Package ‘ordinalpattern’

March 6, 2026

Type Package

Title Tests Based on Ordinal Patterns

Version 0.2.8

Date 2026-03-06

Author Alexander Duerre [aut],
Svenja Fischer [ctb],
Alexander Schnurr [aut],
Angelika Silbernagel [aut, cre]

Depends gtools,mvtnorm

Maintainer Angelika Silbernagel <silbernagel@hsu-hh.de>

Description Ordinal patterns describe the dynamics of a time series by looking at the ranks of subsequent observations. By comparing ordinal patterns of two times series, Schnurr (2014) <doi:10.1007/s00362-013-0536-8> defines a robust and non-parametric dependence measure: the ordinal pattern coefficient. Functions to calculate this and a method to detect a change in the pattern coefficient proposed in Schnurr and Dehling (2017) <doi:10.1080/01621459.2016.1164706> are provided. Furthermore, the package contains a function for calculating the ordinal pattern frequencies. Generalized ordinal patterns as proposed by Schnurr and Fischer (2022) <doi:10.1016/j.csda.2022.107472> are also considered.

License GPL-2 | GPL-3

ByteCompile true

Encoding UTF-8

NeedsCompilation yes

RoxygenNote 7.3.2

Repository CRAN

Date/Publication 2026-03-06 13:00:03 UTC

Contents

countingpatterns *Empirical Ordinal Pattern Distribution*

Description

Calculates the empirical ordinal pattern distribution.

Usage

```
countingpatterns(tsx,d=3,block=FALSE,first=TRUE,tiesmethod=c("random","first"),
generalized=FALSE)

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

Arguments

`tsx`	numeric vector representing the univariate time series.
`d`	numeric value determining the length of the ordinal pattern.
`block`	logical value determining whether patterns are calculated on disjoint blocks or overlapping blocks.
`first`	logical value indicating which observartions are dropped if `block == TRUE` and the time series length is no multiple of `d`.
`tiesmethod`	character string specifying how ties, that is equal values, are treated if `generalized == FALSE`, see 'Details'.
`generalized`	logical value determining whether classical ordinal patterns or their generalization with regard to ties are considered, see 'Details'.
`x`	object of class `"patterncounts"`, which is the output of `countingpatterns`.
`...`	further arguments passed to the internal plotting function.

Details

Ordinal patterns, which are defined as sequences of ranks of `d` subsequent observations, are a useful tool to describe the dependence within or between time series. That sequences of subseqent observations can either move one observation per time or a whole block of `d` observations. The former is preferred since it uses more information. If one chooses the later, one has to decide whether the first or the last observations are removed in case that the time series length is no multiple of `d`. With regard to equal values within a window of consecutive observations (ties), the argument `tiesmethod` determines the approach for computing the respective ordinal patterns. The "`first`" method is in favor of increasing patterns, whereas the default "`random`" puts the equal values in random order.

Beside the classical ordinal patterns, one can also consider the generalized version proposed by Schnurr and Fischer (2022), where the ordinal information of stagnation in the case of ties is also included by taking into account a larger set of patterns.

Value

Object of class "patterncounts" containing the following values:

patterncounts	absolute frequencies of ordinal patterns.
allpatterns	list of all ordinal patterns considered.
d	length of the ordinal pattern.
generalized	logical value determining whether classical ordinal patterns or their generalization with regard to ties are considered.
tiesmethod	character string specifying how ties are treated.
block	logical value determining whether patterns are calculated on disjoint blocks or overlapping blocks.

Author(s)

Angelika Silbernagel

References

Schnurr, A. (2014): An ordinal pattern approach to detect and to model leverage effects and dependence structures between financial time series, *Statistical Papers*, vol. 55, 919–931.

Schnurr, A., Dehling, H. (2017): Testing for Structural Breaks via Ordinal Pattern Dependence, *Journal of the American Statistical Association*, vol. 112, 706–720.

Schnurr, A., Fischer, S. (2022): Generalized ordinal patterns allowing for ties and their application in hydrology, *Computational Statistics & Data Analysis*, vol 171, 107472.

Examples

```
set.seed(1066)
countingpatterns(rnorm(100))
countingpatterns(rpois(100,1), generalized=TRUE)
```

---

patternchange *Changepoint Detection Using Ordinal Patterns*

Description

Test for a change in the dependence structure of two time series using ordinal patterns

Usage

```
patternchange(tsx,tsy,d=3,conf.level,weight=TRUE,weightfun=NULL,bn=log(length(tsx)),
    kernel=function(x){return(max(0,1-abs(x)))})

## S3 method for class 'change'
plot(x, ...)
```

Arguments

`tsx`	numeric vector of first univariate time series.
`tsy`	numeric vector of second univariate time series.
`d`	numeric value determining the length of ordinal pattern.
`conf.level`	numerical value indicating the confidence level of the test.
`weight`	logical value indicating whether one uses weights of the L1 norm or the empirical probability of identical patterns; see details.
`weightfun`	function which defines the weights given the L1 norm between the patterns if `weight=TRUE`. If no weight-function is given, the canonical weight function is used; see details.
`bn`	numerical value determining the bandwidth of the kernel estimator used to estimate the long run variance.
`kernel`	kernel function for estimating the long run variance.
`x`	object of class `"change"`
`...`	further arguments passed to the internal plotting function (`plot`).

Details

Given two timeseries `tsx` and `tsy` a cusum type statistic tests whether there is a change in the patter dependence or not. The test is based on a comparison of patterns of length `d` in `tsx` and `tsy`. One can either choose the number of identical patterns (`weight=FALSE`) or a metric that is defined by the `weightfun` argument to measure the difference between patterns (`weight=TRUE`). If no (`weightfun`) is given, the canonical weightfunction is used, which equals 1 if patterns are identical and 0 if the L1 norm of their difference attains the maximal possible value. The value is linear interpolated in between.

The procedure depends on an estimate of the long run variance. Here a kernel estimator is used. A kernel function and a bandwidth can be set using the arguments `kernel` and `bn`. If none of them is given, the bartlett kernel with a bandwidth of `log(n)`, where `n` equals the length of the timeseries, is used.

Value

Object with classes `"change"` and `"htest"` containing the following values:

`statistic`	the value of the test statistic. Under the null the test statistic follows asymptotically a Kolmogorov Smirnov distribution.
`p.value`	the p-value of the test.
`estimate`	the estimated time of change.
`null.value`	the jump height of the at most one change point model, which is under the null hypothesis always 0.
`alternative`	a character string describing the alternative hypothesis.
`method`	a characters string describing the test.
`trajectory`	the cumulative sum on which the tests are based on. Could be used for additional plots.

Author(s)

Alexander Dürre

References

Schnurr, A. (2014): An ordinal pattern approach to detect and to model leverage effects and dependence structures between financial time series, *Statistical Papers*, vol. 55, 919–931.

Schnurr, A., Dehling, H. (2017): Testing for Structural Breaks via Ordinal Pattern Dependence, *Journal of the American Statistical Association*, vol. 112, 706–720.

See Also

Estimation of the pattern dependence is provided by `patterndependence`.

Examples

```
set.seed(1066)
a1 <- cbind(rnorm(100),rnorm(100))
a2 <- rmvnorm(100,sigma=matrix(c(1,0.8,0.8,1),ncol=2))
A <- rbind(a1,a2)
testresult <- patternchange(A[,1],A[,2])
plot(testresult)
testresult
```

`patterndependence` *Ordinal Pattern Dependence*

Description

Calculates the ordinal pattern coefficient and related values.

Usage

```
patterndependence(tsx,tsy,d=3,block=FALSE,first=TRUE,
tiesmethod=c("random","first"),ordinalcor=c("standard","positive","negative"))

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

Arguments

`tsx`	numeric vector representing the first univariate time series.
`tsy`	numeric vector representing the second univariate time series.
`d`	numeric value determining the length of the ordinal pattern.

`block`	logical value determining whether patterns are calculated on disjoint blocks or overlapping blocks.
`first`	logical value indicating which observartions are dropped if `block == TRUE` and the time series length is no multiple of `d`.
`tiesmethod`	character string specifying how ties, that is equal values, are treated, see 'Details'.
`ordinalcor`	character string specifying which ordinal pattern coefficient is output, see 'Details'.
`x`	object of class `"pattern"`, which is the output of `patterndependence`.
`...`	further arguments passed to the internal plotting function.

Details

The standard ordinal pattern coefficient is a non-parametric and robust measure of dependence between two time series. It is based on ordinal patterns, which are defined as sequences of ranks of `d` subsequent observations. This sequences of subseqent observations can either move one observation per time or a whole block of `d` observations. The former is preferred since it uses more information. If one chooses the later, one has to decide whether the first or the last observations are removed in case that the time series length is no multiple of `d`. With regard to equal values within a window of consecutive observations (ties), the argument `tiesmethod` determines the approach for computing the respective ordinal patterns. The "`first`" method is in favor of increasing patterns, whereas the default "`random`" puts the equal values in random order.

Beside the default standard ordinal pattern coefficient, which range from -1 to 1, one can also look at the positive and negative ordinal pattern coefficient, which roughly measures whether there are unusual many identical or opposite patterns in the time series.

The plot function draws both time series and shows the six most frequent coinciding pattern with counts on the right. At the bottom, the location of these coinciding patterns is visualized.

Value

Object of class `"pattern"` containing the following values:

`patterncoef`	ordinal pattern coefficient.
`numbequal`	number of equal ordinal patterns.
`numbopposite`	number of opposite ordinal patterns.
`PatternXz`	number of ordinal patterns in first time series.
`PatternYz`	number or ordinal patterns in second time series.
`coding`	coding of the ordinal patterns, used in `PatternXz` and `PatternYz`.
`PatternX`	numeric vector representing the time series of patterns in `tsx`.
`PatternY`	numeric vector representing the time series of patterns in `tsy`.
`tsx`	numeric vector representing the first univariate time series.
`tsy`	numeric vector representing the second univariate time series.
`maxpat`	number representing the maximal pattern code.
`ordinalcor`	character string specifying the type of ordinal pattern coefficient.

tiesmethod	character string specifying how ties are treated.
block	logical value determining whether patterns are calculated on disjoint blocks or overlapping blocks.
d	length of the ordinal pattern.
tablesame	numeric vector representing the number of coinciding patterns, apportioned into different patterns.
tableopposite	numeric vector representing the number of reflected patterns, apportioned into different patterns.
indexsame	logic vector indicating whether patterns in both time series coincide.
indexopposite	logic vector indicating whether patterns in both time series are reflected.

Author(s)

Alexander Dürre, Angelika Silbernagel

References

Schnurr, A. (2014): An ordinal pattern approach to detect and to model leverage effects and dependence structures between financial time series, *Statistical Papers*, vol. 55, 919–931.

Schnurr, A., Dehling, H. (2017): Testing for Structural Breaks via Ordinal Pattern Dependence, *Journal of the American Statistical Association*, vol. 112, 706–720.

Examples

```
set.seed(1066)
patternobj <- patterndependence(rnorm(100),rnorm(100))
plot(patternobj)
```

Index