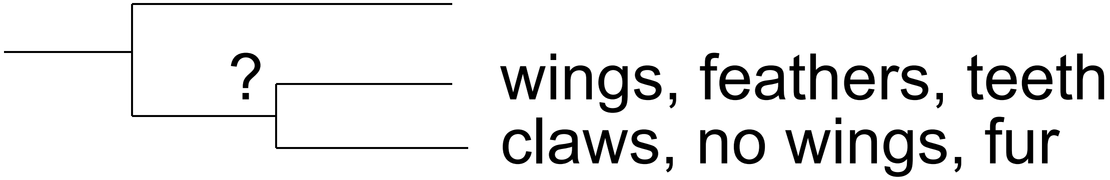
Maximum Parsimony
(character based phylogeny)

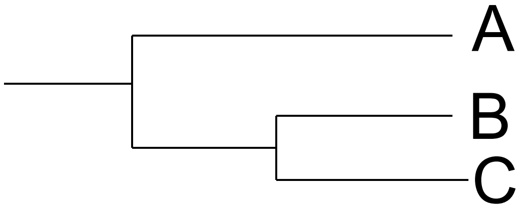
Phylogeny questions

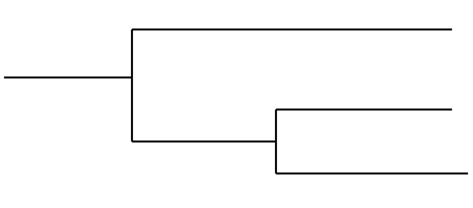
- Given several organisms & a set of features (usually sequence, but also morphological: wing shape/color...)

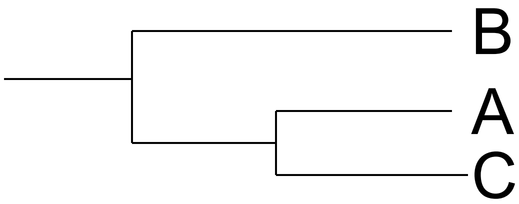
- A. Given a phylogenetic tree – figure out what the ancestors looked like (what are the features of internal nodes)



- B. Find the phylogenetic tree that best describes the common evolutionary heritage of the organisms



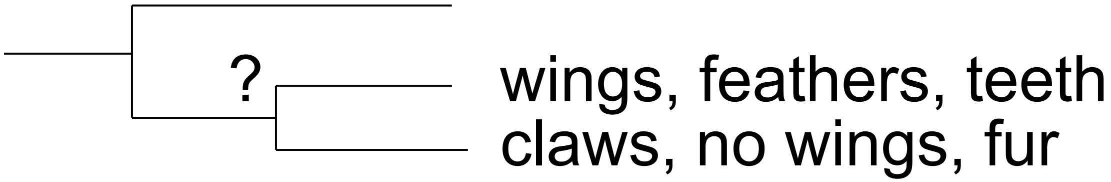




Phylogeny questions

•Given several organisms & a set of features (usually sequence, but also morphological: wing shape/color...)

•A. Given a phylogenetic tree – figure out what the ancestors looked like (what are the features of internal nodes)



?

wings, feathers, teeth

claws, no wings, fur

Need to define objective: minimize number of changes from parent to child (parsimony)

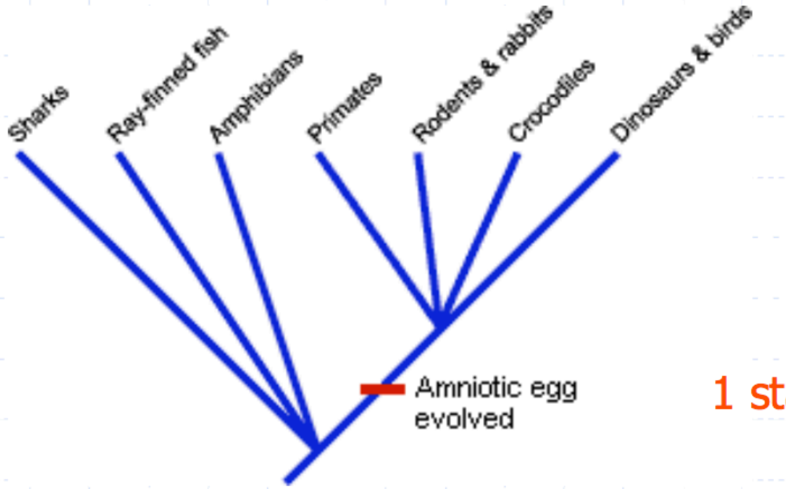
Example

Characters

	Vertebrae?	Bony skeleton?	Four limbs?	Amniotic egg?	Hair?	Two post-orbital fenestrae?
Sharks and relatives	YES	no	no	no	no	no
Ray-finned fishes	YES	YES	no	no	no	no
Amphibians	YES	YES	YES	no	no	no
Primates	YES	YES	YES	YES	YES	no
Rodents and rabbits	YES	YES	YES	YES	YES	no
Crocodiles and relatives	YES	YES	YES	YES	no	YES
Dinosaurs and birds	YES	YES	YES	YES	no	YES

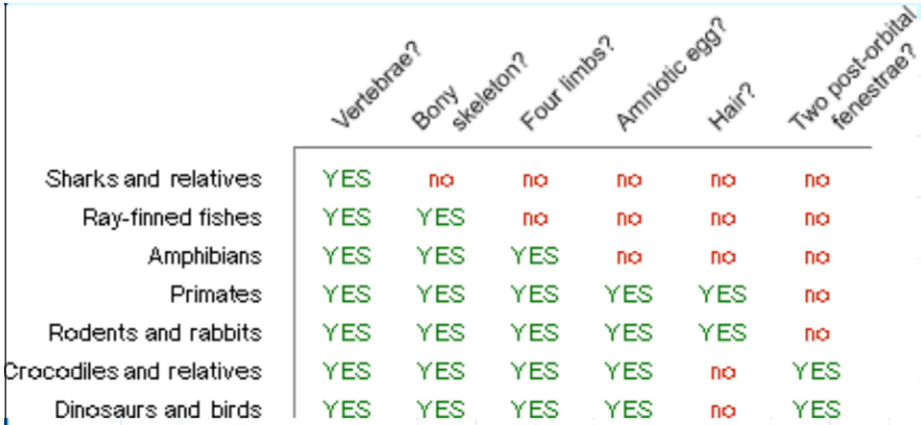
Binary states

Taxa



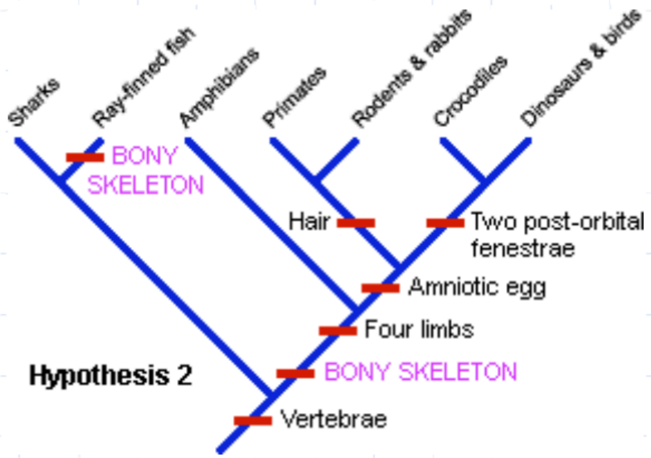
1 state change

Example

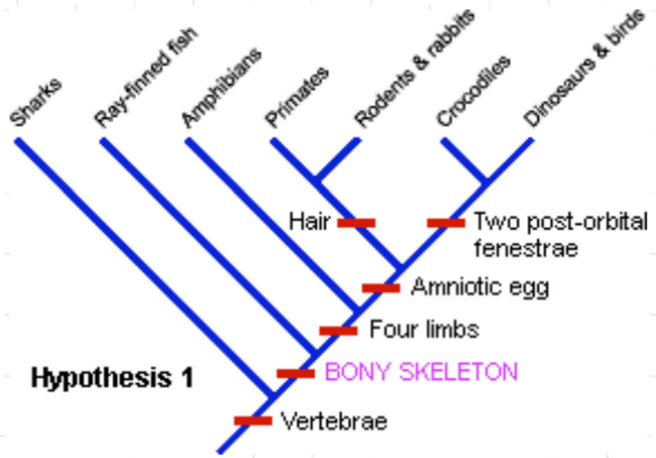


	Vertebrae?	Bony skeleton?	Four limbs?	Amniotic egg?	Hair?	Two post-orbital fenestrae?
Sharks and relatives	YES	no	no	no	no	no
Ray-finned fishes	YES	YES	no	no	no	no
Amphibians	YES	YES	YES	no	no	no
Primates	YES	YES	YES	YES	YES	no
Rodents and rabbits	YES	YES	YES	YES	YES	no
Crocodiles and relatives	YES	YES	YES	YES	no	YES
Dinosaurs and birds	YES	YES	YES	YES	no	YES

7 state changes



6 state changes



Two questions:

- What is the parsimony score of a given tree (small parsimony)
    - Solvable in polynomial time through Sankoff's algorithm
- What is the tree that minimizes the parsimony score over all possible trees (large parsimony)
    - NP-hard - solutions involve variants of random searches

Scoring a tree – Sankoff's algorithm

- Assumption – we try to minimize # of state changes from root to leaves – Parsimony approach

- Small parsimony

–given a tree where leaves are labeled with m-character strings

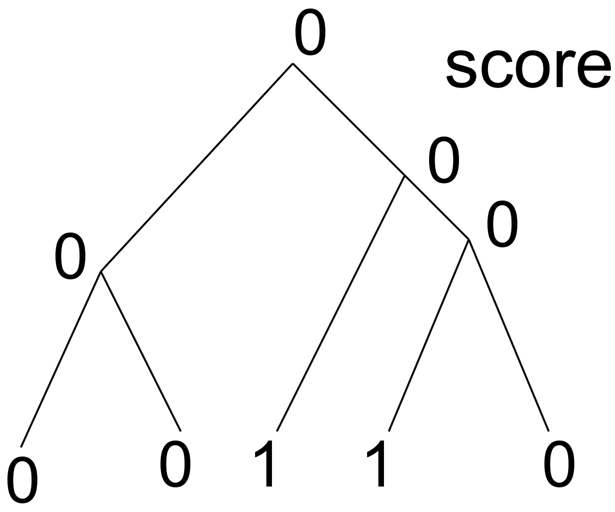
–find labels at internal nodes s.t. # of state transitions is minimzed

- Weighted small parsimony

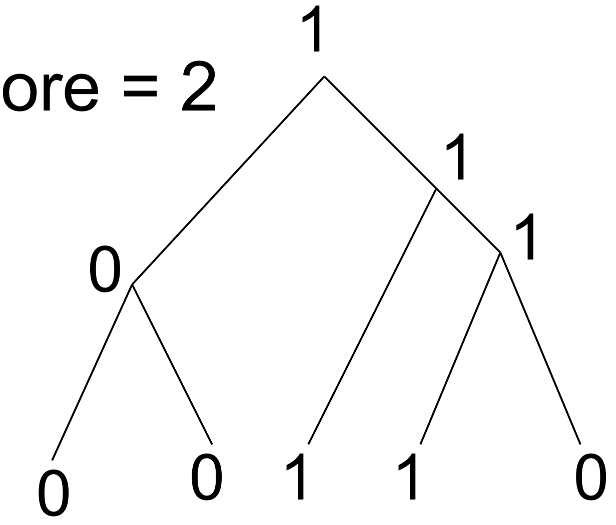
–same as parsimony except that state transitions are assigned weights

–minimize the overall weight of the tree

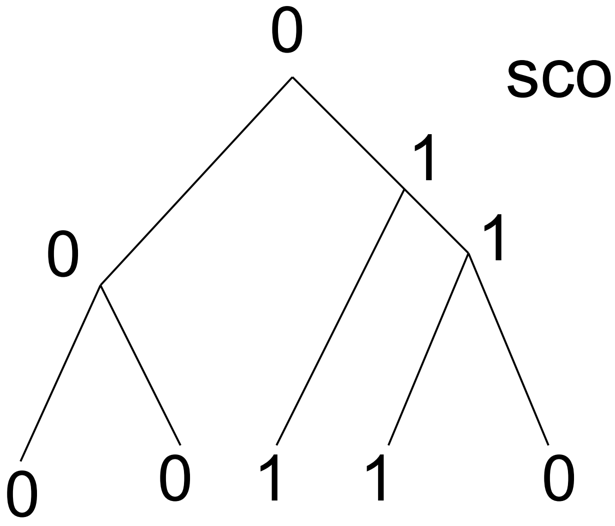
Example



score = 2



score = 2

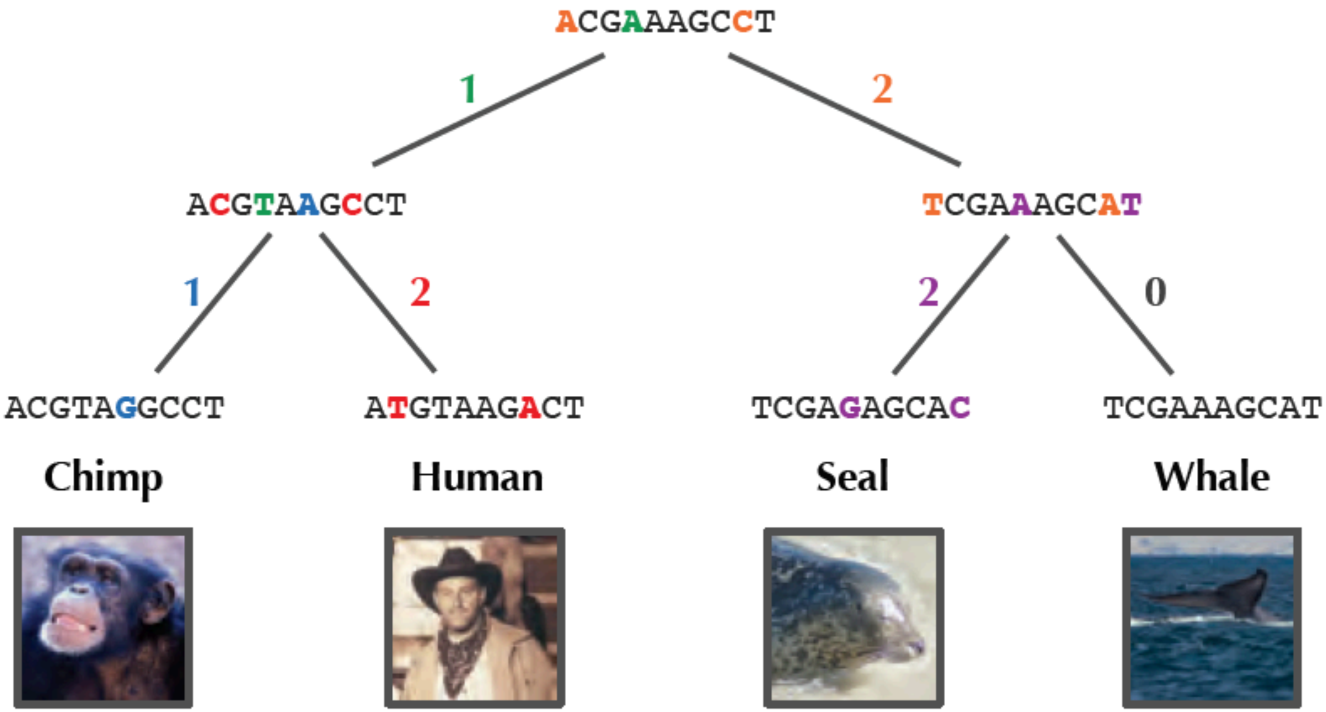


score=2

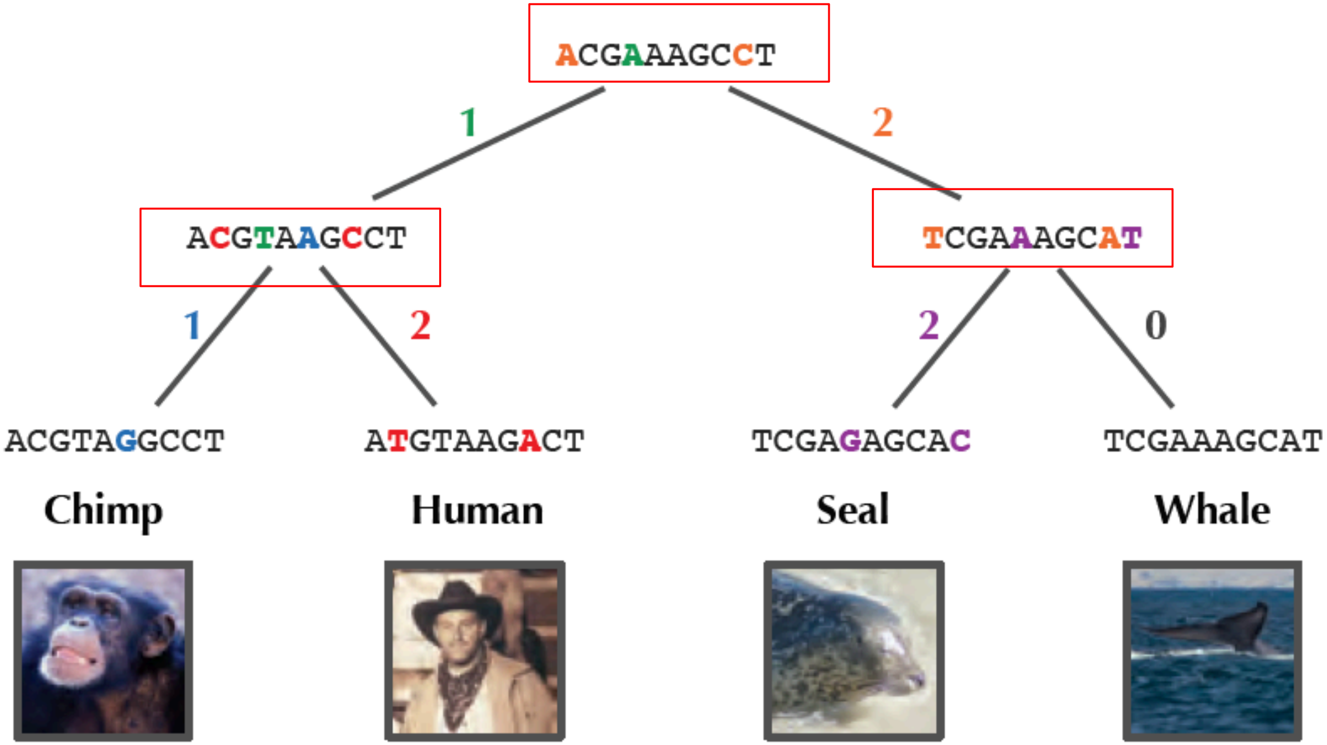
Can you do better?

Scoring a tree – Sankoff's algorithm

score = 8



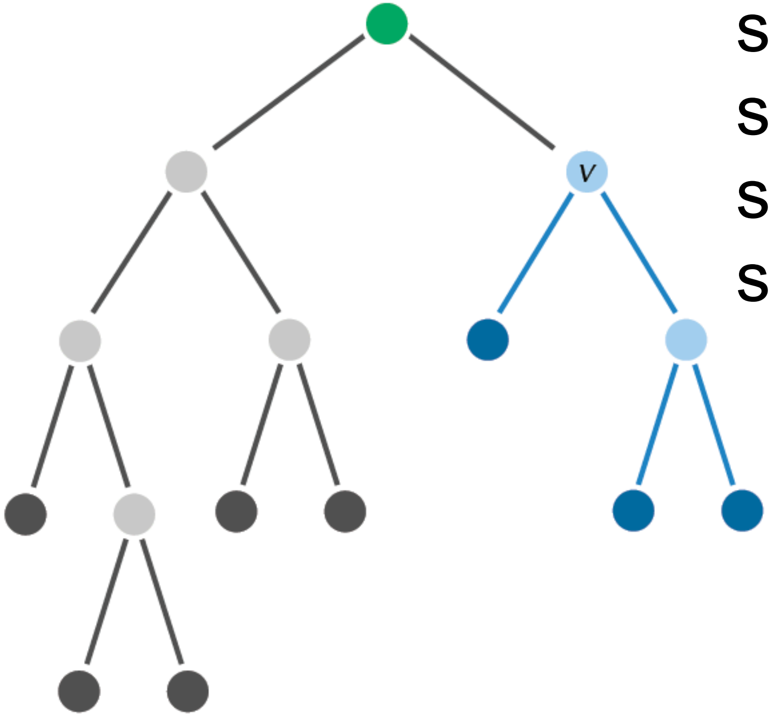
Scoring a tree – Sankoff's algorithm



Make a simplifying assumption – all characters are independent in the sequence
i.e. Run separately for each character then merge results

Sankoff's algorithm – recurrence relation

- At each node v in the tree store s(v,t) – best parsimony score for subtree rooted at v if character stored at v is t



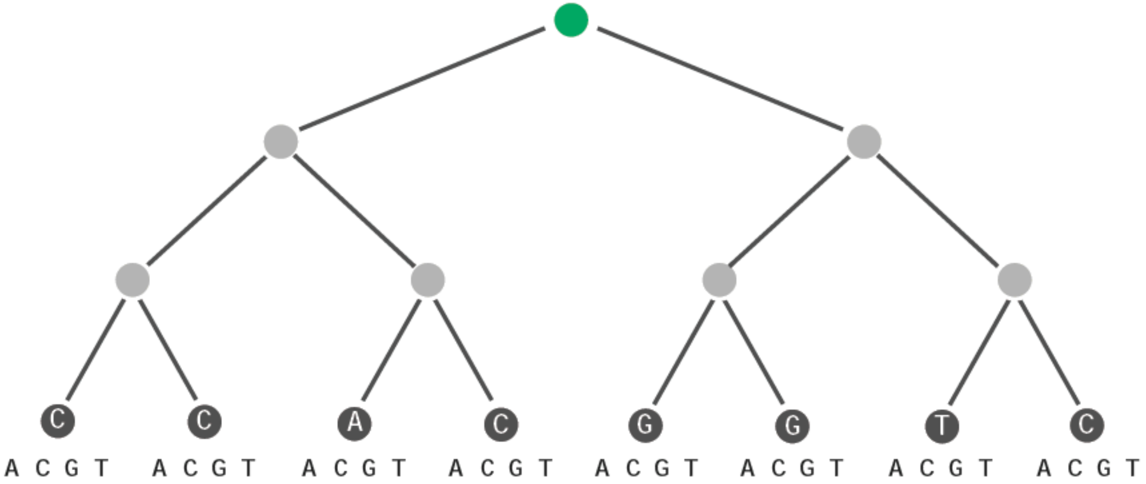
s(v, 'A')
s(v, 'C')
s(v, 'G')
s(v, 'T')

The (blue) subtree T_v of a node *v* within a larger rooted binary tree *T*.

Sankoff's algorithm – recurrence relation

- At each node v in the tree store s(v,t) – best parsimony score for subtree rooted at v if character stored at v is t

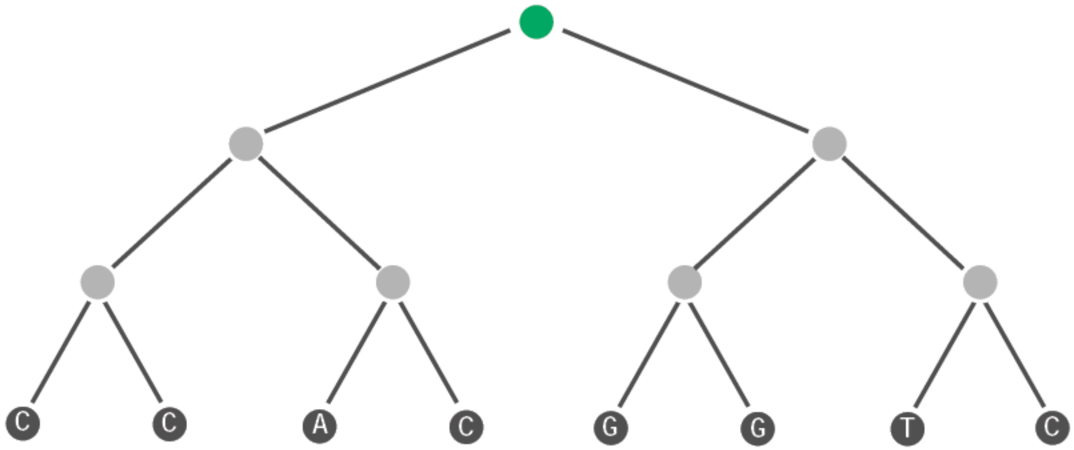
What is the base case?



Sankoff's algorithm – recurrence relation

•At each node v in the tree store s(v,t) – best parsimony score for subtree rooted at v if character stored at v is t

How to get a Base case?

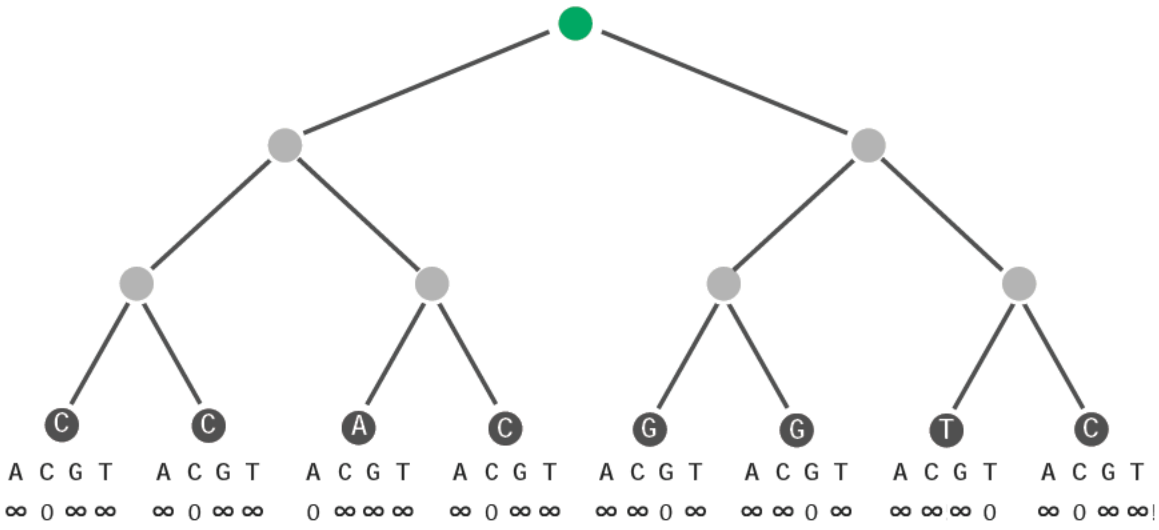


S(v, t)=

A	C	G	T	A	C	G	T	A	C	G	T	A	C	G	T	A	C	G	T	A	C	G	T	A	C	G	T	A	C	G	T
∞	0	∞	∞	∞	0	∞	∞	0	∞	∞	∞	∞	0	∞	∞	∞	∞	0	∞	∞	∞	0	∞	∞	∞	∞	0	∞	0	∞	∞

Sankoff's algorithm – recurrence relation

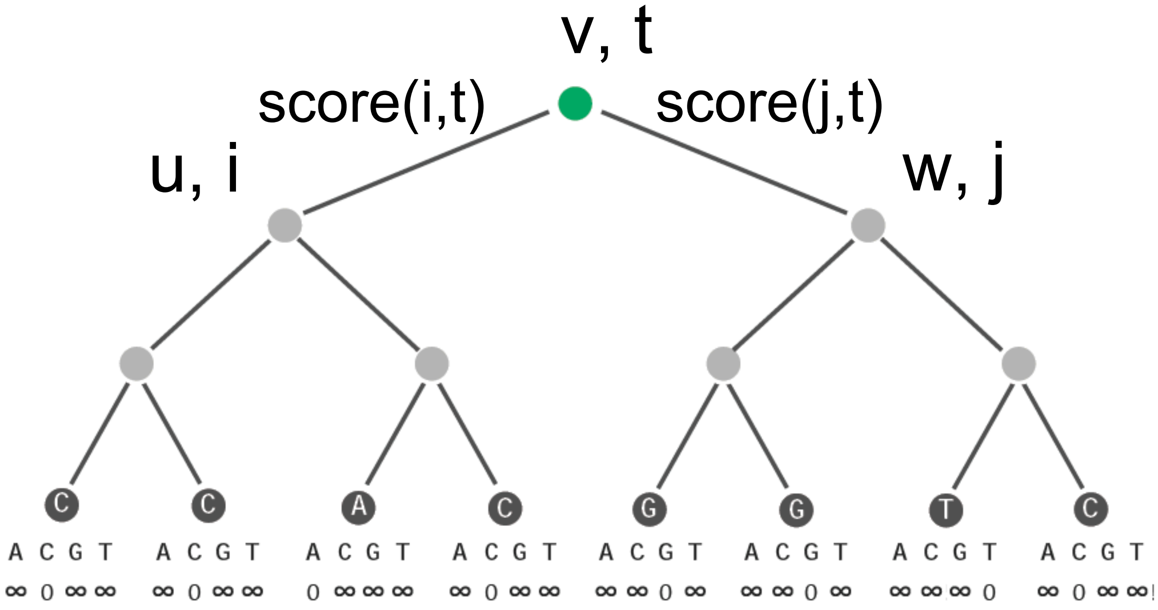
- At each node v in the tree store s(v,t) – best parsimony score for subtree rooted at v if character stored at v is t
- Traverse the tree in post-order and update s(v,t) as follows



Sankoff's algorithm – recurrence relation

–assume node v has children u and w

–$s(v,t) = \min_i \{s(u,i) + score(i,t)\} + \min_j \{s(w,j) + score(j,t)\}$



Sankoff's algorithm – recurrence relation

- Traverse the tree in post-order and update s(v,t) as follows

  – assume node v has children u and w

  – $s(v,t) = \min_i \{s(u,i) + score(i,t)\} + \min_j \{s(w,j) + score(j,t)\}$

- the minimum parsimony score is given by the smallest score s(root,t) over all symbols t

- backtrack to fill the values for the other nodes

- Note – this solves the weighted version. For unweighted set score (i,i) = 0, score(i,j) = 1 for any i,j