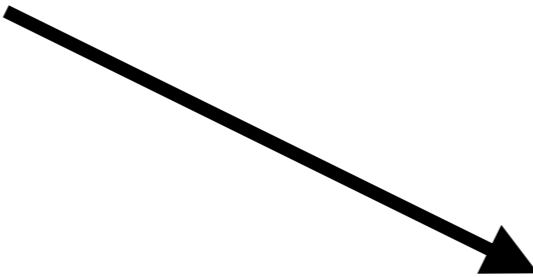


Multiple Sequence Alignment Introduction

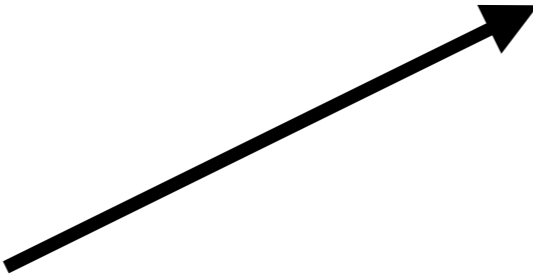
Sequence Alignment

ACGTGGT



ACGTGGT
ACGTCGT

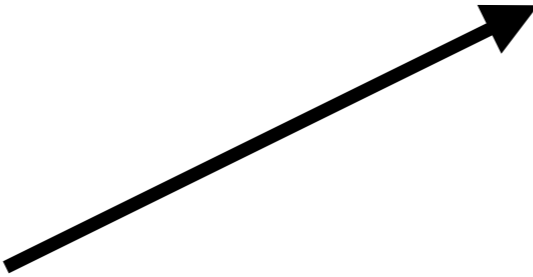
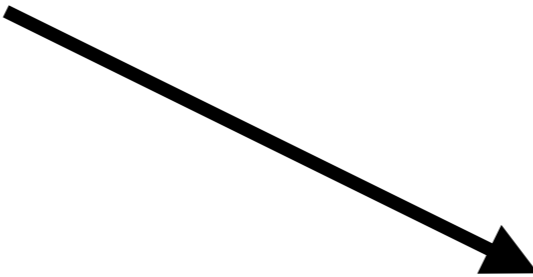
ACGTCGT



Sequence Alignment

ACGTGGT

CGGGTA



ACGTGGT
CG-GGTA

Multiple Sequence Alignment

ACGTGGT

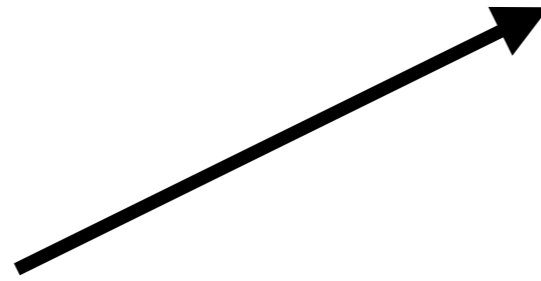
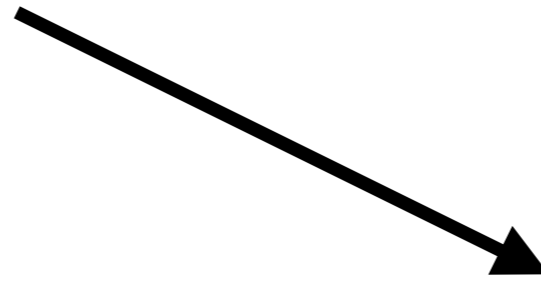
ACTGGTA

CGGGTA

ACGTGGT

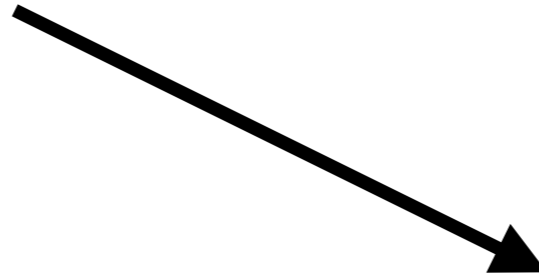
AC-TGGTA

CG-GGTA



Multiple Sequence Alignment

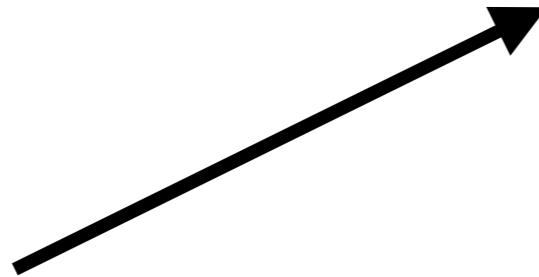
ACGTGGT



GCTGGACCNC



TTCGACT



Multiple Sequence Alignment

Workshop on Molecular Evolution 2000

M	O	L	E	C	U	L	A	R	.	.	.	E	V	O	L	U	T	I	O	N
.	.	.	.	C	A	P	T	A	I	N	.	K	I	D	D
.	.	S	L	E	E	P	.	.	L	E	A	R	N
.	C	H	O	O	S	E	.	T	W	O

Marine Biological Laboratory, Woods Hole, MA

Scoring schemes

Match: +1 / Mismatch: -1 / Indel: -1

SLEEP--LEARN-
---CHOOSE-TWO

Scoring schemes

Match: +1 / Mismatch: -1 / Indel: -1

SLEEP--LEARN-
--CHOOSE-TWO

-1

Total: -1

Scoring schemes

Match: +1 / Mismatch: -1 / Indel: -1

SLEEP--LEARN-
--CHOOSE-TWO

-1

Total: -2

Scoring schemes

Match: +1 / Mismatch: -1 / Indel: -1

SLEEP--LEARN-
--CHOOSE-TWO

-1

Total: -3

Scoring schemes

Match: +1 / Mismatch: -1 / Indel: -1

SLEEP--LEARN-
---CHOOSE-TWO

-1

Total: -4

Scoring schemes

Match: +1 / Mismatch: -1 / Indel: -1

SLEEP--LEARN-
---CHOOSE-TWO

-1

Total: -5

Scoring schemes

Match: +1 / Mismatch: -1 / Indel: -1

SLEEP--LEARN--
---CHOOSE-TWO

-1

Total: -6

Scoring schemes

Match: +1 / Mismatch: -1 / Indel: -1

SLEEP--LEARN--
---CHOOSE--TWO

-1

Total: -7

Scoring schemes

Match: +1 / Mismatch: -1 / Indel: -1

SLEEP--LEARN--
---CHOOSE--TWO

-1

Total: -8

Scoring schemes

Match: +1 / Mismatch: -1 / Indel: -1

SLEEP--LEARN--
---CHOOSE-TWO

+1

Total: -7

Scoring schemes

Match: +1 / Mismatch: -1 / Indel: -1

SLEEP--LEARN--
---CHOOSE-TWO

-1

Total: -8

Scoring schemes

Match: +1 / Mismatch: -1 / Indel: -1

SLEEP--LEARN-
---CHOOSE-TWO

-1

Total: -9

Scoring schemes

Match: +1 / Mismatch: -1 / Indel: -1

SLEEP--LEARN-
---CHOOSE-TWO

-1

Total: -10

Scoring schemes

Match: +1 / Mismatch: -1 / Indel: -1

SLEEP--LEARN-
---CHOOSE-TWO

-1

Total: -11

Gap opening / extension

Match: +1 / Mismatch: -1 / Gap opening: -1 / Gap extension: -0.5

SLEEP--LEARN-
---CHOOSE-TWO

Gap opening / extension

Match: +1 / Mismatch: -1 / Gap opening: -1 / Gap extension: -0.5

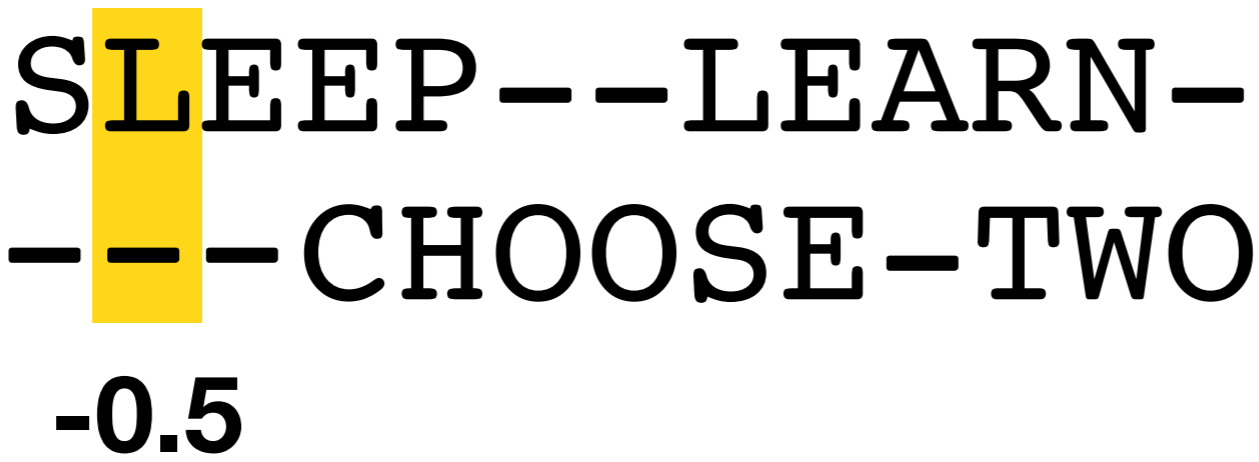
SLEEP--LEARN-
--CHOOSE-TWO

-1

Total: -1

Gap opening / extension

Match: +1 / Mismatch: -1 / Gap opening: -1 / Gap extension: -0.5



Total: -1.5

Gap opening / extension

Match: +1 / Mismatch: -1 / Gap opening: -1 / Gap extension: -0.5

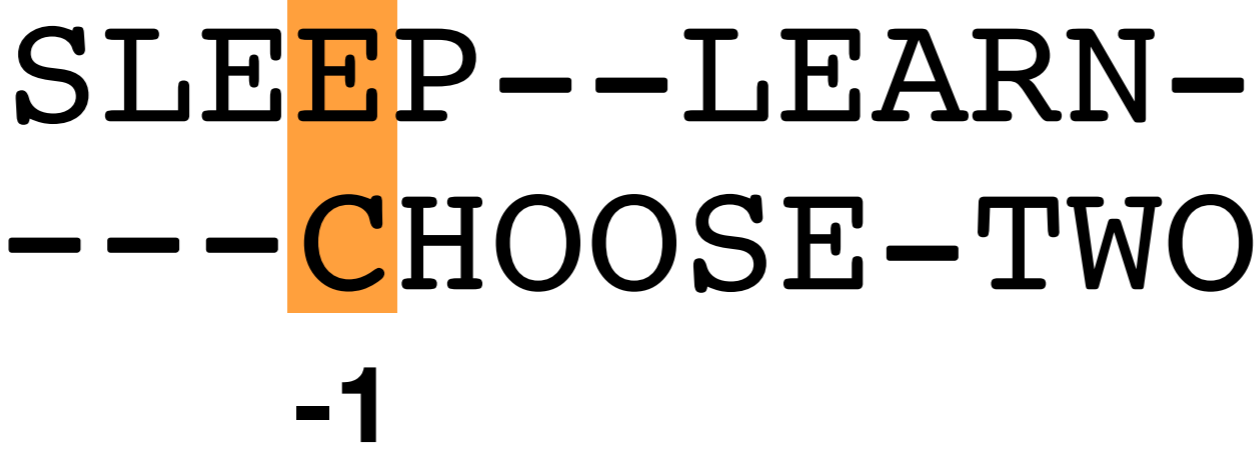
SLEEP--LEARN--
--CHOOSE-TWO

-0.5

Total: -2

Gap opening / extension

Match: +1 / Mismatch: -1 / Gap opening: -1 / Gap extension: -0.5



Total: -3

Gap opening / extension

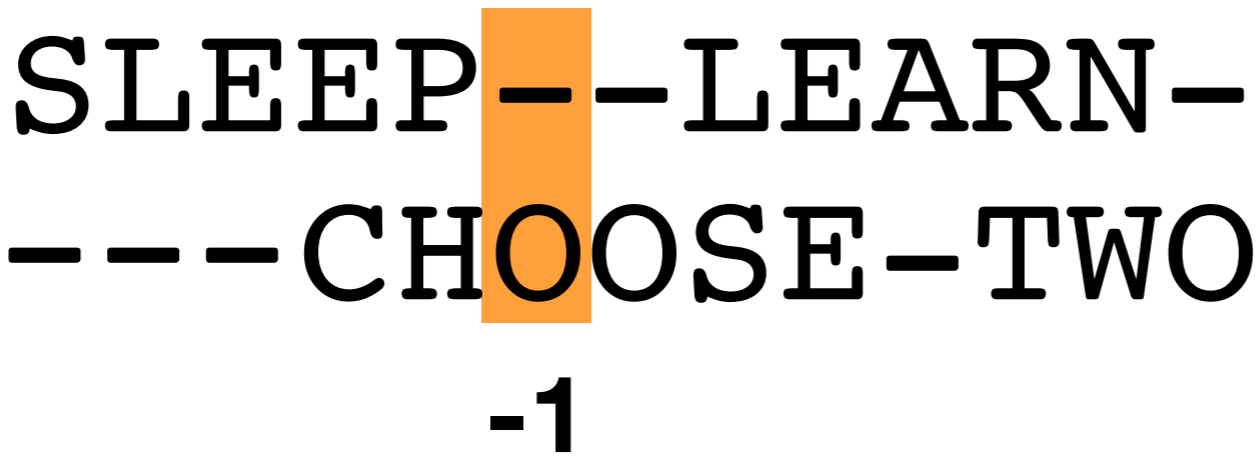
Match: +1 / Mismatch: -1 / Gap opening: -1 / Gap extension: -0.5

SLEEP--LEARN--
---CHOOSE-TWO
-1

Total: -4

Gap opening / extension

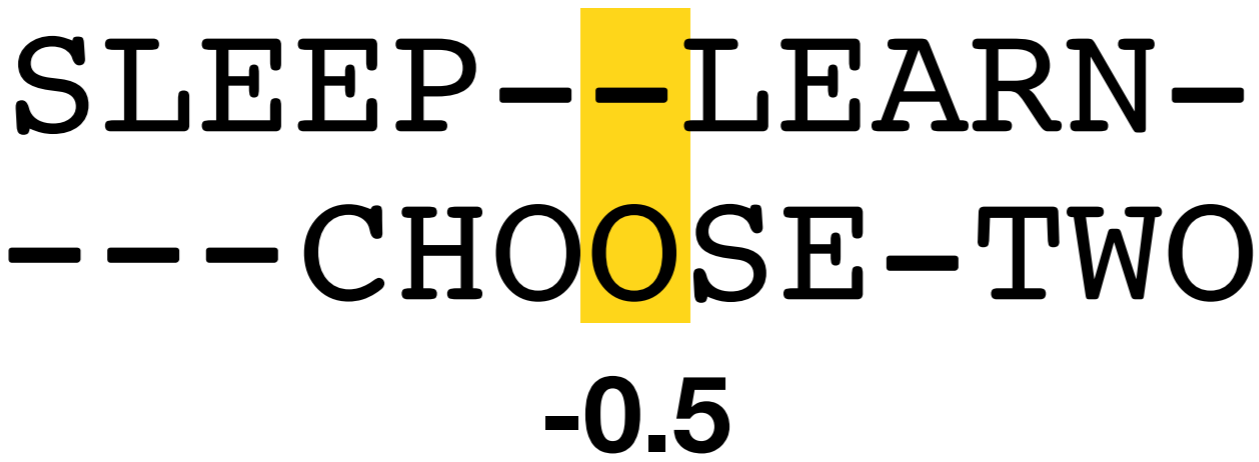
Match: +1 / Mismatch: -1 / Gap opening: -1 / Gap extension: -0.5



Total: -5

Gap opening / extension

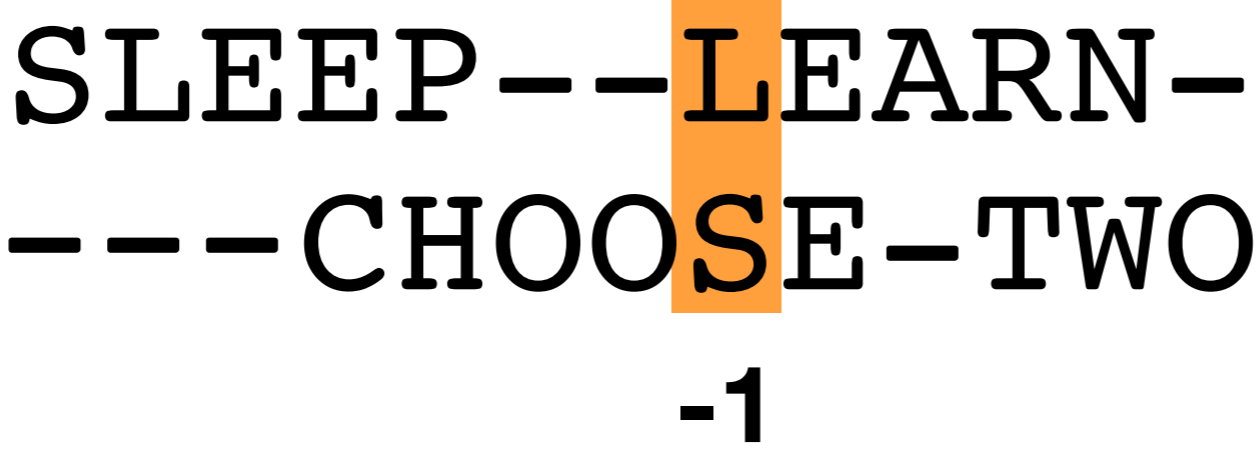
Match: +1 / Mismatch: -1 / Gap opening: -1 / Gap extension: -0.5



Total: -5.5

Gap opening / extension

Match: +1 / Mismatch: -1 / Gap opening: -1 / Gap extension: -0.5



Total: -6.5

Gap opening / extension

Match: +1 / Mismatch: -1 / Gap opening: -1 / Gap extension: -0.5

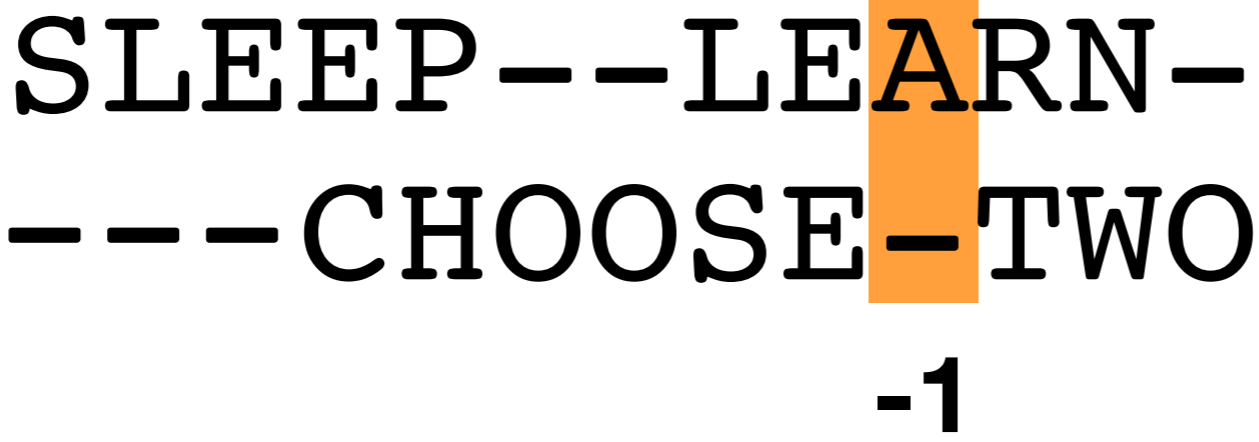
SLEEP--LEARN--
---CHOOSE-TWO

+1

Total: -5.5

Gap opening / extension

Match: +1 / Mismatch: -1 / Gap opening: -1 / Gap extension: -0.5



Total: -6.5

Gap opening / extension

Match: +1 / Mismatch: -1 / Gap opening: -1 / Gap extension: -0.5

SLEEP--LEARN--
---CHOOSE-TWO

-1

Total: -7.5

Gap opening / extension

Match: +1 / Mismatch: -1 / Gap opening: -1 / Gap extension: -0.5

SLEEP--LEARN--
---CHOOSE-TWO

-1

Total: -8.5

Gap opening / extension

Match: +1 / Mismatch: -1 / Gap opening: -1 / Gap extension: -0.5

SLEEP--LEARN-
---CHOOSE-TWO

-1

Total: -9.5

Match / Mismatch: Blosum 62

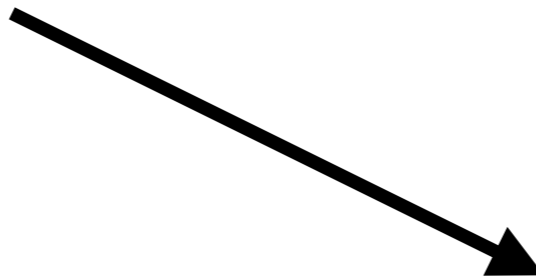
BLOcks SUBstitution Matrix

C	9																			
S	-1	4																		
T	-1	1	5																	
P	-3	-1	-1	7																
A	0	1	0	-1	4															
G	-3	0	-2	-2	0	6														
N	-3	1	0	-2	-2	0	6													
D	-3	0	-1	-1	-2	-1	1	6												
E	-4	0	-1	-1	-1	-2	0	2	5											
Q	-3	0	-1	-1	-1	-2	0	0	2	5										
H	-3	-1	-2	-2	-2	-2	1	-1	0	0	8									
R	-3	-1	-1	-2	-1	-2	0	-2	0	1	0	5								
K	-3	0	-1	-1	-1	-2	0	-1	1	1	-1	2	5							
M	-1	-1	-1	-2	-1	-3	-2	-3	-2	0	-2	-1	-1	5						
I	-1	-2	-1	-3	-1	-4	-3	-3	-3	-3	-3	-3	-3	1	4					
L	-1	-2	-1	-3	-1	-4	-3	-4	-3	-2	-3	-2	-2	2	2	4				
V	-1	-2	0	-2	0	-3	-3	-3	-2	-2	-3	-3	-2	1	3	1	4			
F	-2	-2	-2	-4	-2	-3	-3	-3	-3	-3	-1	-3	-3	0	0	0	-1	6		
Y	-2	-2	-2	-3	-2	-3	-2	-3	-2	-1	2	-2	-2	-1	-1	-1	-1	3	7	
W	-2	-3	-2	-4	-3	-2	-4	-4	-3	-2	-2	-3	-3	-1	-3	-2	-3	1	2	11
	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W

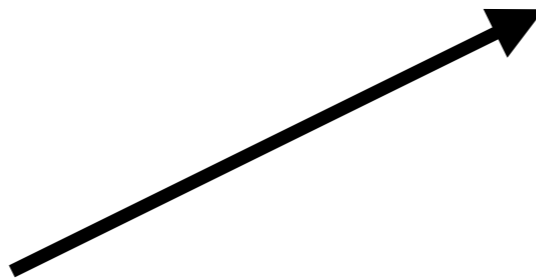
Needleman-Wunsch algorithm

Global alignments

G C A T G C U

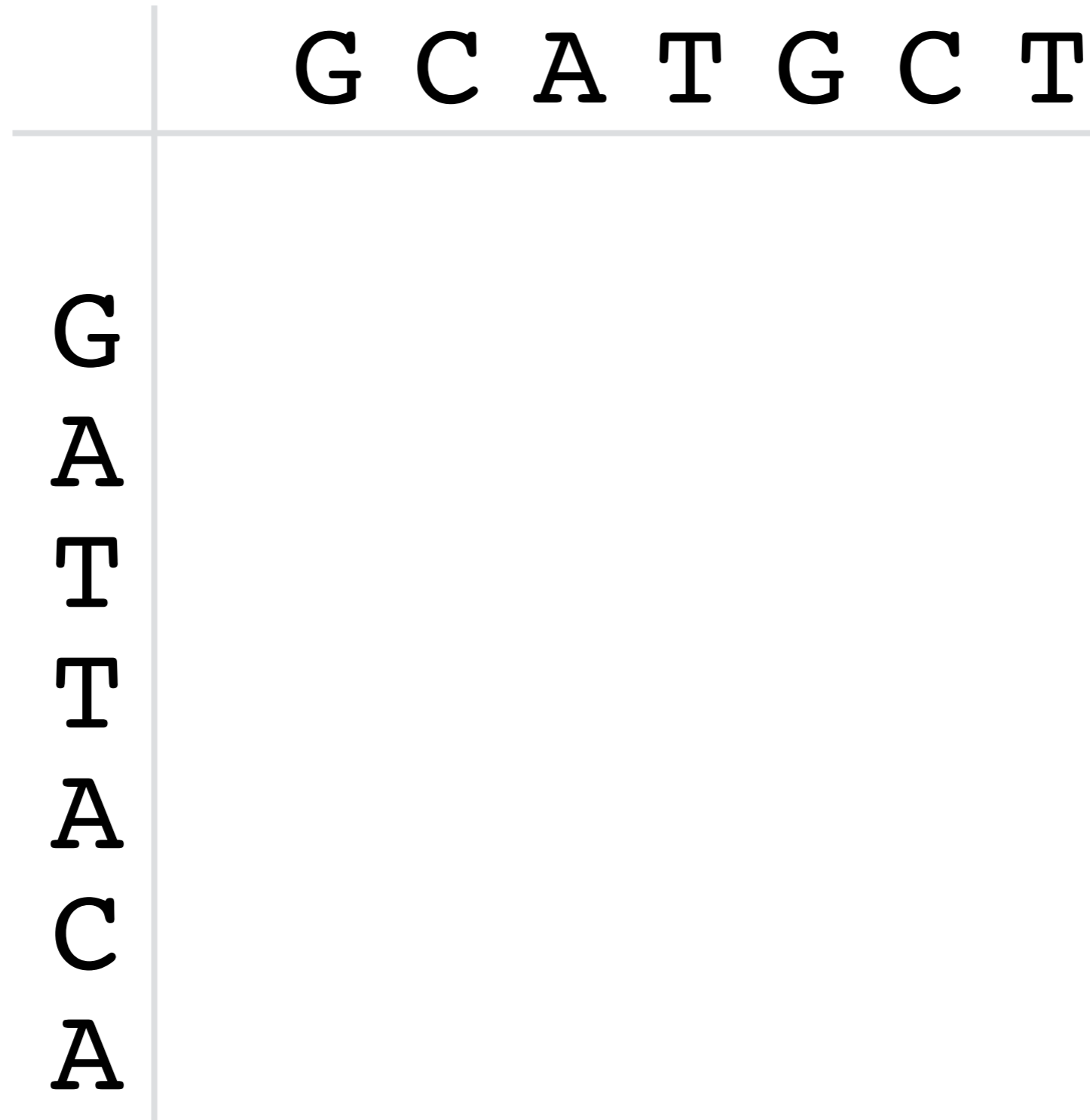


G A T T A C A



Needleman-Wunsch algorithm

Global alignments



Needleman-Wunsch algorithm

Global alignments (match: +1, mismatch: -1, indel: -1)

		G	C	A	T	G	C	T	
		0	-1	-2	-3	-4	-5	-6	-7
G		-1							
A		-2							
T		-3							
T		-4							
A		-5							
C		-6							
A		-7							

Needleman-Wunsch algorithm

Global alignments (match: +1, mismatch: -1, indel: -1)

		G	C	A	T	G	C	T
	0	-1	-2	-3	-4	-5	-6	-7
G	-1							
A	-2							
T	-3							
T	-4							
A	-5							
C	-6							
A	-7							

Needleman-Wunsch algorithm

Global alignments (match: +1, mismatch: -1, indel: -1)

		G	C	A	T	G	C	T
	0	-1	-2	-3	-4	-5	-6	-7
G	-1	?						
A	-2							
T	-3							
T	-4							
A	-5							
C	-6							
A	-7							

Needleman-Wunsch algorithm

Global alignments (match: +1, mismatch: -1, indel: -1)

		G	C	A	T	G	C	T	
		0	-1	-2	-3	-4	-5	-6	-7
G		-1	-2						
A		-2							
T		-3							
T		-4							
A		-5							
C		-6							
A		-7							

Needleman-Wunsch algorithm

Global alignments (match: +1, mismatch: -1, indel: -1)

		G	C	A	T	G	C	T
G	0	-1	-2	-3	-4	-5	-6	-7
A	-1	1						
T	-2							
T	-3							
A	-4							
C	-5							
A	-6							
	-7							

Needleman-Wunsch algorithm

Global alignments (match: +1, mismatch: -1, indel: -1)

		G	C	A	T	G	C	T
	0	-1	-2	-3	-4	-5	-6	-7
G	-1	-2						
A	-2							
T	-3							
T	-4							
A	-5							
C	-6							
A	-7							

Needleman-Wunsch algorithm

Global alignments (match: +1, mismatch: -1, indel: -1)

		G	C	A	T	G	C	T
	0	-1	-2	-3	-4	-5	-6	-7
G	-1	1						
A	-2							
T	-3							
T	-4							
A	-5							
C	-6							
A	-7							

Needleman-Wunsch algorithm

Global alignments (match: +1, mismatch: -1, indel: -1)

	G	C	A	T	G	C	T	
G	0	-1	-2	-3	-4	-5	-6	-7
A	-1	1	?					
T	-2							
T	-3							
A	-4							
C	-5							
A	-6							
A	-7							

Needleman-Wunsch algorithm

Global alignments (match: +1, mismatch: -1, indel: -1)

	G	C	A	T	G	C	T	
G	0	-1	-2	-3	-4	-5	-6	-7
A	-1	1	0					
T	-2							
T	-3							
A	-4							
C	-5							
A	-6							
A	-7							

Needleman-Wunsch algorithm

Global alignments (match: +1, mismatch: -1, indel: -1)

	G	C	A	T	G	C	T	
G	0	-1	-2	-3	-4	-5	-6	-7
A	-1	1	-2					
T	-2							
T	-3							
A	-4							
C	-5							
A	-6							
A	-7							

Needleman-Wunsch algorithm

Global alignments (match: +1, mismatch: -1, indel: -1)

	G	C	A	T	G	C	T	
G	0	-1	-2	-3	-4	-5	-6	-7
A	-1	1	-3					
T	-2							
T	-3							
A	-4							
C	-5							
A	-6							
A	-7							

Needleman-Wunsch algorithm

Global alignments (match: +1, mismatch: -1, indel: -1)

	G	C	A	T	G	C	T	
	0	-1	-2	-3	-4	-5	-6	-7
G	-1	1	0					
A	-2							
T	-3							
T	-4							
A	-5							
C	-6							
A	-7							

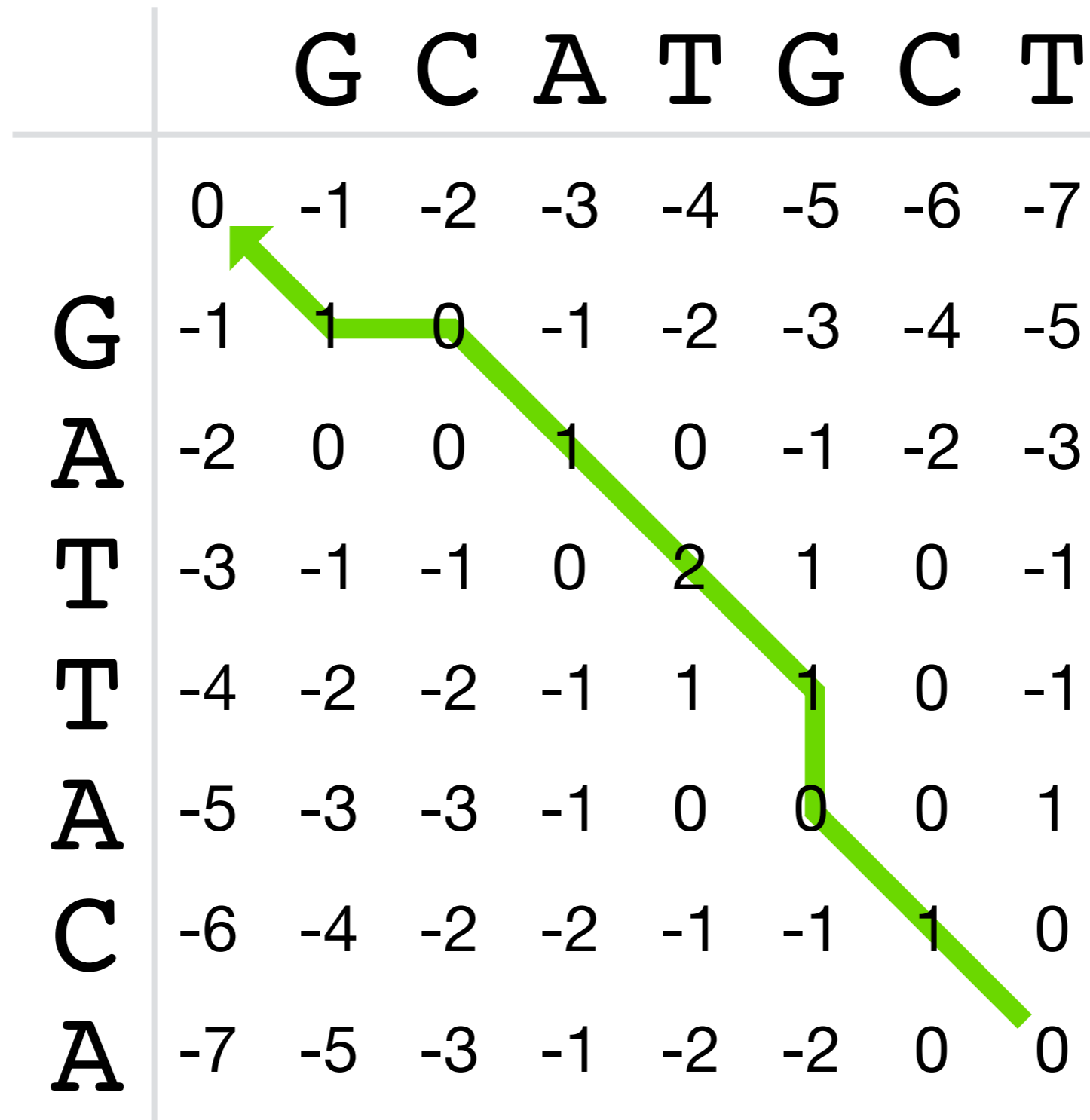
Needleman-Wunsch algorithm

Global alignments (match: +1, mismatch: -1, indel: -1)

		G	C	A	T	G	C	T
	0	-1	-2	-3	-4	-5	-6	-7
G	-1	1	0	?	?	?	?	?
A	-2	?	?	?	?	?	?	?
T	-3	?	?	?	?	?	?	?
T	-4	?	?	?	?	?	?	?
A	-5	?	?	?	?	?	?	?
C	-6	?	?	?	?	?	?	?
A	-7	?	?	?	?	?	?	?

Needleman-Wunsch algorithm

Global alignments (match: +1, mismatch: -1, indel: -1)



Needleman-Wunsch algorithm

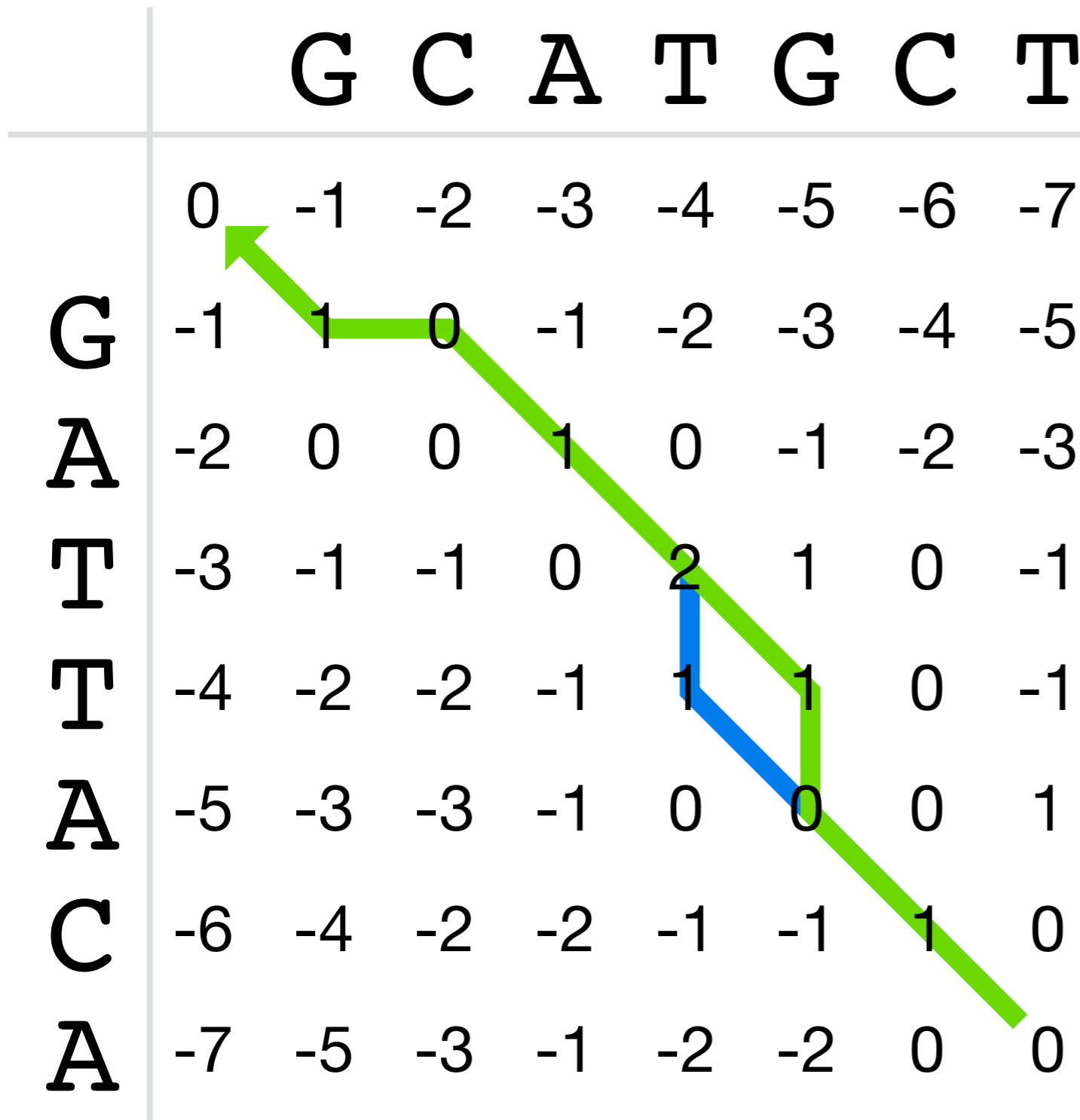
Global alignments (match: +1, mismatch: -1, indel: -1)

	G	C	A	T	G	C	T	
G	0	-1	-2	-3	-4	-5	-6	-7
A	-1	1	0	-1	-2	-3	-4	-5
T	-2	0	0	1	0	-1	-2	-3
T	-3	-1	-1	0	2	1	0	-1
A	-4	-2	-2	-1	1	1	0	-1
C	-5	-3	-3	-1	0	0	0	1
A	-6	-4	-2	-2	-1	-1	1	0
A	-7	-5	-3	-1	-2	-2	0	0

GCATG-CT
G-ATTACA

Needleman-Wunsch algorithm

Global alignments (match: +1, mismatch: -1, indel: -1)



GCATG-CT
G-ATTACA

GCAT-GCT
G-ATTACA

Smith-Waterman algorithm

Local alignments (match: +1, mismatch: -1, indel: -1)

		G	C	A	T	G	C	T
		0	0	0	0	0	0	0
G	0	<p>Maximum of:</p> <ul style="list-style-type: none">- 0- upper left value +1 (if match)- upper left value -1 (if mismatch)- max (value to left minus distance to it)- max (value above minus distance to it)						
A	0							
T	0							
T	0							
A	0							
C	0							
A	0							

Smith-Waterman algorithm

Local alignments (match: +1, mismatch: -1, indel: -1)

		G	C	A	T	G	C	T
	0	0	0	0	0	0	0	0
G	0	1	0	0	0	1	0	0
A	0	0	0	1	0	0	0	0
T	0	0	0	0	2	1	0	1
T	0	0	0	0	1	1	0	1
A	0	0	0	1	0	0	0	0
C	0	0	1	0	0	0	1	0
A	0	0	0	2	1	0	0	0

Smith-Waterman algorithm

Local alignments (match: +1, mismatch: -1, indel: -1)

		G	C	A	T	G	C	T	
		0	0	0	0	0	0	0	
G		0	1	0	0	0	1	0	
A		0	0	0	1	0	0	0	AT
T		0	0	0	0	2	1	0	AT
T		0	0	0	0	1	1	0	
A		0	0	0	1	0	0	0	
C		0	0	1	0	0	0	1	CA
A		0	0	0	2	1	0	0	CA

Benchmarks: SP score

Proportion of correctly aligned sequence pairs

	ClustalW	MAFFT	MUSCLE	Probalign	Probcons	T-Coffee
set 1	0.415	0.538	0.465	0.571	0.547	0.549
set 2	0.799	0.863	0.845	0.883	0.875	0.874
set 3	0.769	0.846	0.820	0.846	0.834	0.833
set 4	0.638	0.769	0.717	0.756	0.751	0.739
set 5	0.637	0.774	0.702	0.766	0.766	0.763

Benchmarks: Execution time

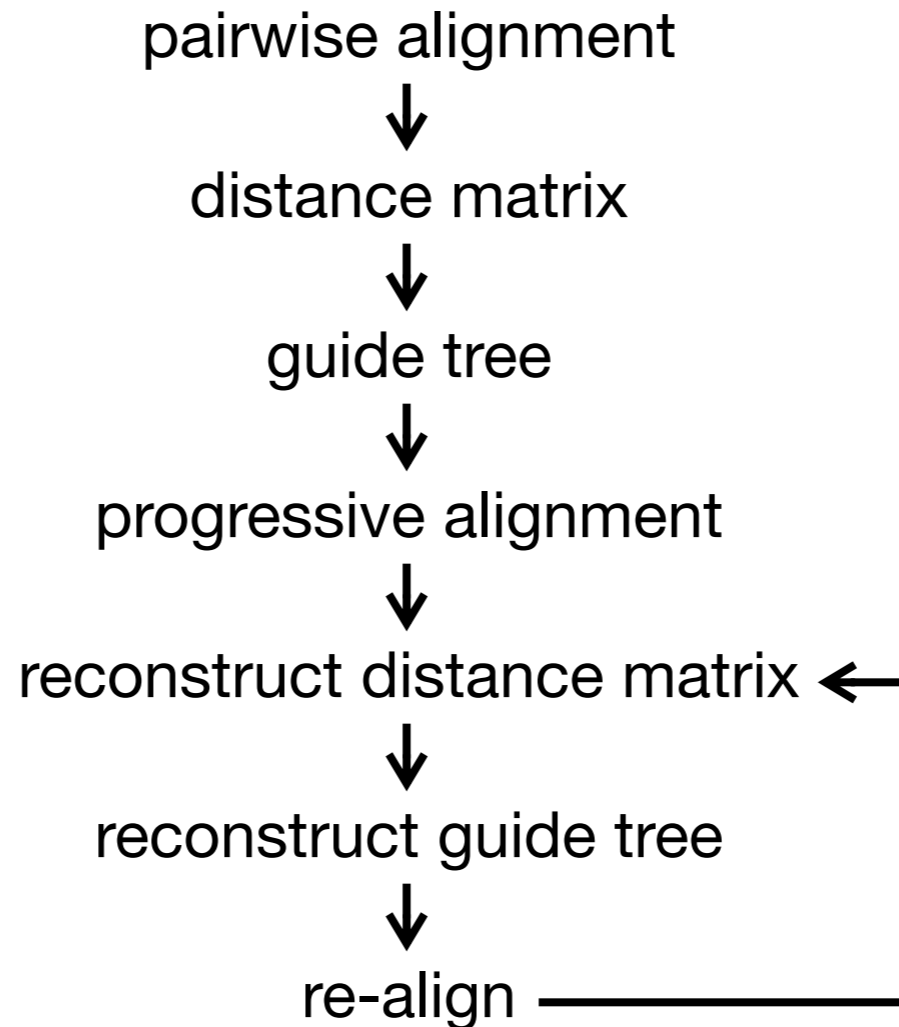
	ClustalW	MAFFT	MUSCLE	Probalign	Probcons	T-Coffee
set 1	7 sec	27 sec	24 sec	42 sec	68 sec	86 sec
set 2	26 sec	46 sec	32 sec	193 sec	243 sec	298 sec
set 3	335 sec	591 sec	259 sec	4171 sec	5170 sec	-
set 4	417 sec	585 sec	266 sec	6040 sec	8143 sec	-
set 5	90 sec	298 sec	133 sec	1370 sec	1813 sec	3805 sec

MAFFT

Multiple Alignment using Fast Fourier Transform

nucleotides are converted to four-dimensional vectors

iterative refinement



MAFFT

```
m_matschiner — bash — 80x26
ceesbob:~ m_matschiner$ mafft --help
-----
MAFFT v7.213 (2014/12/05)
http://mafft.cbrc.jp/alignment/software/
MBE 30:772-780 (2013), NAR 30:3059-3066 (2002)
-----
High speed:
% mafft in > out
% mafft --retree 1 in > out (fast)

High accuracy (for <~200 sequences x <~2,000 aa/nt):
% mafft --maxiterate 1000 --localpair in > out (% linsi in > out is also ok)
% mafft --maxiterate 1000 --genafpair in > out (% einsl in > out)
% mafft --maxiterate 1000 --globalpair in > out (% ginsi in > out)

If unsure which option to use:
% mafft --auto in > out

--op # :          Gap opening penalty, default: 1.53
--ep # :          Offset (works like gap extension penalty), default: 0.0
--maxiterate # : Maximum number of iterative refinement, default: 0
--clustalout :    Output: clustal format, default: fasta
--reorder :      Outorder: aligned, default: input order
--quiet :        Do not report progress
--thread # :     Number of threads (if unsure, --thread -1)
```

MAFFT

Needleman-Wunsch

```
m_matschiner — bash — 80x26
ceesbob:~ m_matschiner$ mafft --help
-----
MAFFT v7.213 (2014/12/05)
http://mafft.cbrc.jp/alignment/software/
MBE 30:772-780 (2013), NAR 30:3059-3066 (2002)
-----
High speed:
% mafft in > out
% mafft --retree 1 in > out (fast)

High accuracy (for <~200 sequences x <~2,000 aa/nt):
% mafft --maxiterate 1000 --localpair in > out (% linsi in > out is also ok)
% mafft --maxiterate 1000 --genafpair in > out (% einsl in > out)
% mafft --maxiterate 1000 --globalpair in > out (% ginsi in > out)

If unsure which option to use:
% mafft --auto in > out

--op # :          Gap opening penalty, default: 1.53
--ep # :          Offset (works like gap extension penalty), default: 0.0
--maxiterate # :  Maximum number of iterative refinement, default: 0
--clustalout :    Output: clustal format, default: fasta
--reorder :      Outorder: aligned, default: input order
--quiet :        Do not report progress
--thread # :     Number of threads (if unsure, --thread -1)
```

MAFFT

Smith-Waterman

```
m_matschiner — bash — 80x26
ceesbob:~ m_matschiner$ mafft --help
-----
MAFFT v7.213 (2014/12/05)
http://mafft.cbrc.jp/alignment/software/
MBE 30:772-780 (2013), NAR 30:3059-3066 (2002)
-----
High speed:
% mafft in > out
% mafft --retree 1 in > out (fast)

High accuracy (for <~200 sequences x <~2,000 aa/nt):
% mafft --maxiterate 1000 --localpair in > out (% linsi in > out is also ok)
% mafft --maxiterate 1000 --genafpair in > out (% einsl in > out)
% mafft --maxiterate 1000 --globalpair in > out (% ginsi in > out)

If unsure which option to use:
% mafft --auto in > out

--op # :          Gap opening penalty, default: 1.53
--ep # :          Offset (works like gap extension penalty), default: 0.0
--maxiterate # :  Maximum number of iterative refinement, default: 0
--clustalout :    Output: clustal format, default: fasta
--reorder :      Outorder: aligned, default: input order
--quiet :        Do not report progress
--thread # :     Number of threads (if unsure, --thread -1)
```

MAFFT

Gap openings

```
m_matschiner — bash — 80x26
ceesbob:~ m_matschiner$ mafft --help
-----
MAFFT v7.213 (2014/12/05)
http://mafft.cbrc.jp/alignment/software/
MBE 30:772-780 (2013), NAR 30:3059-3066 (2002)
-----
High speed:
% mafft in > out
% mafft --retree 1 in > out (fast)

High accuracy (for <~200 sequences x <~2,000 aa/nt):
% mafft --maxiterate 1000 --localpair in > out (% linsi in > out is also ok)
% mafft --maxiterate 1000 --genafpair in > out (% einsl in > out)
% mafft --maxiterate 1000 --globalpair in > out (% ginsi in > out)

If unsure which option to use:
% mafft --auto in > out

--op # :          Gap opening penalty, default: 1.53
--ep # :          Offset (works like gap extension penalty), default: 0.0
--maxiterate # : Maximum number of iterative refinement, default: 0
--clustalout :    Output: clustal format, default: fasta
--reorder :      Outorder: aligned, default: input order
--quiet :        Do not report progress
--thread # :     Number of threads (if unsure, --thread -1)
```

MAFFT

Gap extensions

```
m_matschiner — bash — 80x26
ceesbob:~ m_matschiner$ mafft --help
-----
MAFFT v7.213 (2014/12/05)
http://mafft.cbrc.jp/alignment/software/
MBE 30:772-780 (2013), NAR 30:3059-3066 (2002)
-----
High speed:
% mafft in > out
% mafft --retree 1 in > out (fast)

High accuracy (for <~200 sequences x <~2,000 aa/nt):
% mafft --maxiterate 1000 --localpair in > out (% linsi in > out is also ok)
% mafft --maxiterate 1000 --genafpair in > out (% einsl in > out)
% mafft --maxiterate 1000 --globalpair in > out (% ginsi in > out)

If unsure which option to use:
% mafft --auto in > out

--op # :          Gap opening penalty, default: 1.53
--ep # :          Offset (works like gap extension penalty), default: 0.0
--maxiterate # : Maximum number of iterative refinement, default: 0
--clustalout :    Output: clustal format, default: fasta
--reorder :      Outorder: aligned, default: input order
--quiet :        Do not report progress
--thread # :     Number of threads (if unsure, --thread -1)
```


MAFFT

no idea

```
m_matschiner — bash — 80x26
ceesbob:~ m_matschiner$ mafft --help

-----
MAFFT v7.213 (2014/12/05)
http://mafft.cbrc.jp/alignment/software/
MBE 30:772-780 (2013), NAR 30:3059-3066 (2002)
-----

High speed:
% mafft in > out
% mafft --retree 1 in > out (fast)

High accuracy (for <~200 sequences x <~2,000 aa/nt):
% mafft --maxiterate 1000 --localpair in > out (% linsi in > out is also ok)
% mafft --maxiterate 1000 --genafpair in > out (% einsl in > out)
% mafft --maxiterate 1000 --globalpair in > out (% ginsi in > out)

If unsure which option to use:
% mafft --auto in > out

--op # :          Gap opening penalty, default: 1.53
--ep # :          Offset (works like gap extension penalty), default: 0.0
--maxiterate # : Maximum number of iterative refinement, default: 0
--clustalout :    Output: clustal format, default: fasta
--reorder :      Outorder: aligned, default: input order
--quiet :        Do not report progress
--thread # :     Number of threads (if unsure, --thread -1)
```

MAFFT

default

```
m_matschiner — bash — 80x26
ceesbob:~ m_matschiner$ mafft --help

-----
MAFFT v7.213 (2014/12/05)
http://mafft.cbrc.jp/alignment/software/
MBE 30:772-780 (2013), NAR 30:3059-3066 (2002)
-----

High speed:
% mafft in > out
% mafft --retree 1 in > out (fast)

High accuracy (for <~200 sequences x <~2,000 aa/nt):
% mafft --maxiterate 1000 --localpair in > out (% linsi in > out is also ok)
% mafft --maxiterate 1000 --genafpair in > out (% einsl in > out)
% mafft --maxiterate 1000 --globalpair in > out (% ginsi in > out)

If unsure which option to use:
% mafft --auto in > out

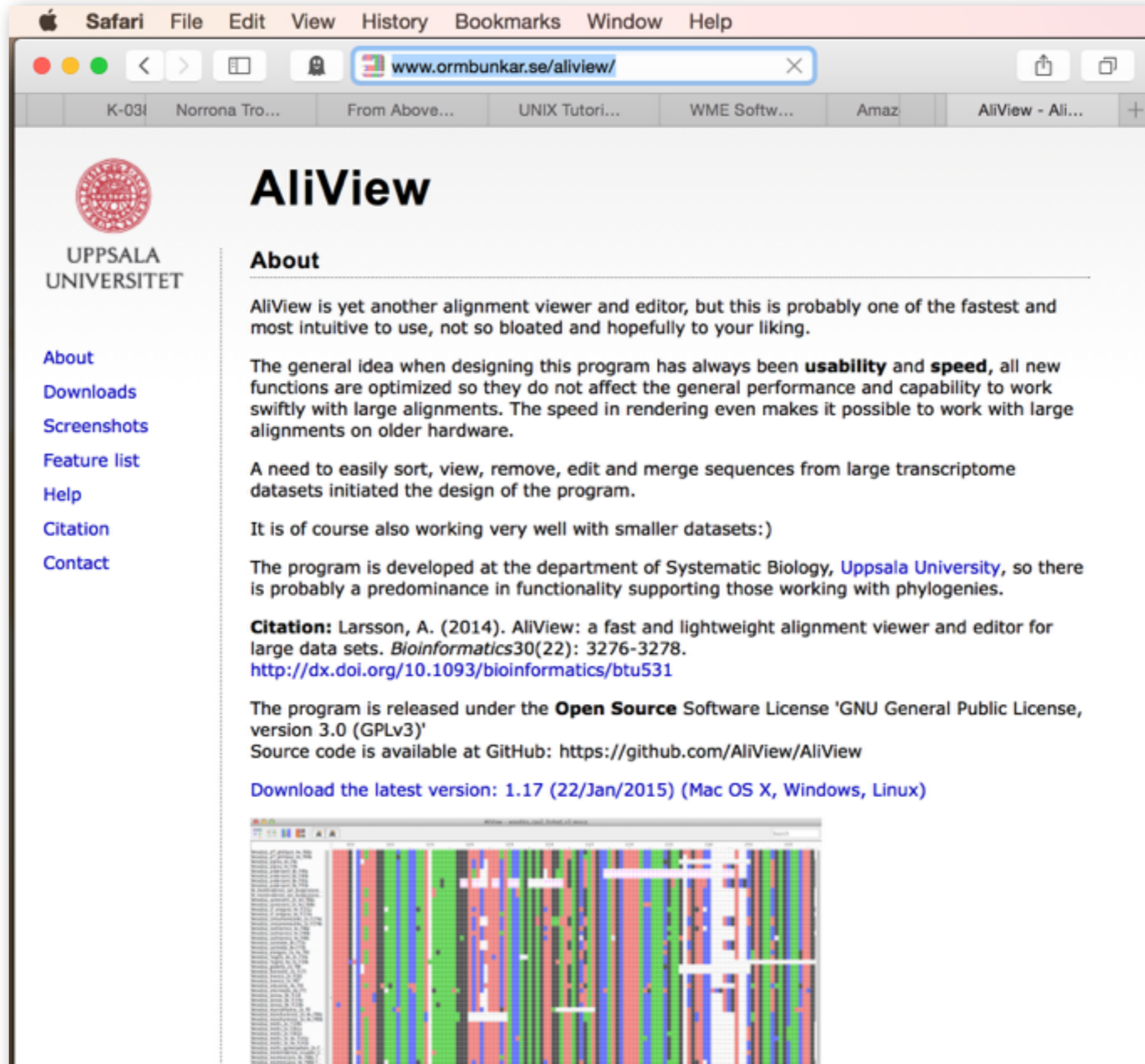
--op # :          Gap opening penalty, default: 1.53
--ep # :          Offset (works like gap extension penalty), default: 0.0
--maxiterate # : Maximum number of iterative refinement, default: 0
--clustalout :    Output: clustal format, default: fasta
--reorder :      Outorder: aligned, default: input order
--quiet :        Do not report progress
--thread # :     Number of threads (if unsure, --thread -1)
```

MAFFT

that's it.

AliView

<http://www.ormbunkar.se/aliview/>



The image shows a screenshot of a Safari browser window displaying the AliView website. The browser's address bar shows the URL www.ormbunkar.se/aliview/. The website features a navigation menu on the left with links for About, Downloads, Screenshots, Feature list, Help, Citation, and Contact. The main content area is titled "AliView" and includes an "About" section. The "About" section describes AliView as a fast and intuitive alignment viewer and editor, highlighting its usability and speed. It mentions that the program is developed at Uppsala University and is released under the GNU General Public License (GPLv3). A citation is provided for Larsson, A. (2014), and a link to the source code on GitHub is included. At the bottom, there is a link to download the latest version (1.17) for Mac OS X, Windows, and Linux. A small thumbnail image at the bottom of the page shows a screenshot of the AliView software interface, which displays a sequence alignment viewer with multiple tracks of colored bars representing different sequences.

AliView

About

AliView is yet another alignment viewer and editor, but this is probably one of the fastest and most intuitive to use, not so bloated and hopefully to your liking.

The general idea when designing this program has always been **usability** and **speed**, all new functions are optimized so they do not affect the general performance and capability to work swiftly with large alignments. The speed in rendering even makes it possible to work with large alignments on older hardware.

A need to easily sort, view, remove, edit and merge sequences from large transcriptome datasets initiated the design of the program.

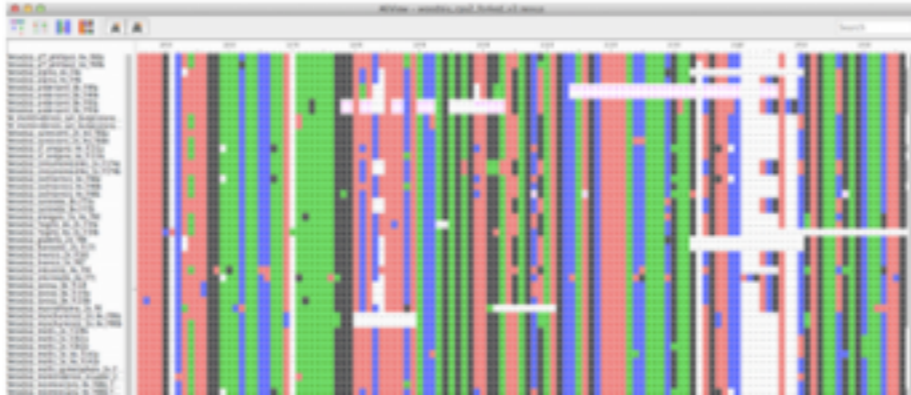
It is of course also working very well with smaller datasets:)

The program is developed at the department of Systematic Biology, [Uppsala University](http://www.ormbunkar.se), so there is probably a predominance in functionality supporting those working with phylogenies.

Citation: Larsson, A. (2014). AliView: a fast and lightweight alignment viewer and editor for large data sets. *Bioinformatics*30(22): 3276-3278.
<http://dx.doi.org/10.1093/bioinformatics/btu531>

The program is released under the **Open Source** Software License 'GNU General Public License, version 3.0 (GPLv3)'
Source code is available at GitHub: <https://github.com/AliView/AliView>

[Download the latest version: 1.17 \(22/Jan/2015\) \(Mac OS X, Windows, Linux\)](#)



AliView

Interface

AliView - 16s_aln.fasta

Search

350 360 370 380 390 400

Acanthapomotis
Acanthocataphr
Acaroninassaxx
Acropomjaponic
Aeoliscstrigat
Aequidechimant
Aequidediadema
Aequidetetrame
Aequidetubicen
Ageneioucayale
Altolamcompres
Amatitlnigrofa
Ambassispcxxxx
Ambloplrurpestr
Amblycipinosxx
Amphilispcxxxx
Amphilocitrine
Amphiprocellar
Anabasxtestudi
Anarhiclupusxx
Andinoabiseria
Andinoacoerule
Andinoapulcher
Andinoarivulat
Anoplopfimbria
Antigoncaprox
Aphredosayanus
Aphyoleperuens
Aplochepanchax
Apteronalbifro
Archocecentrar
Arhocemultisp
Archoplinteru
Aristocchristy
Arripistruttax
Astathelongima
Astathemacraca
Astatoralluud
Astronoocellat
Ateleopjaponic
Auchenocciden

Selected: Acanthapomotis | pos: 346 | pos (ungaped): 244 | Selected seqs: 1 | cols: 1 | total selected chars: 1 Alignment

AliView

Site selection



AliView - 16s_aln.fasta

Search

350 360 370 380 390 400

Acanthapomotis
Acanthocataphr
Acaroninassaxx
Acropomjaponic
Aeoliscstrigat
Aequidechimant
Aequidediadema
Aequidetetrame
Aequidetubicen
Ageneioucayale
Altolamcompres
Amatitlnigrofa
Ambassispcxxxx
Ambloplrurpestr
Amblycipinosxx
Amphilispcxxxx
Amphilocitrine
Amphiprocellar
Anabasxtestudi
Anarhiclupusxx
Andinoabiseria
Andinoacoerule
Andinoapulcher
Andinoarivulat
Anoplopfimbria
Antigoncaprox
Aphredosayanus
Aphyoleperuens
Aplochepanchax
Apteronalbifro
Archocecentrar
Arhocemultisp
Archoplinteru
Aristocchristy
Arripistruttax
Astathelongima
Astathemacraca
Astatoralluud
Astronoocellat
Ateleopjaponic
Auchenocciden

Selected: Acanthapomotis... | pos: 376 | pos (ungaped): 273 | Selected seqs: 294 | cols: 13 | total selected chars: 3822 | Align

AliView

File menu

The screenshot shows the AliView application window with the File menu open. The menu items are as follows:

- New (⌘N)
- Open File (⌘O)
- Recent Files
- Reload file (^R)
- Save (⌘S)
- Save as Fasta
- Save as Nexus
- Save as Nexus (Illegal name chars replaced by _ (e.g. for MrBayes))
- Save as codonpos Nexus (codonpos as charsets - excluded removed)
- Save as Phylip (full names & padded)
- Save as Phylip (other format versions)
- Save as Clustal (aln)
- Save as MSF/GCG
- Save selection as Fasta
- Save Translated alignment (Amino Acid) as Fasta
- Save fasta index file (.fai)
- Export alignment as image
- Print (⌘P)
- Show message log

The background shows a sequence alignment view with a search bar at the top right. The alignment is displayed in a grid format with columns numbered 390 and 400. The sequences are color-coded by nucleotide: A (green), C (blue), G (red), and T (orange). The status bar at the bottom reads: Selected: Acanthapomotis... | pos: 376 | pos (ungaped): 273 | Selected seqs: 294 | cols: 13 | total selected chars: 3822 | Align

AliView

Edit menu

The screenshot displays the AliView application window with the 'Edit' menu open. The menu items and their keyboard shortcuts are as follows:

- Undo (⌘Z)
- Redo (⌘Y)
- Copy selection as fasta (⌘C)
- Copy selection as characters (⇧⌘C)
- Copy name(s) only
- Paste (fasta-sequences) (⌘V)
- Rename sequence (F2)
- Add sequences from file
- Add new empty sequence
- Edit mode
 - Clear selected bases (⌘X)
 - Delete selected (⌘⌫)
 - Delete gap-only columns
 - Delete all gaps in all sequences
 - Trim sequences
 - Delete excluded bases
 - Delete empty sequences
- Find (⌘F)
- Find sequence names from clipboard (⌘M)
- Merge 2 selected sequences (^⌘M)
- Reverse Complement Selected Sequences
- Reverse Complement Alignment
- Complement Alignment
- Reverse Complement Clipboard

The background shows a sequence alignment with a search bar at the top right. The alignment view includes column markers at 380, 390, and 400. The status bar at the bottom reads: Selected: Acanthapomotis... | pos: 376 | pos (ungaped): 273 | Selected seqs: 294 | cols: 13 | total selected chars: 3822 | Align

AliView

Selection menu

The screenshot displays the AliView application window with the 'Selection' menu open. The menu items and their keyboard shortcuts are as follows:

- Select all (⌘A)
- De-select
- Expand selection Right (⇧⇧→)
- Expand selection Left (⇧⇧←)
- Move selected sequences up (⇧↑)
- Move selected sequences down (⇧↓)
- Move selected sequences to top (⇧↑)
- Move selected sequences to bottom (⇧↓)
- Add or Remove selection to Excludes/Exset (⌘E)
- Select Charset (dropdown arrow)
- Set selection as coding (selection starting with codon position=1)
- Set selection as coding (selection starting with codon position=2)
- Set selection as coding (selection starting with codon position=3)
- Set selection as Non-coding

The background shows a sequence alignment with a search bar on the right and a status bar at the bottom. The status bar reads: Selected: Acanthapomotis... | pos: 376 | pos (ungaped): 273 | Selected seqs: 294 | cols: 13 | total selected chars: 3822 | Align

AliView

View menu

The screenshot displays the AliView application window with the 'View' menu open. The menu items are as follows:

- Decrease Font Size -
- Increase Font Size +
- Highlight consensus characters
- Highlight Non-consensus characters
- Highlight diff from a selected sequence
- Sort sequences by name
- Sort sequences by character in selected column
- Show as translation ⌘T
- Show as translation (1 pos per AminoAcid) ⌘⇧T
- Show as Amino acid code (when show translate)
- Ignore gaps when translating ⌘⇧T
- Show codon positions on ruler
- Select genetic code for translation ▶
- Reading frame +1 ⌘⇧+
- Reading frame -1 ⌘⇧-
- Count stop codons ⌘⇧C
- Upper case
- Colors ▶

The background shows a sequence alignment with various species names on the left and a ruler at the top. The status bar at the bottom reads: Selected: Acanthapomotis... | pos: 376 | pos (ungaped): 273 | Selected seqs: 294 | cols: 13 | total selected chars: 3822 | Align

AliView

View menu

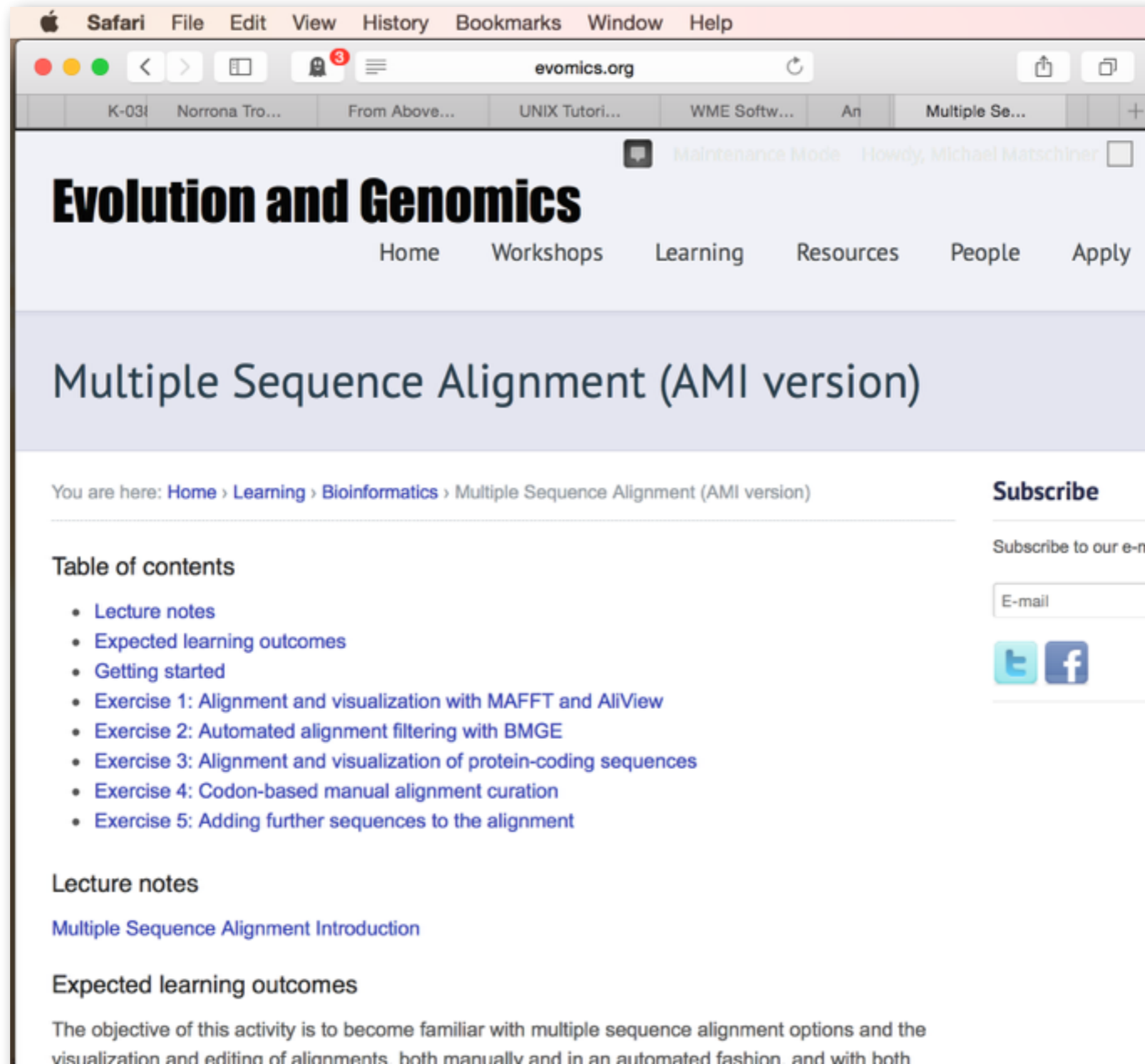
The screenshot displays the AliView application window with the 'View' menu open. The menu items are:

- Add and align sequences from clipboard (fasta)
- Add and align sequences from file (fasta)
- Realign selected block
- Realign selected sequence(s)
- Realign everything
- Realign everything as Translated Amino Acids
- Change default Aligner program
- Move selected positions right
- ← Move selected positions left
- Insert Gap move right
- Insert Gap move left
- Delete Gap at left
- Delete Gap at right

The background shows a sequence alignment with a list of species on the left and sequence positions on the top. The status bar at the bottom reads: Selected: Acanthapomotis... | pos: 376 | pos (ungaped): 273 | Selected seqs: 294 | cols: 13 | total selected chars: 3822

Multiple Sequence Alignment Exercise

<http://evomics.org/learning/bioinformatics/multiple-sequence-alignment-ami-version/>



Safari File Edit View History Bookmarks Window Help

evomics.org

K-031 Norrona Tro... From Above... UNIX Tutori... WME Softw... An Multiple Se... +

Maintenance Mode Howdy, Michael Matschiner

Evolution and Genomics

Home Workshops Learning Resources People Apply

Multiple Sequence Alignment (AMI version)

You are here: [Home](#) > [Learning](#) > [Bioinformatics](#) > Multiple Sequence Alignment (AMI version)

Table of contents

- [Lecture notes](#)
- [Expected learning outcomes](#)
- [Getting started](#)
- [Exercise 1: Alignment and visualization with MAFFT and AliView](#)
- [Exercise 2: Automated alignment filtering with BMGE](#)
- [Exercise 3: Alignment and visualization of protein-coding sequences](#)
- [Exercise 4: Codon-based manual alignment curation](#)
- [Exercise 5: Adding further sequences to the alignment](#)

Lecture notes

[Multiple Sequence Alignment Introduction](#)

Expected learning outcomes

The objective of this activity is to become familiar with multiple sequence alignment options and the visualization and editing of alignments, both manually and in an automated fashion, and with both

Subscribe

Subscribe to our e-m...

E-mail

[t](#) [f](#)