6.047/6.878 Lecture 21: Phylogenomics II

Guest Lecture by
Matt Rasmussen
Scribed by Jerry Wang and Dhruv Garg

November 13, 2012

Contents

List of Figures

1 Introduction

Guest lecturer Matt Rasmussen, a former student of Manoliss presented our secondphylogenomics lecture. The lecture finished explaining max likelihood methods for phylogenetics,and then progressed to more advanced uses of phylogenetics such as inferring orthologs, paralogs, gene duplication and gene loss. This led to learning across gene trees and modeling populations and allele frequencies.

In previous lectures, we studied various algorithms to obtain phylogenetic species trees. Similar studies can be performed to study phylogeny of gene families, or sets of orthologous and paralogous genes. Given multiply aligned sequences, several techniques discussed in previous lectures could be employed for constructing a gene tree, including nearest neighbor joining, and hierarchical clustering. If in addition to the aligned genes, we also have a species tree (which can often be taken as a given for sufficiently diverged species), then we should be able to formulate a consistent view of the evolutionary process; namely, we hope to map the gene tree onto the species tree. These mappings between the two trees are called reconciliations. The standard phylogenomic pipeline can be summarized as follows:

1. Blast protein sequences against each other to score similarities.
2. Use this metric to cluster genes into families of relatedness.
3. Build multiple alignments.
4. From the alignments, build gene trees.
5. Reconcile the gene tree to the species tree.

1.1 Phylogenetics

The two main pipe lines for building trees are distance-based and character-based. Last lecture focused on distance-based pipeline. In distance-based, you form a pair-wise distance matrix using Jukes-Cantor or Kimura. Then use Neighbor Joining or UPGMA to reconstruct a tree from the matrix. Distance based pipelines use a fixed number of steps so and UPGMA and NJ have a running time is $O(n^3)$. However, there are flaws to this approach. Distance-based metrics are overly simplified and under measure the rate of mutation because a nucleotide that gets mutated back to its original form is counted as not mutated.

Todays lecture focuses on the character-based pipeline, which is NP Hard so we have to resort to heuristics. The basic idea is we want to search through different trees and test each one. We start with an initial tree, then compute the probability/likelihood, then explore the tree space using methods such as nearest neighbor interchange (NNI), compute the score again, loop and then return the tree with the highest score as the answer. Using NNI, we can go to all trees in the tree space. The problem is that the tree space is very big (why it is NP hard).

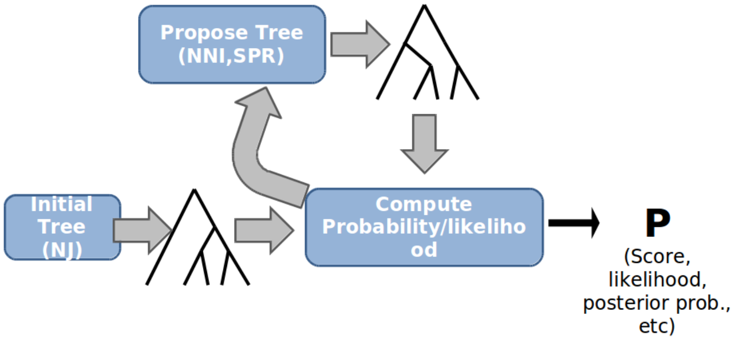

Propose Tree (NNI,SPR)
Initial Tree (NJ)
Compute Probability/likelihood
P
(Score, likelihood, posterior prob., etc)

Figure 1: Heuristic tree search in character-based reconstruction

For the scoring metric, we want to maximize

$$\hat{T}, \hat{\boldsymbol{t}} = \underset{T, \boldsymbol{t}}{\operatorname{argmax}} P(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n | T, \boldsymbol{t})$$

- Where
  - **X** = matrix of sequences, x_{ij} = j^{th} site in i^{th} sequence $\mathbf{x}_j$
  - T = tree topology
  - **t** = branch lengths

Figure 2: Scoring metric for heuristic tree search

Using the Felsenstein peeling algorithm, we can efficiently compute $P(X|T, B)$ by buildign up a dynamic programming problem. We can first look at site evolution along a single brance, then build on that and look at sequence evolution and then look at site evolution along an entire tree. Site evolution uses the Jukes-Cantor model and has the definition

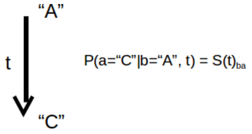


Figure 3: Site evolution uses the Jukes-Cantor model

If we assume site independence, sequence evolution is just the product of site evolution:
Assuming site independence does not always hold for example in RNA coding r gions the sites is not independent due to RNA folding. To move to sequence evolution over an entire tree, we assume branch independence once we condition on the parent sequence.

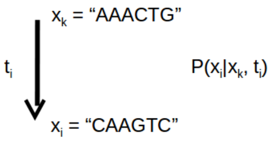


Figure 4: Sequence evolution is the product of site evolution

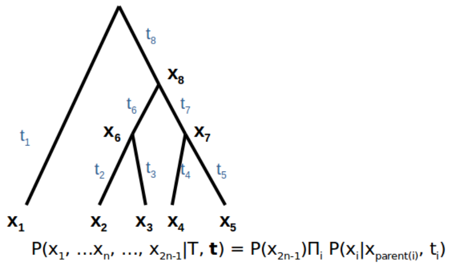

t_8
x_8
t_6
t_7
t_1
x_6
x_7
t_2
t_3
t_4
t_5
x_1
x_2
x_3
x_4
x_5
P(x_1, ...x_n, ..., x_{2n-1}|T, t) = P(x_{2n-1})Π_i P(x_i|x_{parent(i)}, t_i)

Figure 5: Sequence evolution over an entire tree.

From the equation in sequence evolution over an entire tree, we need both internal nodes and leaves $(x_1, ...x_{2n-1})$ but only leaves $(x_1, ...x_n)$ are given so we need to marginalize over unknowns $(x_{n+1}, ..., x_{2n-1})$. Using a factorization trick:

$P(x_1, x_2, x_3, x_4|T, t) = \Sigma x_5 \Sigma x_6 \Sigma x_7 P(x_1, x_2, x_3, x_4, x_5, x_6, x_7|T, t)$
$= \Sigma x_5 \Sigma x_6 \Sigma x_7 P(x_1|x_5, t_1)P(x_2|x_5, t_1)P(x_3|x_6, t_3)P(x_4|x_6, t_4)$
$= \Sigma x_7 P(x_7)\ [\Sigma x_5 P(x_5|x_7, t_5)P(x_1|x_5, t_1)P(x_2|x_5, t_1)]\ [\Sigma x_6 P(x_6|x_7, t_6)P(x_3|x_6, t_3)P(x_4|x_6, t_4)]$

The Peeling algorithm builds a DP table. Each entry contains the probability of seeing the leaf data below node I, give that node I has base a at site j. The leaves of the table are initialized based on the observed sequence. Entries are populated in post-order traversal. The runtime of the Peeling algorithm is $O(nmk^2)$.

$$P(x_1, ..., x_n|T, t) = \prod_j \sum_a L_{2n-1,j,a} P(a)$$

$$L_{i,j,a} = \begin{cases} 1 & \text{if } x_{i,j} = a, i \leq n \\ 0 & \text{if } x_{i,j} \neq a, i \leq n \\ \sum_{b,c} P(b|a, t_{left(i)}) L_{left(i),j,b} & \text{if } i > n \\ P(c|a, t_{right(i)}) L_{right(i),j,c} & \end{cases}$$

Figure 6: Peeling Algorithm

The Peeling algorithm scores one tree and we need to use the search algorithm to search for more trees. The runtime is for one tree while the entire runtime depends on how many trees you want to look at.

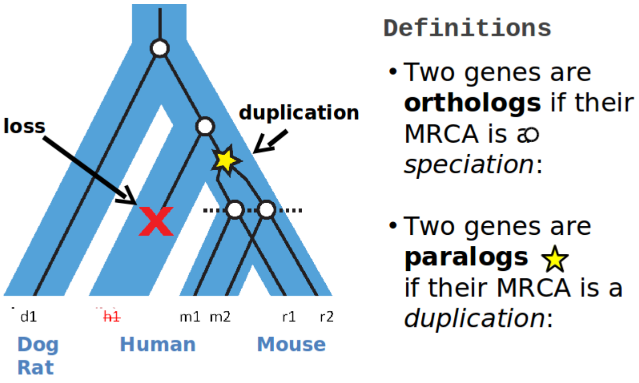
2 Inferring Orthologs/Paralogs, Gene Duplication and Loss

There are two commonly used trees. The species tree uses morphological characters, fossil evidence, etc to create a tree of how species are related (leaves are species). Gene trees look at specific genes in different species (leaves are genes).

Reconciliation is an algorithm to figure out how the gene tree fits inside he species tree. It maps the vertices in the gene tree to vertices in the species tree.

We want to minimize the duplication/loss so we want to map events as low in the tree as possible to when they happened to minimize loss.

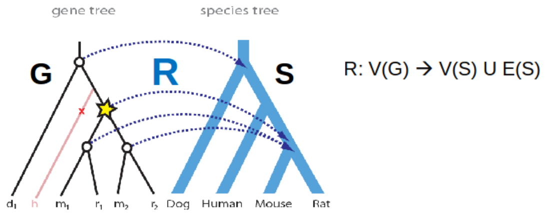
Duplication events map to the same as both of its children. Loss event maps to gap in the mapping. Gene tree accuracy is important; even one branch misplaced can dramatically increases error.



Definitions

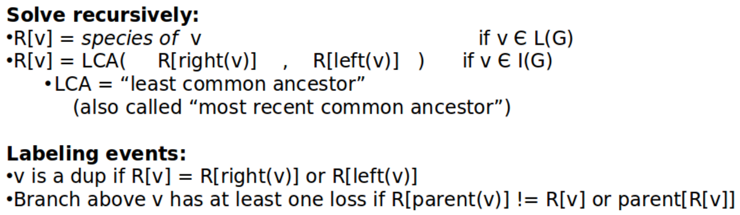
- Two genes are **orthologs** if their MRCA is a *speciation*:
- Two genes are **paralogs** ☆ if their MRCA is a *duplication*:

Figure 7: Gene Family Evolution: Gene Trees and Species Trees



R: V(G) → V(S) U E(S)

Figure 8: Maximum Parsimony Reconciliation (MPR)

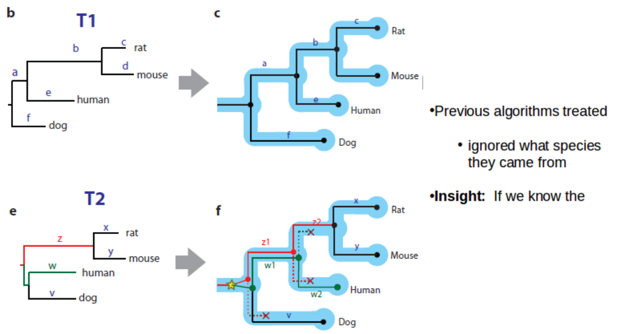


Solve recursively:
•R[v] = *species of* v if v ∈ L(G)
•R[v] = LCA(R[right(v)] , R[left(v)]) if v ∈ I(G)
•LCA = “least common ancestor”
(also called “most recent common ancestor”)

Labeling events:
•v is a dup if R[v] = R[right(v)] or R[left(v)]
•Branch above v has at least one loss if R[parent(v)] != R[v] or parent[R[v]]

Figure 9: Maximum Parsimony Reconciliation Recursive Algorithm

3 Learning Across Gene Trees



•Previous algorithms treated

- ignored what species they came from

•**Insight:** If we know the

Figure 10: Using species trees to improve gene tree reconstruction.

If we knew the species tree we could know beforehand that we expect the branch to be longer. We can develop a model for what kind of branch lengths we can expect. We can use conserved gene order to tell orthologs and build trees.

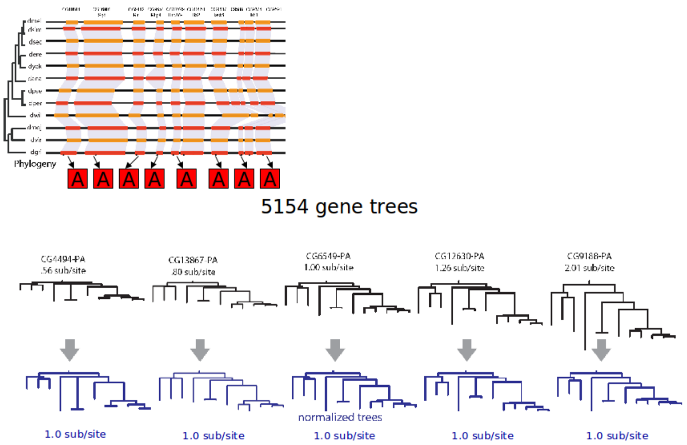


Figure 11: We can develop a model for what kind of branch lengths we can expect. We can use conserved gene order to tell orthologs and build trees.

When gene is fast evolving in one species, it is fast evolving in all species. We can model a branch length as two different rate components. One is gene specific(present across all species) and a species specific which is customized to a specific species.

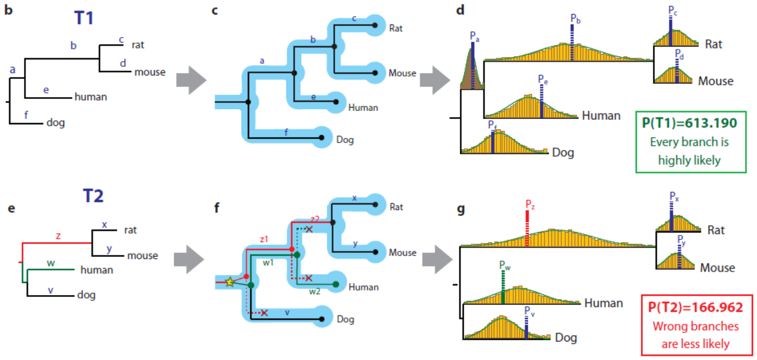


Figure 12: Branch length can be modeled as two different rate components: gene specific and species specific.

This method greatly improves reconstruction accuracy.

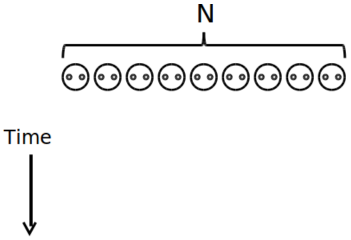
4 Modeling Population and Allele Frequencies

People keep sequencing genomes so looking at how populations evolve is becoming more and more important and feasible. The Wright-fisher model is used to study drifts, bottlenecks, etc. The coalescent model combines the Wright-fisher with trees. Wright-fisher was designed to study the effect of finite population sizes. We need to assume population size is fixed at N, random mating, non-overlapping generations.

Continue for many generations and ignore ordering of chromosomes.

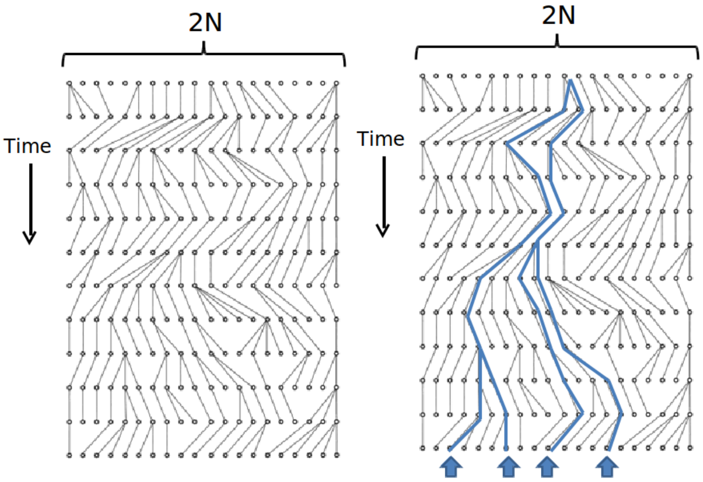
The coalescent model only focuses on the genealogy. It only is concerned about the lineages we have sequences for; do not have to worry about others. It is a probabilistic model that works backwards in time to find when they have common ancestors.

Say we have $2N$ individuals, what is the probability that k lineages do not have any coalescent events in parental generation? What is the probability that the first coalescent of k lineages is at t generations? This process can be seen as a geometric distribution.



- Developed to model the **stochastic** behavior of population structure (Wright 1931)
- Use it to study
  - Effect of drift on allele frequency
  - Migration
  - Population bottle necks
  - Selection
  - age of MRCA
  - Etc.

Figure 13: The Wright-Fisher model



- Track lineages
- Genealogies
  - Randomly sample extant chromosomes
  - Trace back tree until all coalescence

Felsenstein 2004.

Figure 14: The Wright-Fisher model continued over many generations and ignoring the ordering of chromosomes.

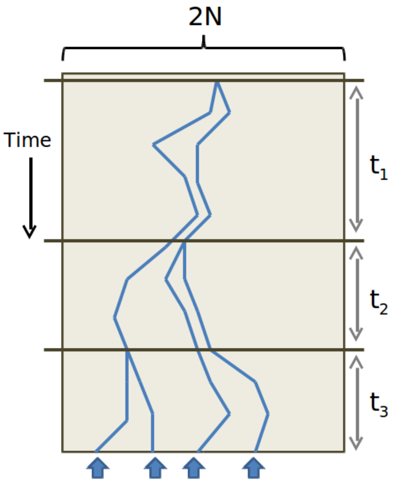
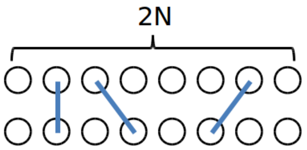


Figure 15: The coalescent model.



$$\frac{(2N-1)}{2N}\frac{(2N-2)}{2N}\cdots\frac{(2N-k+1)}{2N}=\prod_{i=1}^{k-1}\left(1-\frac{i}{2N}\right)$$

$$=1-\sum_{i=1}^{k-1}\frac{j}{2N}+O\left(\frac{1}{N^2}\right)=\boxed{1-\binom{k}{2}\frac{1}{2N}+O\left(\frac{1}{N^2}\right),}$$

For k<<N, O(1/N^2) is very small

Figure 16: Geometric probability distribution for coalescent events in k lineages.

Can repeat to find when all individuals coalesce. Each branch of species tree can be seen as having its own Wright-Fisher inside of it.

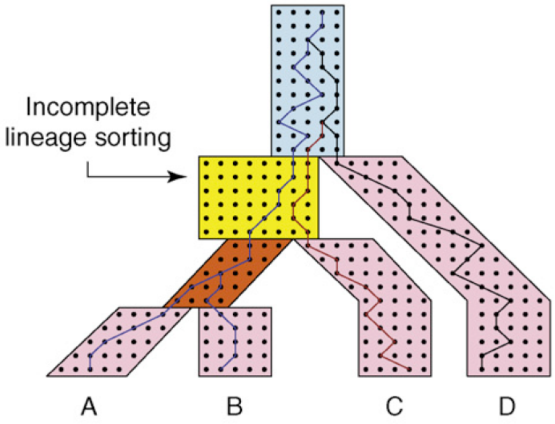


Figure 17: Multispecies Coalescent Model. Leaf branches track one lineage. There is a lag time from when population separated and when two actual gene lineages find a common ancestor. The rate of coalescent slows down as N gets bigger and for short branches. Deep coalescent is depicted in light blue for three lineages. The species and gene tree are incongruent since C and D are sisters in gene tree but not the species tree. There is a $\frac{2}{3}$ chance that incongruence will occur because once we get to the light blue section, the Wright-fisher is memory less and there is only $\frac{1}{3}$ chance that it will be congruent. Effect of incongruence is called incomplete lineage sorting.

5 SPIDIR:Background

As presented in the supplementary information for SPIDIR, a gene family is the set of genes that are descendents of a single gene in the most recent common ancestor (MRCA) of all species under consideration. Furthermore, genetic sequences undergo evolution at multiple scales, namely at the level of base pairs, and at the level of genes. In the context of this lecture, two genes are orthologs if their MRCA is a speciation event; two genes are paralogs if their MRCA is a duplication event.

In the genomic era, the species of a modern genes is often known; ancestral genes can be inferred by reconciling gene- and species-trees. A reconciliation maps every gene-tree node to a species-tree node. A common technique is to perform Maximum Parsimony Reconciliation (MPR), which finds the reconciliation R implying the fewest number of duplications or losses using the recursion over inner nodes v of a gene tree G. MPR fist maps each leaf of the gene tree to the corresponding species leaf of the species tree. Then the internal nodes of G are mapped recursively:

$R(v) = MRCA(R(right(v)), R(left(v)))$

If a speciation event and its ancestral node are mapped to the same node on the species tree. Then the ancestral node must be an duplication event.

Using MPR, the accuracy of the gene tree is crucial. Suboptimal gene trees may lead to an excess of loss and duplication events. For example, if just one branch is misplaced (as in 2) then reconciliation infers 3 losses and 1 duplication event. In [6], the authors show that the contemporaneous current gene tree methods perform poorly (60% accuracy) on single genes. But if we have longer concatenated genes, then accuracy may go up towards 100%. Furthermore, very quickly or slowly evolving genes carry less information as compared with moderately diverging sequences (40-50% sequence identity), and perform correspondingly worse. As corroborated by simulations, single genes lack sufficient information to reproduce the correct species tree. Average genes are too short and contains too few phylogenetically informative characters. While many early gene tree construction algorithms ignored species information, algorithms like SPIDIR capitalize on the insight that the species tree can provide additional information which can be leveraged for gene tree construction. Synteny can be used to independently test the relative accuracy of different gene tree reconstructions. This is because syntenic blocks are regions of the genome where recently diverged organisms have the same gene order, and contain much more information than single genes.

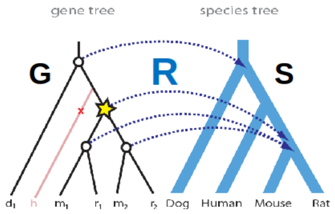


Figure 18: MPR reconciliation of genes and species tree.

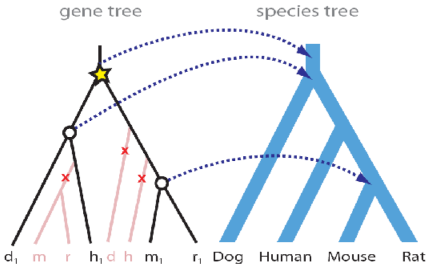


Figure 19: Inaccuracies in gene tree.

There have been a number of recent phylogenomic algorithms including: RIO [2], which uses neighbor joining (NJ) and bootstrapping to deal with incogruencies, Orthostrapper [7], which uses NJ and reconciles to a vague species tree, TreeFAM [3], which uses human curation of gene trees as well as many others. A number of algorithms take a more similar track to SPIDIR [6], including [4], a probabilistic reconciliation algorithm [8], a Bayesian method with a clock,[9],and parsimony method using species tree , as well as more recent developments: [1] a Bayesian method with relaxed clock and [5], a Bayesian method with gene and species specific relaxed rates (an extension to SPIDIR) .

6 SPIDIR: Method and Model

SPIDIR exemplifies an iterative algorithm for gene tree construction using the species tree. In SPIDIR, the authors define a generative model for gene-tree evolution. This consists of a prior for gene-tree topology and branch lengths. SPIDIR uses a birth and death process to model duplications and losses (which informs the prior on topology) and then then learns gene-specific and species-specific substitution rates (which inform the prior on branch lengths). SPIDIR is a *Maximum a posteriori (MAP)* method, and, as such, enjoys several nice optimality criteria.

In terms of the estimation problem, the full SPIDIR model appears as follows:
$argmaxL, T, RP(L, T, R|D, S, \Theta) = argmaxL, T, RP(D|T, L)P(L|T, R, S, \Theta)P(T, R|S, \Theta)$

The parameters in the above equation are: D = alignment data , L = branch length T = gene tree topology , R = reconciliation , S = species tree (expressed in times) , Θ = (gene and species specific parameters [estimated using EM training], λ, μ dup/loss parameters)). This model can be understood through the three terms in the right hand expression, namely:

1. the sequence model– $P(D|T, L)$. The authors used the common HKY model for sequence substitutions, which unifies Kimura's two parameter model for transitions and transversions with Felsenstein's model where substitution rate depends upon nucleotide equilibrium frequency.

2. the first prior term, for the rates model– $P(L|T, R, S, \Theta)$, which the authors compute numerically after learning species and gene specific rates.

3. the second prior term, for the duplication/loss model– $P(T, R|S, \Theta)$, which the authors describe using a birth and death process.

Having a rates model is very rates model very useful, since mutation rates are quite variable across genes. In the lecture, we saw how rates were well described by a decomposition into gene and species specific rates. In lecture we saw that an inverse gamma distribution appears to parametrize the gene specific substitution rates, and we were told that a gamma distribution apparently captures species specific substitution rates. Accounting for gene and species specific rates allows SPIDIR to build gene trees more accurately than previous methods. A training set for learning rate parameters can be chosen from gene trees which are congruent to the species tree. An important algorithmic concern for gene tree reconstructions is devising a fast tree search method. In lecture, we saw how the tree search could be sped up by only computing the full $argmaxL, T, RP(L, T, R|D, S, \Theta)$ for trees with high prior probabilites. This is accomplished through a computational pipeline where in each iteration 100s of trees are proposed by some heuristic. The topology prior $P(T, R|D, S, \Theta)$ can be computed quickly. This is used as a filter where only the topologies with high prior probabilities are selected as candidates for the full likelihood computation.
The performance of SPIDIR was tested on a real dataset of 21 fungi. SPIDER recovered over 96% of the synteny orthologs while other algorithms found less than 65%. As a result, SPIDER invoked much fewer number of duplications and losses.

7 Conclusion

Incorporating species tree information into the gene tree building process via introducing separate gene and species substitution rates allows for accurate parsimonious gene tree reconstructions. Previous gene tree reconstructions probably vastly overestimated the number of duplication and loss events. Reconstructing gene trees for large families remains a challenging problem.

8 Current Research Directions

9 Further Reading

10 Tools and Techniques

11 What Have We Learned?

References

[1] O. Akerborg, B. Sennblad, L. Arvestad, and J. Lagergren. Bayesian gene tree reconstruction and reconciliation analysis. *Proc Natl Acad Sci*, 106(14):5714–5719, Apr 2009.

[2] Zmasek C.M. and Eddy S.R. Analyzing proteomes by automated phylogenomics using resampled inference of orthologs. *BMC Bioinformatics*, 3(14), 2002.

[3] Li H, Coghlan A, Ruan J, Coin LJ, Heriche JK, Osmotherly L, Li R, Liu T, Zhang Z, Bolund L, Wong GK, Zheng W, DEhal P, Wang J, and Durbin R. Treefam: a curated database of phylogenetic trees of animal gene families. *Nucleic Acids Res*, 34, 2006.

[4] Arvestad L., Berglund A., Lagergren J., and Sennblad B. Bayesian gene/species tree reconciliation and orthology analysis using mcmc. *Bioinformatics*, 19 Suppl 1, 2003.

[5] M. D. Rasmussen and M. Kellis. A bayesian approach for fast and accurate gene tree reconstruction. *Mol Biol Evol*, 28(1):273290, Jan 2011.

[6] Matthew D. Rasmussen and Manolis Kellis. Accurate gene-tree reconstruction by learning gene and species-specific substitution rates across multiple complete genomes. *Genome Res*, 17(12):1932–1942, Dec 2007.

[7] C.E.V. Storm and E.L.L. Sonnhammer. Automated ortholog inference from phylogenetic trees and calculation of orthology reliability. *Bioinformatics*, 18(1):92–99, Jan 2002.

[8] Hollich V., Milchert L., Arvestad L., and Sonnhammer E. Assessment of protein distance measures and tree-building methods for phylogenetic tree reconstruction. *Mol Biol Evol*, 22:2257–2264, 2005.

[9] Wapinski, I. A. Pfeffer, N. Friedman, and A. Regev. Automatic genome-wide reconstruction of phylogenetic gene trees. *Bioinformatics*, 23(13):i549–i558, 2007.