Package: sddr (via r-universe)

July 20, 2026

Title Spatial Distribution Dynamics

Version 0.1.1.9000

Description A tidy toolkit for distribution dynamics: analysing how a cross-sectional distribution of values evolves over time and where it settles in the long run. Provides discrete-time, spatial, rank and local indicator of spatial association ('LISA') Markov transition estimation, ergodic analysis (steady-state, mean first passage and sojourn times), rank-mobility measures (Kendall's tau and the Theta statistic) and Markov mobility indices. Methods use long-format 'id'/'time'/'value' data rather than transition matrices and build on the distribution-dynamics literature (Quah (1993); Rey (2001) <doi:10.1111/j.1538-4632.2001.tb00444.x>). Results are validated for numerical parity against the reference 'giddy' library.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

Imports stats

Suggests reticulate, spdep, testthat (>= 3.0.0)

Config/testthat/edition 3

URL https://github.com/mqfarooqi1/sddr

BugReports https://github.com/mqfarooqi1/sddr/issues

Config/roxygen2/version 8.0.0

Repository https://mqfarooqi1.r-universe.dev

Date/Publication 2026-07-20 07:57:06 UTC

RemoteUrl https://github.com/mqfarooqi1/sddr

RemoteRef HEAD

RemoteSha fb060f78ff64180107e6340938a5bace93076137

Contents

`full_rank_markov` *Full-rank Markov chain*

Description

A Markov chain over *ranks* rather than discretised classes: each unit's rank within the cross-section (1 = highest value, `n` = lowest) becomes its state, and the `n` by `n` matrix records how ranks transition from one period to the next (Rey, 2014). Unlike `markov()` no binning is needed — the full rank ordering is retained.

Usage

```
full_rank_markov(data, id, time, value)
```

Arguments

`data`	A data frame in long format with one row per unit and period. The panel must be balanced (every unit present in every period).
`id`, `time`, `value`	Character scalars naming the unit, period, and numeric value columns.

Value

An object of class `sddr_markov` (an `n` by `n` rank transition matrix, its counts, and ergodic distribution).

References

Rey, S. J. (2014). Fast algorithms for a space-time concordance measure. *Computational Statistics*, 29(3-4), 799-811.

See Also

markov(), geo_rank_markov()

Examples

```
set.seed(1)
df <- data.frame(
  id = rep(1:6, times = 8),
  time = rep(1:8, each = 6),
  value = rnorm(48)
)
full_rank_markov(df, "id", "time", "value")
```

---

geo_rank_markov *Geographic-rank Markov chain*

Description

A Markov chain that tracks, for each rank position, *which unit* occupies it over time (Rey, 2016). States are the geographic units, so the n by n matrix records how units exchange rank positions — a spatially explicit view of positional mobility.

Usage

```
geo_rank_markov(data, id, time, value)
```

Arguments

data	A data frame in long format with one row per unit and period. The panel must be balanced (every unit present in every period).
id, time, value	Character scalars naming the unit, period, and numeric value columns.

Value

An object of class sddr_markov (an n by n transition matrix over units, its counts, and ergodic distribution).

References

Rey, S. J. (2016). Space-time patterns of rank concordance: Local indicators of mobility association. *Annals of the AAG*, 106(4), 788-803.

See Also

full_rank_markov(), markov()

Examples

```
set.seed(1)
df <- data.frame(
  id = rep(1:6, times = 8),
  time = rep(1:8, each = 6),
  value = rnorm(48)
)
geo_rank_markov(df, "id", "time", "value")
```

`lisa_markov` *LISA Markov chain*

Description

A Markov chain over the four quadrants of the Moran scatterplot (Rey, 2001). Each unit, in each period, is placed in a quadrant according to the sign of its value (relative to the cross-sectional mean) and the sign of its spatial lag:

`HH` high value, high neighbourhood (quadrant 1).

`LH` low value, high neighbourhood (quadrant 2).

`LL` low value, low neighbourhood (quadrant 3).

`HL` high value, low neighbourhood (quadrant 4).

The resulting 4 by 4 transition matrix describes how units move between local spatial-association types over time — capturing the co-evolution of a unit and its neighbourhood.

Usage

```
lisa_markov(data, id, time, value, weights, row_standardize = TRUE)
```

Arguments

`data`	A data frame in long format with one row per unit and period. The panel must be balanced (every unit present in every period).
`id`, `time`, `value`	Character scalars naming the unit, period, and numeric value columns.
`weights`	Spatial weights: a square numeric matrix whose row and column names match the unit ids, or an **spdep** `listw` or `nb` object.
`row_standardize`	Logical; row-standardise `weights` before computing the spatial lag (default `TRUE`).

Value

An object of class `sddr_markov`: a 4 by 4 transition matrix over the quadrants `HH`, `LH`, `LL`, `HL`, plus the per-unit-per-period `quadrants` matrix.

References

Rey, S. J. (2001). Spatial empirics for economic growth and convergence. *Geographical Analysis*, 33(3), 195-214. doi:10.1111/j.15384632.2001.tb00444.x

See Also

`spatial_markov()`, `markov()`

Examples

```
set.seed(1)
n <- 9; periods <- 8
W <- matrix(0, n, n, dimnames = list(1:n, 1:n))
for (i in 1:n) {                        # ring contiguity
  W[i, i %% n + 1] <- 1
  W[i, (i - 2) %% n + 1] <- 1
}
df <- data.frame(id = rep(1:n, times = periods),
                 time = rep(1:periods, each = n),
                 value = rnorm(n * periods))
lisa_markov(df, "id", "time", "value", weights = W)
```

`markov` *Classic Markov chain for distribution dynamics*

Description

Discretises a continuous longitudinal variable into ordered classes and estimates a discrete-time, first-order Markov transition probability matrix together with its ergodic (steady-state) distribution. This is the classic distribution-dynamics estimator of Quah (1993): the transition matrix summarises how units move between parts of the distribution from one period to the next, and the steady-state distribution describes where the process settles in the long run.

Input is *tidy*, long-format panel data with one row per unit-period. The estimator is agnostic to what the units are (regions, firms, households, ...) and to whether space is involved; spatial conditioning is added by a companion `spatial_markov()` function (available in a later release).

Usage

```
markov(data, id, time, value, k = 5, breaks = NULL, fixed = TRUE)
```

Arguments

`data`	A data frame in long format with one row per unit and period.
`id`	Name (character scalar) of the column identifying the unit.
`time`	Name (character scalar) of the column giving the period. Periods are compared in sorted order within each unit.

value	Name (character scalar) of the numeric column to analyse.
k	Integer number of classes to discretise value into. Default 5 (quintiles). Ignored if breaks is supplied.
breaks	Optional numeric vector of *interior* class boundaries. If supplied these fixed cut points are used instead of data-driven quantiles, giving length(breaks) + 1 classes.
fixed	Logical. If TRUE (default) a single set of quantile breaks is estimated from the pooled distribution and applied to every period (absolute mobility). If FALSE, breaks are re-estimated within each period (relative mobility). Ignored when breaks is supplied.

Value

An object of class sddr_markov: a list with elements

matrix The k by k row-stochastic transition probability matrix P.

transitions The k by k matrix of observed transition counts.

steady_state The ergodic (stationary) distribution of P.

classes Number of classes k.

breaks The class boundaries used (when fixed).

n Number of units contributing at least one transition.

n_transitions Total number of observed transitions.

data The input data augmented with the integer state column.

References

Quah, D. (1993). Empirical cross-section dynamics in economic growth. *European Economic Review*, 37(2-3), 426-434.

See Also

steady_state()

Examples

```
set.seed(1)
df <- data.frame(
  id = rep(1:50, each = 4),
  time = rep(2000:2003, times = 50),
  value = rnorm(200)
)
m <- markov(df, id = "id", time = "time", value = "value", k = 4)
m
m$steady_state
```

`mfpt` *Mean first passage times*

Description

The mean first passage time from class i to class j is the expected number of periods for a unit currently in i to first reach j. Together with the transition matrix these times summarise how quickly a distribution mixes. The diagonal holds the *mean recurrence time* $1/\pi_j$ (the expected return time to a class), where π is the ergodic distribution.

Computed with the Kemeny-Snell fundamental matrix $Z = (I - P + A)^{-1}$, where A has every row equal to π; then $M_{ij} = (\delta_{ij} - Z_{ij} + Z_{jj})/\pi_j$.

Usage

```
mfpt(x, ...)

## Default S3 method:
mfpt(x, ...)

## S3 method for class 'sddr_markov'
mfpt(x, ...)
```

Arguments

`x`	A row-stochastic matrix, or an object of class `sddr_markov`.
`...`	Currently unused.

Value

A numeric matrix of mean first passage times.

References

Kemeny, J. G. and Snell, J. L. (1976). *Finite Markov Chains*. Springer.

See Also

`steady_state()`, `sojourn_time()`

Examples

```
P <- matrix(c(0.8, 0.2,
              0.3, 0.7), nrow = 2, byrow = TRUE)
mfpt(P)
```

`mobility` *Markov mobility indices*

Description

Scalar summaries of how much mixing a transition matrix implies — larger values mean more mobility (a faster-churning distribution), values near zero mean a near-immobile system. Several classical indices are provided:

`prais` Prais index $(k - \mathrm{tr}(P))/(k-1)$ — average probability of leaving a class.

`determinant` $1 - |\det(P)|$.

`L2` $1 - |\lambda_2|$, one minus the second-largest eigenvalue modulus (the mixing rate).

`shorrock1` $(k - k\sum_i \pi_i p_{ii})/(k-1)$, weighted by an initial distribution `ini`.

`shorrock2` $\sum_i \sum_j \pi_i p_{ij} |i-j|/(k-1)$, penalising longer jumps.

Usage

```
mobility(x, measure = "all", ini = NULL)
```

Arguments

`x`	A row-stochastic matrix, or an object of class `sddr_markov`.
`measure`	One of `"prais"`, `"determinant"`, `"L2"`, `"shorrock1"`, `"shorrock2"`, or `"all"` (default) to return every index.
`ini`	Optional initial distribution (length `k`) for the Shorrock indices; defaults to uniform.

Value

A named numeric vector of the requested index or indices.

References

Prais, S. J. (1955). Measuring social mobility. *JRSS A*, 118(1), 56-66. Shorrocks, A. F. (1978). The measurement of mobility. *Econometrica*, 46(5), 1013-1024. doi:10.2307/1911433

Examples

```
P <- matrix(c(0.7, 0.2, 0.1,
              0.2, 0.6, 0.2,
              0.1, 0.3, 0.6), nrow = 3, byrow = TRUE)
mobility(P)
mobility(P, "prais")
```

`sojourn_time` *Sojourn times*

Description

The expected number of consecutive periods a unit remains in a class before leaving it, $1/(1 - p_{ii})$ for each class i. Large values indicate "sticky" classes with strong persistence.

Usage

```
sojourn_time(x, ...)

## Default S3 method:
sojourn_time(x, ...)

## S3 method for class 'sddr_markov'
sojourn_time(x, ...)
```

Arguments

`x`	A row-stochastic matrix, or an object of class `sddr_markov`.
`...`	Currently unused.

Value

A named numeric vector of sojourn times (one per class). Absorbing classes ($p_{ii} = 1$) return `Inf`.

See Also

`mfpt()`, `steady_state()`

Examples

```
P <- matrix(c(0.8, 0.2,
              0.3, 0.7), nrow = 2, byrow = TRUE)
sojourn_time(P)
```

spatial_markov *Spatial Markov chain for distribution dynamics*

Description

Estimates a spatial Markov chain (Rey, 2001): a set of class-transition probability matrices, each conditioned on the *spatial context* of a unit at the start of the period. Where `markov()` asks "given a unit's own class, where does it move next?", `spatial_markov()` asks "given a unit's own class **and the class of its spatial neighbourhood**, where does it move next?". Comparing the conditional matrices with the pooled (aspatial) matrix reveals whether, and how, spatial context shapes distribution dynamics.

The neighbourhood is summarised by the *spatial lag*: a (row-standardised) weighted average of neighbours' values. Supply either a spatial weights matrix via `weights` (the lag is computed for you, per period), or a precomputed spatial-lag column via `lag` (keeping the estimator independent of any particular spatial-data stack).

Usage

```
spatial_markov(
  data,
  id,
  time,
  value,
  weights = NULL,
  lag = NULL,
  k = 5,
  m = k,
  breaks = NULL,
  lag_breaks = NULL,
  fixed = TRUE,
  row_standardize = TRUE
)
```

Arguments

`data`	A data frame in long format with one row per unit and period.
`id`	Name (character scalar) of the column identifying the unit.
`time`	Name (character scalar) of the column giving the period. Periods are compared in sorted order within each unit.
`value`	Name (character scalar) of the numeric column to analyse.
`weights`	Optional spatial weights: a square numeric matrix whose row and column names match the unit ids, or an **spdep** `listw` or `nb` object (e.g. built from an `sf` layer with `spdep::poly2nb()` and `spdep::nb2listw()`). Used to compute the spatial lag of `value` within each period. Requires a balanced panel. Exactly one of `weights` or `lag` must be supplied.

`lag` Optional name (character scalar) of a column giving a precomputed spatial lag of `value`.

`k` Integer number of classes to discretise `value` into. Default `5` (quintiles). Ignored if `breaks` is supplied.

`m` Integer number of spatial-lag classes to condition on. Defaults to `k`.

`breaks` Optional numeric vector of *interior* class boundaries. If supplied these fixed cut points are used instead of data-driven quantiles, giving `length(breaks) + 1` classes.

`lag_breaks` Optional numeric vector of *interior* class boundaries for the spatial lag (the lag analogue of `breaks`). If supplied, these fixed cut points are used instead of data-driven quantiles. Supplying both `breaks` and `lag_breaks` makes the classification fully explicit, which is how exact agreement with `giddy` (`cutoffs` / `lag_cutoffs`) is obtained.

`fixed` Logical. If `TRUE` (default) a single set of quantile breaks is estimated from the pooled distribution and applied to every period (absolute mobility). If `FALSE`, breaks are re-estimated within each period (relative mobility). Ignored when `breaks` is supplied.

`row_standardize` Logical; row-standardise `weights` before computing the lag (default `TRUE`). Ignored when `lag` is supplied.

Value

An object of class `sddr_spatial_markov`: a list with elements

`matrices` Length-`m` list of `k` by `k` conditional transition probability matrices, one per spatial-lag class.

`transitions` Length-`m` list of conditional count matrices.

`pooled` The `k` by `k` pooled (aspatial) transition matrix; this equals the matrix from `markov()` with the same class definition.

`pooled_transitions` Pooled count matrix.

`steady_states` Length-`m` list of conditional ergodic distributions (`NA` where a conditional matrix has unvisited classes).

`classes`, `lag_classes` `k` and `m`.

`n`, `n_transitions` Units contributing, and total transitions.

`data` Input augmented with `lagval`, value state and lag state.

References

Rey, S. J. (2001). Spatial empirics for economic growth and convergence. *Geographical Analysis*, 33(3), 195-214. doi:10.1111/j.15384632.2001.tb00444.x

See Also

`markov()`

Examples

```
set.seed(1)
n <- 12; periods <- 8
# Ring contiguity: each unit neighbours the next and previous on a circle.
W <- matrix(0, n, n, dimnames = list(1:n, 1:n))
for (i in 1:n) {
  W[i, i %% n + 1] <- 1
  W[i, (i - 2) %% n + 1] <- 1
}
df <- do.call(rbind, lapply(seq_len(periods), function(p)
  data.frame(id = 1:n, time = p, value = rnorm(n))))
sm <- spatial_markov(df, "id", "time", "value", weights = W, k = 3)
sm
```

`steady_state` *Ergodic (steady-state) distribution of a Markov chain*

Description

Computes the long-run stationary distribution of a row-stochastic transition matrix, i.e. the probability vector π satisfying $\pi P = \pi$ with $\sum_i \pi_i = 1$. It is obtained as the left eigenvector of `P` associated with the eigenvalue closest to 1.

Usage

```
steady_state(x, ...)

## Default S3 method:
steady_state(x, ...)

## S3 method for class 'sddr_markov'
steady_state(x, ...)
```

Arguments

`x`	A row-stochastic matrix, or an object of class `sddr_markov`.
`...`	Currently unused.

Value

A named numeric vector giving the stationary distribution.

Examples

```
P <- matrix(c(0.8, 0.2,
              0.3, 0.7), nrow = 2, byrow = TRUE)
steady_state(P)
```

`tau` *Kendall's tau rank correlation*

Description

Kendall's τ measures the rank concordance between two variables — for distribution dynamics, typically a set of units' values at two points in time. It is the (concordant minus discordant) pair count, normalised, and is a natural summary of *positional* (exchange) mobility: $\tau = 1$ means ranks are perfectly preserved, $\tau = -1$ a complete reversal, and $\tau \approx 0$ extensive rank shuffling. Ties are handled with the tau-b correction.

Usage

```
tau(x, y)
```

Arguments

`x`, `y`	Numeric vectors of equal length.

Value

An object of class `sddr_tau`: a list with the statistic `tau`, the `concordant` and `discordant` pair counts, an asymptotic (normal) two-sided `p_value`, and `n`.

References

Kendall, M. G. (1938). A new measure of rank correlation. *Biometrika*, 30(1/2), 81-93. doi:10.2307/2332226

See Also

`theta()`

Examples

```
set.seed(1)
x <- rnorm(30)
y <- x + rnorm(30)          # correlated ranks
tau(x, y)
```

tau_local *Local Kendall's tau (local indicator of mobility association)*

Description

Decomposes Kendall's tau into a per-observation contribution. Each unit's local tau is the average sign of its pairwise rank concordance with all other units, $\sum_{j \neq i} S_{ij}/(n-1)$, where $S_{ij} = \text{sign}((x_i - x_j)(y_i - y_j))$. Units with local tau near 1 keep their relative position; near -1 they move against the grain. The overall (tau-a) statistic is the mean of the off-diagonal `S`.

Usage

```
tau_local(x, y)
```

Arguments

`x, y`	Numeric vectors of equal length.

Value

An object of class `sddr_tau_local`: a list with the per-observation `tau_local` vector, the global `tau` (tau-a), the sign matrix `S`, and `n`.

See Also

`tau()`

Examples

```
set.seed(1)
x <- rnorm(20); y <- x + rnorm(20)
tau_local(x, y)$tau_local
```

theta *Theta rank-mobility statistic*

Description

The Theta statistic (Rey, 2004) measures how much of the overall rank movement in a system is *within-regime* exchange. For each period-to-period transition it is the sum over regimes of the absolute *net* within-regime rank change, divided by the total absolute rank change. Values near zero indicate that rank movement is dominated by within-regime shuffling that nets out; larger values indicate coherent regime-level movement.

Usage

```
theta(y, regime)
```

Arguments

y	A numeric matrix of values, units in rows and time periods in columns (at least two columns).
regime	A vector of length `nrow(y)` assigning each unit to a regime.

Value

An object of class `sddr_theta`: a list with `theta` (one value per period-to-period transition), the per-period `total` absolute rank change, and the column rank matrix `ranks`.

References

Rey, S. J. (2004). Spatial dependence in the evolution of regional income distributions. In *Spatial Econometrics and Spatial Statistics*, 194-213.

See Also

`tau()`

Examples

```
set.seed(1)
y <- matrix(rnorm(40), nrow = 10)    # 10 units, 4 periods
regime <- rep(c("a", "b"), each = 5)
theta(y, regime)
```

Index