

Package ‘SeqAlignment’

June 12, 2021

Title Global sequence alignment for nucleotide sequences

Version 0.0.99

Description This R package has been developed in order to find the optimal global alignment between two nucleotide sequences, using both the Needleman-Wunsch (NW) and the FOGSAA algorithms.

Depends R (>= 3.6)

Imports stringi, methods

Suggests BiocStyle, knitr, rmarkdown, testthat

License GPL-2

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

VignetteBuilder knitr

NeedsCompilation no

Author Francesco Morettini [aut, cre]

Maintainer Francesco Morettini <francesco.morettini@mail.polimi.it>

R topics documented:

alignment_score	2
compare	2
compute_scores	3
fogsaa_alignment	3
max_score	4
node	5
nw_alignment	5

Index	7
--------------	----------

alignment_score	<i>Compute the alignment score.</i>
-----------------	-------------------------------------

Description

Compute the alignment score.

Usage

```
alignment_score(al1, al2, match, mismatch, gap)
```

Arguments

al1	Aligned sequence 1.
al2	Aligned sequence 2.
match	An integer which indicates the score for a match between the two sequences.
mismatch	An integer which indicates the score for a mismatch between the two sequences.
gap	An integer which indicates the score for a gap inserted in one of the two sequences

Value

The score of the alignment

Examples

```
alignment_score('ACGGTTGC', 'A-GCGT-C', 1, -1, -2)
```

compare	<i>Comparison of two nodes.</i>
---------	---------------------------------

Description

It compares two nodes based on their pointers P1 and P2 and checking the Tmax score

Usage

```
compare(new_vertex, current_vertex)
```

Arguments

new_vertex	A node
current_vertex	A node

Value

True if the nodes are equal in terms of P1 and P2 and the new node has an higher score, false in any other case

compute_scores	<i>Computation of present and future score of each node.</i>
----------------	--

Description

It computes both the present and future score of each new node, finally it sums the two to obtain the Fitness score which is used to identify the best node

Usage

```
compute_scores(new_vertex, current_vertex, match, mismatch, gap)
```

Arguments

new_vertex	The node for which we compute the scores
current_vertex	The previous node from which we retrieve the PrS score
match	An integer which indicates the score for a match between the two sequences.
mismatch	An integer which indicates the score for a mismatch between the two sequences.
gap	An integer which indicates the score for a gap inserted in one of the two sequences.

Value

The node in which all the attributes regarding the different scores are computed

fogsaa_alignment	<i>FOGSAA algorithm for global sequence alignment.</i>
------------------	--

Description

Computes the optimal global alignment of two sequences using the FOGSAA algorithm.

Usage

```
fogsaa_alignment(s1, s2, match, mismatch, gap)
```

Arguments

s1	A string representing sequence 1.
s2	A string representing sequence 2.
match	An integer which indicates the score for a match between the two sequences.
mismatch	An integer which indicates the score for a mismatch between the two sequences.
gap	An integer which indicates the score for a gap inserted in one of the two sequences.

Value

A list containing the alignment of s1 and s2

References

<https://pubmed.ncbi.nlm.nih.gov/23624407/>

Examples

```
fogsaa_alignment('ACGGTTGC', 'AGCGTC', 1, -1, -2)
```

max_score	<i>Max score for a node.</i>
-----------	------------------------------

Description

It computes the node which has the highest score among the three

Usage

```
max_score(vertex_11, vertex_10, vertex_01)
```

Arguments

vertex_11	A node originated from the current node incrementing both P1 and P2
vertex_10	A node originated from the current node incrementing only P1
vertex_01	A node originated from the current node incrementing only P2

Value

The node with the highest Fitness score

node	<i>A Reference Class to represent the nodes used in the FOGSAA algorithm.</i>
------	---

Description

The class node is used to represent objects in the FOGSAA algorithm, this is the data type on which the algorithm is implemented.

Fields

P1 It is the pointer to the characters in sequence 1.
 P2 It is the pointer to the characters in sequence 2.
 PrS Present score of the node.
 Fmin Minimum of the future score.
 Fmax Maximum of the future score.
 Tmin Minimum of the Fitted score computed as the sum of PrS and Fmin.
 Tmax Maximum of the Fitted score computed as the sum of PrS and Fmax.
 v1 Value of sequence 1 at which P1 is pointing.
 v2 Value of sequence 2 at which P2 is pointing.
 x1 Defined as $n - P2$ where n is the length of sequence 2.
 x2 Defined as $m - P1$ where m is the length of sequence 1.
 code Indicates how the instance was originated, 11 if incrementing both P1 and P2 from the previous node, 01 if only P2 is incremented and 10 if only P1 is incremented.

nw_alignment	<i>Needleman-Wunsch global sequence alignment algorithm.</i>
--------------	--

Description

Computes the global sequence alignment using the Needleman-Wunsh algorithm.

Usage

```
nw_alignment(s1, s2, match, mismatch, gap)
```

Arguments

s1	A string representing sequence 1.
s2	A string representing sequence 2.
match	An integer which indicates the score for a match between the two sequences.
mismatch	An integer which indicates the score for a mismatch between the two sequences.
gap	An integer which indicates the score for a gap inserted in one of the two sequences.

Value

A list containing the alignment of s1 and s2

References

<https://pubmed.ncbi.nlm.nih.gov/5420325/>

Examples

```
nw_alignment('ACGGTTGC','AGCGTC', 1, -1, -2)
```

Index

alignment_score, [2](#)

compare, [2](#)

compute_scores, [3](#)

fogsaa_alignment, [3](#)

max_score, [4](#)

node, [5](#)

nw_alignment, [5](#)