TODAY’S TOPICS

➤ Phylogenetic reconstruction
- Distance methods (NJ)
- Character-based methods (maximum parsimony, maximum likelihood)

Phylogenetic methods

➤ Data types and tree-building methods

<table>
<thead>
<tr>
<th>Distances</th>
<th>Characters</th>
</tr>
</thead>
<tbody>
<tr>
<td>UPGMA</td>
<td>Neighbor joining</td>
</tr>
<tr>
<td>Neighbor joining</td>
<td>Approximates minimum evolution tree</td>
</tr>
<tr>
<td>Minimum evolution</td>
<td>Maximum parsimony</td>
</tr>
<tr>
<td>Fitch-Margoliash</td>
<td>Maximum likelihood</td>
</tr>
</tbody>
</table>

Examine all possible topologies based on a certain criterion

Phylogenetic methods (Distance)

➤ Neighbor joining (NJ)

Saitou and Nei (1987) and correction by Studier and Keppler (1988)

- Clustering method (fast computation)
- A good heuristic method for estimating the minimum evolution tree
  ➜ No guarantee to find the ME tree
  ➜ In practice, the NJ tree is often the same or very similar to the ME tree
- No assumption for the constant evolutionary rate
  ➜ Evolutionary rate can vary among lineages
- reconstructs unrooted trees

Phylogenetic methods (Distance)

Example: a distance matrix for 5 OTUs

1) Start with a star phylogeny
2) Si: the sum of all branch lengths

$S_i = \sum_{j \neq i} d_{ij} / (m - 1)$

$E_{jk}$: branch length between OTU i and node j

For 3 OTUs: $d_{12} = 1.53, d_{13} = 1.99, d_{14} = 1.02, d_{15} = 1.65, d_{23} = 1.52, d_{24} = 2.73, d_{25} = 1.54$

$S_{12} = 1.53 + 1.99 + 1.02 + 1.65 + 1.52 + 2.73 + 1.54 = 12.055$

Find the shortest $S_{ab}$

Phylogenetic methods (Distance)

3) Take 2 OTUs (a and b) as a pair (neighbors) and calculate the tree length ($S_{ab}$) from this topology
4) Find the OTU pair that makes the shortest $S_{ab}$

ME tree: has the topology with the shortest tree length

$S = \sum S_i$

Phylogenetic reconstruction

Phylogenetic methods

Clustering

Optimality criterion

Clustering methods (Bayesian inference)

Phylogenetic methods (Distance)

Neighbor joining (NJ)

3) Calculate the sum of all branch length ($S_{ab}$) when OTUs $a$ and $b$ are neighbors.

- If OTUs 1 and 2 are the neighbors:
  \[
  S_{12} = L_{1X} + L_{2X} + L_{XY} = \sum_{i=1}^{n} L_{iX} + \sum_{j=1}^{m} L_{jY} + L_{XY}
  \]

- Calculate the sum of all branch lengths ($L_{XY}$):
  \[
  L_{XY} = \sum_{i=1}^{n} d_{ai} + \sum_{j=1}^{m} d_{bj} - (m-2)(L_{XX} + L_{YY} - 2L_{XY})
  \]

- Calculate the sum of all distances that include $L_{XY}$:
  \[
  \sum_{i=1}^{n} (d_{ai} + d_{bj}) = d_{ab} + d_{ba} + \sum_{i=1}^{n} d_{ai} + \sum_{j=1}^{m} d_{bj}
  \]

- Calculate the sum of all irrelevant branch lengths:
  \[
  \sum_{i=1}^{n} d_{ai} = \sum_{j=1}^{m} d_{bj}
  \]

- Find the shortest $S_{ab}$:
  \[
  S_{ab} = \min(S_{12}, S_{21})
  \]

- Create a new node (A) that connects OTUs $a$ and $b$.

- Branch lengths are calculated using Fitch-Margoliash method.

- Continue until all OTUs are clustered.

A new distance matrix for 4 OTUs (m=4):

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1.2</td>
<td>0.5</td>
</tr>
<tr>
<td>2</td>
<td>1.2</td>
<td>0</td>
<td>0.3</td>
</tr>
<tr>
<td>3</td>
<td>0.5</td>
<td>0.3</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0.9</td>
<td>0.8</td>
<td>1.1</td>
</tr>
</tbody>
</table>

Neighbor Joining (NJ) tree

Phylogenetic methods (Distance)

Neighbor joining (NJ)

3) Sum of all branch lengths ($S_{ab}$) is calculated for all pairs of OTUs.

4) Find the shortest $S_{ab}$.

5) Create a new node (A) that connects OTUs $a$ and $b$.

6) Branch lengths are calculated using Fitch-Margoliash method.

7) The new distance matrix is calculated using the combined OTU A (for $a$ and $b$).

Distance between the new OTU A and the remaining OTUs k (1≤k≠m where k≠a and k≠b):

- For m=3, b=4, and m=5:
  \[
  d_{ik} = (d_{ik} + d_{ak} - d_{ab}) / 2
  \]

8) Continue until all OTUs are clustered.

For the last 3 OTUs (see Lecture 21, slide 43):

- $L_{XY}$ calculated:
  \[
  L_{XY} = \sum_{i=1}^{n} d_{ai} + \sum_{j=1}^{m} d_{bj} - (m-2)(L_{XX} + L_{YY} - 2L_{XY})
  \]

- New distance matrix for 4 OTUs (m=4):

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1.2</td>
<td>0.5</td>
</tr>
<tr>
<td>2</td>
<td>1.2</td>
<td>0</td>
<td>0.3</td>
</tr>
<tr>
<td>3</td>
<td>0.5</td>
<td>0.3</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0.9</td>
<td>0.8</td>
<td>1.1</td>
</tr>
</tbody>
</table>
Phylogenetic methods (Distance)

**UPGMA**

1. 
2. 
3. 
4. 
5.

**NJ**

1. 
2. 
3. 
4. 
5.

---

Phylogenetic methods (Distance)

**UPGMA**

1. 
2. 
3. 
4. 
5.

**NJ**

1. 
2. 
3. 
4. 
5.

---

Phylogenetic methods

- **Data types and tree-building methods**
  - **Distances**
    - Clustering: UPGMA, Neighbor joining
    - Optimality criterion: Minimum evolution, Maximum parsimony, Maximum likelihood
  - **Characters**

---

Phylogenetic methods (Character-based)

- **Maximum Parsimony (MP)**
  - Choose the tree(s) that require(s) the fewest evolutionary changes = the shortest tree(s)
  - Data: individual sites
  - For each site (column), reconstruct the evolution of that site based on a given tree topology and with fewest possible evolutionary changes
  - Tree length = Sum of the minimum numbers of character changes over all sites

\[ L(t) = \sum_{i=1}^{N} l_i \]

- \( t \): tree topology
- \( N \): number of sites (characters)
- \( l_i \): tree length for a single site \( i \)
  - (amount of character change)

---

Phylogenetic methods (Character-based)

- **Maximum Parsimony (MP)**

---

Phylogenetic methods (Character-based)

- **Maximum Parsimony (MP)**

---
Phylogenetic methods (Character-based)

- **Maximum Parsimony (MP)**

<table>
<thead>
<tr>
<th>Sites</th>
<th>OTUs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 ATATT&lt;br&gt;2 ATCGT&lt;br&gt;3 GCAGT&lt;br&gt;4 GCCGT</td>
<td>1&lt;br&gt;2&lt;br&gt;3&lt;br&gt;4</td>
</tr>
</tbody>
</table>

**Most parsimonious!**

1 change

Tree length: \( L = 1 + 1 + 2 + 1 + 0 = 5 \)

**Tree I is the MP tree!**

Find the tree length from each possible tree topology.
Phylogenetic methods (Character-based)

- **Terminologies for character evolution**
  - Homologous:
    - Phylogenetically informative sites
    - Phylogenetically uninformative sites
  - Homoplasy:
    - Independently acquired similarity
    - Not homologous

- **Maximum Parsimony (MP)**
  - Not based on an explicit model of evolution
  - What do we do if substitution patterns are biased?
    - (e.g., saturation in transitional substitutions)
  - Cost matrix (or weight matrix or step matrix)
    - When counting the number of changes, use different weighting depending on the reliability of character change information
      - Transversion weighting
        - Ts could be saturated and may not reflect the correct evolutionary relationships (less phylogenetic information)
        - Down-weight Ts
        - Phylogenetic information from Tv is more reliable
        - Up-weight Tv

- **Identity substitution matrix used for alignment**
  - To minimize the changes
  - To maximize the identity

- **Cost matrix**
  - A 0 1 1
  - T 0 0 1
  - C 1 0 0
  - G 1 0 0

- **Formulas**
  - \( L(\tau) = \sum_{i} l_{i} \)
  - \( l_{i} = \sum_{a=0}^{2n-3} C_{x,y} \)
  - \( C_{xy} \) is the cost associated with the change from state \( x \) to \( y \)
  - The states assigned to the nodes at either end of the branch are assigned to the nodes at either end of the branch.

- **Example**
  - ACGT
  - ACGT
Phylogenetic methods (Character-based)

- Maximum Parsimony (MP)
  - Simple parsimony
    - \( C_{xy} = 0 \) for \( x = y \)
    - \( C_{xy} = 1 \) for \( x \neq y \)

- Generalized parsimony (weighted parsimony)
  - \( C_{xy} = 0 \) for \( x = y \)
  - \( C_{xy} = 1 \) for transition
  - \( C_{xy} = w \) for transversion

\[ \sum_{i=1}^{w} l_i \]

.Tree:

- MP tree!

- Transversions are weighted more than transitions

Tree length:

\[ L = 1 + w \]

Sites:

- \( l_1 \) changes
- \( l_2 \) changes
- \( l_3 \) changes
- \( l_4 \) changes

\[ L(T) = L + \sum_{i=1}^{w} l_i \]

Tree length:

\[ L = 1 + 1 + 8 + 4 + 0 = 14 \]
Phylogenetic methods (Character-based)

- Maximum Parsimony (MP): weighted

<table>
<thead>
<tr>
<th>Tree</th>
<th>Sites</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: (L2a,3,b)</td>
<td>1 1 1 2 4 6</td>
</tr>
<tr>
<td>2: (L3a,2,b)</td>
<td>2 2 4 4 0</td>
</tr>
<tr>
<td>3: (L4a,2,b)</td>
<td>2 2 2 4 4</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>16</td>
</tr>
</tbody>
</table>

- Maximum Parsimony (MP): examples

Maximum Parsimony (MP) is a method for inferring evolutionary relationships among species. It aims to find the tree with the fewest changes, making it a weighted method. The formula for Maximum Parsimony is

\[ L_T = \sum_{i=1}^{w} c_i \]

where \( L_T \) is the total length of the tree, and \( c_i \) is the cost of compiling the species from the species (data) at the ith site.

Phylogenetic methods (Character-based)

- Maximum Likelihood (ML)
  - Chooses the tree that makes the observed data the most probable evolutionary outcome
  - Likelihood = Conditional probability of obtaining the observed sequences given a hypothesis (substitution model and tree)

\[ L(\mathbf{\tau}, \mathbf{\theta}) = \text{Prob(Data} | \mathbf{\tau}, \mathbf{\theta}) = \text{Prob(Alied sequences} | \text{tree, model of evolution}) \]

- Minimum evolution
- Fitch-Margoliash
- Maximum parsimony
- Maximum likelihood

Phylogenetic methods (Character-based)

- Maximum Likelihood (ML)

\[ L(\mathbf{\tau}, \mathbf{\theta}) = \text{Prob(Data} | \mathbf{\tau}, \mathbf{\theta}) \]

Find the topology that gives the maximum \( L(\mathbf{\tau}, \mathbf{\theta}) \), and simultaneously estimate all required parameters.

To compute the likelihood of a given tree, \( \mathbf{\tau} \): the topology and the maximum likelihood estimates for the tree’s branch lengths \( \{d_1, d_2, \ldots\} \) need to be found

\[ \mathbf{\theta} \]: the best values for the parameters for the evolutionary model need to be found, too.

- Clustering
  - UPGMA
  - Neighbor joining
- Optimality criterion
- Minimum evolution
- Fitch-Margoliash
- Maximum parsimony
- Maximum likelihood

Phylogenetic methods

Data types and tree-building methods

<table>
<thead>
<tr>
<th>Data types</th>
<th>Characters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distances</td>
<td>Clustering</td>
</tr>
<tr>
<td></td>
<td>UPGMA</td>
</tr>
<tr>
<td></td>
<td>Neighbor joining</td>
</tr>
<tr>
<td>Optimality criterion</td>
<td>Minimum evolution</td>
</tr>
<tr>
<td></td>
<td>Fitch-Margoliash</td>
</tr>
<tr>
<td></td>
<td>Maximum parsimony</td>
</tr>
<tr>
<td></td>
<td>Maximum likelihood</td>
</tr>
</tbody>
</table>

Examining all possible topologies based on a certain criterion.
Phylogenetic methods (Character-based)

- **Maximum Likelihood (ML)**

  - To compute the likelihood of a given tree, L(\(\pi, \theta\) = Prob(Data | \(\pi, \theta\))
  - \(\pi\): the maximum likelihood estimates for the tree’s branch lengths \(\{a, b, ...\}\) need to be found
  - \(\theta\): the best values for the parameters for the evolutionary model need to be found, too
  - \([JC\ model\]: single evolutionary rate)
  - \([JCAM\ model\]: transition/transversion ratio (TS/TV)
  - base composition, gamma shape parameter (\(\alpha\))
  - Number of the parameters depends on the model

**Phylogenetic methods: pros and cons**

- **Criticisms to distance methods**
  - Depend on distance estimation method
  - Summarizing a set of sequences by a pairwise distance matrix loses information

- **Criticisms to maximum parsimony methods**
  - “Long branch attraction” problem
    - If the internal branch is short relative to the terminal branches, the MP tree is likely to be incorrect
  - No multiple hit correction

- **Criticisms to maximum likelihood methods**
  - Require an explicit model of evolution
  - Which model to use?
Phylogenetic methods

- Clustering vs. search methods
  - Clustering methods (UPGMA, Neighbor-joining)
    - Do not search all possible topologies
    - Very fast
    - Produce only one tree
  - Search methods
    - Use optimality criterion (minimum evolution, maximum parsimony, maximum likelihood)
    - Exhaustive search for all possible topologies is not possible for a large number of taxa
    - A heuristic search algorithm needs to be used

How can we search trees?

Phylogenetic methods: tree searching

- Number of possible tree topologies

<table>
<thead>
<tr>
<th>Number of OTUs</th>
<th>Number of rooted trees</th>
<th>Number of unrooted trees</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>15</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>105</td>
<td>15</td>
</tr>
<tr>
<td>7</td>
<td>954</td>
<td>105</td>
</tr>
<tr>
<td>8</td>
<td>135,135</td>
<td>135,135</td>
</tr>
<tr>
<td>9</td>
<td>2,027,025</td>
<td>2,027,025</td>
</tr>
<tr>
<td>10</td>
<td>34,459,425</td>
<td>34,459,425</td>
</tr>
</tbody>
</table>

Impossible to examine all of the possible trees!

Tree-searching methods (unrooted)

- Exhaustive search
  - Possible only for a few taxa (11 or fewer)
  - Number of possible unrooted tree:
    \[ B(i) = \prod_{l=1}^{i/2} (2l - 5) \]
    
    \( B(7) = 1 \times 3 \times 5 \times 7 = 945 \)
    \( B(10) = B(7) \times 11 \times 13 \times 15 > 2 \times 10^6 \)
    \( B(20) > 2 \times 10^{10} \)
  - An algorithm is required to guarantee generation of all possible trees
Tree-searching methods (unrooted)

- **Exhaustive search**

  If there are 2 taxa
  
  Only one topology possible

  There is only one place to add the third taxon

  If there are 3 taxa

  There are three places to add the fourth taxon

  Three topologies are possible

- **Exhaustive search for 5-taxon trees**

  If there are 4 taxa

  There are five places to add the fifth taxon

  Five topologies are possible

- **Exhaustive search**

  Searches all possible trees

  Guarantees to find the optimal trees

  Impractical for many taxa

Each of the possible topologies (3 x 5 = 15 trees for 5-taxon tree) will be evaluated to identify optimal trees (ME, MP, or ML trees)