Chapter 4: Combinatorics of Trees II

SPLITS, CLADES, QUARTETS

Theorems: Split Equivalence Theorem

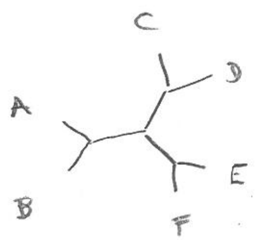
Methods: Tree-Popping, Refinement + Consensus Trees

Super-trees

Splits and Clades:

Assume first that a phylogenetic tree is given

Unrooted Setting:



Rooted Setting

Splits on trees correspond to edges in T.

{A}, {B, C, D, E, F}

{B}, {A, C, D, E, F} trivial

etc.

{A,B}, {C,D,E,F}

{C,D}, {A,B, E, F}

{E, F}, {A,B, C, D}

Such splits, also denoted

AB | CDEF etc are

DISPLAYED on T

Clades on rooted trees correspond to sets of taxon labels descendant from a node in T^{ρ}.

trivial clades: ...

non-trivial clades

{A, B}, {A, B, C}, {A, B, C, D}

X

Clades correspond to

MONOPHYLETIC GROUPS on trees

The notion of split (and clade) is more general:

Defn: A SPLIT of X is a bipartition of X

$X = X_0 \sqcup X_1$, written $X_0 | X_1$, with $X_0 \neq \emptyset$, $X_1 \neq \emptyset$. (and sometimes dropping braces)

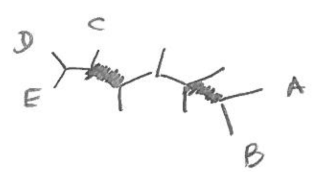
If $|X| = n$, then there are $2^{n-1} - 1$ splits, yet a binary tree has only $2n-3$ edges $\Rightarrow$ splits versus splits on trees.

A natural question is: when does a collection of splits fit a tree?

Informal Explanation: Need the notion of compatibility demo on T

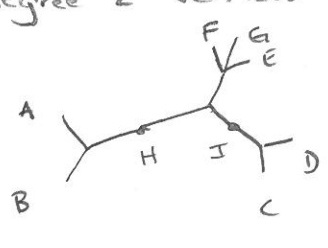
Defn: Two splits $X_0 | X_1$ and $Y_0 | Y_1$ are COMPATIBLE if for some i, j $X_i \cap Y_j = \emptyset$

It should be clear that any collection of splits on a tree are compatible.



This is .5 of the SPLIT EQUIVALENCE THEOREM:

Defn: Let X be a set of labels. An X-tree is a tree T together with a labelling map such that all leaves and degree 2 vertices are labelled. possible ambiguity?



Theorem: Splits Equivalence Theorem

Let S be a collection of splits on X. Then

there exists an X-tree T displaying exactly the splits in S **if, and only if,** the splits in S are pairwise compatible.

Moreover, T is unique up to isomorphism.

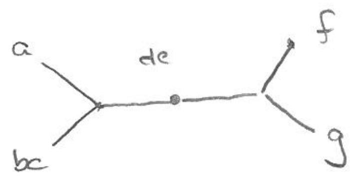
Proof. ⇒) We've already argued the splits displayed by a tree are compatible.

⇐) TREE-POPPING algorithm

Example: Check that the collection of splits

$a|bcdefg \quad f|abcdeg \quad g|abcdeg \quad fg|abcde \quad abc|defg \quad bc|adefg$

is pairwise compatible. Then use Tree-popping to construct the unique X-tree T for these splits.

Ans: 

Each step in tree-popping gives a REFINEMENT (finer resolution). Formally,

a tree T′ is is REFINEMENT of T if every split of T is displayed on T′.