Global alignment
vs
Local alignment

Group 1

October 9, 2013

Contents

- Background
- Needleman-Wunsch global alignment
- Scoring function and gap penalty
- Alignment significance evaluation
- Smith-Waterman local alignment
- Comparisions between the two algorithms
- References

Background

- Why alignment?

  “Nothing in Biology Makes Sense Except in the Light of Evolution”

  ——T.Dobzhansky,1973

- Why computer arithmetic?

- Why dynamic programming?

Needleman-Wunsch Algorithm

	A	B	C	N	J	R	O	C	L	C	R	P	M
A													
J													
C													
J													
N													
R													
C													
K													
C													
R													
B													
P													

(modified from fig.1 of Needleman&Wunsch,1970)

Needleman-Wunsch Algorithm

	A	B	C	N	J	R	O	C	L	C	R	P	M
A	1	0	0	0	0	0	0	0	0	0	0	0	0
J	0	0	0	0	1	0	0	0	0	0	0	0	0
C	0	0	1	0	0	0	0	1	0	1	0	0	0
J	0	0	0	0	1	0	0	0	0	0	0	0	0
N	0	0	0	1	0	0	0	0	0	0	0	0	0
R	0	0	0	0	0	1	0	0	0	0	1	0	0
C	0	0	1	0	0	0	0	1	0	1	0	0	0
K	0	0	0	0	0	0	0	0	0	0	0	0	0
C	0	0	1	0	0	0	0	1	0	1	0	0	0
R	0	0	0	0	0	1	0	0	0	0	1	0	0
B	0	1	0	0	0	0	0	0	0	0	0	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

(modified from fig.1 of Needleman&Wunsch,1970)

Needleman-Wunsch Algorithm

	A	B	C	N	J	R	O	C	L	C	R	P	M
A	1												
J					1								
C			1					1		1			
J					1								
N				1									
R						1					1		
C			1					1		1			
K													
C			1					1		1			
R						1					1		
B		1											
P												1	

(modified from fig.1 of Needleman&Wunsch,1970)

$$F_{ij} = \max_{h<i,k<j} \{F_{h,j-1} + S(A_i, B_j), F_{i-1,k} + S(A_i, B_j)\}$$

(source:http://en.wikipedia.org/wiki/Needleman–Wunsch_algorithm)

Needleman-Wunsch Algorithm

	A	B	C	N	J	R	O	C	L	C	R	P	M
A	1												
J					1								
C			1					1		1			
J					1								
N				1									
R						1					1		
C			1					1		1			
K													
C			1					1		1			
R						1					1		
B		1											
P	0	0	0	0	0	0	0	0	0	0	0	1	0

(modified from fig.1 of Needleman&Wunsch,1970)

$$F_{ij} = \max_{h<i,k<j} \{F_{h,j-1} + S(A_i, B_j), F_{i-1,k} + S(A_i, B_j)\}$$

(source:http://en.wikipedia.org/wiki/Needleman–Wunsch_algorithm)

Needleman-Wunsch Algorithm

	A	B	C	N	J	R	O	C	L	C	R	P	M
A	1												
J					1								
C			1					1		1			
J					1								
N				1									
R						1					1		
C			1					1		1			
K													
C			1					1		1			
R						1					1		
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

(modified from fig.1 of Needleman&Wunsch,1970)

$$F_{ij} = \max_{h<i,k<j} \{F_{h,j-1} + S(A_i, B_j), F_{i-1,k} + S(A_i, B_j)\}$$

(source:http://en.wikipedia.org/wiki/Needleman–Wunsch_algorithm)

Needleman-Wunsch Algorithm

	A	B	C	N	J	R	O	C	L	C	R	P	M
A	1												
J					1								
C			1					1		1			
J					1								
N				1									
R						1	4	3	3	2	2	0	0
C	3	3	4	3	3	3	3	4	3	3	1	0	0
K	3	3	3	3	3	3	3	3	3	2	1	0	0
C	2	2	3	2	2	2	2	3	2	3	1	0	0
R	2	1	1	1	1	2	1	1	1	1	2	0	0
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

(modified from fig.1 of Needleman&Wunsch,1970)

$$F_{ij} = \max_{h<i,k<j} \{F_{h,j-1} + S(A_i, B_j), F_{i-1,k} + S(A_i, B_j)\}$$

(source:http://en.wikipedia.org/wiki/Needleman–Wunsch_algorithm)

Needleman-Wunsch Algorithm

	A	B	C	N	J	R	O	C	L	C	R	P	M
A	1												
J					1								
C			1					1		1			
J					1								
N				1									
R						1	4	3	3	2	2	0	0
C	3	3	4	3	3	3	3	4	3	3	1	0	0
K	3	3	3	3	3	3	3	3	3	2	1	0	0
C	2	2	3	2	2	2	2	3	2	3	1	0	0
R	2	1	1	1	1	2	1	1	1	1	2	0	0
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

(modified from fig.1 of Needleman&Wunsch,1970)

$$F_{ij} = \max_{h<i,k<j} \{F_{h,j-1} + S(A_i, B_j), F_{i-1,k} + S(A_i, B_j)\}$$

(source:http://en.wikipedia.org/wiki/Needleman–Wunsch_algorithm)

Needleman-Wunsch Algorithm

	A	B	C	N	J	R	O	C	L	C	R	P	M
A	1												
J					1								
C			1					1		1			
J					1								
N				1									
R						1	4	3	3	2	2	0	0
C	3	3	4	3	3	3	3	4	3	3	1	0	0
K	3	3	3	3	3	3	3	3	3	2	1	0	0
C	2	2	3	2	2	2	2	3	2	3	1	0	0
R	2	1	1	1	1	2	1	1	1	1	2	0	0
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

(modified from fig.1 of Needleman&Wunsch,1970)

$$F_{ij} = \max_{h<i,k<j} \{F_{h,j-1} + S(A_i, B_j), F_{i-1,k} + S(A_i, B_j)\}$$

(source:http://en.wikipedia.org/wiki/Needleman–Wunsch_algorithm)

Needleman-Wunsch Algorithm

	A	B	C	N	J	R	O	C	L	C	R	P	M
A	1												
J					1								
C			1					1		1			
J					1								
N				1									
R						5	4	3	3	2	2	0	0
C	3	3	4	3	3	3	3	4	3	3	1	0	0
K	3	3	3	3	3	3	3	3	3	2	1	0	0
C	2	2	3	2	2	2	2	3	2	3	1	0	0
R	2	1	1	1	1	2	1	1	1	1	2	0	0
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

(modified from fig.1 of Needleman&Wunsch,1970)

$$F_{ij} = \max_{h<i,k<j} \{F_{h,j-1} + S(A_i, B_j), F_{i-1,k} + S(A_i, B_j)\}$$

(source:http://en.wikipedia.org/wiki/Needleman–Wunsch_algorithm)

Needleman-Wunsch Algorithm

	A	B	C	N	J	R	O	C	L	C	R	P	M
A	8	7	6	6	5	4	4	3	3	2	1	0	0
J	7	7	6	6	6	4	4	3	3	2	1	0	0
C	6	6	7	6	5	4	4	4	3	3	1	0	0
J	6	6	6	5	6	4	4	3	3	2	1	0	0
N	5	5	5	6	5	4	4	3	3	2	1	0	0
R	4	4	4	4	4	5	4	3	3	2	2	0	0
C	3	3	4	3	3	3	3	4	3	3	1	0	0
K	3	3	3	3	3	3	3	3	3	2	1	0	0
C	2	2	3	2	2	2	2	3	2	3	1	0	0
R	2	1	1	1	1	2	1	1	1	1	2	0	0
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

(modified from fig.1 of Needleman&Wunsch,1970)

$$F_{ij} = \max_{h<i,k<j} \{F_{h,j-1} + S(A_i, B_j), F_{i-1,k} + S(A_i, B_j)\}$$

(source:http://en.wikipedia.org/wiki/Needleman–Wunsch_algorithm)

Needleman-Wunsch Algorithm

	A	B	C	N	J	R	O	C	L	C	R	P	M
A	8	7	6	6	5	4	4	3	3	2	1	0	0
J	7	7	6	6	6	4	4	3	3	2	1	0	0
C	6	6	7	6	5	4	4	4	3	3	1	0	0
J	6	6	6	5	6	4	4	3	3	2	1	0	0
N	5	5	5	6	5	4	4	3	3	2	1	0	0
R	4	4	4	4	4	5	4	3	3	2	2	0	0
C	3	3	4	3	3	3	3	4	3	3	1	0	0
K	3	3	3	3	3	3	3	3	3	2	1	0	0
C	2	2	3	2	2	2	2	3	2	3	1	0	0
R	2	1	1	1	1	2	1	1	1	1	2	0	0
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

(modified from fig.1 of Needleman&Wunsch,1970)

$$F_{ij} = \max_{h<i,k<j} \{F_{h,j-1} + S(A_i, B_j), F_{i-1,k} + S(A_i, B_j)\}$$

(source:http://en.wikipedia.org/wiki/Needleman–Wunsch_algorithm)

Needleman-Wunsch Algorithm

	A	B	C	N	J	R	O	C	L	C	R	P	M
A	8	7	6	6	5	4	4	3	3	2	1	0	0
J	7	7	6	6	6	4	4	3	3	2	1	0	0
C	6	6	7	6	5	4	4	4	3	3	1	0	0
J	6	6	6	5	6	4	4	3	3	2	1	0	0
N	5	5	5	6	5	4	4	3	3	2	1	0	0
R	4	4	4	4	4	5	4	3	3	2	2	0	0
C	3	3	4	3	3	3	3	4	3	3	1	0	0
K	3	3	3	3	3	3	3	3	3	2	1	0	0
C	2	2	3	2	2	2	2	3	2	3	1	0	0
R	2	1	1	1	1	2	1	1	1	1	2	0	0
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

(modified from fig.1 of Needleman&Wunsch,1970)

$$F_{ij} = \max_{h<i,k<j} \{F_{h,j-1} + S(A_i, B_j), F_{i-1,k} + S(A_i, B_j)\}$$

(source:http://en.wikipedia.org/wiki/Needleman–Wunsch_algorithm)

Needleman-Wunsch Algorithm

	A	B	C	N	J	R	O	C	L	C	R	P	M
A	8	7	6	6	5	4	4	3	3	2	1	0	0
J	7	7	6	6	6	4	4	3	3	2	1	0	0
C	6	6	7	6	5	4	4	4	3	3	1	0	0
J	6	6	6	5	6	4	4	3	3	2	1	0	0
N	5	5	5	6	5	4	4	3	3	2	1	0	0
R	4	4	4	4	4	5	4	3	3	2	2	0	0
C	3	3	4	3	3	3	3	4	3	3	1	0	0
K	3	3	3	3	3	3	3	3	3	2	1	0	0
C	2	2	3	2	2	2	2	3	2	3	1	0	0
R	2	1	1	1	1	2	1	1	1	1	2	0	0
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

(modified from fig.1 of Needleman&Wunsch,1970)

$$F_{ij} = \max_{h<i,k<j} \{F_{h,j-1} + S(A_i, B_j), F_{i-1,k} + S(A_i, B_j)\}$$

(source:http://en.wikipedia.org/wiki/Needleman–Wunsch_algorithm)

Needleman-Wunsch Algorithm

	A	B	C	N	J	R	O	C	L	C	R	P	M
A	8	7	6	6	5	4	4	3	3	2	1	0	0
J	7	7	6	6	6	4	4	3	3	2	1	0	0
C	6	6	7	6	5	4	4	4	3	3	1	0	0
J	6	6	6	5	6	4	4	3	3	2	1	0	0
N	5	5	5	6	5	4	4	3	3	2	1	0	0
R	4	4	4	4	4	5	4	3	3	2	2	0	0
C	3	3	4	3	3	3	3	4	3	3	1	0	0
K	3	3	3	3	3	3	3	3	3	2	1	0	0
C	2	2	3	2	2	2	2	3	2	3	1	0	0
R	2	1	1	1	1	2	1	1	1	1	2	0	0
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

(modified from fig.1 of Needleman&Wunsch,1970)

$$F_{ij} = \max_{h<i,k<j} \{F_{h,j-1} + S(A_i, B_j), F_{i-1,k} + S(A_i, B_j)\}$$

(source:http://en.wikipedia.org/wiki/Needleman–Wunsch_algorithm)

Needleman-Wunsch Algorithm

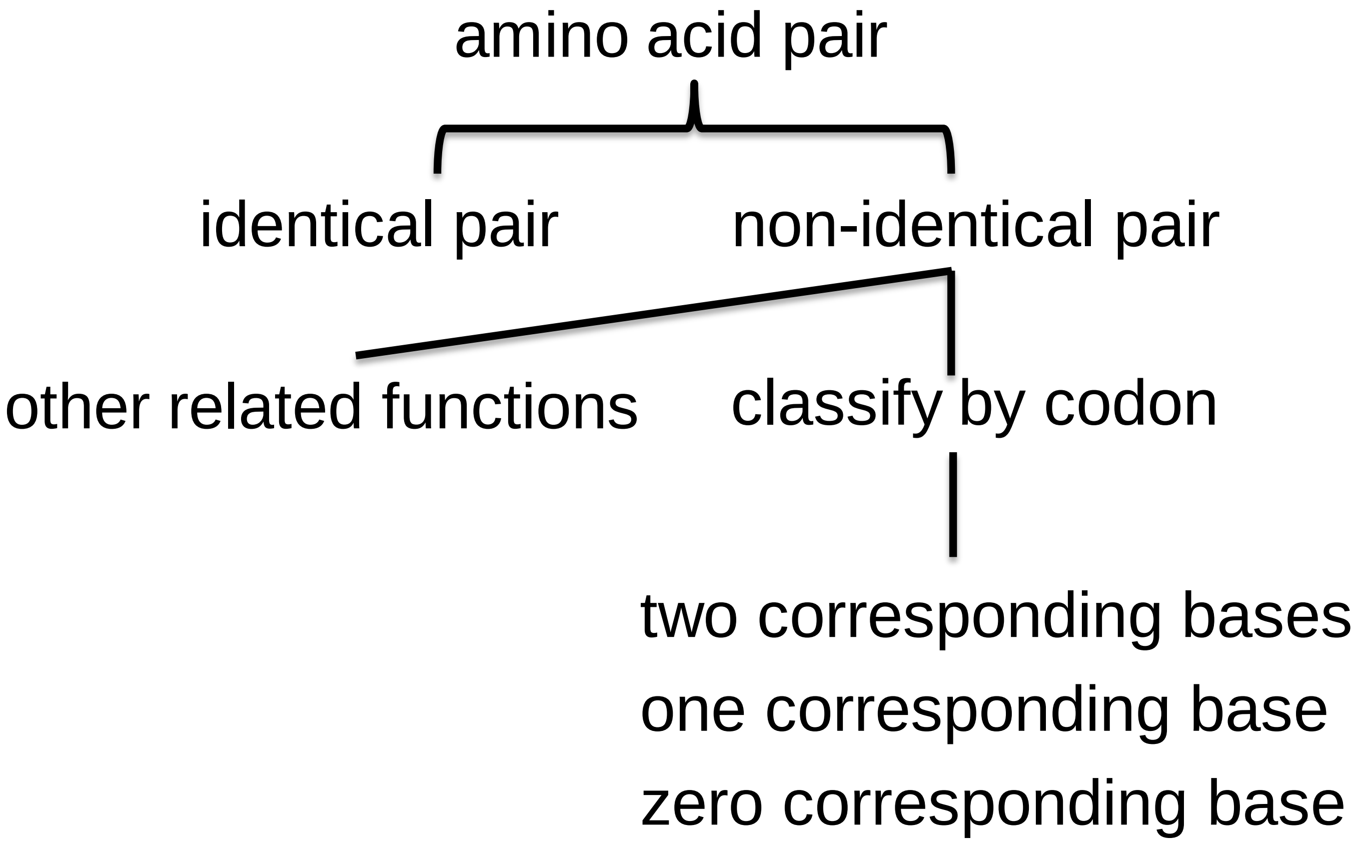
	A	B	C	N	J	R	O	C	L	C	R	P	M
A	8	7	6	6	5	4	4	3	3	2	1	0	0
J	7	7	6	6	6	4	4	3	3	2	1	0	0
C	6	6	7	6	5	4	4	4	3	3	1	0	0
J	6	6	6	5	6	4	4	3	3	2	1	0	0
N	5	5	5	6	5	4	4	3	3	2	1	0	0
R	4	4	4	4	4	5	4	3	3	2	2	0	0
C	3	3	4	3	3	3	3	4	3	3	1	0	0
K	3	3	3	3	3	3	3	3	3	2	1	0	0
C	2	2	3	2	2	2	2	3	2	3	1	0	0
R	2	1	1	1	1	2	1	1	1	1	2	0	0
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

(modified from fig.1 of Needleman&Wunsch,1970)

$$F_{ij} = \max_{h<i,k<j} \{F_{h,j-1} + S(A_i, B_j), F_{i-1,k} + S(A_i, B_j)\}$$

(source:http://en.wikipedia.org/wiki/Needleman–Wunsch_algorithm)

Scoring function and Gap Penalty



amino acid pair

identical pair

non-identical pair

other related functions

classify by codon

two corresponding bases

one corresponding base

zero corresponding base

Gap penalty: avoid too many gaps

Scoring function and Gap Penalty

- Needleman-Wunsch Algorithm

$$F_{i0} = d * i$$

$$F_{0j} = d * j$$

$$F_{ij} = \max(F_{i-1,j-1} + S(A_i, B_j),\ F_{i,j-1} + d,\ F_{i-1,j} + d)$$

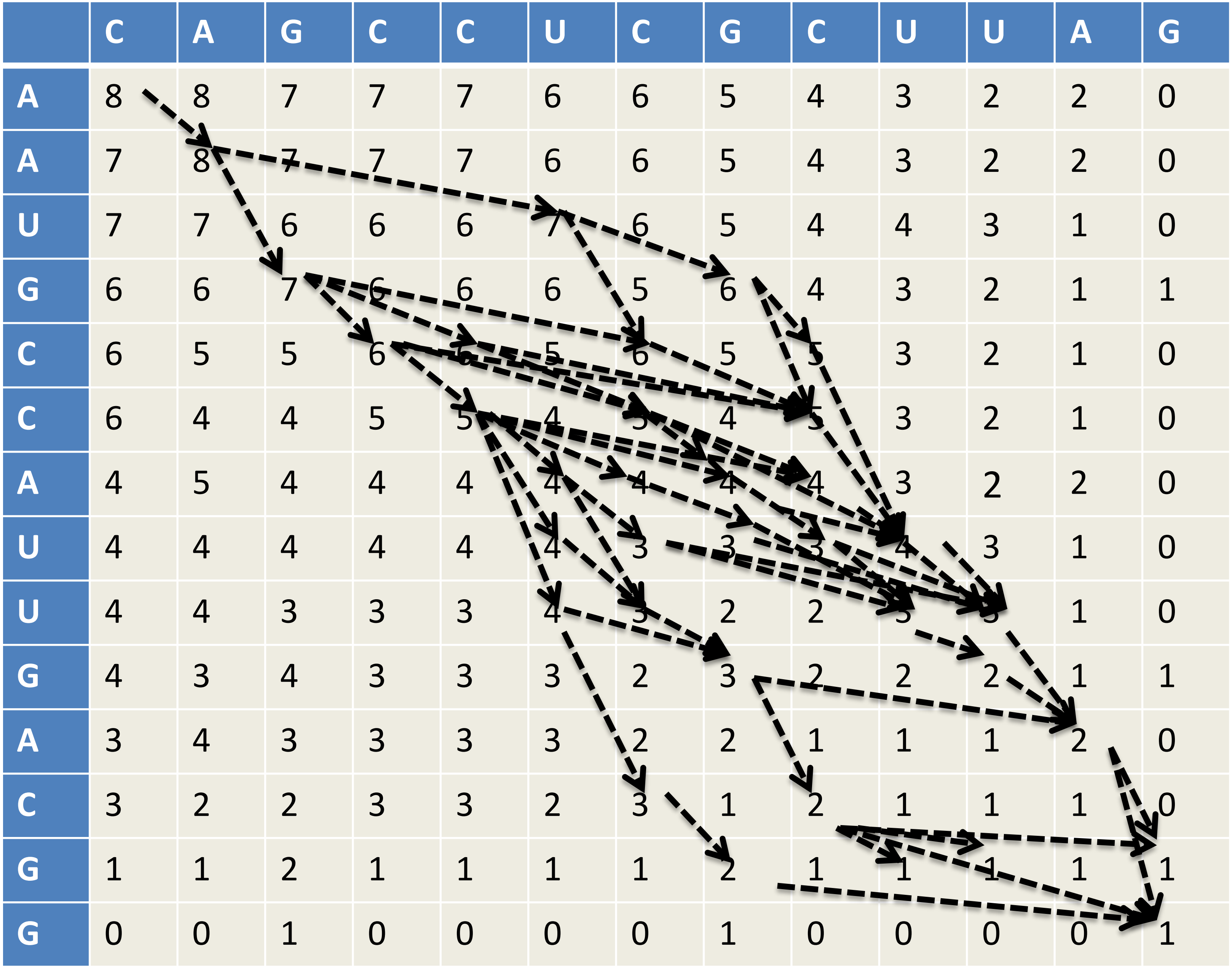
$$F_{ij} = \max_{h<i,k<j} \{F_{h,j-1} + S(A_i, B_j), F_{i-1,k} + S(A_i, B_j)\}$$

(source: http://en.wikipedia.org/wiki/Needleman–Wunsch_algorithm)

Scoring function and Gap Penalty

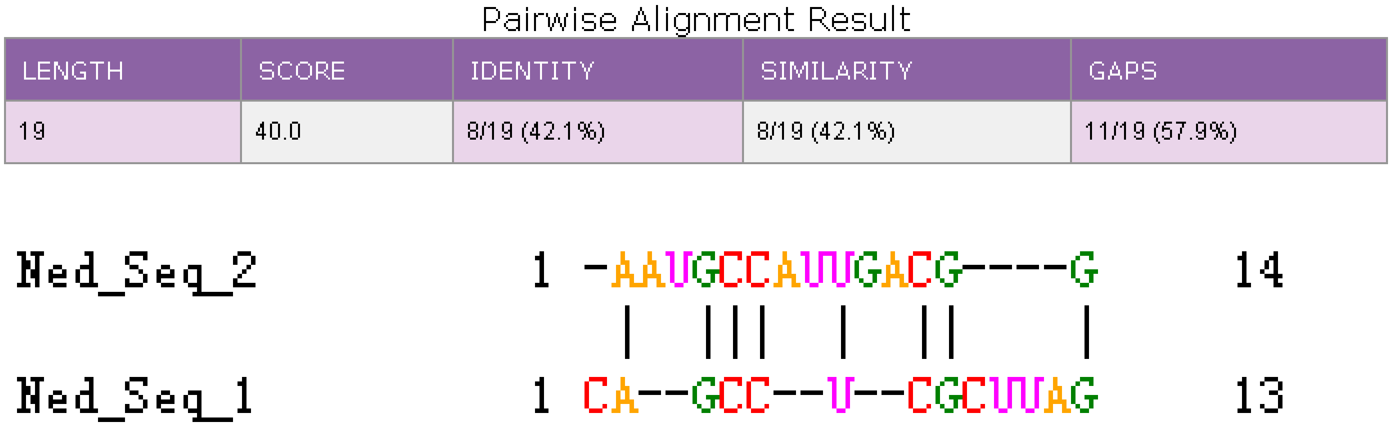
	C	A	G	C	C	U	C	G	C	U	U	A	G
A													
A													
U													
G													
C													
C													
A													
U													
U													
G													
A													
C													
G													
G													

Scoring function and Gap Penalty



	C	A	G	C	C	U	C	G	C	U	U	A	G
A	8	8	7	7	7	6	6	5	4	3	2	2	0
A	7	8	7	7	7	6	6	5	4	3	2	2	0
U	7	7	6	6	6	7	6	5	4	4	3	1	0
G	6	6	7	6	6	6	5	6	4	3	2	1	1
C	6	5	5	6	6	5	6	5	5	3	2	1	0
C	6	4	4	5	5	4	5	4	5	3	2	1	0
A	4	5	4	4	4	4	4	4	4	3	2	2	0
U	4	4	4	4	4	4	3	3	3	4	3	1	0
U	4	4	3	3	3	4	3	2	2	3	3	1	0
G	4	3	4	3	3	3	2	3	2	2	2	1	1
A	3	4	3	3	3	3	2	2	1	1	1	2	0
C	3	2	2	3	3	2	3	1	2	1	1	1	0
G	1	1	2	1	1	1	1	2	1	1	1	1	1
G	0	0	1	0	0	0	0	1	0	0	0	0	1

Scoring function and Gap Penalty

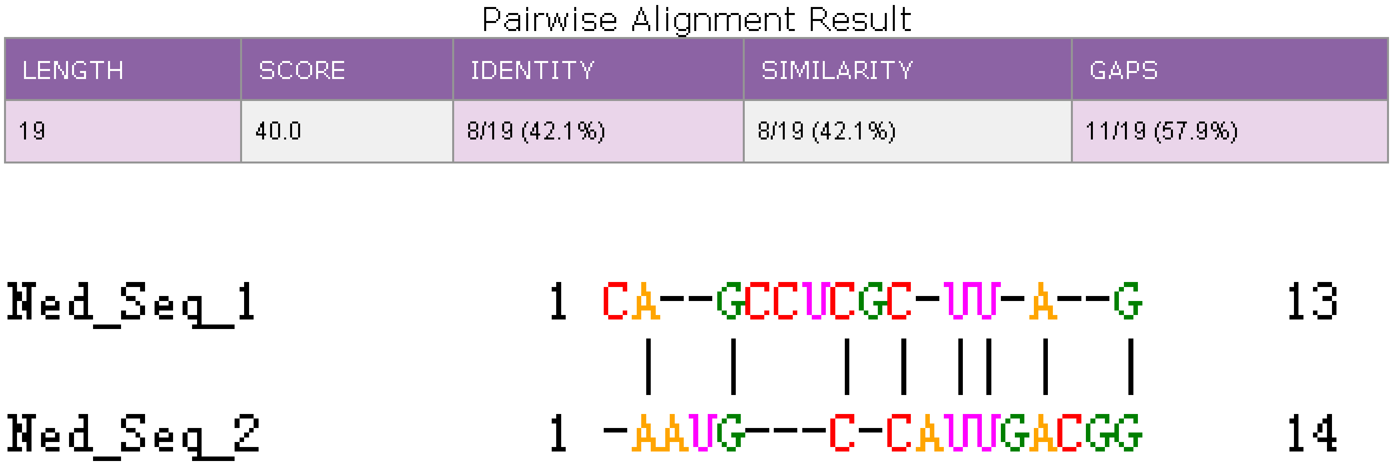


Pairwise Alignment Result

LENGTH	SCORE	IDENTITY	SIMILARITY	GAPS
19	40.0	8/19 (42.1%)	8/19 (42.1%)	11/19 (57.9%)

```
Ned_Seq_2      1 -AAUGCCAUUGACG----G     14
                  |  |||  |  ||    |
Ned_Seq_1      1 CA--GCC--U--CGCUUAG     13
```

a



Pairwise Alignment Result

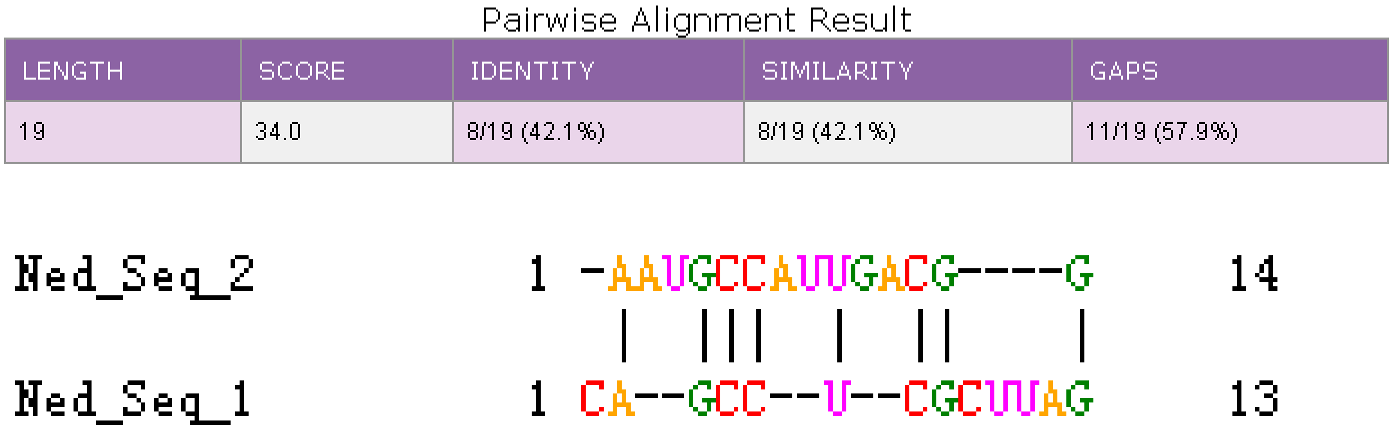
LENGTH	SCORE	IDENTITY	SIMILARITY	GAPS
19	40.0	8/19 (42.1%)	8/19 (42.1%)	11/19 (57.9%)

```
Ned_Seq_1      1 CA--GCCUCGC-UU-A--G     13
                  |  |   | | || |  |
Ned_Seq_2      1 -AAUG---C-CAUUGACGG     14
```

b

Needleman algorithm (simplified)

Scoring function and Gap Penalty

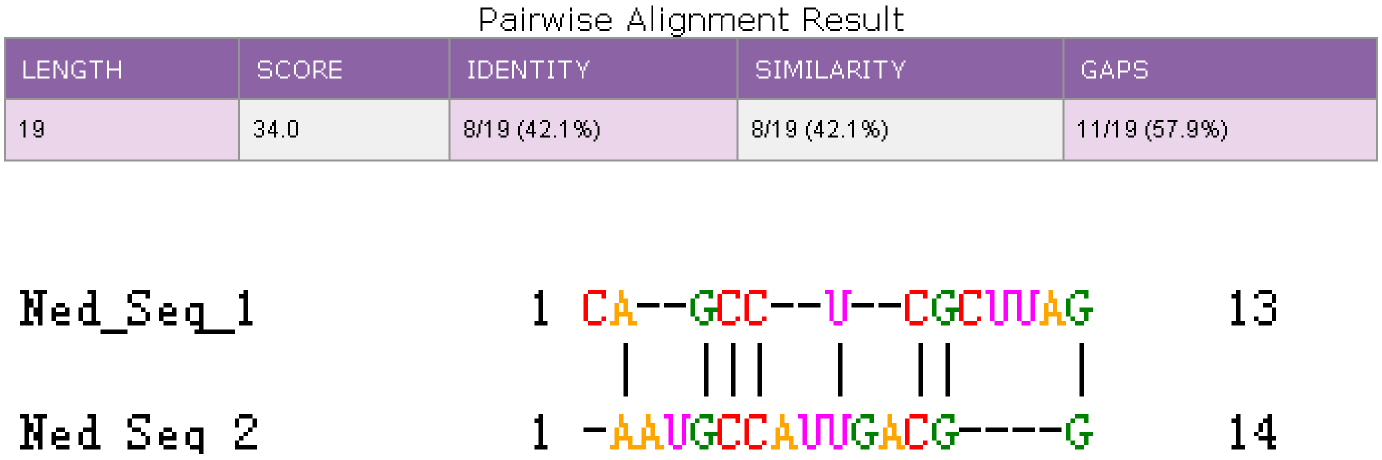


Pairwise Alignment Result

LENGTH	SCORE	IDENTITY	SIMILARITY	GAPS
19	34.0	8/19 (42.1%)	8/19 (42.1%)	11/19 (57.9%)

```
Ned_Seq_2      1 -AAUGCCAUUGACG----G      14
                  |  |||  |  ||    |
Ned_Seq_1      1 CA--GCC--U--CGCUUAG      13
```

c



Pairwise Alignment Result

LENGTH	SCORE	IDENTITY	SIMILARITY	GAPS
19	34.0	8/19 (42.1%)	8/19 (42.1%)	11/19 (57.9%)

```
Ned_Seq_1      1 CA--GCC--U--CGCUUAG      13
                  |  |||  |  ||    |
Ned Seq 2      1 -AAUGCCAUUGACG----G      14
```

d

Needleman algorithm

Alignment significance evaluation

- Question:

Whether a particular result differs significance from a fortuitous match between two random sequences ?

- To answer this question we can do:

  Sequence alignment between two sets of random sequences

  Or:

  Sequence alignment between one set of random sequences and a real sequence

Smith-Waterman Algorithm

$$H_{k0} = H_{0l} = 0 \text{ for } 0 \le k \le n \text{ and } 0 \le l \le m.$$

$$H_{ij} = \max\{H_{i-1,j-1} + s(a_i,b_j), \max_{k \ge 1}\{H_{i-k,j} - W_k\}, \max_{l \ge 1}\{H_{i,j-l} - W_l\}, 0\}. \quad (1)$$

$1 \le i \le n$ and $1 \le j \le m$.

$$W_k = 1.0 + 1/3*k.$$

(source: Smith & Waterman,1981)

	gap	a1	a2	a3	a4
gap	0	1 gap	2 gaps	3 gaps	4 gaps
b1	1 gap	A11	A21	A31	A41
b2	2 gaps	A12	A22	A32	A42
b3	3 gaps	A13	A23	A33	A43

(modified from Fig.3.9 of D.W.Mount
Bioinformatics Sequence and Gene
Analysis, Second Eition, P74)

Smith-Waterman Algorithm

$$H_{k0} = H_{0l} = 0 \text{ for } 0 \le k \le n \text{ and } 0 \le l \le m.$$

$$H_{ij} = \max\{H_{i-1,j-1} + s(a_i,b_j), \max_{k \ge 1}\{H_{i-k,j} - W_k\}, \max_{l \ge 1}\{H_{i,j-l} - W_l\}, 0\}. \quad (1)$$

$1 \le i \le n$ and $1 \le j \le m$.

$$W_k = 1.0 + 1/3 * k.$$

(source: Smith & Waterman,1981)

	gap	a1	a2	a3	a4
gap	0	1 gap	2 gaps	3 gaps	4 gaps
b1	1 gap	A11	A21	A31	A41
b2	2 gaps	A12	A22	A32	A42
b3	3 gaps	A13	A23	A33	A43

(modified from Fig.3.9 of D.W.Mount Bioinformatics Sequence and Gene Analysis, Second Eition, P74)

Smith-Waterman Algorithm

$$H_{k0} = H_{0l} = 0 \text{ for } 0 \le k \le n \text{ and } 0 \le l \le m.$$

$$H_{ij} = \max\{H_{i-1,j-1} + s(a_i,b_j). \max_{k\ge 1}\{H_{i-k,j} - W_k\}. \max_{l\ge 1}\{H_{i,j-l} - W_l\}. 0\}. \qquad (1)$$

$1 \le i \le n$ and $1 \le j \le m.$

$$W_k = 1{\cdot}0 + 1/3 * k.$$

(source: Smith & Waterman,1981)

	gap	a1	a2	a3	a4
gap	0	1 gap	2 gaps	3 gaps	4 gaps
b1	1 gap	A11	A21	A31	A41
b2	2 gaps	A12	A22	A32	A42
b3	3 gaps	A13	A23	A33	A43

O

(modified from Fig.3.9 of D.W.Mount Bioinformatics Sequence and Gene Analysis, Second Eition, P74)

Smith-Waterman Algorithm

$$H(i,0) = 0,\ 0 \le i \le m$$

$$H(0,j) = 0,\ 0 \le j \le n$$

if $a_i = b_j$ then $w(a_i, b_j) = w(\text{match})$ or if $a_i \ne b_j$ then $w(a_i, b_j) = w(\text{mismatch})$

$$H(i,j) = \max \left\{ \begin{array}{cc} & 0 \\ H(i-1, j-1) + \ w(a_i, b_j) & \text{Match/Mismatch} \\ H(i-1, j) + \ w(a_i, -) & \text{Deletion} \\ H(i, j-1) + \ w(-, b_j) & \text{Insertion} \end{array} \right\},\ 1 \le i \le m, 1 \le j \le n$$

(source: http://en.wikipedia.org/wiki/Smith-Waterman_algorithm)

	gap	a1	a2	a3	a4
gap	0	1 gap	2 gaps	3 gaps	4 gaps
b1	1 gap	A11	A21	A31	A41
b2	2 gaps	A12	A22	A32	A42
b3	3 gaps	A13	A23	A33	A43

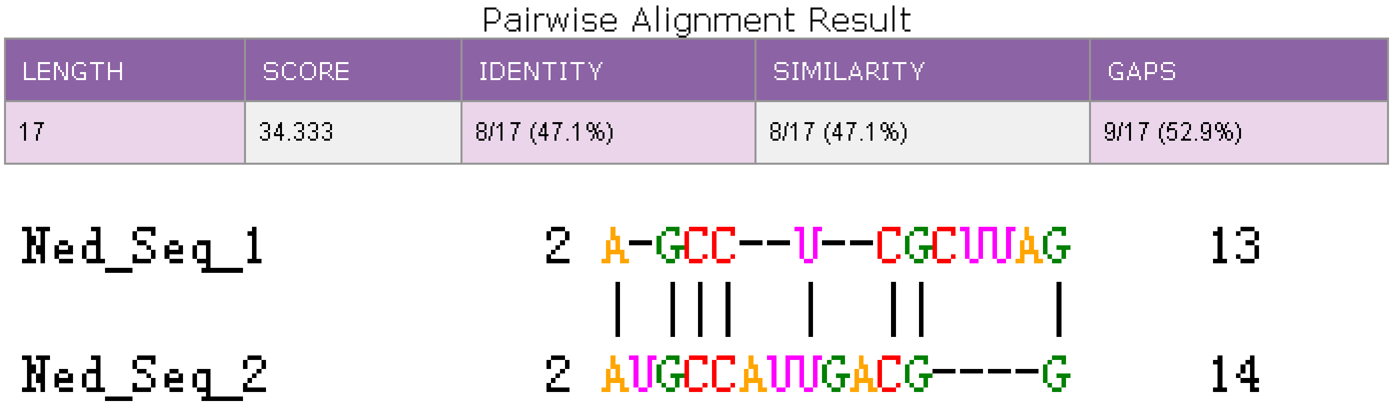
O

(modified from Fig.3.9 of D.W.Mount Bioinformatics Sequence and Gene Analysis, Second Eition, P74)

Smith-Waterman Algorithm

- Compared with global alignment：
  - Zero could terminate the current local alignment
  - Mismatch must be negative scored
- Other properties：
  - Suitable to identify conserved local sequence (substring)
  - Guaranteed to find the best local alignment
  - Perform poorly when dealing with separated regions within a long sequence

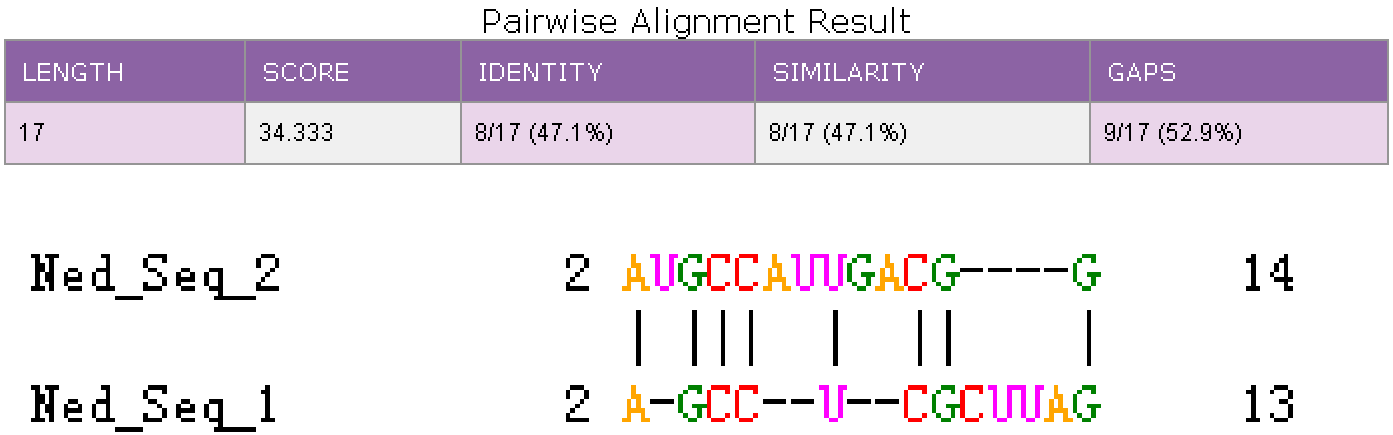
Smith-Waterman Algorithm



Pairwise Alignment Result

LENGTH	SCORE	IDENTITY	SIMILARITY	GAPS
17	34.333	8/17 (47.1%)	8/17 (47.1%)	9/17 (52.9%)

```
Ned_Seq_1          2 A-GCC--U--CGCUUAG      13
                     | |||  |  ||    |
Ned_Seq_2          2 AUGCCAUUGACG----G      14
```



Pairwise Alignment Result

LENGTH	SCORE	IDENTITY	SIMILARITY	GAPS
17	34.333	8/17 (47.1%)	8/17 (47.1%)	9/17 (52.9%)

```
Ned_Seq_2          2 AUGCCAUUGACG----G      14
                     | |||  |  ||    |
Ned_Seq_1          2 A-GCC--U--CGCUUAG      13
```

Comparision

Match	1
Mismatch	0
Gap	0

	A	K	C	A	C	K
C	2	2	3	1	1	0
A	1	1	1	2	0	0
C	0	0	1	0	1	0

Needleman-Wunsch (Simplified)

Match	1
Mismatch	-1
Gap	-1

		A	K	C	A	C	K
	0	0	0	0	0	0	0
C	0	0	0	1	0	1	0
A	0	1	0	0	2	0	0
C	0	0	0	1	0	3	1

Needleman-Wunsch
Smith-Waterman

References

1. Needleman, S. B., & Wunsch, C. D. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of molecular biology*, 48(3), 443-453.
2. Sellers, P. H. (1974). On the theory and computation of evolutionary distances.*SIAM Journal on Applied Mathematics*, *26*(4), 787-793..
3. Waterman, M. S., Smith, T. F., & Beyer, W. A. (1976). Some biological sequence metrics*. Advances in Mathematics*, 20(3), 367-387.
4. Dayhoff, M.O., Schwartz, R. and Orcutt, B.C. (1978). A model of Evolutionary Change in Proteins. *Atlas of protein sequence and structure* (volume 5, supplement 3 ed.). *Nat. Biomed. Res. Found*. 345–358.
5. Smith, T. F., & Waterman, M. S. (1981). Comparison of biosequences. *Advances in Applied Mathematics*, *2*(4), 482-489.
6. David, W. M.(2004).Bioinformatics: Sequence and Genome Analysis(Second Edition).*Cold Spring Harbor Laboratory Press*, 69-84.
7. http://en.wikipedia.org/wiki/Needleman–Wunsch_algorithm
8. http://en.wikipedia.org/wiki/Smith-Waterman_algorithm

Acknowledgments

- **Yu-Jian Kang**
- **Jing Ma**
- **Xiao-Xu Yang**
- **Si-Jin Cheng**
- **Yu-qi Meng**
- **Yong-Xin Ye**
- **Xiao-Tong Yang**
- **Peng-Ze Wu**
- **Our Supervisors**
- **And those pioneer bioinformaticians**

Thanks for your attention!