Project: Phylogeny inference using minimum spanning trees

Prabhav Kalaghatgi, Persia Akbari Omgba, Jose Dasari

January 10, 2022

Due date: Feb 15 2022, 2 pm

Overview

Minimum spanning trees (MSTs) computed using additive distances share a topological correspondence with phylogenetic trees [1,2]. The topological correspondence states that internal edges of MSTs correspond to splits in phylogenetic trees. Consequently, it is possible to constrain search through tree-space using MSTs.

Your project involves designing a method for inferring phylogenetic trees by using MSTs to constrain search through tree-space. There are three main parts to your method: (i) computing an MST using distances between sequences, (ii) partitioning the total set of sequences into mutually disjoint sets by deleting edges of MST; the resulting sets are referred to hereafter as local sets, (iii) inferring a local tree for each local set, and (iv) merging local trees using the reconnection step of tree bisection and reconnection moves. You will validate your method on simulated data. The main steps of the project are described in further detail below.

Simulated data

Download trees and model parameters from the RAxML grove database [3] (`https://github.com/angtft/RAxMLGrove`). Select 100 pairs of trees and models from the database. Some of the models are multi-partition, i.e., distinct models were fit to different parts of genomes. If this is the case then select model parameters from one of the partitions. Additionally, ignore rate heterogeneity. Simulate sequence evolution using seq-gen [4] for each pair of model and tree. Vary sequence length from 10^2 bp to 10^5 bp in a reasonable number of steps.

Constructing a minimum spanning tree (MST)

Let S denote a set of sequences simulated for a pair of tree and model. Compute Jukes-Cantor distances for each pair of sequences in S. Construct a complete graph G such that each vertex in G corresponds to a sequence in S. Weight the edges of G with Jukes-Cantor distances. Compute an MST M of G using functions available in scipy.

Partitioning the complete set S using edges of the MST

Select internal edges E_d of the MST M. Removing edges E_d from M will create a disconnected graph F. Note that F is a forest because each component of F is a tree. Let s_i denote the vertex set of component i of the F. Let the total number of components be n. Let $S_M = \{s_1, s_2, \ldots, s_n\}$ denote the superset comprising the vertex set of each component of F. Infer a local tree t_i for each set s_i in S_M using the procedure described below.

The selection of internal edges is an important parameter of your method. The larger the number of edges you remove, the greater is the extent to which you are constraining the search through tree-space. On the flip side this may also reduce the quality of tree inference because the edges in the MST that are removed will be splits in the global phylogenetic tree.

It may be that large edges are more likely to induce reliable splits than short edges. If a dataset is easy (as seems to be the case for some virus datasets) then it may be that a large number of edges can be removed from the MST without much loss in reconstruction accuracy. Thus, you can quantify the difficulty of tree search by assessing the

extent to which you can constrain search through tree-space without affecting the accuracy of tree reconstruction too much.

You are expected to systematically investigate the selection of edges of the MST to be deleted.

Inferring local trees

Select a criterion to optimize among the following criteria:

- Balanced minimum evolution (BME)
- Maximum parsimony (MP)
- Maximum likelihood (ML)

Construct local trees using step-wise addition. Perform tree search using nearest neighbor interchange (NNI).

Merging local trees

Note that removing an edge $\{u, v\}$ from M creates two components, one containing u and the other containing v. It follows that there is a unique pair of vertex sets s_i and s_j in S_M for each edge $\{u, v\}$ in E_d such that s_i contains u and s_j contains v. Let t_i and t_j denote the local trees inferred using sequences in sets s_i and s_j respectively.

Merge t_i and t_j using the reconnection step of a tree bisection and reconnection move. Let n_i and n_j be the number of branches in t_i and t_j respectively. There are $n_i \times n_j$ possible ways to merge t_i and t_j. Each merger will create a distinct topology. Select the topology that optimizes the criterion that you have selected.

Note that merging a pair of trees will reduce the total number of trees to be merged by one, because there will be a new tree — the one obtained from tree merger — that is added to the list of trees that need to be merged.

Merge trees iteratively until a global phylogenetic tree has been constructed.

If you choose to perform tree operations (step-wise addition, NNI moves and tree merger) using ML then use RAxML-NG in order to optimize branch lengths for a fixed topology and fixed rate matrix (use the rate matrix that you used for simulation).

Assessing the quality of your method

Compute the Robinson-Foulds distance (RF) between the tree used for simulation T and the tree estimated by your method $\hat{T}$. How does RF distance change depending on edges selected in E_d? How does RF distance change depending on the length of the simulated sequences? Characterize the branches in T that are present and absent in $\hat{T}$, respectively, along the following lines

- Branch length
- Branch depth

The depth of a branch $\{u, v\}$ is defined as the smaller of the two distances: the distance from u to the leaf that is closest to u, and the distance from v to the leaf that is closest to v.

To deliver

You must submit your colab notebook and all other files that are necessary to reproduce your results in the Google Classroom by February 15 2022 at 2pm. You must also present your method, its results and your observations during the lecture on February 22. The presentation should take around 25-30 minutes. Make sure that the time is divided equally among all group members. Your report is due on March 15 2022 at 2 pm.

Your report should contain following points:

1. Introduction
2. Methods:
   - Explain your method in detail
   - What software and what tools did you use?
   - Was was the rationale behind your decisions?
3. Materials:
   - Characterize the trees and models that you selected for simulated data
     - How many taxa/leaves do the trees contain?
     - Which models were selected?
     - What is average branch length?
4. Results:
   - Report the performance of your method on simulated data
   - How does the selection of edges to delete in the MST affect the performance of your method?
5. Discussion
   - Why would branch length or branch depth impact reconstruction accuracy?
   - Why would sequence length impact reconstruction accuracy?
   - How does removing edges in the MST constrain search through tree-space?

The final report should have 6 to 9 pages.

References

[1] Choi, M.J., and Tan, V.Y.F., Anandkumar A., and Willsky, A.S., "Learning latent tree graphical models." *Journal of Machine Learning Research*, pp. 1771–1812,2011.

[2] Kalaghatgi, P., and Lengauer, T., "Computing phylogenetic trees using topologically related minimum spanning trees." *Journal of Graph Algorithms and Applications*, pp. 1003–1025,2017.

[3] Höhler, D., Pfeiffer, W., Ioannidis, V., Stockinger, H., and Stamatakis, A., "RAxML Grove: an empirical phylogenetic tree database" *Bioinformatics*, 2021.

[4] Rambaut, A. and Grassly, N.C., "Seq-Gen: an application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees." *Computer Application in the Biosciences*, pp. 235-238,1997.