New tree

Version 1.0.0, by Giorgio Bianchini

Description: Creates a new random tree or builds a tree from a sequence alignment.

Module type: MenuAction

Module ID: `36a54db0-4e24-4786-8b84-ad4e188c3285`

This module is used to create a new tree. This can either be a random tree, or a tree created using the neighbour-joining/UPGMA methods.

Further information

Random trees

A random tree can be created using one of four models: the proportional-to-distinguished-arrangements (PDA) model, the Yule-Harding-Kingman (YHK) model (also known as the Yule or pure-birth model), the coalescent model and the birth-death process model.

The PDA and YHK models produce tree topologies without branch lengths; branch lengths can be associated to these by drawing them from a probability distribution (available distributions are the uniform, gamma and exponential distributions). The coalescent model and the birth-death model determine branch lengths as part of the tree sampling. The PDA and YHK model can be used to produce rooted or unrooted topologies, while the coalescent and birth-death process produce rooted clock-like trees.

The trees generated can either be labelled or unlabelled. Is is possible to specify the number of tips in the tree either directly, or by providing a list of tip labels. When the option to keep extinct lineages in the birth-death process is selected, these are left unlabelled.

Labelled trees can be constrained to follow a fixed topology, by using the tree that is currently open in TreeViewer as a constraint. When this option is enabled, the topology induced by the leaves that are present in both the constraint tree and the new tree (identified by their labels) is forced to be compatible with the constraint tree. The constraint tree can be multifurcating; when this happens, all topologies compatible with the multifurcation are allowed (e.g., both `(((A,B),C),D);` and `(((A,C),B),D);` are compatible with `((A,B,C),D);`). The constraint tree can be rooted or unrooted. When a constrained tree is created, the constraint is highlighted in the resulting plot.

Under the PDA model, all labelled tree topologies have the same probability of being generated; this is not true under the other models. Note that applying a constraint will change the tree probabilities for all models except the PDA model (this is because the constraint is applied at each step during the tree growth). This means that if tree A has probability $\mathbb{P}_A$ in the unconstrained model and probability $\mathbb{P}^\star_A$ in the constrained model and, and tree B has probabilities $\mathbb{P}_B$ and $\mathbb{P}^\star_B$, then it is not true that $\frac{\mathbb{P}_A}{\mathbb{P}_B} = \frac{\mathbb{P}^\star_A}{\mathbb{P}^\star_B}$.

When creating a tree using the birth-death process, if the death rate is too high compared with the birth rate, the simulation may get "stuck", as new lineages keep being added and removed from the tree. If this happens, the `Cancel` button can be used to stop the simulation.

Neighbor-joining/UPGMA trees

The neighbor-joining and UPGMA methods both use a distance matrix to estimate a phylogenetic tree. The distance matrix can either be loaded from a file in PHYLIP format, or it can be computed by TreeViewer starting from a sequence alignment.

If a sequence alignment is provided, an evolutionary model must be chosen to determine the distance between each pair of sequences. The options are:

- The Hamming distance, in which the distance between two sequences is directly proportional to the number of differences between them, ignoring multiple substitutions at the same site.
- The Jukes-Cantor model, in which multiple substitutions are accounted for, and all changes are equally probable (both for proteins and for DNA).
- For DNA, the Kimura 1980 model, in which transitions and transversions happen with different rates, but all nucleotides have the same equilibrium frequency.
- For proteins, the Kimura 1983 model, which estimates PAM distances based on the percentage of differing amino acids between the sequences.
- For DNA, the GTR model, in which every state change has a different rate and nucleotides have different equilibrium frequencies (note that this model is much slower then the others).
- For proteins, the Scoredist method applied to the BLOSUM62 scoring matrix (in which amino acid substitutions are given a certain score based on their frequency).

In this case, bootstrapping can be used to estimate support values for the branches of the tree.

In principle, the neighbor-joining method can produce branches with negative lengths; an option is provided to circumvent this.

With both methods, it is possible to constrain the resulting tree to follow a fixed topology; this works similarly to constraints applied to random trees. Note that applying a constraint will significantly slow down the tree estimation.