally intensive estimation loop is written in C++ (via Rcpp) and is a faithful translation of James Stimson's original 'MCalc' program. The R layer handles data ingestion, temporal aggregation, result formatting, and visualisation.

Main functions

`extract` Run the algorithm on a data frame.

Input data format

The primary input is a data frame where each row is one survey marginal:

- A column identifying the opinion **variable** (issue series).
- A **date** column (any format coercible by `as.Date`).
- An **index** column: the survey proportion or percentage.
- An optional **n** column: number of respondents (used as weight; defaults to 1000 if omitted).

Author(s)

Maintainer: Dave Armstrong <davearmstrong.ps@gmail.com>

Authors:

- James Stimson (Original C++ implementation)

References

Stimson, J. A. (1991). *Public Opinion in America: Moods, Cycles, and Swings*. Boulder, CO: Westview Press.

Stimson, J. A. (1999). *Public Opinion in America*, 2nd ed. Westview.

`boot_dr` *Bootstrap the Dyad Ratios estimate*

Description

The bootstrapping algorithm was added to the package and code by Dave Armstrong. It is worth noting that this was not something Stimson had originally implemented nor even something that he necessarily even endorsed. While bootstrapping is a common method for estimating uncertainty in complex estimators, like this one, use these results at your own risk.

Usage

```
boot_dr(
  obj,
  data,
  R = 200L,
  level = 0.95,
  pw = FALSE,
  seed = NULL,
  parallel = FALSE
)
```

Arguments

`obj`	An object of class `"extract"` returned by `extract`. The stored `call` is used to replay the estimation identically on each bootstrap draw.
`data`	The original data frame that was passed to `extract` to produce `obj`. The index column is perturbed for each replication; all other columns are used as-is.
`R`	Integer. Number of bootstrap replications. Default `200`.
`level`	Numeric in (0, 1). Confidence level for the interval. Default `0.95`.
`pw`	Logical. Whether to calculate pairwise differences between times
`seed`	Integer or `NULL`. Passed to `set.seed` before the bootstrap loop. Default `NULL`.
`parallel`	Logical. Parallelise the outer loop using **parallel** if available? Useful for large `R`. Default `FALSE`.

Details

Generates a sampling distribution around the latent mood trajectory by repeatedly drawing synthetic survey marginals from a binomial model and re-running `extract`. The original (unperturbed) estimate is used as the point estimate; the bootstrap draws characterise uncertainty around it.

All model parameters (aggregation interval, column names, smoothing, etc.) are taken directly from the stored call inside `obj`, so there is no risk of the bootstrap replications being run with different settings than the original.

Each replication draws $y_i \sim \text{Binomial}(n_i, p_i)$ where p_i is the observed proportion and n_i is the sample size, then replaces the index column with y_i/n_i (rescaled to the same 0–100 vs 0–1 convention as the original) and re-runs `extract` using the exact call stored in `obj`. Replications that error (e.g. due to a degenerate draw) are silently discarded; `attr(result, "R")` reports how many succeeded.

Value

A list of class `"boot_dr"` with two or four components depending on whether the original `obj` used one or two dimensions:

`estimates` A `data.frame` with one row per time period and columns `period`, `year`, `month`, `quarter`, `mood` (original point estimate), `lower`, and `upper` (confidence bounds at `(1-level)/2` and `1-(1-level)/2`). When `n_dim = 2`, three additional columns are appended: `mood_dim2`, `lower_dim2`, and `upper_dim2`.

`samples` An `n_periods` × `R` matrix of raw bootstrap dimension-1 trajectories.

`samples_dim2` (`n_dim = 2` only) An `n_periods` × `R` matrix of raw bootstrap dimension-2 trajectories.

The list also carries attributes `R` (number of successful replications), `level`, `agg_interval`, and `n_dim`.

See Also

`extract`, `plot.boot_dr`

Examples

```
# Build a small synthetic dataset: 4 items measured annually over 20 years
set.seed(42)
n_years <- 20
years   <- seq(1980, length.out = n_years)
items   <- c("item_a", "item_b", "item_c", "item_d")

dat <- do.call(rbind, lapply(items, function(item) {
  data.frame(
    varname = item,
    date    = as.Date(paste0(years, "-07-01")),
    index   = 50 + cumsum(rnorm(n_years, 0, 1.5)) + rnorm(n_years, 0, 2),
    n       = sample(800:1200, n_years, replace = TRUE)
  )
}))
```

```
# Run the original estimate first
res <- extract(dat, n_col = "n", smoothing = FALSE)

# Bootstrap with 100 replications (use more in practice)
boot <- boot_dr(res, dat, R = 100, seed = 1)

# estimates is the summary data frame
head(boot$estimates)

# mood is the original point estimate; lower/upper are the 95% CI
boot$estimates[, c("year", "mood", "lower", "upper")]

# samples is the raw n_periods x R matrix of bootstrap trajectories
dim(boot$samples)

# Plot the trajectory with uncertainty ribbon
plot(boot)
```

`extract` *Run the Dyad Ratios Algorithm*

Description

Estimates one or two latent opinion dimensions from a collection of time-series issue variables using the Dyad Ratios algorithm developed by James Stimson (Stimson 1991, 1999). This function accepts already-loaded data as a `data.frame`, handles aggregation to the requested interval, standardises the issue matrix, passes it to the compiled C++ core, and returns a richly-annotated result object.

Usage

```
extract(
  data,
  varname_col = "varname",
  date_col = "date",
  index_col = "index",
  n_col = NULL,
  agg_interval = c("annual", "quarterly", "monthly", "daily", "multi_year"),
  multiple = 1L,
  start_date = NULL,
  end_date = NULL,
  n_dim = 1L,
  smoothing = TRUE,
  tol = 0.001,
  fiscal_year_end = 12L,
  smooth_seed = 1L
)
```

Arguments

data	A data.frame with at minimum the columns specified in varname_col, date_col, index_col, and optionally n_col. Each row represents one survey marginal: the proportion (or count) of respondents taking a particular position on one issue at one point in time. The variable name column identifies which opinion item the row belongs to.
varname_col	Character. Name of the column that identifies the opinion variable (issue series). Default "varname".
date_col	Character. Name of the column containing the observation date. Must be coercible to Date via as.Date(). Default "date".
index_col	Character. Name of the column containing the survey marginal value (e.g. proportion liberal, per cent approving, etc.). Default "index".
n_col	Character or NULL. Name of the column containing the number of respondents (sample size / weight). When NULL (default) all observations receive equal weight of 1000, matching the original program's default.
agg_interval	Character. Temporal aggregation level. One of "annual" (default), "quarterly", "monthly", "daily", or "multi_year".
multiple	Integer. Number of years per period when agg_interval = "multi_year". Default 1.
start_date	Optional Date (or character coercible by as.Date) giving the earliest date to include. Default: the earliest date found in data.
end_date	Optional Date (or character coercible by as.Date) giving the latest date to include. Default: the latest date found in data.
n_dim	Integer 1 or 2. Number of latent dimensions to extract. Default 1.
smoothing	Logical. Apply optimal exponential smoothing to each forward and backward pass. Default TRUE.
tol	Numeric. Convergence tolerance (maximum weighted change in item-mood correlations between iterations). Default 0.001.
fiscal_year_end	Integer 1–12. Final month of the fiscal / policy year; use 12 (default) for calendar years. When set to a value < 12 observations after this month are rolled forward into the next year, consistent with the original program's FinalMonth parameter.
smooth_seed	Integer. Seed for the C random number generator used inside the exponential-smoothing optimiser (eSmooth). The default value of 1 replicates MCalc's implicit behaviour: MCalc never calls srand(), so C's rand() starts from seed 1 on every run. Change this only if you need to explore sensitivity to the smoothing optimiser's random fallback step. Has no effect when smoothing = FALSE.

Details

The algorithm iterates between a forward pass and a backward pass. In each pass every period's latent score is estimated as the weighted average of ratios of issue values relative to all other periods for which both values are non-missing. The weights are the squared correlations of each issue with the current mood estimate. Convergence is declared when the maximum weighted change in these correlations falls below tol.

Issues are standardised to a mean of 100 and a standard deviation of 10 before estimation, then the final mood series is rescaled to have the weighted mean and standard deviation of the raw issue series.

Value

An object of class `"extract"`, which is a named list containing:

`mood` Numeric vector of estimated public mood, one value per period. When `n_dim = 2` this is the first dimension.

`mood_dim2` Numeric vector of the second dimension, or `NA` when `n_dim = 1`.

`periods` A `data.frame` describing each period (year, month/quarter where applicable) matching the length of `mood`.

`loadings` A `data.frame` of variable names with loading on each extracted dimension.

`iterations` A `data.frame` of the iteration history (convergence, reliability, smoothing alphas per iteration).

`n_series` Integer. Number of opinion series retained after dropping constant series.

`n_periods` Integer. Number of time periods.

`n_obs` Integer. Total number of input observations used.

`alpha_F` Final forward-pass smoothing parameter.

`alpha_B` Final backward-pass smoothing parameter.

`eigenvalue` Eigenvalue estimate for the first dimension.

`variance_explained` Proportion of total variance accounted for by the first dimension.

`variance_explained_dim2` (`n_dim = 2` only) Additional proportion of total variance accounted for by the second dimension (i.e. its share of the variance left unexplained by dimension 1). `NA` for single-dimension solutions.

`call` The matched `call`.

`settings` List of all parameter values used.

References

Stimson, J. A. (1991). *Public Opinion in America: Moods, Cycles, and Swings*. Westview Press.

Stimson, J. A. (1999). *Public Opinion in America*, 2nd ed. Westview Press.

Examples

```
# Minimal synthetic example
set.seed(42)
n <- 60
dates <- seq(as.Date("1980-01-01"), by = "year", length.out = n / 3)
dat <- data.frame(
  varname = rep(c("item_a", "item_b", "item_c"), each = length(dates)),
  date    = rep(dates, 3),
  index   = c(seq(40, 65, length.out = length(dates)) + rnorm(length(dates), 0, 2),
              seq(55, 35, length.out = length(dates)) + rnorm(length(dates), 0, 2),
              seq(45, 70, length.out = length(dates)) + rnorm(length(dates), 0, 3)),
```

```
  n       = 1000
)
result <- extract(dat, n_col = "n", smoothing = FALSE)
print(result)
```

`get_mood` *Extract mood estimates as a data frame*

Description

Combines the period descriptor table from an `extract` result with the estimated mood trajectory, returning a plain `data.frame` that is convenient for further analysis or export.

Usage

```
get_mood(obj, ...)
```

Arguments

`obj`	An object of class `"extract"` returned by `extract`.
`...`	Ignored; included for potential future use.

Value

A `data.frame` with one row per time period. Columns `period`, `year`, `month`, and `quarter` are inherited from the period table inside `obj`. When a single dimension was estimated an additional column `mood` is appended. When two dimensions were estimated, columns `mood_dim1` and `mood_dim2` are appended instead.

Examples

```
set.seed(1)
dat <- data.frame(
  varname = rep(c("a", "b", "c"), each = 20),
  date    = rep(seq(as.Date("1980-01-01"), by = "year", length.out = 20), 3),
  index   = 50 + rnorm(60, 0, 5),
  n       = 1000L
)
res <- extract(dat, n_col = "n", smoothing = FALSE)
mood_df <- get_mood(res)
head(mood_df)
```

jennings *Jennings Government Trust Data*

Description

A dataset of survey marginals from the British Social Attitudes (BSA) survey, measuring public trust in government. These marginals are commonly used as input to the Dyad Ratios Algorithm for constructing latent time series. We replaced missing sample sizes with a value of 850, which is roughly the minimum sample size observed in the data.

Format

A data frame with 4 variables and ‘r nrow(jennings)‘ rows:

variable Character string identifying the survey question or series.

date Date the survey was fielded.

value percentage of people indicating distrust in the government.

n Sample size for the survey wave.

Source

Jennings, W. N. Clarke, J. Moss and G. Stoker (2017). "The Decline in Diffuse Support for National Politics: The Long View on Political Discontent in Britain" In *Public Opinion Quarterly*, 81(3), 748-758. doi:10.1093/poq/nfx020

Examples

```
data(jennings)
head(jennings)
```

plot.boot_dr *Plot a boot_dr object*

Description

Draws a ribbon plot of the bootstrap confidence interval around the original mood estimate. Requires **ggplot2**; falls back to base R if unavailable.

Usage

```
## S3 method for class 'boot_dr'
plot(x, dim = 1L, title = "Estimated Public Mood", ylab = "Mood", ...)
```

Arguments

x	A boot_dr object returned by boot_dr.
dim	Integer. Which dimension to plot (1 or 2). Dimension 2 is only available when the original extract call used n_dim = 2. Default 1.
title	Character. Plot title.
ylab	Character. Y-axis label.
...	Ignored.

Value

Invisibly, the ggplot2 object or NULL (base R).

---

plot.extract *Plot estimated public mood*

Description

Produces a time-series plot of the estimated latent mood. Requires **ggplot2**. If **ggplot2** is not installed, falls back to base R graphics.

Usage

```
## S3 method for class 'extract'
plot(x, dim = 1L, title = "Estimated Public Mood", ylab = "Mood", ...)
```

Arguments

x	A extract object.
dim	Integer. Which dimension to plot (1 or 2). Default 1.
title	Character. Plot title. Default "Estimated Public Mood".
ylab	Character. Y-axis label. Default "Mood".
...	Additional arguments passed to the plotting function.

Value

Invisibly, the plot object (ggplot2) or NULL (base R).

print.extract *Print method for extract objects*

Description

Print method for extract objects

Usage

```
## S3 method for class 'extract'
print(x, ...)
```

Arguments

x	A extract object returned by extract.
...	Ignored.

summary.extract *Summary method for extract objects*

Description

Prints a detailed report similar to the log file produced by the original 'MCalc' program, including the iteration history, variable loadings, and variance accounting.

Usage

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

Arguments

object	A extract object.
...	Ignored.

Value

Invisibly returns object.

Index

∗ **datasets**
jennings, 9

boot_dr, 3, *10*

call, *7*

DyadRatios *(*dyadratios-package*)*, 2
DyadRatios-package
*(*dyadratios-package*)*, 2
dyadratios-package, 2

extract, *2–4*, 5, *8*, *10, 11*

get_mood, 8

jennings, 9

plot.boot_dr, *4*, 9
plot.extract, 10
print.extract, 11

summary.extract, 11