Package 'dyadicMarkov'

May 8, 2026

Type Package

Title Pattern Identification for Dyadic Sequences Using Transition Matrices

Version 0.1.0

Description Provides methods for analyzing dyadic interaction sequences using transition matrices within the Actor-Partner Interdependence Model. The package supports the computation of empirical transition counts, maximum likelihood estimation of transition probabilities and identification of interaction patterns in univariate and bivariate dyadic interaction sequences.

License MIT + file LICENSE

URL https://github.com/BoellenruecherM/dyadicMarkov-public

BugReports https://github.com/BoellenruecherM/dyadicMarkov-public/issues

Encoding UTF-8

Language en-US

Depends R (>= 4.1.0)

Suggests testthat (>= 3.0.0), knitr, rmarkdown

VignetteBuilder knitr

RoxygenNote 7.3.3

Config/testthat/edition 3

NeedsCompilation no

Author Mattia Böllenrücher [aut, cre, cph],
Mégane Bollenrucher [aut],
Jean-Philippe Antonietti [aut]

Maintainer Mattia Böllenrücher <mattia.boellenruecher@student.unisg.ch>

Repository CRAN

Date/Publication 2026-03-16 19:50:08 UTC

Contents

`bivariateCase` *Classify the bivariate dependence case*

Description

Classifies the bivariate case as `"trivial"`, `"univariate"`, `"partial"`, or `"complete"` using two chi-squared tests against constrained models (states = 2 only).

Usage

```
bivariateCase(empirical, alpha = 0.05)
```

Arguments

`empirical`	An empirical bivariate count matrix (must be 16x2; states = 2).
`alpha`	A single number in (0, 1) giving the significance level.

Value

A list with components `testUnivariate`, `testPartial`, and `case`.

Examples

```
chainFM_V1 <- c(1L, 2L, 1L, 2L, 2L, 1L)
chainSM_V1 <- c(2L, 1L, 2L, 1L, 1L, 2L)
chainFM_V2 <- c(1L, 1L, 2L, 2L, 1L, 2L)
chainSM_V2 <- c(2L, 2L, 1L, 1L, 2L, 1L)
emp <- countEmpBivariate(chainFM_V1, chainSM_V1, chainFM_V2, chainSM_V2, states = 2L)
bivariateCase(emp, alpha = 0.05)
```

completePattern *Select the best complete bivariate pattern by AIC*

Description

Compares complete bivariate patterns (C, D1–D4, E1–E4) using AIC and returns the selected pattern.

Usage

```
completePattern(empirical)
```

Arguments

`empirical`	An empirical bivariate count matrix (must be 16x2; states = 2).

Details

Requires a bivariate empirical count matrix for `states = 2` (output of `countEmpBivariate`).

Value

A list with components `aic` (a data frame with columns `pattern`, `matrix`, `aic`) and `pattern` (the selected pattern label).

Examples

```
chainFM_V1 <- c(1L, 2L, 1L, 2L, 2L, 1L)
chainSM_V1 <- c(2L, 1L, 2L, 1L, 1L, 2L)
chainFM_V2 <- c(1L, 1L, 2L, 2L, 1L, 2L)
chainSM_V2 <- c(2L, 2L, 1L, 1L, 2L, 1L)
emp <- countEmpBivariate(chainFM_V1, chainSM_V1, chainFM_V2, chainSM_V2, states = 2L)
completePattern(emp)
```

countEmp *Empirical transition counts for dyadic Markov chains*

Description

Computes empirical transition counts for a dyadic Markov process from two observed state sequences (FM and SM). Rows correspond to dyad states (FM, SM) and columns to the next FM state.

Usage

```
countEmp(chainFM, chainSM, states)
```

Arguments

chainFM	Vector of observed states for the first member (FM).
chainSM	Vector of observed states for the second member (SM).
states	A single integer >= 2 giving the number of states.

Value

An integer matrix with $states^2$ rows and states columns.

Examples

```
chainFM <- c(1L, 2L, 1L, 2L, 2L, 1L)
chainSM <- c(2L, 1L, 2L, 1L, 1L, 2L)
countEmp(chainFM, chainSM, states = 2L)
```

---

countEmpBivariate *Empirical transition counts for the bivariate dyadic model*

Description

Computes empirical transition counts for the bivariate dyadic model (two variables). The current implementation supports states = 2 only.

Usage

```
countEmpBivariate(chainFM_V1, chainSM_V1, chainFM_V2, chainSM_V2, states = 2L)
```

Arguments

chainFM_V1, chainSM_V1	Vectors of observed states for variable 1 (FM and SM).
chainFM_V2, chainSM_V2	Vectors of observed states for variable 2 (FM and SM).
states	A single integer. Currently only 2 is supported.

Value

An integer matrix of counts with 16 rows and 2 columns (when states = 2).

Examples

```
chainFM_V1 <- c(1L, 2L, 1L, 2L, 2L, 1L)
chainSM_V1 <- c(2L, 1L, 2L, 1L, 1L, 2L)
chainFM_V2 <- c(1L, 1L, 2L, 2L, 1L, 2L)
chainSM_V2 <- c(2L, 2L, 1L, 1L, 2L, 1L)
emp <- countEmpBivariate(chainFM_V1, chainSM_V1, chainFM_V2, chainSM_V2, states = 2L)
dim(emp)
```

mleEstimation *Maximum likelihood estimation from empirical counts*

Description

Estimates transition probabilities by maximum likelihood from an empirical count matrix returned by `countEmp` (or related counters).

Usage

```
mleEstimation(empirical)
```

Arguments

`empirical`	An empirical transition count matrix (typically from `countEmp`).

Value

A numeric matrix of MLE transition probabilities with the same dimensions as `empirical`.

Examples

```
chainFM <- c(1L, 2L, 1L, 2L, 2L, 1L)
chainSM <- c(2L, 1L, 2L, 1L, 1L, 2L)
emp <- countEmp(chainFM, chainSM, states = 2L)
mleEstimation(emp)
```

partialPattern *Select the best partial bivariate pattern by AIC*

Description

Compares the partial bivariate patterns B1/B2/B3 using AIC and returns the selected pattern.

Usage

```
partialPattern(empirical)
```

Arguments

`empirical`	An empirical bivariate count matrix (must be 16x2; states = 2).

Details

Requires a bivariate empirical count matrix for `states = 2` (output of `countEmpBivariate`).

Value

A list with components `aic` (a data frame) and `pattern` (the selected pattern label).

Examples

```
chainFM_V1 <- c(1L, 2L, 1L, 2L, 2L, 1L)
chainSM_V1 <- c(2L, 1L, 2L, 1L, 1L, 2L)
chainFM_V2 <- c(1L, 1L, 2L, 2L, 1L, 2L)
chainSM_V2 <- c(2L, 2L, 1L, 1L, 2L, 1L)
emp <- countEmpBivariate(chainFM_V1, chainSM_V1, chainFM_V2, chainSM_V2, states = 2L)
partialPattern(emp)
```

`univariatePattern` *Univariate pattern classification for dyadic Markov chains*

Description

Computes empirical transition counts, fits the unrestricted model by maximum likelihood, and performs chi-squared goodness-of-fit tests against Actor-only (AM) and Partner-only (PM) constrained models to classify the univariate dyadic pattern.

Usage

```
univariatePattern(chainFM, chainSM, states, alpha = 0.05)
```

Arguments

`chainFM`	Vector of observed states for the first member (FM).
`chainSM`	Vector of observed states for the second member (SM).
`states`	A single integer >= 2 giving the number of states.
`alpha`	A single number in (0, 1) giving the significance level.

Value

A list with two `htest` objects (`TEST.AM`, `TEST.PM`) and a string `pattern`.

Examples

```
chainFM <- c(1L, 2L, 1L, 2L, 2L, 1L)
chainSM <- c(2L, 1L, 2L, 1L, 1L, 2L)
univariatePattern(chainFM, chainSM, states = 2L, alpha = 0.05)
```

Index