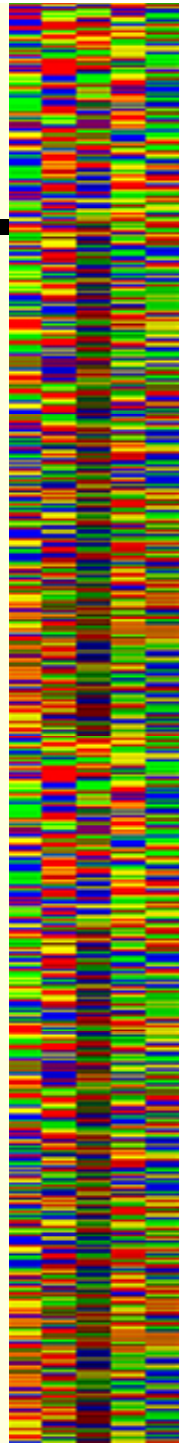
Genomics

8-12 September

6	M 8	MG	Scoring Matrices		Ch 3 and Ch 4
7	W 10	MG	Pairwise Alignment		
8	F 12	MG	Pairwise Alignment	Hw2	

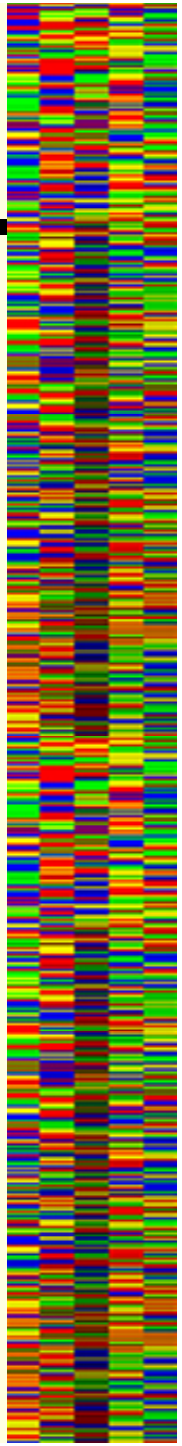
Reading: Mount - ch 3 and 4



Genomics - Sequence Alignment

Dynamic Programming Alignment

- ***Dynamic programming uses a recursive definition of an optimal alignment***
- ***Alignment is guaranteed to be "optimal"***
  - Given: the scoring systems used and gap penalties
- ***Don't confuse optimal with correct - Even unrelated sequences can be optimally aligned!***



Genomics - Sequence Alignment

Dynamic Programming Alignment

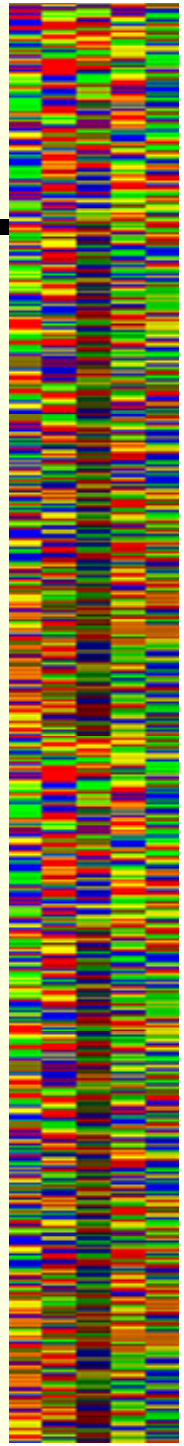
- *Assume this alignment is optimal:*

```
AATGC
|  ||
AG.GC
```

- *If we remove the last base pair, like so*

```
AATG        C
|  |   +    |
AG.G        C
```

- *The remaining part on the left HAS to be optimal*



Genomics - Sequence Alignment

Dynamic Programming Alignment

- ***What if the remainder isn't optimal?***

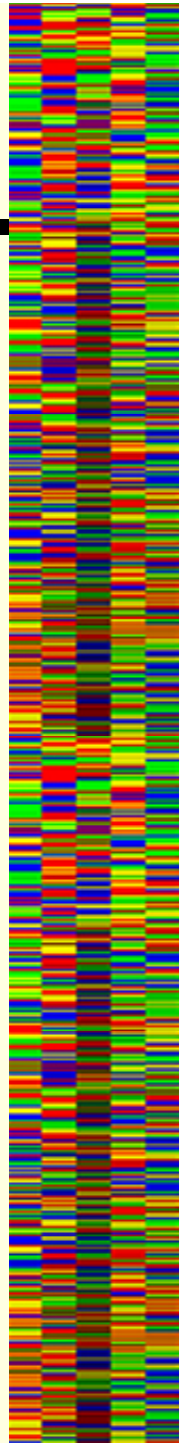
```
AATG        C
|  |   +   |
AG.G        C
```

- ***Suppose that there was something better***

```
AATG
|  :|
A.GG
```

- ***The original would not be optimal, there would be a better one***

```
AATGC                    AATGC
|   ||                   |  :||
AG.GC                    A.GGC
```



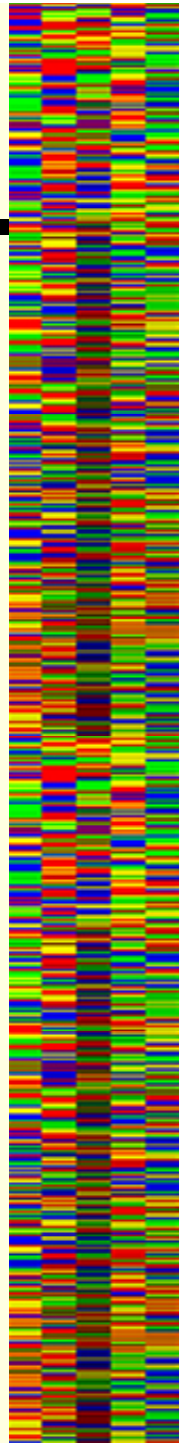
Dynamic Programming Alignment

- ***Remove one more pair***

```
AAT       G       C
|     +   |   +   |
AG.       G       C
```

- ***The remainder on the left has to be optimal***

- ***Look at this another way***
  - We can tell what the optimal alignment containing the G:G and C:C pairs is, IF we know the optimal alignment up to that point
  - i.e., if we know the optimal alignment between AAT and AG



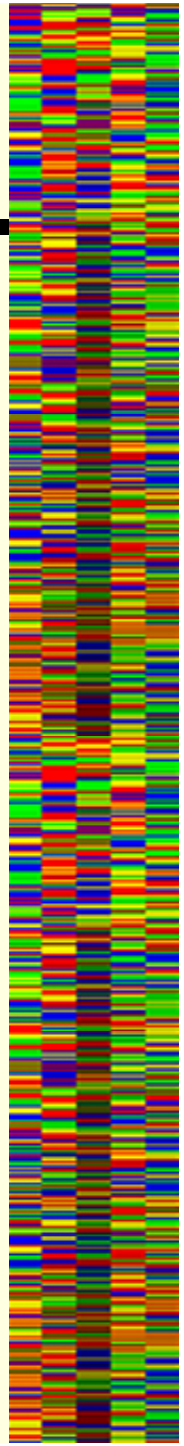
Genomics - Sequence Alignment

Dynamic Programming Alignment

- ***What are the possibilities?***

```
AAT        AAT        AAT
|          |           |
AG.        A.G        .AG
```

- ***Alignment must use the next position in one of the sequences because we can't align gaps with gaps***
- ***If we can't enumerate the possibilities, we can remove another position***
  - there are only three possibilities: (T.) ,(TG) or (G.)



Genomics - Sequence Alignment

Dynamic Programming Alignment

- ***Pick the best possible alignment of AAT and AG and then add the pairs we removed back***

```
AAT       G       C             AATGC
|     +   |   +   |     ->      |  ||
AG.       G       C             AG.GC
```

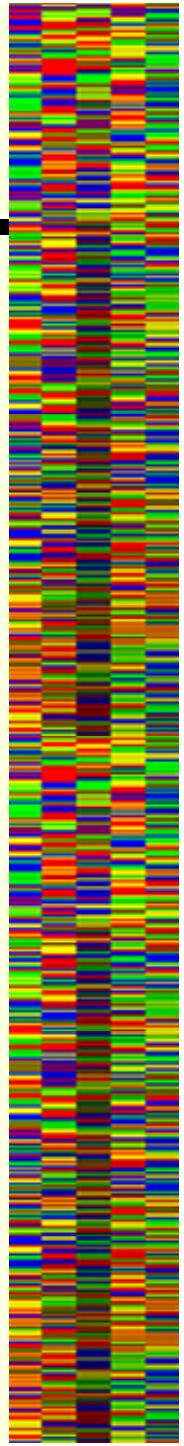
This is the best alignment containing G:G and C:C

- ***If we did the same thing without removing the G:G pair, it would be the best alignment ending in C:C***

Genomics – Sequence Alignment

Dynamic Programming alignment

- ***Recursive process***
  - For each possible ending pair, calculate the score as the score for the pair + the best score for the part before that pair (left-hand part)
  - If the left-hand part is too big, for each possible ending pair do the same
  - If the left hand part is simple, e.g., a single pair look up score in scoring table
  - Fortunately, many of the left-hand parts can be reused in the calculation



Genomics - Sequence Alignment

Dynamic Programming Alignment

- *Dynamic programming alignments look at every possible terminal pair and recursively calculates the optimal alignment*

```
        A A T G C              A A T G C

         A G G C                A G G C

AATG  C    AAT GC    AA  TGC     A   ATGC        AATGC
AGG   C    AGG C.    AGG C..     AGG C...    AGG C....

AATG C    AATG C.   AATG C..   AATG C...
AGG  C    AG   GC   A    GGC        AGGC
```

Genomics - Sequence Alignment

Dynamic Programming Alignment

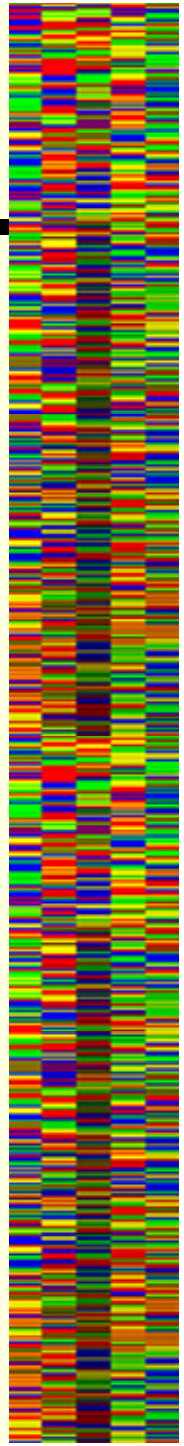
- ***Starting from a terminal pair (5' or C-terminal, right end) and looking back to the left, for instance if we consider the (CCT) pair***

  ```
  AATG  C
  AGG   C
  ```

  the previous pair must use either the

  - previous letter in both sequences (GG),
  - the previous letter in the top sequence vs a gap (G.), or
  - the previous letter in the bottom sequence vs a gap (.G)
- ***Because, we don't consider, gaps vs gaps to be alignments***

  ```
  AATG.C
  AGG..C
  ```



Genomics - Sequence Alignment

Dynamic Programming Recursion

- For a sequence A with characters A_i where $1 \le i \le n$, and sequence B with characters B_j , $1 \le j \le m$, where $s_{i,j}$ is the score for aligning letter A_i and B_j
- The score S_{ij} for the alignment ending with character A_i aligned with character B_j is

$$S_{i,j} = s_{i,j} + \max \begin{Bmatrix} S_{i-1,j-1} \\ \max\limits_{k=2,i-1} \left(S_{i-k,j-1} + \text{gap}_k \right) \\ \max\limits_{k=2,j-1} \left(S_{i-1,j-k} + \text{gap}_k \right) \end{Bmatrix}$$

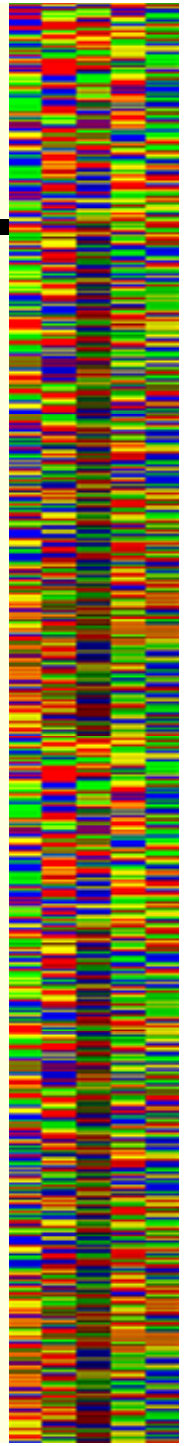
- Where k, the gap length, determines the gap penalty

$$\text{gap}_k = g_{\text{open}} + (k-1) \cdot g_{\text{extension}}$$

Genomics - Sequence Alignment

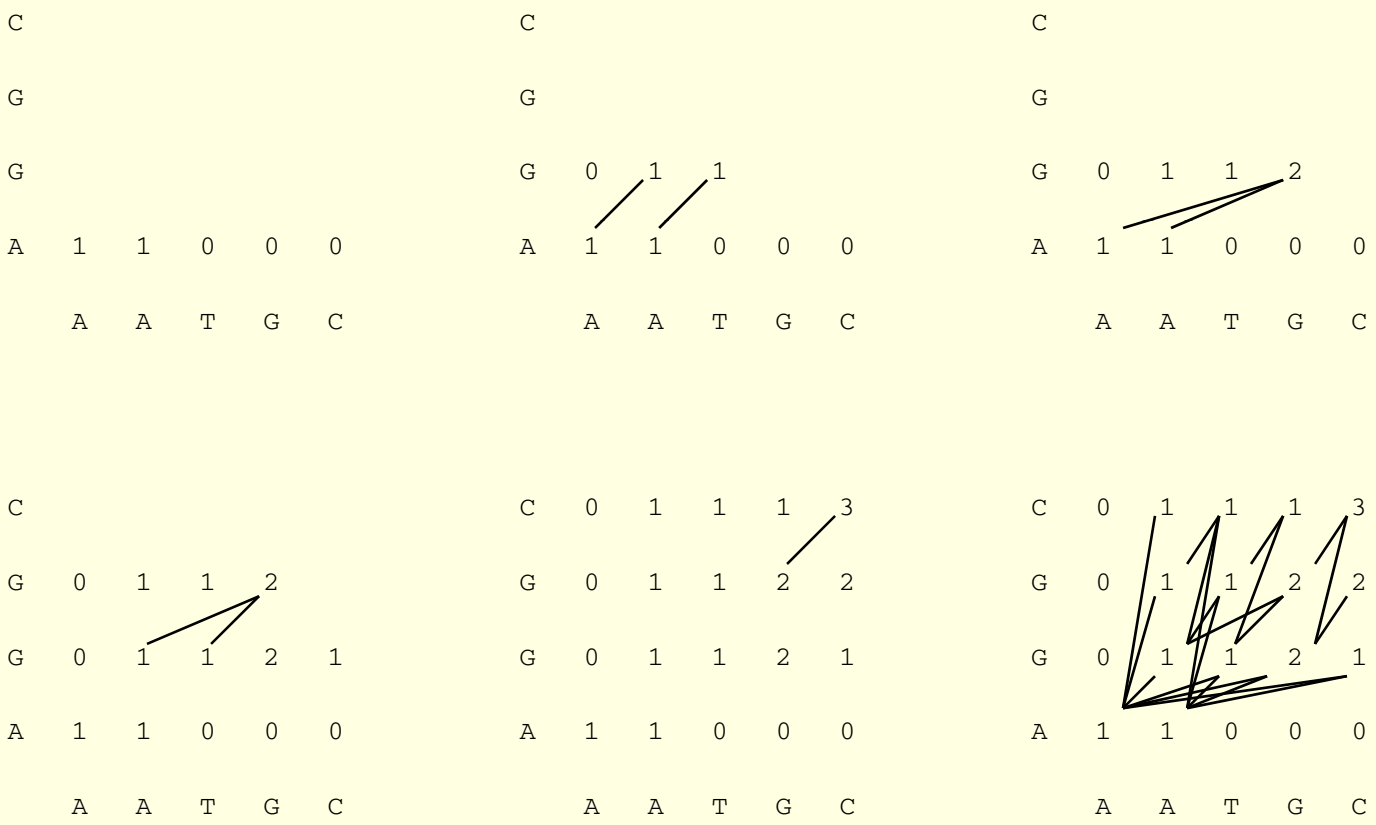
Dynamic Programming Alignment

- ***A simple example, alignment of AATGC and AGGC***
- ***Scoreing system***
  - Match = +1
  - Mismatch = 0
  - Gaps = 0
- ***Global alignment (all sequence characters used from both sequences)***



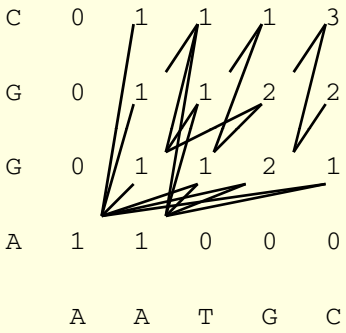
Genomics - Sequence Alignment

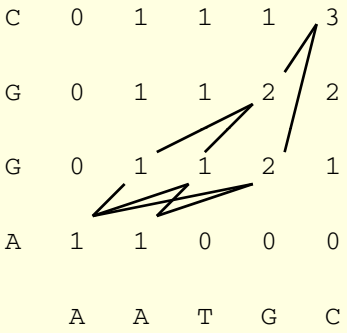
Dynamic Programming Alignment



Genomics - Sequence Alignment

Dynamic Programming Alignment





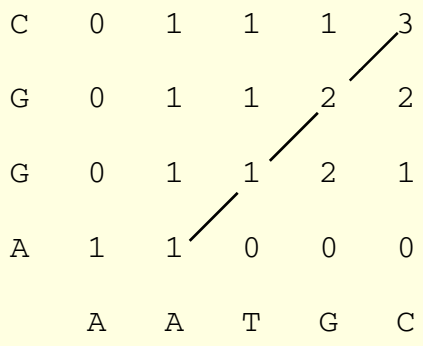
5 equivalent alignments, EVERY ONE IS OPTIMAL

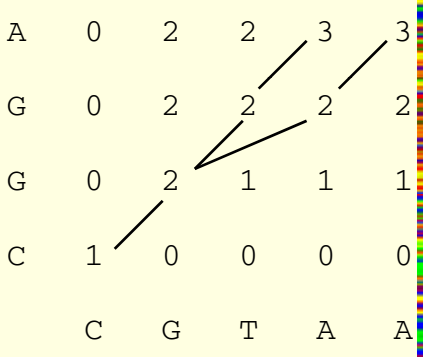
```
AG.GC     A.GGC     .AGGC     .A.GGC     A..GGC
AATGC     AATGC     AATGC     AATG.C     AATG.C
```

Genomics - Sequence Alignment

Dynamic Programming Alignment

- ***Most alignment programs save only one path pointer per position***





- ***How do you know if there is more than one alignment?***
  - Try reversing sequences
  - Look at dotplot

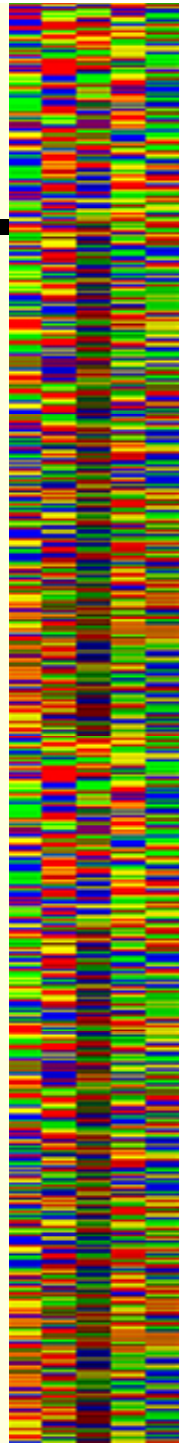
Genomics - Sequence Alignment

Global Alignments

- *use any scoring system*
- *generate alignments that use all letters from each both sequences*
- *best score is on the edge of the dp matrix*
- *most suitable for situations where you know you have end-to-end match – i.e., orthologous genes/proteins*

Local alignments

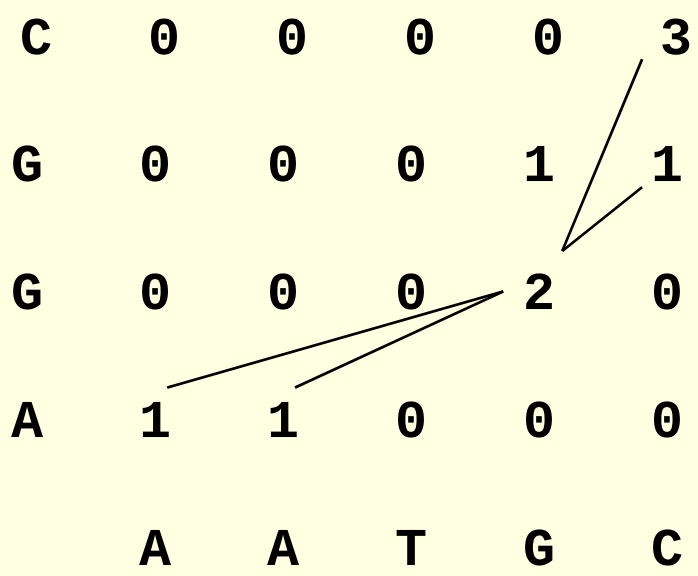
- *Use positive and negative scores, average must be below 0!*
- *Values are truncated at zero*
- *Maximum can be anywhere in the dp matrix*
- *most suitable when lengths are different or with incomplete or nested match situations*



Genomics - Sequence Alignment

Local Alignment (same sequences as before)

- *Match = +1, mismatch = -1, gaps = 0*



Genomics - Sequence Alignment

Local Dynamic Programming Alignment

E	0	0	2	1	3	2	3	4	5	6	7	4	4	6	4	5	8	7	6	7	6	6	8	6	6	6	7	**11**
N	0	0	0	1	1	2	3	6	5	6	5	5	4	4	6	5	6	7	8	6	5	5	5	6	5	6	**10**	7
I	0	0	0	1	1	2	3	4	5	6	6	5	5	5	5	5	5	6	7	6	6	5	5	5	6	5	**9**	6	6
L	0	0	0	1	1	2	3	4	5	7	6	6	6	6	6	6	6	6	7	6	6	6	6	6	6	**8**	6	6	6
D	0	0	0	1	1	2	3	4	**8**	5	5	5	5	5	5	5	6	7	6	7	6	6	6	6	6	6	6	6	6
N	0	0	0	1	1	2	3	**7**	4	4	4	4	4	4	6	5	6	7	8	5	6	5	5	6	5	5	7	5	
O	0	0	0	1	1	2	**6**	3	3	3	3	3	3	3	4	5	6	7	6	7	5	5	5	6	6	5	5	5	
C	0	0	0	1	1	**5**	2	2	2	2	2	3	2	3	4	7	6	7	6	6	6	6	6	7	6	6	6	6	
E	0	0	2	1	**4**	2	2	1	1	1	4	2	2	5	4	5	8	7	7	6	6	6	8	6	6	6	6	8	
S	0	0	0	**3**	1	2	2	2	1	3	1	3	2	3	4	5	6	8	7	7	7	7	7	7	7	7	7	7	
E	0	0	2	1	3	2	3	2	2	2	4	2	2	5	4	5	9	6	6	6	6	6	8	6	6	6	6	8	
C	0	0	0	1	1	4	1	0	2	1	1	1	2	3	4	8	5	5	5	5	5	5	5	5	5	5	5	5	
N	0	0	0	1	1	2	1	3	1	1	1	1	2	3	7	4	4	4	6	4	4	4	4	4	4	4	6	4	
E	0	0	2	1	3	2	1	1	0	0	2	1	2	6	3	3	5	3	3	3	3	3	5	3	3	3	3	5	
U	0	0	0	1	1	2	2	1	1	1	1	1	5	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
Q	0	0	0	1	1	3	2	2	2	2	2	4	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	
E	0	0	2	1	4	1	1	1	1	1	3	1	1	3	1	1	3	1	1	1	1	1	3	1	1	1	1	3	
S	0	0	0	3	1	1	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	2	1	1	1	1	1	1	
T	1	0	0	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3	1	1	1	1	1	1	1	
S	0	0	0	3	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	1	1	1	
R	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	1	0	
I	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2	0	0	
F	0	0	0	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	
E	0	0	**3**	0	2	0	0	0	0	0	2	0	0	2	0	0	2	0	0	0	0	0	3	0	0	0	0	2	
H	0	**2**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	
T	**1**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	
	T	H	E	S	E	C	O	N	D	S	E	Q	U	E	N	C	E	A	N	O	T	H	E	R	L	I	N	E	

Match = 1
Mismatch = -1
Gap Initiation = -1
Gap Extension = 0

Genomics - Sequence Alignment

Dynamic Programming Alignment

		T	H	E	S	E	C	O	N	D	S	E	Q	U	E	N	C	E	A	N	O	T	H	E	R	L	I	N	E
E	0	0	2	1	3	2	3	4	5	6	7	4	4	6	4	5	8	7	6	7	6	6	8	6	6	6	7	11
N	0	0	0	1	1	2	3	6	5	6	5	5	4	4	6	5	6	7	8	6	5	5	5	6	5	6	10	7
I	0	0	0	1	1	2	3	4	5	6	6	5	5	5	5	5	5	6	7	6	6	5	5	5	6	5	9	6	6
L	0	0	0	1	1	2	3	4	5	7	6	6	6	6	6	6	6	7	6	6	6	6	6	6	8	6	6	6
D	0	0	0	1	1	2	3	4	8	5	5	5	5	5	5	5	6	7	6	7	6	6	6	6	6	6	6	6
N	0	0	0	1	1	2	3	7	4	4	4	4	4	4	6	5	6	7	8	5	6	5	5	6	5	5	7	5
O	0	0	0	1	1	2	6	3	3	3	3	3	3	3	4	5	6	7	6	7	5	5	5	6	6	5	5	5
C	0	0	0	1	1	5	2	2	2	2	2	3	2	3	4	7	6	7	6	6	6	6	6	7	6	6	6	6
E	0	0	2	1	4	2	2	1	1	1	4	2	2	5	4	5	8	7	7	6	6	6	8	6	6	6	6	8
S	0	0	0	3	1	2	2	2	1	3	1	3	2	3	4	5	6	8	7	7	7	7	7	7	7	7	7	7
E	0	0	2	1	3	2	3	2	2	2	4	2	2	5	4	5	9	6	6	6	6	6	8	6	6	6	6	8
C	0	0	0	1	1	4	1	0	2	1	1	1	2	3	4	8	5	5	5	5	5	5	5	5	5	5	5	5
N	0	0	0	1	1	2	1	3	1	1	1	1	2	3	7	4	4	4	6	4	4	4	4	4	4	4	6	4
E	0	0	2	1	3	2	1	1	0	0	2	1	2	6	3	3	5	3	3	3	3	3	5	3	3	3	3	5
U	0	0	0	1	1	2	2	1	1	1	1	1	5	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Q	0	0	0	1	1	3	2	2	2	2	2	4	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
E	0	0	2	1	4	1	1	1	1	1	3	1	1	3	1	1	3	1	1	1	1	1	3	1	1	1	1	3
S	0	0	0	3	1	1	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	2	1	1	1	1	1	1
T	1	0	0	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3	1	1	1	1	1	1	1
S	0	0	0	3	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	1	1	1
R	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	1	0
I	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2	0	0
F	0	0	0	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1
E	0	0	3	0	2	0	0	0	0	0	2	0	0	2	0	0	2	0	0	0	0	0	3	0	0	0	0	2
H	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0
T	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Match = 1
Mismatch = -1
Gap Initiation = -1
Gap Extension = 0

Genomics - Sequence Alignment

Dynamic Programming Alignment

- ***Optimal alignment is "better"***

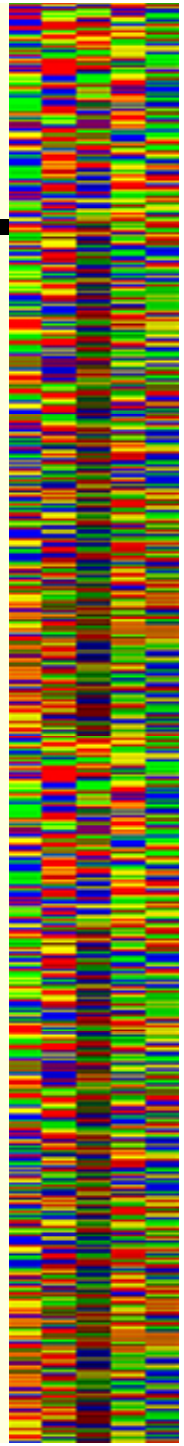
```
THEFIRSTSEQUENCESECOND----------------LINE
|||              ||||||                  ||||
THE--------------SECONDSEQUENCEANOTHERLINE
```

  - 13 Matches, 0 Mismatches, 2 Gaps: Score = 11

- ***Alternate alignment is "worse"***

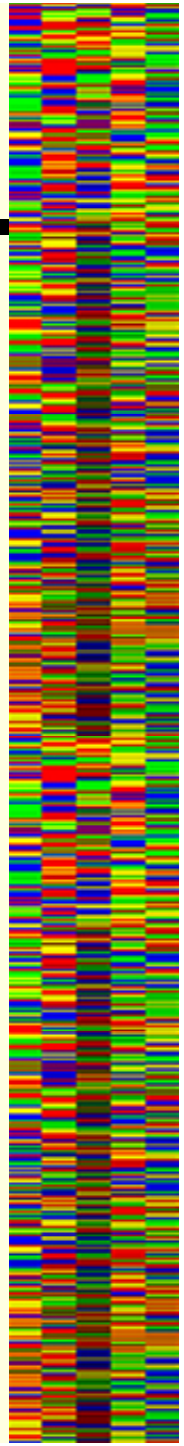
```
THEFIRST----SEQUENCESECOND----LINE
|||    |     ||||||||     |     ||||
THE---SECONDSEQUENCEA---NOTHERLINE
```

  - 17 Matches, 3 Mismatches, 4 Gaps: Score = 10



Dynamic Programming Alignment

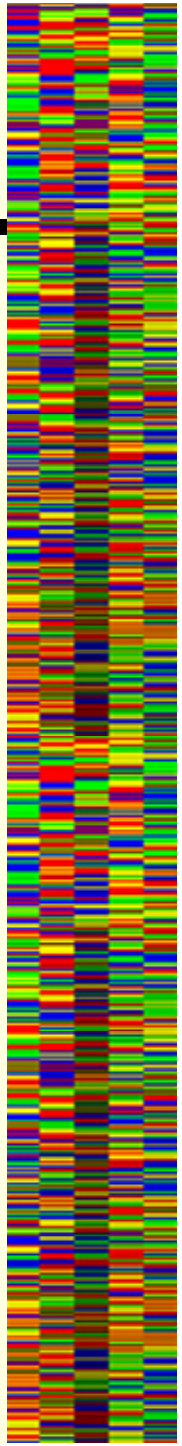
- ***End Gaps***
  - The gaps at the end are “special”
    - Many proteins differ at the ends with little effect on structure
    - Sequence may be a fragment
  - Using a gap cost=0 for end gaps was an early approximation to local alignment
  - produces a quasi-local alignment in which weakly matched residues/bases at end are pushed right to avoid the cost of the first gap (by using an end gap instead)



Genomics - Sequence Alignment

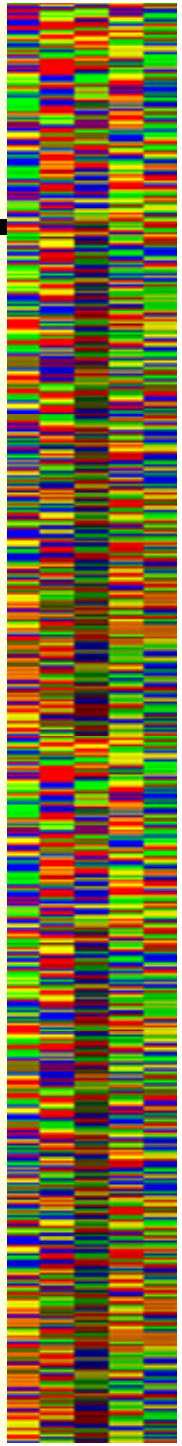
Dynamic Programming Alignment

- ***Global vs local methods***
  - Global - uses all positions of both sequences
  - Local - best region of alignment
- ***Use global if***
  - You expect and want the entire sequence to match, *e.g.* two close homologs
- ***Use local***
  - You expect only part of the sequences to match
  - Sequences are very different in length



Dynamic Programming Alignment

- ***Gap penalties***
  - Affine gap penalty
    - Penalty = Init + Ext x Length (or length-1)
    - Init = length independent term (gap opening, penalty)
    - Ext = length dependent term (bias)
  - Recommended values (empirical)
    - Init approx 3 x identities
    - Ext about 5% or less of Init



Genomics - Sequence Alignment

Dynamic Programming Alignment

```
                .         .         .         .         .
151 VEKGGKHKTGPNLHGLFGRKTGQAPGYSYTAANKNKGIIWGEDTLMEYLE 200
                          ... .   |   |    . :   | ||
  1 ...................VLSPADKTNVKAAWGKVGAHAGEYGAEALE 30
          .         .         .         .         .
201 NPKKYIPGTKMIFVGIKKKEERADLIAYLKKATNE.............. 235
        .| ||  |  .   .  |.. .  || ..
 31 RMFLSFPTTKTYFPHFDLSHGSAQVKGHGKKVADALTNAVAHVDDMPNAL 80
```

Global alignment
PAM250 scoring matrix
Gap opening = -12
Gap extension = -4

```
 52 LSDGEWQLVLNVWGKVEADIPGHGQEVL 79
    || .:   |   |||| |.    | | |
  2 LSPADKTNVKAAWGKVGAHAGEYGAEAL 29
```

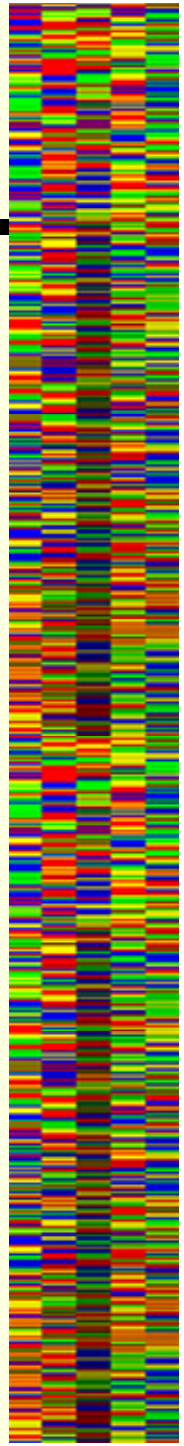
Local
PAM250 scoring matrix
Gap opening = -12
Gap extension = -4

Genomics - Sequence Alignment

Problems with alignments

- *rearrangements: ABC vs CAB*
- *duplications and repeated sequences*
- *low entropy or degenerate sequences*
- *mismatch between biological "truth" and mathematical model*

- *Note that dotplots, even though they are qualitative don not sufer from these problems*



Genomics - Sequence Alignment

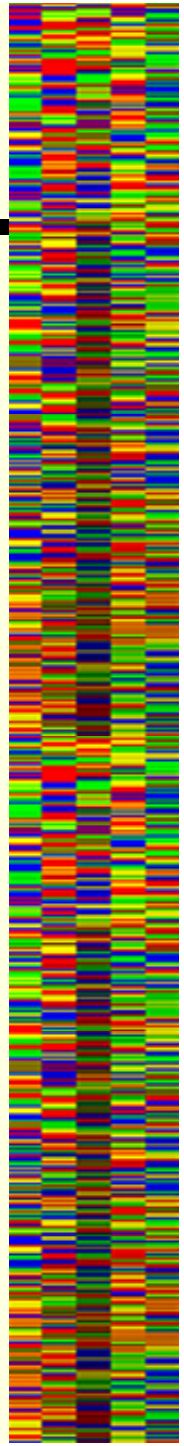
Dynamic Programming Alignment

Review

- ***Dynamic programming alignment is mathematically rigorous***
  - considers all possible alignments with all possible gaps in all possible positions
- ***Two steps***
  - Build score matrix - each cell represents the score for the best alignment that ends in that position
  - Follow “best previous” pointers back to get alignment (traceback)
- ***Highly dependent on the scoring system and gap penalties***

Points to remember

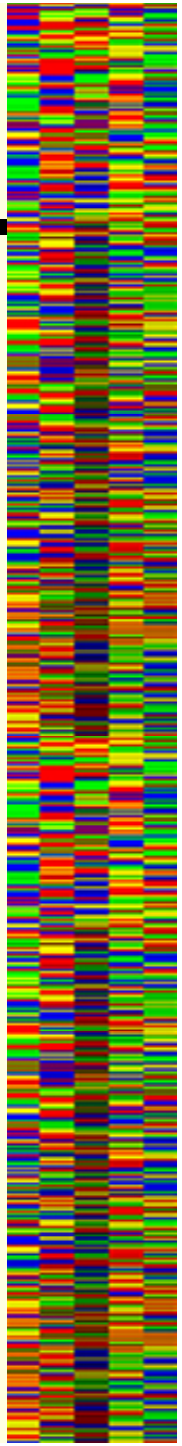
- ***There may be other equally good alignments***
- ***Gap penalties have a large effect on alignments***
  - Do not rely on a single setting
- ***Are you using a global or local method ?***
- ***Are end gaps weighted or unweighted?***
- ***How does your score compare to unrelated sequences?***



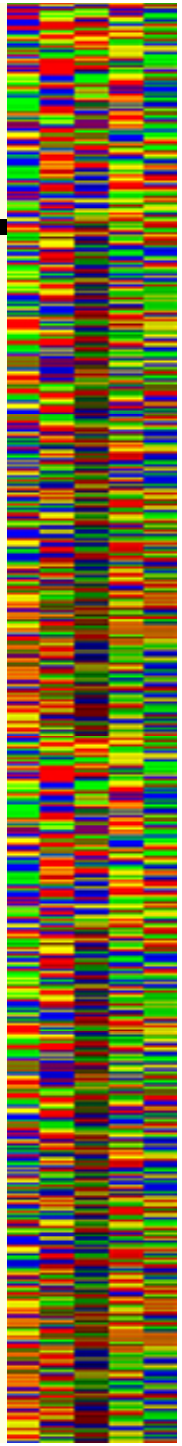
Genomics - Sequence Alignment

Statistics

- *Alignments are often evaluated using a random sequence model and a Monte Carlo procedure*
- *If a sequences are significantly more similar than random sequences, they must be homologous sequences*



Genomics - Sequence Alignment

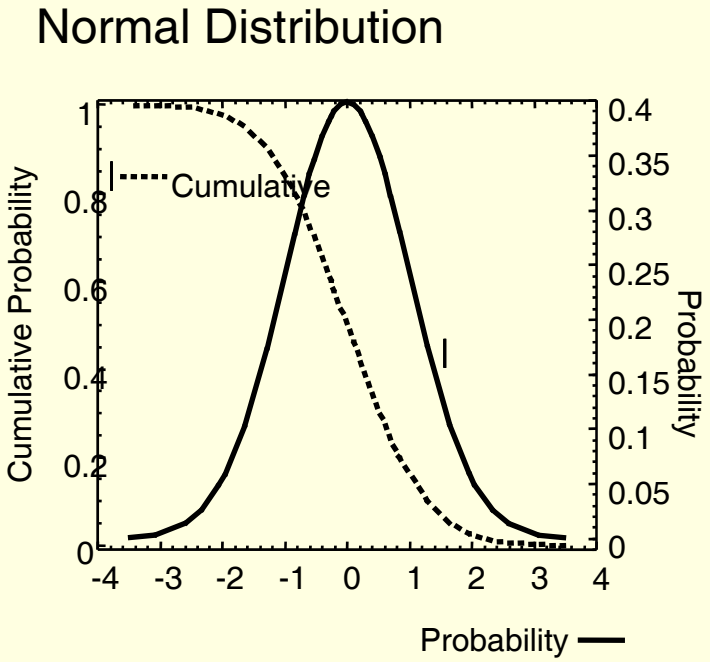


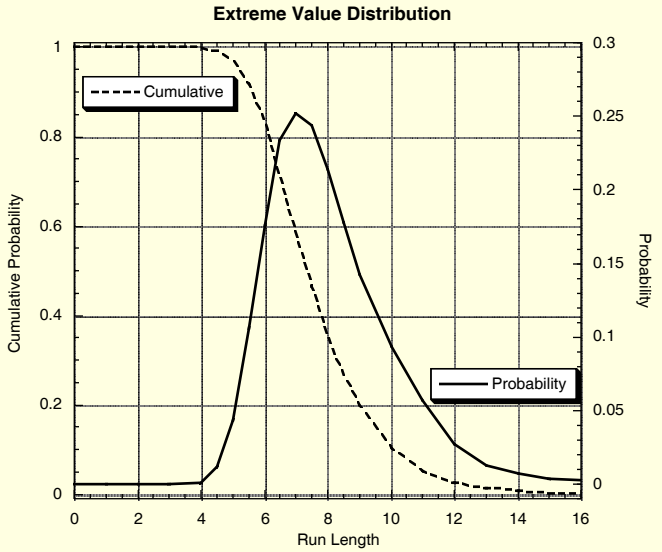
Monte Carlo Approach

- ***Compares result to randomized result, similarly to results generated by a roulette wheel at Monte Carlo***
- ***Typical procedure for alignments***
  - Randomize sequence A
  - Align to sequence B
  - Repeat many times (hundreds - thousands)
  - Use average as expected score to predict behavior of unrelated sequences
- ***A common statistic is the Z score (standardized score, standard normal deviate)***
  - Z = (Obs_score - Exp_score) / Std_deviation
  - Expected score depends on model
- ***Bad Rule:***
  - Z < 3 No evidence of homology
  - 3 < Z < 6 Homology possible
  - 6 < Z Strong evidence of homology, (Z > 8) better

Genomics - Sequence Alignment

Matches without optimization follow a normal destribution

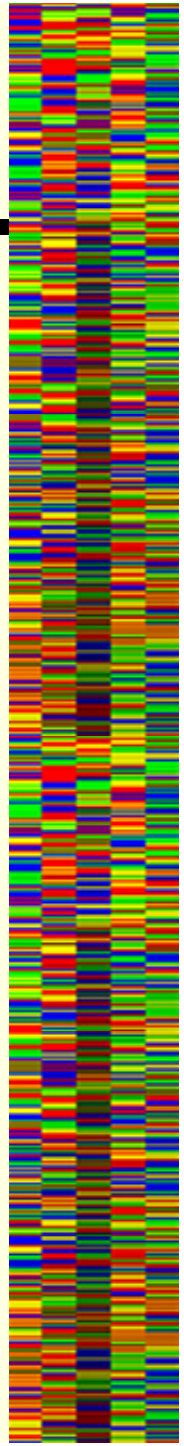




Genomics - Sequence Alignment

Extreme Value Distributions

- ***Z score can't be directly converted to probability because it not a "Normal" or "Gaussian" distribution***
  - e.g. Z=3 has a normal P-value = 0.0013 but an extreme value distribution P-value ~ 0.12
  - about 100-fold error (error gets worse for smaller P-values)
- ***Whenever you are looking at a distribution of maxima***
  - longest run of heads in coin toss
  - maximum scores for each sequence in database



Genomics - Sequence Alignment

Updated Monte Carlo procedure for dynamic programming alignments

- ***Karlin-Altschul statistics are approximately correct***

  $n = KNe^{-\lambda S}$

  $\ln(n/N) = \ln K - \lambda S$ *this is a linear equation*

  $y = b + mx$

- ***Plot the log of the "observed P-value" vs score for randomized alignments of the same length and composition and determine P from the linear plot***
  - Randomize sequences and align using same parameters a large number of times, e.g. >10,000. Rank results by score. Observed P-value is rank divided by number of samples