

# Interactive demo for Needleman-Wunsch algorithm

The motivation behind this demo is that I had some difficulty understanding the algorithm, so to gain better understanding I decided to implement it. This is not meant for serious use, What I tried to do here is to illustrate visually how the matrix is constructed and how the algorithm works. Also I wanted to allow some freedom for the user to construct a custom path along the matrix and see how paths translate into alignment "or mis-alignments". The code is available on [github](#) and is released under GNU/GPL3. If you wish to contact me send me an email to [mostafa.mohammed@gmail.com](mailto:mostafa.mohammed@gmail.com).

Sequence 1  
Sequence 2  
Match Score  
1

ACTA

AT

Mismatch Score

-1

Gap Score

-2

Compute Optimal Alignment

Clear Path

Custom Path

A

A

-

T

-

A

C

T

A

Score = -2

	A	T
0	-2	-4
A	-2	
	1	-1
C	-4	
	-1	0
T	-6	
	-3	0
A	-8	
	-5	-2

# Interactive demo for Needleman-Wunsch algorithm

The motivation behind this demo is that I had some difficulty understanding the algorithm, so to gain better understanding I decided to implement it. This is not meant for serious use, What I tried to do here is to illustrate visually how the matrix is constructed and how the algorithm works. Also I wanted to allow some freedom for the user to construct a custom path along the matrix and see how paths translate into alignment "or mis-alignments". The code is available on [github](#) and is released under GNU/GPL3. If you wish to contact me send me an email to [mostafa.mohammed@gmail.com](mailto:mostafa.mohammed@gmail.com).

Sequence 1

ACTA

Sequence 2

ATGCTA

Match Score

1

Mismatch Score

-1

Gap Score

-2

Score = 0

Compute Optimal Alignment

Clear Path

Custom Path

	A	T	G	C	T	A	
0	-2	-4	-6	-8	-10	-12	
A	-2	1	-1	-3	-5	-7	-9
C	-4	-1	0	-2	-2	-4	-6
T	-6	-3	0	-1	-3	-1	-3
A	-8	-5	-2	-1	-2	-3	0

# Interactive demo for Needleman-Wunsch algorithm

The motivation behind this demo is that I had some difficulty understanding the algorithm, so to gain better understanding I decided to implement it. This is not meant for serious use, What I tried to do here is to illustrate visually how the matrix is constructed and how the algorithm works. Also I wanted to allow some freedom for the user to construct a custom path along the matrix and see how paths translate into alignment "or mis-alignments". The code is available on [github](#) and is released under GNU/GPL3. If you wish to contact me send me an email to [mostafa.mohammed@gmail.com](mailto:mostafa.mohammed@gmail.com).

Sequence 1  
MTP

Sequence 2  
MSRDETHTP

Match Score  
1

Mismatch Score  
-1

Gap Score  
-2

Compute Optimal Alignment

Clear Path

Custom Path

M

S

R

D

E

T

H

T

P

M

S

R

D

E

T

H

T

P

Score = -9

	0	-2	-4	-6	-8	-10	-12	-14	-16	-18
M	-2	1	-1	-3	-5	-7	-9	-11	-13	-15
T	-4	-1	0	-2	-4	-6	-6	-8	-10	-12
P	-6	-3	-2	-1	-3	-5	-7	-7	-9	-9

# Interactive demo for Needleman-Wunsch algorithm

The motivation behind this demo is that I had some difficulty understanding the algorithm, so to gain better understanding I decided to implement it. This is not meant for serious use, What I tried to do here is to illustrate visually how the matrix is constructed and how the algorithm works. Also I wanted to allow some freedom for the user to construct a custom path along the matrix and see how paths translate into alignment "or mis-alignments". The code is available on [github](#) and is released under GNU/GPL3. If you wish to contact me send me an email to [mostafa.mohammed@gmail.com](mailto:mostafa.mohammed@gmail.com).

Sequence 1  
PTM

Sequence 2  
PTHTEDRSM

Match Score  
1

Mismatch Score  
-1

Gap Score  
-2

Compute Optimal Alignment

Clear Path

Custom Path

P  
T  
H  
T  
E  
D  
R  
S  
M

T  
-  
-  
-  
-  
-  
-  
-  
-  
M

P  
T  
H  
T  
E  
D  
R  
S  
M

Score = -9

	P	T	H	T	E	D	R	S	M
0	-2	-4	-6	-8	-10	-12	-14	-16	-18
-2	1	-1	-3	-5	-7	-9	-11	-13	-15
-4	-1	2	0	-2	-4	-6	-8	-10	-12
-6	-3	0	1	-1	-3	-5	-7	-9	-9



# Interactive demo for Needleman-Wunsch algorithm

(sequences as in NW-2190)

The motivation behind this demo is that I had some difficulty understanding the algorithm, so to gain better understanding I decided to implement it. This is not meant for serious use, What I tried to do here is to illustrate visually how the matrix is constructed and how the algorithm works. Also I wanted to allow some freedom for the user to construct a custom path along the matrix and see how paths translate into alignment "or mis-alignments". The code is available on [github](https://github.com/mostafaio/public-needleman-wunsch) and is released under GNU/GPL3. If you wish to contact me send me an email to [mostafa.mohammed@gmail.com](mailto:mostafa.mohammed@gmail.com).

Sequence 1

Sequence 2

Match Score  Mismatch Score  Gap Score

M P R C C L C C Q R J J N C B A  
P B R C C K C R N J - C J A  
Score = -2

M P R C L C Q R J N C B A

0 -2 -4 -6 -8 -10 -12 -14 -16 -18 -20 -22 -24 -26

P -2 -1 -1 -3 -5 -7 -9 -11 -13 -15 -17 -19 -21 -23

B -4 -3 -2 -2 -4 -6 -8 -10 -12 -14 -16 -18 -18 -20

R -6 -5 -4 -1 -3 -5 -7 -9 -11 -13 -15 -17 -19

C -8 -7 -6 -3 0 -2 -4 -6 -8 -10 -12 -14 -16

K -10 -9 -8 -5 -2 -1 -3 -5 -7 -9 -11 -13 -15

C -12 -11 -10 -7 -4 -3 0 -2 -4 -6 -8 -10 -12 -14

R -14 -13 -12 -9 -6 -5 -2 -1 -1 -3 -5 -7 -9 -11

[illegible]

# Interactive demo for Needleman-Wunsch algorithm

The motivation behind this demo is that I had some difficulty understanding the algorithm, so to gain better understanding I decided to implement it. This is not meant for serious use. What I tried to do here is to illustrate visually how the matrix is constructed and how the algorithm works. Also I wanted to allow some freedom for the user to construct a custom path along the matrix and see how paths translate into alignment "or mis-alignments". The code is available on [github](#) and is released under GNU/GPL3. If you wish to contact me send me an email to [mostafa.mohammed@gmail.com](mailto:mostafa.mohammed@gmail.com).

Sequence 1  
Sequence 2  
Match Score  
1

PBRCKGRNJCA  
MPRCLCQRJNCBA  
Mismatch Score  
0

Gap Score  
0

Compute Optimal Alignment

Clear Path

Custom Path

M P - R C - L C Q R - J N C B A

- P B R C K - C - R N J - C J - A

Score = 8

original score

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

=====

A