Pair HMMs and Profile HMMs

COMPSCI 260 – Spring 2016
An HMM is a *discrete-time stochastic machine* $M=(Q, \alpha, P_t, P_e)$ consisting of the following:

- a finite set of states, $Q=\{q_0, q_1, \ldots, q_m\}$
- a finite alphabet $\alpha=\{s_0, s_1, \ldots, s_n\}$
- a transition distribution $P_t: Q \times Q \mapsto \mathbb{R}$
- an emission distribution $P_e: Q \times \alpha \mapsto \mathbb{R}$

**An Example**

$M_1=({q_0, q_1, q_2}, \{Y, R\}, P_t, P_e)$

$P_t=\{(q_0, q_1, 1), (q_1, q_1, 0.8), (q_1, q_2, 0.15), (q_1, q_0, 0.05), (q_2, q_2, 0.7), (q_2, q_1, 0.3)\}$

$P_e=\{(q_1, Y, 1), (q_1, R, 0), (q_2, Y, 0), (q_2, R, 1)\}$
Three views of an HMM
Questions we can address with an HMM

INPUT

The HMM model \( M \): \( Q = \{ q_0, q_1, \ldots, q_m \} \); \( P_t(q_j | q_i) \); \( P_e(s_j | q_i) \)

A sequence \( S = x_0 x_1 \ldots x_{L-1} \) and a path \( \phi = y_0 y_1 \ldots y_{L+1} \)

```
CGATATTGATTCTACGCGCGTATACTAGCTTATCTGATC
01111111122222222111111122221111111122221111110
```

OUTPUT

What is the probability of generating sequence \( S \) from path \( \phi \) according to the model \( M \)? \( P(S | \phi, M) \)

\[
P(S | \phi) = \prod_{i=0}^{L-1} P_e(x_i | y_{i+1})
\]

emission prob.
Questions we can address with an HMM

INPUT

The HMM model $M$: $Q=\{q_0, q_1, \ldots, q_m\}$; $P_t(q_j | q_i)$; $P_e(s_j | q_i)$

A sequence $S = x_0x_1 \ldots x_{L-1}$ and a path $\phi = y_0y_1 \ldots y_{L+1}$

OUTPUT

What is the joint probability of sequence $S$ and path $\phi$ according to the model $M$? $P(S, \phi | M)$

\[
P(S, \phi) = P(S | \phi)P(\phi)
\]

\[
P(S | \phi) = \prod_{i=0}^{L-1} P_e(x_i | y_{i+1})
\]

$P(\phi) = \prod_{i=0}^{L} P_t(y_{i+1} | y_i)$

CGATATTTCGATTCTACGCATGCATGCTATCTGATC

01111111222222211111112222111111122221111110

diagram: emission prob. and transition prob.
Questions we can address with an HMM

**INPUT**

The HMM model $M$: $Q=\{q_0, q_1, \ldots, q_m\}; P_t(q_j|q_i); P_e(s_j|q_i)$

A sequence $S$:

CGATATTCTACGCGCGTATACTAGCTTATCTGATC

**OUTPUT**

What is the most probable path for generating sequence $S$ according to the model $M$? — “Decoding”

$$\phi_{max} = \arg\max_{\phi} P(\phi|S, M)$$
“Decoding” with an HMM – Viterbi decoding

\[
\phi_{\text{max}} = \arg\max_{\phi} P(\phi | S) = \arg\max_{\phi} \frac{P(\phi, S)}{P(S)}
\]

\[
= \arg\max_{\phi} P(\phi, S)
\]

\[
= \arg\max_{\phi} P(S | \phi)P(\phi)
\]

\[
P(S | \phi) = \prod_{i=0}^{L-1} P_e(x_i | y_{i+1})
\]

\[
P(\phi) = \prod_{i=0}^{L} P_t(y_{i+1} | y_i)
\]

\[
\phi_{\text{max}} = \arg\max_{\phi} P_t(q_0 | y_L) \prod_{i=0}^{L-1} P_e(x_i | y_{i+1})P_t(y_{i+1} | y_i)
\]

\[S = x_0x_1...x_{L-1}\]

\[\phi = y_0y_1...y_{L+1}\]
“Decoding” with an HMM

- Viterbi gives us two things:
  - The “best” parse: $\phi_{\text{max}} = \arg\max_\phi P(\phi|S)$
  - The joint probability: $P(\phi_{\text{max}}, S)$

- What if we are interested in the state that generated a particular character?
  \[ P(y_k = q_i | S) = \frac{P(S, y_k = q_i)}{P(S)} \]

- What if we are interested in the marginal probability of emitting $S$, regardless of the path?
  \[ P(S) \]

- We can compute these using the **Forward** and **Backward** algorithms, and “posterior” decoding
"Decoding" with an HMM - Posterior decoding

Posterior decoding:

\[ P(y_k = q_i | S) = \frac{P(S, y_k = q_i)}{P(S)} = \frac{F(i, k)B(i, k)}{P(S)} \]
Training an HMM with labeled sequences: $S, \phi$ given

```
CGATATTGCATTCTACGCAGCTATACTAGCTTTATCTGATC
0111111112222222111111122222111111122222111110
```

<table>
<thead>
<tr>
<th>transitions from state</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0 (0%)</td>
<td>1 (100%)</td>
<td>0 (0%)</td>
</tr>
<tr>
<td>1</td>
<td>1 (4%)</td>
<td>21 (84%)</td>
<td>3 (12%)</td>
</tr>
<tr>
<td>2</td>
<td>0 (0%)</td>
<td>3 (20%)</td>
<td>12 (80%)</td>
</tr>
</tbody>
</table>

\[
a_{i,j} = \frac{A_{i,j}}{\sum_{h=0}^{|Q|-1} A_{i,h}}
\]

<table>
<thead>
<tr>
<th>emissions in state</th>
<th>A</th>
<th>C</th>
<th>G</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6 (24%)</td>
<td>7 (28%)</td>
<td>5 (20%)</td>
<td>7 (28%)</td>
</tr>
<tr>
<td>2</td>
<td>3 (20%)</td>
<td>3 (20%)</td>
<td>2 (13%)</td>
<td>7 (47%)</td>
</tr>
</tbody>
</table>

\[
e_{i,k} = \frac{E_{i,k}}{\sum_{h=0}^{|\Sigma|-1} E_{i,h}}
\]
Training an HMM with unlabeled sequences: only $S$ given

**INPUT:** A set of sequences $S$ generated by the HMM; $Q=\{q_0, q_1, \ldots, q_m\}$

**OUTPUT:** The parameters of the HMM: $P_t(q_j \mid q_i); P_e(s_j \mid q_i)$

**Two Solutions:**

1. **Viterbi Training:** Start with random HMM parameters. Use Viterbi to find the most probable path for each training sequence, and then label the sequence with that path. Use labeled sequence training on the resulting set of sequences and paths. Iterate until Viterbi paths do not change.

2. **Baum-Welch Training:** Start with random HMM parameters. Use posterior decoding to compute the ‘forward’ and ‘backward’ probabilities. Sum over all possible paths (rather than the single most probable one) to estimate expected counts $A_{i,j}$ and $E_{i,k}$; then use the same formulas as for labeled sequence training on these expected counts. Iterate until the change in $P(S \mid M) < \varepsilon$
HMMs for gene prediction
A *Pair HMM* is an HMM which has *two output channels* rather than one; each state can emit a symbol into one or the other (or both) channels.

More general Pair HMMs can have many more states, but those states can all be classified as *insertion states*, *deletion states*, or *match/mismatch states*. 
The most probable state path through the Pair HMM determines the optimal alignment.
Pair HMMs can be used for simultaneous alignment and annotation

A Pair HMM with Functional States

Generalization: Profile HMMs
Profile HMMs application: Pfam protein domains

Pfam 29.0 (December 2015, 16295 entries)

The Pfam database is a large collection of protein families, each represented by multiple sequence alignments and hidden Markov models (HMMs). More...
Profile HMMs application: Pfam protein domains
Position weight matrices (PWMs) (PSSMs)

PWMs are a special case of an HMM:

What are the transition probabilities?
Profile HMMs for protein families

- Consider the PWM for a conserved segment of a protein family

![Profile PWM]

- The profile consists of the frequencies of amino acids at each position

![Profile Diagram]

- However, this type of profile does not allow for gaps (insertions/deletions)
Profile HMMs for protein families

• Adding HMM states that capture insertions:

• For any particular insert state, we may have different transition probabilities for entering it the first time vs. staying in the insert state

• What kind of gap penalty model can Profile HMMs capture?

• Affine gap penalty

• What is the length distribution of the inserted segments (number of elements emitted from an insert state $I_k$)?

• Geometric distribution
Profile HMMs for protein families

- Adding HMM states that capture **deletions**
- One could model deletions as:

\[
\begin{array}{ccccccc}
\text{Begin} & M_1 & M_2 & M_3 & \ldots & M_L & \text{End} \\
\end{array}
\]

- However, arbitrarily long gaps introduce lots of transitions in the model
- Instead, we will introduce **delete states** that do not emit any symbols

\[
\begin{array}{ccccccc}
\text{Begin} & M_1 & M_2 & M_3 & \ldots & M_L & \text{End} \\
\end{array}
\]
Profile HMMs for protein families

- The full Profile HMM model:

  - The model incorporates states for *insertions* and *deletions*, as well as \( L \) *match/mismatch* states
  - Have we finished building the model?
  - No, we have just given the overall *topology* of a profile-HMM, but we still need to decide *how many states* our HMM has (\( L \)), what are the *transition probabilities*, what are the *emission probabilities*. 
Profile HMMs for protein families

• A few important points to keep in mind
  – The delete states emit no characters
  – Emission probabilities exist only for insert and match/mismatch states
  – Transition probabilities between states determine how likely it is to insert or delete an element
Profile HMMs for protein families
Family: AphA_like (PF14557)

HMM logo

HMM logos are one way of visualising profile HMMs. Logos provide a quick overview of the properties of an HMM in a graphical form. You can see a more detailed description of HMM logos and find out how you can interpret them here. More...
**Pfam** protein domains

**Family: AphA_like (PF14557)**

HMM logos are one way of visualising profile HMMs. Logos provide a quick overview of the properties of an HMM in a graphical form. You can see a more detailed description of HMM logos and find out how you can interpret them [here]. More...

Column height is relative entropy (or information content) at that position:

\[
\sum \frac{p_i \cdot \log(p_i/q_i)}{q_i}
\]

- Information content (bits)
- Letter heights are proportional to frequency
- Insertion probability
Pfam protein domains
Given the topology of a profile HMM, what algorithm(s) do we use to train the model?

- It depends on what kind of training data is available: labeled or unlabeled
  - Labeled training data: counting (maximum likelihood approach)
  - Unlabeled training data: the Baum-Welch algorithm
Profile HMMs – training using labeled (?) data

- How do we pick the length of the HMM (i.e., how many match states do we have in the profile)?
- One heuristic is to only include those columns that have amino acids in at least half of the sequences.

- For example, in the above alignment, there would be match states for each column except for the fourth and fifth columns.
- Another reasonable guess is the mean or median length of all the input sequences.

VGA---HAGEY
V------NVDEV
VEA--DVAGH
VKG-------D
VYS--TYETS
FNA--NIPKH
IAGADNGAGY

- How do we pick the length of the HMM (i.e., how many match states do we have in the profile)?
- One heuristic is to only include those columns that have amino acids in at least half of the sequences.

- For example, in the above alignment, there would be match states for each column except for the fourth and fifth columns.
- Another reasonable guess is the mean or median length of all the input sequences.

VGA---HAGEY
V------NVDEV
VEA--DVAGH
VKG-------D
VYS--TYETS
FNA--NIPKH
IAGADNGAGY
Profile HMMs – training using labeled (?) data

- How do we pick the length of the HMM (i.e., how many match states do we have in the profile)?
- One heuristic is to only include those columns that have amino acids in at least half of the sequences

- How do we compute emission probabilities?
- Counts – similarly to PWMs (we may want to add pseudocounts)

\[ e_{i,k} = \frac{E_{i,k}}{\sum_{h=0}^{|\alpha|-1} E_{i,h}} \]

- How do we compute transition probabilities?
- Counts

\[ a_{i,j} = \frac{A_{i,j}}{\sum_{h=0}^{|\Omega|-1} A_{i,h}} \]
Profile HMMs – training using **unlabeled** data

- We can use the **Baum-Welch** algorithm
- Note, however, that the topology of the HMM is fixed before learning
- In practice, Profile HMMs are typically learned from aligned sequences
Using Profile HMMs to identify and align new family members

- Given a Profile HMM for a protein family, we can use it to query the GENBANK or some other database to find new family members.
- Given the parameters of the model \(P_t(q_j | q_i)\) and \(P_e(s_j | q_i)\) and a new sequence \(S\), it is possible to determine the most probable sequence of states for generating \(S\) ...
  ... using Viterbi decoding.
- By doing that, we automatically align the new sequence to the other family members.
- It is also possible to determine the origin of a particular amino acid (whether it is from some match column \(j\), or an insertion) ....
  ... using posterior decoding.
- Using either of the two methods, we can select those sequences \(S\) that have large probability of being generated by the given profile.
- But where does the Profile HMM come from?
Chicken and egg problem ...

• Given a *multiple sequence alignment* between members of a protein family, we can build a *Profile HMM* for the family

• Given a *Profile HMM*, we can find the *multiple sequence alignment* between members of the family
We can use an iterative approach

- We start with a single sequence $S$ and use BLAST to get more sequences like $S$
- We use these sequences to construct a profile representation
- We use the profile to search the database again for similar sequences
- The new sequences are used to improve the profile representation and the process is repeated...
- This is the principle behind **PSIBLAST - Position Specific Iterated BLAST**
HMMER is a software suite for protein sequence similarity searches using probabilistic methods. …. Methods are available for searching either a **single** protein sequence, **multiple** protein sequence alignment or profile HMM against a target sequence database, and for searching a **protein sequence against** Pfam...

More information....

http://hmmer.org
HMMER: biosequence analysis using profile hidden Markov models

HMMER is used for searching sequence databases for sequence homologs, and for making sequence alignments. It implements methods using probabilistic models called profile hidden Markov models (profile HMMs).

HMMER is often used together with a profile database, such as Pfam or many of the databases that participate in Interpro. But HMMER can also work with query sequences, not just profiles, just like BLAST. For example, you can search a protein query sequence against a database with phmmr, or do an iterative search with jackhmmr.

HMMER is designed to detect remote homologs as sensitively as possible, relying on the strength of its underlying probability models. In the past, this strength came at significant computational expense, but as of the new HMMER3 project, HMMER is now essentially as fast as BLAST.