

Package ‘regsubseq’

February 20, 2015

Type Package

Title Detect and Test Regular Sequences and Subsequences

Version 0.12

Date 2014-03-06

Author Yanming Di

Maintainer Yanming Di<diy@stat.oregonstate.edu>

Description For a sequence of event occurrence times, we are interested in finding subsequences in it that are too “regular”. We define regular as being significantly different from a homogeneous Poisson process. The departure from the Poisson process is measured using a L1 distance. See Di and Perlman 2007 for more details.

License GPL-2

Depends R (>= 2.10)

NeedsCompilation no

Repository CRAN

Date/Publication 2014-03-09 19:18:52

R topics documented:

qtables	2
test.gaplin	2
test.lin	3

Index	6
--------------	----------

 qtables

Quantile Tables of the Linearity/Gap-Linearity Tests

Description

The data set provide quantile tables for the linearity/gap-linearity test statistics for $N=2, \dots, 50$ and $k=2, \dots, N$, for each N . These tables will be used to compute p-values corresponding to test statistics.

Usage

qtables

Format

R rda files. Within each quantile table, the first row indicates at which probability values the quantiles are computed.

 test.gaplin

Detect and Test Almost Gap-Linear Subsequences.

Description

`test.gaplin.t` find the most almost gap-linear length $k+1$ subsequence of a given sequence and compute the almost gap-linearity test statistic for this subsequence. `test.gaplin.p` compute the p-value corresponding to a computed test statistic. `test.gaplin` compute the test statistics and the p-values for subsequences of all lengths.

Usage

```
test.gaplin(Tn);
test.gaplin.t(Tn, k);
test.gaplin.p(t, n, k);
```

Arguments

Tn	A sequence of numbers. Currently, only support sequence of length less than 50.
k	The length of the subsequences for which we want to test for almost gap-linearity.
n	The length of the sequence for which we want to test for subsequence almost gap-linearity.
t	Test statistic computed for a length $k+1$ subsequence of a length $n+1$ sequence.

Details

Almost gap-linear means the spacings of a subsequence are almost in proportion to the spacings of the corresponding indices. For example, for $T_n=c(11, 14, (.), 20)$, the subsequence (11, 14, 20) is gap-linear, since the spacings (3, 6) is in proportion with the spacings of the corresponding indices (1, 2). Equivalently, almost gap-linearity can be measured by the distance between the standardized spacings of the subsequence and the standardized spacings of the corresponding indices. See Di and Perlman (2007) for more details.

Value

`test.gaplin.t` returns the most gap-linear length $k+1$ subsequence of the input sequence and corresponding almost gap-linearity test statistic. `test.gaplin.p` returns the p-value corresponding to the input test statistic `t`. `test.lin` has no return value, instead, a table containing the most almost gap-linear subsequences, corresponding test statistics and p-values will be outputted.

Author(s)

Yanming Di

References

Di and Perlman, 2007

See Also

[test.lin](#).

Examples

```
## A sequence representing arrival times of events.
Tn = c(13, 21, 24, 33, 40, 55, 59, 63, 72, 85, 87);

## Test for almost linearity.
t = test.gaplin.t(Tn, 4);
print(t$sub);
p = test.gaplin.p(t$t, 10, 4);
print(p);
test.gaplin(Tn);
```

test.lin

Detect and Test Almost Linear Subsequences.

Description

`test.lin.t` find the most almost-linear length $k+1$ subsequence of a given sequence and compute the almost-linearity test statistic for this subsequence. `test.lin.p` compute the p-value corresponding to a computed test statistic. `test.lin` compute the test statistics and the p-values for subsequences of all lengths.

Usage

```
test.lin(Tn);  
test.lin.t(Tn, k);  
test.lin.p(t, n, k);
```

Arguments

Tn	A sequence of numbers. Currently, only support sequences of length less than 50.
k	The length of the subsequences for which we want to test for almost-linearity.
n	The length of the sequence for which we want to test for subsequence almost-linearity.
t	Test statistic computed for a length k+1 subsequence of a length n+1 sequence.

Details

Almost-linear means the spacings of the sequence are almost equal, or the distance between the standardized spacings as a vector and $(1/k, \dots, 1/k)$ is too small. The p-value is computed by comparing the test statistic to a precomputed test statistic quantile table. See Di and Perlman (2007) for more details.

Value

`test.lin.t` returns the most linear length k+1 subsequence of the input sequence and corresponding almost-linearity test statistic. `test.lin.p` returns the p-value corresponding to the input test statistic `t`. `test.lin` has no return value, instead, a table containing the most almost linear subsequences, corresponding test statistics and p-values will be outputted.

Author(s)

Yanming Di

References

Di and Perlman, 2007

See Also

[test.gaplin](#).

Examples

```
## A sequence representing arrival times of events.  
Tn = c(13, 21, 24, 33, 40, 55, 59, 63, 72, 85, 87);  
  
## Test for almost linearity.  
t = test.lin.t(Tn, 4);  
print(t$sub);  
p = test.lin.p(t$t, 10, 4);
```

```
print(p);  
test.lin(Tn);
```

Index

*Topic **datasets**

qtables, 2

*Topic **htest**

test.gaplin, 2

test.lin, 3

q.testgaplin.n10 (qtables), 2
q.testgaplin.n11 (qtables), 2
q.testgaplin.n12 (qtables), 2
q.testgaplin.n13 (qtables), 2
q.testgaplin.n14 (qtables), 2
q.testgaplin.n15 (qtables), 2
q.testgaplin.n16 (qtables), 2
q.testgaplin.n17 (qtables), 2
q.testgaplin.n18 (qtables), 2
q.testgaplin.n19 (qtables), 2
q.testgaplin.n2 (qtables), 2
q.testgaplin.n20 (qtables), 2
q.testgaplin.n21 (qtables), 2
q.testgaplin.n22 (qtables), 2
q.testgaplin.n23 (qtables), 2
q.testgaplin.n24 (qtables), 2
q.testgaplin.n25 (qtables), 2
q.testgaplin.n26 (qtables), 2
q.testgaplin.n27 (qtables), 2
q.testgaplin.n28 (qtables), 2
q.testgaplin.n29 (qtables), 2
q.testgaplin.n3 (qtables), 2
q.testgaplin.n30 (qtables), 2
q.testgaplin.n31 (qtables), 2
q.testgaplin.n32 (qtables), 2
q.testgaplin.n33 (qtables), 2
q.testgaplin.n34 (qtables), 2
q.testgaplin.n35 (qtables), 2
q.testgaplin.n36 (qtables), 2
q.testgaplin.n37 (qtables), 2
q.testgaplin.n38 (qtables), 2
q.testgaplin.n39 (qtables), 2
q.testgaplin.n4 (qtables), 2
q.testgaplin.n40 (qtables), 2

q.testgaplin.n41 (qtables), 2
q.testgaplin.n42 (qtables), 2
q.testgaplin.n43 (qtables), 2
q.testgaplin.n44 (qtables), 2
q.testgaplin.n45 (qtables), 2
q.testgaplin.n46 (qtables), 2
q.testgaplin.n47 (qtables), 2
q.testgaplin.n48 (qtables), 2
q.testgaplin.n49 (qtables), 2
q.testgaplin.n5 (qtables), 2
q.testgaplin.n50 (qtables), 2
q.testgaplin.n6 (qtables), 2
q.testgaplin.n7 (qtables), 2
q.testgaplin.n8 (qtables), 2
q.testgaplin.n9 (qtables), 2
q.testlin.n10 (qtables), 2
q.testlin.n11 (qtables), 2
q.testlin.n12 (qtables), 2
q.testlin.n13 (qtables), 2
q.testlin.n14 (qtables), 2
q.testlin.n15 (qtables), 2
q.testlin.n16 (qtables), 2
q.testlin.n17 (qtables), 2
q.testlin.n18 (qtables), 2
q.testlin.n19 (qtables), 2
q.testlin.n2 (qtables), 2
q.testlin.n20 (qtables), 2
q.testlin.n21 (qtables), 2
q.testlin.n22 (qtables), 2
q.testlin.n23 (qtables), 2
q.testlin.n24 (qtables), 2
q.testlin.n25 (qtables), 2
q.testlin.n26 (qtables), 2
q.testlin.n27 (qtables), 2
q.testlin.n28 (qtables), 2
q.testlin.n29 (qtables), 2
q.testlin.n3 (qtables), 2
q.testlin.n30 (qtables), 2
q.testlin.n31 (qtables), 2

q.testlin.n32 (qtables), 2
q.testlin.n33 (qtables), 2
q.testlin.n34 (qtables), 2
q.testlin.n35 (qtables), 2
q.testlin.n36 (qtables), 2
q.testlin.n37 (qtables), 2
q.testlin.n38 (qtables), 2
q.testlin.n39 (qtables), 2
q.testlin.n4 (qtables), 2
q.testlin.n40 (qtables), 2
q.testlin.n41 (qtables), 2
q.testlin.n42 (qtables), 2
q.testlin.n43 (qtables), 2
q.testlin.n44 (qtables), 2
q.testlin.n45 (qtables), 2
q.testlin.n46 (qtables), 2
q.testlin.n47 (qtables), 2
q.testlin.n48 (qtables), 2
q.testlin.n49 (qtables), 2
q.testlin.n5 (qtables), 2
q.testlin.n50 (qtables), 2
q.testlin.n6 (qtables), 2
q.testlin.n7 (qtables), 2
q.testlin.n8 (qtables), 2
q.testlin.n9 (qtables), 2
qtables, 2

test.gaplin, 2, 4
test.lin, 3, 3