Consensus Methods:

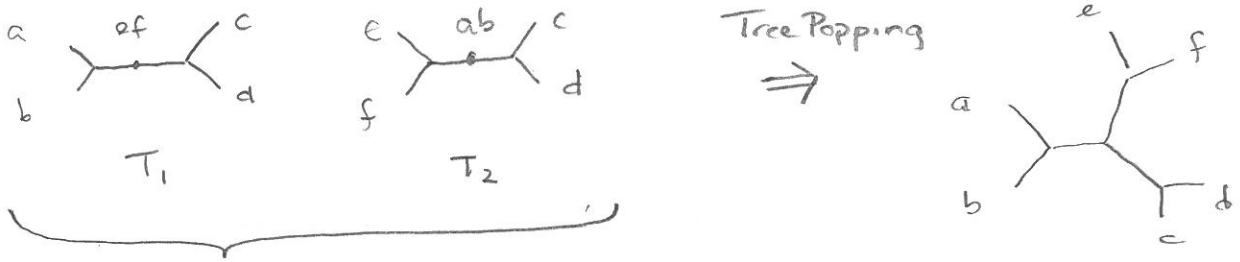
The Splits Equivalence Theorem: Single tree

compatible splits $\longleftrightarrow$ tree T

Now: Suppose we have k trees, $T_1, \ldots, T_k$, how can we find <u>one</u> tree that summarizes them all.

Clearly, if $\mathcal{S}$ = collection of splits on all k-trees and all splits are pairwise compatible, then there exists an X-tree T whose displayed splits are exactly those in $\mathcal{S}$.

Eg. $\mathcal{S}$ contains all trivial splits and ab|cdef, cd|abde, ef|abcd are pairwise compatible ✓ and from tree popping we get



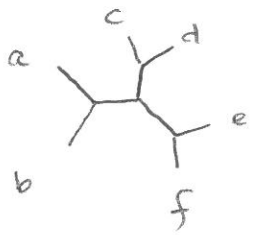
2 trees with Compatible Splits

When splits are incompatible, need a way to reconcile them.

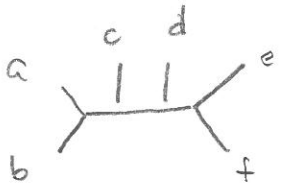
Consensus Method 1: Strict consensus.

For this and all subsequent examples, trivial splits are not explicitly written.

Strict Consensus:



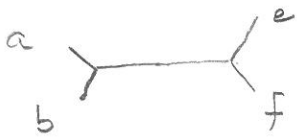
T_1



T_2

The strict consensus tree displays only those splits in all trees.

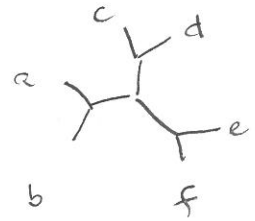
T_1	T_2	
ab | cdef	ab | cdef	ab | cdef , ef | abcd
~~cd | abef~~	~~abc | def~~	→
ef | abcd	ef | abcd	



Majority Rule Consensus Tree T:

Display only those trees that occur in a majority of tree T_i

T_1



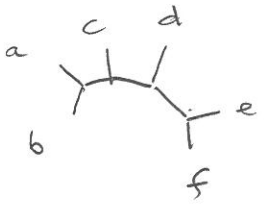
Trivial Splits

ab \| cdef

cd \| abef

ef \| abcd

T_2



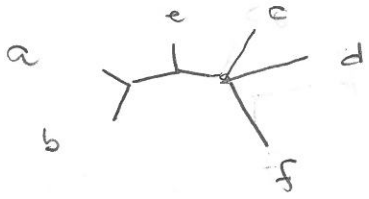
Trivial Splits

ab \| cdef

abc \| def

ef \| abcd

T_3



Trivial Splits

ab \| cdef

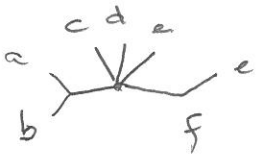
abe \| cdf

majority = 2 or 3 / 3 trees 66% or 100% > 50%

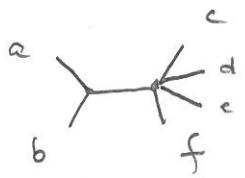
Trivial Splits, ab \| cdef ef \| abcd

100% 100% 66%

Majority Rule Tree:



Strict Consensus Tree:



Theorem: Majority Rule gives a set of pairwise compatible splits and therefore a tree. Formally, any two splits that occur in 50% or more of the trees $T_1, \ldots, T_K$ are compatible.

Pf: Suppose $X_0|X_1$ and $Y_0|Y_1$ occure in $\geq 50\%$ of the K-trees. Then by the pigeon-hole principle they occur on at least one tree. Since the occur on a tree, they are compatible.

Note that strict consensus and majority rule are at the ends of a spectrum. If p is any percentage $.5 \leq p \leq 1$, then you can define a consensus tree which displays splits in at least p of them.

Aside: if $p < .5$, then the resulting splits might not be compatible.

To get around this, one might use GREEDY CONSENSUS (pick most frequent, place on tree, remove those incompatible, repeat)

Eg. 100 trees on 40 taxa

...

Demo + HW

Another approach to building consensus trees is to use

SUPER-TREE methods. Particularly good if some 'taxa are missing.

Eg. Quartet Methods

Defn: A QUARTET TREE is an unrooted, binary tree with 4 labelled leaves.

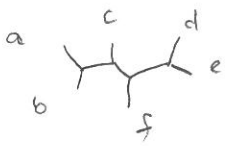
Eg. If $X = \{a, b, c, d\}$, there are four quartet trees

$ab|cd$ $\quad$ $ac|bd$ $\quad$ $ad|bc$

Theorem: The collection of quartets on a tree $Q(T)$ determines a binary unrooted tree.
(non-binary too)

Proof by induction.

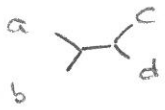
First: Given any binary tree, some quartet singles out each internal edge.

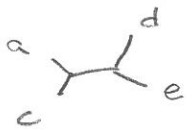


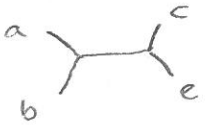
$\rightsquigarrow$ quartets $ab|cf$, $bc|df$, $af|de$

Quartet Method: (informally)

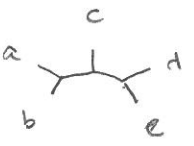
Eg 1. Compatible



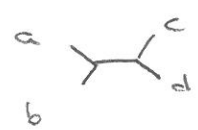


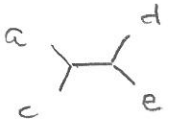


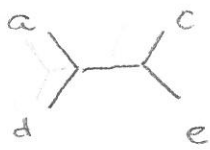
$\equiv$



Eg 2. Incompatible







3rd quartet not on tree.