Announcements

• HW3 is Due Nov. 5 (Both parts)
**Current Topics**

- **Last time**
  - A brief introduction to Genome Sequencing (Shotgun & Assembly)
  - Genome sequencing – Wrap-up
  - Intro to Phylogenetic Trees (some with Gongbo)

- **Today**
  - Phylogenetic Trees (continued)
  - End Intro
  - UPGMA
  - Ultrametric Distance (and the UPGMA)
  - Neighbor Joining

---

**Phylogenetic Trees: Review**

*Graph-theoretical Concepts: Tree* - directed and undirected

*Components of Phylogenetic Trees*

- **Leaves:** Species (taxon/taxa) / Genes / Proteins /… (labeled)
- **Internal nodes:** Hypothetical common ancestry (unlabeled)
- **Root** (if there is one): Represents the common ancestor
- **An un-rooted tree:** Relations - but no directed evolutionary path.
- **Edge length:** Captures evolutionary distance among nodes

---

**Phylogenetic Trees: Why?**

- Understand/expose evolutionary relationships among species
- Identify significant components (genes/proteins/others) that are conserved among genomic sequences
- Understand the (possibly) functional evolution/modification of genomic sequences
- Justify/modify multiple genome alignment
Relations and co-existence among Species: Example

![Gene Tree of Human Globins](http://www.muhlenberg.edu/depts/biology/courses/bio152/BioinformaticsLab/betaglobininfo.html)

**Functional evolution of genes: Example**

**Gene Tree of Human Globins**

**Cross-species Globin Tree**

EMBO reports 3, 12, 1146–1151 (2002)
doi:10.1093/embo-reports/kvf248

Pesce, Bolognesi, Bocedi, Ascenzi, Dewilde, Moens, Hankeln & Burmester

![Gene Tree of Human Globins](http://www.muhlenberg.edu/depts/biology/courses/bio152/BioinformaticsLab/betaglobininfo.html)

**Phylogenetic Tree – How? (Cont.)**

**The problem:**
Given sequence data about genes/species or numbers representing the distance between pairs of genes (or species), find the optimal phylogenetic tree.

**Approaches:**
- Attempt to minimize distance between adjacent nodes.
- Maximize parsimony: Search for the tree that requires minimum number of substitutions to evolve. (For sequence data)
- Maximize likelihood: Uses a probabilistic evolution model and attempts to maximize the probability of the data given the model. (For sequence data)
Constructing Phylogenetic Tree from Distance Data

Input: n items $G_1,...,G_n$ (gene sequences, or species info)

$M$, an $n \times n$ matrix where $M_{ij} = \text{Distance}(G_i, G_j) = d_{ij}$

Output: An edge-weighted tree s.t. tree-distance between every pair of leaves $G_i, G_j$ is $M_{ij}$

Example:

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>6</td>
<td>4</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>8</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>8</td>
<td>0</td>
<td>8</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>0</td>
<td>8</td>
<td>0</td>
</tr>
</tbody>
</table>

© 2019. Hagit Shatkay, CIS

Constructing Tree from Distance Data (cont.)

The UPGMA method: Iterative clustering

(Un-weighted Pair-Groups Method with Arithmetic mean)

Given $n$ items $G_1,...,G_n$, and an $n \times n$ distance matrix $M$

Initialization: Place each item $G_i$ as a leaf in the tree, and create $n$ singleton clusters, $C_1,...,C_n$ where $C_i = \{G_i\}$.

The distance between each pair of clusters $C_i$ and $C_j$ is defined as:

$$D_{ij} = \frac{1}{|C_i| \cdot |C_j|} \sum_{p \in C_i, q \in C_j} d_{pq}$$

Iterate:

• Select clusters $C_i$ and $C_j$ s.t. $D_{ij}$ is minimal.
• Define a new cluster $C_k = C_i \cup C_j$, and remove clusters $C_i$ and $C_j$.
• Define a new node $k$ with children $i$ and $j$; Place it at height $D_{ij}/2$.

Termination: When only 2 clusters $C_i$ and $C_j$ remain, join them and place the root at height $D_{ij}/2$.

© 2019. Hagit Shatkay, CIS

Constructing the Tree using UPGMA

Example:

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>8</td>
<td>6</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>8</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>8</td>
<td>0</td>
<td>8</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>0</td>
<td>8</td>
<td>0</td>
</tr>
</tbody>
</table>

© 2019. Hagit Shatkay, CIS
Constructing Tree from Distance Data (cont.)

The UPGMA method: Limitations and Assumptions?

Observation: UPGMA produces a tree whose leaves are equal-distant from the root. (All paths from root to leaves are equal)

What if the original tree looked like this?

Molecular-Clock Assumption: UPGMA assumes that sequence divergence occurs at the same rate throughout the tree.

The distance \( M_{ij} = \text{Distance}(G_i, G_j) = d_{ij} \) is an Ultrametric distance

© 2019. Hagit Shatkay, CIS

14
Constructing Tree from Distance Data (cont.)

Neighbor Joining: Iterative clustering
(Without the Molecular Clock Assumption)

• An iterative process for merging nodes into a tree
• Relaxes the molecular clock assumption;
  Assumes Distance Additivity*: The distance between every pair of leaves, \( \text{Distance}(G_i, G_j) \), is equal to the sum of the weights (lengths) of the edges on the path between \( G_i \) and \( G_j \).
• Produces an un-rooted tree.

* Also satisfied/imposed by the UPGMA algorithm

© 2019. Hagit Shatkay, CIS

16

Constructing Tree from Distance Data (cont.)

Neighbor Joining: Basic idea

Given pairwise distances between leaves (a distance matrix)• Find a pair of neighboring leaves with minimum distance between them
• Join them into a cluster
• Recalculate distance between the cluster and all other clusters
• Repeat until the whole tree is joined…

Caveat: Given only the distance matrix - How do we know which leaves are neighbors?

© 2019. Hagit Shatkay, CIS

17

Constructing Tree from Distance Data (cont.)

Neighbor Joining (cont.)

Caveat (cont.): Distance(\( A, C \)) is smallest - while only AB and CD are neighbors...

Identifying neighbors given a distance matrix
(Assuming distance is Additive!)

Theorem (Saitou & Nei, 1987): Let \( l \) be the number of leaves in the tree. For each leaf \( i \), define \( r_i \) as:
\[
    r_i = \sum_{j=1}^{l} \frac{d_{ij}}{1L_{ij} - 2}
\]

A pair of leaves \( i \) and \( j \) are neighboring if and only if the measure:
\[
    D_{ij} = d_{ij} - (r_i + r_j)
\]
is minimal.

© 2019. Hagit Shatkay, CIS

18
Constructing Tree from Distance Data (cont.)

Example (cont.):

\[ d_{ij} = d_{ij} - (r_i + r_j) \]

\[ d_{AC} = 0.3 \quad r_A = 1.4/2 = 0.7 \quad D_{AC} = -1.1 \]
\[ d_{AB} = 0.5 \quad r_B = 2/2 = 1.0 \quad D_{AB} = -1.2 \]
\[ d_{AD} = 0.6 \quad r_D = 2/2 = 1.0 \quad D_{AD} = -1.1 \]
\[ d_{BC} = 0.6 \quad D_{BC} = -1.1 \]
\[ d_{BD} = 0.9 \quad D_{BD} = -1.1 \]
\[ d_{CD} = 0.5 \quad D_{CD} = -1.2 \]

Recalculate \( r_k, r_C, r_D \) and \( D_{kC}, D_{kD} \); Reiterate

The Neighbor Joining Algorithm

Given \( n \) items \( G_1, \ldots, G_n \) and an \( n \times n \) distance matrix \( M \)

Initialization:
- Place each item \( G_i \) as a leaf in the tree, thus creating \( n \) nodes.
- Let \( L \) denote the set of leaves in the tree.
- The initial distance between pairs of nodes is \( d_{ij} = M_{ij} \).

Iterate

- Select a pair of nodes \( i \) and \( j \) in \( L \), s.t. \( D_{ij} \) (defined before) is minimal
- Define a new node \( k \), joining nodes \( i \) and \( j \) with edges \( k_i \) and \( k_j \).
- Update the distances among nodes as follows:
  \[ d_{km} = \frac{d_{im} + d_{jm} - d_{ij}}{2} \text{; for every other leaf } m \text{ in } L. \]
  \[ d_{ik} = \frac{d_{ij} + r_i - r_j}{2} \]
  \[ d_{jk} = d_{ij} - d_{ik} \]
- Add the new node \( k \) to \( L \) and remove nodes \( i \) and \( j \).

Termination: When \( L \) has two remaining nodes \( i \) and \( j \), join them with an edge of length \( d_{ij} \)
Constructing Tree from Distance Data: Conclusion

- Intuitive (relatively) and simple to implement. Fast to run.
- Depend on having a distance matrix:
  - Not affected by small sequence variations/errors
  - BUT – depends on the definition of distance…
- If the original tree/data was ultrametric, the UPGMA algorithm will reconstruct the correct tree.
  - If the data is indeed additive, the neighbor-joining algorithm will reconstruct the correct tree.
  - Otherwise – the tree constructed is not necessarily right…
- Distance-based methods do not reconstruct ancestral sequences – only relationships among sequences/species.

Phylogenetic Tree – How? (Revisited)

The problem:
Given sequence data about genes/species or numbers representing the distance between pairs of genes (or species), find the optimal phylogenetic tree.

Approaches:
- Attempt to minimize distance between adjacent nodes.
  (Clustering, neighbor joining)
- Maximize parsimony. Search for the tree that requires minimum number of mutations to evolve. (For sequence data)
- Maximize likelihood. Uses a probabilistic evolution model and attempts to maximize the probability of the data given the model. (For sequence data)

Constructing Tree from Character Data: Parsimony

Basic Idea:
Input: n items $G_1, \ldots, G_n$ (typically, genomic sequences)
Output: A tree s.t. the number of mutations along the evolutionary tree is minimized.
Example: Input sequences: AAG, AAA, GGA, AGA
Two (of many...) possible phylogenetic trees:

© 2019, Hagit Shatkay, CIS