ALINHAMENTOS LOCAIS E SEMI-GLOBAIS

Katia Guimarães

Alinhamento Semi-global

Similar ao alinhamento global, mas ignora espaços nos extremos das seqüências.
Aplicação: Montagem de Fragmentos

Ex:

- - - - - - - - - CAGCACTTGGATTAGAC

TACCTGCGGCAGCG - TGG - - - - - - - - -

Teremos 6 matches, 2 mismatches e 1 space.

Alinhamento Semi-global

CAGCA - CTTGGATTCTCGG

- - - CAGCGTGG - - - - - - - - - (-19)

Nem sempre é o melhor alinhamento global entre as duas seqüências (-12):

CAGCACTTGGATTCTCGG

CAGC - - - - - G - T - - - - GG

Achando o alinhamento semi-global

Para não computar os espaços após o último caracter de *s*, ignoramos o sufixo de *t* que casa com aqueles espaços.

O que sobra é um alinhamento entre *s* e um prefixo de *t*:

Ex: ***t* = TGGATTCTCGG**

***s* = TGG - - - - - - - - -**

Achando o alinhamento semi-global

O que sobra é um alinhamento entre *s* e um prefixo de *t*:

Ex: ***t* = TGGATTCTCGG**

***s* = TGG - - - - - - - - -**

Logo, devemos tomar o maior valor na última linha de *M*, isto é:

$$\text{sim}(s, t) = \text{máx}_{j=1..n} \; M[m, j]$$

Achando o alinhamento semi-global

Considerando este maior valor na última linha de M:

$$\text{sim}(s, t) = \text{máx}_{j=1..n} \; M[m, j]$$

Para conseguir o alinhamento é só começar a busca numa célula (m, k) da matriz, onde k é tal que

$$\text{sim}(s, t) = M[m, k].$$

Achando o alinhamento semi-global

Por um argumento análogo, quando não queremos considerar o custo de espaços finais em *t*, tomamos o máximo na última coluna de M.

Combinando os dois casos, tomamos o valor máximo da matriz na borda:

última linha ou última coluna.

Achando o alinhamento semi-global

Para não computar os espaços iniciais em s, (equivalente a um alinhamento entre *s* e um sufixo de *t*), não penalizamos os gaps iniciais.

Ex: *t* = CAGCA - CTTGG

s = - - - CAGCGTGG

Como conseguir este efeito na tabela?

Achando o alinhamento semi-global

INICIALIZAÇÃO.

Para não penalizar por gaps no início das cadeias, ao invés de penalizar com (i . g) ou (j . g), inicializamos a primeira linha e a primeira coluna com zeros.

Achando o alinhamento semi-global

Em resumo:

Onde espaços são grátis	**Ação**
Remoção no Início	Inic. 1a. linha com 0's
Inserção no início	Inic. 1a. coluna com 0's
Remoção no Final	Máx. na última linha
Inserção no Final	Máx. na última coluna

Alinhamento Local

Dadas duas seqüências *s* e *t*, identificar o alinhamento de melhor *score* entre um substring de *s* e um substring de *t*.

O algoritmo será o mesmo que usamos para alinhamento global, com algumas alterações.

Alinhamentos Locais

Ainda teremos uma matriz $(m+1) \times (n+1)$ mas cada entrada (i, j) vai conter o maior *score* de um alinhamento entre um sufixo de $s[1..i]$ e um sufixo de $t[1..j]$.

Para isso, a primeira linha e a primeira coluna vão ser inicializadas com zeros.

Observação

Para qualquer entrada (i, j) há sempre o alinhamento entre os sufixos vazios de $s[1..i]$ e de $t[1..j]$, que tem *score* zero.

Portanto, este array terá todas as entradas maiores ou iguais a zero.

Usaremos ainda Programação Dinâmica

$M(i, j) =$

max

$M(i, j\text{-}1) - 2$	(último passo = I)
$M(i\text{-}1, j\text{-}1) + p(i,j)$	(último passo = S/M)
$M(i\text{-}1, j) - 2$	(último passo =R)
0	(alinhamento vazio)

No final ...

Encontrar a maior entrada em todo o array. Este será o *score* de um alinhamento local ótimo.

O alinhamento é obtido a partir dali, seguindo de volta, e parando quando não houver aresta saindo
(ou seja, onde o *score* for zero).

Alinhamento Local - Exemplo

	ε	C	A	G	C	A	C	T	C	A	T
ε	0	0	0	0	0	0	0	0	0	0	0
T	0	0	0	0	0	0	0	1	0	0	1
C	0	1	0	0	1	0	1	0	2	0	0
C	0	1	0	0	1	0	1	0	1	1	0
A	0	0	2	0	0	2	0	0	0	2	0
G	0	0	0	3	1	0	1	0	0	0	1
C	0	1	0	1	4	2	1	0	1	0	0
T	0	0	0	0	2	3	1	2	0	0	1
C	0	1	0	0	1	1	4	2	3	1	0
G	0	0	0	1	0	0	2	3	1	2	0

Alinhamento Local - Exemplo

	ε	C	A	G	C	A	C	T	C	A	T
:	.	.	.	.	.	.	.	.	.	.	.
G	0	0	0	3	1	0	1	0	0	0	1
C	0	1	0	1	4	2	1	0	1	0	0
T	0	0	0	0	2	3	1	2	0	0	1
C	0	1	0	0	1	1	4	2	3	1	0
G	0	0	0	1	0	0	2	3	1	2	0

CAGCACTCAT

TCCAGCTCG -

Alinhamento Local - Exemplo

	ε	C	A	G	C	A	C	T	C	A	T
ε		…									
T	0	0	…								
C	0	1	0	…							
C	0	1	0	0	…						
A	0	0	2	0	0	2	0	0	0	2	0
G	0	0	0	3	1	0	1	0	0	0	1
C	0	1	0	1	4	2	1	0	1	0	0
T	0	0	0	0	2	3	1	2	0	0	1
C	0	1	0	0	1	1	4	2	3	1	0
G	0	0	0	1	0	0	2	3	1	2	0

CAGCACTCAT

TCCAGCTCG