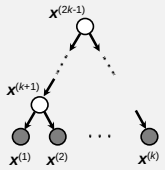
1

BTRY 4840/6840, CS 4775
Computational Genetics and Genomics



October 4, 2018

2

Announcements

- Problem set 3 due in one week
- My office hours next week: Wednesday 4:30-5:30
  - Enjoy Fall Break!
- Talk next Wednesday, October 10, 4:15pm:
  - "Likelihood Estimation of Large Species Trees Using the Coalescent Process" by Arindam RoyChoudhury
  - Room G01 Biotech

3

Today's lecture

- More phylogenetics:
  - Jukes-Cantor model of sequence evolution
  - Felsenstein's algorithm

4

Recall: How do we reconstruct trees?

1. Parsimony method:
   - For a given tree, finds the ancestral sequence that results in the fewest changes over full tree
   - To reconstruct tree: compute parsimony for all possible trees, choose tree that results from fewest changes overall
2. Heuristic approaches using distance matrices:
   - Given distances between sequences according to some metric, find tree that best approximates this matrix
3. Statistical:
   - Given a sequence evolution model, perform maximum likelihood or Bayesian inference

5

Tree reconstruction method:

Statistical

6

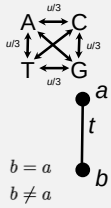
Tree reconstruction method 3: Statistical

- Goal: calculate likelihood of tree for given set of sequences
- How should we do this?
  - Must consider all possible ancestral sequences at unobserved (internal) nodes
    - Felsenstein's algorithm (similar to Sankoff's algorithm)
  - Need to define the probability of a given assignment of ancestral sequences: a sequence evolution model
    - Sequence evolution model gives probability of bases mutating along each branch
    - We use the Jukes-Cantor model – an extremely simple model – but others exist (e.g., Kimura)

Overview: Jukes-Cantor model of sequence evolution

7

- Continuous time Markov model of sequence evolution
- Equilibrium frequency of bases assumed identical
  $$\pi_A = \pi_C = \pi_G = \pi_T = 1/4$$
  - What is equilibrium frequency? Also "stationary distribution"
    - Frequency of states after running model to convergence (long)
- Mutations modeled as Poisson process
  - Mutation rate: $4u/3$ – can mutate to any base (even same; self-loops not shown)
  $$P(0 \text{ mutations}|t,u) = e^{-4ut/3}$$
  $$P(>0 \text{ mutations}|t,u) = 1 - e^{-4ut/3}$$



- Probability of base b at time t starting from a:

$$P(b|a,t) = \begin{cases} e^{-4ut/3} + \frac{1}{4}(1-e^{-4ut/3}) = \frac{1}{4}(1+3e^{-4ut/3}) & b = a \\ \frac{1}{4}(1-e^{-4ut/3}) & b \neq a \end{cases}$$

Can calculate likelihoods for pair of sequences from evolutionary model

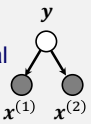
8

- Given (gapless) alignment $\boldsymbol{X}$ of $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}$

$\boldsymbol{X}_i$

$\boldsymbol{x}^{(1)}$ = AATCGGTACGA...
$\boldsymbol{x}^{(2)}$ = ATTCAGCACGT...

- The $\boldsymbol{x}$'s descend from unobserved ancestral sequence $\boldsymbol{y}$



- First, assume columns are independent:
  - Let $\boldsymbol{\Theta}$ be evolutionary model parameters (u in Jukes-Cantor)

$$P(\boldsymbol{X}|\boldsymbol{\Theta}, t, \boldsymbol{\pi}) = \prod_{i=1}^{L} P(\boldsymbol{X}_i|\boldsymbol{\Theta}, t, \boldsymbol{\pi}) = \prod_{i=1}^{L} \sum_{y_i} P(x_i^{(1)}, x_i^{(2)}, y_i|\boldsymbol{\Theta}, t, \boldsymbol{\pi})$$

- Because $x_i^{(1)}$ and $x_i^{(2)}$ conditionally independent:

$$P\left(x_i^{(1)}, x_i^{(2)}, y_i \middle| \boldsymbol{\Theta}, t, \boldsymbol{\pi}\right) = P(y_i) P\left(x_i^{(1)} \middle| y_i, \boldsymbol{\Theta}, t\right) P\left(x_i^{(2)} \middle| y_i, \boldsymbol{\Theta}, t\right)$$

(Assuming $\boldsymbol{y}$ has stationary distribution of bases)

$$= \pi_{y_i} P\left(x_i^{(1)} \middle| y_i, \boldsymbol{\Theta}, t\right) P\left(x_i^{(2)} \middle| y_i, \boldsymbol{\Theta}, t\right)$$

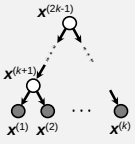
Calculating likelihoods for multiple sequences

9

- Now for $\boldsymbol{X}$ a multiple sequence alignment related by a given phylogeny

$\boldsymbol{X}_i$

$\boldsymbol{x}^{(1)}$ = AATCGGTACGA...
$\boldsymbol{x}^{(2)}$ = ATTCAGCACGT...
⋮
$\boldsymbol{x}^{(k)}$ = GTTGACTATGA...



$x^{(1)}, \ldots x^{(k)}$ are leaves, others unobserved ancestral sequences

- What is $P\left(x_i^{(1)}, \ldots, x_i^{(2k-1)}\right)$? Product over branches:

$$P\left(x_i^{(1)}, \ldots, x_i^{(2k-1)}\right) = \pi_{x_i^{(2k-1)}} \prod_{j=1}^{2k-2} P\left(x_i^{(j)} \mid x_i^{\text{parent}(j)}, t_j\right)$$

- Thus, for observed data:

$$P\left(x_i^{(1)}, \ldots, x_i^{(k)}\right) = \sum_{x_i^{(k+1)}, \ldots, x_i^{(2k-1)}} P\left(x_i^{(1)}, \ldots, x_i^{(2k-1)}\right)$$

Recall: Sankoff's algorithm

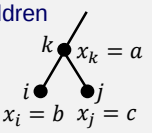
10

- Let x_k be the base at node k at a given column c
- Let $S_k(a)$ = minimum # changes beneath k if $x_k = a$
- Base case – k a leaf node

$$S_k(a) = \begin{cases} 0 & x_k = a \\ \infty & \text{otherwise} \end{cases}$$

- Recurrence – k ancestor node, i, j children

$$S_k(a) = \min_b (S_i(b) + w(a \to b)) + \min_c (S_j(c) + w(a \to c))$$



- Termination: $S_T(X_c) = \min_a S_{\text{root}}(a)$

Felsenstein's algorithm: computing likelihood

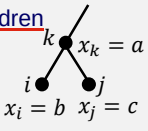
11

- Let $P(x^{(\underline{k})}|x^{(k)} = a)$ be the probability of the observed bases at or below node k given $x^{(k)} = a$
- Base case – k a leaf node

$$P(x^{(\underline{k})}|x^{(k)} = a) = \begin{cases} 1 & x^{(k)} = a \\ 0 & \text{otherwise} \end{cases}$$

- Recurrence – k ancestor node, i, j children

$$P(x^{(\underline{k})}|x^{(k)} = a) = \sum_b P(x^{(\underline{i})}|x^{(i)} = b) P(b|a, t_i) \times \sum_c P(x^{(\underline{j})}|x^{(j)} = c) P(c|a, t_j)$$



- Termination: $P(x^{(1)}, \ldots, x^{(k)}) = \sum_a \pi_a P(x^{(\underline{2k-1})}|x^{(2k-1)} = a)$

Example likelihood calculation

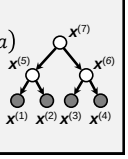
12

- Example – identical form applies to all internal nodes:

$$\begin{aligned} P(x^{(\underline{7})}|x^{(7)} = a) &= P(x^{(\underline{5})}, x^{(\underline{6})}|x^{(7)} = a) \\ &= P(x^{(\underline{5})}|x^{(7)} = a) P(x^{(\underline{6})}|x^{(7)} = a) \\ &= \sum_{b \in \{A,C,G,T\}} P(x^{(\underline{5})}|x^{(5)} = b) P(x^{(5)} = b|x^{(7)} = a, t_5) \\ &\times \sum_{c \in \{A,C,G,T\}} P(x^{(\underline{6})}|x^{(6)} = c) P(x^{(6)} = c|x^{(7)} = a, t_6) \end{aligned}$$

- Without the underscore notation (i.e., $x^{(7)}$):

$$\begin{aligned} &P(x^{(1)}, x^{(2)}, x^{(3)}, x^{(4)}|x^{(7)} = a) \\ &= P(x^{(1)}, x^{(2)}|x^{(7)} = a) P(x^{(3)}, x^{(4)}|x^{(7)} = a) \\ &= \sum_b P(x^{(1)}, x^{(2)}|x^{(5)} = b) P(b|a, t_5) \\ &\times \sum_c P(x^{(3)}, x^{(4)}|x^{(6)} = c) P(c|a, t_6) \end{aligned}$$



What can we do with sequence evolution models, Felsenstein's algorithm?

13

- Integrate over all assignments of bases at ancestral nodes
  - Felsenstein's algorithm does this
  - Aside: what do I mean by integrate over?
    - Calculate integral/sum over all possible assignments of variable(s)
    - Application of law of total probability
- Integrate over all possible substitutions on branches
  - Relies on continuous time Markov models (e.g., Jukes-Cantor)

➢ So: can integrate over all possible substitution histories consistent with observed data Gives likelihood of a topology

- All of this very efficiently:
  - For n sequences of length L, a letters in alphabet:
  - How many nodes in tree? $2n-1$ Complexity? $O(Lna^2)$

What can we do with likelihoods of trees?

14

- Statistical framework for analyzing trees – many applications!
- Estimating parameters: informative about biology
  - Estimating branch length; transition-transversion ratio; …
- Finding maximum likelihood tree [next few slides]
- Compare molecular evolution models
- Estimating ancestral sequences (posterior distribution) [upcoming slide]
- Bayesian inference of parameters of interest
- Hypothesis testing (e.g., to identify selected loci)

Finding maximum likelihood tree

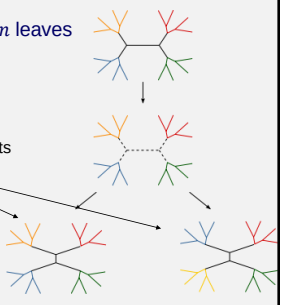
15

- As before, to find best tree, must consider all possibilities
  - Huge number of trees for moderate number of sequences, so intractable
- Can instead use divide and conquer heuristics
  - Can optimize subtrees and then piece together via heuristics
  - More practical because subtree search space is much smaller
  - Not guaranteed to find best solution, but effective way to explore the state space

Heuristic: Nearest neighbor interchange (NNI)

16

1. Start from unrooted tree with n leaves
2. For every internal edge:
   1. Dissolve the edge and two links connected to it on each end
      - ➢ Four subtrees to be rearranged
   2. Form the two other arrangements of the subtrees
   3. Repeat (see below)

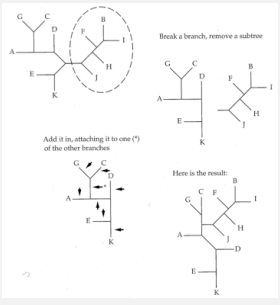


- How many internal edges?
  - $n-3$, so $2(n-3)$ arrangements (total $2n-3$ edges, n at leaves)
- Approach 1: Consider all internal edges for a single tree, choose best at end
- Approach 2: Greedy (switch instantly to improved tree)

Heuristic: subtree pruning and regrafting

17

1. Given unrooted tree with n leaves
2. For every edge:
   1. Break edge ⇒ two subtrees
   2. Reattach one subtree at every possible position in other
   3. Compute likelihood
   4. Repeat



- Examines $4(n-3)(n-2)$ trees from one start tree
  - See Felsenstein for derivation
- Again must decide on greediness
- Can save time by keeping partial likelihood of subtrees

Felsenstein (2004)

Posterior distribution of bases at root of tree

18

- Can integrate over possible assignments of bases at ancestral nodes [via Felsenstein's algorithm]
- Posterior distribution of bases at the root: [using Bayes rule]

$$P(x^{(2k-1)}=a|x^{(1)},\ldots,x^{(k)})=\frac{P(x^{(1)},\ldots,x^{(k)}|x^{(2k-1)}=a)\pi_a}{P(x^{(1)},\ldots,x^{(k)})}$$

- Felsenstein's algorithm computes both numerator and denominator!
- Statistically grounded way to find likely ancestral sequences
  - Preferred to parsimony
- Can compute for all internal nodes simultaneously with "inside/outside" algorithm (similar to forward/backward)

19

Final notes

- Summary:
  - Jukes-Cantor model of molecular evolution
    - Estimating branch lengths from divergence
  - Felsenstein's algorithm – statistics on phylogenies:
    - Many applications of likelihood calculations
    - Ability to infer posterior distribution of bases at tree nodes
  - In section tomorrow: Neighbor-Join method for tree inference
    - Widely used approach for inferring trees!