Profile HMM for multiple sequences
Pair HMM

HMM for pairwise sequence alignment, which incorporates affine gap scores.

“Hidden” States

- Match (M)
- Insertion in x (X)
- Insertion in y (Y)

Observation Symbols

- Match (M): \{(a,b)| a,b \text{ in } \Sigma \}.
- Insertion in x (X): \{(a,-)| a \text{ in } \Sigma \}.
- Insertion in y (Y): \{(-,a)| a \text{ in } \Sigma \}.
Alignment: a path $\rightarrow$ a hidden state sequence

\begin{align*}
\begin{array}{cccccccc}
\text{w} & A & T & C & G & T & A & C \\
0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\
\text{v} & A & T & G & T & T & T & A & T \\
0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\
& A & T & C & G & T & - & A & C \\
& M & M & Y & M & M & X & M & M \\
\end{array}
\end{align*}
Pair HMMs
Multiple sequence alignment (Globin family)

Helix
HBA_HUMAN
HBB_HUMAN
MYG_PHYCA
GLB3_CHITP
GLB5_PETMA
LGB2_LUPLU
GLB1_GLYDI

Consensus

Helix
HBA_HUMAN
HBB_HUMAN
MYG_PHYCA
GLB3_CHITP
GLB5_PETMA
LGB2_LUPLU
GLB1_GLYDI

Consensus
Profile model (PSSM)

• A natural probabilistic model for a conserved region would be to specify independent probabilities $e_i(a)$ of observing nucleotide (amino acid) $a$ in position $i$

• The probability of a new sequence $x$ according to this model is

$$P(x \mid M) = \prod_{i=1}^{L} e_i(x_i)$$
Profile / PSSM

• DNA / proteins Segments of the same length L;

• Often represented as Positional frequency matrix;

The DNA-binding helix-turn-helix motif of the CAP family
Searching profiles: inference

- Give a sequence $S$ of length $L$, compute the likelihood ratio of being generated from this profile vs. from background model:
  - $R(S|P) = \prod_{i=1}^{L} \frac{e_i(x_i)}{b_s}$
  - Searching motifs in a sequence: sliding window approach
Match states for profile HMMs

• Match states
  – Emission probabilities
    \[ e_{M_i}(\alpha) \]
Components of profile HMMs

- Insert states \( e_{I_i}(a) \)
  - Emission prob.
    - Usually background distribution \( q_a \).
  - Transition prob.
    - \( M_i \) to \( I_i \), \( I_i \) to itself, \( I_i \) to \( M_{i+1} \)
  - Log-odds score for a gap of length \( k \) (no log-odds from emission)
    \[
    \log a_{M_jI_j} + \log a_{I_jM_{j+1}} + (k - 1) \log a_{I_jI_j}
    \]
Components of profile HMMs

- Delete states
  - No emission prob.
  - Cost of a deletion
    - \( M \rightarrow D, D \rightarrow D, D \rightarrow M \)
    - Each \( D \rightarrow D \) might be different

```
Begin \rightarrow M_j \rightarrow D_j \rightarrow End
```
Full structure of profile HMMs
Deriving HMMs from multiple alignments

• Key idea behind profile HMMs
  – Model representing the consensus for the alignment of sequence from the same family
  – Not the sequence of any particular member

<table>
<thead>
<tr>
<th>Gene</th>
<th>Consensus Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>HBA_HUMAN</td>
<td>...VGA--HAGEY...</td>
</tr>
<tr>
<td>HBB_HUMAN</td>
<td>...V----NVDEV...</td>
</tr>
<tr>
<td>MYG_PHYCA</td>
<td>...VEA--DVAGH...</td>
</tr>
<tr>
<td>GLB3_CHITP</td>
<td>...VKG------D...</td>
</tr>
<tr>
<td>GLB5_PETMA</td>
<td>...VYS--TYETS...</td>
</tr>
<tr>
<td>LGB2_LUPLU</td>
<td>...FNA--NIPKH...</td>
</tr>
<tr>
<td>GLB1_GLYDI</td>
<td>...IAGADNGAGV...</td>
</tr>
<tr>
<td></td>
<td>*** *****</td>
</tr>
</tbody>
</table>
Deriving HMMs from multiple alignments

• Basic profile HMM parameterization
  – Aim: making the higher probability for sequences from the family

• Parameters
  – the probabilities values: trivial if many of independent alignment sequences are given.

\[
a_{kl} = \frac{A_{kl}}{\sum_{l'} A_{kl'}} \quad e_k(a) = \frac{E_k(a)}{\sum_{a'} E_k(a')}
\]

– length of the model: heuristics or systematic way
Sequence conservation: entropy profile of the emission probability distributions
Searching with profile HMMs

• Main usage of profile HMMs
  – Detecting potential sequences in a family
  – Matching a sequence to the profile HMMs
    • Viterbi algorithm or forward algorithm
  – Comparing the resulting probability with random model
    \[ P(x \mid R) = \prod_{i} q_{x_i} \]
Searching with profile HMMs

- Viterbi algorithm (optimal log-odd alignment)

\[
V_j^M(i) = \log \frac{e_M(x_i)}{q_{x_i}} + \max \begin{cases} 
V_{j-1}(i-1) + \log a_{M_{j-1}M_j}, \\
V_{j-1}(i-1) + \log a_{I_{j-1}M_j}, \\
V_{j-1}(i-1) + \log a_{D_{j-1}M_j}; 
\end{cases}
\]

\[
V_j^I(i) = \log \frac{e_I(x_i)}{q_{x_i}} + \max \begin{cases} 
V_{j-1}(i-1) + \log a_{M_{j-1}I_j}, \\
V_{j-1}(i-1) + \log a_{I_{j-1}I_j}, \\
V_{j-1}(i-1) + \log a_{D_{j-1}I_j}; 
\end{cases}
\]

\[
V_j^D(i) = \max \begin{cases} 
V_{j-1}(i) + \log a_{M_{j-1}D_j}, \\
V_{j-1}(i) + \log a_{I_{j-1}D_j}, \\
V_{j-1}(i) + \log a_{D_{j-1}D_j}; 
\end{cases}
\]
Searching with profile HMMs

- **Forward algorithm**: summing over all potent alignments

\[
F_j^M(i) = \log \frac{e_{M_j}(x_i)}{q_{x_i}} + \log[a_{M_{j-1}M_j} \exp(F_{j-1}^M(i-1)) \\
+ a_{I_{j-1}M_j} \exp(F_{j-1}^I(i-1)) + a_{D_{j-1}M_j} \exp(F_{j-1}^D(i-1))]\
\]

\[
F_j^I(i) = \log \frac{e_{I_j}(x_i)}{q_{x_i}} + \log[a_{M_{jI_j}} \exp(F_{j}^M(i-1)) \\
+ a_{I_{jI_j}} \exp(F_{j}^I(i-1)) + a_{D_{jI_j}} \exp(F_{j}^D(i-1))]\
\]

\[
F_j^D(i) = \log[a_{M_{j-1}D_j} \exp(F_{j-1}^M(i)) + a_{I_{j-1}D_j} \exp(F_{j-1}^I(i)) \\
+ a_{D_{j-1}D_j} \exp(F_{j-1}^D(i))]\
\]
Variants for non-global alignments

• Local alignments (flanking model)
  – Emission prob. in flanking states use background values $q_a$.
  – Looping prob. close to 1, e.g. $(1-\eta)$ for some small $\eta$. 
Variants for non-global alignments

- Overlap alignments
  - Only transitions to the first model state are allowed.
  - When expecting to find either present as a whole or absent
  - Transition to first delete state allows missing first residue
Variants for non-global alignments

- Repeat alignments
  - Transition from right flanking state back to random model
  - Can find multiple matching segments in query string
Estimation of prob.

- Maximum likelihood (ML) estimation
  - given observed freq. \( c_{ja} \) of residue \( a \) in position \( j \).

\[
e_{M_j}(a) = \frac{c_{ja}}{\sum_{a'} c_{ja'}}
\]

- Simple pseudocounts
  - \( q_a \): background distribution
  - \( A \): weight factor

\[
e_{M_j}(a) = \frac{c_{ja} + Aq_a}{A + \sum_{a'} c_{ja'}}
\]
Optimal model construction: mark columns

(a) Multiple alignment:

\[
\begin{array}{l}
\text{bat} \quad A \quad G \quad - \quad - \quad - \quad C \\
\text{rat} \quad A \quad - \quad A \quad G \quad - \quad - \\
\text{cat} \quad A \quad G \quad - \quad A \quad A \quad - \\
\text{gnat} \quad - \quad - \quad A \quad A \quad A \quad C \\
\text{goat} \quad A \quad G \quad - \quad - \quad - \quad C \\
\end{array}
\]

(b) Profile-HMM architecture:

(c) Observed emission/transition counts

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>-</td>
<td>4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>C</td>
<td>-</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>G</td>
<td>-</td>
<td>0</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>T</td>
<td>-</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>A</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>G</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>T</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>M-M</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>M-D</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>M-I</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>I-M</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>I-D</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>I-I</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>D-M</td>
<td>-</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>D-D</td>
<td>-</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>D-I</td>
<td>-</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>
Optimal model construction

• MAP (match-insert assignment)
  – Recursive calculation of a number $S_j$
    • $S_j$: log prob. of the optimal model for alignment up to and including column $j$, assuming $j$ is marked.
    • $S_j$ is calculated from $S_i$ and summed log prob. between $i$ and $j$.
    • $T_{ij}$: summed log prob. of all the state transitions between marked $i$ and $j$.
      \[
      T_{ij} = \sum_{x,y \in \{M,D,I\}} c_{xy} \log a_{xy}
      \]
      – $c_{xy}$ are obtained from partial state paths implied by marking $i$ and $j$. 

Optimal model construction

- Algorithm: MAP model construction
  - Initialization:
    - $S_0 = 0$, $M_{L+1} = 0$.
  - Recurrence: for $j = 1, \ldots, L+1$:
    \[
    S_j = \max_{0 \leq i < j} S_i + T_{ij} + M_j + I_{i+1,j-1} + \lambda;
    \]
    \[
    \sigma_j = \arg \max_{0 \leq i < j} S_i + T_{ij} + M_j + I_{i+1,j-1} + \lambda;
    \]
  - Traceback: from $j = \sigma_{L+1}$, while $\sigma_j > 0$:
    - Mark column $j$ as a match column
    - $j = \sigma_j$. 
Weighting training sequences

• Input sequences are random?
• “Assumption: all examples are independent samples” might be incorrect
• Solutions
  – Weight sequences based on similarity
Weighting training sequences

• Simple weighting schemes derived from a tree
  – Phylogenetic tree is given.
  – [Thompson, Higgins & Gibson 1994b]

\[ \Delta w_i = t_n \frac{w_i}{\sum_{\text{leaves } k \text{ below } n} w_k} \]
Weighting training sequences

\[ t_4 = 8 \]
\[ t_3 = 5 \]
\[ t_2 = 2 \]
\[ t_1 = 2 \]

\[ t_5 = 3 \]
\[ t_6 = 3 \]

\[ 1 \]
\[ 2 \]
\[ 3 \]
\[ 4 \]

\[ w_1 : w_2 : w_3 : w_4 = 35 : 35 : 50 : 64 \]

\[ I_1 : I_2 : I_3 : I_4 = 20 : 20 : 32 : 47 \]
Multiple alignment by training profile HMM

- Sequence profiles could be represented as probabilistic models like profile HMMs.
  - Profile HMMs could simply be used in place of standard profiles in progressive or iterative alignment methods.
  - ML methods for building (training) profile HMM (described previously) are based on multiple sequence alignment.
  - Profile HMMs can also be trained from initially unaligned sequences using the Baum-Welch (EM) algorithm
Multiple alignment by profile HMM training-
Multiple alignment with a known profile HMM

• Before we estimate a model and a multiple alignment simultaneously, we consider as simpler problem: derive a multiple alignment from a known profile HMM model.
  - This can be applied to align a large member of sequences from the same family based on the HMM model built from the (seed) multiple alignment of a small representative set of sequences in the family.
Multiple alignment with a known profile HMM

- Align a sequence to a profile HMM → Viterbi algorithm
- Construction a multiple alignment just requires calculating a Viterbi alignment for each individual sequence.
  - Residues aligned to the same match state in the profile HMM should be aligned in the same columns.
Multiple alignment with a known profile HMM

- Given a preliminary alignment, HMM can align additional sequences.
Multiple alignment with a known profile
HMM

<table>
<thead>
<tr>
<th>Position</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>insert</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>D</td>
<td>LS</td>
<td>H</td>
<td>G</td>
<td>S</td>
<td>A</td>
<td>Q</td>
<td></td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>E</td>
<td>S</td>
<td>F</td>
<td>G</td>
<td>D</td>
<td>LSTPDAV</td>
<td>M</td>
<td>G</td>
<td>N</td>
<td>P</td>
<td>K</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>D</td>
<td>R</td>
<td>F</td>
<td>K</td>
<td>H</td>
<td>LKTEAEM</td>
<td>K</td>
<td>A</td>
<td>S</td>
<td>E</td>
<td>D</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>T</td>
<td>Q</td>
<td>F</td>
<td>A</td>
<td>G</td>
<td>KDLESI</td>
<td>K</td>
<td>G</td>
<td>T</td>
<td>A</td>
<td>P</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>P</td>
<td>K</td>
<td>F</td>
<td>K</td>
<td>G</td>
<td>LTTADQL</td>
<td>K</td>
<td>K</td>
<td>S</td>
<td>A</td>
<td>D</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>S</td>
<td>F</td>
<td>L</td>
<td>K</td>
<td></td>
<td>GTSEVP</td>
<td>Q</td>
<td>N</td>
<td>N</td>
<td>P</td>
<td>E</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>G</td>
<td>F</td>
<td>S</td>
<td>G</td>
<td>AS</td>
<td></td>
<td>-</td>
<td>-</td>
<td>D</td>
<td>P</td>
<td>G</td>
</tr>
</tbody>
</table>
Multiple alignment with a known profile HMM

- Important difference with other MSA programs
  - Viterbi path through HMM identifies inserts
  - Profile HMM does not align inserts
  - Other multiple alignment algorithms align the whole sequences.
Profile HMM training from unaligned sequences

- Harder problem
  - estimating both a model and a multiple alignment from initially unaligned sequences.
  - Initialization: Choose the length of the profile HMM and initialize parameters.
  - Training: estimate the model using the Baum-Welch algorithm (iteratively).
  - Multiple Alignment: Align all sequences to the final model using the Viterbi algorithm and build a multiple alignment as described in the previous section.
Profile HMM training from unaligned sequences

• Initial Model
  – The only decision that must be made in choosing an initial structure for Baum-Welch estimation is the length of the model $M$.
  – A commonly used rule is to set $M$ be the average length of the training sequence.
  – We need some randomness in initial parameters to avoid local maxima.
Multiple alignment by profile HMM training

• Avoiding Local maxima
  – Baum-Welch algorithm is guaranteed to find a LOCAL maxima.
    • Models are usually quite long and there are many opportunities to get stuck in a wrong solution.

  – Solution
    • Start many times from different initial models.
    • Use some form of stochastic search algorithm, e.g. simulated annealing.
Multiple alignment by profile HMM - similar to Gibbs sampling

• The ‘Gibbs sampler’ algorithm described by Lawrence et al.[1993] has substantial similarities.
  – The problem was to simultaneously find the motif positions and to estimate the parameters for a consensus statistical model of them.
  – The statistical model used is essentially a profile HMM with no insert or delete states.
Multiple alignment by profile HMM training-Model surgery

• We can modify the model after (or during) training a model by manually checking the alignment produced from the model.
  – Some of the match states are redundant
  – Some insert states absorb too many sequences
• Model surgery
  – If a match state is used by less than $\frac{1}{2}$ of training sequences, delete its module (match-insert-delete states)
  – If more than $\frac{1}{2}$ of training sequences use a certain insert state, expand it into $n$ new modules, where $n$ is the average length of insertions
  – ad hoc, but works well
Phylo-HMMs: model multiple alignments of syntenic sequences

• A phylo-HMM is a probabilistic machine that generates a multiple alignment, column by column, such that each column is defined by a phylogenetic model.

• Unlike single-sequence HMMs, the emission probabilities of phylo-HMMs are complex distributions defined by phylogenetic models.
Applications of Phylo-HMMs

- Improving phylogenetic modeling that allow for variation among sites in the rate of substitution (Felsenstein & Churchill, 1996; Yang, 1995)
- Protein secondary structure prediction (Goldman et al., 1996; Thorne et al., 1996)
- Detection of recombination from DNA multiple alignments (Husmeier & Wright, 2001)
- Recently, comparative genomics (Siepel, et al. Haussler, 2005)
Phylo-HMMs: combining phylogeny and HMMs

• Molecular evolution can be viewed as a combination of two Markov processes
  – One that operates in the dimension of space (along a genome)
  – One that operates in the dimension of time (along the branches of a phylogenetic tree)

• Phylo-HMMs model this combination
Single-sequence HMM

A

\[ X = \text{TAACGGCAGA} \ldots \]

Phylo-HMM

B

\[ X = \text{TTAGGCAAGG} \text{ AAGGCGCCGA} \ldots \]
Phylogenetic models

• Stochastic process of substitution that operates independently at each site in a genome
• A character is first drawn at random from the background distribution and assigned to the root of the tree; character substitutions then occur randomly along the tree branches, from root to leaves
• The characters at the leaves define an alignment column
Phylogenetic Models

- The different phylogenetic models associated with the states of a phylo-HMM may reflect different overall rates of substitution (e.g. in conserved and non-conserved regions), different patterns of substitution or background distributions, or even different tree topologies (as with recombination).
Phylo-HMMs: Formal Definition

• A phylo-HMM is a 4-tuple $\theta = (S, \psi, A, b)$:
  - $S = \{s_1, \square, s_M\}$: set of hidden states
  - $\psi = \{\psi_1, \square, \psi_M\}$: set of associated phylogenetic models
  - $A = \{a_{j,k}\} \ (1 \leq j, k \leq M)$: transition probabilities
  - $b = (b_1, \square, b_M)$: initial probabilities
The Phylogenetic Model

• \( \psi_j = (Q_j, \pi_j, \tau_j, \beta_j) \):
  - \( Q_j \) : substitution rate matrix
  - \( \pi_j \) : background frequencies
  - \( \tau_j \) : binary tree
  - \( \beta_j \) : branch lengths
The Phylogenetic Model

• The model is defined with respect to an alphabet $\Sigma$ whose size is denoted $d$
• The substitution rate matrix has dimension $d \times d$
• The background frequencies vector has dimension $d$
• The tree has $n$ leaves, corresponding to $n$ extant taxa
• The branch lengths are associated with the tree
Probability of the Data

- Let $X$ be an alignment consisting of $L$ columns and $n$ rows, with the $i^{\text{th}}$ column denoted $X_i$
- The probability that column $X_i$ is emitted by state $s_j$ is simply the probability of $X_i$ under the corresponding phylogenetic model, $P(X_i \mid \psi_j)$
- This is the likelihood of the column given the tree, which can be computed efficiently using Felsenstein’s “pruning” algorithm (which we will describe in later lectures)
Substitution Probabilities

• Felsenstein’s algorithm requires the conditional probabilities of substitution for all bases \( a, b \in \Sigma \) and branch lengths \( t \in \beta_j \).
• The probability of substitution of a base \( b \) for a base \( a \) along a branch of length \( t \), denoted \( P(b \mid a, t, \psi_j) \), is based on a continuous-time Markov model of substitution, defined by the rate matrix \( Q_j \).
Substitution Probabilities

In particular, for any given non-negative value $t$, the conditional probabilities $P(b \mid a, t, \psi_j)$ for all $a, b \in \Sigma$ are given the $d \times d$ matrix $P_j(t) = \exp(Q_j t)$, where

$$\exp(Q_j t) = \sum_{k=0}^{\infty} \frac{(Q_j t)^k}{k!}$$
Example: HKY model

\[
Q_j = \begin{pmatrix}
- & \pi_{C,j} & \kappa_j \pi_{G,j} & \pi_{T,j} \\
\pi_{A,j} & - & \pi_{G,j} & \kappa_j \pi_{T,j} \\
\kappa_j \pi_{A,j} & \pi_{C,j} & - & \pi_{T,j} \\
\pi_{A,j} & \kappa_j \pi_{C,j} & \pi_{G,j} & -
\end{pmatrix}
\]

\[
\pi_j = (\pi_{A,j}, \pi_{C,j}, \pi_{G,j}, \pi_{T,j})
\]

\(\kappa_j\) represents the transition/transversion rate ratio for \(\psi_j\)

‘-’s indicate quantities required to normalize each row.
State sequences in Phylo-HMMs

- A state sequence through the phylo-HMM is a sequence $\phi = (\phi_1, \square, \phi_L)$ such that $\phi_i \in S \ \forall 1 \leq i \leq L$
- The joint probability of a path and and alignment is

$$P(\phi, X \mid \theta) = \beta_{\phi_1} P(X_1 \mid \psi_{\phi_1}) \prod_{i=2}^{L} a_{\phi_{i-1}\phi_i} P(X_i \mid \psi_{\phi_i})$$
Phylo-HMMs

• The likelihood is given by the sum over all paths (forward algorithm)

\[ P(X \mid \theta) = \sum_{\phi} P(\phi, X \mid \theta) \]

• The maximum-likelihood path is (Vertebi’s)

\[ \phi = \text{argmax}_\phi P(\phi, X \mid \theta) \]
Computing the Probabilities

- The likelihood can be computed efficiently using the **forward** algorithm.
- The maximum-likelihood path can be computed efficiently using the **Viterbi** algorithm.
- The **forward** and **backward** algorithms can be combined to compute the posterior probability:

\[ P(\phi_i = j \mid X, \theta) \]
Higher-order Markov Models for Emissions

• It is common with gene-finding HMMs to condition the emission probability of each observation on the observations that immediately precede it in the sequence.

• For example, in a 3-rd-codon-position state, the emission of a base $x_i$=“A” might have a fairly high probability if the previous two bases are $x_{i-2}$=“G” and $x_{i-1}$=“A” (GAA=Glu), but should have zero probability if the previous two bases are $x_{i-2}$=“T” and $x_{i-1}$=“A” (TAA=stop).
Higher-order Markov Models for Emission

- Considering the $N$ observations preceding each $x_i$ corresponds to using an $N^{th}$ order Markov model for emissions
- An $N^{th}$ order model for emissions is typically parameterized in terms of $(N+1)$-tuples of observations, and conditional probabilities are computed as

$$P(x_i|x_{i-N}, \ldots, x_{i-1}) = \frac{P(x_{i-N}, \ldots, x_{i-1}, x)}{\sum_y P(x_{i-N}, \ldots, x_{i-1}, y)}$$
N\textsuperscript{th} Order Phylo-HMMs

Probability of the N-tuple

\[ P(X_i|X_{i-N+1}, \ldots, X_{i-1}) = \frac{P(X_{i-N+1}, \ldots, X_{i-1}, X_i)}{\sum_Y P(X_{i-N+1}, \ldots, X_{i-1}, Y)} \]

Sum over all possible alignment columns Y
(can be calculated efficiently by a slight modification of Felsenstein’s “pruning” algorithm)