

Bioinformatics

CS300

Chap 3

**Sequence Alignment:
Investigating an Influenza Outbreak**

Fall 2019

Oliver BONHAM-CARTER



Pairwise Alignment Similarity and Relatedness

Alignment of a gene from two closely related viruses

Hemagglutinin gene from virus A: ATGAACGCAATACTCGTAGTT...

||||| ||||| |||||

Hemagglutinin gene from virus B: ATGAAGGCAATACTAGTAGTT...

Few Mismatches



Alignment of a gene from two distantly related viruses

Hemagglutinin gene from virus A: ATGAACGCAATACTCGTAGTT...

||| ||| ||| |||| | |

Hemagglutinin gene from virus C: ATGCACGAAATGCTCGGACCT...

Lots of Mismatches





Concept Questions: Discuss With Your Group

- How is similarity between genes related to the biological concept of descent from a common ancestor?
- The influenza virus mutates so rapidly that you would likely be able to identify at least a couple of mutations over the length of the complete virus genome even if you sequenced two viruses from two different patients within the same influenza outbreak. What do you think might be some considerations in deciding whether two viruses with different genome sequences actually represent two different strains?
- RNA viruses (retro viruses) are prone to genetic mutations during replication. After a mutation, parts of their genetics have been completely changed. Is it still possible to study their genetics (over long periods of time) given their code will quickly change? Why or why not?

THINK



What is Sequence Alignment?

- Sequence alignment is a way of arranging the sequence of genetic material (DNA, RNA or protein) to identify regions of similarity that may be a consequence of functional, structural or evolutionary relationships between the sequences.

How much of this gene
can be found in
the other sequence?



What is Global Sequence Alignment?

- We search for matches, matches and gaps between two sequences to determine their relatedness.
- (*) indicate matches or similar nucleotides along sequence
- Here, the sequences may share some common ancestor

ACGTACT
ACTACGT
** *

ACGTAC-T
AC-TACGT
** *** *

ACGTACT----
----ACTACGT



Ex: Sequence Alignment of Some Organisms

- We compare protein samples from several different organisms.

	*																																*	
Human	W	N	Q	S	T	A	R	W	L	R	R	L	V	F	Q	H	S	R	A	W	P	L	L	Q	T	F	A	F	S	A	W	W	H	G
Pig	W	N	H	S	T	A	Q	W	L	R	R	L	V	F	Q	Q	G	R	T	W	P	L	L	Q	T	F	V	F	S	A	W	W	H	G
Cow	W	N	Q	S	T	A	R	W	L	R	R	L	V	F	Q	Q	R	R	T	W	P	L	L	Q	T	F	L	F	S	A	W	W	H	G
Dog	W	N	Q	S	T	A	R	W	L	R	R	L	V	F	Q	Q	R	R	T	W	P	L	L	Q	T	F	L	F	S	A	W	W	H	G
Rat	W	N	R	S	T	A	Q	W	L	K	R	L	V	F	Q	R	S	R	R	W	P	V	L	Q	T	F	A	F	S	A	W	W	H	G
Mouse	W	N	R	S	T	A	L	W	L	R	R	L	V	F	R	K	S	R	R	W	P	L	L	Q	T	F	A	F	S	A	W	W	H	G
Chicken	W	N	R	S	T	S	L	W	L	R	R	L	V	F	Q	R	C	P	V	Q	P	L	L	A	T	F	A	F	S	A	W	W	H	G
Zebrafish	W	N	Q	T	T	V	D	W	L	R	K	I	V	F	N	R	T	S	R	S	P	L	F	M	T	F	G	F	S	A	L	W	H	G



Ex: Sequence Alignment of Many Organisms

- The common code have same nucleotides but there are still breaks in these common regions.

Q5E940_BOVIN	-----MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKOMQIIRMSLRGK-AVVLGMGKNTMMRKAIRGHLENN--PALE	76
RLA0_HUMAN	-----MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKOMQIIRMSLRGK-AVVLGMGKNTMMRKAIRGHLENN--PALE	76
RLA0_MOUSE	-----MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKOMQIIRMSLRGK-AVVLGMGKNTMMRKAIRGHLENN--PALE	76
RLA0_RAT	-----MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKOMQIIRMSLRGK-AVVLGMGKNTMMRKAIRGHLENN--PALE	76
RLA0_CHICK	-----MPREDRATWKSNYFMKIIQLLDDYPKCFVVGADNVGSKOMQIIRMSLRGK-AVVLGMGKNTMMRKAIRGHLENN--PALE	76
RLA0_RANSY	-----MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKOMQIIRMSLRGK-AVVLGMGKNTMMRKAIRGHLENN--SALE	76
Q7ZUG3_BRARE	-----MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKOMQIIRMSLRGK-AVVLGMGKNTMMRKAIRGHLENN--PALE	76
RLA0 ICTPU	-----MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKOMQIIRMSLRGK-AVVLGMGKNTMMRKAIRGHLENN--PALE	76
RLA0 DROME	-----MVRENKAAWKAQYFIKVVLFDEFPPKCFIVGADNVGSKOMQIIRMSLRGK-AVVLGMGKNTMMRKAIRGHLENN--PALE	76
RLA0 DICDI	-----MSGAG-SKRKKLFIEKATKLFTTYDKMIVAEADFVGSSQLQKIRKSIRGI-GAVLMGKKTMRKVIRDLADSK--PELD	75
Q54LP0 DICDI	-----MSGAG-SKRKNVFIEKATKLFTTYDKMIVAEADFVGSSQLQKIRKSIRGI-GAVLMGKKTMRKVIRDLADSK--PELD	75
RLA0 PLAF8	-----MAKLSKQKKQMYIEKLSSLIQQYSKILIVHVDNVGSKOMQIIRMSLRGK-ATILMGKNTMRIRALKKNLQAV--PQIE	76
RLA0 SULAC	-----MIGLAVTTTTKKIAKWKVDEVAELTEKLKTHKTIIANIEGFPADKLHEIRKKLRGK-ADIKVTKNLNFNIALKNAG----YDTK	79
RLA0 SULTO	-----MRIMAVITQERKIAKWKIEEVKELEOKLREYHTIIIANIEGFPADKLHDIRKKMRGM-AEIKVTKNLTLFGIAAKNAG----LDVS	80
RLA0 SULSO	-----MKRLALALKQRKVASWKLIEVKELTELKNSNTILIGNLEGFPADKLHEIRKKLRGK-ATIKVTKNLTLFKIAAKNAG----IDIE	80
RLA0 AERPE	MSVVSIVGQMYKREKPIPEWKTLMLELEELFSKHRVVLFAFLTGTPTFVVQVRVKKLWKK-YPMMAVAKKRIILRAMKAAGLE--LDDN	86
RLA0 PYRAE	MMLAIGKRRYVTRQYPARKVKIVSEATELLQKYPYVFLFDLHGLSSRILHEYRYRLRRY-GVIKIIKPTLFLKIAFTKVYGG--IPAE	85
RLA0 METAC	-----MAEERHHTTEHIPQWKKDEIENIKELIQSHKVFVGMVIEGILATKMKIRRDLDV-AVLKVSNTLTERALNQLG----ETIP	78
RLA0 METMA	-----MAEERHHTTEHIPQWKKDEIENIKELIQSHKVFVGMVIEGILATKMKIRRDLDV-AVLKVSNTLTERALNQLG----ESIP	78
RLA0 ARCFU	-----MAAVRGS---PPEYKVRAVEEIKRMISSEKPVVAIVSFRNVPAGOMKIRREFRKG-AEIKVVKNLTLERALDALG----GDYL	75
RLA0 METKA	MAVKAKGQPPSGYE PKVAEWWKREVEKELKLMDEYENVGLVDLEGIPAPOLQEIARAKLRERDTIIRMSRNTLMRIALEEKLDER--PELE	88
RLA0 METTH	-----MAHVAEWKKKEVQELHDLIKGYEVVGIANLADIPAROLQKMRQTLRDS-ALIRMSKKTLLISLALEKAGREL--ENVD	74
RLA0 METTL	-----MITAESEHKIAPWKIEEVNKLKELLKNGQIVALVDMMVEVPAROLQEIARDKIR-GTMTLKMSRNTLIERAIEVAEETGNPEFA	82
RLA0 METVA	-----MIDAKSEHKIAPWKIEEVNALKELLKSANVIALIDMVEVPAVOLQEIARDKIR-DQMTLKMSRNTLIKRAVEEVAEETGNPEFA	82
RLA0 METJA	-----METKVKAHVAPWKIEEVKTLKGLIKSKPVVAIVDMMVPAVOLQEIARDKIR-DKVKLMSRNTLIERALKEAAEELNNPKLA	81
RLA0 PYRAB	-----MAHVAEWKKKEVEELANLIKSYVIALVDVSSMPAYPLSQMRRLIRENGGLLRVSRNTLIELAIKKAQAELGKPELE	77
RLA0 PYRHO	-----MAHVAEWKKKEVEELAKLIKSYVIALVDVSSMPAYPLSQMRRLIRENGGLLRVSRNTLIELAIKKAQAELGKPELE	77
RLA0 PYRFU	-----MAHVAEWKKKEVEELANLIKSYVIALVDVSSMPAYPLSQMRRLIRENGGLLRVSRNTLIELAIKKAQAELGKPELE	77
RLA0 PYRKO	-----MAHVAEWKKKEVEELANLIKSYVIALVDVAGVPAVPLSKMRDKLR-GKALLRVSRNTLIELAIKRAQAELGQPELE	76
RLA0 HALMA	-----MSAESERKTETIPEWKQEEVDAIVEMIESYESVGVVNIAGIPSRLOQDMRRDLHGT-AELRVSRNTLIERALDDVD----DGLE	79
RLA0 HALVO	-----MSESEVRQTEVIPQWKREEVDLVDLIESYESVGVVGVAGIPSRLOQSMRRELHGS-AAVRMSRNTLVNRRALDEVN----DGFE	79
RLA0 HALSA	-----MSAEEQRTTEEVPEWKRQEVAEVLDDLETYSVGVVNVVTGIPSKLOQDMRRGLHGT-AALRVSRNTLIERALEEAG----DGLD	79
RLA0 THEAC	-----MKEVSQKKELVNEITRIKASRSVAIVDTAGIRTRQIDIRGKNRKG-INLKVIKKTLLFKALDENLGD----EKLS	72
RLA0 THEVO	-----MRKINPKKKEIVSELAODITKSKAVAIYDIKGVRTROMQDIRAKNRDK-VKIKVVKKTLLFKALDSIND----EKLT	72
RLA0 PICTO	-----MTEPAQWKIDFVKNLENEINSRKVAAIVSIRKLRNNEFQKIRNSIRDK-ARIKVSRRARLLRLAIENTGK----NNIV	72
ruler	1.....10.....20.....30.....40.....50.....60.....70.....80.....90	



Needleman-Wunsch Algorithm Background

- Global Alignment: Used to determine which parts of a sequence (inside the sequence) are shared (common) with another sequence.
- Developed by Saul B. Needleman and Christian D. Wunsch in 1970.
- Dynamic programming to find optimal solution for matching the characters of the two sequences.



Global Pairwise Alignment

- Steps to begin
 - Initialization of the matrix
 - Calculation of the scores given for a character by character comparison.
 - Filling in a system of arrows in a trace-back matrix to uncover a path back to the start in the score matrix.
 - Deducing the alignment by following the arrows in the trace-back matrix.

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

Needleman-Wunsch Algorithm

-	-	A	T	C	G	A	C
-	0	-4	-8	-12	-16	-20	-24
C	-4	-3	-7	-3	-7	-11	-15
A	-8	1	-3	-7	-6	-2	-6
T	-12	-3	6	2	-2	-6	-5
A	-16	-7	2	3	-1	3	-1
C	-20	-11	-2	-1	0	-1	8

- Create N x M matrix and place each sequence along one axis
- Place score 0 at the up-left corner
- Fill in 1st row & column with gap penalty multiples
- Fill in the matrix with max value of 3 possible moves:
 - Vertical move: Score + gap penalty
 - Horizontal move: Score + gap penalty
 - Diagonal move: Score + match/mismatch score
- To reconstruct the optimal alignment, trace back where the max at each step came from, stop when hit the origin.



Terms

- Alignment is divided up into sub problems
- Solutions are scored; the best solutions for char by char comparison are kept in the overall solution.
- **Match** – bases of each sequence at position ARE same
- **Mismatch** – bases of each sequence at position are NOT same
- **Gap** – bases are not the same, some insert or deletion may have occurred.

AGGCTATCACCTGACCTCCAGGCCGATGCCC
TAGCTATCACGACCGCGGTCGATTGCCCCGAC

–AGGCTATCACCTGACCTCCAGGCCGA––TGCCC––
TAG–CTATCAC––GACCGC––GGTCGATTGCCCCGAC



For each Element: Three Calculations

- Recursion, based on the principle of optimality:

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

The pseudo-code for the algorithm to compute the F matrix therefore looks like this:

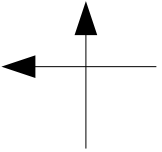
```
for i=0 to length(A)
  F(i,0) ← d*i
for j=0 to length(B)
  F(0,j) ← d*j
for i=1 to length(A)
  for j=1 to length(B)
  {
    Match ← F(i-1,j-1) + S(Ai, Bj)
    Delete ← F(i-1, j) + d
    Insert ← F(i, j-1) + d
    F(i,j) ← max(Match, Insert, Delete)
  }
```



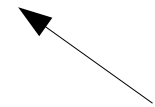
Let's Calculate!

	_	A	T	C	G
_					
T					
C					
A					

- Gap: -1



- Match: 1
- Mismatch: 0

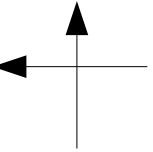




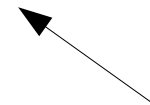
Add The Outer Values

		A	T	C	G
	—				
—	0	-1	-2	-3	-4
T	-1				
C	-2				
A	-3				

- Gap: -1



- Match: 1
- Mismatch: 0

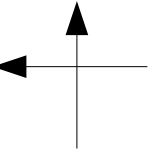




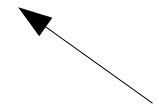
Mismatch: $A \neq T$

	—	A	T	C	G
—	0	-1	-2	-3	-4
T	-1				
C	-2				
A	-3				

- Gap: -1



- Match: 1
- Mismatch: 0



Upperbox: $-1 - 1$
 $= -2$

Sidebox: $-1 - 1$
 $= -2$

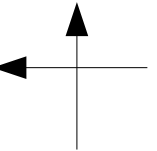
Diag: $0 - 0$
 $= 0$



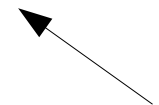
Mismatch: $A \neq T$

	—	A	T	C	G
—	0	-1	-2	-3	-4
T	-1	0			
C	-2				
A	-3				

- Gap: -1



- Match: 1
- Mismatch: 0



Upperbox: $-1 - 1$
 $= -2$

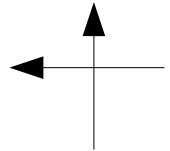
Sidebox: $-1 - 1$
 $= -2$

Diag: $0 - 0$ **Max**
 $= 0$ **value**

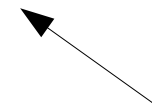
Match: T = T

	_	A	T	C	G
_	0	-1	-2	-3	-4
T	-1	0	0		
C	-2				
A	-3				

• Gap: -1



- Match: 1
- Mismatch: 0



Upperbox: $-2 - 1$
= -3

Sidebox: $0 - 1$
= -1

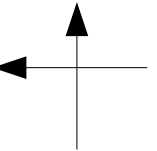
Diag: $-1 + 1$ **Max value**
= 0



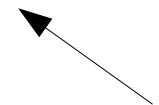
Mismatch: **T** != **C**

		A	T	C	G
	—	0	-1	-2	-3
T	-1	0	0	-1	
C	-2				
A	-3				

- Gap: -1



- Match: 1
- Mismatch: 0



Upperbox: $-3 - 1$
 $= -4$

Sidebox: $0 - 1$
 $= -1$ **Max value**

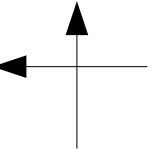
Diag: $-2 + 0$
 $= -2$



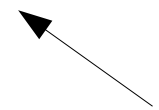
Mismatch: **T** != **G**

	_	A	T	C	G
_	0	-1	-2	-3	-4
T	-1	0	0	-1	-2
C	-2				
A	-3				

- Gap: -1



- Match: 1
- Mismatch: 0



Upperbox: $-4 - 1$
 $= -5$

Sidebox: $-1 - 1$
 $= -2$ **Max value**

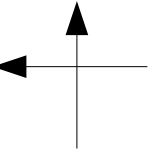
Diag: $-3 - 0$
 $= -3$



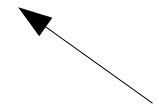
Filling in The Rest of the Values

	_	A	T	C	G
_	0	-1	-2	-3	-4
T	-1	0	0	-1	-2
C	-2	-1	0	1	0
A	-3	-1	-1	0	1

- Gap: -1



- Match: 1
- Mismatch: 0



Alignment:

A T C G
_ T C A

Follow the Arrows Back To Find the Sequence Alignment

Sequence 1

Sequence 2

Match Score Mismatch Score Gap Score

A	T	C	G
-	T	C	A

Score = 1

		A	T	C	G
	0	-1	-2	-3	-4
T	-1	0 ↖	0 ↖	-1 ←	-2 ←
C	-2	-1 ↖	0 ↖	1 ↖	0 ←
A	-3	-1 ↖	-1 ↖	0 ↖	1 ↖



To Get the Alignment

- With each calculation, we placed an arrow to show how the score was calculated and to give us the actual alignment.

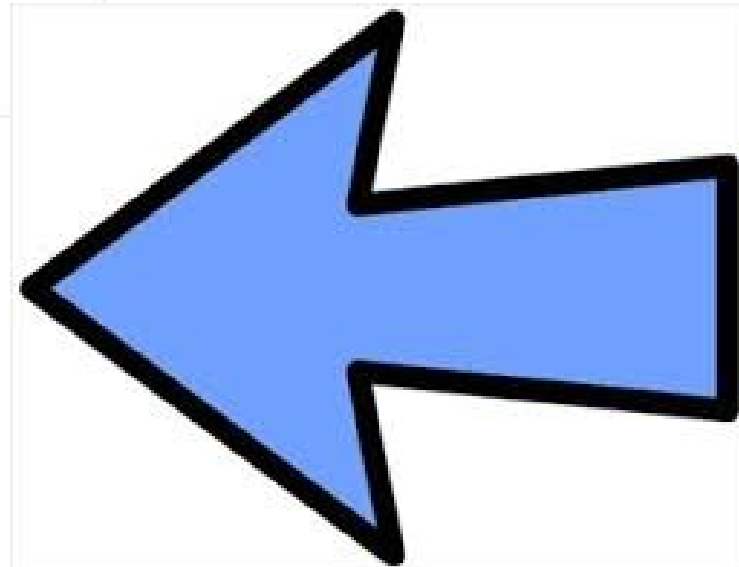
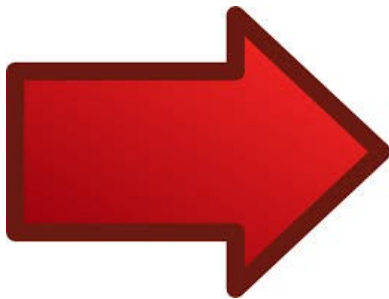
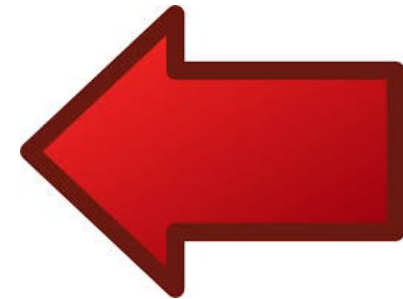
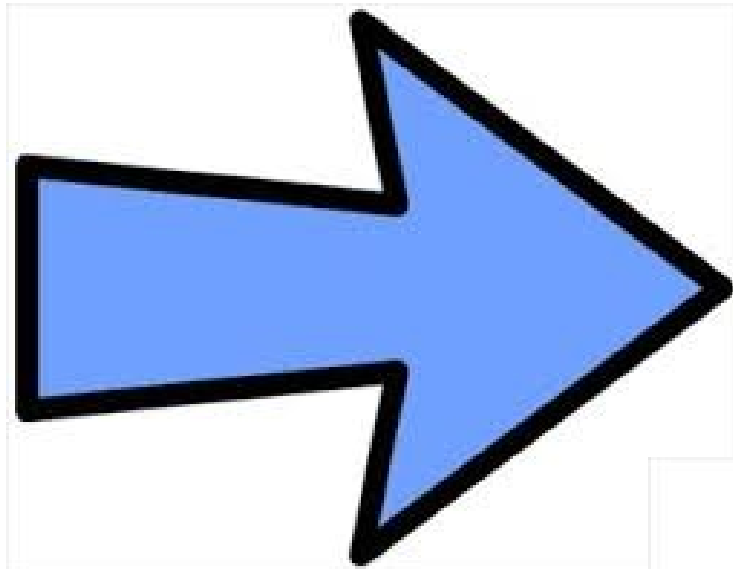
How do we read
this output?

Alignment:

A	T	C	G
_	T	C	A



Up Ahead: More Examples of Arrows





Follow the Arrows!

Sequence 1

CGCA

Sequence 2

CACGTAT

Match Score

Mismatch Score

Gap Score

1

0

-1

Compute Optimal Alignment

Clear Path

Custom Path

C A C G T A

C G C - - A

Score = 0

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

- S1 = CGCA
- S2 = CACGTAT



Example

Alignment score = 0

Let:

Match = +1

Mismatch = 0

Gap = -1

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



Example

Alignment score = 0

Let:

Match = +1

Mismatch = 0

Gap = -1

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

CACGTAT

--CGCA--



Example

Alignment score = 0

Let:

Match = +1

Mismatch = 0

Gap = -1

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

CACGTAT

C--GCA-



Example

Alignment score = 0

Let:

Match = +1

Mismatch = 0

Gap = -1

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

CACGTAT

CGC--A-



Example

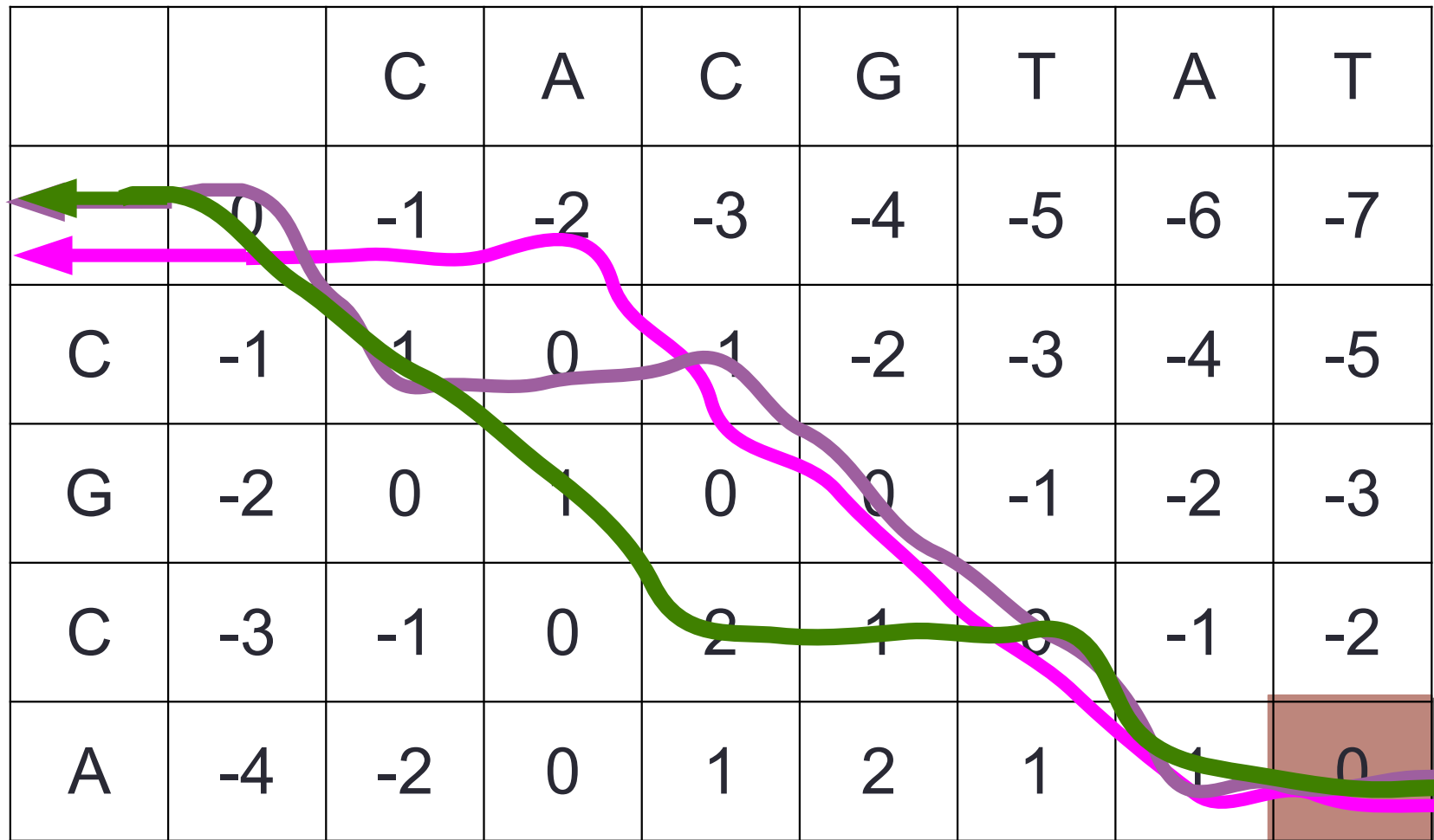
Alignment score = 0

Let:

Match = +1

Mismatch = 0

Gap = -1



CACGTAT
--CGCA--

CACGTAT
C--GCA--

CACGTAT
CGC--A--