Comparative Genomics 2018

Practical 4: Phylogenomics

Group number: 6

Group members: Kyle Kimler, Kajetan Juszczak

**Summary**

The field of phylogenomics was born with the same technology that gave birth to genomics. Phylogenomics utilizes full genome searches to understand overall systemic differences between species. In this practical, entire genomes of prokaryotes are aligned to attempt to find and understand distantly orthologous genes. With this information we can better understand which pieces of a gene effect the different physiologies in the speciation of organisms. This can be done by the usual blastp, searching each genome against one another and building lists of aligned proteins, and then building trees in a variety of ways. In this practical we have created 2 different phylogenomic trees, one where we have built a “metagene” from the first 10 blastp outputs, a concatenation of multiple genes' multiple alignment files for use as a minimal aligned-gene-only dataset for phylogenetic tree building. This tree was analysed by bootstrapping, to understand its quality. Since bootstrapping randomly samples bits in the metagene, we also created a consensus tree to put more weight on trees built by individual gene alignments. By combining trees built from each ortholog, a consensus tree was built with arms’ distance calculated by frequency of their presence in each cluster tree.

**Key Questions to Answer**

**Ex. 2**

**2.1.2**: tree is attached - tree\_meta: no outgroup, instead 09/51 and 03/20 split from the LCA.

**2.1.3**: We used -b 100 in original tree\_meta build, and it gave this unexpected result as above. With -b 1000 we had a very similar result, bootstrapped splits ~70/95%

**Ex. 3**

**3.1.2**: Comparing each of these ortholog-cluster trees to our meta-tree, we found great variability - proteins 4,5, and 6 show 51 as outgroup - protein 5 shows genome.03 as most recently divergent from genome.09. We think the major problem with these trees is the fact that the clusters were chosen based on alignment of ORFs that shared very little homology - only around 40%. Although that's above the twilight zone of sequence preservation, upon closer inspection the proteins that were clustered are often only "hypothetical" proteins, even in the well known E. coli, which from our understanding means that we probably can't use domain comparisons to understand distant paralogy/orthology - and any sequence similarity in sequence between shared domains among the clusters may only be superficial, leading to incorrectly split topologies in our trees. Since bootstrapping got 70%/95% of the same splits, the bootstrap-informed tree matched with a tree created without bootstrapping. The tree without bootstrapping just misses the extra information showing that 09/51 aren't closest related in 30% of randomly sampled datasets. This is surprisingly low given the distance of sequence similarity in our clusters. Maybe since none of the clustered sequences are really orthologous/homologous, this is giving us more information about total genome GC content or total nucleotide content than anything else.

**3.1.3**: As written in the previous explanation, a quick blast/uniprot search of our clustered ORFs shows that similarity appears to be only superficial - amylases are grouped with hypothetical lipoproteins, or in the case of cluster 7, the tree shows (((03,09),51),20), virulence factor clustered with 2 hypothetical proteins and a 50S ribosomal protein. From cluster 5 a tree was built that matches the meta-tree.

**3.2.1.3**: In Phylip consense you can choose which species to set as the outgroup, and the default is species 1. You also can choose whether you want your tree to be rooted at all. Consensus tree programs in general are built by combining many trees and finding where each group would fall in that topology. A general consensus is then taken according to whatever rule you choose on the program - default being majority rule, while a strict rule may leave species out altogether. Branch/split lengths are then calculated according to frequency of their appearance during input. Consensus trees are similar to bootstrapping trees in that they are a combination of multiple, but consensus in this case puts more weight on trees built from each ORF instead of random sample selection.

**All trees, alignments, and clusters are tarballed with the report. Thank you!**

**Reference**

<http://biopython.org/DIST/docs/api/Bio.Blast.Record-pysrc.html#HSP>

<https://blast.ncbi.nlm.nih.gov>

Zvelebil and Baum – “Understanding Bioinformatics”

<http://etetoolkit.org/treeview/>

<http://evolution.genetics.washington.edu/phylip/doc/consense.html>