Package: dynasim (via r-universe)

October 21, 2026

Type Package

Title Dynamics Similarity Coefficient

Version 1.1

Description Implements the quantile-graph based Dynamics Similarity Coefficient (DSC) for comparing intrinsic dynamics of time series.

License GPL-3

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Chun-Xiao Nie [aut, cre] (ORCID: <https://orcid.org/0000-0002-7790-0803>)

Maintainer Chun-Xiao Nie <niechunxiao2009@163.com>

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

Date/Publication 2026-04-23 21:11:26 UTC

RemoteUrl https://github.com/cran/dynasim

RemoteRef HEAD

RemoteSha 4176ca3d9faf2040b3917fb66deed05fe9d4c439

Contents

cross_transition_matrix 2
discretize_quantile 2
dsc 3
dsc_matrix 4
gdsc 4
gdsc_matrix 5
mdsc 6
transition_matrix 7

Index **8**

cross_transition_matrix

Multivariate DSC (MDSC) Input: X and Y are matrices where columns are components

Description

Multivariate DSC (MDSC) Input: X and Y are matrices where columns are components

Usage

```
cross_transition_matrix(x, y, n_s = 15, lag = 1)
```

Arguments

x	Numeric vector, first component.
y	Numeric vector, second component.
n_s	Integer, number of states.
lag	Integer, lag order.

Value

Matrix of dimension n_s x n_s.

discretize_quantile

Quantile discretization and transition probability matrix

Description

Quantile discretization and transition probability matrix

Usage

```
discretize_quantile(x, n_s = 15)
```

Arguments

x	Numeric vector representing the time series.
n_s	Integer, number of states (quantile bins). Default is 15.

Value

Integer vector of state labels (1 to n_s).

Examples

```
x <- rnorm(1000)
states <- discretize_quantile(x, n_s = 10)
```

`dsc` *Compute Dynamics Similarity Coefficient (DSC) between two time series*

Description

Compute Dynamics Similarity Coefficient (DSC) between two time series

Usage

```
dsc(x, y, n_s = 15, lag = 1, th_step = 0.005)
```

Arguments

`x`	Numeric vector, first time series.
`y`	Numeric vector, second time series.
`n_s`	Integer, number of states. Default 15.
`lag`	Integer, lag order. Default 1.
`th_step`	Numeric, threshold step size. Default 0.005.

Value

A numeric value between 0 and 1, where higher values indicate greater similarity of intrinsic dynamics.

References

Nie, Chun-Xiao. "Distinguishing time series generated by different intrinsic dynamics using quantile graphs." Communications in Nonlinear Science and Numerical Simulation (2026): 110002.

Examples

```
x <- arima.sim(model = list(ar = 0.5), n = 1000)
y <- arima.sim(model = list(ar = -0.5), n = 1000)
dsc(x, y, n_s = 15, lag = 1)
```

dsc_matrix *Compute pairwise DSC matrix for a collection of time series*

Description

Compute pairwise DSC matrix for a collection of time series

Usage

```
dsc_matrix(X, n_s = 15, lag = 1, th_step = 0.005, show_progress = FALSE)
```

Arguments

X	Matrix with time series in columns (rows = time points).
n_s	Integer, number of states.
lag	Integer, lag order.
th_step	Numeric, threshold step size.
show_progress	Logical, whether to show a progress bar.

Value

A symmetric matrix of DSC values between columns.

Examples

```
n <- 1000
X <- matrix(0,n,4)
X[,1]=arima.sim(model = list(ar = 0.5), n = n)
X[,2]=arima.sim(model = list(ar = 0.5), n = n)
X[,3]=arima.sim(model = list(ar = -0.5), n = n)
X[,4]=arima.sim(model = list(ar = -0.5), n = n)
dsc_matrix(X, n_s = 10,lag=1,th_step = 0.005)
```

gdsc *Global DSC (GDSC) across multiple lags*

Description

Global DSC (GDSC) across multiple lags

Usage

```
gdsc(x, y, n_s = 15, lags = 1:5, th_step = 0.005)
```

Arguments

x	Numeric vector, first time series.
y	Numeric vector, second time series.
n_s	Integer, number of states.
lags	Vector of lag orders to include.
th_step	Numeric, threshold step size.

Value

Numeric, average DSC over specified lags.

References

Nie, Chun-Xiao. "Distinguishing time series generated by different intrinsic dynamics using quantile graphs." Communications in Nonlinear Science and Numerical Simulation (2026): 110002.

Examples

```
x <- arima.sim(model = list(ar = c(0.5, -0.3)), n = 1000)
y <- arima.sim(model = list(ar = c(0.5, -0.3)), n = 1000)
gdsc(x, y, n_s = 15, lags = 1:3)
```

---

gdsc_matrix *Compute pairwise GDSC matrix for a collection of series*

Description

Compute pairwise GDSC matrix for a collection of series

Usage

```
gdsc_matrix(X, n_s = 15, lags = 1:5, th_step = 0.005, show_progress = FALSE)
```

Arguments

X	Matrix with time series in columns (rows = time points).
n_s	Integer, number of states.
lags	Vector of lag orders.
th_step	Numeric, threshold step size.
show_progress	Logical.

Value

Matrix of GDSC values between columns.

Examples

```
n <- 1000
X <- matrix(0,n,4)
X[,1]=arima.sim(model = list(ar = 0.5), n = n)
X[,2]=arima.sim(model = list(ar = 0.5), n = n)
X[,3]=arima.sim(model = list(ar = -0.5), n = n)
X[,4]=arima.sim(model = list(ar = -0.5), n = n)
gdsc_matrix(X, n_s = 10,lags=1:3,th_step = 0.005)
```

`mdsc` *Compute Multivariate Dynamics Similarity Coefficient (MDSC)*

Description

Compute Multivariate Dynamics Similarity Coefficient (MDSC)

Usage

```
mdsc(X, Y, n_s = 15, lag = 1, th_step = 0.005)
```

Arguments

`X`	Matrix or data.frame, each column is a component time series.
`Y`	Matrix or data.frame, each column is a component time series.
`n_s`	Integer, number of states.
`lag`	Integer, lag order.
`th_step`	Numeric, threshold step size.

Value

Numeric, MDSC value.

References

Nie, Chun-Xiao. "Distinguishing time series generated by different intrinsic dynamics using quantile graphs." Communications in Nonlinear Science and Numerical Simulation (2026): 110002.

Examples

```
X <- matrix(rnorm(3000), ncol = 3)
Y <- matrix(rnorm(3000), ncol = 3)
mdsc(X, Y, n_s = 10)
```

`transition_matrix` *Compute state transition probability matrix for a given lag*

Description

Compute state transition probability matrix for a given lag

Usage

```
transition_matrix(x, n_s = 15, lag = 1)
```

Arguments

`x`	Numeric vector, the time series.
`n_s`	Integer, number of states (quantile bins). Default 15.
`lag`	Integer, lag order for transition. Default 1.

Value

A matrix of dimension n_s x n_s with row-wise transition probabilities.

Examples

```
x <- arima.sim(model = list(ar = 0.5), n = 1000)
P <- transition_matrix(x, n_s = 15, lag = 1)
```

Index

cross_transition_matrix, 2

discretize_quantile, 2
dsc, 3
dsc_matrix, 4

gdsc, 4
gdsc_matrix, 5

mdsc, 6

transition_matrix, 7