Mol Bio 2: lectures 4 and 5

Sequence alignment

Substitution matrices

Multiple sequence alignment

BLAST

How sequences evolve

- point mutations (single base changes)
- deletion (loss of residues within the sequence)
- insertion (gain of residue within the sequence)
- truncation (loss of either end)
- extension (gain of residues at either end)

Mechanisms of insertion or extension:

- duplication or whole gene or domain
- polymerase "stutter"
- transposable element
- more??

How evolution is measured

- point mutations .... substitution matrix score
- insertion/deletion .... gap penalty
- truncation/extension .... end gap penalty

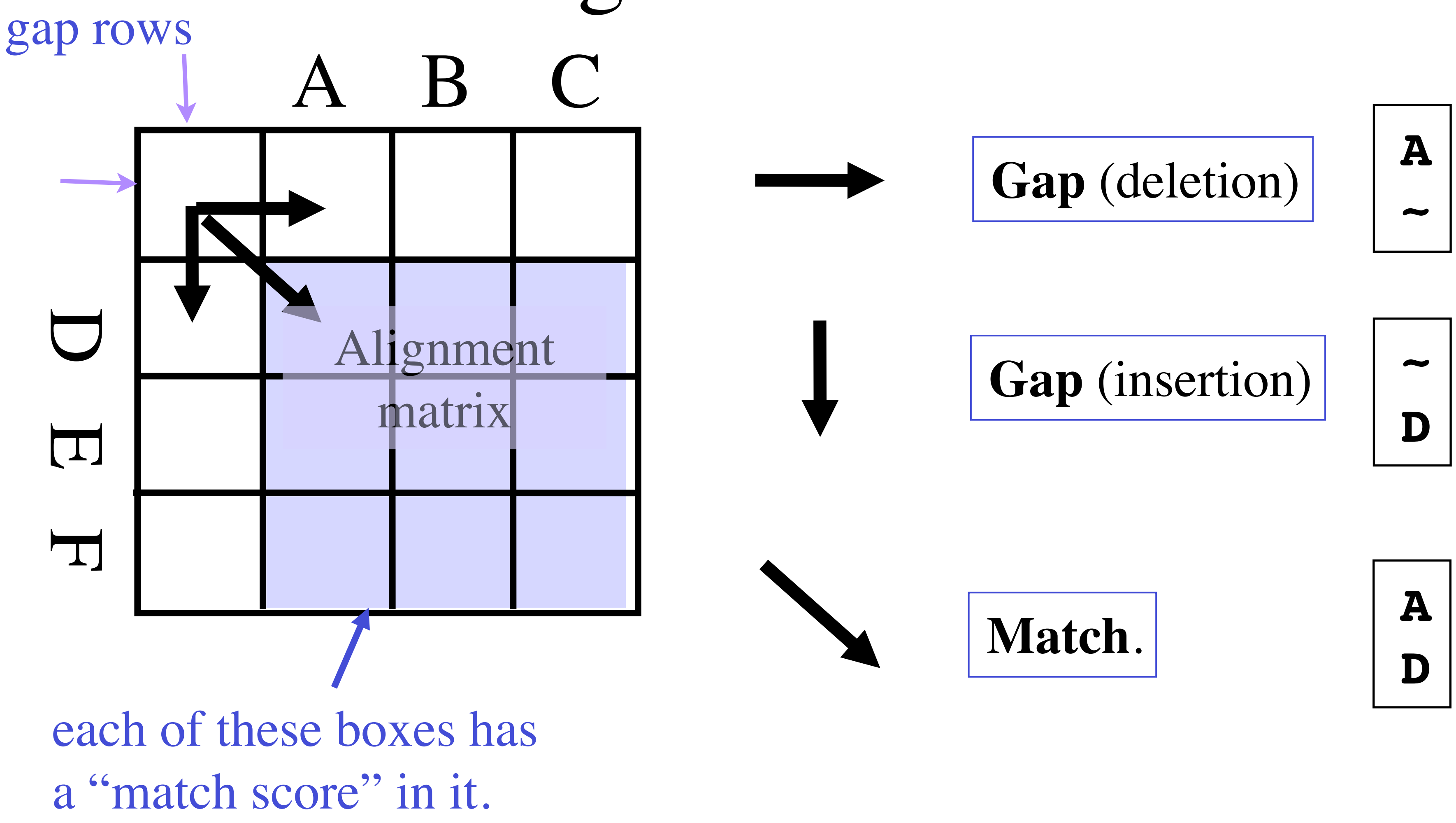
```
~ L I G H T N I ~ N G
A L I G ~ ~ N M E N T
```

Yes, an **alignment algorithm** is really
A Model for Sequence Evolution!

> *That means the way we do alignment should be closely aligned to what we know about how things evolve.*

- point mutations .... relatively frequent, usually bad
- deletion .... infrequent, always bad, location dependent
- insertion .... infrequent, always bad, location dependent
- truncation .... frequent, not so bad
- extension .... frequent, not so bad

An Alignment as a Path through the Alignment Matrix



gap rows

each of these boxes has
a "match score" in it.

Gap (deletion)

A
~

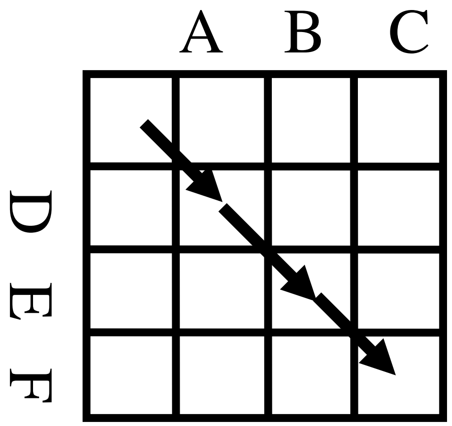
Gap (insertion)

~
D

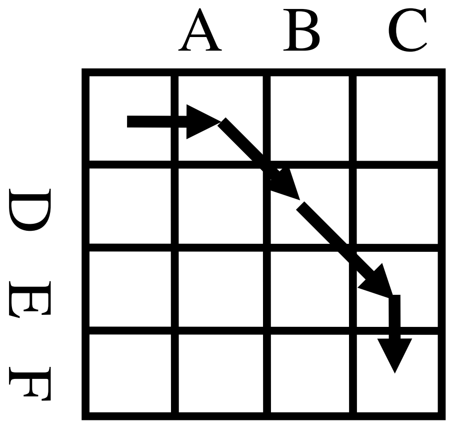
Match.

A
D

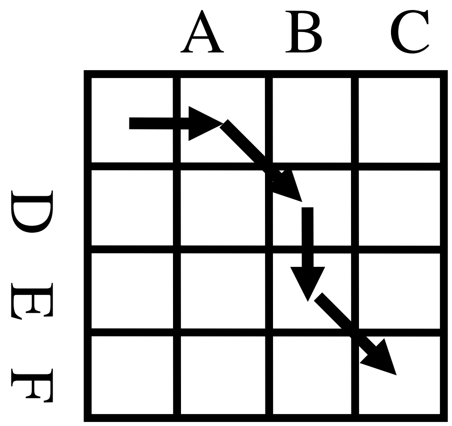
A walk through the alignment matrix



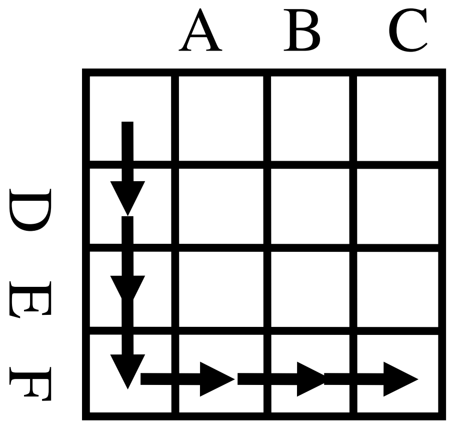
```
ABC
DEF
```



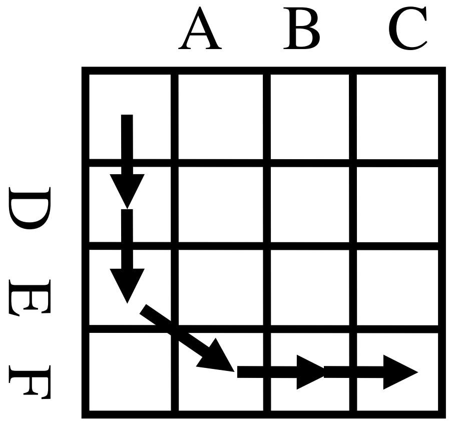
```
ABC~
~DEF
```



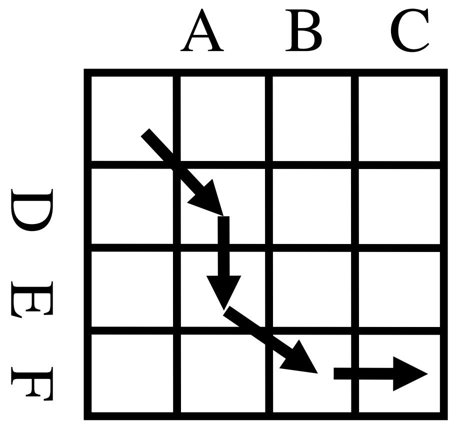
```
AB~C
~DEF
```



```
~~~ABC
DEF~~~
```

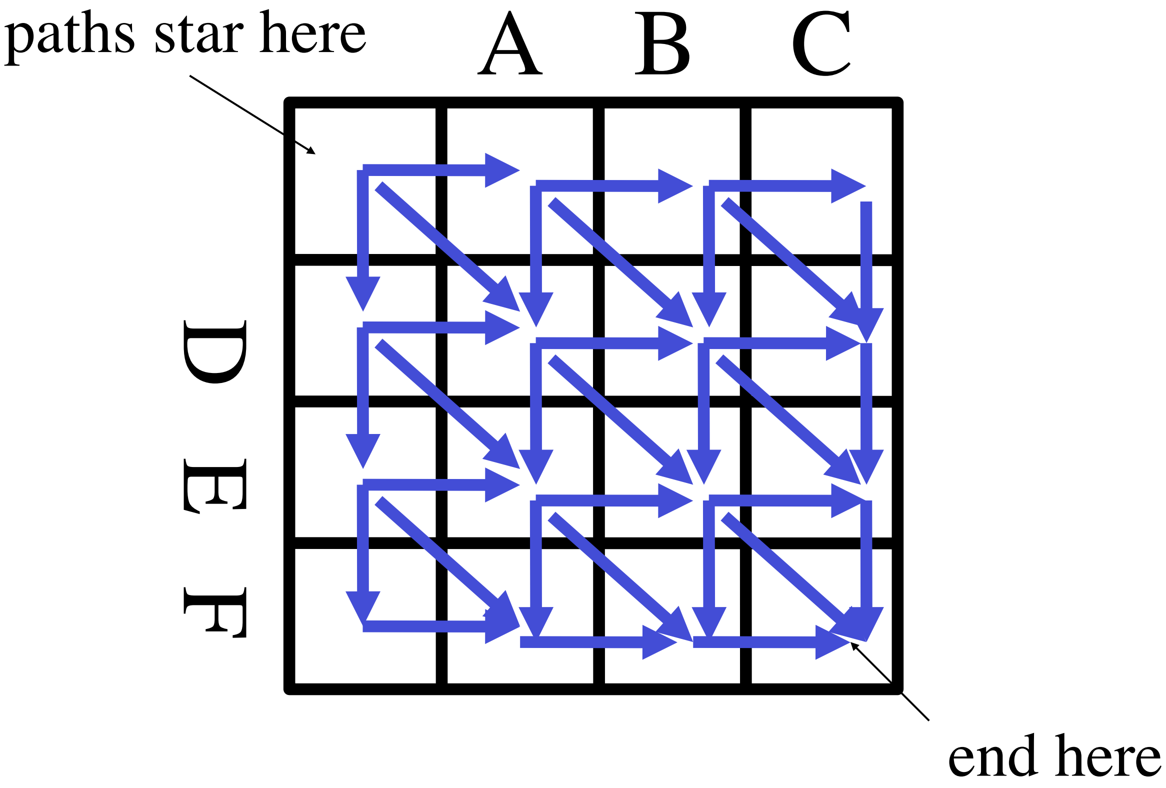


```
~~ABC
DEF~~
```



```
A~BC
DEF~
```

All possible arrow paths = all possible alignments



paths star here

end here

“Dynamic Programming”

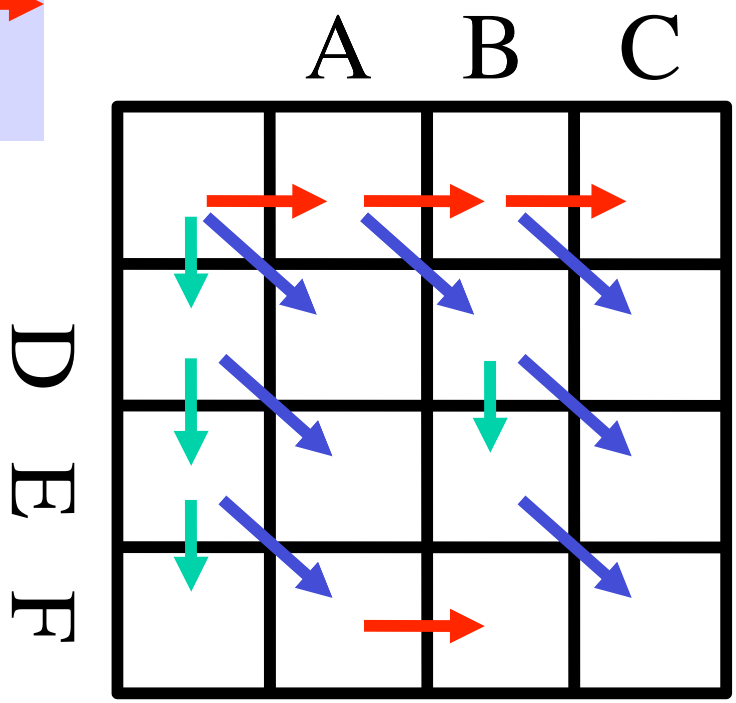
- easiest form: known as Needleman-Weunch alignment

Step 1: For each box, keep the highest scoring arrow.

$S_{i,j} = \max \{ S_{i-1,j-1} + s(i,j),$ ↘

$S_{i-1,j}$ - gap, →

$S_{i,j-1}$ - gap } ↓

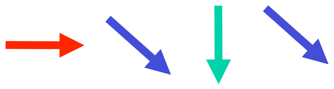


Traceback

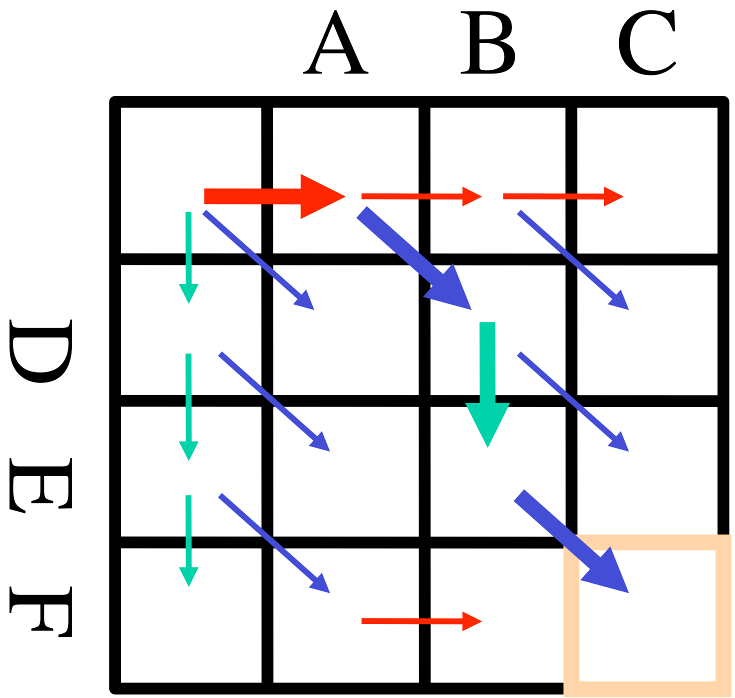
Step 2: Trace arrows back to start.

Step 3: Alignment is constructed from the traceback arrows.

Traceback starts from the **last box**



```
AB~C
~DEF
```



Try it: dynamic programming

Match=score in lower right Gap penalty = 1

		A	D	G	T	F	R	M	G	G
	0									
D		-2	6	-1	-1	-3	-2	-3	-1	-1
G		0	-1	6	-2	-3	-2	-3	6	6
Y		-2	-3	-3	-2	3	-2	-1	-3	-3
R		-1	-2	-2	-1	-3	5	-1	-2	-2
I		-2	-3	-4	-1	0	-3	1	-4	-4
G		6	-1	6	-2	-3	-2	-3	6	6

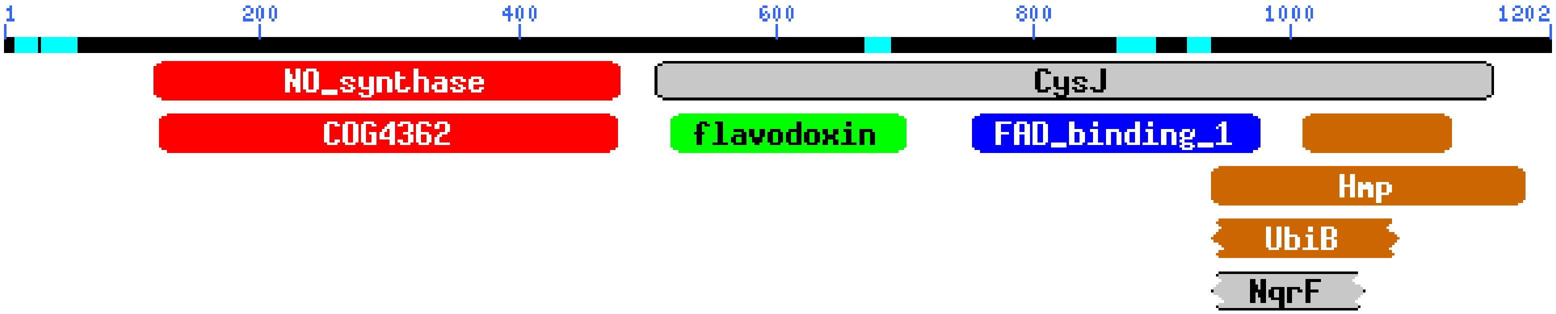
Does gap-to-gap make sense???

Special rules may apply for going directly from *insertion to deletion arrow*.

```
AGGCTACT~TATCA
 GGCTACTA~ATCA
```

I to D can simply be **disallowed** in the DP algorithm. Most programs do this.

Extension/truncation and end gaps



If we penalize end gaps, what happens to the score of *this* the **true** alignment?

So, do "end gaps" (extension/truncations) have a strong negative selective pressure?

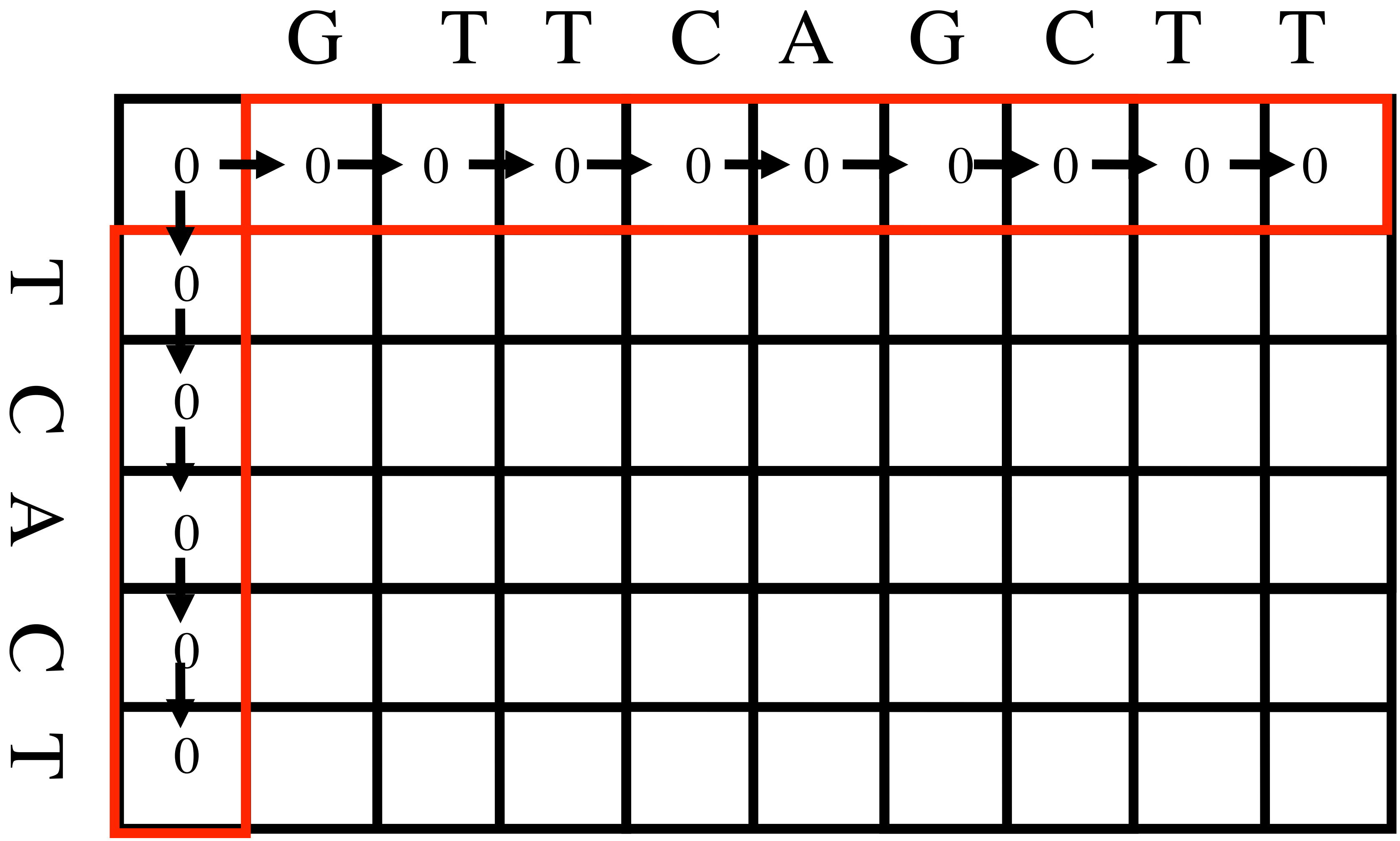
Since truncation is common in evolution,

it makes sense to NOT penalize end gaps,

or penalize them less than internal gaps.

How to NOT penalize end gaps

First: *To ignore* ***starting*** *gap penalties*, set gap rows to zero (keep the traceback arrows).



		G	T	T	C	A	G	C	T	T
	0	0	0	0	0	0	0	0	0	0
T	0									
C	0									
A	0									
C	0									
T	0									

How to NOT penalize end gaps

*Second: To ignore **ending** gap penalty, start the traceback with the MAX score at the **end of either sequence**.*

(i.e. use last row or column)

		G	T	T	C	A	G	C	T	T
	0	0	0	0	0	0	0	0	0	0
T	0									
C	0									
A	0									
C	0									
T	0									

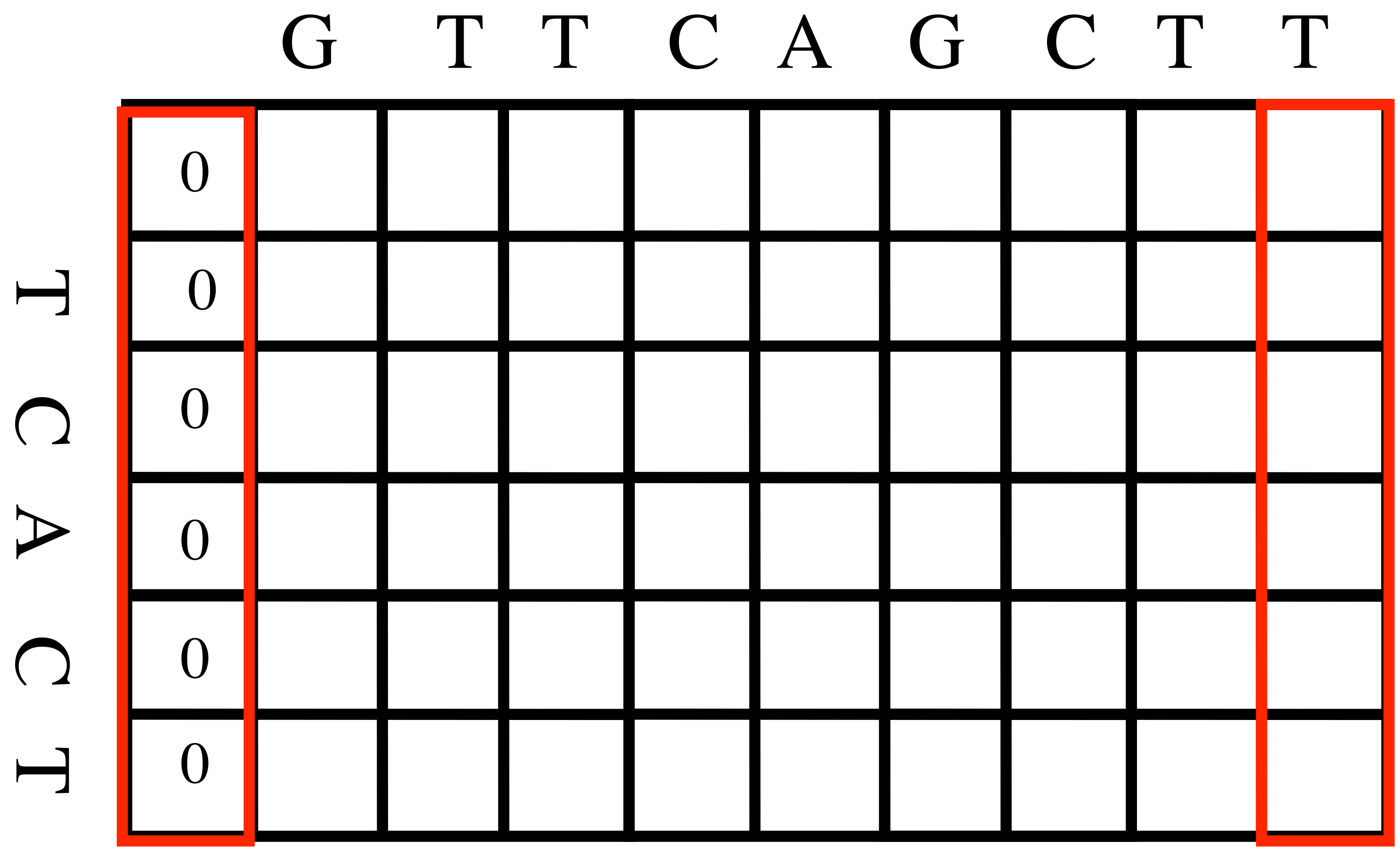
Semi-global: Penalize end gaps on one side

If we penalize end gaps in sequence 2 but not in sequence 1, we are asking for an alignment that *contains all of sequence 2 within sequence 1*.

		G	T	T	C	A	G	C	T	T
	0	0	0	0	0	0	0	0	0	0
T										
C										
A										
C										
T										

Semi-global: Penalize end gaps on one side

If we penalize end gaps in sequence 1 but not in sequence 2, we are asking for an alignment that *contains all of sequence 1 within sequence 2*.



		G	T	T	C	A	G	C	T	T
	0									
T	0									
C	0									
A	0									
C	0									
T	0									

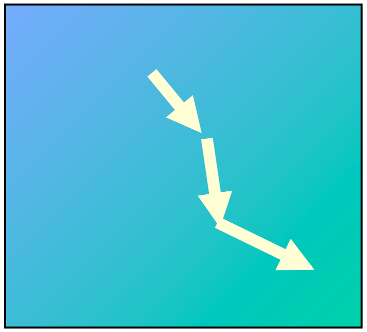
Semi-global: no end gaps

If we penalize end gaps in neither sequence, we are asking for the best alignment that contains at least two of the 4 termini.

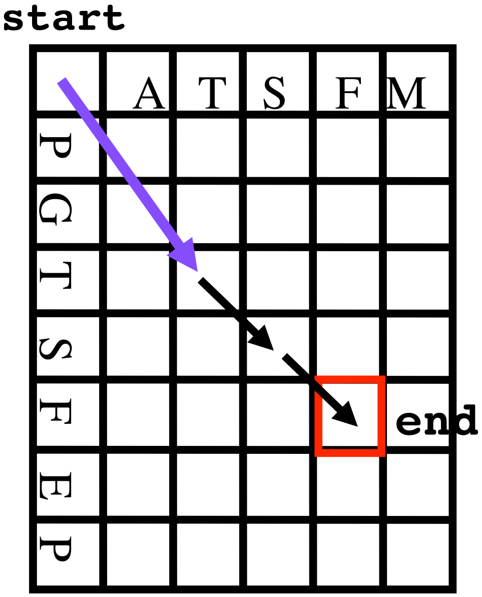
Good for identifying terminal domains in two multi-domain proteins.

		G	T	T	C	A	G	C	T	T
	0	0	0	0	0	0	0	0	0	0
T	0									
C	0									
A	0									
C	0									
T	0									

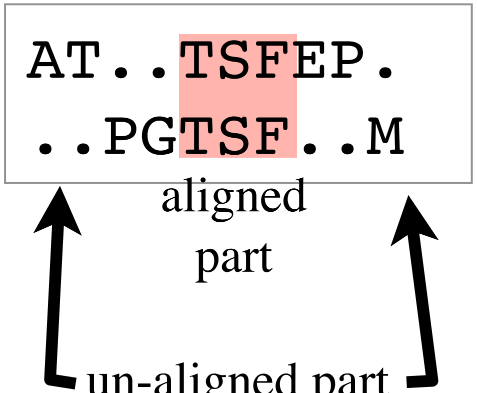
Local Alignment



A local alignment can start anywhere and end anywhere in the alignment matrix.

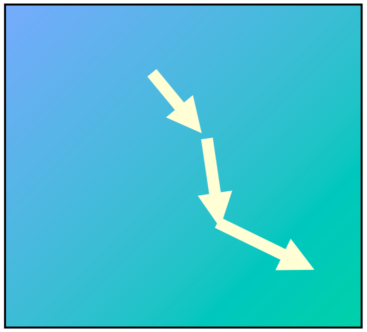


$$A(i,j) = \mathrm{MAX}\begin{cases} A(i\text{-}1,j\text{-}1) + S(i,j) & \searrow \\ A(i,j\text{-}1) + gap & \downarrow \\ A(i\text{-}1,j) + gap & \rightarrow \\ 0 + S(i,j) & \rightarrow \text{start} \end{cases}$$



```
AT..TSFEP.
..PGTSF..M
```

end
is the maximum score
anywhere in the matrix.



Local Alignment

- Asks for longest contiguous similarity between two sequences.
- Worst local alignment score is zero (0) “no alignment”
- Usually more appropriate than global or semi-global.
- Local alignment is always used for database searches.
- Local alignment scores have a theoretical distribution, used to obtain “e-values”

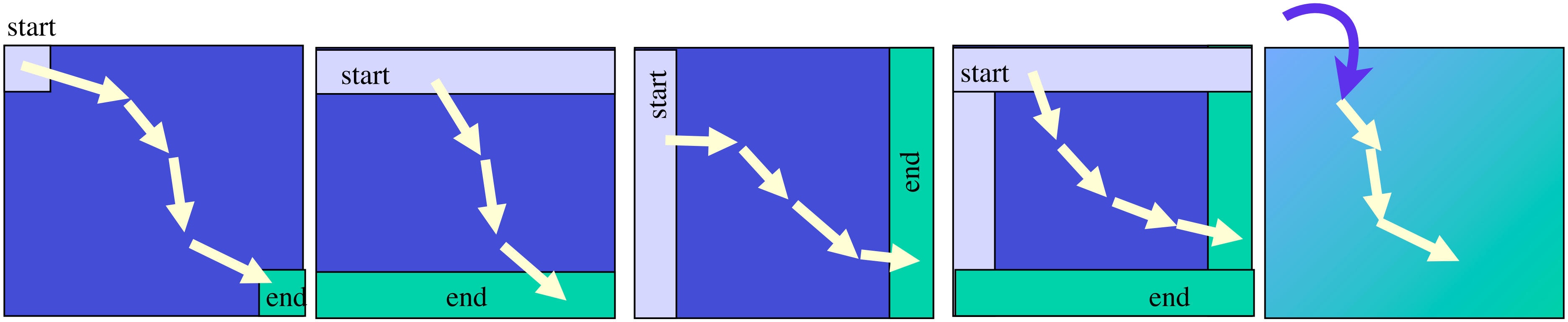
Global, semi-global, and local alignment

The choice of alignment method makes a statement about how the sequences are related. Was one sequence inserted into the other?

- **Global alignment** (end gaps) requires that all 4 termini are counted. In general, the two sequences are about the same length.

- **Semi-global** (no end gaps in 1 or both seqs) requires that one of the two sequences be completely contained in the other or that 2 or the 4 the termini be included.

- **Local alignment** finds subsequences in both. Does not require that the termini be included in the alignment.



Which alignment is intuitively better?

```
AGGCTACT~T~TCA
 GGCTACTATATCA
```

```
AGGCTACTTT~~CA
 GGCTACTATATCA
```

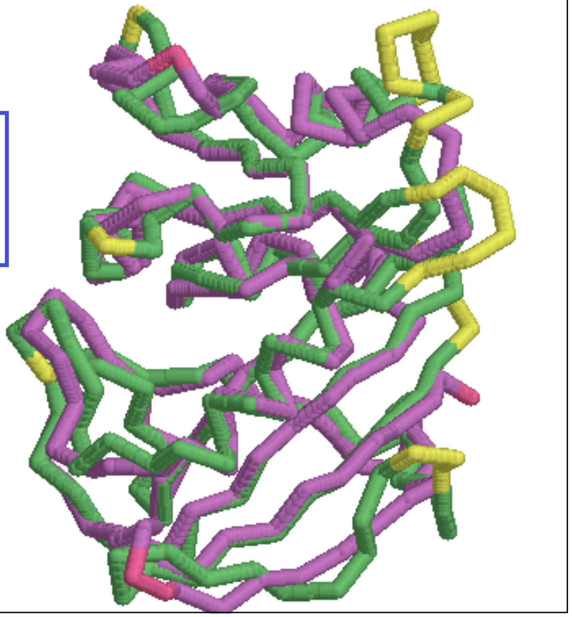
Structure-based alignments are the "gold standard"

A structure-based alignment is a sequence alignment that comes from a protein structure superposition.

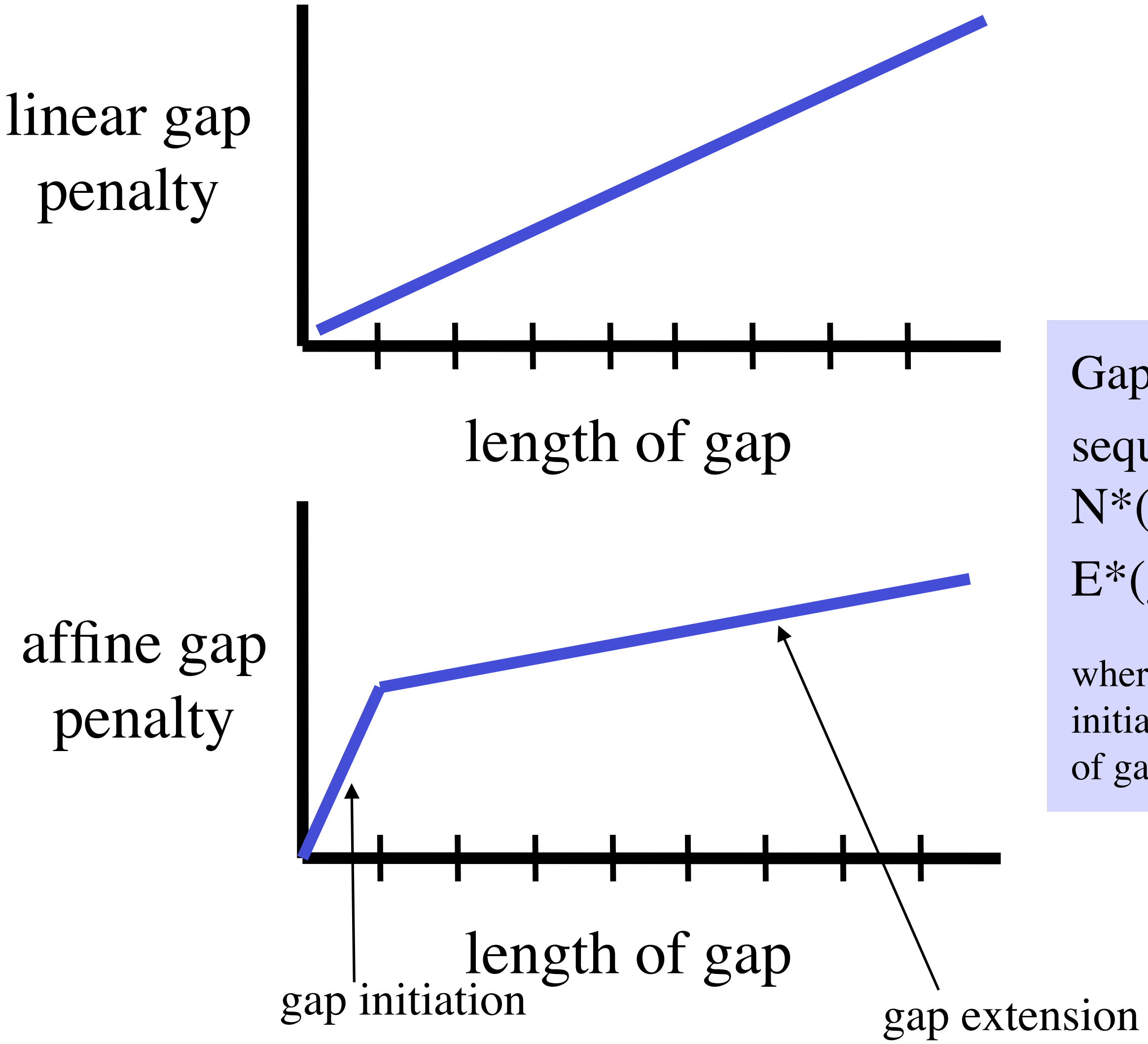
```
2DRC:A     1/2      MISLIAALAVDRVIGMENAM-PFNLPADLAWFKRNTL-------DKPVIMGRHTWESIG-
1DRF:_     3/4      SLNCIVAVSQNMGIGKNGDLPWPPLRNEFRYFQRMTTTSSVEGKQNLVIMGKKTWFSIPE

2DRC:A    52/53     --RPLPGRKNIILSSQP--GTDDRVTWVKSVDEAIAACG------DVPEIMVIGGGRVYE
1DRF:_    63/64     KNRPLKGRINLVLSRELKEPPQGAHFLSRSLDDALKLTEQPELANKVDMVWIVGGSSVYK

2DRC:A   102/103    QFLPK--AQKLYLTHIDAEVEGDTHFPDYEPDDWESVF------SEFHDADAQNSHSYCF
1DRF:_   123/124    EAMNHPGHLKLFVTRIMQDFESDTFFPEIDLEKYKLLPEYPGVLSDVQEE---KGIKYKF

2DRC:A   154/155    EILERR
1DRF:_   180/181    EVYEKN
```

What do you see? Lots of mismatches (id=38%), few gaps (8), gaps are usually *long* (1-7).

Two similar structures may be superimposed. The parts that overlay well are the matches (purple and green), and the parts that do not overlay well are the insertions (yellow and red).
Aligned positions have similar chemical 3D environment



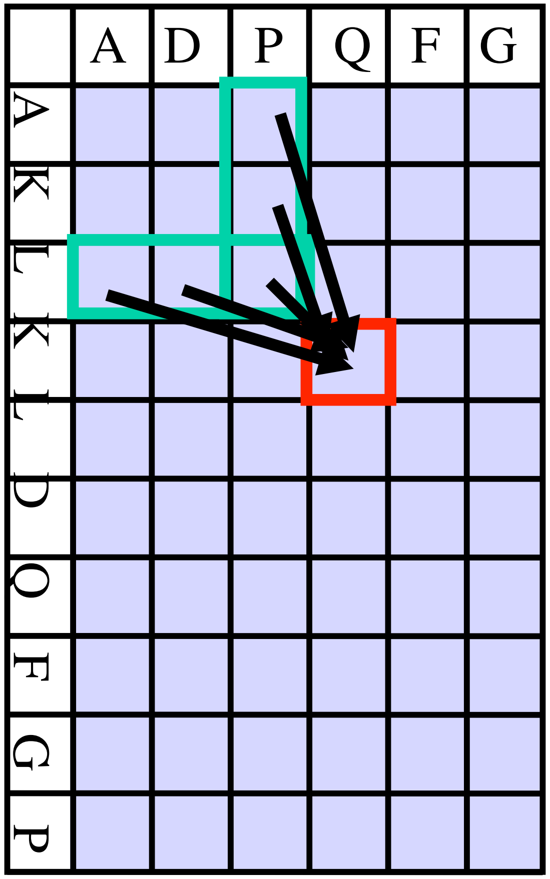
Linear versus Affine gap penalty.



Gap penalty for the whole sequence is the function. N*(gap initiation penalty) + E*(gap extension penalty)

where N is the number of gap initiation characters, E is the number of gap extension characters

Affine gap Dynamic Programming algorithm using variable length arrows



$$S_{i,j} = \max_{n} \{ S_{i-1,j-1} + s(i,j),$$
$$S_{i-1-n,j-1} + s(i,j) - g_{init} - (n-1)\, g_{ext},$$
$$S_{i-1,j-1-n} + s(i,j) - g_{init} - (n-1)\, g_{ext} \}$$

...where s(i,j) is the substiution score, *n* is the length of the gap, g_{init} is the gap initiation penalty, and g_{ext} is the gap extension penalty.

Notes: All arrows end in match. Gap-to-gap not possible. Local or semi-global only. End-gaps not scored. Arrows still translate to an alignment. Still optimal.

In class exercise: do an alignment using BLAST

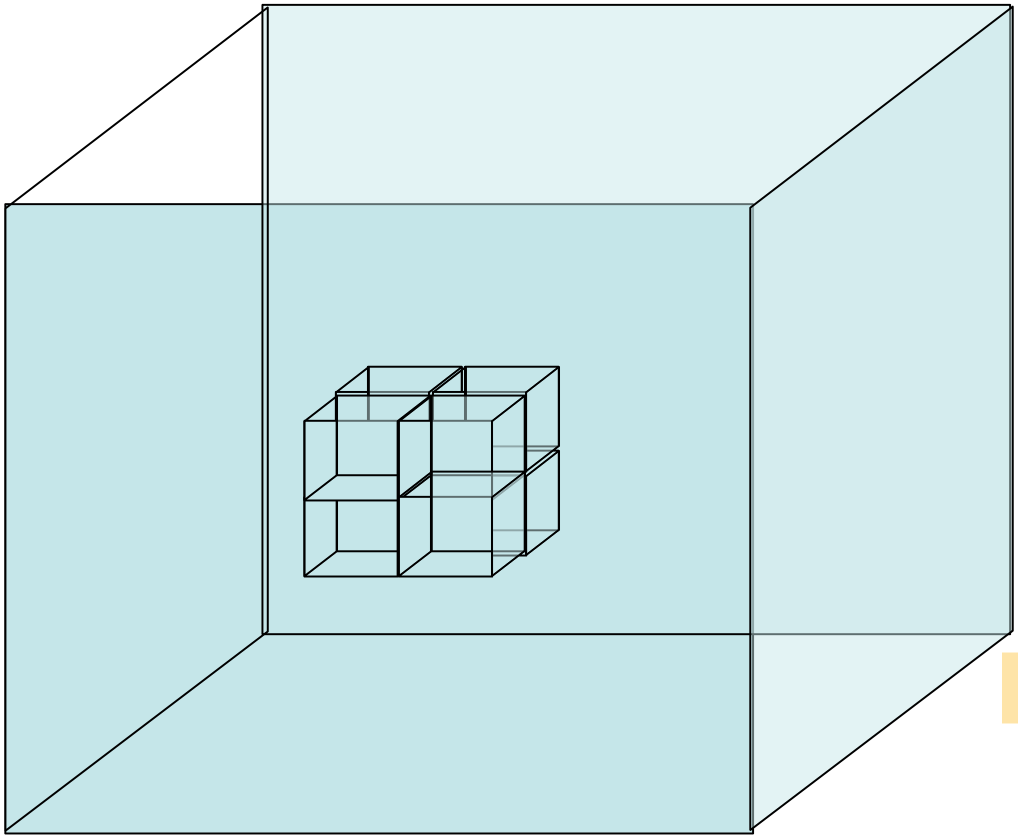
- In a browser, goto to NCBI BLAST
- Protein blast.
- Align two or more sequences.
- Query: 4DDR_A
- Subjects: 2DRC_A, CAD25017
- BLAST
- Other reports: Multiple alignment. (Cobalt)
- View format: expanded, Conservation setting: Identity
- Where is the conserved region of this enzyme?

Some things to ponder

- *How does scoring approximate the evolutionary distance*
- *How could you locate domain boundaries using Semi-global alignment*
- *How is dynamic programming different for local alignment?*
- *Is the affine gap penalty more biologically relevant than a linear gap penalty? Why?*
- *Why are structure-based alignments considered the gold standard of sequence alignment?*
- *What does it mean for a deletion to follow immediately after an insertion, evolutionarilly? Structurally?*

- Multiple sequence alignment

3D dynamic programming...



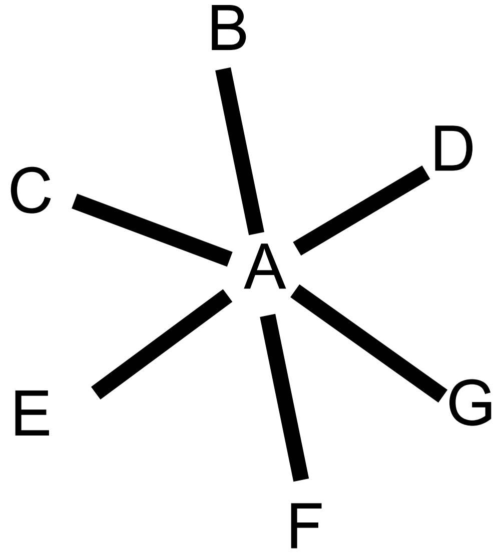
more arrows...

S(i,j,k) = MAX {
A(i-1,j-1,k-1)+S(i,j,k),
A(i-1,j,k)-gap,
A(i,j-1,k)-gap,
A(i,j,k-1)-gap,
A(i-1,j-1,k)-gap,
A(i-1,j,k-1)-gap,
A(i,j-1,k-1)-gap }

Computationally intractable....

Multiple sequence alignment -- Star method

1. Align all sequences to one sequence.
2. Stack them up.

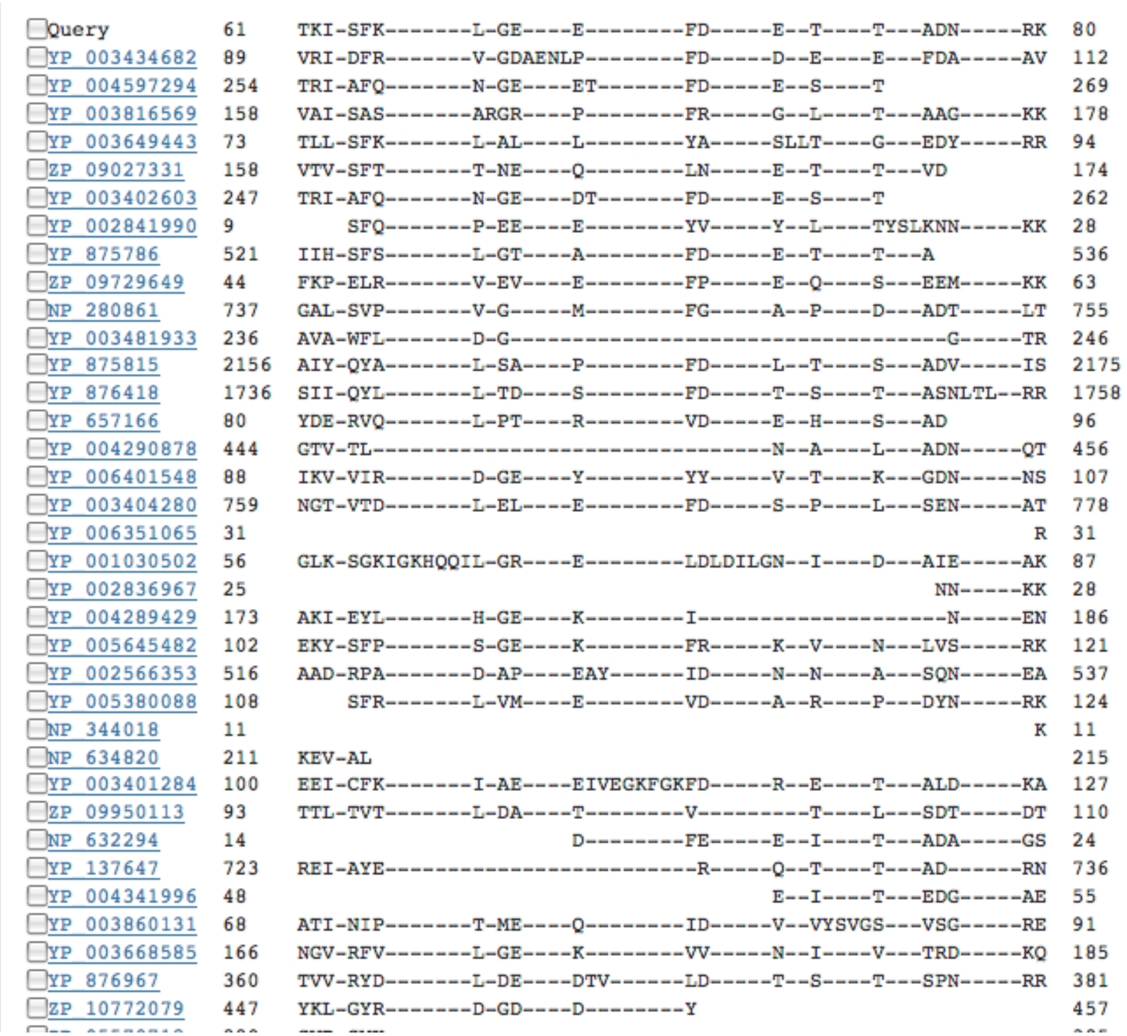


Potential **problems** with star alignment:
- Unaligned gaps.
- Ambiguous associations

```
A G H . I . W W . P F W P
A G H . I I F W . P Y . .
A G H I I . . W F P F W P
A G H . I P W W . P . . .
```

Each pairwise alignment by itself looks fine, but when you stack them up, you see disagreements.

BLAST “query-anchored” alignments are star alignments



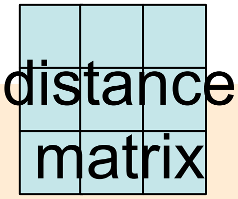
```
☐Query         61    TKI-SFK-------L-GE----E--------FD-----E--T----T---ADN-----RK  80
☐YP_003434682  89    VRI-DFR-------V-GDAENLP--------FD-----D--E----E---FDA-----AV  112
☐YP_004597294  254   TRI-AFQ-------N-GE----ET-------FD-----E--S----T               269
☐YP_003816569  158   VAI-SAS-------ARGR----P--------FR-----G--L----T---AAG-----KK  178
☐YP_003649443  73    TLL-SFK-------L-AL----L--------YA-----SLLT----G---EDY-----RR  94
☐ZP_09027331   158   VTV-SFT-------T-NE----Q--------LN-----E--T----T---VD          174
☐YP_003402603  247   TRI-AFQ-------N-GE----DT-------FD-----E--S----T               262
☐YP_002841990  9         SFQ-------P-EE----E--------YV-----Y--L----TYSLKNN-----KK  28
☐YP_875786     521   IIH-SFS-------L-GT----A--------FD-----E--T----T---A           536
☐ZP_09729649   44    FKP-ELR-------V-EV----E--------FP-----E--Q----S---EEM-----KK  63
☐NP_280861     737   GAL-SVP-------V-G-----M--------FG-----A--P----D---ADT-----LT  755
☐YP_003481933  236   AVA-WFL-------D-G-----------------------------------G-----TR  246
☐YP_875815     2156  AIY-QYA-------L-SA----P--------FD-----L--T----S---ADV-----IS  2175
☐YP_876418     1736  SII-QYL-------L-TD----S--------FD-----T--S----T---ASNLTL--RR  1758
☐YP_657166     80    YDE-RVQ-------L-PT----R--------VD-----E--H----S---AD          96
☐YP_004290878  444   GTV-TL--------------------------------N--A----L---ADN-----QT  456
☐YP_006401548  88    IKV-VIR-------D-GE----Y--------YY-----V--T----K---GDN-----NS  107
☐YP_003404280  759   NGT-VTD-------L-EL----E--------FD-----S--P----L---SEN-----AT  778
☐YP_006351065  31                                                               R  31
☐YP_001030502  56    GLK-SGKIGKHQQIL-GR----E--------LDLDILGN--I----D---AIE-----AK  87
☐YP_002836967  25                                                       NN-----KK  28
☐YP_004289429  173   AKI-EYL-------H-GE----K--------I--------------------N-----EN  186
☐YP_005645482  102   EKY-SFP-------S-GE----K--------FR-----K--V----N---LVS-----RK  121
☐YP_002566353  516   AAD-RPA-------D-AP----EAY------ID-----N--N----A---SQN-----EA  537
☐YP_005380088  108       SFR-------L-VM----E--------VD-----A--R----P---DYN-----RK  124
☐NP_344018     11                                                               K  11
☐NP_634820     211   KEV-AL                                                        215
☐YP_003401284  100   EEI-CFK-------I-AE----EIVEGKFGKFD-----R--E----T---ALD-----KA  127
☐ZP_09950113   93    TTL-TVT-------L-DA----T--------V---------T----L---SDT-----DT  110
☐NP_632294     14                          D--------FE-----E--I----T---ADA-----GS  24
☐YP_137647     723   REI-AYE-------------------------R-----Q--T----T---AD------RN  736
☐YP_004341996  48                                          E--I----T---EDG-----AE  55
☐YP_003860131  68    ATI-NIP-------T-ME----Q--------ID-----V--VYSVGS---VSG-----RE  91
☐YP_003668585  166   NGV-RFV-------L-GE----K--------VV-----N--I----V---TRD-----KQ  185
☐YP_876967     360   TVV-RYD-------L-DE----DTV------LD-----T--S----T---SPN-----RR  381
☐ZP_10772079   447   YKL-GYR-------D-GD----D--------Y                              457
```

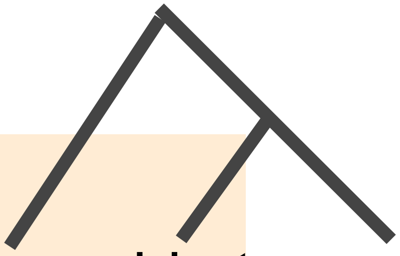
How can you tell? Very gappy.

Multiple sequence alignment -- Progressive method

Method for progressive alignment

1. Align all pairs. Save scores in a distance matrix
2. Pairwise align *two most similar.*
3. Align the next two most similar sequence. Etc.
4. Add sequences until all sequences are aligned





guide tree

Current alignment

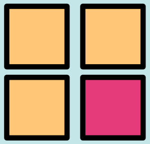
gap

```
A G H I . W W P F
A G H I I F W P Y
```

sequence to add

```
A
W
P
Y
```

DP alignment matrix



S(P,[W,F]) =(1/2)(S(P,W) + S(P,F))

In class: progressive alignment

Making a guide tree

Identities

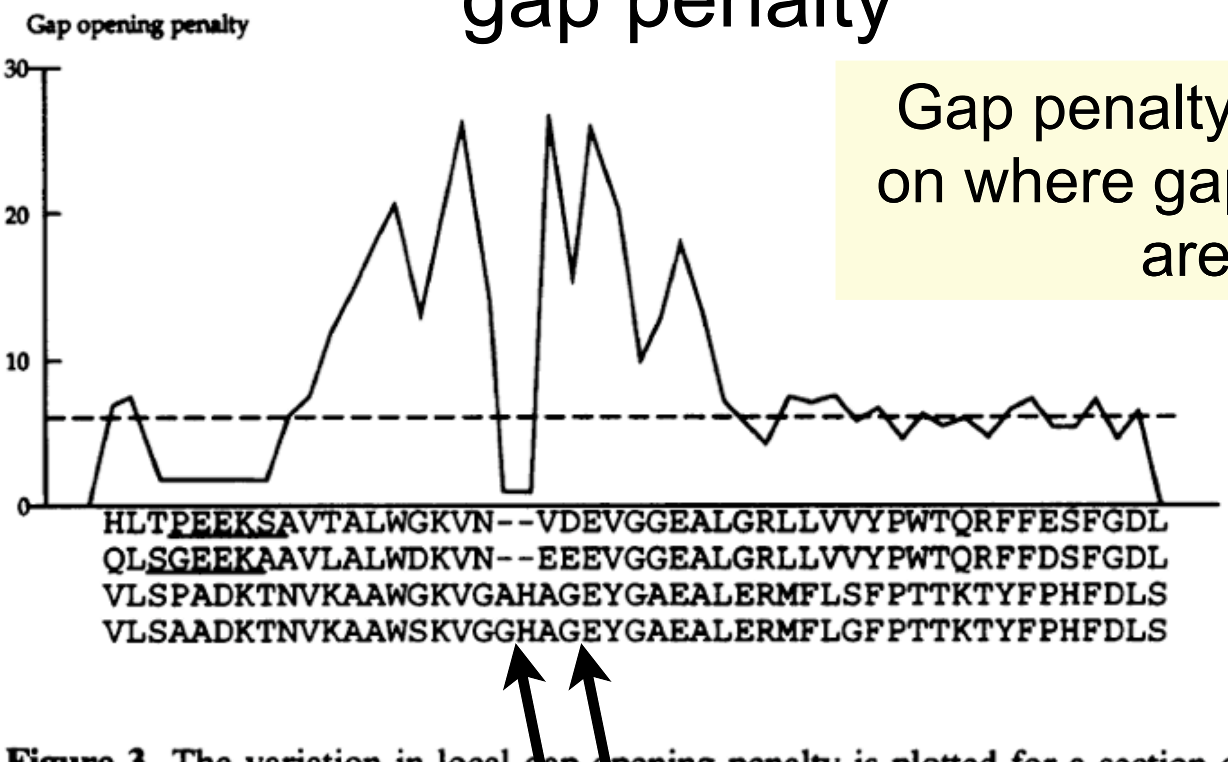
	A	B	C	D	E	F
A		97	81	82	59	32
B			77	80	55	31
C				90	65	40
D					61	42
E						33
F						

Neighbor-joining algorithm:

A B C D E F

Draw guide tree here

CLUSTALW Progressive multiple sequence alignment with position specific gap penalty



Gap penalty depends on where gaps already are.

no penalty if there is already a gap there. Big penalty to put a gap near an existing one.

Figure 3. The variation in local gap opening penalty is plotted for a section of alignment. The inital gap ne. Two hydrophilic stretches are u the ends of the alignment, the hyd aps. The highest values are within 8 residues of the two gap positions. The rest of the variation is caused by the residue specific gap penalties (12).

Substitution matrices

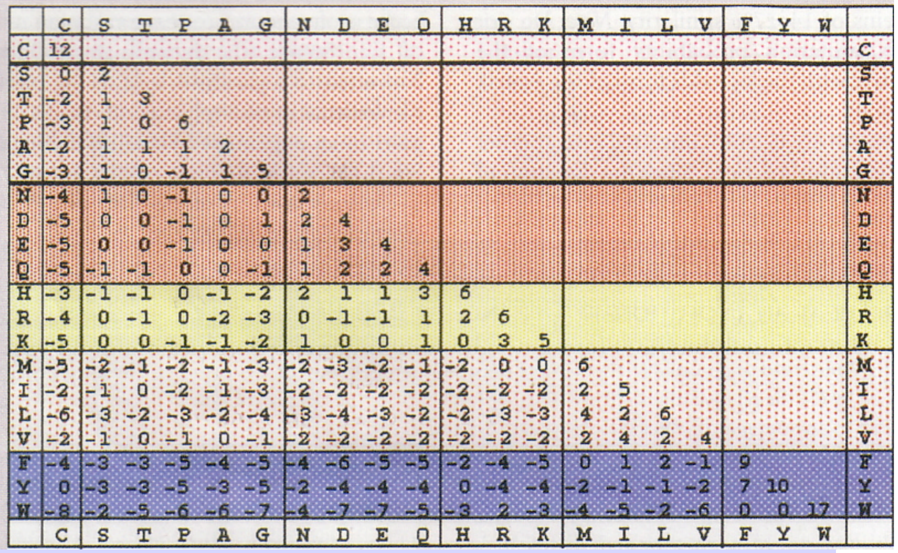
- Used to score aligned positions, usually of amino acids.
- Expressed as the *log-likelihood ratio of mutation* (or *log-odds ratio*)
- Derived from multiple sequence alignments

Most commonly used: PAM and BLOSUM

- PAM = **p**ercent **a**ccepted **m**utations (Dayhoff)
- BLOSUM = **Blo**cks **su**bstitution **m**atrix (Henikoff)

PAM

M Dayhoff, 1978



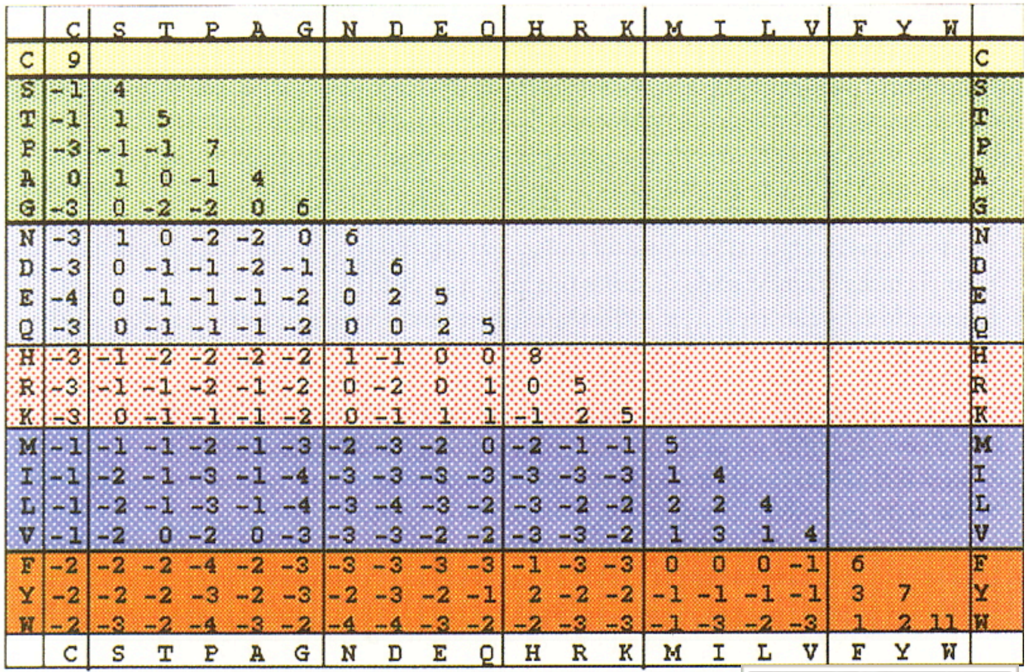
	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W	
C	12																				C
S	0	2																			S
T	-2	1	3																		T
P	-3	1	0	6																	P
A	-2	1	1	1	2																A
G	-3	1	0	-1	1	5															G
N	-4	1	0	-1	0	0	2														N
D	-5	0	0	-1	0	1	2	4													D
E	-5	0	0	-1	0	0	1	3	4												E
Q	-5	-1	-1	0	0	-1	1	2	2	4											Q
H	-3	-1	-1	0	-1	-2	2	1	1	3	6										H
R	-4	0	-1	0	-2	-3	0	-1	-1	1	2	6									R
K	-5	0	0	-1	-1	-2	1	0	0	1	0	3	5								K
M	-5	-2	-1	-2	-1	-3	-2	-3	-2	-1	-2	0	0	6							M
I	-2	-1	0	-2	-1	-3	-2	-2	-2	-2	-2	-2	-2	2	5						I
L	-6	-3	-2	-3	-2	-4	-3	-4	-3	-2	-2	-3	-3	4	2	6					L
V	-2	-1	0	-1	0	-1	-2	-2	-2	-2	-2	-2	-2	2	4	2	4				V
F	-4	-3	-3	-5	-4	-5	-4	-6	-5	-5	-2	-4	-5	0	1	2	-1	9			F
Y	0	-3	-3	-5	-3	-5	-2	-4	-4	-4	0	-4	-4	-2	-1	-1	-2	7	10		Y
W	-8	-2	-5	-6	-6	-7	-4	-7	-7	-5	-3	2	-3	-4	-5	-2	-6	0	0	17	W
	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W	

- Stands for Percent Accepted Mutations
- The PAM1 matrix is made from alignments with 1% changes (99% identities).
- To get the relative frequency of each type of mutation, we count the times it was observed in a database over a large set of sequence alignments.
- Based on global alignments

Margaret Oakley Dayhoff

BLOSUM

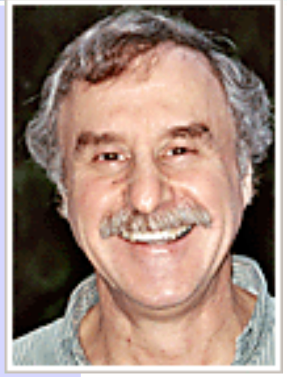
Henikoff & Henikoff, 1992



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

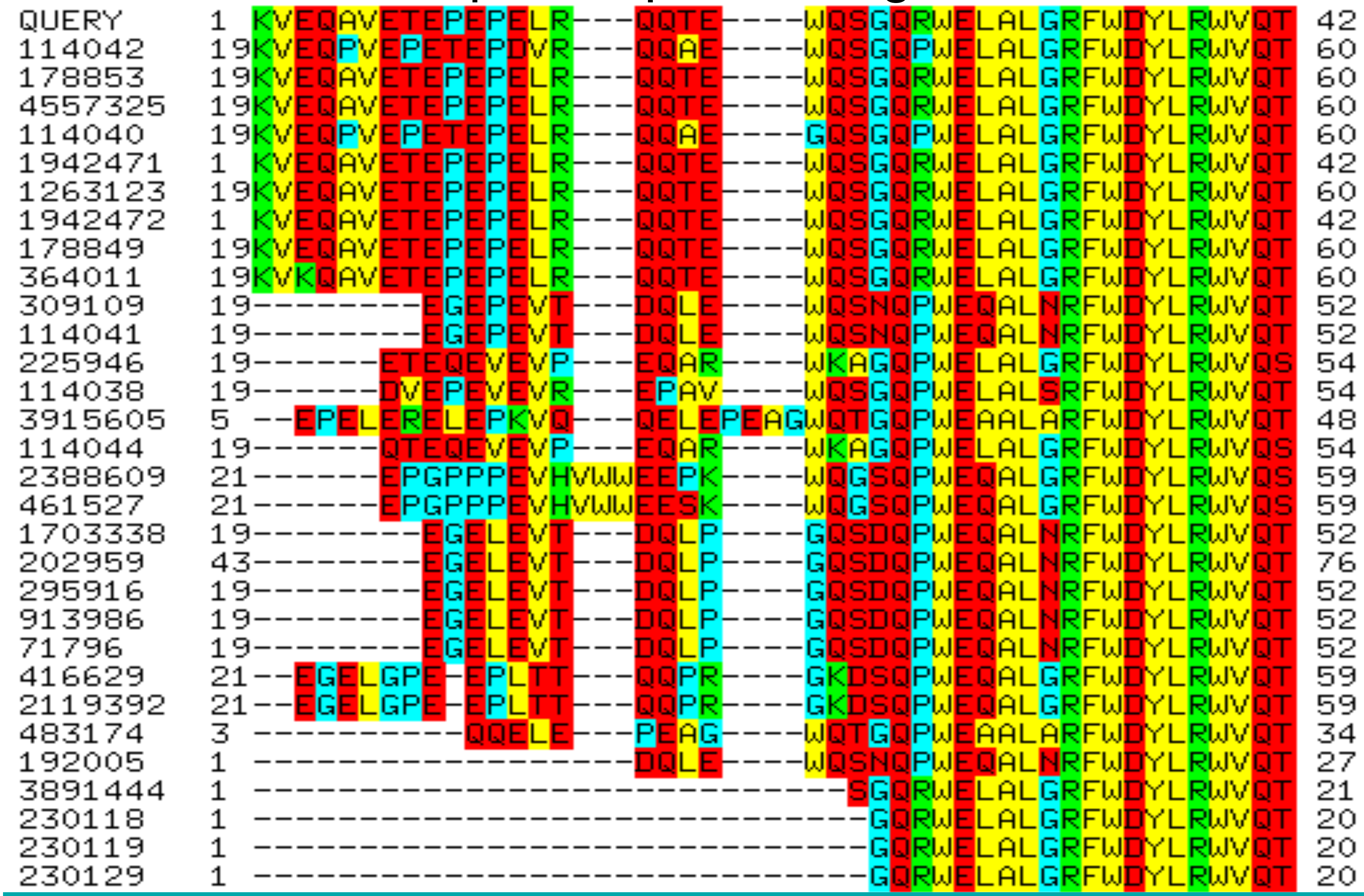
- Based on database of ungapped local alignments (BLOCKS database)

- BLOSUM number indicates the percent identity level of sequences in the alignment. For example, for BLOSUM62 sequences with approximately 62% identity were counted.



Steven Henikoff

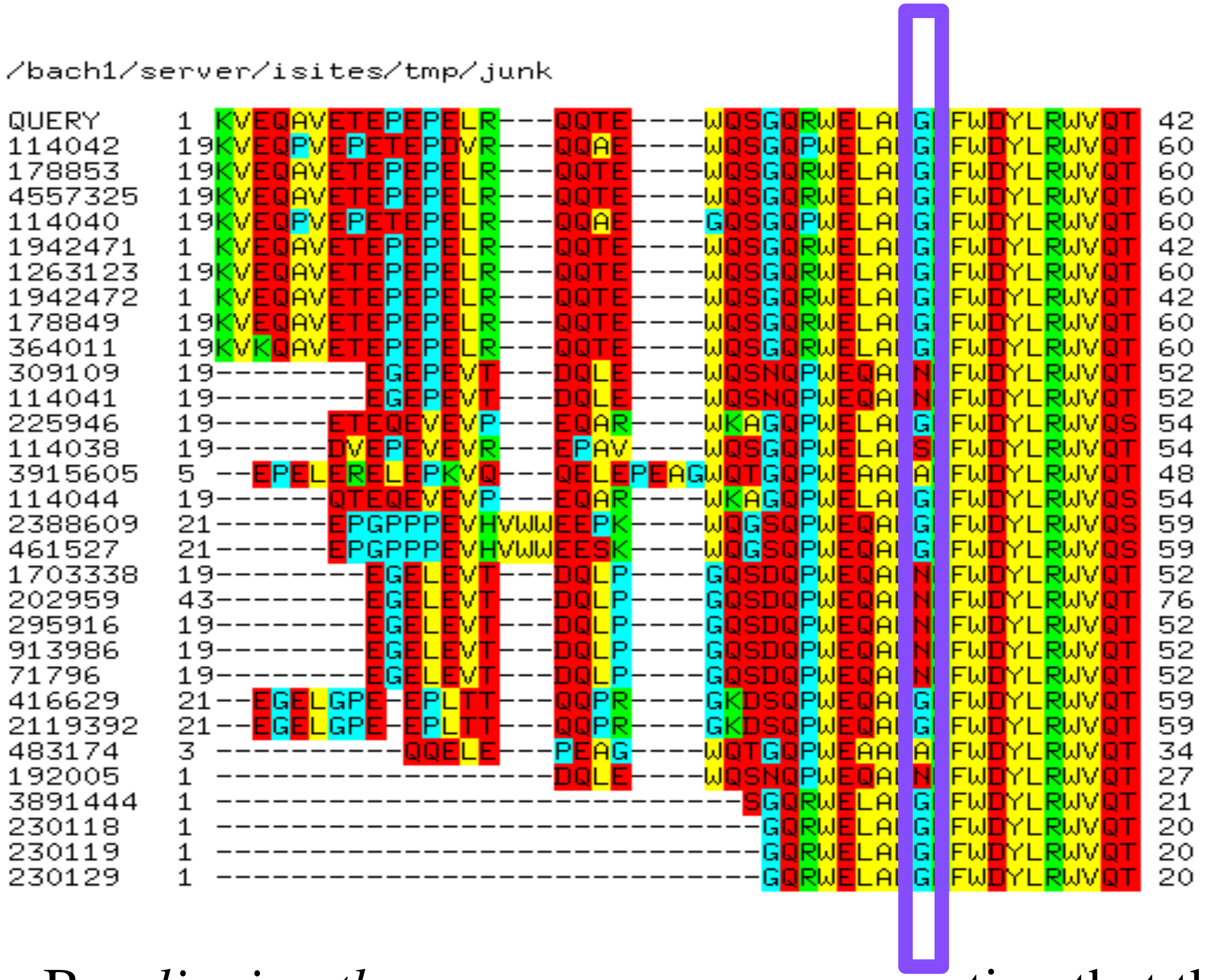
Multiple Sequence Alignment



```
QUERY    1 KVEQAVETEPEPELR---QQTE----WQSGQRWELALGRFWDYLRWVQT 42
114042   19KVEQPVEPETEPDVR---QQAE----WQSGQPWELALGRFWDYLRWVQT 60
178853   19KVEQAVETEPEPELR---QQTE----WQSGQRWELALGRFWDYLRWVQT 60
4557325  19KVEQAVETEPEPELR---QQTE----WQSGQRWELALGRFWDYLRWVQT 60
114040   19KVEQPVEPETEPELR---QQAE----GQSGQPWELALGRFWDYLRWVQT 60
1942471  1 KVEQAVETEPEPELR---QQTE----WQSGQRWELALGRFWDYLRWVQT 42
1263123  19KVEQAVETEPEPELR---QQTE----WQSGQRWELALGRFWDYLRWVQT 60
1942472  1 KVEQAVETEPEPELR---QQTE----WQSGQRWELALGRFWDYLRWVQT 42
178849   19KVEQAVETEPEPELR---QQTE----WQSGQRWELALGRFWDYLRWVQT 60
364011   19KVKQAVETEPEPELR---QQTE----WQSGQRWELALGRFWDYLRWVQT 60
309109   19--------EGEPEVT---DQLE----WQSNQPWEQALNRFWDYLRWVQT 52
114041   19--------EGEPEVT---DQLE----WQSNQPWEQALNRFWDYLRWVQT 52
225946   19------ETEQEVEVP---EQAR----WKAGQPWELALGRFWDYLRWVQS 54
114038   19------DVEPEVEVR---EPAV----WQSGQPWELALSRFWDYLRWVQT 54
3915605  5 --EPELERELEPKVQ---QELEPEAGWQTGQPWEAALARFWDYLRWVQT 48
114044   19------QTEQEVEVP---EQAR----WKAGQPWELALGRFWDYLRWVQS 54
2388609  21------EPGPPPEVHVWWEEPK----WQGSQPWEQALGRFWDYLRWVQS 59
461527   21------EPGPPPEVHVWWEESK----WQGSQPWEQALGRFWDYLRWVQS 59
1703338  19--------EGELEVT---DQLP----GQSDQPWEQALNRFWDYLRWVQT 52
202959   43--------EGELEVT---DQLP----GQSDQPWEQALNRFWDYLRWVQT 76
295916   19--------EGELEVT---DQLP----GQSDQPWEQALNRFWDYLRWVQT 52
913986   19--------EGELEVT---DQLP----GQSDQPWEQALNRFWDYLRWVQT 52
71796    19--------EGELEVT---DQLP----GQSDQPWEQALNRFWDYLRWVQT 52
416629   21--EGELGPE-EPLTT---QQPR----GKDSQPWEQALGRFWDYLRWVQT 59
2119392  21--EGELGPE-EPLTT---QQPR----GKDSQPWEQALGRFWDYLRWVQT 59
483174   3 ----------QQELE---PEAG----WQTGQPWEAALARFWDYLRWVQT 34
192005   1 ------------------DQLE----WQSNQPWEQALNRFWDYLRWVQT 27
3891444  1 ----------------------------SGQRWELALGRFWDYLRWVQT 21
230118   1 -----------------------------GQRWELALGRFWDYLRWVQT 20
230119   1 -----------------------------GQRWELALGRFWDYLRWVQT 20
230129   1 -----------------------------GQRWELALGRFWDYLRWVQT 20
```

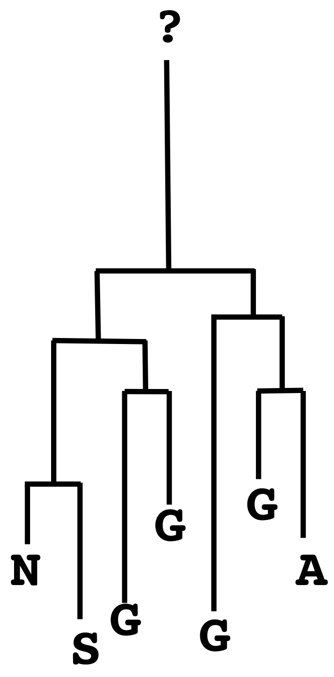
A multiple sequence alignment is made using many pairwise sequence alignments

Columns in a MSA have a common evolutionary history



```
/bach1/server/isites/tmp/junk

QUERY      1 KVEQAVETEPEPELR---QQTE----WQSGQRWELA G FWDYLRWVQT 42
114042     19KVEQPVEPETEPDVR---QQAE----WQSGQPWELA G FWDYLRWVQT 60
178853     19KVEQAVETEPEPELR---QQTE----WQSGQRWELA G FWDYLRWVQT 60
4557325    19KVEQAVETEPEPELR---QQTE----WQSGQRWELA G FWDYLRWVQT 60
114040     19KVEQPVEPETEPELR---QQAE----GQSGQPWELA G FWDYLRWVQT 60
1942471    1 KVEQAVETEPEPELR---QQTE----WQSGQRWELA G FWDYLRWVQT 42
1263123    19KVEQAVETEPEPELR---QQTE----WQSGQRWELA G FWDYLRWVQT 60
1942472    1 KVEQAVETEPEPELR---QQTE----WQSGQRWELA G FWDYLRWVQT 42
178849     19KVEQAVETEPEPELR---QQTE----WQSGQRWELA G FWDYLRWVQT 60
364011     19KVKQAVETEPEPELR---QQTE----WQSGQRWELA G FWDYLRWVQT 60
309109     19--------EGEPEVT---DQLE----WQSNQPWEQA N FWDYLRWVQT 52
114041     19--------EGEPEVT---DQLE----WQSNQPWEQA N FWDYLRWVQT 52
225946     19------ETEQEVEVP---EQAR----WKAGQPWELA G FWDYLRWVQS 54
114038     19------DVEPEVEVR---EPAV----WQSGQPWELA S FWDYLRWVQT 54
3915605    5 --EPELERELEPKVQ---QELEPEAGWQTGQPWEAA A FWDYLRWVQT 48
114044     19------QTEQEVEVP---EQAR----WKAGQPWELA G FWDYLRWVQS 54
2388609    21------EPGPPPEVHVWWEEPK----WQGSQPWEQA G FWDYLRWVQS 59
461527     21------EPGPPPEVHVWWEESK----WQGSQPWEQA G FWDYLRWVQS 59
1703338    19--------EGELEVT---DQLP----GQSDQPWEQA N FWDYLRWVQT 52
202959     43--------EGELEVT---DQLP----GQSDQPWEQA N FWDYLRWVQT 76
295916     19--------EGELEVT---DQLP----GQSDQPWEQA N FWDYLRWVQT 52
913986     19--------EGELEVT---DQLP----GQSDQPWEQA N FWDYLRWVQT 52
71796      19--------EGELEVT---DQLP----GQSDQPWEQA N FWDYLRWVQT 52
416629     21--EGELGPE-EPLTT---QQPR----GKDSQPWEQA G FWDYLRWVQT 59
2119392    21--EGELGPE-EPLTT---QQPR----GKDSQPWEQA G FWDYLRWVQT 59
483174     3 ----------QQELE---PEAG----WQTGQPWEAA A FWDYLRWVQT 34
192005     1 ------------------DQLE----WQSNQPWEQA N FWDYLRWVQT 27
3891444    1 ----------------------------SGQRWELA G FWDYLRWVQT 21
230118     1 -----------------------------GQRWELA G FWDYLRWVQT 20
230119     1 -----------------------------GQRWELA G FWDYLRWVQT 20
230129     1 -----------------------------GQRWELA G FWDYLRWVQT 20
```

a phylogenetic tree for one position in the alignment

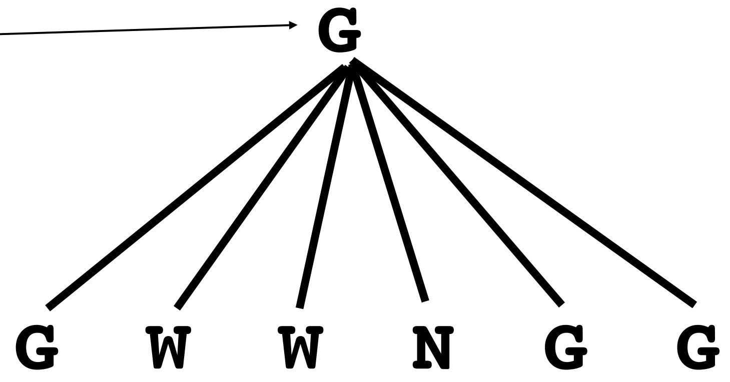


By *aligning the sequences*, we are asserting that the aligned residues in each column had a common ancestor.

Counting mutations without knowing ancestral sequences

Naíve way: Assume *any* of the characters could be the ancestral one. Assume equal distance to the ancestor from each taxon.

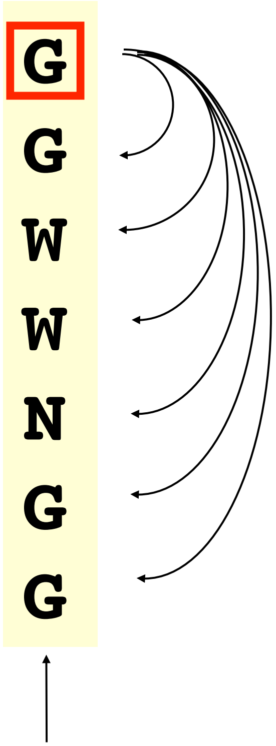
```
L K F G R L S K K P
L K F G R L S K K P
L K F W R L T K K P
L K F W R L S K K P
L K F N R L S R K P
L K F G R L T R K P
L K F G R L ~ K K P
```



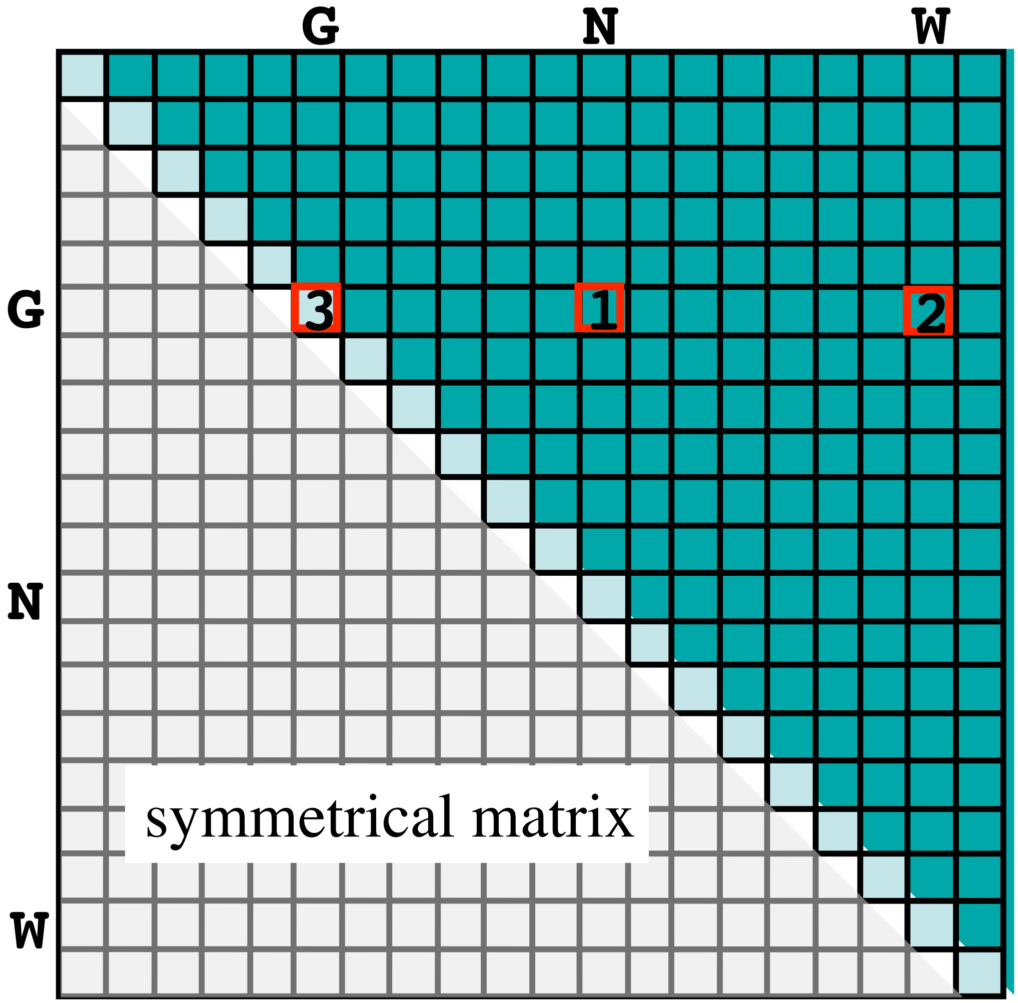
If G was the ancestor, then it mutated to a W twice, to N once, and stayed G three times.

Summing the substitution counts

We assume the ancestor is one of the observed amino acids, but we don't know which, so we try them all.



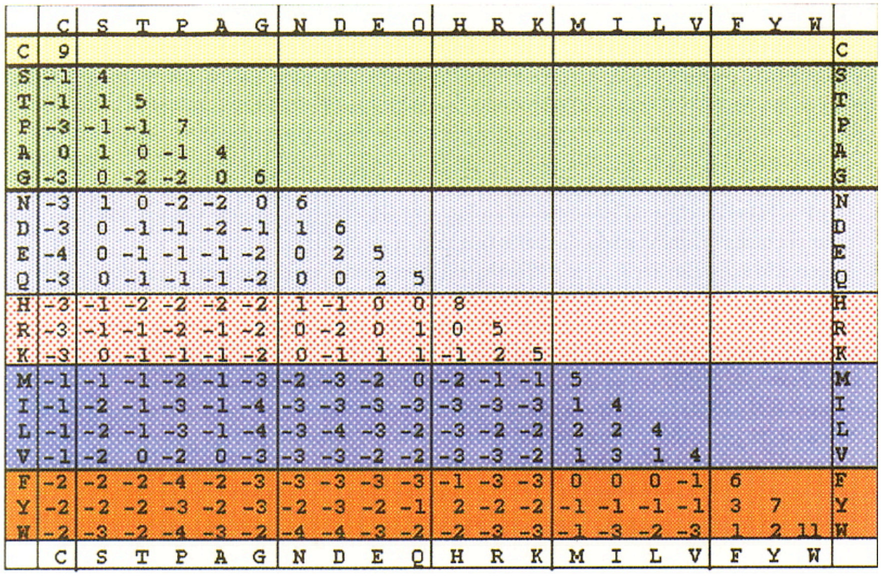
one column of a MSA



symmetrical matrix

Substitution scores are expressed as log odds ratios

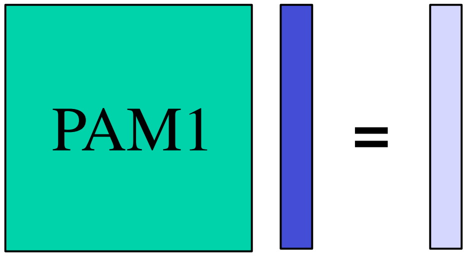
$$\text{log odds ratio} = \log_2(\text{observed/expected})$$

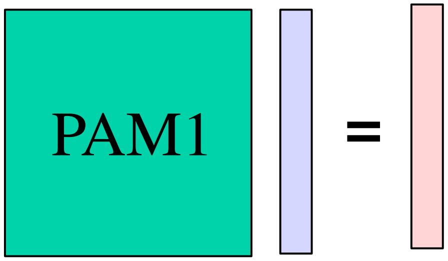


	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W	
C	9																				C
S	-1	4																			S
T	-1	1	5																		T
P	-3	-1	-1	7																	P
A	0	1	0	-1	4																A
G	-3	0	-2	-2	0	6															G
N	-3	1	0	-2	-2	0	6														N
D	-3	0	-1	-1	-2	-1	1	6													D
E	-4	0	-1	-1	-1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

PAM assumes Markovian evolution

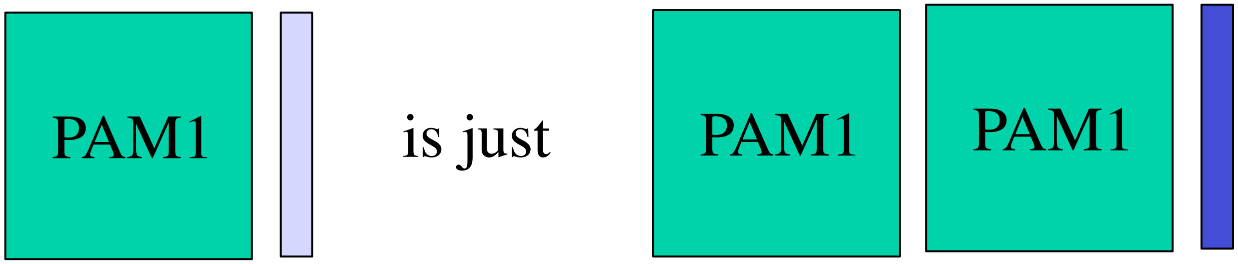
Start with one sequence. One position. Say Gly. **Wait 1 million years**. What amino acids are now found at that position?



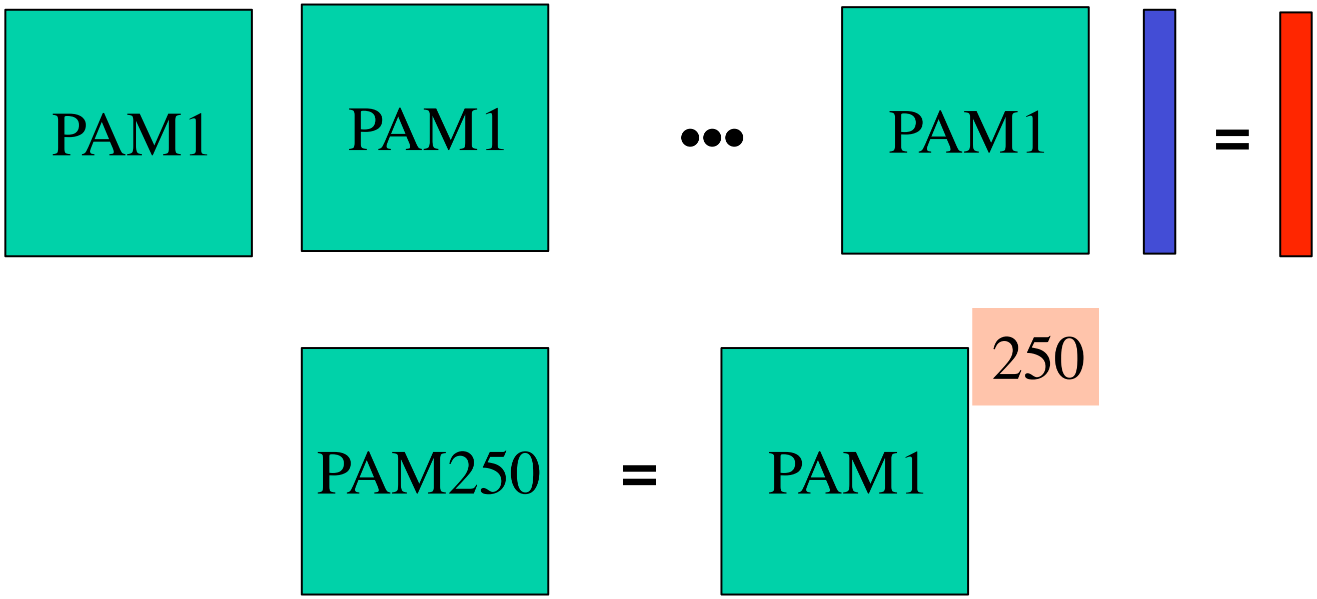


Wait another million years.

But,...



250 million years?



PAM1 PAM1 ••• PAM1 =

PAM250 = PAM1 250

The number after PAM denotes the power to which PAM1 was taken.

- NOTE OF CLARIFICATION:
- PAM does **not** stand for Plus A Million years (or anything like that). It stands for Percent Accepted Mutations.
- One PAM1 unit does **not** correspond to 1 million years of evolution. There is no timescale associated with PAM.
- PAM1 corresponds to 1% mutations. (or 99% identity). The timescale depends on the species.

Protein versus DNA alignments

Are protein alignment better?

- Protein alphabet = 20, DNA alphabet = 4.
  - Protein alignment is more informative
  - Less chance of homoplasy with proteins.
  - Homology detectable at greater edit distance
  - Protein alignment more informative
- Better Gold Standard alignments are available for proteins.
  - Better statistics from G.S. alignments.
- On the other hand, DNA alignments are more sensitive to short evolutionary distances.