Package 'sawnuti'

December 13, 2021

Type Package

Title Comparing Sequences with Non-Uniform Time Intervals

Version 0.1.1

Date 2021-11-13

Description The SAWNUTI algorithm performs sequence comparison for finite sequences of discrete events with non-uniform time intervals. Further description of the algorithm can be found in the paper: A. Murph, A. Flynt, B. R. King (2021). Comparing finite sequences of discrete events with non-uniform time intervals, Sequential Analysis, 40(3), 291-313. <doi:10.1080/07474946.2021.1940491>.

License MIT + file LICENSE

RoxygenNote 6.0.1

NeedsCompilation no

Author Alexander Murph [aut, cre] (<https://orcid.org/0000-0001-7170-867X>)

Maintainer Alexander Murph <acm026@bucknell.edu>

Repository CRAN

Date/Publication 2021-12-13 14:10:02 UTC

R topics documented:

---

sawnuti	*Comparing Sequences with Non-Uniform Time Intervals*

Description

The SAWNUTI algorithm performs sequence comparison for finite sequences of discrete events with non-uniform time intervals. Further description of the algorithm can be found in the paper: A. Murph, A. Flynt, B. R. King (2021). Comparing finite sequences of discrete events with non-uniform time intervals, Sequential Analysis, 40(3), 291-313. <doi:10.1080/07474946.2021.1940491>.

Usage

```
sawnuti(string1, string2, times1, times2, global = T, alpha,
match_function, gap_penalty)
```

Arguments

`string1`	character. Character of representations of each element in the first sequence separated by a single white space.
`string2`	character. Character of representations of each element in the first sequence separated by a single white space.
`times1`	character. Numerical time values between each value in the first sequence. Should be written as a single character string, where each value is separated by a single white space.
`times2`	character. Numerical time values between each value in the second sequence. Should be written as a single character string, where each value is separated by a single white space.
`global`	logical. Logical representing whether a user wishes to perform a global or local alignment. T = global, F = local.
`alpha`	double. Time interval penalty bias. Weights the influence of time on the alignment calculation.
`match_function`	R function. Score given for alignment of particular values. Must be able to take in two values from the universe of possible events, and return a numerical score for the alignment of those two elements. Simplest is a constant score returned, can also be implemented as a matrix look-up.
`gap_penalty`	numerical. This implementation only allows for constant gap penalties.

Details

Zero-one scale of all time values in the entire dataset is assumed before calculation. See Examples for possible formatting of the match_function.

Value

Vector containing 'similarity score', 'alignment', and 'scoring matrix'.

PREPROCESSING ASSUMPTION

Zero-one scale of all time values in the entire dataset is assumed before calculation. Since this preprocessing step must be done with ALL possible time intervals, it cannot be done in this alignment which only takes in two particular observations.

References

A. Murph, A. Flynt, B. R. King (2021). Comparing finite sequences of discrete events with non-uniform time intervals, Sequential Analysis, 40(3), 291-313.

Examples

```
matchFunction = function(a,b){ifelse(a==b, 1, -1)}

sawnuti(string1="a b c", string2="d b c", times1="1 2 3",
times2="3 2 1", alpha = 1, match_function = matchFunction, gap_penalty = 1)
```

Index

sawnuti,