Phylogenetic Inference

distance methods
character-based methods
statistical methods

Reading:
treeterm.pdf and treebuilding.pdf (on-line)

Basic Method

- Collect sequences to be used as the basis for tree construction
- Align them with a global multiple alignment
- If using a character-based method, select sites to compare:
  - some positions will be identical in all species: no information
  - others will have too much variation: not “informative”
  - what about gaps? many methods discard sites with gaps
- Build and analyze the tree

Types of Tree-Building Algorithms

- There are three main classes of tree-building algorithms:
  - distance methods
    - use pairwise distances (e.g. edit distance) as basis for comparing taxa; similar sequences will be close in the tree
  - character-based methods
    - use the sequence data itself to form the tree
  - statistical methods
    - compute probability that various tree topologies fit the observed data
      examples: maximum likelihood and Bayesian methods

Distance Methods

- We saw two examples of distance methods (UPGMA and Neighbor Joining) in the lecture on multiple alignment
- Both start with construction of a matrix of distances:
  \[ M[i, j] = \text{distance from sequence } i \text{ to sequence } j \]
- Examples of methods for estimating distances:
  - use dynamic programming (edit distance) score
  - count number of mismatches in the alignment
Jukes-Cantor Correction

- To use distance to build a tree we need to correct for the fact that over time multiple mutations may have occurred at the same location.
- The Jukes-Cantor method assumes each type of point mutation in DNA has the same probability $\alpha$.

\[
\begin{align*}
A & \xrightarrow{\alpha} G \\
& \quad \uparrow \alpha \alpha \alpha \\
C & \xrightarrow{\alpha} T
\end{align*}
\]

Jukes-Cantor Correction (cont’d)

- At time $t = 2$, the base
  - might still be $A$,
  - might have changed to one of the other bases, or
  - might have changed to one of the other bases and then mutated back into an $A$.
- All of these cases can be expressed in terms of the probability of observing $A$ at $t = 1$:

\[
p_A(2) = (1 - 3\alpha)p_A(1) + \alpha[1 - p_A(1)]
\]

Was $A$, stayed that way
Was not $A$, changed to $A$.

Jukes-Cantor Correction (cont’d)

- In general, the probability of seeing an $A$ at some future time can be expressed in terms of seeing $A$ at the previous time:

\[
p_A(t+1) = (1 - 3\alpha)p_A(t) + \alpha[1 - p_A(t)]
\]

- Doing a little bit of algebra*:

\[
\Delta p_A = p_A(t+1) - p_A(t)
= -3\alpha p_A(t) + \alpha[1 - p_A(t)]
= -4\alpha p_A(t) + \alpha
\]

* see Li and Graur, Fundamentals of Molecular Evolution
Write the discrete equation in continuous form, integrate to get

\[ p_{ii}(t) = \frac{1}{4} + \left( \frac{3}{4} \right) e^{-4\alpha t} \]

\[ p_{ij}(t) = \frac{1}{4} - \left( \frac{3}{4} \right) e^{-4\alpha t} \]

The previous equations describe probabilities of changes in a single sequence as a function of time \( t \) and mutation probability \( \alpha \).

Next consider a situation where a sequence is copied (via duplication or speciation).

The probability of seeing an A in one sequence after time \( t \) at the same location where there was an A in the ancestor is \( p_{AA}(t) \).

The probability of observing an A at this location in both sequences, assuming they evolve independently, is \( p_{AA(t)}^2 \).

The probability that any nucleotide is unchanged is

\[ I_t = p_{AA(t)}^2 + p_{CC(t)}^2 + p_{GG(t)}^2 + p_{TT(t)}^2 \]

\[ = \frac{1}{4} + \left( \frac{3}{4} \right) e^{-8\alpha t} \]

If the probability of any one change in time \( t \) is \( \alpha t \),

- the probability of a single change in one sequence is \( 3\alpha t \)
- the probability of a change in either sequence is \( 2(3\alpha t) \)
- the equation on the previous slide was for \( 8\alpha t \)

The equation at right estimates the number of mutations that have occurred as a function of the proportion of different sites

\[ K = -\frac{3}{4} \ln(1 - \frac{4}{3} p) \]
Other Corrections

- See Li and Graur or Mount for an explanation of corrections based on
  - Kimura's two-parameter model
    (A→G and T→C have a higher
     probability than other changes)
  - corrections for coding sequences
    (silent and replacement sites)
  - corrections for protein sequences,
    based on PAM matrices
- Bottom line: adjust distances computed by alignment before using the distances to infer phylogenies…

Distance Metrics

- If the pairwise distances satisfy certain criteria, the UPGMA method can give an
  accurate reconstruction of the phylogenetic tree
- A relation is a metric if it satisfies these criteria for any set of sequences A, B, and C:
  - \( d(A,B) \geq 0 \)
  - \( d(A,B) = d(B,A) \)
  - \( d(A,B) \leq d(A,C) + d(C,B) \)
- The last criterion is the triangle inequality
  - it shouldn’t be easier to get from A to B via C than from a direct route...

Ultrametric Distances

- Adding one more constraint to the distances makes them ultrametric:
  - \( d(A,B) \leq \max(d(A,C), d(B,C)) \)
- This constraint means the two largest distances in the set \( d(A,B) \), \( d(A,C) \) and \( d(B,C) \) must be equal
  - \( d(a,c) \) and \( d(b,c) \) must also satisfy this property
  - the distances form an isosceles triangle

Here is an example of an (artificial) ultrametric distance matrix and the resulting tree
- the simple UPGMA algorithm will recreate the true tree when the data are in fact ultrametric

\begin{align*}
\begin{array}{c|c|c|c|c|c|c|c|c|c|c}
   & A & B & C & D & E \\
A & 8 & 6 & 8 & 2 \\
B & 8 & 3 & 8 &  \\
C & 8 & 6 &  \\
D & 8 &  \\
E &  \\
\end{array}
\end{align*}

\begin{align*}
\begin{array}{c|c|c|c|c|c|c|c|c|c}
   & A & E & C & D & B \\
A & 1 & 2 & 1.5 & 2.5 &  \\
E & 1 & 3 & 1.5 &  \\
C & 1.5 &  \\
D &  \\
B &  \\
\end{array}
\end{align*}

note the isosceles triangles...
Here is the complete UPGMA algorithm:

- construct a distance matrix $D$ with pairwise distances $D_{ij}$
- let $n_k$ be the number of sequences in group $k$ (initially $n = 1$ for all groups) where a “group” means a cell in the matrix (corresponding to a node in the tree)
- repeat until the matrix has one element:
  - find groups $i, j$ with the smallest value of $D_{ij}$
  - create group $(ij)$ with $n_{ij} = n_i + n_j$
  - connect $i$ and $j$ to a new tree node $(ij)$; set the distance from $(ij)$ to leaves below to $D_{ij} / 2$
  - for all other groups $k$:
    - $D_{(ij),k} = \left( \frac{n_i}{n_i + n_j} \right) D_{ik} + \left( \frac{n_j}{n_i + n_j} \right) D_{jk}$
  - update $D$: delete row $i$, col $j$, add row and column for $(ij)$

* See J. Felsenstein, Inferring Phylogenies

---

**In-Class Example**

Let’s do the first few steps using an example matrix from Felsenstein

<table>
<thead>
<tr>
<th>D</th>
<th>B</th>
<th>R</th>
<th>W</th>
<th>S</th>
<th>L</th>
<th>C</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>D</td>
<td>32</td>
<td>48</td>
<td>51</td>
<td>50</td>
<td>48</td>
<td>98</td>
<td>148</td>
</tr>
<tr>
<td>B</td>
<td>26</td>
<td>34</td>
<td>29</td>
<td>33</td>
<td>84</td>
<td>136</td>
<td></td>
</tr>
<tr>
<td>R</td>
<td>42</td>
<td>44</td>
<td>44</td>
<td>92</td>
<td>152</td>
<td></td>
<td></td>
</tr>
<tr>
<td>W</td>
<td>86</td>
<td>142</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td></td>
<td></td>
<td>24</td>
<td>89</td>
<td>142</td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td></td>
<td></td>
<td></td>
<td>90</td>
<td>142</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>148</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

D: Dog  
B: Bear  
R: Raccoon  
W: Weasel  
S: Seal  
L: Sea Lion  
C: Cat  
M: Monkey

**First Round**

- The two closest taxa are $S$ and $L$
  - distance is 24, so branch lengths below new node for $SL$ are both 12
- New matrix:

<table>
<thead>
<tr>
<th>D</th>
<th>B</th>
<th>R</th>
<th>W</th>
<th>C</th>
<th>M</th>
<th>SL</th>
</tr>
</thead>
<tbody>
<tr>
<td>D</td>
<td>32</td>
<td>48</td>
<td>51</td>
<td>98</td>
<td>148</td>
<td>49</td>
</tr>
<tr>
<td>B</td>
<td>26</td>
<td>34</td>
<td>84</td>
<td>136</td>
<td>31</td>
<td>30</td>
</tr>
<tr>
<td>R</td>
<td>42</td>
<td>92</td>
<td>152</td>
<td>44</td>
<td>36</td>
<td></td>
</tr>
<tr>
<td>W</td>
<td>86</td>
<td>142</td>
<td>44</td>
<td>34</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td>148</td>
<td>89.5</td>
<td></td>
</tr>
<tr>
<td>M</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>142</td>
<td></td>
</tr>
<tr>
<td>SL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>24</td>
</tr>
</tbody>
</table>

---

**Third Round**

- The second round combined $B$ and $R$ and computed distances from $BR$ to $SL$, $D$, $W$, $C$, and $M$
- The third round combines the two new interior nodes
  - the distance from $SL$ to $BR$ is $D = 37.5$
  - that means the path from $S$ or $L$ up to the new interior node and then back to $B$ or $R$ should be a total of 37.5
  - $D / 2 = 18.75$
  - since the $SL$ branches are already 12.0, the path from there to the new node is 6.75
  - the $BR$ branches are already 13.0, so their path to the new node is 5.75
**UPGMA Tree**

* Here is the final tree:

![UPGMA Tree Diagram]

---

**Neighbor-Joining**

* Neighbor-joining is another distance-based method
  * like UPGMA it is accurate when data fit the requirements (i.e. they are metrics)
  * is efficient and works well for sequences that are not too far apart
  * is a good (best?) method for making a guide tree for multiple alignment

* Starting point: make a star topology with one node for each taxon

* Construction step: “resolve” two nodes, replace them with new interior node

Note: this method builds an unrooted tree

---

**In-Class Example**

* The unrooted neighbor-joining tree

![In-Class Example Diagram]
NJ vs UPGMA

- Comparing the two trees
  - use the longest branch in the NJ tree as an “outgroup” (the likely root of the tree)

![Diagram of NJ vs UPGMA trees]

Maximum Parsimony

- The most common character-based tree building method is maximum parsimony
- For molecular evolution, the characters compared in this method are the letters at corresponding locations in globally aligned sequences
- Philosophy: evolutionary changes are rare, and the tree that requires fewest events to explain observed data is the most likely tree

Parsimony (cont’d)

- General overview:
  - create a global alignment of all sequences
  - locate **informative sites** in the aligned sequences
    - informative sites will help choose among alternative trees
  - generate all (unrooted) trees for each informative site
  - count the number of changes defined in each tree
  - determine which tree requires the fewest changes across all sites

![Diagram of Parsimony process]

Informative Sites

- An informative site is an alignment position that has information to prefer one tree over the others
  - Example (from Li and Graur)

<table>
<thead>
<tr>
<th>Site</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>G</td>
<td>T</td>
<td>G</td>
<td>C</td>
<td>A</td>
</tr>
<tr>
<td>B</td>
<td>A</td>
<td>G</td>
<td>C</td>
<td>C</td>
<td>G</td>
<td>T</td>
<td>G</td>
<td>C</td>
<td>G</td>
</tr>
<tr>
<td>C</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>T</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>C</td>
<td>A</td>
</tr>
<tr>
<td>D</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>C</td>
<td>G</td>
</tr>
</tbody>
</table>
Informative Sites (cont’d)

- Site 1 does clearly has no information (all are A’s)
- At right are three possible trees for site 2
- A • on a branch where shows where a change occurs
- This site also does not prefer any tree

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>G</td>
<td>T</td>
<td>G</td>
<td>C</td>
<td>A</td>
</tr>
<tr>
<td>B</td>
<td>A</td>
<td>G</td>
<td>C</td>
<td>C</td>
<td>G</td>
<td>T</td>
<td>G</td>
<td>C</td>
<td>G</td>
</tr>
<tr>
<td>C</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>T</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>C</td>
<td>A</td>
</tr>
<tr>
<td>D</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>C</td>
<td>G</td>
</tr>
</tbody>
</table>

Informative Sites (cont’d)

- Site 3 also does not prefer one tree over any other -- two mutations are required in each of these trees to account for the observed data

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>G</td>
<td>T</td>
<td>G</td>
<td>C</td>
<td>A</td>
</tr>
<tr>
<td>B</td>
<td>A</td>
<td>G</td>
<td>C</td>
<td>C</td>
<td>G</td>
<td>T</td>
<td>G</td>
<td>C</td>
<td>G</td>
</tr>
<tr>
<td>C</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>T</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>C</td>
<td>A</td>
</tr>
<tr>
<td>D</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>C</td>
<td>G</td>
</tr>
</tbody>
</table>

Informative Sites (cont’d)

- Same story for site 4

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>G</td>
<td>T</td>
<td>G</td>
<td>C</td>
<td>A</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>A</td>
<td>G</td>
<td>C</td>
<td>C</td>
<td>G</td>
<td>T</td>
<td>G</td>
<td>C</td>
<td>G</td>
</tr>
<tr>
<td>C</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>T</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>C</td>
<td>A</td>
</tr>
<tr>
<td>D</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>C</td>
<td>G</td>
</tr>
</tbody>
</table>

Informative Sites (cont’d)

- Finally -- site 5 shows a preference for one of the trees
- tree (1) shows that if a and b are grouped, and c and d are grouped, one mutation is sufficient to account for the observed pattern

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>G</td>
<td>T</td>
<td>G</td>
<td>C</td>
<td>A</td>
</tr>
<tr>
<td>B</td>
<td>A</td>
<td>G</td>
<td>C</td>
<td>C</td>
<td>G</td>
<td>T</td>
<td>G</td>
<td>C</td>
<td>G</td>
</tr>
<tr>
<td>C</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>T</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>C</td>
<td>A</td>
</tr>
<tr>
<td>D</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>C</td>
<td>G</td>
</tr>
</tbody>
</table>
Informative Sites (cont’d)

- The general rule for informative sites:
  - there are at least two different characters at each site
  - each character is represented at least twice

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>G</td>
<td>T</td>
<td>C</td>
<td>C</td>
<td>A</td>
</tr>
<tr>
<td>B</td>
<td>A</td>
<td>G</td>
<td>C</td>
<td>C</td>
<td>G</td>
<td>T</td>
<td>C</td>
<td>C</td>
<td>G</td>
</tr>
<tr>
<td>C</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>T</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>C</td>
<td>A</td>
</tr>
<tr>
<td>D</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>G</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>C</td>
<td>G</td>
</tr>
</tbody>
</table>

Sum Over All Sites

- Compute the sum of changes over all trees
  - sites 5 and 7 prefer tree (1)
  - site 9 prefers tree (2)

Choosing a Root

- An unrooted tree can be turned into a rooted tree by grafting a root onto any of the branches

Outgroup

- A method that produces an unrooted tree can be used to infer a rooted phylogeny
- Use an “outgroup” of one or more sequences that are known (or assumed) to be more distantly related
- Example:
  - goal is a phylogeny of vertebrates (mammals, fish, …)
  - include a sequence from an invertebrate animal with the data set
  - where this branch connects to the rest of the tree is a likely ancestral (root) node
Pitfalls for Parsimony

- All phylogenetic inference methods have problems when different branches have different rates of evolution
- See link to simulation study from class web page
  - generate random tree, see how well different methods reconstruct known tree
  - MP: many incorrect trees when rates differ by 3:1 and 50% or less similarity on branches
  - ML: worked better up to 12:1 rate ratio and 30% similarity

Long Branch Attraction

- The problem: “long branch attraction”
- Two branches with higher rates will be clustered together by distance and character based methods

Applications

- Several programs that use the maximum parsimony method to construct trees
  - PAUP*
    - originally “phylogenetic analysis using parsimony”
    - now has several methods
  - PHYLIP
    - dnapars
    - protpars

Statistical Methods

- The types of methods covered in the previous slides are
  - distance methods (UPGMA and Neighbor-Joining) that calculate pairwise distances and then use the distances to create a tree
  - a character based method (Maximum Parsimony) that uses differences between sequences to find the tree that requires the least complex story to account for observed data
- The next two methods are statistical methods
  - based on probabilities and techniques for making statistical inferences
  - Maximum Likelihood was developed for phylogenetic inference by Felsenstein in the early 1980s
  - Bayesian Inference was applied to phylogeny in the late 1990s and is becoming more widely used
**Probability**

- Probability is a term used when predicting the odds of future events
  - example: a standard deck of 52 cards has 4 aces, 4 kings, etc
  - the probability of drawing an ace is 1/13
    (the number of aces divided by the number of cards)
  - the probability of drawing a card worth 10 in blackjack is 16/52
- Notation:
  - \( p(\text{ace}) = 0.077 \)
  - \( p(\text{10 points}) = 0.308 \)

**Joint Probability**

- The joint probability of events \( x \) and \( y \) is the probability of both occurring
  - written \( p(x,y) \) or \( p(x \land y) \)
  - If \( x \) and \( y \) are independent \( p(x,y) = p(x) \times p(y) \)
    - the odds of drawing two aces from a standard deck:
      - \( p(\text{ace}_1) = 4/52 = 0.077 \)
      - \( p(\text{ace}_2) = 3/51 = 0.059 \)
      - \( p(\text{ace}_1, \text{ace}_2) = 0.0045 \)

**Conditional Probability**

- If \( x \) and \( y \) are not independent the joint probability can still be defined if we know their conditional probability
  - the notation \( p(x \mid y) \) means “the probability of \( x \) given \( y \)”
  - informally: the probability of \( x \) occurring once we know \( y \) has occurred or will occur
    - does not mean \( y \) causes \( x \), just that if \( y \) occurs it will modify the probability of \( x \)
- The formula for joint probability if we know \( p(x \mid y) \):
  \[ p(x,y) = p(x \mid y) \times p(y) \]

**Example:**

- Pinochle is played with 48 cards
  - 8 each of A, K, Q, J, 10, 9
  - the probability of drawing an ace from a pinochle deck is 8/48 = 0.167 (compared to 4/52 = 0.077 for a standard deck)
- Suppose there are two decks of cards on a table
  - one is standard, the other Pinochle, but you don’t know which is which
  - you choose a deck at random, and then choose a card from that deck
  - the odds of picking an ace from the Pinochle deck are
    \[ p(\text{ace, pin}) = p(\text{ace} \mid \text{pin}) \times p(\text{pin}) \]
    \[ = 0.167 \times 0.5 \]
    \[ = 0.0833 \]

Q: Is this the same as throwing all the cards in a bag and choosing a Pinochle ace from the bag? i.e. \( p(\text{ace, pin}) = 8/100 \)?
Marginal Probability

We can make a table to show the probability of drawing each type of card in this experiment.

The sum of the probabilities in a row or column defines a marginal probability:

- the probability of the event labeling the row or column
- example:

\[
p(\text{ace}) = .0385 + .0833 = .1218
\]

\[
p(\text{std}) = 13 \times .0385 = .5
\]

<table>
<thead>
<tr>
<th></th>
<th>standard</th>
<th>pinochle</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.0385</td>
<td>0.0833</td>
</tr>
<tr>
<td>K</td>
<td>0.0385</td>
<td>0.0833</td>
</tr>
<tr>
<td>Q</td>
<td>0.0385</td>
<td>0.0833</td>
</tr>
<tr>
<td>3</td>
<td>0.0385</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0.0385</td>
<td>0</td>
</tr>
</tbody>
</table>

Bayes’ Theorem

Suppose we pick up a deck at random and deal a poker hand (5 cards) and find we have a royal flush (A, K, Q, J, 10 of one suit).

- There is a good chance we picked up the pinochle deck
- How can we quantify this?
- It’s straightforward to compute \( p(\text{rf} \mid \text{pin}) \), but what about \( p(\text{pin} \mid \text{rf}) \)?

\[
p(\text{rf} \mid \text{pin}) \quad \frac{\text{A}: 20/52 \times 4/51 \times 3/50 \times 2/49 \times 1/48 = .00000153}{\text{A}: 5/6 \times 8/47 \times 6/46 \times 4/45 \times 2/44 = .0000747}
\]

Q: what is \( p(\text{rf} \mid \text{pin}) \)? \( p(\text{rf} \mid \text{std}) \)?

Earlier we saw this definition of conditional probability \( p(x \mid y) \):

\[
p(x, y) = p(x \mid y) \times p(y)
\]

The same formula also defines \( p(y \mid x) \) since \( p(x, y) = p(y, x) \):

\[
p(x, y) = p(y \mid x) \times p(x)
\]

Putting these together and dividing by \( p(x) \) gives Bayes’ theorem:

\[
p(y \mid x) = \frac{p(x \mid y) \times p(y)}{p(x)}
\]

Bayes’ Theorem

So for the royal flush example,

\[
p(\text{pin} \mid \text{rf}) = \frac{p(\text{rf} \mid \text{pin}) \times p(\text{pin})}{p(\text{rf})}
\]

- we already calculated \( p(\text{rf} \mid \text{pin}) \)
- \( p(\text{pin}) \) is .5
- \( p(\text{rf}) \) is the marginal probability of a royal flush, which is the sum over all types of decks of the probability of a royal flush in that type of deck:

\[
p(\text{rf}) = \sum_i p(\text{rf} \mid d_i) \times p(d_i)
\]

Q: what is \( p(\text{pin} \mid \text{rf}) \)?
Bayesian Inference

- Bayes’ theorem is used in scientific contexts to develop or refine hypotheses about data

\[ p(H|D) = \frac{p(D|H) \times p(H)}{p(D)} \]

- \( p(H) \) is the prior probability of hypothesis \( H \) (what one expects without having seen the data)
- \( p(D) \) is the marginal probability of the data: the probability of observing the data independent of any hypothesis (or summed over all possible hypotheses)
- \( p(D|H) \) is the probability of the data given the hypothesis, also called the likelihood of the data
- \( p(H|D) \) is the posterior probability of the hypothesis after taking the data into account

Likelihood and Phylogeny

- In phylogenetic inference, the likelihood function is \( p(D|T) \)
  - \( D \) is the data, a set of aligned sequences
  - \( T \) is a hypothesis about the evolution of a set of sequences, in the form of a phylogenetic tree
  - \( T \) includes a topology that shows ancestral relations, branch lengths, and an evolutionary parameters that includes substitution rates, transition/transversion ratios, etc
- The function is often written as \( p(D|T, \theta) \) to show that \( D \) depends on the tree structure \( T \) and evolutionary parameters \( \theta \)

Bayesian Methods for Phylogeny

- The goal of a Bayesian approach to phylogenetic inference is to compute the posterior probability of a tree

\[ p(T|D) = \frac{p(D|T) \times p(T)}{p(D)} \]

- or, to make it clear that the marginal probability of the data is summed over all possible trees:

\[ p(T|D) = \frac{p(D|T) \times p(T)}{\sum_i p(D|T_i) \times p(T_i)} \]

- The difficulties with this approach:
  - there are a lot of trees (computing that sum is going to take a long time)
  - what is the prior expectation \( p(T) \) for a tree?

- A common choice: uniform priors
  - all trees are equally likely
  - but is this really true? or close enough so the method produces an accurate result?
  - consider how many different branch length combinations there are...
Monte Carlo Integration

- Since there are an infinite number of trees the sum is an integral
- A famous technique for numeric integration is the Monte Carlo method
  - based on random sampling
  - choose a set of random points from a region determined by boundaries $a$, $b$, and $c$
  - the area under the curve can be estimated by proportion of samples that fall inside the region $R$
- Imagine throwing darts, keeping track of how many land in $R$

Monte Carlo Integration (cont'd)

- In mathematical terms:
  $$P[x_i, y_i \in R] = \frac{A}{c(b-a)} \approx \frac{M}{N}$$
  where
  - $A$ is the area under the curve
  - $M$ is the number of samples in $R$
  - $N$ is the total number of samples
- The value of the integral is $A$, the area of $R$:
  $$\int_a^b f(x) \, dx \approx \frac{c(b-a)M}{N}$$

MCMC

- A version of Monte Carlo integration known as Markov Chain Monte Carlo (MCMC) can be used to sample from the space of phylogenetic trees
  - pick a random tree as a starting point
  - define a function that generates a new tree from the current tree
  - apply the function a zillion times to generate a useful random sample by doing a random walk through tree space
  - $p(T_i)$ is proportional to the number of times $T_i$ is encountered

Application

- The most widely used application for Bayesian inference of trees is probably MrBayes
  - see Huelsenbeck (2001) [available from the class web site]
  - see also Huelsenbeck (2002) for a discussion of the MCMC method
  - the MrBayes home page (docs, downloads, ...) is http://mrbayes.csit.fsu.edu
Maximum Likelihood

- The likelihood function from the Bayes equation is $p(D|T, \theta)$
  - the probability of the data given tree $T$ and model $\theta$
- In the maximum likelihood method the goal is to find the values of $T$ and $\theta$ that maximize this function
- Why use ML instead of Bayesian inference?
  - difficulty in defining the prior distribution of trees $p(T)$
  - some feel the use of prior expectations is too subjective
  - philosophical differences (e.g. "in science there is no place for a hypothesis independent of data")

Likelihood Calculation

- The outer loop of the maximum likelihood method uses an optimization algorithm:
  - assign an initial value to each parameter of the model $\theta$
  - find the tree with the highest likelihood
  - repeat
    - adjust the parameter settings to get $\theta'$
    - find the tree with the highest likelihood given $\theta'$
    - if the new tree has a higher likelihood remember it
  - stop when adjustments produce no increase in likelihood

Likelihood Calculation (cont’d)

- To find the most likely tree we need to consider all labelings (interior and exterior) of all trees
  - for each site in the alignment:
    - for each possible unrooted tree topology $T$:
      - for each possible labeling $i$:
        - compute the likelihood $L_i$ using current model $\theta'$
        $$L_T = \Sigma L_i$$
        - select the tree with the highest $L_T$

Likelihood Example

- An example with four sequences (from Mount):
  - There are three possible unrooted trees for each site
  - For each unrooted tree there are five possible rooted trees
  - the rooted trees shown at right are for the unrooted tree on the top left
Likelihood Example (cont’d)

- The inner loop calculates likelihoods for each of the rooted trees
- Each rooted tree has 3 interior nodes
- There are $4^3 = 64$ possible assignments of bases to the three interior nodes
  - Each assignment defines 7 probabilities
  - $L_1$ is the probability of observing the base at the root
  - $L_{i>1}$ is the probability of the change indicated on branch $i$

\[
\begin{align*}
T &\quad T &\quad A &\quad G \\
1 &\quad 2 &\quad 3 &\quad 4
\end{align*}
\]

- Repeat for all 64 assignments of bases to interior nodes
- Since each labeling is independent, the total likelihood for this tree is the sum over all labelings
- Repeat for all 3 unrooted trees using a representative set of branch lengths
  - branch lengths are not discrete so we can’t generate all possibilities

\[
L_{\text{tree}} = L_{AAA} + L_{ABC} + \cdots
\]

Likelihood Example (cont’d)

- Suppose the interior nodes are T, T, and G
  - $L_1$ = background frequency of T
  - $L_2$ = probability of T remaining T
  - $L_3$ = probability of T changing to G
  - ...
  - $L_7$ = probability of G remaining G

- Note the probabilities depend on branch lengths (which are also parameters that can vary)
- The likelihood of this labeling is the product of each individual likelihood

\[
L_{\text{TTG}} = \prod_i L_i
\]

Likelihood Example (cont’d)

- Maximum likelihood depends on several key assumptions:
  - each site evolves independently
  - changes at a site are independent and reversible (i.e. mutation at each site is a Markov process)
  - the mutation rate remains the same across all branches
  - mutations are symmetric ( $p(X,Y) = p(Y,X) $ )
  - otherwise location of root is important
Heuristic Search

- An exhaustive analysis of each possible tree is prohibitive
- A search over the complete space of evolutionary parameters is obviously very intensive
- Most applications use some sort of heuristics or approximations to look for likely trees
  - see papers (and book) by Felsenstein for an algorithm

Applications

- There are several implementations of the maximum likelihood method
- In the PHYLIP package:
  - dnaml
  - dnamlk (molecular clock)
- PAML (Phylogenetic Analysis by Maximum Likelihood)
  - codeml considers groups of three bases at a time
- fastDNAml
  - parallel version of dnaml from UIUC