<table>
<thead>
<tr>
<th>Taxon</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>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>D</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
Interestingly, without polarization Hennig’s method can infer unrooted trees. We can get the tree topology, but be unable to tell paraphyletic from monophyletic groups.

The outgroup method amounts to inferring an unrooted tree and then rooting the tree on the branch that leads to an outgroup.
Inadequacy of logic

Unfortunately, though Hennigian logic is valid we quickly find that we do not have a reliable method of generating accurate homology statements.

The logic is valid, but we don’t know that the premises are true.

In fact, we almost always find that it is impossible for all of our premises to be true.
Character conflict

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Homo sapiens</td>
<td>AGTTCAAGT</td>
</tr>
<tr>
<td>Rana catesbiana</td>
<td>AATTCAAGT</td>
</tr>
<tr>
<td>Drosophila melanogaster</td>
<td>AGTTCAAGC</td>
</tr>
<tr>
<td>C. elegans</td>
<td>AATTCAAGC</td>
</tr>
</tbody>
</table>

The red character implies that either \((Homo + Drosophila)\) is a group (if G is derived) and/or \((Rana + C. elegans)\) is a group.

The green character implies that either \((Homo + Rana)\) is a group (if T is derived) and/or \((Drosophila + C. elegans)\) is a group.

The green and red character cannot both be correct.
<table>
<thead>
<tr>
<th>Taxon</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>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>D</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
Character conflict

Two characters are compatible if they can both be mapped on the same tree so that all of the character states displayed could be homologous.

Incompatible characters are evidence of homoplasy in the data.

Homoplasy literally means the “same change” has occurred more than once in the evolutionary history of the group.

The presence of homoplasy undermines Hennigian analyses.
white = space of all possible matrices
blue = space of matrices with the pattern:

\[
\begin{array}{cccc}
A & B & C & D \\
- & * & * & - \\
\end{array}
\]
red = space of matrices with the pattern:

\[
\begin{array}{cccc}
A & B & C & D \\
- & * & - & * \\
\end{array}
\]
yellow = space
of matrices with
the pattern:

A B C D
--- * *
all eight categories of matrices
blue = space of matrices compatible with tree:

(A, (B, C), D)
blue = space of matrices compatible with tree:

(A, C, (B, D))
blue = space of matrices compatible with tree:

(A, B, (C, D))
Hennigian:
grey = any tree
blue = B+C
red = B+D
yellow = C+D
white = no tree
(conflicting characters)
What can we do if our data end up in the white (character conflict) or grey (uninformative characters only) zone?

• can we detect character conflict?

• is there a logic-based solution to the problem of character conflict?
Detecting character conflict in binary characters

Consider the four possible combinations of states in a two-character matrix.

The characters are incompatible *iff* (when you look across all taxa) you see all four state combinations.

<table>
<thead>
<tr>
<th></th>
<th>Char 1</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>×</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>×</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>×</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>×</td>
</tr>
</tbody>
</table>
What can we do if our data end up in the white (character conflict) or grey (uninformative characters only) zone?

• Can we detect character conflict? Yes

• Is there a logic-based solution to the problem of character conflict?
  – recoding characters?
  – “reciprocal illumination”?

What can we do if our data end up in the white (character conflict) or grey (uninformative characters only) zone?

- Can we detect character conflict? Yes

- Is there a logic-based solution to the problem of character conflict? No, nothing purely based on logic (and the suggestions for culling data to make matrices suitable for logical inference can lead to unsatisfyingly subjective analyses).

- What can we do?

We must have an “error model”
There are many ways to derive estimators, we are going to talk about maximum likelihood estimation:

\[ \theta \in \Theta \]
\[ X \in \mathcal{X} \]
\[ x \sim \Pr(X = x|\theta) \]
\[ \mathcal{L}(\theta) = \Pr(X = x|\theta) \]
\[ \hat{\theta} = \text{arg max} \mathcal{L}(\theta) \]
\[ \theta \in \Theta \]
\[ \Pr(X = x | \theta) \]
\[ x \sim \Pr(X = x | \theta) \]
$x$ represents

<table>
<thead>
<tr>
<th></th>
<th>0000000000</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0000000000</td>
</tr>
<tr>
<td>B</td>
<td>1111111110</td>
</tr>
<tr>
<td>C</td>
<td>1111111110</td>
</tr>
<tr>
<td>D</td>
<td>1111111111</td>
</tr>
</tbody>
</table>
\[ \theta_1 \]

\[ \theta_2 \]

\[ \theta_3 \]

\[ \Pr(x | \theta_1) = 0.00024 \]

\[ \begin{array}{c}
A & 000000000000 \\
B & 11111111110 \\
C & 11111111110 \\
D & 11111111111 \\
\end{array} \]
\[ \Pr(x|\theta_2) = 0.0002 \]
\[ \Pr(x | \theta_3) = 0.00022 \]
\[ \hat{\theta} = \arg \max \mathcal{L}(\theta) \]
ML Estimation

• Flexible form of inference

• Requires a model: \( \Pr(X = x|\theta) \)

Under mild conditions, ML estimation is asymptotically:

• not very biased,

• efficient

How can we come up with a model?