Bioinformatics 1: lecture 4

Molecular evolution

Global, semi-global and local

Affine gap penalty

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 scored

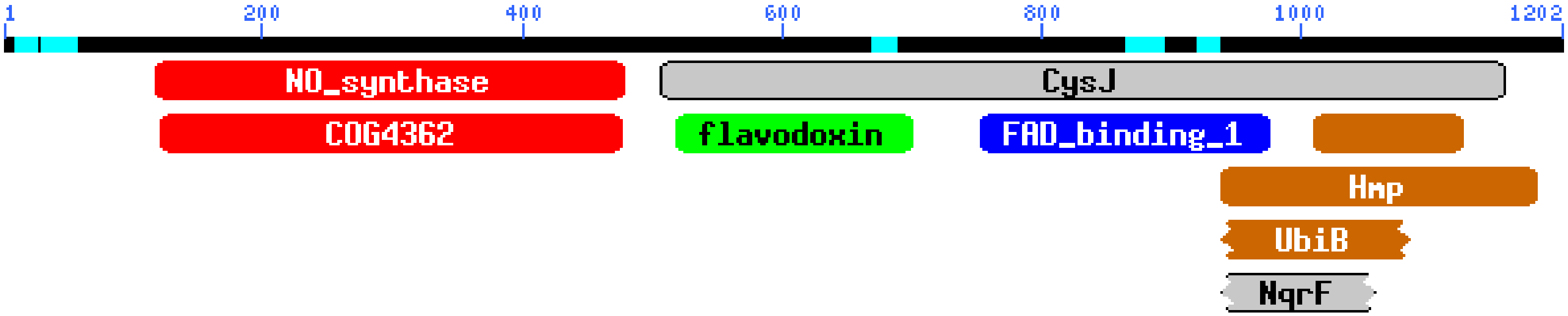
- point mutations .... substitution matrix
- deletion .... gap penalty
- insertion .... gap penalty
- truncation .... end gap penalty
- extension .... end gap penalty

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

Extension/truncation, domains : end gaps

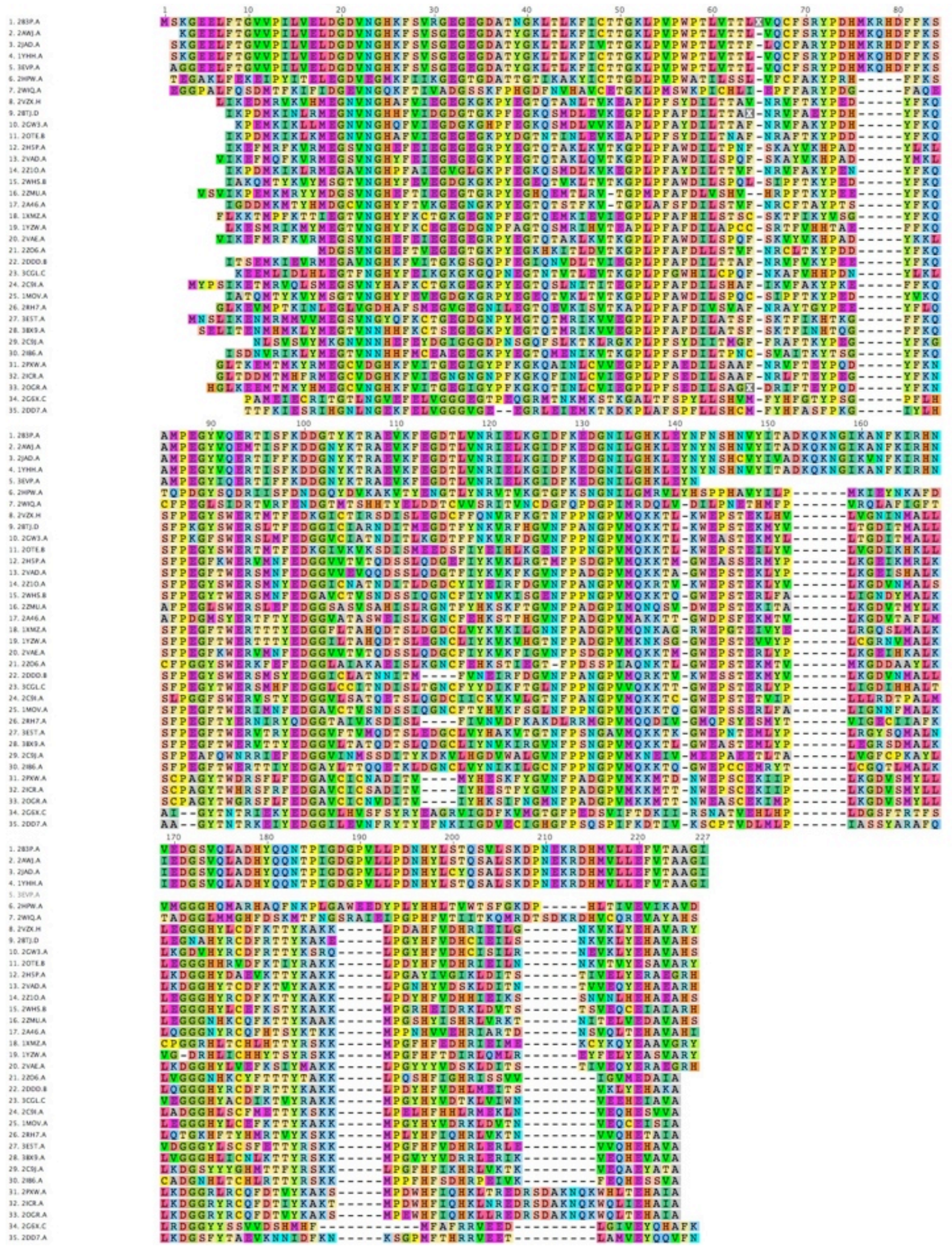


Example: here is an alignment of mouse nitric oxide synthase (thick black line). It has multiple domains which are homologous to several shorter proteins. If we penalize end gaps, what happens to the score of the true alignment? Did "end gaps" evolve the same way as internal gaps? (no!)

Unless the two proteins are known to be single domains,
it makes more sense NOT to penalize end gaps.

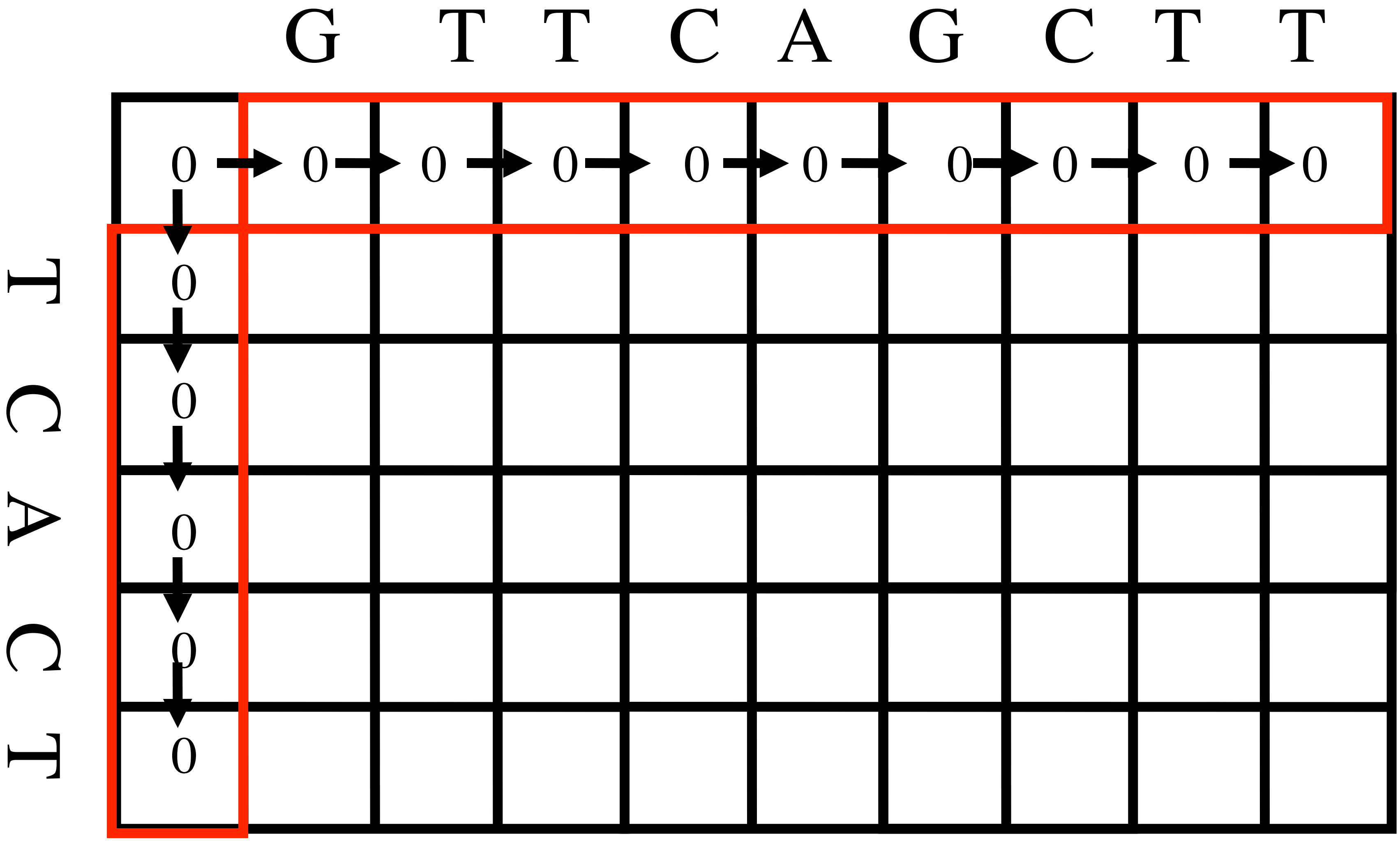
A structure-based multiple sequence alignment of fluorescent proteins.

Note the ragged N-terminal edge, small number of indels, large number of point substitutions distributed unevenly over the sequence (conserved regions versus hot spots)



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									

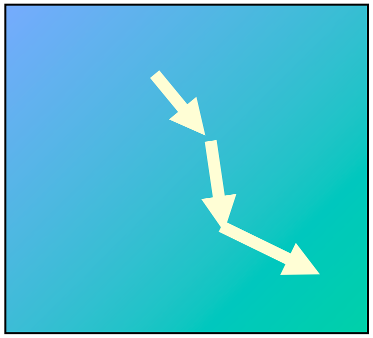
Semi-global: no end gaps

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

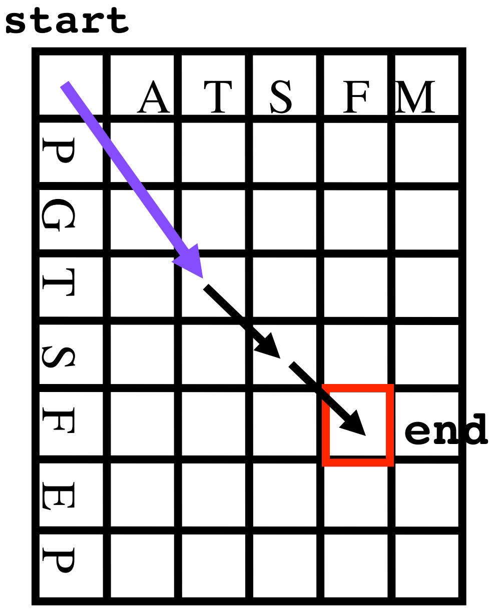
Good for identifying terminal domains in two multi-domain proteins.

		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.

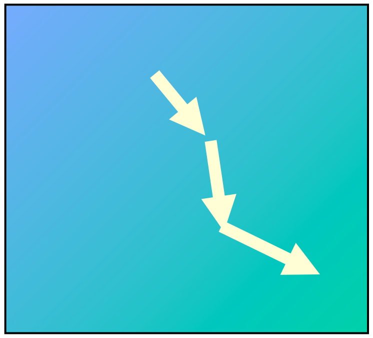


$$A(i,j) = \text{MAX}\begin{cases} A(i\text{-}1,j\text{-}1) + S(i,j) & \searrow \\ A(i,j\text{-}1) + \text{gap} & \downarrow \\ A(i\text{-}1,j) + \text{gap} & \rightarrow \\ 0 + S(i,j) & \rightarrow \text{start} \end{cases}$$

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

end is the maximum score *anywhere in the matrix*.

Local Alignment



- Asks for largest domain, sub-domain, or set of contiguous domains that are in common between two sequences.
- Worst score is always zero (0) for “no alignment”
- More appropriate than global or semi-global when there are no assumptions about the sequence relationship.
- Used for database searches.
- Required to obtain e-values

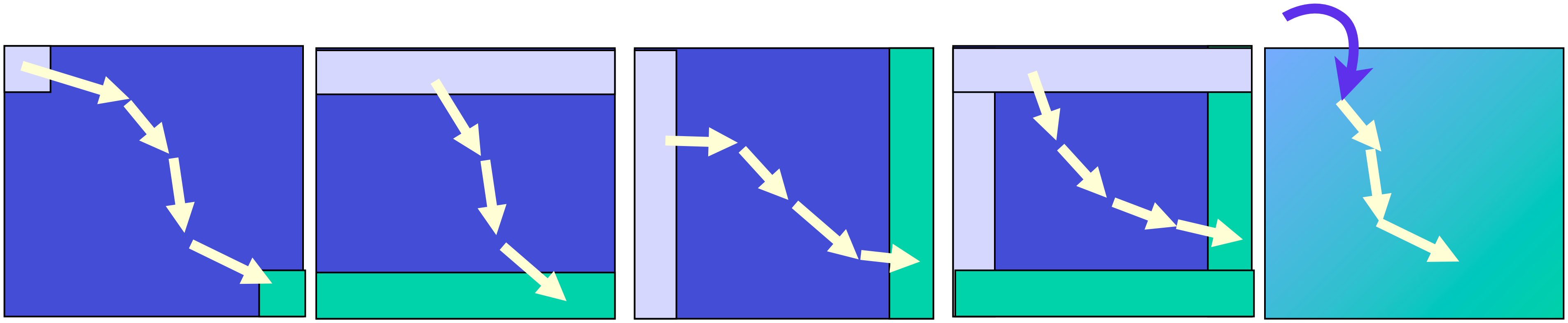
Global, semi-global, and local alignment

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.



The **optimal** alignment may be **no alignment**

If the maximum score in the alignment matrix is < 0., then the optimal local alignment has score = 0 and looks like this:

```
ATSFM~~~~~~~

~~~~~PGTSFEP
```

In class exercise: gaps

- In a browser, goto to NCBI. Search Protein database
- for **1DRF**. Save as FASTA file
- for **2DRC**. Do the same.
- Open both in Ugene. Select. Save as Alignment. Select. Align using Kalign. Gap open=12, Gap extend=3, Global with free end gaps.
- Count the number of gaps in the resulting alignment (initiations, not characters)
- Re-align (right click. Align, Kalign). Do the experiments on the next page.

Parametric search

gap extension
penalty

gap opening
penalty

	0	1	3	10
0	86, 28.8			>50, 24.7
1				
3				
12			6 , 26.2	
20				
50	0, 0.0			0, 0.0

Record: # of gaps , % Identity

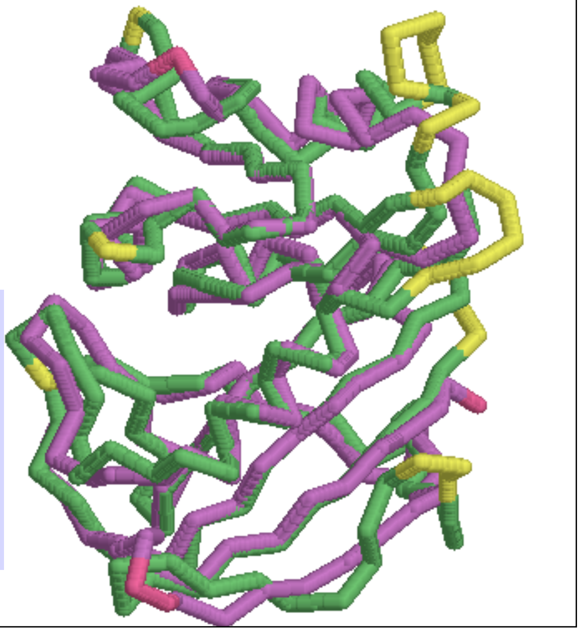
Structure-based alignments are the "gold standard"

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
```

Look carefully. What do you see? Lots of mismatches (id=38%), few gaps (8), gaps are 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



BAliBase

- A database of curated multiple sequence alignments derived from structure-based alignments. The Gold Standard for multiple sequence alignment!
- http://www-bio3d-igbmc.u-strasbg.fr/balibase/

Affine gap penalty-- theory

•Each gap represents an evolutionary event (duplication, polymerase stutter, deletion/ligation, etc.)

•If the alignment has "evolutionary distance" meaning, then the gap penalty score should be proportional to the number of gaps.

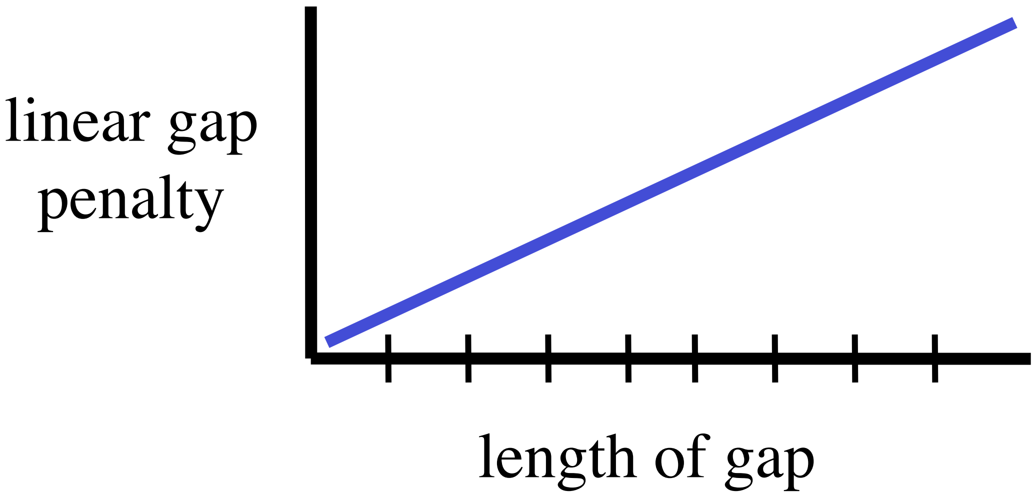
Are long gaps proportionally less likely?

Which alignment is intuitively better?

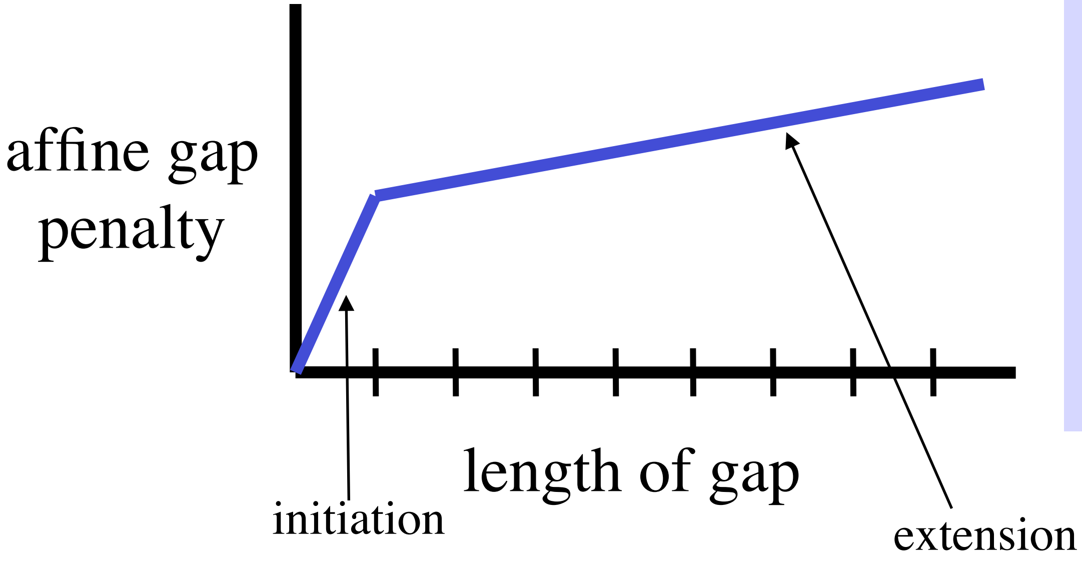
```
AGGCTACT~T~TCA
 GGCTACTATATCA
```

```
AGGCTACTTT~~CA
 GGCTACTATATCA
```

Linear versus Affine gap penalty. Making alignments agree with biological intuition.



Gap penalty for the whole sequence is just the total number of gap characters times a constant.



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

Example: affine gap

gap *initiation* = -5 gap *extension* = -1

```
        -5  -5
AGGCTACTT~T~TCA
 GGCTACTATATCA
```

-10

```
          -5 -1
AGGCTACTTT~~CA
 GGCTACTATATCA
```

-6

Affine Gap DP, either...

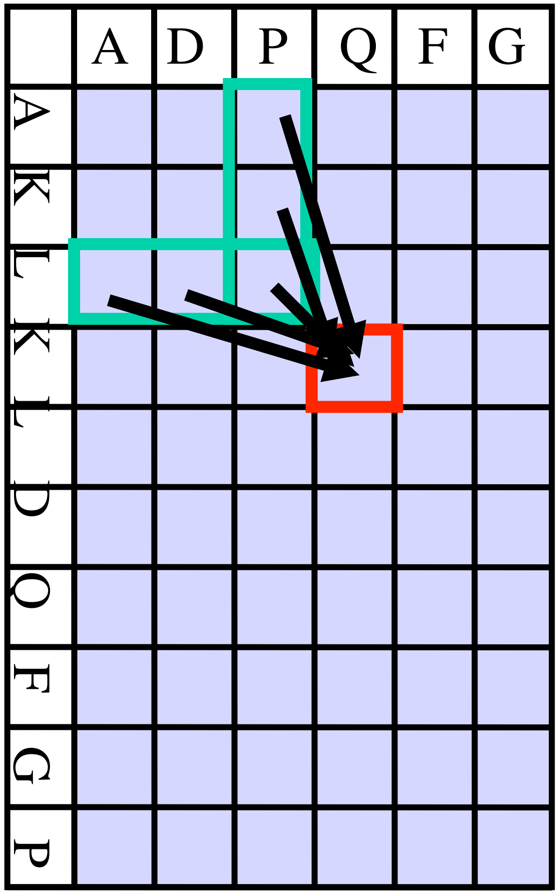
- You can have **5 types of arrows**, instead of just three.

(1) Match
(2) Open a gap in first sequence.
(3) Open a gap in second sequence.
(4) Extend a gap in first sequence.
(5) Extend a gap in second sequence.

---or---

- You can have variable length arrows.

Affine gap DP algorithm using variable length arrows



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

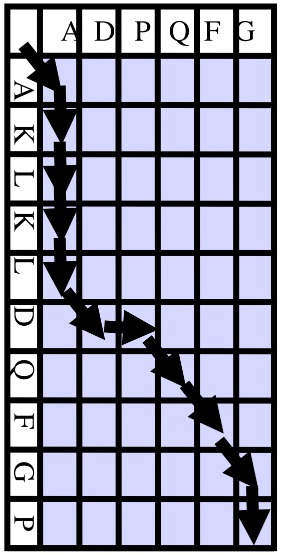
...where s(i,j) is the substiution score, *n* is the length of the gap, g_{init} is the gap initiation penalty, and g_{ext} is the gap extension penalty.

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.

Traceback for linear gap penalty

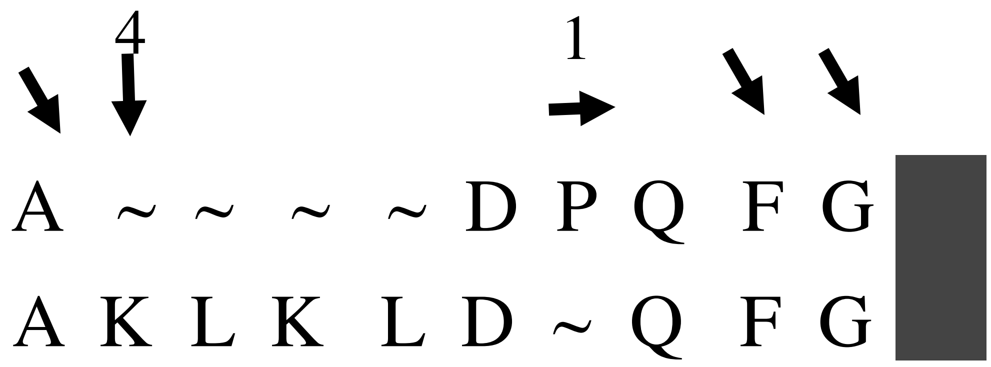
Each arrow advances either one sequence or both, by 1. Each column has one arrow.

```
↘  ↓  ↓  ↓  ↓  ↘  →  ↘  ↘  ↘  ↓
A  ~  ~  ~  ~  D  P  Q  F  G  ~
A  K  L  K  L  D  ~  Q  F  G  P
```



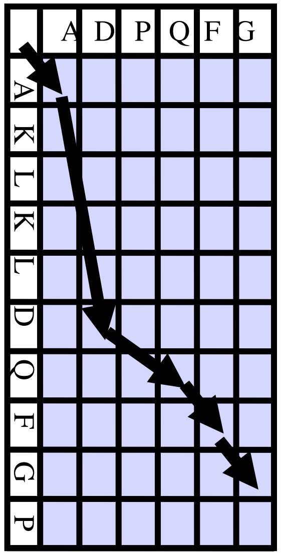
Traceback for affine gap DP

Each arrow advances one sequence by 1, the other sequence by *n*. Output of one arrow is *n* columns. Last of *n* columns is a match. Number of arrows is ≤ number of columns.



A ~ ~ ~ ~ D P Q F G

A K L K L D ~ Q F G



Does gap to gap make sense???

Special rules may apply for going from *I* to *D* and *D* to *I*.

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

If you think this alignment does not make sense, then D to I and I to D can simply be **disallowed** in the DP algorithm. Most programs do this.

[Exception: For a global alignment, D-to-I or I-to-D arrows are allowed at the ends of alignments because there is no other way to complete the matrix.]

What is the best gap penalty?

ANSWER: find out by “machine learning”. Here’s how.

- Create a database of sequence alignments (BAliBase)
  - Divide database into Training set and Test set.
  - Alignments must be **non-redundant**. No duplicates.
  - Alignments must be representative. Adding more won’t change results.
  - Test set must have essentially **no overlap** with Training set.
- Define an objective function, ε
  - ε is **one number**, a function of alignments given parameters.
  - ε approaches maximum as calculated alignments approach **true** alignments in Training set
- Explore parameter space (try all reasonable gap penalties)
  - may be exhaustive search,... or something smarter. Bracketing?
- Cross-validate.
  - Report the accuracy on a **Test set** not **Training set**.

Bioinformatics research story:

Bioinformatics 22(4):413-420, 2006

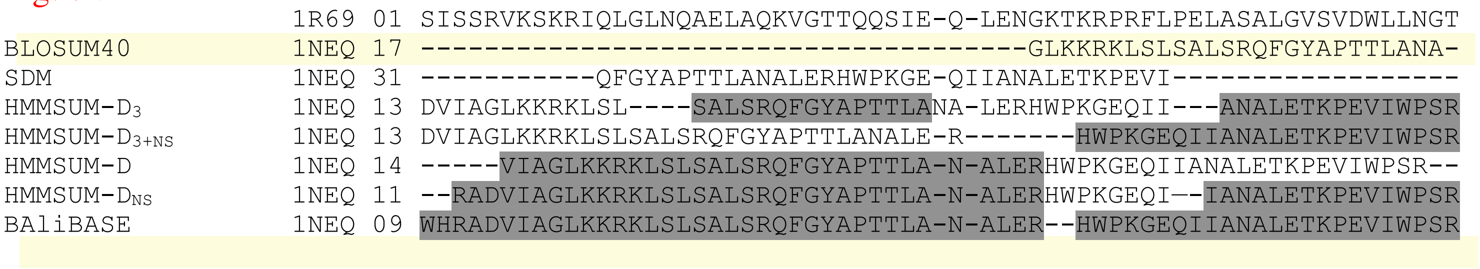
Improved pairwise alignments of proteins in the Twilight Zone using local structure predictions

*Yao-ming Huang and *Christopher Bystroff*

Center for Bioinformatics, Dept of Biology, Rensselaer Polytechnic Institute,
Troy, New York 12180 USA

Objective function ε: number of True Match characters (compared to bAliBase alignment).

Fig. 6-a



```
                 1R69 01 SISSRVKSKRIQLGLNQAELAQKVGTTQQSIE-Q-LENGKTKRPRFLPELASALGVSVDWLLNGT
BLOSUM40         1NEQ 17 --------------------------------------GLKKRKLSLSALSRQFGYAPTTLANA-
SDM              1NEQ 31 -----------QFGYAPTTLANALERHWPKGE-QIIANALETKPEVI------------------
HMMSUM-D3        1NEQ 13 DVIAGLKKRKLSL----SALSRQFGYAPTTLANA-LERHWPKGEQII---ANALETKPEVIWPSR
HMMSUM-D3+NS     1NEQ 13 DVIAGLKKRKLSLSALSRQFGYAPTTLANALE-R-------HWPKGEQIIANALETKPEVIWPSR
HMMSUM-D         1NEQ 14 -----VIAGLKKRKLSLSALSRQFGYAPTTLA-N-ALERHWPKGEQIIANALETKPEVIWPSR--
HMMSUM-DNS       1NEQ 11 --RADVIAGLKKRKLSLSALSRQFGYAPTTLA-N-ALERHWPKGEQI—-IANALETKPEVIWPSR
BAliBASE         1NEQ 09 WHRADVIAGLKKRKLSLSALSRQFGYAPTTLA-N-ALER--HWPKGEQIIANALETKPEVIWPSR
```

Different substitution matrices (left) gave different alignments when sequence similarity is in the "Twilight zone" (very difficult alignments).

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?*