Sequence Analysis 2

A. Multiple sequence alignment

Cheong Xin Chan (CX)

c.chan1@uq.edu.au

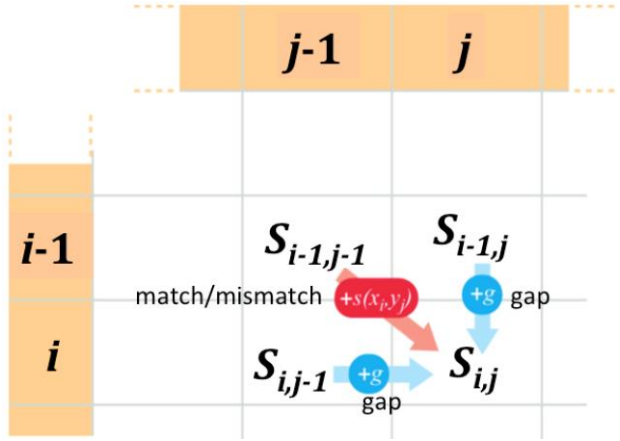
Australian Centre for Ecogenomics
School of Chemistry & Molecular Biosciences
The University of Queensland

SCIE2100 | BINF6000 | Bioinformatics 1 – Introduction

Lecture outline

- **Dynamic programming with affine gap penalty**
- **Multiple sequence alignment (MSA)**
  - Progressive MSA
    - Step-by-step example using Clustal, including tree-guided clustering using UPGMA and Neighbour-joining
    - Limitations
  - Iterative progressive MSA
  - Other MSA approaches
  - Measuring significance of an alignment
  - Sequence alignment versus structural alignment
- **MSA: issues and challenges**

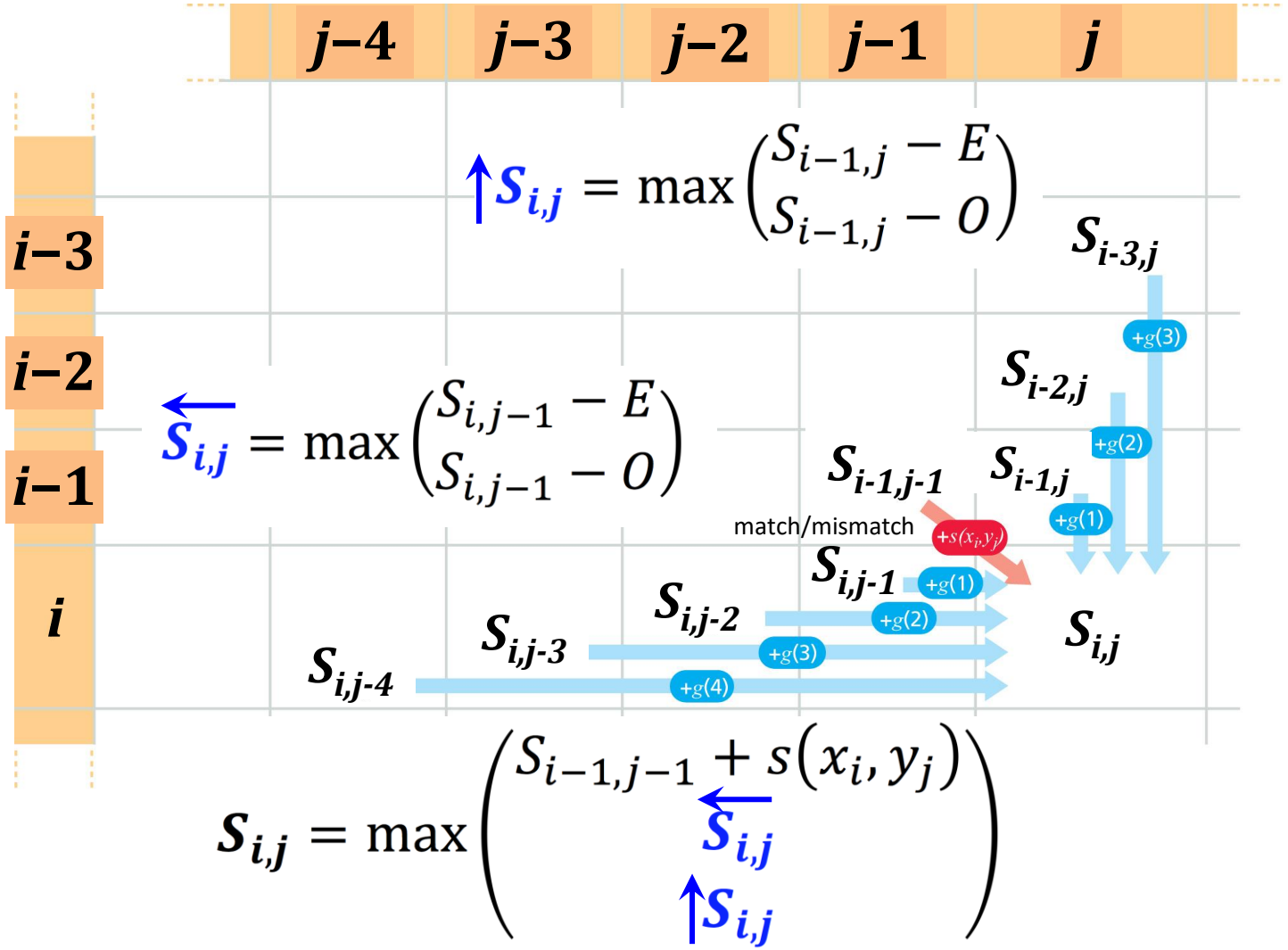
Affine gap penalty in dynamic programming



$$S_{i,j} = \max \begin{pmatrix} S_{i-1,j-1} + s(x_i, y_j) \\ S_{i-1,j} + g \\ S_{i,j-1} + g \end{pmatrix}$$

Needleman-Wunsch algorithm (*Week 3*)

In that example, all gap position is treated the same, i.e. using ***linear gap penalty***



Affine gap penalty (distinction between *gap open*, ***O*** and *gap extend*, ***E***) can be applied in a more-realistic scheme

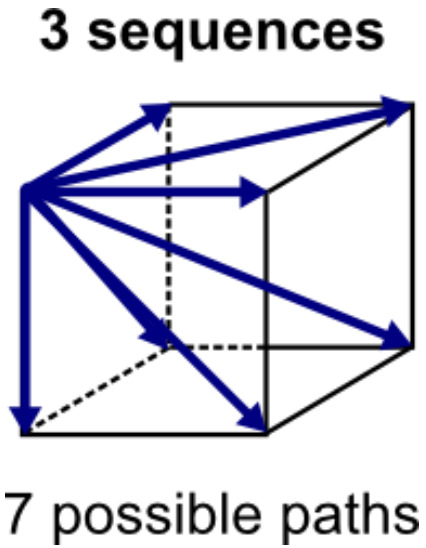
Dimensionality of scoring matrix in DP

- One dimension for each sequence in the alignment
- time and space grows **exponentially** with the number of sequences

Example: global alignment of three sequences, $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{z}$

$\boldsymbol{S_{i,j,k}}$ is the score for the **best** alignment of the initial segments of sequences $\boldsymbol{x, y, z}$ ending at position $\boldsymbol{i, j, k}$, respectively

$$\boldsymbol{S_{i,j,k}} = \max \begin{pmatrix} S_{i-1,j-1,k-1} + s(x_i, y_j) + s(x_i, z_k) + s(y_j, z_k) \\ S_{i-1,j,k} + g + g + g \\ S_{i,j-1,k} + g + g + g \\ S_{i,j,k-1} + g + g + g \\ S_{i-1,j-1,k} + s(x_i, y_j) + g + g \\ S_{i-1,j,k-1} + s(x_i, z_k) + g + g \\ S_{i,j-1,k-1} + s(y_j, z_k) + g + g \end{pmatrix}$$



Multiple sequence alignment (MSA)

"The purpose of an MSA algorithm is to assemble alignments reflecting the **biological relationship** between several sequences. Computing exact MSAs is computationally almost impossible, and in practice **approximate algorithms (heuristics)** are used to align sequences, by maximizing their similarity."

Cédric Notredame (2007) *PLoS Computational Biology* 3(8): e123.

Multiple sequence alignment (MSA)

- alignment of three or more biological sequences
- a key step for inferring phylogenetic (evolutionary) relationships among a set of sequences
- **greater** information content (at each position) than pairwise alignment can illustrate sequence **constraints** and **integrity**, e.g. common signatures or protein domains, genetic variation etc.

Pairwise alignment

p110α	TFILGIGDRHNSNIMVKDDG-QLFHIDFGHFLDHKKKKFGYKRERVPFVLT--QDFLIVI	142
cAMP-kinase	QIVLTFEYLHSLDLIYRDLKPENLLIDQQGYIQVTDFGFAKRVKGRTWXLCGTPEYLAPE	179

Example

Multiple sequence alignment

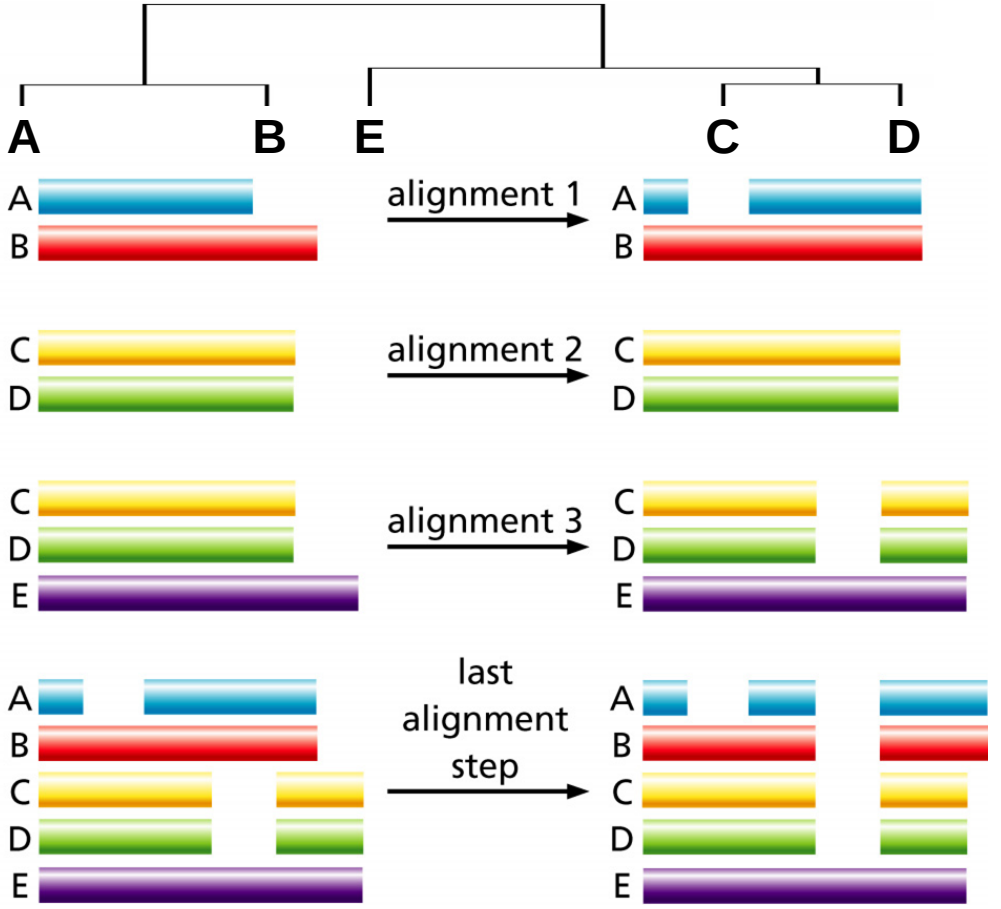
p110β	SYVLGIG----------DRHSDNINVKKTGQLFHIDFGHILGNFKSKFGIKRERVPFILT	136
p110δ	TYVLGIG----------DRHSDNIMIRESGQLFHIDFGHFLGNFKTKFGINRERVPFILT	136
p110α	TFILGIG----------DRHNSNIMVKDDGQLFHIDFGHFLDHKKKKFGYKRERVPFVLT	135
p110γ	TFVLGIG----------DRHNDNIMITETGNLFHIDFGHILGNYKSFLGINKERVPFVLT	135
p110_dicti	TYVLGIG----------DRHNDNLMVTKGGRLFHIDFGHFLGNYKKKFGFKRERAPFVFT	135
cAMP-kinase	QIVLTFEYLHSLDLIYRDLKPENLLIDQQGYIQVTDFGFAKRVKGRTWXLCG--TPEYLA	177

Progressive multiple sequence alignment

- the most widely used **heuristic** technique in MSA
- **Heuristics**: a practical method **not guaranteed** to be optimal or perfect, but sufficient for the immediate goals

Generally a three-step process:

1. Assess **pairwise sequence similarity**, e.g. build a similarity matrix
2. Build a **guide tree** based on pairwise similarity and define an order of addition of sequences to alignments (from the **most similar** sequence-pair to the **most dissimilar** pair)
3. Align sequences **progressively** based on the defined order



Align these five sequences …

```
s1: ACCGTGAAGCCAATAC
s2: ACGTGCAACCATTAC
s3: AGCGTGCAGCCAATAC
s4: AGGGTGCCGCAATAC
s5: AGGGTGCCACAATAC
```

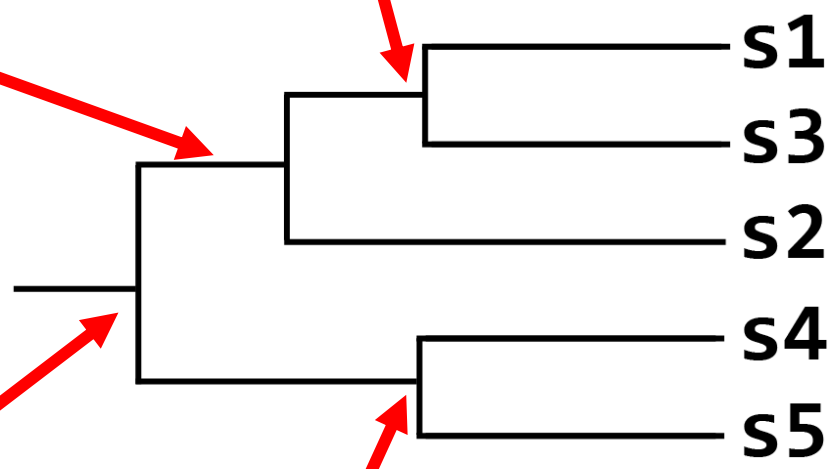
Alignment 2

```
s1: ACCGTGAAGCCAATAC
s3: AGCGTGCAGCCAATAC
```

Alignment 3

```
s1: ACCGTGAAGCCAATAC
s3: AGCGTGCAGCCAATAC

s2: A-CGTGCAACCATTAC
```



Alignment 4

```
s1: ACCGTGAAGCCAATAC
s3: AGCGTGCAGCCAATAC

s2: A-CGTGCAACCATTAC

s4: AGGGTGCCGC-AATAC
s5: AGGGTGCCAC-AATAC
```

Alignment 1

```
s4: AGGGTGCCGCAATAC
s5: AGGGTGCCACAATAC
```

Clustal: a progressive alignment approach

- a series of MSA tools based on the **progressive alignment**
- Clustal (Higgins & Sharp 1988), ClustalV (Higgins et al. 1992)
- **ClustalW** (Thompson *et al.* 1994) – improvement through sequence weighting, position-specific gap penalties and weight matrix choice
- **Clustal Omega** (Sievers *et al.* 2011) – more scalable

Three basic steps:

1. Assess **pairwise sequence similarity** using scores from all possible pairwise alignments
2. Establish an hierarchical order using a **guide tree** based on **UPGMA** or **Neighbour-joining (NJ)**
3. Align sequences **progressively** based on the defined order

Step 1: pairwise alignment

- given a set of sequences, pairwise alignment is performed on all possible pairs
- pairwise distance for each pairwise alignment is then determined
- $\boldsymbol{n}$ number of sequences gives $\boldsymbol{n(n\text{-}1)/2}$ pairwise alignments, i.e. "$\boldsymbol{n}$ choose $\boldsymbol{2}$": $\boldsymbol{C(n,2)}$ or ${}^{n}\boldsymbol{C}_{2}$

Example

```
Cattle   STCVLSAYWKDLNNYH
Human    STCMLGTYQDFNKFH
Pig      STCVLSAYWRNELNNFH
Rat      STCMLGTYQDLNKFH
Salmon   STCVLGKLSQELHKLQ
Sheep    STCVLSAYWKDLNNYH
```

Step 1: pairwise alignment

```
Cattle   STCVLSAYWKDLNNYH
Human    STCMLGTYQDFNKFH
Pig      STCVLSAYWRNELNNFH
Rat      STCMLGTYQDLNKFH
Salmon   STCVLGKLSQELHKLQ
Sheep    STCVLSAYWKDLNNYH
```

Example

```
Sheep    STCVLSAYWKDLNNYH
Cattle   STCVLSAYWKDLNNYH

Sheep    STCVLSAYWK-DLNNYH
Pig      STCVLSAYWRNELNNFH

Sheep    STCVLSAYWKDLNNYH
Human    STCMLGTY-QDFNKFH

Sheep    STCVLSAYWKDLNNYH
Rat      STCMLGTY-QDLNKFH

Sheep    STCVLSAYWKD-LNNYH
Salmon   STCVLGKL-SQELHKLQ

Pig      STCVLSAYWRNELNNFH
Human    STCMLGTY-QD-FNKFH

Pig      STCVLSAYWRNELNNFH
Rat      STCMLGTY-QD-LNKFH

Pig      STCVLSAYWRNELNNFH
Salmon   STCVLGKL-SQELHKLQ

Human    STCMLGTYQDFNKFH
Rat      STCMLGTYQDLNKFH

Human    STCMLGTY-QDFNKFH
Salmon   STCVLGKLSQELHKLQ

Rat      STCMLGTY-QDLNKFH
Salmon   STCVLGKLSQELHKLQ

etc.
```

Step 2: establish hierarchical order

Calculate **pairwise distance** (e.g. number of differing aligned positions)

Example

```
Cattle   STCVLSAYWKDLNNYH
Human    STCMLGTYQDFNKFH
Pig      STCVLSAYWRNELNNFH
Rat      STCMLGTYQDLNKFH
Salmon   STCVLGKLSQELHKLQ
Sheep    STCVLSAYWKDLNNYH
```

0
```
Sheep    STCVLSAYWKDLNNYH
Cattle   STCVLSAYWKDLNNYH
```

4
```
Sheep    STCVLSAYWK-DLNNYH
Pig      STCVLSAYWRNELNNFH
```

8
```
Sheep    STCVLSAYWKDLNNYH
Human    STCMLGTY-QDFNKFH
```

7
```
Sheep    STCVLSAYWKDLNNYH
Rat      STCMLGTY-QDLNKFH
```

11
```
Sheep    STCVLSAYWKD-LNNYH
Salmon   STCVLGKL-SQELHKLQ
```

9
```
Pig      STCVLSAYWRNELNNFH
Human    STCMLGTY-QD-FNKFH
```

8
```
Pig      STCVLSAYWRNELNNFH
Rat      STCMLGTY-QD-LNKFH
```

10
```
Pig      STCVLSAYWRNELNNFH
Salmon   STCVLGKL-SQELHKLQ
```

1
```
Human    STCMLGTYQDFNKFH
Rat      STCMLGTYQDLNKFH
```

9
```
Human    STCMLGTY-QDFNKFH
Salmon   STCVLGKLSQELHKLQ
```

8
```
Rat      STCMLGTY-QDLNKFH
Salmon   STCVLGKLSQELHKLQ
```

etc.

…

Step 2: establish hierarchical order

Example

	Sheep	Cattle	Pig	Human	Rat	Salmon
Sheep	0	0	4	8	7	11
Cattle	**0**	0	4	8	7	11
Pig	**4**	**4**	0	9	8	10
Human	**8**	**8**	**9**	0	1	9
Rat	**7**	**7**	**8**	**1**	0	8
Salmon	**11**	**11**	**10**	**9**	**8**	0

- the most similar sequences should be aligned first, as these are the easiest, introducing the fewest mistakes (i.e. the **greedy principle**)
- we may need to create several intermediate alignments that will later be joined

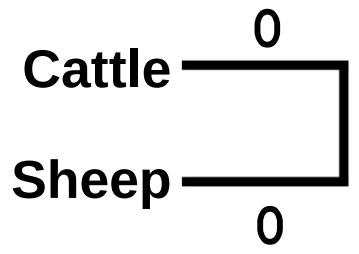
Step 2: establish hierarchical order using UPGMA

Unweighted **P**air **G**roup **M**ethod with **A**rithmetic mean

- agglomerative (“bottom up”) hierarchical clustering method
- at each step, the nearest two elements/clusters are combined (merged) into a higher-level cluster
- assumes **ultrametricity** (i.e. same root-to-tip distance for every branch tip)
- distance between clusters A & B = average distance between all element-pairs in A and in B

Example

	Sheep	Cattle	Pig	Human	Rat	Salmon
Sheep	0	0	4	8	7	11
Cattle	**0**	0	4	8	7	11
Pig	**4**	**4**	0	9	8	10
Human	**8**	**8**	**9**	0	1	9
Rat	**7**	**7**	**8**	**1**	0	8
Salmon	**11**	**11**	**10**	**9**	**8**	0

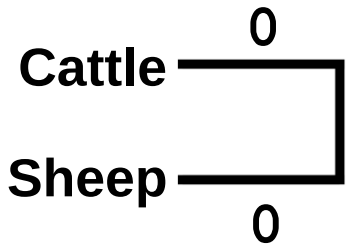


branch lengths not to scale

	Sheep	Cattle	Pig	Human	Rat	Salmon
Sheep	0					
Cattle	**0**	0				
Pig	**4**	**4**	0			
Human	**8**	**8**	**9**	0		
Rat	**7**	**7**	**8**	**1**	0	
Salmon	**11**	**11**	**10**	**9**	**8**	0

	Sheep+ Cattle	Pig	Human	Rat	Salmon
Sheep+ Cattle	0				
Pig	**4**	0			
Human	**8**	**9**	0		
Rat	**7**	**8**	**1**	0	
Salmon	**11**	**10**	**9**	**8**	0

Sheep and Cattle have the shortest distance (**0**), so they are merged first

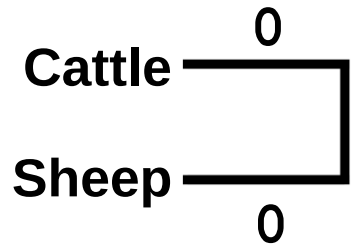


branch lengths not to scale

	Sheep+ Cattle	Pig	Human	Rat	Salmon
Sheep+ Cattle	0				
Pig	**4**	0			
Human	**8**	**9**	0		
Rat	**7**	**8**	**1**	0	
Salmon	**11**	**10**	**9**	**8**	0

	Sheep+Cattle	Pig	Human+Rat	Salmon
Sheep+Cattle	0			
Pig	**4**	0		
Human+Rat	**7.5**	**8.5**	0	
Salmon	**11**	**10**	**8.5**	0

Next, Human and Rat have the shortest distance (**1**), so they are merged



branch lengths not to scale

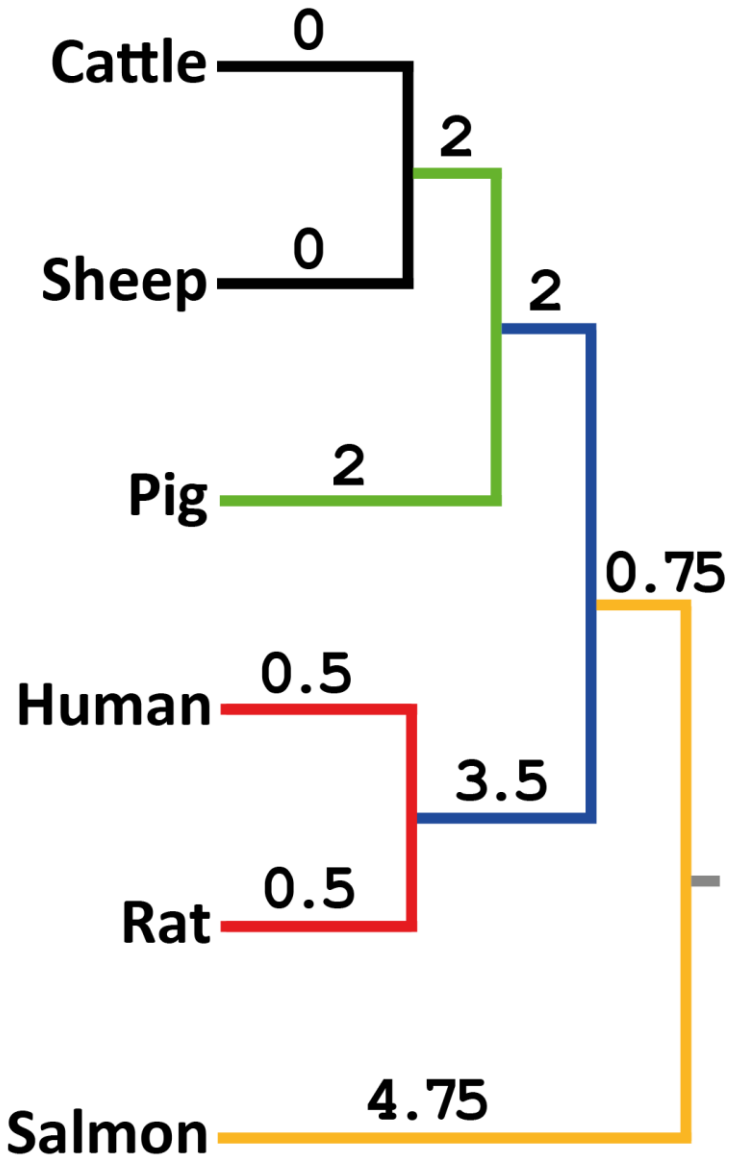
	Sheep+Cattle	Pig	Human+Rat	Salmon
Sheep+Cattle	0			
Pig	4	0		
Human+Rat	7.5	8.5	0	
Salmon	11	10	8.5	0

	Sheep+Cattle+Pig	Human+Rat	Salmon
Sheep+Cattle+Pig	0		
Human+Rat	8	0	
Salmon	10.5	8.5	0

	Sheep+Cattle+Pig+Human+Rat	Salmon
Sheep+Cattle+ Pig+Human+Rat	0	
Salmon	9.5	0

Example

and so forth ...

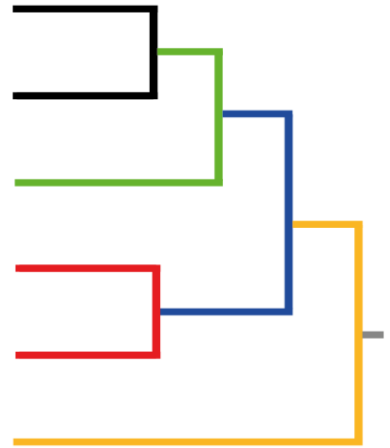


branch lengths not to scale

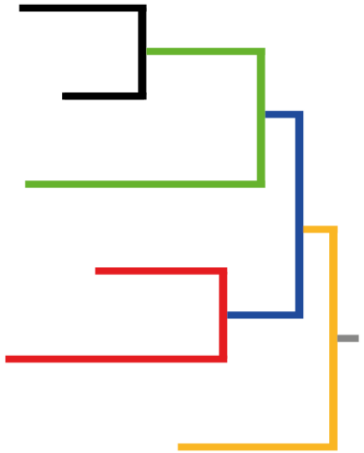
Step 2: establish hierarchical order

using Neighbour-Joining (NJ)

- proceeds in similar way as UPGMA, but based on a different distance matrix
- NJ does not assumes **ultrametricity**
- NJ is considerably more-robust to deviation from ultrametricity than UPGMA
- Progressively adding structure to the tree by joining the pair of clusters separated by the **shortest mean distance**
- default in ClustalW; slower than UPGMA



UPGMA tree



NJ tree

Step 3: progressive alignment

- pairwise alignment of alignments (**profile alignment**)
- dynamic programming can be applied:

$$S_{i,j} = \max \begin{pmatrix} S_{i-1,j-1} + m(x_i, y_j) \\ S_{i-1,j} + g \\ S_{i,j-1} + g \end{pmatrix}$$

where $m(x_i, y_j)$ is the similarity score averaged over characters at that position, and that x_i and y_j each is a set of aligned residues from one or more sequences

Step 3: progressive alignment

Alignment 1

```
Cattle   STCVLSAYWKDLNNYH
Sheep    STCVLSAYWKDLNNYH
```

Alignment 2

```
Human    STCMLGTYQDFNKFH
Rat      STCMLGTYQDLNKFH
```

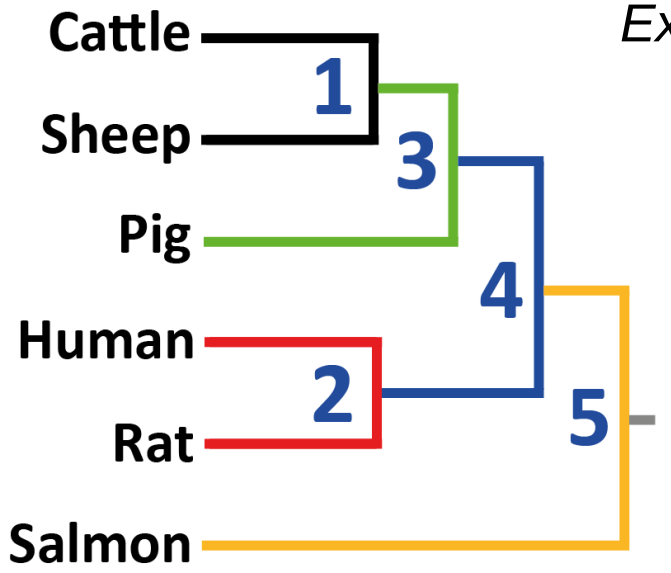
Alignment 3

```
Cattle   STCVLSAYWK-DLNNYH
Sheep    STCVLSAYWK-DLNNYH
Pig      STCVLSAYWRNELNNFH
```

Alignment 4

```
Cattle   STCVLSAYWK-DLNNYH
Sheep    STCVLSAYWK-DLNNYH
Pig      STCVLSAYWRNELNNFH
Human    STCMLGTYQD--FNKFH
Rat      STCMLGTYQD--LNKFH
```

Example



Alignment 5

```
Sheep    STCVLSAYWK-DLNNYH
Cattle   STCVLSAYWK-DLNNYH
Pig      STCVLSAYWRNELNNFH
Human    STCMLGTYQD--FNKFH
Rat      STCMLGTYQD--LNKFH
Salmon   STCVLGKLSQ-ELHKLQ
         ****:*.      :::  :
```

Final alignment – might not be the best/optimal solution

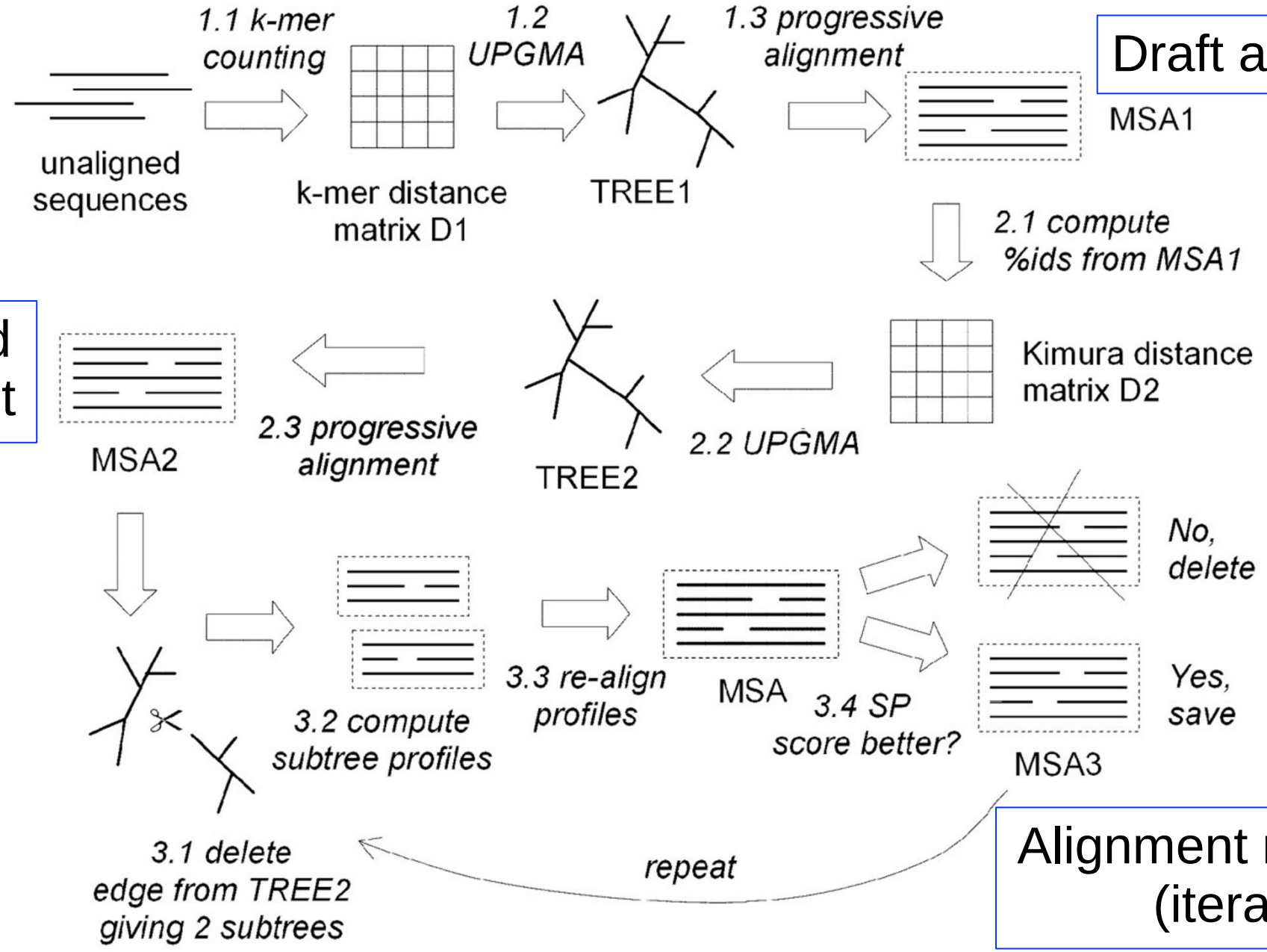
Progressive MSA: limitations

- tree might be **incorrect**, thus causing incorrect ordering of how sequences should be stacked up in the alignment
- once sequences are aligned and gaps introduced, these are **not altered**
- these early errors will be **propagated** and reflected in the final alignment, e.g. ClustalW finds a local optimum when early alignment decisions are “locked in” by the **“greedy”** algorithm
- final results **prone to errors** in alignment - some positions might be misaligned, i.e. the alignment could have a lower score than another alignment if a different ordering were used

Iterative progressive MSA

- aims to reduce the errors inherent in progressive methods
- works similarly to progressive methods (i.e., they are **iterative progressive** methods)
- repeatedly **realigns** the initial sequences as well as **adding new sequences** to the growing MSA
- can return to previously calculated pairwise alignments, or sub-alignments (subset of an alignment) incorporating the query sequence, in attempts to **optimise/refine** the overall MSA (to yield MSA with a higher score)
- MUSCLE is the most popular program: refines each tree branch independently using a draft tree and a refined tree
- other programs include DIALIGN, PRRN/PRRP

Iterative MSA: MUSCLE overview



Edgar RC (2004)
Nucl. Acids Res.
32: 1792-1797.

Other MSA approaches

Consensus methods

- attempt to find the optimal MSA given multiple different MSAs of the same set of sequences (i.e. a library of MSAs) based on **consensus**, e.g. T-COFFEE
- could adopt a meta-method approach, making use of the different MSA programs e.g. M-COFFEE

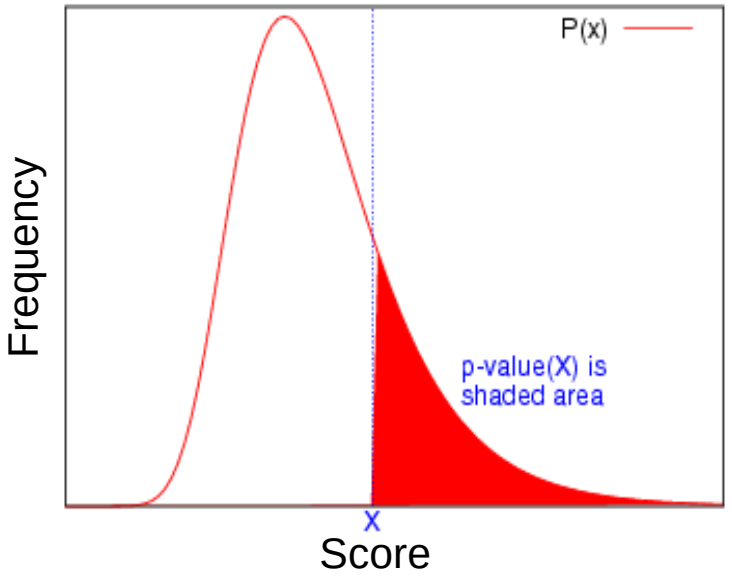
Methods based on **Hidden Markov models** (HMMs)

- uses **probabilistic models** to assign likelihoods to all possible combinations of gaps, matches, and mismatches to determine the most likely MSA or set of possible MSAs

Others e.g. **machine-learning** methods (genetic algorithms, simulated annealing) and **phylogeny-aware** methods are available; they are more computationally expensive (and less commonly used)

Measuring significance of an alignment

- statistical significance of an alignment score is used to assess whether an alignment is a result of **homology** or simply **random chance** (i.e. the biological relevance of the alignment)
- The ***p*-value** of an alignment score is the probability that a random alignment would have a an equal or higher score
- Of particular importance in database searching



Modelling score distribution

For ungapped local alignments, the distribution can be computed analytically

For gapped alignments, it must be estimated empirically

Sequence alignment *vs* structural alignment

A. Structural alignment from BAliBase (an alignment benchmark database)

```
1csy  SHEKMPWFHGKISREESEQIVLIGSKTNGKFLIRARD--NNGSYALCLLHEGKVLHYRIDKDKTGKLSIPEGK-KFDTLWQLVEHYSYKA------DGLLRVL-TVPCQK
1gri  EMKPHPWFFGKIPRAKAEEML-SKQRHDGAFLIRESES-APGDFSLSVKFGNDVQHFKVLRDGAGKYFL-WVV-KFNSLNELVDYHRSTS-VSRNQQIFLRDIEQVPQQ-
1aya  ---MRRWFHPNITGVEAENLLLTRG-VDGSFLARPSKS-NPGDFTLSVRRNGAVTHIKIQN--TGDYYDLYGGEKFATLAELVQYYMEHHGQLKEKNGDVIEL-KYPLN-
2pna  -LQDAEWYWGDISREEVNEKLRDT--ADGTFLVRDASTKMHGDYTLTLRKGGNNKLIKIFH-RDGKYGFSDPL-TFNSVVELINHYRNES-LAQYNPKLDVKL-LYPVS-
1bfi  HHDEKTWNVGSSNRNKAENLLRGK--RDGTFLVRESS--KQGCYACSVVVDGEVKHCVINKTATG-YGFAEPYNLYSSLKELVLHYQHTS-LVQHNDSLNVTL-AYPVYA
```

B. Multiple sequence alignment using DIALIGN (iterative method)

```
1csy  SHEKMPWFHGKISREESEQIVLIGSKT-NGKFLIRAR-DN--NGSYALCLLHEGKVLHYRIDKDKTGKLSIPEGKK-FDTLWQLVEHYSYKA-------DGLLRVLT-VPCQK
1gri  EMKPHPWFFGKIPRAKAEEML--SKQRHDGAFLIRESESA--PGDFSLSVKFGNDVQHFKVLRDGAGKYFLWVV-K-FNSLNELVDYHRST--SVSRNQQIFLRDIEQVPQQ-
1aya  M---RRWFHPNITGVEAENLLLTRGV--DGSFLARPSKSN--PGDFTLSVRRNGAVTHIKIQNTGDYYDLYG-GEK-FATLAELVQYYMEHHGQLKEKNGDV-IELK-YPLN-
2pna  LQDAE-WYWGDISREEVNEKL--RDTA-DGTFLVRDA-STKMHGDYTLTLRKGGNNKLIKIFHRDGKYGFSD-PLT-FNSVVELINHYRNE--SLAQYNPKLDVKLL-YPVS-
1bfi  HHDEKTWNVGSSNRNKAENLL--RGKR-DGTFLVRES-SK--QGCYACSVVVDGEVKHCVINKTATGYGFAE-PYNLYSSLKELVLHYQHT--SLVQHNDSLNVTLA-YPVYA
```

pink/red: alpha-helices
yellow: beta-sheets

C. Multiple sequence alignment using ClustalW (progressive method)

```
1csy  SHEKMPWFHGKISREESEQIVLIGSKTNGKFLIRARDN--NGSYALCLLHEGKVLHYRIDKDKTGKLSIPEGKKFD-TLWQLVEHYSYK------ADGLLRVLTVPCQK
1gri  EMKPHPWFFGKIPRAKAEE-MLSKQRHDGAFLIRESES-APGDFSLSVKFGNDVQHFKVLRDGAGKY-FLWVVKFN-SLNELVDYHRSTS-VSRNQQIFLRDIEQVPQQ
1aya  ---MRRWFHPNITGVEAEN-LLLTRGVDGSFLARPSKS-NPGDFTLSVRRNGAVTHIKIQNT-GDYYDLYGGEKFA-TLAELVQYYMEHHGQLKEKNGDVIELKYPLN-
2pna  -LQDAEWYWGDISREEVN--EKLRDTADGTFLVRDASTKMHGDYTLTLRKGGNNKLIKIFHR-DGKYGFSDPLTFN-SVVELINHYRNES-LAQYNPKLDVKLLYPVS-
1bfi  HHDEKTWNVGSSNRNKAE--NLLRGKRDGTFLVRESSK--QGCYACSVVVDGEVKHCVINKT-ATGYGFAEPYNLYSSLKELVLHYQHTS-LVQHNDSLNVTLAYPVYA
```

Which one of these alignments is the best?

"All the existing validation approaches have in common their reliance on the "one size fits all" **assumption** that structurally correct alignments are the best possible MSAs for modeling any kind of **biological signal** (**evolution**, **homology**, or **function**). … it may be reasonable to ask *whether better alignments always result in better phylogenetic trees*, and, more systematically, to *question and quantify the relationship between the accuracy of MSAs and the biological relevance of any model drawn upon them*."

Cédric Notredame (2007) *PLoS Computational Biology* 3(8): e123.

MSA: issues and challenges

```
Seq1 ATTTAAACGTCTAGATTTAA----------TAGCATGCGA
Seq2 ----------CTAGATTTAAATTTAAACGTTAGCATGCGA
```

- based on strict assumption of **whole-sequence contiguity**; largely heuristics (for feasibility)
- relevance of alignment scores to homology can be difficult to assess statistically
- loss of phylogenetic information in instances of genome rearrangements, lateral genetic transfer etc.

```
Seq1   ATTTAAACGT   CTAGATTTAA      TAGCATGCGA
Seq2   ATTTAAACGT   CTAGATTTAA      TAGCATGCGA
```

- ***Alignment-free (AF)* methods:** distances based on sub sequences of defined length (e.g., *k*-mers) – no assumption of whole-sequence contiguity
- AF methods are more scalable: on-going active field of research

Reflection

- *Is dynamic programming scalable for aligning multiple sequences?*
- *What information can we observe from an MSA? What do we commonly use MSA for?*
- *Why are heuristic approaches used for MSA?*
- *What is progressive MSA, and what are the key steps involved?*
- *What are the two common methods adopted to establish hierarchical order in progressive MSA?*
- *What is the main difference between UPGMA and NJ?*
- *What are the limitations of progressive MSA, and how can we improve it?*
- *What are some limitations of MSA, and how can we attempt to resolve these?*