

# Sequence alignment

Progressive multiple alignment and  
iterative refinement

# Outline

- Progressive alignment
- Iterative refinement

# Tree-based progressive alignments

- Key heuristics:
  - Construct a multiple alignment through a series of pairwise alignments
  - Align the “most similar” sequences first
- Align sequences according to a guide tree
  - leaves represent sequences
  - internal nodes represent alignments
- Determine alignments from bottom of tree upward
  - return multiple alignment represented at the root of the tree

# Tree-based progressive alignment

- Depending on the internal node in the tree, we may have to align a
  - a sequence with a sequence
  - a sequence with a partial alignment
  - a partial alignment with a partial alignment
- In all cases we use the same basic pairwise alignment algorithm, but will modify the the scoring system
  - For sequence with a sequence, we use the standard scoring system for pairwise alignments
  - For aligning alignments or a sequence to an alignment, we use sum of pairs scoring

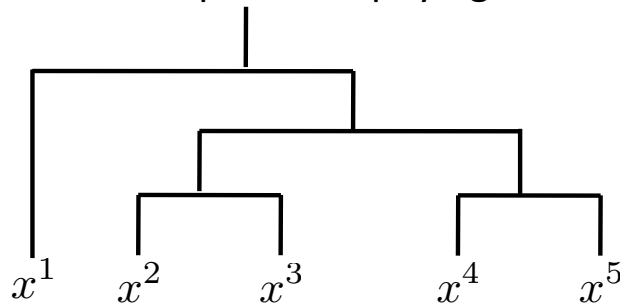
# Tree alignment example

- Starting sequences

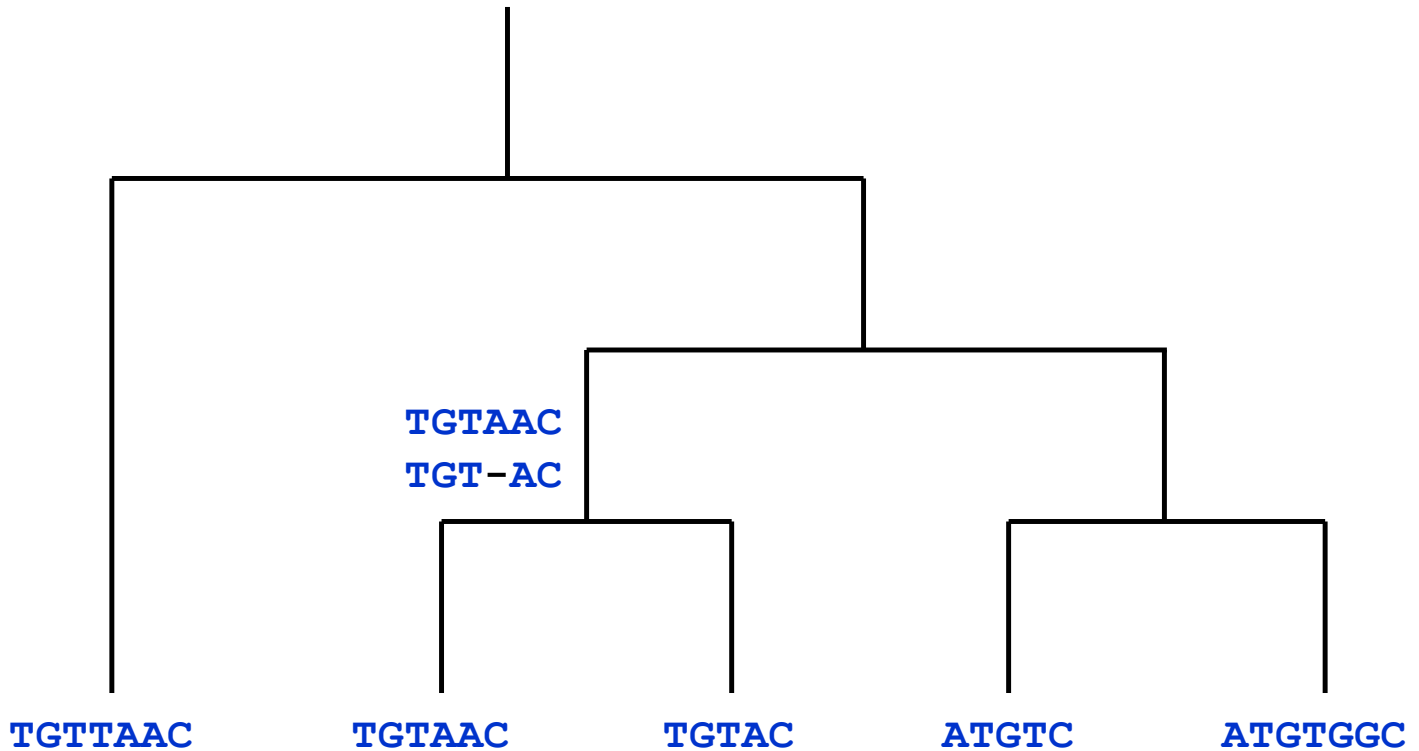
$x^1$  **TGTTAAC**  
 $x^2$  **TGTAAC**  
 $x^3$  **TGTAC**  
 $x^4$  **ATGTC**  
 $x^5$  **ATGTGGC**

- Create a guide tree

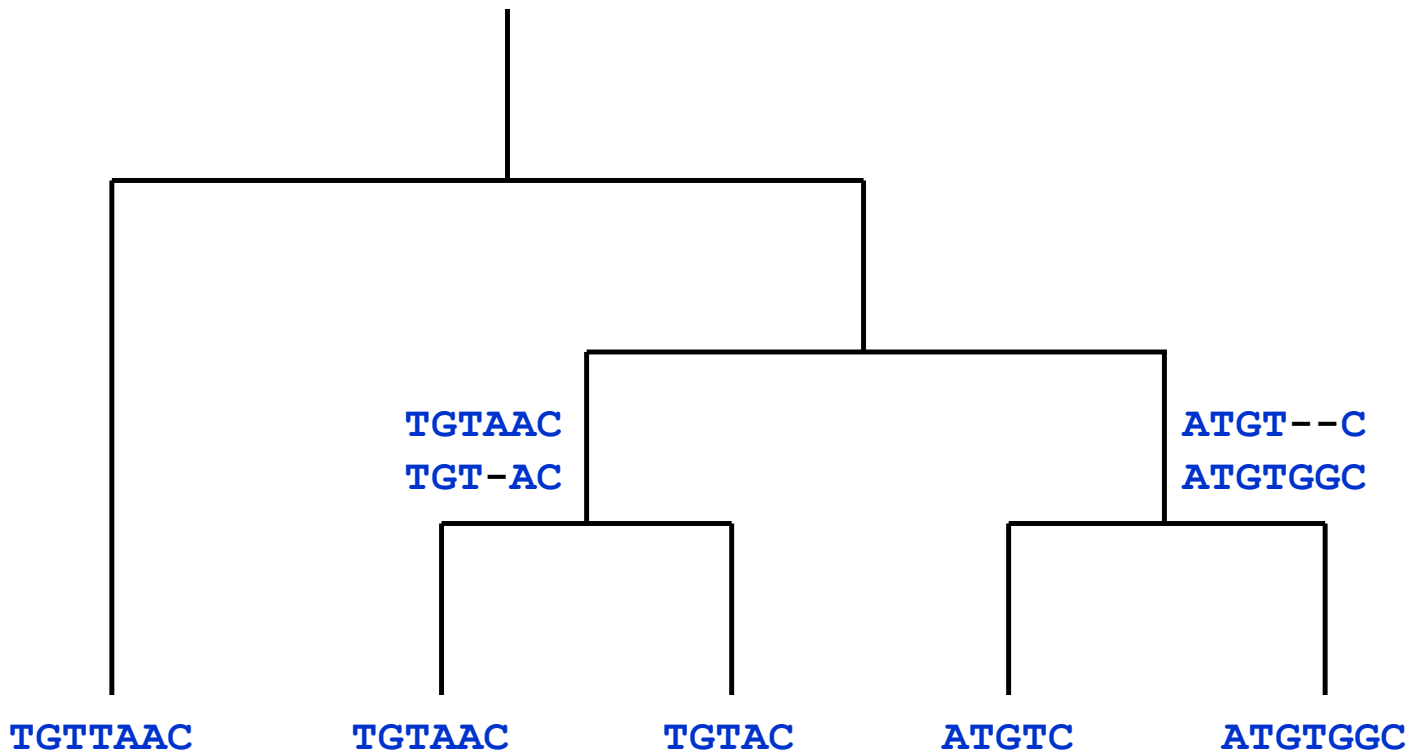
- Using pairwise distances (we will cover this in subsequent lectures)
- Approach similar to but simpler than phylogenetic trees



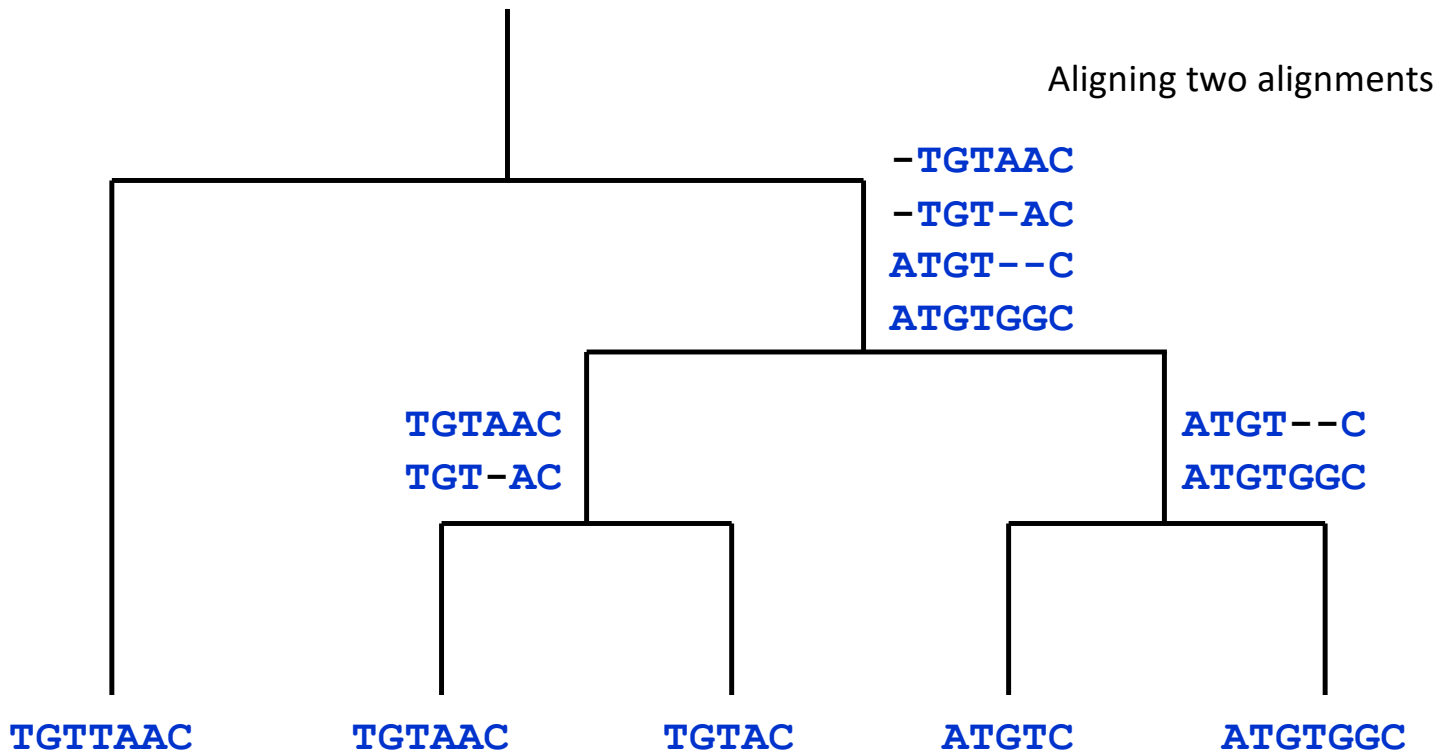
# Tree Alignment Example



# Tree Alignment Example



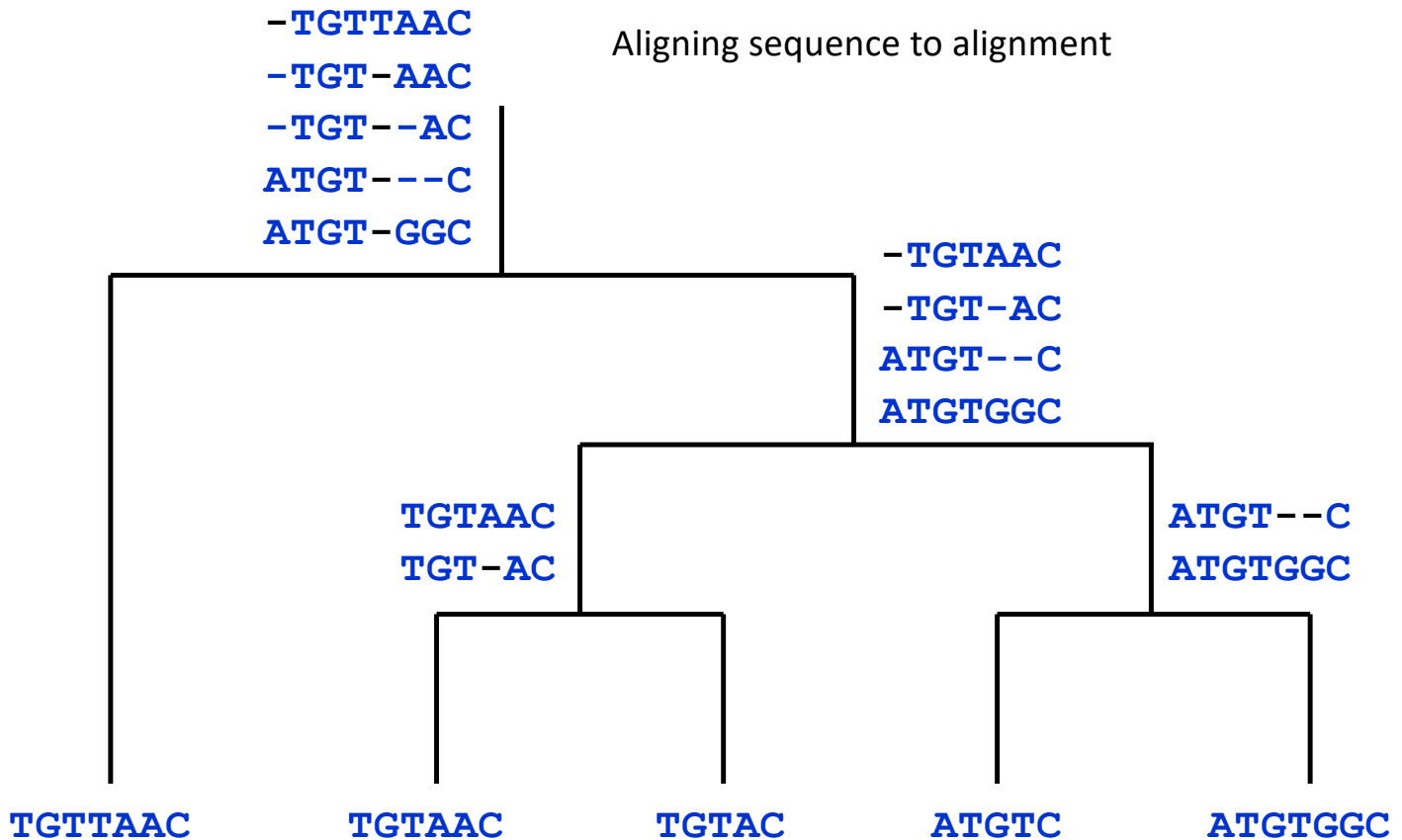
# Tree Alignment Example





# Tree Alignment Example

Aligning sequence to alignment



# Scoring an alignment of partial alignments

- Recall the sum of pairs score for a column  $i$

$$S(m_i) = \sum_{k < l} s(m_i^k, m_i^l)$$

- Let  $1$  to  $n$  represent sequences from the first alignment
- Let  $n+1$  to  $N$  represent sequences from the second alignment,  $N$  denotes total number of sequences
- Alignment at column  $i$  can be written as

$$\begin{aligned} S(m_i) = & \sum_{k < l \leq n} s(m_i^k, m_i^l) && \text{Within first alignment} \\ & + \sum_{n < k < l \leq N} s(m_i^k, m_i^l) && \text{Within second alignment} \\ & + \sum_{k \leq n, n < l \leq N} s(m_i^k, m_i^l) && \text{Between two alignments} \end{aligned}$$

# Aligning two alignments

- Assume we have two alignments corresponding to intermediate nodes of the guide tree

Alignment A1

AAAC

-GAC

Alignment A2

AGC

ACC

- Alignment of two alignments = pairwise alignment of sequences of *columns*
- Filling entry  $(i, j)$  of the DP matrix we maximize over
  - aligning column  $i$  in A1 to a column  $j$  in A2
  - aligning column  $i$  in A1 to gaps in A2
  - aligning column  $j$  in A2 to gaps in A1

## Comments about tree-based progressive alignment

- Exploits partial alignment information
- But, greedy
  - The tree might not be correct, that is, reflect an incorrect ordering of how sequences should be stacked up in the alignment
  - Final results prone to errors in alignment
    - Some positions might be misaligned (that is have a lower score than if a different ordering is used).

# Ordering matters

Consider aligning GG, DGG and DGD

1	2
D G D	D G D
- G G	G G -

Are as good. But when we include DGG

1	2
D G D	D G D
- G G	G G -
D G G	D G G

1 is better than 2, assuming a match score of 2, mismatch score =1, gap penalty=-2

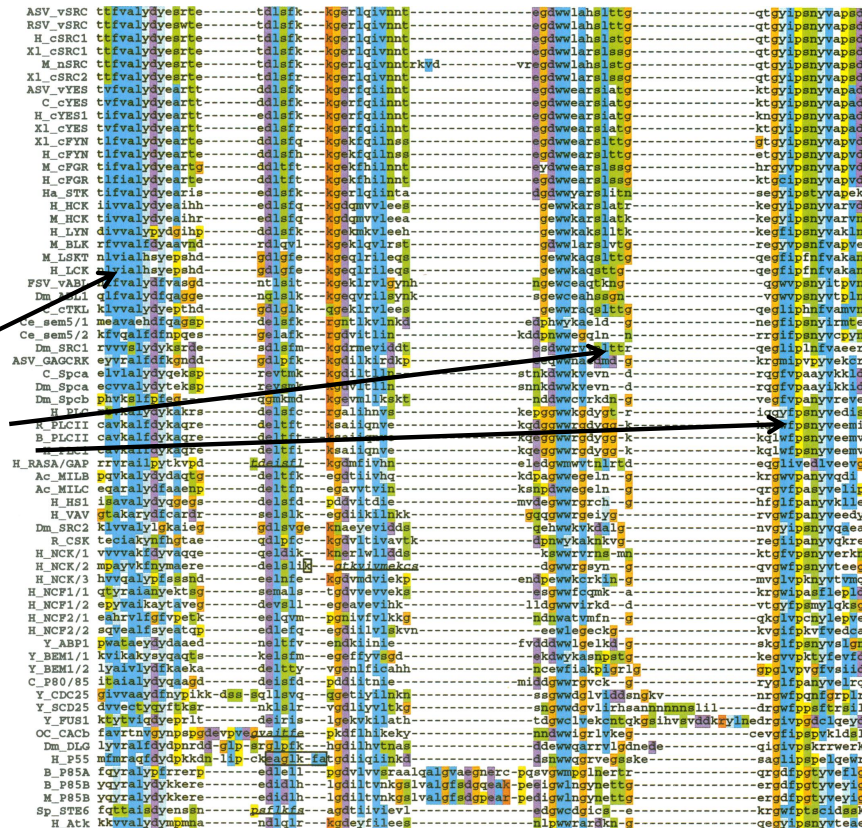
# Iterative refinement

- The order of selection of sequences can influence the alignment
- How to avoid committing to a non-optimal pairwise decision?
  - Revisit alignments
- Basic iterative refinement algorithm
  - Remove a sequence from the current multiple alignment
  - Realign the removed sequence back to the multiple alignment
  - Repeat until removal and realignment of any sequence does not improve the alignment score

# The ClustalW algorithm

- A famous progressive alignment method: the CLUSTALW algorithm [Thompson et al. 1994]
- Tailored to handle very divergent sequences: 25-30% similarity
- Dynamically varies the gap penalties in a position and residue specific manner
- Weight different sequences differently
  - Closely related sequences need to be down-weighted
  - Divergent sequences are up-weighted
- Dynamically switch between substitution matrices depending upon the average similarity between sequences being aligned

# Applying ClustalW to SH3 domain proteins



ASV_VSRC	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
RSV_VSRC	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_CSRC1	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
X1_CSRC1	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
M_NSRC	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
X1_CSRC2	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
ASV_VYSE	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
C_CYES	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_CYES1	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
X1_CYES	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
X1_CFIN	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_CFIN	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
M_CFOR	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_CFOR	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Ha_STK	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_HCK	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
M_HCK	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_LYN	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
M_LYN	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
M_LSTK	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_LCK	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
PSV_VARI	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Dm_LM1	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
C_CTKL	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Ce_sen5/1	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Ce_sen5/2	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Dm_SRC1	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
ASV_GAGCRK	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
C_SPCA	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Dm_SPCA	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Dm_SPCB	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_BPC	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
B_PLIC1	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
B_PLIC2	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_PRC1	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_PRC2	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Ac_MILB	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Ac_MILC	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_HS1	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_VAV	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Dm_SRC2	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
C_CSK	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_NCK/1	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_NCK/2	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_NCK/3	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_NCF1/1	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_NCF1/2	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_NCF2/1	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_NCF2/2	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Y_AB91	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Y_BEM1/1	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Y_BEM1/2	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
C_P80/85	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Y_CDC25	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Y_SCD25	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Y_PU81	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
OC_CACB	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Dm_DLG	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_P55	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
B_P85A	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
B_P85B	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
M_P85B	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
Sp_STK6	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad
H_Atk	tfvvalydye	tlslfk	xgerqlvnn	sgdwlaahltt	gtgypsnypapad

Alignment blocks correspond to beta strand secondary structures

Proteins share <12% sequence identity



# Summary

- Tree-based progressive alignment
  - Employs a guide tree to determine an order of pairwise alignments leading to a multiple alignment
  - Uses alignments of pairs of partial alignments
  - Is greedy in terms of fixing partial alignments lower in the tree
- Iterative refinement
  - Allows for the realignment of individual sequences within a multiple alignment
  - Can compensate for greedy multiple sequence alignment approaches
- ClustalW is a prime example of a tree-based progressive alignment method that also uses iterative refinement.