Phylogeny: traditional and Bayesian approaches

5-Feb-2014

DEKM book

Notes from Dr. B. John

Holder and Lewis, Nature Reviews Genetics 4, 275-284, 2003

Phylogeny

- A graph depicting the ancestor-descendent relationships between organisms or gene sequences.
- Sequences are tips of the tree
- Branches connect the tips to their unobservable ancestral sequence

Box 1 | **Applications of phylogenetic methods**

Detection of orthology and paralogy

Phylogenetics is commonly used to sort out the history of gene duplications for gene families. This application is now included in even preliminary examinations of sequence data; for example, the initial analysis of the mouse genome[46] included neighbour-joining trees to identify duplications in cytochrome P450 and other gene families.

Estimating divergence times

Bayesian implementations of new models[37,38] allowed Aris-Brosou and Yang[40] to estimate when animal phyla diverged without assuming a molecular clock.

Reconstructing ancient proteins

Chang *et al.*[47] used maximum likelihood (ML) to reconstruct the sequence of visual pigments in the last common ancestor of birds and alligators; the protein was then synthesized in the laboratory (see REF. 48 for a recent discussion of the methodology of ancestral-character-state reconstruction).

Finding the residues that are important to natural selection

Amino-acid sites on the surface of influenza that are targeted by the immune system can be detected by an excess of non-synonymous substitutions[49–51]. This information might assist vaccine preparation.

Detecting recombination points

New Bayesian methods[52] can help determine which strains of human immunodeficiency virus-1 (HIV-1) arose from recombination.

Identifying mutations likely to be associated with disease

The lack of structural, biochemical and functional data from many genes implicated in disease means it is unclear which missense mutations are important. Fleming *et al.*[53] used Bayesian phylogenetics to identify missense mutations in conserved regions and regions under positive selection in the breast cancer gene *BRCA1*. These data allowed them to prioritize these mutations for future functional and population studies.

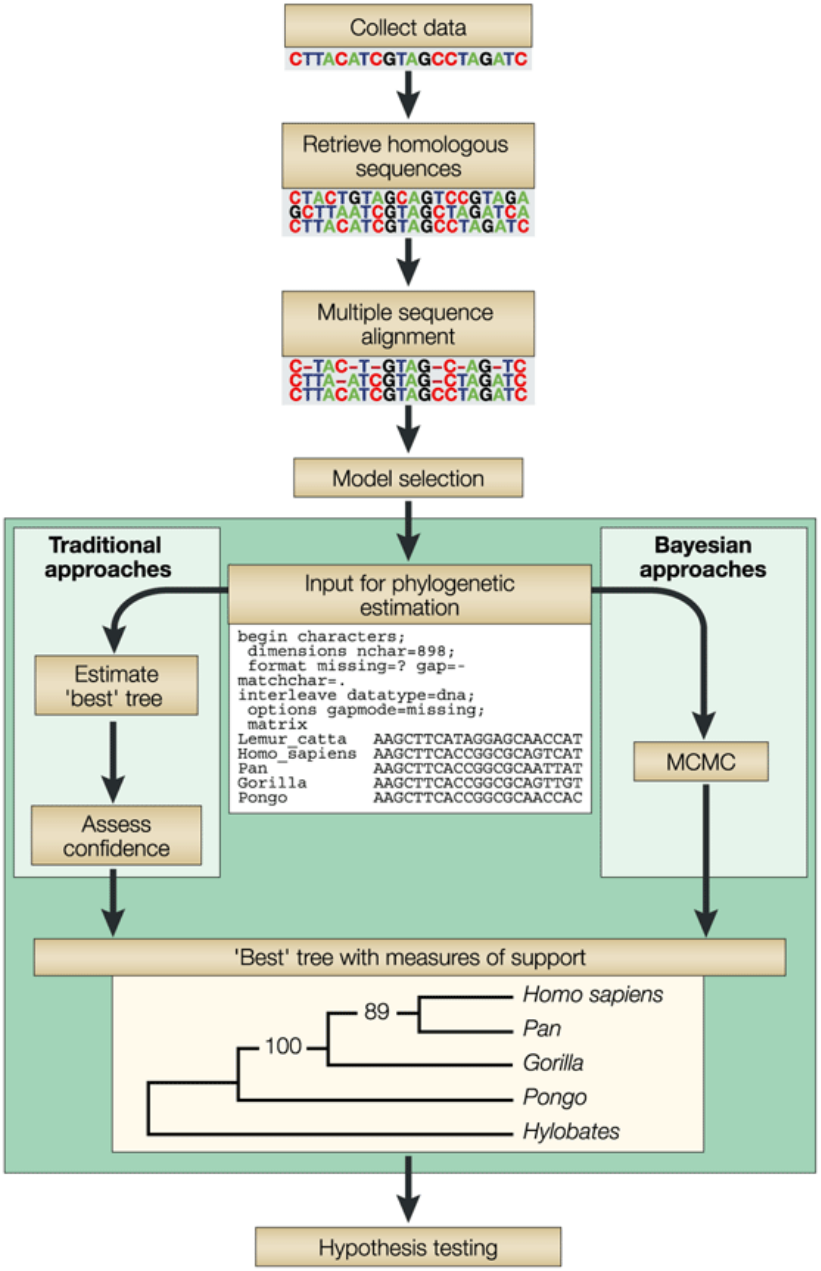
Determining the identity of new pathogens

Phylogenetic analysis is now routinely performed after polymerase chain reaction (PCR)

Table 1 | **Comparison of methods**

Method	Advantages	Disadvantages	Software
Neighbour joining	Fast	Information is lost in compressing sequences into distances; reliable estimates of pairwise distances can be hard to obtain for divergent sequences	PAUP* MEGA PHYLIP
Parsimony	Fast enough for the analysis of hundreds of sequences; robust if branches are short (closely related sequences or dense sampling)	Can perform poorly if there is substantial variation in branch lengths	PAUP* NONA MEGA PHYLIP
Minimum evolution	Uses models to correct for unseen changes	Distance corrections can break down when distances are large	PAUP* MEGA PHYLIP
Maximum likelihood	The likelihood fully captures what the data tell us about the phylogeny under a given model	Can be prohibitively slow (depending on the thoroughness of the search and access to computational resources)	PAUP* PAML PHYLIP
Bayesian	Has a strong connection to the maximum likelihood method; might be a faster way to assess support for treesthan maximum likelihood bootstrapping	The prior distributions for parameters must be specified; it can be difficult to determine whether the Markov chain Monte Carlo (MCMC) approximation has run for long enough	MrBayes BAMBE

For a more complete list of software implementations, see online link to Phylogeny Programs. For software URLs, see online links box.



Nature Reviews | Genetics

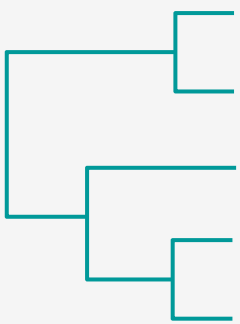
Table 2 | **Tree construction and tree searching methods**

Method	Description	Advantages	Disadvantages
Tree construction methods			
Stepwise addition	Builds a complete tree, starting with three sequences and attaching new sequences one at a time to the branch that yields the optimum tree at each step	Fast; later steps can reverse earlier pairing decisions	Yields one tree, often not global optimum; alternative additional sequences might yield different trees; not as fast as neighbour-joining
Star decomposition	Builds a completely resolved tree, starting with all sequences connected to a single 'hub' node. At each step, two lineages attached to the hub node are joined, becoming neighbours. Neighbours are chosen so that tree is optimal at each step	Fast; addition sequence irrelevant	Yields one tree, often not global optimum; neighbours cannot be dismantled at later steps; ties broken arbitrarily by some implementations
Neighbour joining	A star-decomposition method that uses an approximation to the minimum-evolution optimality criterion	One of the fastest of all tree construction methods	The same as those listed for star decomposition
Tree searching methods			
Heuristic search	Given a starting tree containing all sequences of interest, performs branch swapping to generate alternative trees in an attempt to find a better tree under a given optimality criterion. Strict hill-climber: if a better tree is found, the process begins again, stopping only if a local optimum is attained. Typically uses a stepwise addition or neighbour-joining tree as the starting tree	Faster than exact searches	Can miss the global optimal tree
Exact search	Exhaustive searches examine every possible tree and are guaranteed to return the best tree. Branch-and-bound techniques can eliminate some bad trees from consideration and still guarantee that they will return the best tree	The only methods that are guaranteed to find the best trees	Time-consuming: only practical for a few sequences (<20)

What is phylogeny?

- The inference of evolutionary relationships
- The inference of putative common ancestors
- Trees (with branches and leaves!)

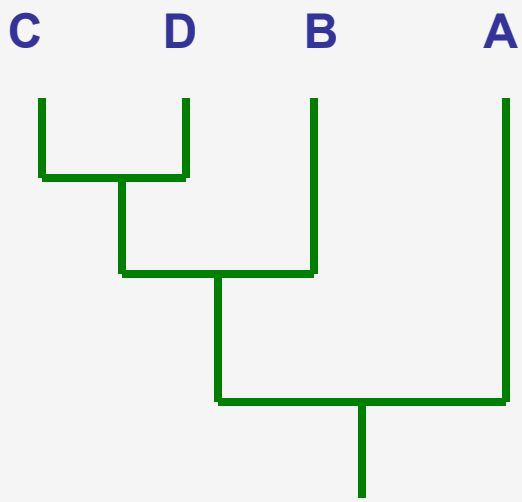
Example:



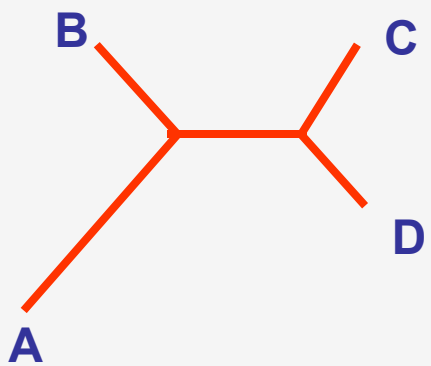
mouse:	ACAGTGACGCCCCAAACGT
rat:	ACAGTGACGCTACAAACGT
marmoset	CCTGTGACGTAACAAACGA
Chimp:	CCTGTGACGTAGCAAACGA
Human:	CCTGTGACGTAGCAAACGA

Rooted vs Unrooted tree (dendograms)

Rooted tree

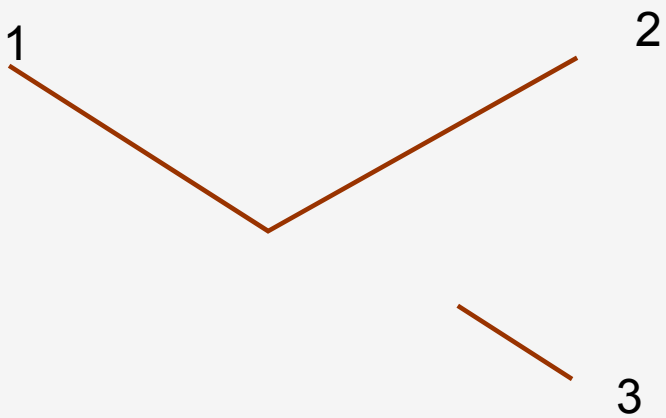


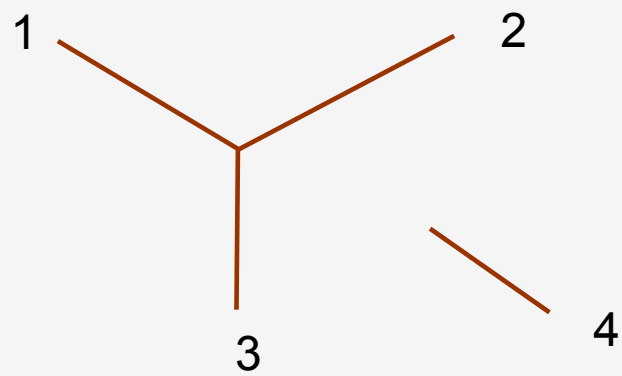
Unrooted tree



OTUs – Operational taxonomic units (leaves)
HTUs – Hypothetical taxonomic units (internal nodes)

Calculating the number of unrooted trees





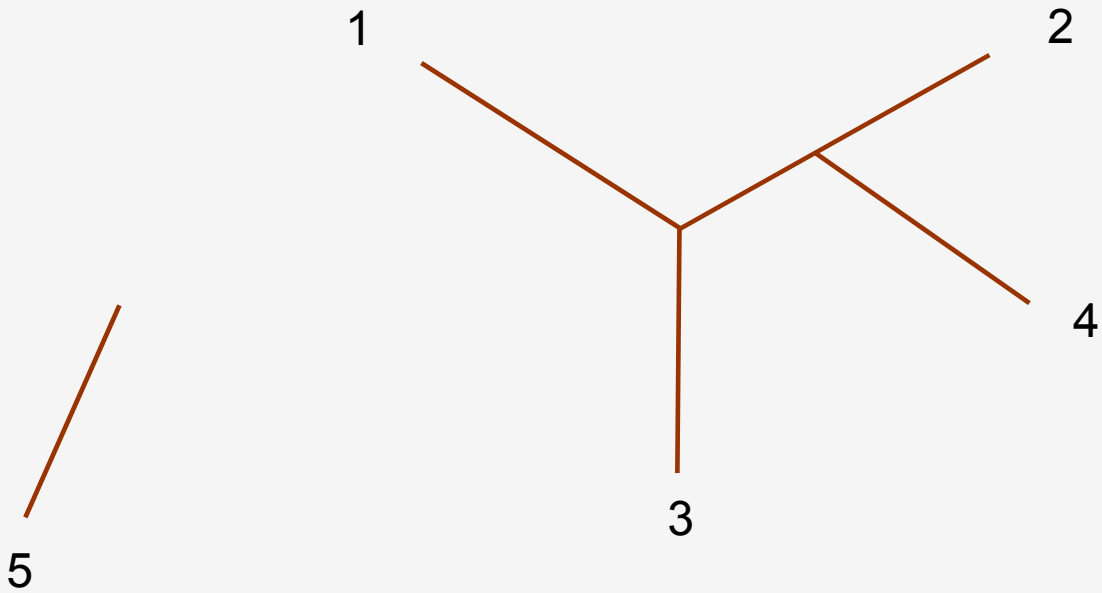
How many possibilities are there for adding leaf 3?

- 1

How many possibilities are there for adding leaf 4?

- 3

Calculating the number of unrooted trees

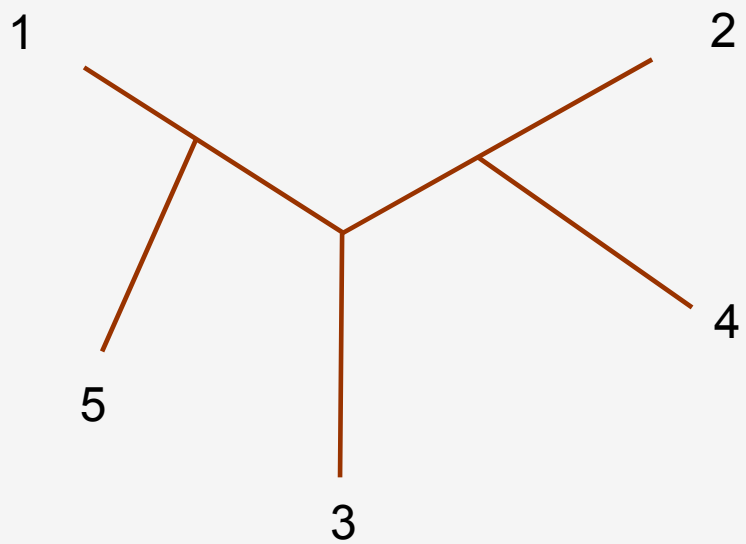


- How many possibilities are there for leaf 5?

  For the 5th leaf, there are 5 possibilities
- *ie* to add the nth leaf, we can add at the immediate leaf branches+ the number of internal branches

ie (n-1) “terminal-branches” + (n-4) “internal branches”=2n-5

Total number of trees?



N = 10

#unrooted: 2,027,025
#rooted: 34,459,425

N = 30
#unrooted: 8.7×10^{36}
#rooted: 4.95×10^{38}

- #unrooted trees for *n* taxa: $(2n-5)*(2n-7)*...*3*1 = (2n-5)! / [2^{n-3}*(n-3)!]$
- #rooted trees for *n* taxa: $(2n-3)*(2n-5)*(2n-7)*...*3 = (2n-3)! / [2^{n-2}*(n-2)!]$

Commonly represented as 2n-5!! or 2n-3!!

Two main problems in Tree Construction

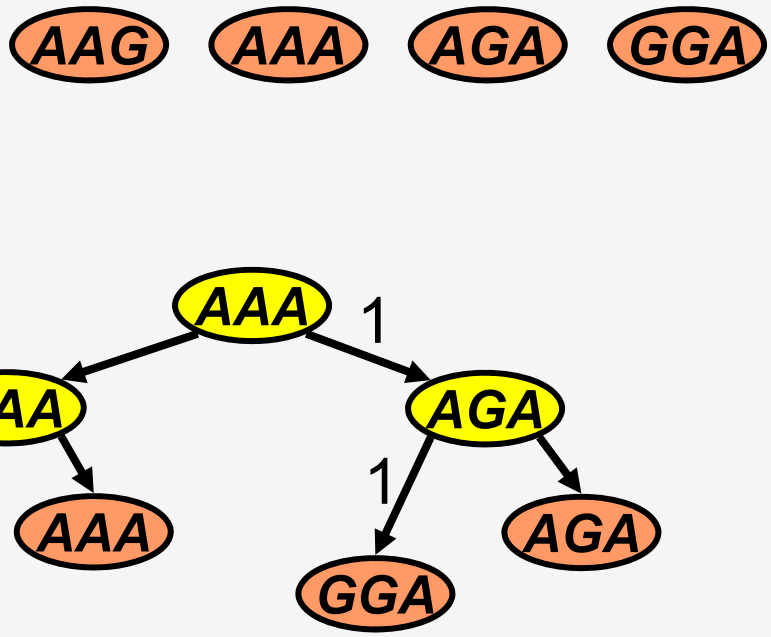
- Evaluate tree and/or assign branch lengths
- Construction of the best tree among the many possibilities

Parsimony Approach to Evolutionary Tree Evaluation

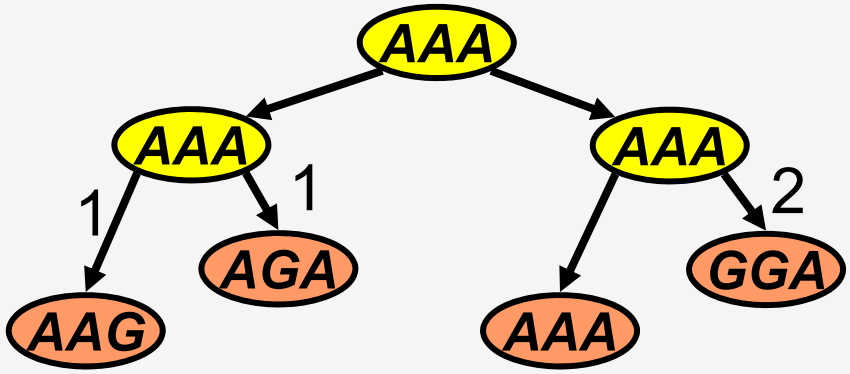
Parsimony ≡ frugality, " less is better"

- Assumes observed character differences resulted from the fewest possible mutations
- Seeks the tree that yields lowest possible **parsimony score -** sum of cost of all mutations found in the tree

Example for calculating parsimony score



Total #substitutions = 3



Total #substitutions = 4

The left tree is preferred over the right tree.

The total number of changes is called the **parsimony score**.

Sankoff algorithm to count evolutionary changes in a given tree

Step-1: Define a cost matrix $[c_{ij}]$, representing changes from character state *i* to state *j*

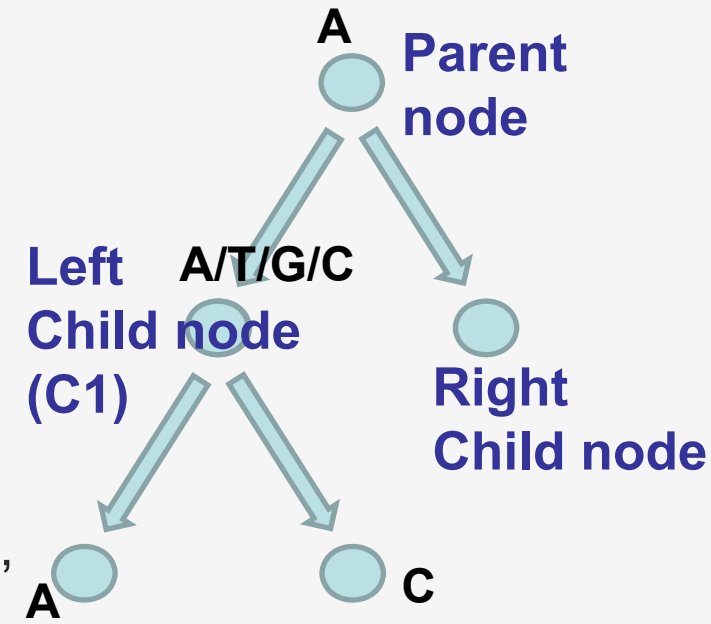
	A	C	G	T
A	0	2.5	1	2.5
C	2.5	0	2.5	1
G	1	2.5	0	2.5
T	2.5	1	2.5	0

Step-2: Starting from the leaves, we work down at each node, k to calculate a score $S_k(i)$ for each state, *i (i= 1.. 4 for DNA)*

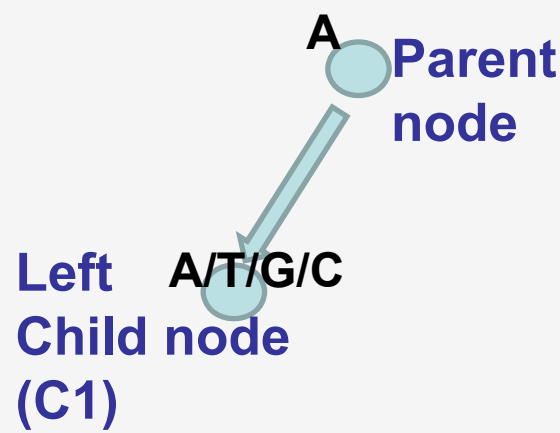
- *S_k is the minimal cost of events for all subtrees above that node, k*

Sankoff algo – cont'd

- Remember for each internal node ("parent") we have two sub-nodes ("children")
- So the scores need to be added up on both branches
- First, consider say the left branch, with left child node at state *i* and we want to evaluate the score for parent node at state A.
  - If S(left child node) is known for each states, the S(parent node@j) will be



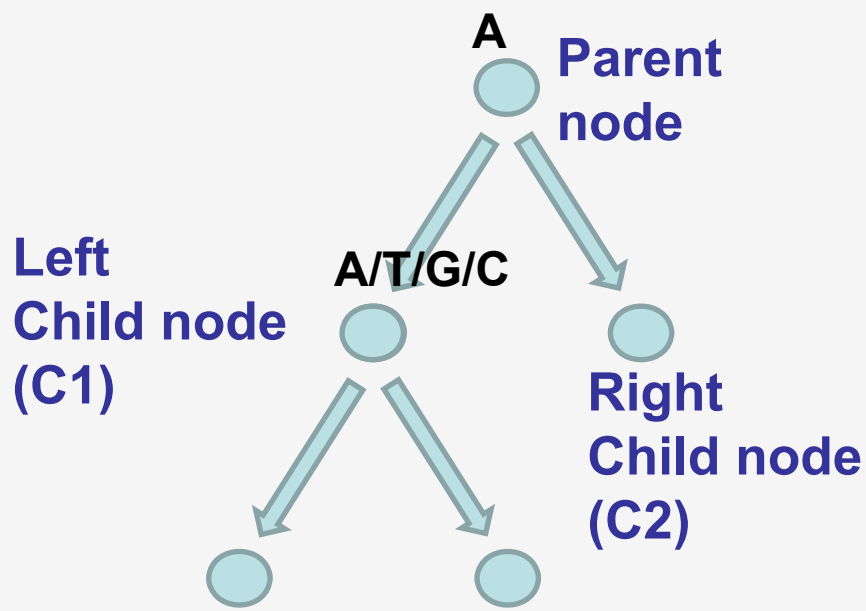
$$S_p(A)|_{LeftArm} = \min \begin{Bmatrix} c_{A\to A} + S_{C1}(A) \\ c_{A\to T} + S_{C1}(T) \\ c_{A\to G} + S_{C1}(G) \\ c_{A\to C} + S_{C1}(C) \end{Bmatrix}$$



Sankoff algo – cont'd

$$S_p(A)|_{LeftBranch} = \min \begin{Bmatrix} c_{A\to A} + S_{C1}(A) \\ c_{A\to T} + S_{C1}(T) \\ c_{A\to G} + S_{C1}(G) \\ c_{A\to C} + S_{C1}(C) \end{Bmatrix}$$

$$= \underbrace{\min_{j=A/T/G/C} \left\{ c_{A\to j} + S_{C1}(j) \right\}}_{LeftBranch}$$



$$S_p(i) = S_p'(i)_{left} + S_p'(i)_{right}$$

$$= \underbrace{\min_{j=A/T/G/C} \left\{ c_{i\to j} + S_{C1}(j) \right\}}_{LeftBranch} + \underbrace{\min_{k=A/T/G/C} \left\{ c_{i\to k} + S_{C2}(k) \right\}}_{RightBranch}$$

Sankoff algorithm – Example

$$S_p(i) = \underbrace{\min_{j=A/T/G/C}\left\{c_{i\to j} + S_{C1}(j)\right\}}_{LeftArm} + \underbrace{\min_{k=A/T/G/C}\left\{c_{i\to k} + S_{C2}(k)\right\}}_{RightArm}$$

A	C	G	T
∞	0	∞	∞

{C}

A	C	G	T
0	∞	∞	∞

{A}

A	C	G	T
∞	0	∞	∞

{C}

A	C	G	T
0	∞	∞	∞

{A}

A	C	G	T
∞	∞	0	∞

{G}

A	C	G	T
2.5	2.5	3.5	3.5

A	C	G	T
1	5	1	5

A	C	G	T
3.5	3.5	3.5	4.5

A	C	G	T
6	6	7	8

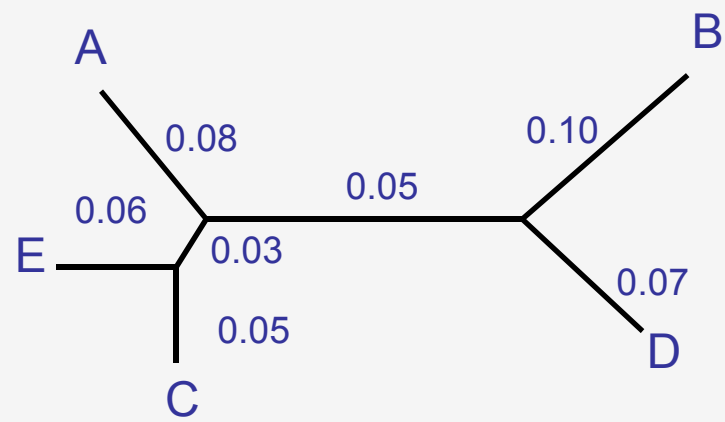
Cost matrix

	A	C	G	T
A	0	2.5	1	2.5
C	2.5	0	2.5	1
G	1	2.5	0	2.5
T	2.5	1	2.5	0

Problem 1: Assigning branch lengths using Least squares

We have an observed matrix of Distances between sequences from all possible pair-wise comparisons – Remember D and K in JC model?

	A	B	C	D	E
A	0	0.23	0.16	0.20	0.17
B	0.23	0	0.23	0.17	0.24
C	0.16	0.23	0	0.15	0.11
D	0.20	0.17	0.15	0	0.21
E	0.17	0.24	0.11	0.21	0

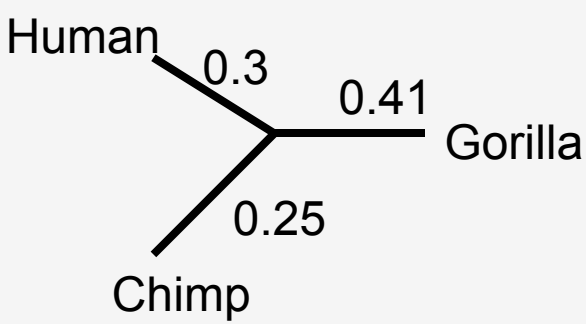


Observed distances denoted by D_{ij} and the "real branch" lengths to be <u>predicted</u> as d_{ij}

$$Q(T) = \sum_{i=1}^{n} \sum_{j=1; j \neq i}^{n} (D_{ij} - d_{ij})^2$$

Motivation: From a distance table to a tree

Each tree has branch lengths from which “predicted” set of distances can be computed: d(i,j) (small d, denotes the distance of the branches, unlike the observed pairwise distances D).



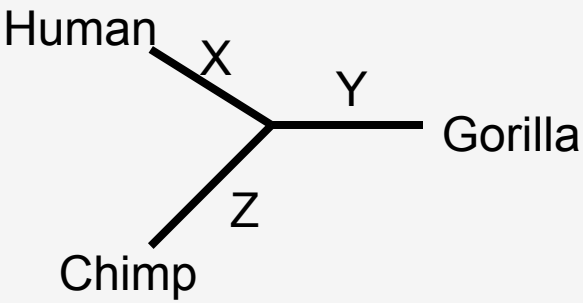
D(Human,Chimp) = 0.55

D(Human,Gorilla) = 0.71

D(Chimp, Gorilla) = 0.66

Motivation: From a distance table to a tree

The question is can we find branch lengths, so that the d's are equal to the D's?



D(Human,Chimp) = 0.3 =X + Z

D(Human,Gorilla) = 0.4 = X + Y

D(Chimp, Gorilla) = 0.5 = Y + Z

X+Z = 0.3

X+Y = 0.4

Y+Z = 0.5

Y-Z = 0.1

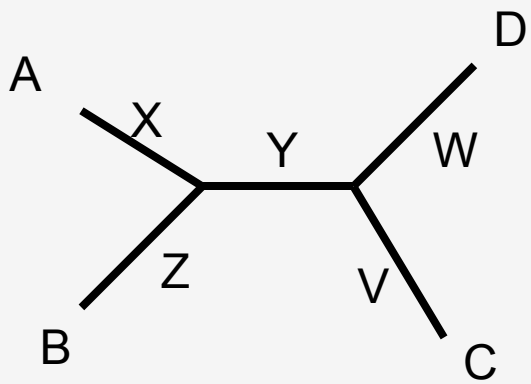
Y+Z = 0.5

Y = 0.3

Z = 0.2

X = 0.1

Is there always a solution??



5 Variables,

6 Equations,

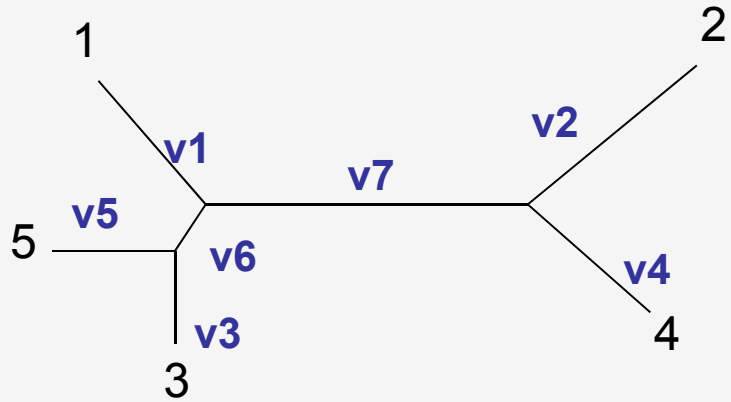
It might be that there's no single solution

$$\underbrace{NC_2}_{Number\ Of\ Equations} > \underbrace{2N-3}_{Number\ of\ variables} \quad \text{for N} > 3$$

Least squares – Cont'd

$$d_{ij} = \sum_k x_{ij,k} v_k$$

introduce an indicator variable, $x_{ij,k}$ which is 1 if branch v_k lies in the path from species i to species j and 0 otherwise



$$d_{12} = 1v_1 + 1v_2 + 0v_3 + 0v_4 + 0v_5 + 0v_6 + 1v_7$$

$$d_{13} = 1v_1 + 0v_2 + 1v_3 + 0v_4 + 0v_5 + 1v_6 + 0v_7$$

$$\ldots$$

$$d_{45} = 0v_1 + 0v_2 + 0v_3 + 1v_4 + 1v_5 + 1v_6 + 1v_7$$

LS- Cont'd

when the weights are 1.0 $Q(T) = \sum_{i=1}^{n} \sum_{j=1; j \neq i}^{n} (D_{ij} - d_{ij})^2$

$$= \sum_{i=1}^{n} \sum_{j=1; j \neq i}^{n} (D_{ij} - \sum_{k} x_{ijk} v_k)^2 \Rightarrow \frac{dQ}{dv_k} = -2 \sum_{i=1}^{n} \sum_{j=1;; j \neq i}^{n} x_{ij,k} (D_{ij} - \sum_{k} x_{ij,k} v_k) = 0$$

$$X^T D = (X^T X) v$$

$$v = (X^T X)^{-1} X^T D$$

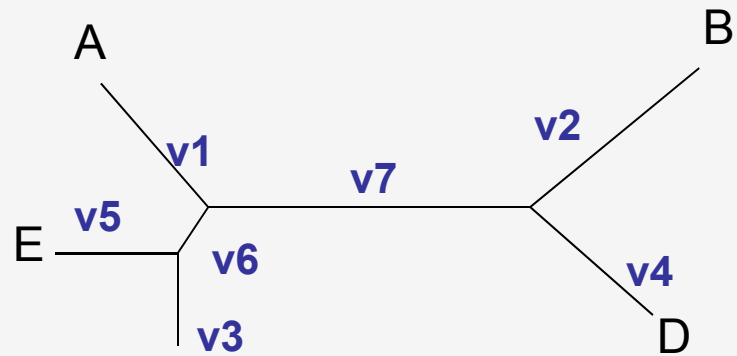
X

No of rows~n^2 (nC_2)
No of columns=2n-3 (eq to k)

D

No of rows nC_2
Columns=1

$$d_{ij} = \sum_{k} x_{ij,k} v_k$$



Problem 2: Construct the best tree

P2: Fast clustering algorithm for tree construction

UPGMA (unweighted pair group method using arithmetic averages)

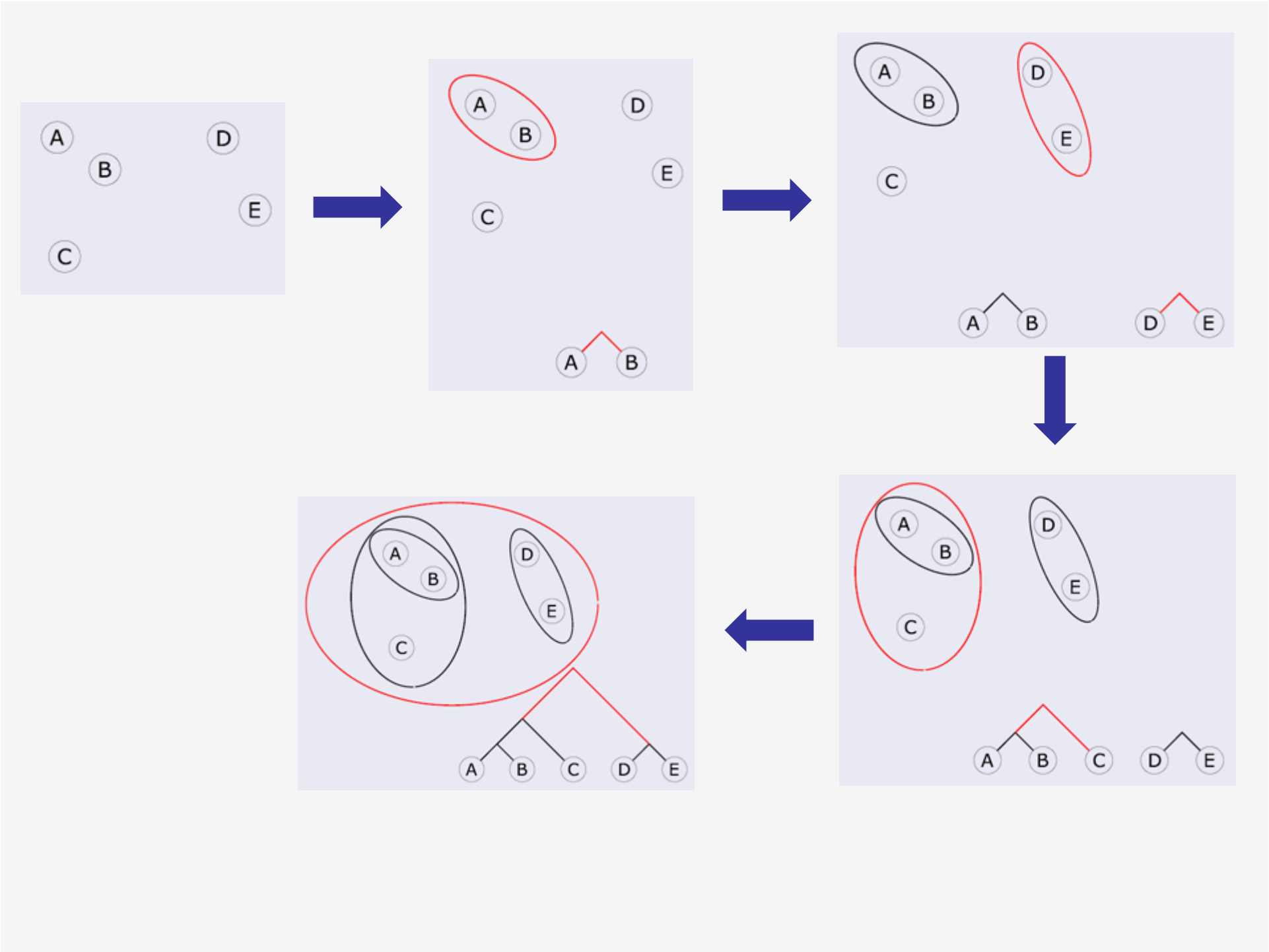
Given two disjoint clusters C_i, C_j of sequences,

$$d_{ij} = \frac{1}{|C_i| \times |C_j|} \Sigma_{\{p \in Ci,\ q \in Cj\}} d_{pq}$$

Note that if $C_k = C_i \cup C_j$, then distance to another cluster C_l is:

$$d_{kl} = \frac{d_{il}\,|C_i| + d_{jl}\,|C_j|}{|C_i| + |C_j|} = \text{UPGMA distance}$$

$$D((ij),l) = \left(\frac{n(i)}{n(i)+n(j)}\right)D(i,l) + \left(\frac{n(j)}{n(i)+n(j)}\right)D(j,l)$$


A
B
C
D
E
A
B
C
D
E
A
B
A
B
C
D
E
A
B
C
D
E
A
B
C
D
E
A
B
C
D
E

UPGMA algorithm

Find *i* and *j* with smallest D_{ij}

Create new group X by joining nodes *i* & *j*

Compute distances between X (new member) and others (old)

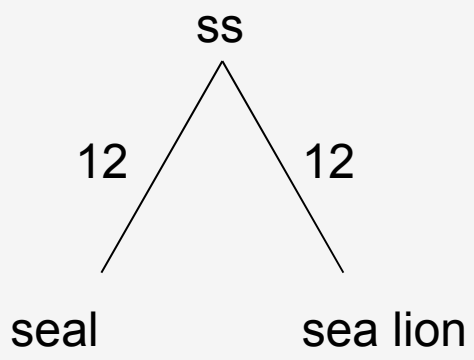
Place node X at $D_{ij}/2$

Delete i and j; replace with X in D-matrix

The distance table

	dog	bear	raccoon	weasel	seal	sea lion	cat	chimp
dog	0	32	48	51	50	48	98	148
bear		0	26	34	29	33	84	136
raccoon			0	42	44	44	92	152
weasel				0	44	38	86	142
seal					0	24	89	142
sea lion						0	90	142
cat							0	148
chimp								0

Distance between these two taxa was 24, so each branch has a length of 12.



We call the parent node of seal and sea lion “ss”.

Removing the seal and sea-lion rows and columns, and adding the ss row and columns

	dog	**bear**	**raccoon**	**weasel**	**ss**	**cat**	**chimp**
dog	0	32	48	51	?	98	148
bear		0	26	34	?	84	136
raccoon			0	42	?	92	152
weasel				0	?	86	142
ss					0	89	142
cat						0	148
chimp							0

Computing dog-ss distance

	dog	bear	raccoon	weasel	seal	sea lion	cat	chimp
dog	0	32	48	51	50	48	98	148

$$D((ij),k) = \left(\frac{n(i)}{n(i)+n(j)}\right)D(i,k) + \left(\frac{n(j)}{n(i)+n(j)}\right)D(j,k)$$

Here, i=seal, j=sea lion, k = dog.

n(i)=n(j)=1.

D(ss,dog) = 0.5D(sea lion,dog) + 0.5D(seal,dog) =
=0.5*48+0.5*50=49

The new table. Starting second iteration

	dog	bear	raccoon	weasel	ss	cat	chimp
dog	0	32	48	51	49	98	148
bear		0	26	34	31	84	136
raccoon			0	42	44	92	152
weasel				0	41	86	142
ss					0	89	142
cat						0	148
chimp							0

Inferring tree

Distance between bear and raccoon was 26, so each branch has a length of 13.

We call the parent node of bear and raccoon “br”.

Computing br-ss distance

	dog	bear	raccoon	weasel	ss	cat	chimp
ss	49	31	44	41	0	89.5	142

$$D((ij),k) = \left(\frac{n(i)}{n(i)+n(j)}\right)D(i,k) + \left(\frac{n(j)}{n(i)+n(j)}\right)D(j,k)$$

Here, i=raccoon, j=bear, k = ss.

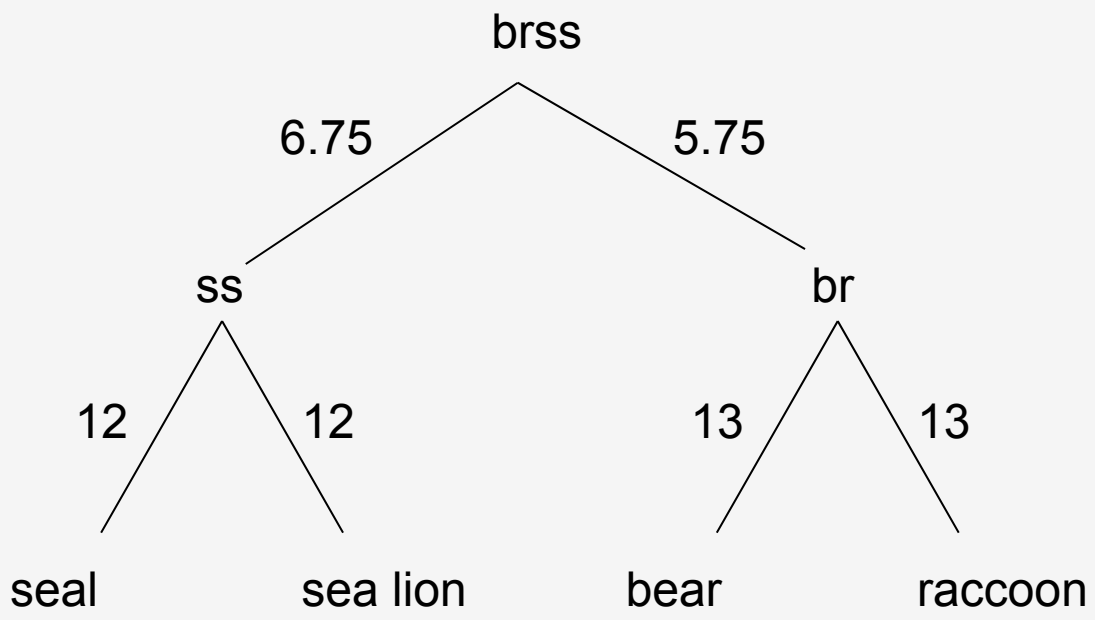
n(i)=n(j)=1. D(br,ss) = 0.5D(bear,ss)+0.5D(raccoon,ss)=37.5.

The new table. Starting next iteration

	dog	br	weasel	ss	cat	chimp
dog	0	40	51	49	98	148
br		0	38	37.5	88	144
weasel			0	41	86	142
ss				0	89	142
cat					0	148
chimp						0

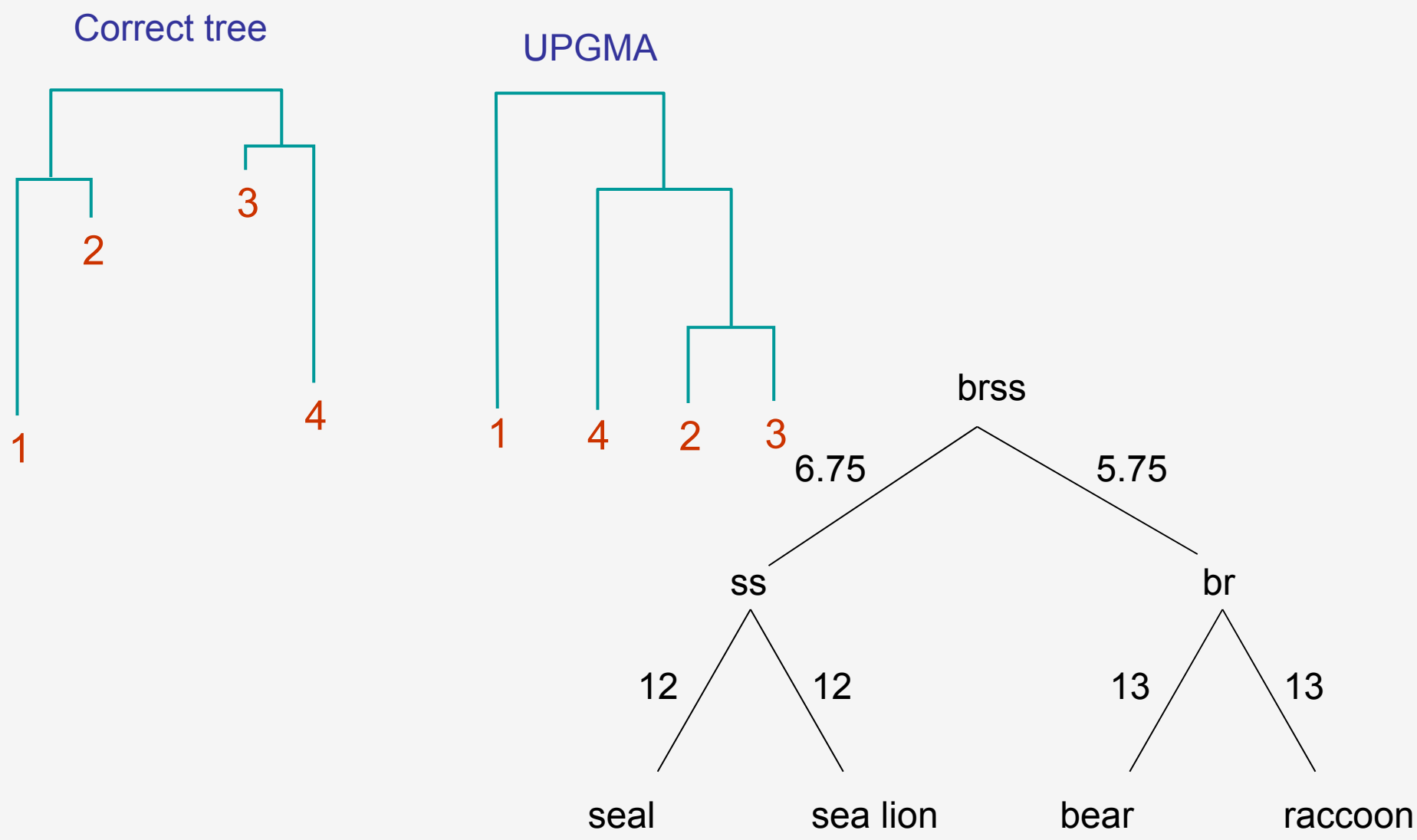
Inferring tree

Distance between br and ss was 37.5, so each branch has a length of 18.75. But this is the distance from brss to the leaves. The distance brss to ss is 18.75-12=6.75. The distance between brss to br is 18.75-13=5.75

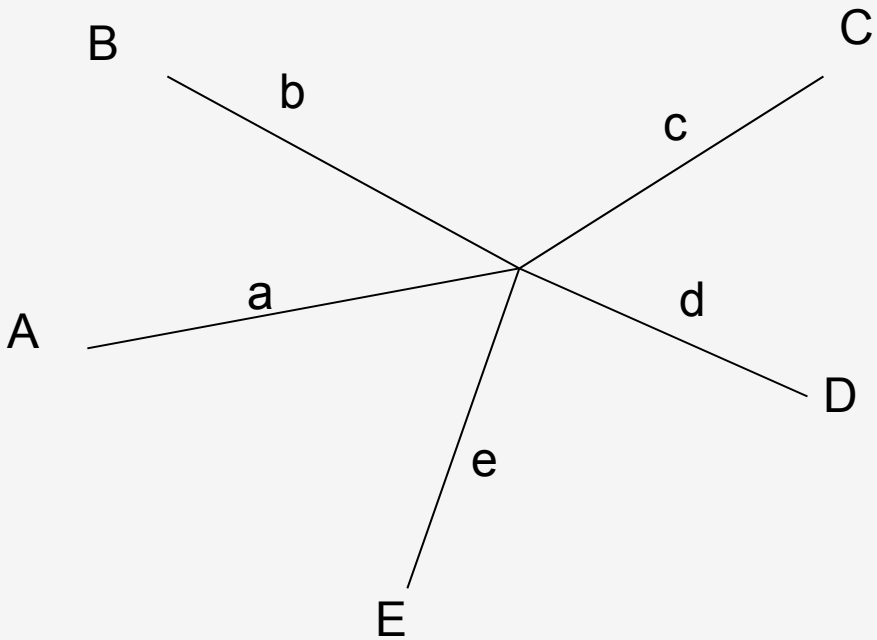


And so on ..

UPGMA's Weakness: Example



Neighbor joining: Star topology

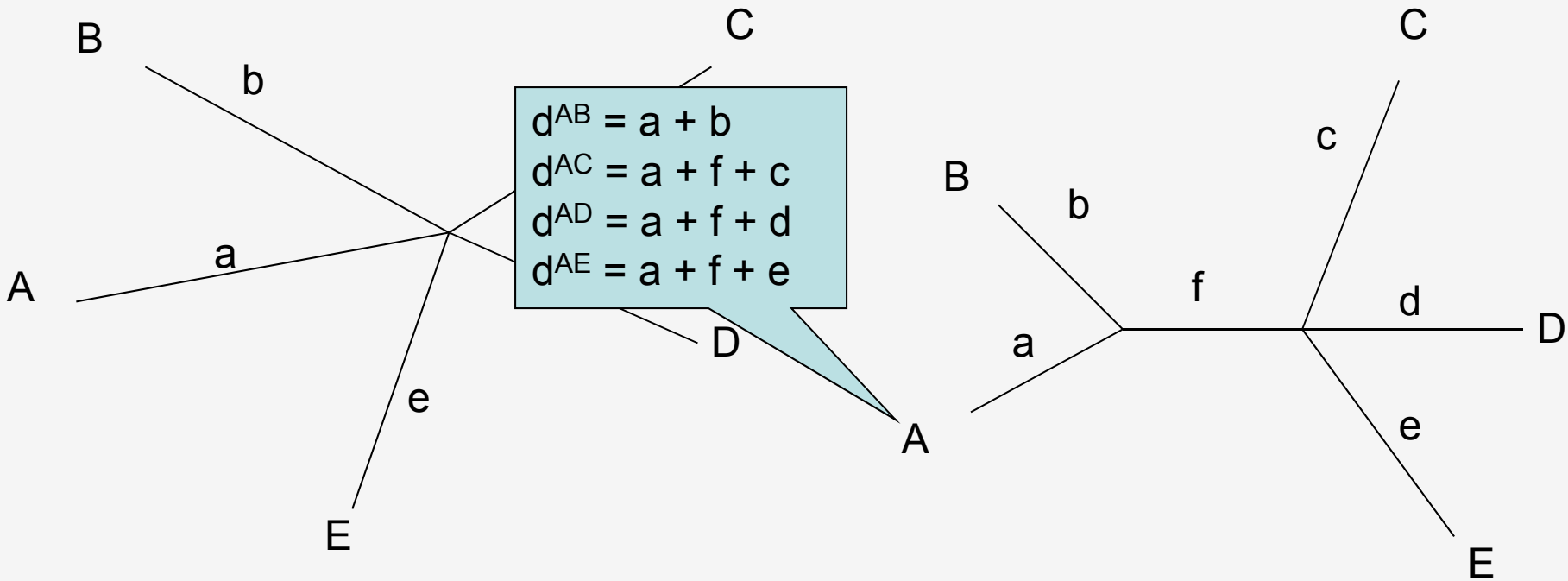


Sum of all branches is S*=a+b+c+d+e.

Summing all distances in the matrix counts each edge four times (e.g., d^{AB}, d^{AC}, d^{AD} and d^{AE}).

Hence, the sum of all distances in the matrix is 4S*.

Add one branch (the “first potential tree”)



Sum of branches with the new branch is

$$S = a + b + c + d + e + f$$

$$= (d^{AC} + d^{AD} + d^{AE} + d^{BC} + d^{BD} + d^{BE})/6 + d^{AB}/2 + (d^{CD} + d^{CE} + d^{DE})/3$$

Neighbor joining (general idea)

1. Add one branch to the star topology and compute the difference between S* and S.
2. Repeat for each pair of leaves in the tree.
3. Choose the pair that yields the largest difference (the closest neighbors).
4. Join that pair.
5. Repeat until all pairs are joined.

NJ algorithm

- For each tip compute $$u_i = \sum_{j:j\neq i}^{n} D_{ij} / (n-2)$$
- Choose i and j for which, $D_{ij} - u_i - u_j$ is smallest
- Join nodes i and j to X. Compute branch length from i to X and j to X

$$v_{i\to X} = (D_{ij} + u_i - u_j)/2$$

$$v_{j\to X} = (D_{ij} + u_j - u_i)/2$$

- Compute the distance between X and remaining nodes

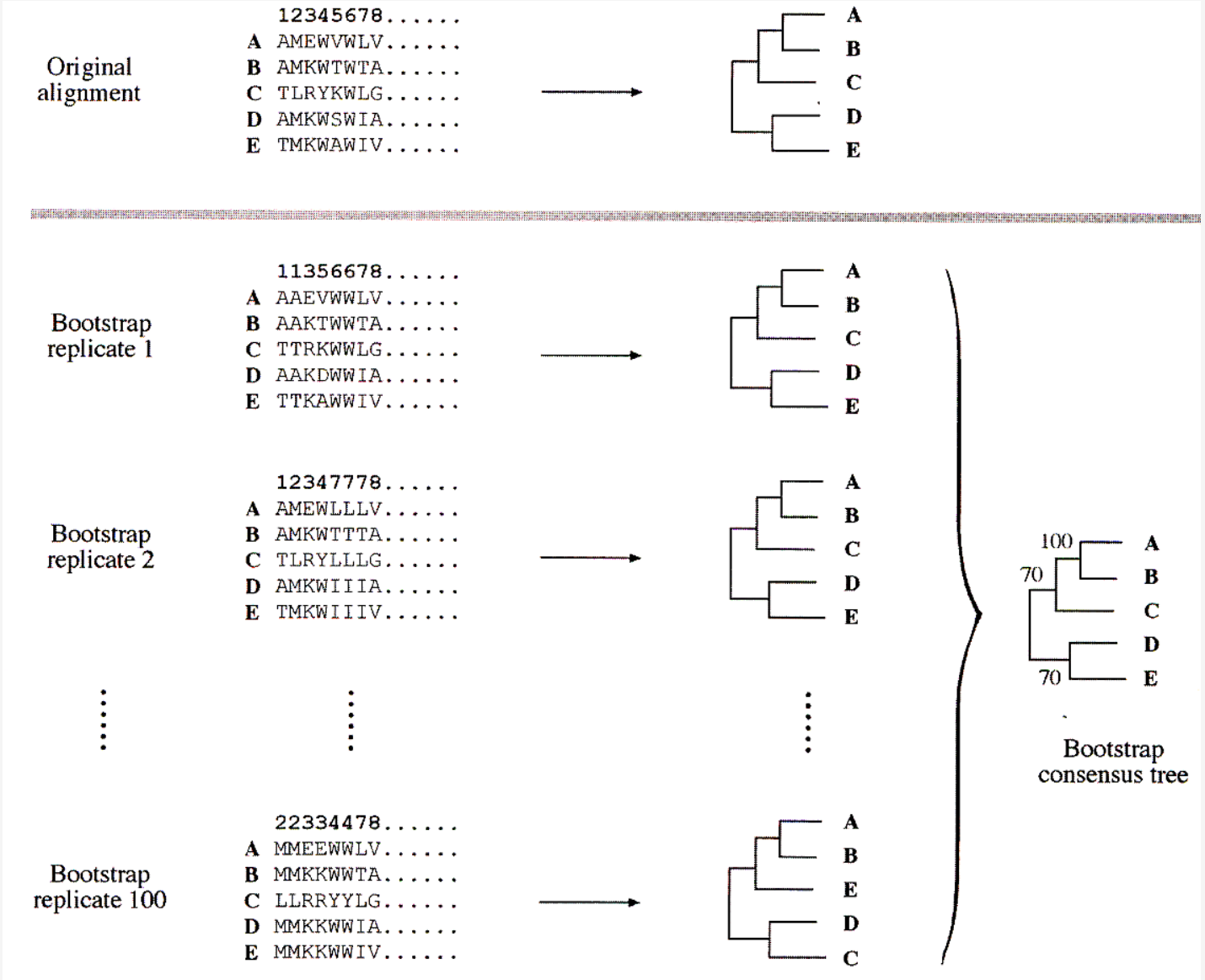
$$D_{k\to X} = (D_{ik} + D_{jk} - D_{ij})/2$$

NJ algorithm – Cont'd

- New node X is treated as a new tip and old nodes I, j are deleted

- If more than two nodes remain go back to step-1, else connect the two nodes (*l,m*) by $D_{l,m}$

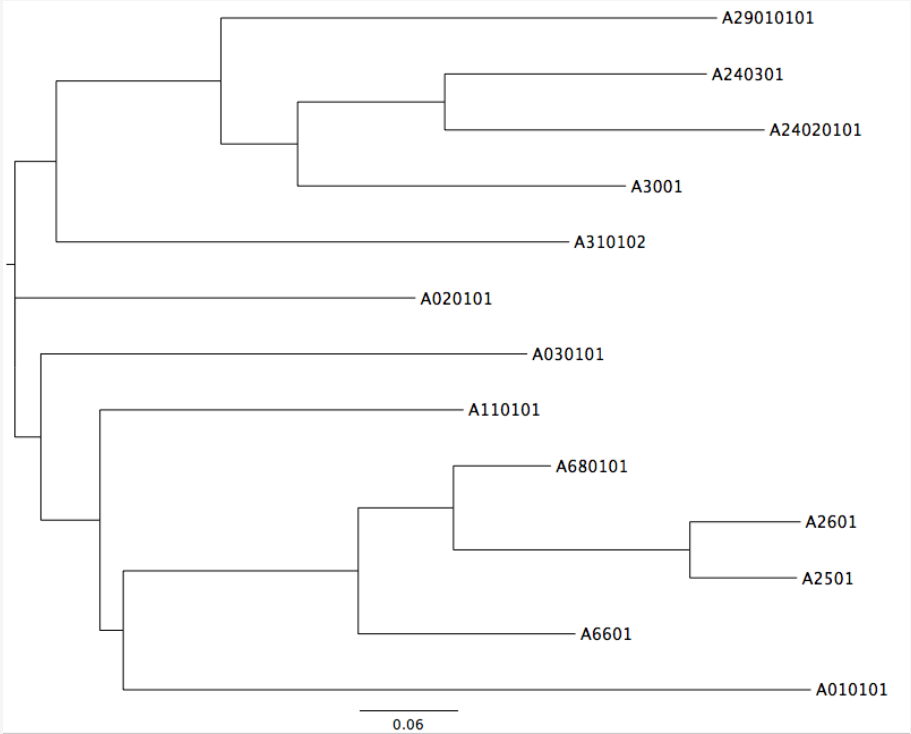
Bootstrapping:Confidence in trees



Trees: representation in computer files

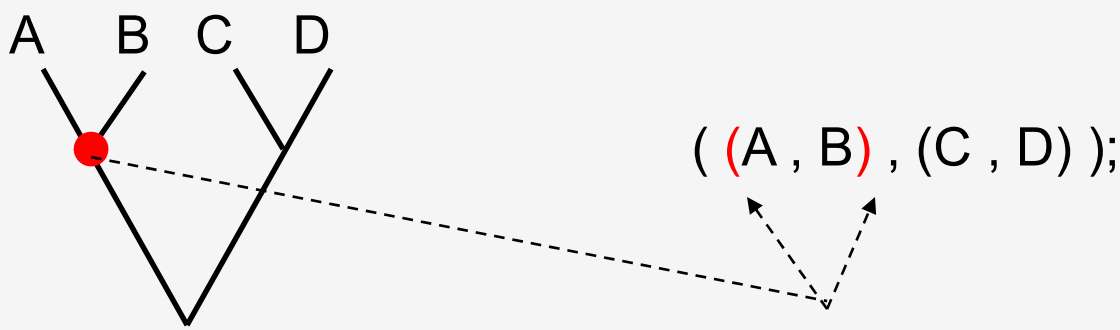
```
(((A29010101:0.312020,((A240301:0.164850,A24020101:0.201150):0.092520,A3001:0.
206480):0.048230):0.103580,A310102:0.322670):0.025970,(A020101:0.251900,(A030101
:0.305870,(A110101:0.228630,(((A680101:0.061300,(A2601:0.069640,A2501:0.067360):
0.148700):0.059880,A6601:0.136620):0.147790,A010101:0.432460):0.014620):0.037160
):0.016320):0.000000);
```

Newick format



Newick format: named for seafood restaurant where standard was decided upon

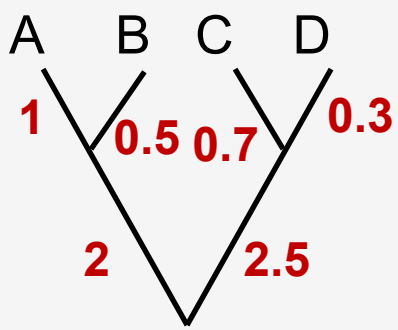
Trees: representation in computer files



Newick format:

- Leafs: represented by taxon name
- Internal nodes: represented by pair of matching parentheses
- Descendants of internal node given as comma-delimited list.
- Tree string terminated by semicolon

Trees: representation in computer files



((A:1 , B:0.5) :2 , (C:0.7 , D:0.3) :2.5);

((A:1 , B:0.5)L:2 , (C:0.7 , D:0.3)M:2.5)N;

Newick format:

- Leafs: represented by taxon name
- Internal nodes: represented by pair of matching parentheses
- Descendants of internal node given as comma-delimited list.
- Tree string terminated by semicolon

How to make a Tree?

1. Know the distances between each pair – e.g, multiple sequence alignment, or pairwise alignments

2. Apply UPGMA or NJ

3. Calculate statistics: Bootstrap

4. Visualize your tree (*e.g.*, Treeview)

5. Or use an all-in-one program :-) e.g. CLUSTALW

Phylogenetic software

Software packages

- Freely available
  - Phylip (widely used)
  - BioNJ
  - PhyML
  - Tree Puzzle
  - MrBayes
- Commercial
  - PAUP (widely used)
  - MEGA

www.ebi.ac.uk

European Bioinformatics Institute

http://www.ebi.ac.uk/

Talk Reason:...apologetics CBS DTU NCBI Literature Evolution Math/Statistics Mac Programming Wifi Wikipedia

EMBL-EBI

EB-eye Search | All Databases | Enter Text Here | Go | Reset | Advanced Search | Give us feedback

Databases | Tools | EBI Groups | Training | Industry | About Us | Help | Site Index

Data Resources &Tools

- EMBL-BANK
- UniProt
- ArrayExpress
- Ensembl
- InterPro
- PDB-EBI
- Genomes
- Nucleotide Sequences
- Protein Sequences
- Macromolecular Structures
- Small Molecules
- Gene Expression
- Molecular Interactions
- Reactions & Pathways
- Protein Families
- Enzymes
- Literature
- Taxonomy
- Ontologies
- Sequence Similarity & Analysis
- Pattern & Motif Searches
- Structure Analysis
- Text Mining
- Downloads

European Bioinformatics Institute

About the EBI

- Research
- PhD Studies
- Training
- Industry Support
- Group & Team Leaders
- EBI Funders
- User Support
- EBI Mission
- People
- Events at the EBI
- How to Find us

EBI hosted EU Project Websites

- BioSapiens
- E-MeP
- ELIXIR
- EMBRACE
- EMERALD
- ENFIN
- FELICS
- SYMBIOmatics

Hands-on Courses

Registration is open for:

- **Patterns, Similarities and Differences in Biological Data,** 9-11 June 2008... more.
- **Programmatic access of Proteomics Resources,** 28-31 July 2008... more

See the full hands-on programme.

Research Highlights

May 07, 2008

Platypus genetic blueprint reveals the early history of mammals

Researchers at the European Molecular Biology Laboratory's European Bioinformatics Institute in Cambridge and the Medical Research Council Functional Genomics Unit in Oxford have revealed the DNA blueprint of the platypus. The analysis is part of an international research collaboration including scientists from the UK, the USA and Australia and is reported in the 8 May issue of Nature... more

Phylogenetic servers

- http://www.phylogeny.fr/
- http://bioweb.pasteur.fr/seqanal/phylogeny/intro-uk.html
- http://atgc.lirmm.fr/phyml/
- http://phylobench.vital-it.ch/raxml-bb/
- http://www.fbsc.ncifcrf.gov/app/htdocs/appdb/drawpage.php?appname=PAUP
- http://power.nhri.org.tw/power/home.htm