Package: otsfeatures (via r-universe)

October 23, 2024

Type Package

Title Ordinal Time Series Analysis

Version 1.0.0

Description An implementation of several functions for feature extraction in ordinal time series datasets. Specifically, some of the features proposed by Weiss (2019) <doi:10.1080/01621459.2019.1604370> can be computed. These features can be used to perform inferential tasks or to feed machine learning algorithms for ordinal time series, among others. The package also includes some interesting datasets containing financial time series. Practitioners from a broad variety of fields could benefit from the general framework provided by 'otsfeatures'.

License GPL-2

Encoding UTF-8

LazyData true

LazyDataCompression xz

Depends R (>= 4.0.0)

RoxygenNote 7.1.2

Imports ggplot2, astsa, latex2exp, Rdpack, Bolstad2

RdMacros Rdpack

NeedsCompilation no

Author Angel Lopez-Oriona [aut, cre], Jose A. Vilar [aut]

Maintainer Angel Lopez-Oriona `<oriona38@hotmail.com>`

Date/Publication 2023-03-01 10:40:02 UTC

Repository https://anloor7.r-universe.dev

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

RemoteRef HEAD

RemoteSha 5009a2ae3b83f9b5a395de46902218e70a92af40

Contents

AustrianWages . . . 2
binarization . . . 3
ci_ordinal_asymmetry . . . 4
ci_ordinal_dispersion . . . 5
ci_ordinal_skewness . . . 6
conditional_probabilities . . . 7
CreditRatings . . . 8
c_binarization . . . 9
c_conditional_probabilities . . . 10
c_joint_probabilities . . . 11
c_marginal_probabilities . . . 12
index_ordinal_variation . . . 13
joint_probabilities . . . 14
marginal_probabilities . . . 15
ordinal_asymmetry . . . 16
ordinal_cohens_kappa . . . 17
ordinal_dispersion_1 . . . 18
ordinal_dispersion_2 . . . 19
ordinal_location_1 . . . 20
ordinal_location_2 . . . 21
ordinal_skewness . . . 22
ots_plot . . . 23
plot_ordinal_cohens_kappa . . . 24
SyntheticData1 . . . 25
SyntheticData2 . . . 26
SyntheticData3 . . . 27
test_ordinal_asymmetry . . . 27
test_ordinal_dispersion . . . 29
test_ordinal_skewness . . . 30
total_c_correlation . . . 31
total_mixed_c_correlation_1 . . . 32
total_mixed_c_correlation_2 . . . 33

Index **35**

`AustrianWages` *AustrianWages*

Description

Ordinal time series (OTS) of yearly categories of salaries for different Austrian employees

Usage

```
data(AustrianWages)
```

Format

A `list` with one element, which is:

`data` A list with 9402 MTS.

Details

Each element in `data` is an ordinal time series containing 6 states (yearly categorized wages). 9402 Austrian workers considered. The series exhibit individual lengths ranging from 2 to 32 years with the median length being equal to 22. For more information, see López-Oriona et al. (2023).

References

López-Oriona Á, Weiß C, Vilar JA (2023). "Fuzzy clustering of ordinal time series based on two novel distances with financial applications." *Manuscript submitted for publication*, 000-000.

`binarization` *Constructs the binarized time series associated with a given ordinal time series*

Description

`binarization` constructs the binarized time series associated with a given ordinal time series.

Usage

```
binarization(series, states)
```

Arguments

`series`	An OTS (numerical vector with integers).
`states`	A numeric vector containing the corresponding states.

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_n\}$ ($s_0 < s_1 < s_2 < \dots < s_n$), $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, the function constructs the binarized time series, which is defined as $\overline{\boldsymbol{Y}}_t = \{\overline{\boldsymbol{Y}}_1, \dots, \overline{\boldsymbol{Y}}_T\}$, with $\overline{\boldsymbol{Y}}_k = (\overline{Y}_{k,0}, \overline{Y}_{k,1}, \dots, \overline{Y}_{k,n})^\top$ such that $\overline{Y}_{k,i} = 1$ if $\overline{X}_k = s_i$ ($k = 1, \dots, T, , i = 0, \dots, n$). The binarized series is constructed in the form of a matrix whose rows represent time observations and whose columns represent the states in the original series.

Value

The binarized time series.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2018). *An introduction to discrete-valued time series*. John Wiley and Sons. López-Oriona Á, Vilar JA, D'Urso P (2023). "Hard and soft clustering of categorical time series based on two novel distances with an application to biological sequences." *Information Sciences*, **624**, 467–492.

Examples

```
binarized_series <- binarization(AustrianWages$data[[100]],
states = 0 : 5) # Constructing the binarized
# time series for one OTS in dataset AustrianWages
```

---

ci_ordinal_asymmetry — *Constructs a confidence interval for the ordinal asymmetry (block distance)*

Description

`ci_ordinal_asymmetry` constructs a confidence interval for the ordinal asymmetry (block distance)

Usage

```
ci_ordinal_asymmetry(
  series,
  states,
  level = 0.95,
  temporal = TRUE,
  max_lag = 1
)
```

Arguments

`series`	An OTS (numerical vector with integers).
`states`	A numeric vector containing the corresponding states.
`level`	The confidence level (default is 0.95).
`temporal`	Logical. If `temporal = TRUE` (default), the interval is computed for a time series. Otherwise, the interval is computed for i.i.d. data.
`max_lag`	If `temporal = TRUE`, the maximum considered lag to compute the estimates related to the cumulative joint probabilities.

Details

If `temporal = TRUE` (default), the function constructs the confidence interval for the ordinal asymmetry relying on Theorem 7.1.1 in Weiß (2019). Otherwise, the interval is constructed according to Theorem 4.1 in Weiß (2019).

Value

The confidence interval.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
ci_asymmetry <- ci_ordinal_asymmetry(AustrianWages$data[[100]],
states = 0 : 5) # Constructing a confidence interval for the
# ordinal asymmetry for one OTS in dataset AustrianWages
```

---

ci_ordinal_dispersion *Constructs a confidence interval for the ordinal dispersion (block distance)*

Description

`ci_ordinal_dispersion` constructs a confidence interval for the ordinal dispersion (block distance)

Usage

```
ci_ordinal_dispersion(
  series,
  states,
  level = 0.95,
  temporal = TRUE,
  max_lag = 1
)
```

Arguments

`series`	An OTS (numerical vector with integers).
`states`	A numeric vector containing the corresponding states.
`level`	The confidence level (default is 0.95).
`temporal`	Logical. If `temporal` = `TRUE` (default), the interval is computed for a time series. Otherwise, the interval is computed for i.i.d. data.
`max_lag`	If `temporal` = `TRUE`, the maximum considered lag to compute the estimates related to the cumulative joint probabilities.

Details

If `temporal = TRUE` (default), the function constructs the confidence interval for the ordinal dispersion relying on Theorem 7.1.1 in Weiß (2019). Otherwise, the interval is constructed according to Theorem 4.1 in Weiß (2019).

Value

The confidence interval.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
ci_dispersion <- ci_ordinal_dispersion(AustrianWages$data[[100]],
states = 0 : 5) # Constructing a confidence interval for the
# ordinal dispersion for one OTS in dataset AustrianWages
```

---

ci_ordinal_skewness — *Constructs a confidence interval for the ordinal skewness (block distance)*

Description

`ci_ordinal_skewness` constructs a confidence interval for the ordinal skewness (block distance)

Usage

```
ci_ordinal_skewness(series, states, level = 0.95, temporal = TRUE, max_lag = 1)
```

Arguments

`series`	An OTS (numerical vector with integers).
`states`	A numeric vector containing the corresponding states.
`level`	The confidence level (default is 0.95).
`temporal`	Logical. If `temporal = TRUE` (default), the interval is computed for a time series. Otherwise, the interval is computed for i.i.d. data.
`max_lag`	If `temporal = TRUE`, the maximum considered lag to compute the estimates related to the cumulative joint probabilities.

Details

If `temporal = TRUE` (default), the function constructs the confidence interval for the ordinal skewness relying on Theorem 7.1.1 in Weiß (2019). Otherwise, the interval is constructed according to Theorem 4.1 in Weiß (2019).

Value

The confidence interval.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
ci_skewness <- ci_ordinal_skewness(AustrianWages$data[[100]],
states = 0 : 5) # Constructing a confidence interval for the
# ordinal skewness for one OTS in dataset AustrianWages
```

`conditional_probabilities`
Computes the conditional probabilities of an ordinal time series

---

Description

`conditional_probabilities` returns a matrix with the conditional probabilities of an ordinal time series

Usage

```
conditional_probabilities(series, lag = 1, states)
```

Arguments

`series`	An OTS.
`lag`	The considered lag (default is 1).
`states`	A numerical vector containing the corresponding states.

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_n\}$ ($s_0 < s_1 < s_2 < \dots < s_n$), $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, the function computes the matrix $\widehat{\boldsymbol{P}}^c(l) = \left(\widehat{p}^c_{i-1j-1}(l)\right)_{1 \leq i,j \leq n+1}$, with $\widehat{p}^c_{ij}(l) = \frac{TN_{ij}(l)}{(T-l)N_i}$, where N_i is the number of elements equal to s_i in the realization $\overline{X}_t$ and $N_{ij}(l)$ is the number of pairs $(\overline{X}_t, \overline{X}_{t-l}) = (s_i, s_j)$ in the realization $\overline{X}_t$.

Value

A matrix with the conditional probabilities.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). “Distance-based analysis of ordinal data and ordinal time series.” *Journal of the American Statistical Association*.

Examples

```
matrix_cp <- conditional_probabilities(series = AustrianWages$data[[100]],
states = 0 : 5) # Computing the matrix of
# conditional probabilities for one series in dataset AustrianWages
```

`CreditRatings` *CreditRatings*

Description

Ordinal time series (OTS) of monthly credit ratings of different European countries

Usage

```
data(CreditRatings)
```

Format

A `list` with one element, which is:

`data` A list with 28 MTS.

Details

Each element in `data` is an ordinal time series containing 23 states (monthly credit ratings). The 28 countries of the European Union plus the United Kingdom are considered. The sample period spans from January 2000 to December 2017, thus resulting serial realizations of length $T = 216$. For more information, see Weiß (2019).

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

---

c_binarization — *Constructs the cumulative binarized time series associated with a given ordinal time series*

Description

`c_binarization` constructs the cumulative binarized time series associated with a given ordinal time series.

Usage

```
c_binarization(series, states)
```

Arguments

`series`	An OTS (numerical vector with integers).
`states`	A numeric vector containing the corresponding states.

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_n\}$ ($s_0 < s_1 < s_2 < \dots < s_n$), $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, the function constructs the cumulative binarized time series, which is defined as $\overline{\boldsymbol{Y}}_t = \{\overline{\boldsymbol{Y}}_1, \dots, \overline{\boldsymbol{Y}}_T\}$, with $\overline{\boldsymbol{Y}}_k = (\overline{Y}_{k,0}, \overline{Y}_{k,1}, \dots, \overline{Y}_{k,n-1})^\top$ such that $\overline{Y}_{k,i} = 1$ if $\overline{X}_k \leq s_i$ ($k = 1, \dots, T, , i = 0, \dots, n-1$). The cumulative binarized series is constructed in the form of a matrix whose rows represent time observations and whose columns represent the states in the original series.

Value

The binarized time series.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2018). *An introduction to discrete-valued time series*. John Wiley and Sons.

López-Oriona Á, Vilar JA, D'Urso P (2023). "Hard and soft clustering of categorical time series based on two novel distances with an application to biological sequences." *Information Sciences*, **624**, 467–492.

Examples

```
c_binarized_series <- c_binarization(AustrianWages$data[[100]],
states = 0 : 5) # Constructing the cumulative binarized
# time series for one OTS in dataset AustrianWages
```

`c_conditional_probabilities`

Computes the cumulative conditional probabilities of an ordinal time series

---

Description

`c_conditional_probabilities` returns a matrix with the cumulative conditional probabilities of an ordinal time series

Usage

```
c_conditional_probabilities(series, lag = 1, states)
```

Arguments

`series`	An OTS.
`lag`	The considered lag (default is 1).
`states`	A numerical vector containing the corresponding states.

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_n\}$ ($s_0 < s_1 < s_2 < \dots < s_n$), $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, the function computes the matrix $\widehat{\boldsymbol{F}}^c(l) = \big(\widehat{f}^c_{i-1j-1}(l)\big)_{1 \le i,j \le n}$, with $\widehat{f}^c_{ij}(l) = \frac{TN_{ij}(l)}{(T-l)N_i}$, where N_i is the number of elements less one or equal to s_i in the realization $\overline{X}_t$ and $N_{ij}(l)$ is the number of pairs $(\overline{X}_t, \overline{X}_{t-l})$ in the realization $\overline{X}_t$ such that $\overline{X}_t \le s_i$ and $\overline{X}_{t-l} \le s_j$.

Value

A matrix with the conditional probabilities.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
matrix_ccp <- c_conditional_probabilities(series = AustrianWages$data[[100]],
states = 0 : 5) # Computing the matrix of
# cumulative conditional probabilities for one series in dataset AustrianWages
```

---

c_joint_probabilities *Computes the cumulative joint probabilities of an ordinal time series*

Description

`c_joint_probabilities` returns a matrix with the cumulative joint probabilities of an ordinal time series

Usage

```
c_joint_probabilities(series, lag = 1, states)
```

Arguments

`series`	An OTS.
`lag`	The considered lag (default is 1).
`states`	A numerical vector containing the corresponding states.

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_n\}$ ($s_0 < s_1 < s_2 < \dots < s_n$), $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, the function computes the matrix $\widehat{\boldsymbol{F}}(l) = \big(\widehat{f}_{i-1j-1}(l)\big)_{1 \le i,j \le n}$, with $\widehat{f}_{ij}(l) = \frac{N_{ij}(l)}{T-l}$, where $N_{ij}(l)$ is the number of pairs $(\overline{X}_t, \overline{X}_{t-l})$ in the realization $\overline{X}_t$ such that $\overline{X}_t \le s_i$ and $\overline{X}_{t-l} \le s_j$.

Value

A matrix with the jcumulative oint probabilities.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
matrix_cjp <- c_joint_probabilities(series = AustrianWages$data[[100]],
states = 0 : 5) # Computing the matrix of
# cumulative joint probabilities for one series in dataset AustrianWages
```

c_marginal_probabilities

Computes the cumulative marginal probabilities of an ordinal time series

Description

c_marginal_probabilities returns a vector with the cumulative marginal probabilities of an ordinal time series

Usage

```
c_marginal_probabilities(series, states)
```

Arguments

series	An OTS (numerical vector with integers).
states	A numerical vector containing the corresponding states.

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_n\}$ $(s_0 < s_1 < s_2 < \dots < s_n)$, $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, the function computes the vector $\widehat{\boldsymbol{f}} = (\widehat{f}_0, \dots, \widehat{f}_n)$, with $\widehat{f}_i = \frac{N_i}{T}$, where N_i is the number of elements less than or equal to s_i in the realization $\overline{X}_t$.

Value

A vector with the cumulative marginal probabilities.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
vector_cmp <- c_marginal_probabilities(series = AustrianWages$data[[100]],
states = 0 : 5) # Computing the vector of
# cumulative marginal probabilities for one series in dataset AustrianWages
```

`index_ordinal_variation` *Computes the estimated index of ordinal variation (IOV) of an ordinal time series*

Description

`index_ordinal_variation` computes the estimated index of ordinal variation of an ordinal time series

Usage

```
index_ordinal_variation(series, states)
```

Arguments

`series`	An OTS.
`states`	A numerical vector containing the corresponding states.

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_n\}$ ($s_0 < s_1 < s_2 < \dots < s_n$), $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, the function computes the estimated IOV given by $\widehat{IOV} = \frac{4}{n}\sum_{k=1}^{n-1} \widehat{f}_k(1 - \widehat{f}_k)$, where $\widehat{f}_k$ is the standard estimate of the cumulative marginal probability for state s_k computed from the series $\overline{X}_t$.

Value

The estimated IOV.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
estimated_iov <- index_ordinal_variation(series = AustrianWages$data[[100]],
states = 0 : 5) # Computing the estimate of the IOV
# for one series in dataset AustrianWages
```

`joint_probabilities` *Computes the joint probabilities of an ordinal time series*

Description

`joint_probabilities` returns a matrix with the joint probabilities of an ordinal time series

Usage

```
joint_probabilities(series, lag = 1, states)
```

Arguments

`series`	An OTS.
`lag`	The considered lag (default is 1).
`states`	A numerical vector containing the corresponding states.

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_n\}$ ($s_0 < s_1 < s_2 < \dots < s_n$), $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, the function computes the matrix $\widehat{\boldsymbol{P}}(l) = \big(\widehat{p}_{i-1j-1}(l)\big)_{1 \leq i,j \leq n+1}$, with $\widehat{p}_{ij}(l) = \frac{N_{ij}(l)}{T-l}$, where $N_{ij}(l)$ is the number of pairs $(\overline{X}_t, \overline{X}_{t-l}) = (s_i, s_j)$ in the realization $\overline{X}_t$.

Value

A matrix with the joint probabilities.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
matrix_jp <- joint_probabilities(series = AustrianWages$data[[100]],
states = 0 : 5) # Computing the matrix of
# joint probabilities for one series in dataset AustrianWages
```

`marginal_probabilities` *Computes the marginal probabilities of an ordinal time series*

Description

`marginal_probabilities` returns a vector with the marginal probabilities of an ordinal time series

Usage

```
marginal_probabilities(series, states)
```

Arguments

`series`	An OTS (numerical vector with integers).
`states`	A numerical vector containing the corresponding states

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_n\}$ ($s_0 < s_1 < s_2 < \dots < s_n$), $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, the function computes the vector $\widehat{\boldsymbol{p}} = (\widehat{p}_0, \dots, \widehat{p}_n)$, with $\widehat{p}_i = \frac{N_i}{T}$, where N_i is the number of elements equal to s_i in the realization $\overline{X}_t$.

Value

A vector with the marginal probabilities.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
vector_mp <- marginal_probabilities(series = AustrianWages$data[[100]],
states = 0 : 5) # Computing the vector of
# marginal probabilities for one series in dataset AustrianWages
```

`ordinal_asymmetry` *Computes the estimated asymmetry of an ordinal time series*

Description

`ordinal_asymmetry` computes the estimated asymmetry of an ordinal time series

Usage

```
ordinal_asymmetry(series, states, distance = "Block", normalize = FALSE)
```

Arguments

`series`	An OTS.
`states`	A numerical vector containing the corresponding states.
`distance`	A function defining the underlying distance between states. The Hamming, block and Euclidean distances are already implemented by means of the arguments "Hamming", "Block" (default) and "Euclidean". Otherwise, a function taking as input two states must be provided.
`normalize`	Logical. If `normalize = FALSE` (default), the value of the estimated asymmetry is returned. Otherwise, the function returns the normalized estimated asymmetry.

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_n\}$ ($s_0 < s_1 < s_2 < \dots < s_n$), $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, the function computes the estimated asymmetry given by $\widehat{asym}_d = \widehat{\boldsymbol{p}}^\top (\boldsymbol{J} - \boldsymbol{I}) \boldsymbol{D} \widehat{\boldsymbol{p}}$, where $\widehat{\boldsymbol{p}} = (\widehat{p}_0, \widehat{p}_1, \dots, \widehat{p}_n)^\top$, with $\widehat{p}_k$ being the standard estimate of the marginal probability for state s_k, $\boldsymbol{I}$ and $\boldsymbol{J}$ are the identity and counteridentity matrices of order $n+1$, respectively, and $\boldsymbol{D}$ is a pairwise distance matrix for the elements in the set $\mathcal{S}$ considering a specific distance between ordinal states, $d(\cdot, \cdot)$. If `normalize = TRUE`, then the normalized estimate is computed, namely $\frac{\widehat{asym}_d}{max_{s_i, s_j \in \mathcal{S}} d(s_i, s_j)}$.

Value

The estimated asymmetry.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). “Distance-based analysis of ordinal data and ordinal time series.” *Journal of the American Statistical Association*.

Examples

```
estimated_asymmetry <- ordinal_asymmetry(series = AustrianWages$data[[100]],
states = 0 : 5) # Computing the asymmetry estimate
# for one series in dataset AustrianWages using the block distance
```

---

ordinal_cohens_kappa   *Computes the estimated ordinal Cohen's kappa of an ordinal time series*

Description

`ordinal_cohens_kappa` computes the estimated ordinal Cohen's kappa of an ordinal time series

Usage

```
ordinal_cohens_kappa(series, states, distance = "Block", lag = 1)
```

Arguments

`series`	An OTS.
`states`	A numerical vector containing the corresponding states.
`distance`	A function defining the underlying distance between states. The Hamming, block and Euclidean distances are already implemented by means of the arguments "Hamming", "Block" (default) and "Euclidean". Otherwise, a function taking as input two states must be provided.
`lag`	The considered lag.

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_n\}$ ($s_0 < s_1 < s_2 < \dots < s_n$), $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, the function computes the estimated ordinal Cohen's kappa given by $\widehat{\kappa}_d(l) = \frac{\widehat{disp}_d(X_t) - \widehat{E}[d(X_t, X_{t-l})]}{\widehat{disp}_d(X_t)}$, where $\widehat{disp}_d(X_t) = \frac{T}{T-1}\sum_{i,j=0}^{n} d\big(s_i, s_j\big)\widehat{p}_i\widehat{p}_j$ is the DIVC estimate of the dispersion, with $d(\cdot,\cdot)$ being a distance between ordinal states and $\widehat{p}_k$ being the standard estimate of the marginal probability for state s_k, and $\widehat{E}[d(X_t, X_{t-l})] = \frac{1}{T-l}\sum_{t=l+1}^{T} d(\overline{X}_t, \overline{X}_{t-l})$.

Value

The estimated ordinal Cohen's kappa.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
estimated_ock <- ordinal_cohens_kappa(series = AustrianWages$data[[100]],
states = 0 : 5) # Computing the estimated ordinal Cohen's kappa
# for one series in dataset AustrianWages using the block distance
```

`ordinal_dispersion_1` *Computes the standard estimated dispersion of an ordinal time series*

Description

`ordinal_dispersion_1` computes the standard estimated dispersion of an ordinal time series

Usage

```
ordinal_dispersion_1(series, states, distance = "Block", normalize = FALSE)
```

Arguments

`series`	An OTS.
`states`	A numerical vector containing the corresponding states.
`distance`	A function defining the underlying distance between states. The Hamming, block and Euclidean distances are already implemented by means of the arguments "Hamming", "Block" (default) and "Euclidean". Otherwise, a function taking as input two states must be provided.
`normalize`	Logical. If `normalize = FALSE` (default), the value of the standard estimated dispersion is returned. Otherwise, the function returns the normalized standard estimated dispersion.

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_n\}$ ($s_0 < s_1 < s_2 < \dots < s_n$), $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, the function computes the standard estimated dispersion given by $\widehat{disp}_{loc,d} = \frac{1}{T}\sum_{t=1}^{T} d\big(\overline{X}_t, \widehat{x}_{loc,d}\big)$, where $\widehat{x}_{loc,d}$ is the standard estimate of the location and $d(\cdot,\cdot)$ is a distance between ordinal states. If `normalize = TRUE`, then the normalized dispersion is computed, namely $\widehat{disp}_{loc,d}/\max_{s_i,s_j \in \mathcal{S}} d(s_i, s_j)$.

Value

The standard estimated dispersion.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
estimated_dispersion <- ordinal_dispersion_1(series = AustrianWages$data[[100]],
states = 0 : 5) # Computing the standard dispersion estimate
# for one series in dataset AustrianWages using the block distance
```

`ordinal_dispersion_2` *Computes the estimated dispersion of an ordinal time series according to the approach based on the diversity coefficient (DIVC)*

Description

`ordinal_dispersion_2` computes the estimated dispersion of an ordinal time series according to the approach based on the diversity coefficient

Usage

```
ordinal_dispersion_2(series, states, distance = "Block", normalize = FALSE)
```

Arguments

`series`	An OTS.
`states`	A numerical vector containing the corresponding states.
`distance`	A function defining the underlying distance between states. The Hamming, block and Euclidean distances are already implemented by means of the arguments "Hamming", "Block" (default) and "Euclidean". Otherwise, a function taking as input two states must be provided.
`normalize`	Logical. If `normalize = FALSE` (default), the value of the estimated dispersion is returned. Otherwise, the function returns the normalized estimated dispersion.

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_n\}$ ($s_0 < s_1 < s_2 < \dots < s_n$), $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, the function computes the DIVC estimated dispersion given by $\widehat{disp}_d = \frac{T}{T-1} \sum_{i,j=0}^{n} d\big(s_i, s_j\big)\widehat{p}_i\widehat{p}_j$, where $d(\cdot,\cdot)$ is a distance between ordinal states and $\widehat{p}_k$ is the standard estimate of the marginal probability for state s_k. If `normalize = TRUE`, and `distance = "Block"` or `distance = "Euclidean"`, then the normalized versions are computed, that is, the corresponding estimates are divided by the factors $2/m$ or $2/m^2$, respectively.

Value

The estimated dispersion according to the approach based on the diversity coefficient.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). “Distance-based analysis of ordinal data and ordinal time series.” *Journal of the American Statistical Association*.

Examples

```
estimated_dispersion <- ordinal_dispersion_2(series = AustrianWages$data[[100]],
states = 0 : 5) # Computing the DIVC dispersion estimate
# for one series in dataset AustrianWages using the block distance
```

`ordinal_location_1` *Computes the standard estimated location of an ordinal time series*

Description

`ordinal_location_1` computes the standard estimated location of an ordinal time series

Usage

```
ordinal_location_1(series, states, distance = "Block", normalize = FALSE)
```

Arguments

`series`	An OTS.
`states`	A numerical vector containing the corresponding states.
`distance`	A function defining the underlying distance between states. The Hamming, block and Euclidean distances are already implemented by means of the arguments "Hamming", "Block" (default) and "Euclidean". Otherwise, a function taking as input two states must be provided.
`normalize`	Logical. If `normalize = FALSE` (default), the value of the standard estimated location is returned. Otherwise, the function returns the normalized standard estimated location.

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_n\}$ $(s_0 < s_1 < s_2 < \dots < s_n)$, $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, the function computes the standard estimated location given by $\widehat{x}_{loc,d}$ =argmin$_{s \in \mathcal{S}} \frac{1}{T} \sum_{t=1}^{T} d\big(\overline{X}_t, s\big)$, where $d(\cdot, \cdot)$ is a distance between ordinal states.

Value

The standard estimated location.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
estimated_location <- ordinal_location_1(series = AustrianWages$data[[100]],
states = 0 : 5) # Computing the standard location estimate
# for one series in dataset AustrianWages using the block distance
```

`ordinal_location_2` *Computes the estimated location of an ordinal time series with respect to the lowest category*

Description

`ordinal_location_2` computes the estimated location of an ordinal time series with respect to the lowest category

Usage

```
ordinal_location_2(series, states, distance = "Block", normalize = FALSE)
```

Arguments

`series`	An OTS.
`states`	A numerical vector containing the corresponding states.
`distance`	A function defining the underlying distance between states. The Hamming, block and Euclidean distances are already implemented by means of the arguments "Hamming", "Block" (default) and "Euclidean". Otherwise, a function taking as input two states must be provided.
`normalize`	Logical. If `normalize = FALSE` (default), the value of the standard estimated location is returned. Otherwise, the function returns the normalized standard estimated location.

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, s_2, \ldots, s_n\}$ ($s_0 < s_1 < s_2 < \ldots < s_n$), $\overline{X}_t = \{\overline{X}_1, \ldots, \overline{X}_T\}$, the function computes the estimated location with respect to the lowest state, that is, the state s_j such that $a_j = d(s_j, s_0)$ is the closest to $\frac{1}{T}\sum_{t=1}^{T} d\big(\overline{X}_t, s_0\big)$ is determined, where $d(\cdot, \cdot)$ is a distance between ordinal states.

Value

The estimated location with respect to the lowest category.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
estimated_location <- ordinal_location_2(series = AustrianWages$data[[100]],
states = 0 : 5) # Computing the location estimate
# with respect to the lowest state for one series in dataset AustrianWages
```

`ordinal_skewness` *Computes the estimated skewness of an ordinal time series*

Description

`ordinal_skewness` computes the estimated skewness of an ordinal time series

Usage

```
ordinal_skewness(series, states, distance = "Block", normalize = FALSE)
```

Arguments

`series`	An OTS.
`states`	A numerical vector containing the corresponding states.
`distance`	A function defining the underlying distance between states. The Hamming, block and Euclidean distances are already implemented by means of the arguments "Hamming", "Block" (default) and "Euclidean". Otherwise, a function taking as input two states must be provided.
`normalize`	Logical. If `normalize` = `FALSE` (default), the value of the estimated skewness is returned. Otherwise, the function returns the normalized estimated skewness.

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, s_2, \dots, s_n\}$ ($s_0 < s_1 < s_2 < \dots < s_n$), $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, the function computes the estimated skewness given by $\widehat{skew}_d = \sum_{i=0}^{n} \big(d(s_i, s_n) - d(s_i, s_0)\big)\widehat{p}_i$, where $d(\cdot, \cdot)$ is a distance between ordinal states and $\widehat{p}_k$ is the standard estimate of the marginal probability for state s_k computed from the realization $\overline{X}_t$.

Value

The estimated skewness.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
estimated_skewness <- ordinal_skewness(series = AustrianWages$data[[100]],
states = 0 : 5) # Computing the skewness estimate
# for one series in dataset AustrianWages using the block distance
```

---

ots_plot — *Constructs an ordinal time series plot*

Description

`ots_plot` constructs an ordinal time series plot

Usage

```
ots_plot(series, states, title = "Time series plot", labels = NULL)
```

Arguments

`series`	An OTS.
`states`	A numerical vector containing the corresponding states.
`title`	The title of the graph.
`labels`	The labels of the graph.

Details

Constructs an ordinal time series plot for a given OTS.

Value

The ordinal time series plot.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2018). *An introduction to discrete-valued time series*. John Wiley and Sons.

Examples

```
ordinal_time_series_plot <- ots_plot(series = AustrianWages$data[[100]],,
states = 0 : 5) # Constructs an ordinal
# time series plot for one series in
# dataset AustrianWages
```

---

plot_ordinal_cohens_kappa

Constructs a serial dependence plot based on the ordinal Cohen's kappa considering the block distance

---

Description

`plot_ordinal_cohens_kappa` constructs a serial dependence plot of an ordinal time series based on the ordinal Cohen's kappa considering the block distance

Usage

```
plot_ordinal_cohens_kappa(
  series,
  states,
  max_lag = 10,
  alpha = 0.05,
  plot = TRUE,
  title = "Serial dependence plot",
  bar_width = 0.12,
  ...
)
```

Arguments

`series`	An OTS.
`states`	A numerical vector containing the corresponding states.
`max_lag`	The maximum lag represented in the plot (default is 10).
`alpha`	The significance level for the corresponding hypothesis test (default is 0.05).
`plot`	Logical. If `plot = TRUE` (default), returns the serial dependence plot. Otherwise, returns a list with the values of the ordinal Cohens's kappa, the critical value and the corresponding p-values.
`title`	The title of the graph.
`bar_width`	The width of the corresponding bars.
`...`	Additional parameters for the function.

Details

Constructs a serial dependence plot based on the ordinal Cohens's kappa, $\widehat{\kappa}_d(l)$, for several lags, where d is the block distance between ordinal states, that is, $d(s_i, s_j) = |i - j|$ for two states s_i and s_j. A dashed lined is incorporated indicating the critical value of the test based on the following asymptotic approximation (under the i.i.d. assumption):

$$\sqrt{\frac{T\widehat{disp}_d^{\,2}}{4\sum_{k,l=0}^{n-1}(\widehat{f}_{min\{k,l\}} - \widehat{f}_k\widehat{f}_l)^2}}\left(\widehat{\kappa}_d(l) + \frac{1}{T}\right) \sim N\big(0, 1\big),$$

where T is the series length, $\widehat{f}_k$ is the estimated cumulative probability for state s_k and $\widehat{disp}_d$ is the DIVC estimate of the dispersion.

Value

If `plot = TRUE` (default), returns the serial dependence plot based on the ordinal Cohens's kappa. Otherwise, the function returns a list with the values of the ordinal Cohens's kappa, the critical value and the corresponding p-values.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
plot_ock <- plot_ordinal_cohens_kappa(series = AustrianWages$data[[100]],
states = 0 : 5, max_lag = 3) # Representing
# the serial dependence plot
list_ck <- plot_ordinal_cohens_kappa(series = AustrianWages$data[[100]],
states = 0 : 5, max_lag = 3, plot = FALSE) # Obtaining
# the values of the ordinal Cohens's kappa, the critical value and the p-values
```

SyntheticData1 *SyntheticData1*

Description

Synthetic dataset containing 80 OTS generated from four different generating processes.

Usage

```
data(SyntheticData1)
```

Format

A `list` with two elements, which are:

- `data` A list with 80 OTS.
- `classes` A numeric vector indicating the corresponding classes associated with the elements in `data`.

Details

Each element in `data` is a 6-state OTS of length 600. Series 1-20, 21-40, 41-60 and 61-80 were generated from binomial AR(p) processes with different coefficients (see Scenario 1 in López-Oriona et al. (2023)). Therefore, there are 4 different classes in the dataset.

References

López-Oriona Á, Weiß C, Vilar JA (2023). "Fuzzy clustering of ordinal time series based on two novel distances with financial applications." *Manuscript submitted for publication*, 000-000.

`SyntheticData2` *SyntheticData2*

Description

Synthetic dataset containing 80 OTS generated from four different generating processes.

Usage

```
data(SyntheticData2)
```

Format

A `list` with two elements, which are:

- `data` A list with 80 OTS.
- `classes` A numeric vector indicating the corresponding classes associated with the elements in `data`.

Details

Each element in `data` is a 6-state OTS of length 600. Series 1-20, 21-40, 41-60 and 61-80 were generated from binomial INARCH(p) processes with different coefficients (see Scenario 2 in López-Oriona et al. (2023)). Therefore, there are 4 different classes in the dataset.

References

López-Oriona Á, Weiß C, Vilar JA (2023). "Fuzzy clustering of ordinal time series based on two novel distances with financial applications." *Manuscript submitted for publication*, 000-000.

`SyntheticData3` *SyntheticData3*

Description

Synthetic dataset containing 80 OTS generated from four different generating processes.

Usage

```
data(SyntheticData3)
```

Format

A `list` with two elements, which are:

- `data` A list with 80 OTS.
- `classes` A numeric vector indicating the corresponding classes associated with the elements in `data`.

Details

Each element in `data` is a 6-state OTS of length 600. Series 1-20, 21-40, 41-60 and 61-80 were generated from ordinal logit AR(1) processes with different coefficients (see Scenario 3 in López-Oriona et al. (2023)). Therefore, there are 4 different classes in the dataset.

References

López-Oriona Á, Weiß C, Vilar JA (2023). "Fuzzy clustering of ordinal time series based on two novel distances with financial applications." *Manuscript submitted for publication*, 000-000.

`test_ordinal_asymmetry` *Performs the hypothesis test associated with the ordinal asymmetry for the block distance*

Description

`test_ordinal_asymmetry` performs the hypothesis test associated with the ordinal asymmetry for the block distance

Usage

```
test_ordinal_asymmetry(
  series,
  states,
  true_asymmetry,
  alpha = 0.05,
  temporal = TRUE,
  max_lag = 1
)
```

Arguments

series	An OTS (numerical vector with integers).
states	A numeric vector containing the corresponding states.
true_asymmetry	The value for the true asymmetry.
alpha	The significance level (default is 0.05).
temporal	Logical. If temporal = TRUE (default), the test is performed for a time series. Otherwise, the test is performed for i.i.d. data.
max_lag	If temporal = TRUE, the maximum considered lag to compute the estimates related to the cumulative joint probabilities.

Details

If temporal = TRUE (default), the function performs the hypothesis test based on the ordinal asymmetry relying on Theorem 7.1.1 in Weiß (2019). Otherwise, the test based on Theorem 4.1 in Weiß (2019) is carried out.

Value

The results of the hypothesis test.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
results_test <- test_ordinal_asymmetry(AustrianWages$data[[100]],
states = 0 : 5, true_asymmetry = 2) # Performing the hypothesis test associated with the
# ordinal asymmetry for one OTS in dataset AustrianWages
```

test_ordinal_dispersion — *Performs the hypothesis test associated with the ordinal dispersion for the block distance*

Description

`test_ordinal_dispersion` performs the hypothesis test associated with the ordinal dispersion for the block distance

Usage

```
test_ordinal_dispersion(
  series,
  states,
  true_dispersion,
  alpha = 0.05,
  temporal = TRUE,
  max_lag = 1
)
```

Arguments

`series`	An OTS (numerical vector with integers).
`states`	A numeric vector containing the corresponding states.
`true_dispersion`	The value for the true dispersion.
`alpha`	The significance level (default is 0.05).
`temporal`	Logical. If `temporal = TRUE` (default), the test is performed for a time series. Otherwise, the test is performed for i.i.d. data.
`max_lag`	If `temporal = TRUE`, the maximum considered lag to compute the estimates related to the cumulative joint probabilities.

Details

If `temporal = TRUE` (default), the function performs the hypothesis test based on the ordinal dispersion relying on Theorem 7.1.1 in Weiß (2019). Otherwise, the test based on Theorem 4.1 in Weiß (2019) is carried out.

Value

The results of the hypothesis test.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). “Distance-based analysis of ordinal data and ordinal time series.” *Journal of the American Statistical Association*.

Examples

```
results_test <- test_ordinal_dispersion(AustrianWages$data[[100]],
states = 0 : 5, true_dispersion = 2) # Performing the hypothesis test associated with the
# ordinal dispersion for one OTS in dataset AustrianWages
```

---

test_ordinal_skewness *Performs the hypothesis test associated with the ordinal skewness for the block distance*

Description

`test_ordinal_skewness` performs the hypothesis test associated with the ordinal skewness for the block distance

Usage

```
test_ordinal_skewness(
  series,
  states,
  true_skewness,
  alpha = 0.05,
  temporal = TRUE,
  max_lag = 1
)
```

Arguments

`series`	An OTS (numerical vector with integers).
`states`	A numeric vector containing the corresponding states.
`true_skewness`	The value for the true skewness.
`alpha`	The significance level (default is 0.05).
`temporal`	Logical. If `temporal = TRUE` (default), the test is performed for a time series. Otherwise, the test is performed for i.i.d. data.
`max_lag`	If `temporal = TRUE`, the maximum considered lag to compute the estimates related to the cumulative joint probabilities.

Details

If `temporal = TRUE` (default), the function performs the hypothesis test based on the ordinal skewness relying on Theorem 7.1.1 in Weiß (2019). Otherwise, the test based on Theorem 4.1 in Weiß (2019) is carried out.

Value

The results of the hypothesis test.

Author(s)

Ángel López-Oriona, José A. Vilar

References

Weiß CH (2019). "Distance-based analysis of ordinal data and ordinal time series." *Journal of the American Statistical Association*.

Examples

```
results_test <- test_ordinal_skewness(AustrianWages$data[[100]],
states = 0 : 5, true_skewness = 2) # Performing the hypothesis test associated with the
# ordinal skewness for one OTS in dataset AustrianWages
```

`total_c_correlation` *Computes the total cumulative correlation of an ordinal time series*

Description

`total_c_correlation` returns the value of the total cumulative correlation for an ordinal time series

Usage

```
total_c_correlation(series, lag = 1, states, features = FALSE)
```

Arguments

`series`	An OTS.
`lag`	The considered lag (default is 1).
`states`	A numerical vector containing the corresponding states.
`features`	Logical. If `features` = `FALSE` (default), the value of the total cumulative correlation is returned. Otherwise, the function returns a matrix with the individual components of the total cumulative correlation

Details

Given an OTS of length T with range $\mathcal{S} = \{s_0, s_1, \dots, s_n\}$, $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, and the cumulative binarized time series, which is defined as $\overline{\boldsymbol{Y}}_t = \{\overline{\boldsymbol{Y}}_1, \dots, \overline{\boldsymbol{Y}}_T\}$, with $\overline{\boldsymbol{Y}}_k = (\overline{Y}_{k,0}, \dots, \overline{Y}_{k,n-1})^\top$ such that $\overline{Y}_{k,i} = 1$ if $\overline{X}_k \leq s_i$ ($k = 1, \dots, T, , i = 0, \dots, n-1$), the function computes the estimated average $\widehat{\Psi}(l)^c = \frac{1}{n^2} \sum_{i,j=0}^{n-1} \widehat{\psi}_{ij}(l)^2$, where $\widehat{\psi}_{ij}(l)$ is the estimated correlation $\widehat{Corr}(Y_{t,i}, Y_{t-l,j})$, $i, j = 0, 1, \dots, n-1$. If `features` = `TRUE`, the function returns a matrix whose components are the quantities $\widehat{\psi}_{ij}(l)$, $i, j = 0, 1, \dots, n-1$.

Value

If `features = FALSE` (default), returns the value of the total cumulative correlation. Otherwise, the function returns a matrix of features, i.e., the matrix contains the features employed to compute the total cumulative correlation.

Author(s)

Ángel López-Oriona, José A. Vilar

Examples

```
tcc <- total_c_correlation(series = AustrianWages$data[[100]],
states = 0 : 5) # Computing the total cumulative correlation
# for one of the series in dataset AustrianWages
feature_matrix <- total_c_correlation(series = AustrianWages$data[[100]],
states = 0 : 5) # Computing the corresponding matrix of features
```

`total_mixed_c_correlation_1`
Computes the total mixed cumulative linear correlation (TMCLC) between an ordinal and a real-valued time series

---

Description

`total_mixed_c_correlation_1` returns the TMCLC between an ordinal and a real-valued time series

Usage

```
total_mixed_c_correlation_1(
  o_series,
  n_series,
  lag = 1,
  states,
  features = FALSE
)
```

Arguments

`o_series`	An OTS.
`n_series`	A real-valued time series.
`lag`	The considered lag (default is 1).
`states`	A numerical vector containing the corresponding states.
`features`	Logical. If `features = FALSE` (default), the value of the TMCLC is returned. Otherwise, the function returns a vector with the individual components of the TMCLC.

Details

Given a OTS of length T with range $\mathcal{S} = \{s_0, s_1, \ldots, s_n\}$, $\overline{X}_t = \{\overline{X}_1, \ldots, \overline{X}_T\}$, and the cumulative binarized time series, which is defined as $\overline{\boldsymbol{Y}}_t = \{\overline{\boldsymbol{Y}}_1, \ldots, \overline{\boldsymbol{Y}}_T\}$, with $\overline{\boldsymbol{Y}}_k = (\overline{Y}_{k,0}, \ldots, \overline{Y}_{k,n-1})^\top$ such that $\overline{Y}_{k,i} = 1$ if $\overline{X}_k \leq s_i$ ($k = 1, \ldots, T, i = 0, \ldots, n-1$), the function computes the estimated TMCLC given by

$$\widehat{\Psi}_1^m(l) = \frac{1}{n} \sum_{i=0}^{n-1} \widehat{\psi}_i^*(l)^2,$$

where $\widehat{\psi}_i^*(l) = \widehat{Corr}(Y_{t,i}, Z_{t-l})$, with $\overline{Z}_t = \{\overline{Z}_1, \ldots, \overline{Z}_T\}$ being a T-length real-valued time series. If features = TRUE, the function returns a vector whose components are the quantities $\widehat{\psi}_i(l)$, $i = 0, 1, \ldots, n-1$.

Value

If features = FALSE (default), returns the value of the TMCLC. Otherwise, the function returns a vector of features, i.e., the vector contains the features employed to compute the TMCLC.

Author(s)

Ángel López-Oriona, José A. Vilar

Examples

```
tmclc <- total_mixed_c_correlation_1(o_series = SyntheticData1$data[[1]],
n_series = rnorm(600), states = 0 : 5) # Computing the TMCLC
# between the first series in dataset SyntheticData1 and white noise
feature_vector <- total_mixed_c_correlation_1(o_series = SyntheticData1$data[[1]],
n_series = rnorm(600), states = 0 : 5, features = TRUE) # Computing the corresponding
# vector of features
```

---

total_mixed_c_correlation_2

Computes the total mixed cumulative quantile correlation (TMCQC) between an ordinal and a real-valued time series

---

Description

total_mixed_c_correlation_2 returns the TMCQC between an ordinal and a real-valued time series

Usage

```
total_mixed_c_correlation_2(
  o_series,
  n_series,
  lag = 1,
  states,
  features = FALSE
)
```

Arguments

`o_series`	An OTS.
`n_series`	A real-valued time series.
`lag`	The considered lag (default is 1).
`states`	A numerical vector containing the corresponding states.
`features`	Logical. If `features = FALSE` (default), the value of the TMCLC is returned. Otherwise, the function returns a vector with the individual components of the TMCQC.

Details

Given a OTS of length T with range $\mathcal{S} = \{s_0, s_1, \dots, s_n\}$, $\overline{X}_t = \{\overline{X}_1, \dots, \overline{X}_T\}$, and the cumulative binarized time series, which is defined as $\overline{\boldsymbol{Y}}_t = \{\overline{\boldsymbol{Y}}_1, \dots, \overline{\boldsymbol{Y}}_T\}$, with $\overline{\boldsymbol{Y}}_k = (\overline{Y}_{k,0}, \dots, \overline{Y}_{k,n-1})^\top$ such that $\overline{Y}_{k,i} = 1$ if $\overline{X}_k \leq s_i$ ($k = 1, \dots, T, i = 0, \dots, n-1$), the function computes the estimated TMCQC given by

$$\widehat{\Psi}_2^m(l) = \frac{1}{n}\sum_{i=0}^{n-1}\int_0^1 \widehat{\psi}_i^\rho(l)^2 d\rho,$$

where $\widehat{\psi}_i^\rho(l) = \widehat{Corr}\big(Y_{t,i}, I(Z_{t-l} \leq q_{Z_t}(\rho))\big)$, with $\overline{Z}_t = \{\overline{Z}_1, \dots, \overline{Z}_T\}$ being a T-length real-valued time series, $\rho \in (0,1)$ a probability level, $I(\cdot)$ the indicator function and q_{Z_t} the quantile function of the corresponding real-valued process. If `features = TRUE`, the function returns a vector whose components are the quantities $\int_0^1 \widehat{\psi}_i^\rho(l)^2 d\rho$, $i = 0, 1, \dots, n-1$.

Value

If `features = FALSE` (default), returns the value of the TMCQC. Otherwise, the function returns a vector of features, i.e., the vector contains the features employed to compute the TMCLC.

Author(s)

Ángel López-Oriona, José A. Vilar

Examples

```
tmclc <- total_mixed_c_correlation_2(o_series = SyntheticData1$data[[1]],
n_series = rnorm(600), states = 0 : 5) # Computing the TMCQC
# between the first series in dataset SyntheticData1 and white noise
feature_vector <- total_mixed_c_correlation_2(o_series = SyntheticData1$data[[1]],
n_series = rnorm(600), states = 0 : 5, features = TRUE) # Computing the corresponding
# vector of features
```

Index

∗ **datasets**
AustrianWages, 2
CreditRatings, 8
SyntheticData1, 25
SyntheticData2, 26
SyntheticData3, 27

AustrianWages, 2

binarization, 3

c_binarization, 9
c_conditional_probabilities, 10
c_joint_probabilities, 11
c_marginal_probabilities, 12
ci_ordinal_asymmetry, 4
ci_ordinal_dispersion, 5
ci_ordinal_skewness, 6
conditional_probabilities, 7
CreditRatings, 8

index_ordinal_variation, 13

joint_probabilities, 14

marginal_probabilities, 15

ordinal_asymmetry, 16
ordinal_cohens_kappa, 17
ordinal_dispersion_1, 18
ordinal_dispersion_2, 19
ordinal_location_1, 20
ordinal_location_2, 21
ordinal_skewness, 22
ots_plot, 23

plot_ordinal_cohens_kappa, 24

SyntheticData1, 25
SyntheticData2, 26
SyntheticData3, 27

test_ordinal_asymmetry, 27
test_ordinal_dispersion, 29
test_ordinal_skewness, 30
total_c_correlation, 31
total_mixed_c_correlation_1, 32
total_mixed_c_correlation_2, 33