Hidden Markov Model (HMM)

Soma Biswas
Department of Electrical Engineering,
Indian Institute of Science, Bangalore.
Introduction

Sequential Data

- Data are sequentially generated according to time or index
- Spatial information along time or index

Examples of sequential data:

- Rainfall measurements on successive days at a location
- Daily values of a currency exchange rate
- Sequence of characters in an English sentence.
- Sequence of nucleotide base pairs along a DNA strand

- Stationary case: data evolves in time, but the distribution from which it is generated remains the same.
- Non-stationary case: the generative distribution itself is evolving with time.

Successful Application Areas of HMM

- On-line handwriting recognition
- Speech recognition
- Gesture recognition
- Language modeling
- Motion video analysis and tracking
- Protein sequence/gene sequence alignment
- Stock price prediction
- ...
What’s HMM?

Hidden Markov Model

Hidden + Markov Model

What is ‘hidden’? What is ‘Markov model’?
Markov Model – Graphical Representation

- Classify a weather into three states
  - State 1: rain or snow
  - State 2: cloudy
  - State 3: sunny

- By carefully examining the weather of some city for a long time, we found following weather change pattern

<table>
<thead>
<tr>
<th>Today</th>
<th>Tomorrow</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Rain/snow</td>
</tr>
<tr>
<td>Rain/Snow</td>
<td>0.4</td>
</tr>
<tr>
<td>Cloudy</td>
<td>0.2</td>
</tr>
<tr>
<td>Sunny</td>
<td>0.1</td>
</tr>
</tbody>
</table>

Assumption: tomorrow weather depends only on today one!

- Visual illustration with diagram
  - Each state corresponds to one observation
  - Sum of outgoing edge weights is one
For many applications, such as financial forecasting, we wish to be able to predict the next value given observations of the previous values.
Markov Model – Definition Contd..

- State transition matrix

\[ A = \begin{bmatrix}
    a_{11} & a_{12} & \cdots & a_{1N} \\
    a_{21} & a_{22} & \cdots & a_{2N} \\
    \vdots & \vdots & \ddots & \vdots \\
    a_{N1} & a_{N2} & \cdots & a_{NN}
\end{bmatrix} \]

- Where

\[ a_{ij} = P(q_t = j \mid q_{t-1} = i), \quad 1 \leq i, j \leq N \]

- With constraints

\[ a_{ij} \geq 0, \quad \sum_{j=1}^{N} a_{ij} = 1 \]

- Initial state probability

\[ \pi_i = P(q_1 = i), \quad 1 \leq i \leq N \]
Markov Model – Sequence Probability

- Conditional probability

\[ P(A, B) = P(A \mid B)P(B) \]

- Sequence probability of Markov model

\[
\begin{align*}
P(q_1, q_2, \ldots, q_T) & \quad \text{Chain rule} \\
= P(q_1)P(q_2 \mid q_1) \cdots P(q_{T-1} \mid q_1, \ldots, q_{T-2})P(q_T \mid q_1, \ldots, q_{T-1}) \\
= P(q_1)P(q_2 \mid q_1) \cdots P(q_{T-1} \mid q_{T-2})P(q_T \mid q_{T-1}) \\
\end{align*}
\]

\[ 1^{st} \text{ order Markov assumption} \]
Example

- Question: What is the probability that the weather for the next 7 days will be “sun-sun-rain-rain-sun-cloudy-sun” when today is sunny?

\[ S_1 : \text{rain}, \quad S_2 : \text{cloudy}, \quad S_3 : \text{sunny} \]

\[ = 1.536 \times 10^{-4} \]
Example

- Question: What is the probability that the weather for the next 7 days will be “sun-sun-rain-rain-sun-cloudy-sun” when today is sunny?
  
  \[ S_1 : \text{rain}, \quad S_2 : \text{cloudy}, \quad S_3 : \text{sunny} \]

  \[
P(O \mid \text{model}) = P(S_3, S_3, S_3, S_1, S_1, S_3, S_2, S_3 \mid \text{model})
  = P(S_3) \cdot P(S_3 \mid S_3) \cdot P(S_3 \mid S_3) \cdot P(S_1 \mid S_3)
  \cdot P(S_1 \mid S_1)P(S_3 \mid S_1)P(S_2 \mid S_3)P(S_3 \mid S_2)
  = \pi_3 \cdot a_{33} \cdot a_{33} \cdot a_{31} \cdot a_{11} \cdot a_{32} \cdot a_{23}
  = 1 \cdot (0.8)(0.8)(0.1)(0.4)(0.3)(0.1)(0.2)
  = 1.536 \times 10^{-4}
\]
Markov Model: State Probability

Step 1: Work out how to compute $P(Q)$ for any path $Q = q_0 \ q_1 \ q_2 \ q_3 \ldots \ q_t$

Given we know the start state $q_0$

$$P(q_0 \ q_1 \ldots \ q_t) = P(q_0 \ q_1 \ldots \ q_{t-1}) \ P(q_t | q_0 \ q_1 \ldots \ q_{t-1})$$

$$= P(q_0 \ q_1 \ldots \ q_{t-1}) \ P(q_t | q_{t-1})$$

$$= P(q_1 | q_0) \ P(q_2 | q_1) \ldots \ P(q_t | q_{t-1})$$

Step 2: Use this knowledge to get $P(q_t = s)$

$$P(q_t = s) = \sum_{Q \in \text{Paths of length } t \text{ that end in } s} P(Q)$$

Computation is exponential in $t$
Another Way

- For each state $s_i$, define
  $$p_t(i) = \text{Prob. state is } s_i \text{ at time } t$$
  $$= P(q_t = s_i)$$

- Easy to do inductive definition

\[ \forall i \quad p_0(i) = \begin{cases} 1 & \text{if } s_i \text{ is the start state} \\ 0 & \text{otherwise} \end{cases} \]

\[ \forall j \quad p_{t+1}(j) = P(q_{t+1} = s_j) = \sum_{i=1}^{N} P(q_{t+1} = s_j \land q_t = s_i) = \sum_{i=1}^{N} P(q_{t+1} = s_j \mid q_t = s_i) P(q_t = s_i) = \sum_{i=1}^{N} a_{ij} p_t(i) \]

- Computation is simple.
- Just fill in this table in this order:

<table>
<thead>
<tr>
<th>$t$</th>
<th>$p_t(1)$</th>
<th>$p_t(2)$</th>
<th>...</th>
<th>$p_t(N)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>...</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$t_{final}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Cost of computing $P_t(i)$ for all states $S_i$ is now $O(t N^2)$
HMM - Example

**Sequence generating algorithm**

- Step 1: Pick initial urn according to some random process
- Step 2: Randomly pick a ball from the urn and then replace it
- Step 3: Select another urn according to a random selection process
- Step 4: Repeat steps 2 and 3

- N urns containing color balls
- M distinct colors
- Each urn contains different number of color balls
HMM – What is hidden?

- We can just see the chosen balls
- We can’t see which urn is selected at a time
- So, urn selection (state transition) information is hidden
HMM - Definition

- Notation: \( \lambda = (A, B, \Pi) \)
  1. \( N \): Number of states
  2. \( M \): Number of symbols observable in states
     \[
     V = \{v_1, \cdots, v_M\}
     \]
  3. \( A \): State transition probability distribution
     \[
     A = \{a_{ij}\}, \quad 1 \leq i, j \leq N
     \]
  4. \( B \): Observation symbol probability distribution
     \[
     B = \{b_i(v_k)\}, \quad 1 \leq i \leq N, 1 \leq j \leq M
     \]
  5. \( \Pi \): Initial state distribution
     \[
     \pi_i = P(q_1 = i), \quad 1 \leq i \leq N
     \]
HMM – Dependency Structure

- 1-st order Markov assumption of transition

\[ P(q_t \mid q_1, q_2, \ldots, q_{t-1}) = P(q_t \mid q_{t-1}) \]

- Conditional independency of observation parameters

\[ P(X_t \mid q_t, X_1, \ldots, X_{t-1}, q_1, \ldots, q_{t-1}) = P(X_t \mid q_t) \]
HMM Example Revisited

- # of states: N=3
- # of observation: M=3
  \[ V = \{ R, G, B \} \]
- Initial state distribution
  \[ \pi = \{ P (q_1 = i) \} = [1, 0, 0] \]
- State transition probability distribution
  \[
  A = \{ a_{ij} \} =
  \begin{bmatrix}
  0.6 & 0.2 & 0.2 \\
  0.1 & 0.3 & 0.6 \\
  0.3 & 0.1 & 0.6 \\
  \end{bmatrix}
  \]
- Observation symbol probability distribution
  \[
  B = \{ b_i(v_k) \} =
  \begin{bmatrix}
  3/6 & 2/6 & 1/6 \\
  1/6 & 3/6 & 2/6 \\
  1/6 & 1/6 & 4/6 \\
  \end{bmatrix}
  \]
HMM – Three Problems

- What is the probability of generating an observation sequence?
  - Model evaluation
  \[ P( X = x_1, x_2, \ldots, x_T \mid \lambda) = ? \]

- Given observation, what is the most probable transition sequence?
  - Segmentation or path analysis

- How do we estimate or optimize the parameters of an HMM?
  - Training problem
  \[ P( X \mid \lambda = (A, B, \pi)) < P( X \mid \lambda' = (A', B', \pi')) \]
Another HMM Example

Start randomly in state 1 or 2
Choose one of the output symbols in each state at random.

N = 3  
M = 3  
\( \pi_1 = \frac{1}{2} \)  
\( \pi_2 = \frac{1}{2} \)  
\( \pi_3 = 0 \)

\( a_{11} = 0 \)  
\( a_{12} = \frac{1}{3} \)  
\( a_{13} = \frac{2}{3} \)

\( a_{21} = \frac{1}{3} \)  
\( a_{22} = 0 \)  
\( a_{23} = \frac{2}{3} \)

\( a_{31} = \frac{1}{3} \)  
\( a_{32} = \frac{1}{3} \)  
\( a_{33} = \frac{1}{3} \)

\( b_1 (X) = \frac{1}{2} \)  
\( b_1 (Y) = \frac{1}{2} \)  
\( b_1 (Z) = 0 \)

\( b_2 (X) = 0 \)  
\( b_2 (Y) = \frac{1}{2} \)  
\( b_2 (Z) = \frac{1}{2} \)

\( b_3 (X) = \frac{1}{2} \)  
\( b_3 (Y) = 0 \)  
\( b_3 (Z) = \frac{1}{2} \)
HMM

N = 3  
M = 3  
\(\pi_1 = \frac{1}{2}\)  
\(\pi_2 = \frac{1}{2}\)  
\(\pi_3 = 0\)  

\(a_{11} = 0\)  
\(a_{12} = \frac{1}{3}\)  
\(a_{13} = \frac{2}{3}\)  
\(a_{22} = 0\)  
\(a_{23} = \frac{1}{3}\)  
\(a_{32} = \frac{2}{3}\)  
\(a_{33} = \frac{1}{3}\)  

\(b_1(X) = \frac{1}{2}\)  
\(b_1(Y) = \frac{1}{2}\)  
\(b_1(Z) = 0\)  
\(b_2(X) = 0\)  
\(b_2(Y) = \frac{1}{2}\)  
\(b_2(Z) = \frac{1}{2}\)  
\(b_3(X) = \frac{1}{2}\)  
\(b_3(Y) = 0\)  
\(b_3(Z) = \frac{1}{2}\)

Start randomly in state 1 or 2  
Choose one of the output symbols in each state at random.  
Let's generate a sequence of observations:

<table>
<thead>
<tr>
<th>q0=</th>
<th>O0=</th>
<th>q1=</th>
<th>O1=</th>
<th>q2=</th>
<th>O2=</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Start randomly in state 1 or 2
Choose one of the output symbols in each state at random.
Let’s generate a sequence of observations:

\[ N = 3 \]
\[ M = 3 \]
\[ \pi_1 = \frac{1}{2} \]
\[ \pi_2 = \frac{1}{2} \]
\[ \pi_3 = 0 \]

\[ a_{11} = 0 \]
\[ a_{12} = \frac{1}{3} \]
\[ a_{13} = \frac{2}{3} \]
\[ a_{12} = \frac{1}{3} \]
\[ a_{22} = 0 \]
\[ a_{13} = \frac{2}{3} \]
\[ a_{13} = \frac{1}{3} \]
\[ a_{32} = \frac{1}{3} \]
\[ a_{13} = \frac{1}{3} \]

\[ b_1 (X) = \frac{1}{2} \]
\[ b_1 (Y) = \frac{1}{2} \]
\[ b_1 (Z) = 0 \]
\[ b_2 (X) = 0 \]
\[ b_2 (Y) = \frac{1}{2} \]
\[ b_2 (Z) = \frac{1}{2} \]
\[ b_3 (X) = \frac{1}{2} \]
\[ b_3 (Y) = 0 \]
\[ b_3 (Z) = \frac{1}{2} \]
N = 3
M = 3

$\pi_1 = \frac{1}{2}$  $\pi_2 = \frac{1}{2}$  $\pi_3 = 0$

$a_{11} = 0$  $a_{12} = \frac{1}{3}$  $a_{13} = \frac{2}{3}$
$a_{12} = \frac{1}{3}$  $a_{22} = 0$  $a_{13} = \frac{2}{3}$
$a_{13} = \frac{1}{3}$  $a_{32} = \frac{1}{3}$  $a_{13} = \frac{1}{3}$
$b_1 (X) = \frac{1}{2}$  $b_1 (Y) = \frac{1}{2}$  $b_1 (Z) = 0$
$b_2 (X) = 0$  $b_2 (Y) = \frac{1}{2}$  $b_2 (Z) = \frac{1}{2}$
$b_3 (X) = \frac{1}{2}$  $b_3 (Y) = 0$  $b_3 (Z) = \frac{1}{2}$

Start randomly in state 1 or 2
Choose one of the output symbols in each state at random.
Let’s generate a sequence of observations:

Goto $S_3$ with probability $2/3$ or $S_2$ with prob. $1/3$
Start randomly in state 1 or 2
Choose one of the output symbols in each state at random.
Let’s generate a sequence of observations:

\[ \begin{align*}
N &= 3 \\
M &= 3 \\
\pi_1 &= \frac{1}{2} \\
\pi_2 &= \frac{1}{2} \\
\pi_3 &= 0 \\
a_{11} &= 0 \\
a_{12} &= \frac{1}{3} \\
a_{13} &= \frac{1}{3} \\
a_{22} &= 0 \\
a_{23} &= \frac{1}{3} \\
a_{32} &= \frac{1}{3} \\
a_{33} &= \frac{1}{3} \\
b_1(X) &= \frac{1}{2} \\
b_1(Y) &= \frac{1}{2} \\
b_1(Z) &= 0 \\
b_2(X) &= 0 \\
b_2(Y) &= \frac{1}{2} \\
b_2(Z) &= \frac{1}{2} \\
b_3(X) &= \frac{1}{2} \\
b_3(Y) &= 0 \\
b_3(Z) &= \frac{1}{2}
\end{align*} \]
\[ N = 3 \]
\[ M = 3 \]
\[ \pi_1 = \frac{1}{2} \quad \pi_2 = \frac{1}{2} \quad \pi_3 = 0 \]
\[ a_{11} = 0 \quad a_{12} = \frac{1}{3} \quad a_{13} = \frac{2}{3} \]
\[ a_{12} = \frac{1}{3} \quad a_{22} = 0 \quad a_{13} = \frac{2}{3} \]
\[ a_{13} = \frac{1}{3} \quad a_{32} = \frac{1}{3} \quad a_{13} = \frac{1}{3} \]
\[ b_1 (X) = \frac{1}{2} \quad b_1 (Y) = \frac{1}{2} \quad b_1 (Z) = 0 \]
\[ b_2 (X) = 0 \quad b_2 (Y) = \frac{1}{2} \quad b_2 (Z) = \frac{1}{2} \]
\[ b_3 (X) = \frac{1}{2} \quad b_3 (Y) = 0 \quad b_3 (Z) = \frac{1}{2} \]

Start randomly in state 1 or 2
Choose one of the output symbols in each state at random.
Let’s generate a sequence of observations:

<table>
<thead>
<tr>
<th>State</th>
<th>Output</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>(q_0)</td>
<td>(S_1)</td>
<td>(O_0) = (X)</td>
</tr>
<tr>
<td>(q_1)</td>
<td>(S_3)</td>
<td>(O_1) = (X)</td>
</tr>
<tr>
<td>(q_2)</td>
<td>___</td>
<td>(O_2) = ___</td>
</tr>
</tbody>
</table>
N = 3
M = 3
\( \pi_1 = \frac{1}{2} \)
\( \pi_2 = \frac{1}{2} \)
\( \pi_3 = 0 \)

\( a_{11} = 0 \)
\( a_{12} = \frac{1}{3} \)
\( a_{13} = \frac{1}{3} \)

\( a_{12} = \frac{1}{3} \)
\( a_{22} = 0 \)
\( a_{32} = \frac{1}{3} \)

\( a_{13} = \frac{2}{3} \)
\( a_{13} = \frac{2}{3} \)
\( a_{13} = 1/3 \)

\( b_1 (X) = \frac{1}{2} \)
\( b_1 (Y) = \frac{1}{2} \)
\( b_1 (Z) = 0 \)

\( b_2 (X) = 0 \)
\( b_2 (Y) = \frac{1}{2} \)
\( b_2 (Z) = \frac{1}{2} \)

\( b_3 (X) = \frac{1}{2} \)
\( b_3 (Y) = 0 \)
\( b_3 (Z) = \frac{1}{2} \)

Start randomly in state 1 or 2
Choose one of the output symbols in each state at random.
Let's generate a sequence of observations:

<table>
<thead>
<tr>
<th>( q_0 = )</th>
<th>( S_1 )</th>
<th>( O_0 = )</th>
<th>( X )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( q_1 = )</td>
<td>( S_3 )</td>
<td>( O_1 = )</td>
<td>( X )</td>
</tr>
<tr>
<td>( q_2 = )</td>
<td>( S_3 )</td>
<td>( O_2 = )</td>
<td>( _ )</td>
</tr>
</tbody>
</table>
Start randomly in state 1 or 2
Choose one of the output symbols in each state at random.
Let’s generate a sequence of observations:

$$N = 3$$
$$M = 3$$
$$\pi_1 = \frac{1}{2}$$
$$\pi_2 = \frac{1}{2}$$
$$\pi_3 = 0$$

$$a_{11} = 0$$
$$a_{12} = \frac{1}{3}$$
$$a_{13} = \frac{2}{3}$$

$$a_{21} = \frac{1}{3}$$
$$a_{22} = 0$$
$$a_{23} = \frac{2}{3}$$

$$a_{31} = \frac{1}{3}$$
$$a_{32} = \frac{1}{3}$$
$$a_{33} = \frac{1}{3}$$

$$b_1(X) = \frac{1}{2}$$
$$b_1(Y) = \frac{1}{2}$$
$$b_1(Z) = 0$$

$$b_2(X) = 0$$
$$b_2(Y) = \frac{1}{2}$$
$$b_2(Z) = \frac{1}{2}$$

$$b_3(X) = \frac{1}{2}$$
$$b_3(Y) = 0$$
$$b_3(Z) = \frac{1}{2}$$

<table>
<thead>
<tr>
<th>$q_0$</th>
<th>$S_1$</th>
<th>$O_0$</th>
<th>$X$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_1$</td>
<td>$S_3$</td>
<td>$O_1$</td>
<td>$X$</td>
</tr>
<tr>
<td>$q_2$</td>
<td>$S_3$</td>
<td>$O_2$</td>
<td>$Z$</td>
</tr>
</tbody>
</table>
State Estimation

Start randomly in state 1 or 2
Choose one of the output symbols in each state at random.
Let’s generate a sequence of observations:

<table>
<thead>
<tr>
<th>( q_0 )</th>
<th>?</th>
<th>( O_0 )</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>( q_1 )</td>
<td>?</td>
<td>( O_1 )</td>
<td>X</td>
</tr>
<tr>
<td>( q_2 )</td>
<td>?</td>
<td>( O_2 )</td>
<td>Z</td>
</tr>
</tbody>
</table>

N = 3
M = 3
\( \pi_1 = \frac{1}{2} \)
\( \pi_2 = \frac{1}{2} \)
\( \pi_3 = 0 \)

\( a_{11} = 0 \)
\( a_{12} = \frac{1}{3} \)
\( a_{13} = \frac{1}{3} \)

\( a_{12} = \frac{1}{3} \)
\( a_{22} = 0 \)
\( a_{32} = \frac{1}{3} \)

\( a_{13} = \frac{1}{3} \)
\( a_{23} = \frac{2}{3} \)
\( a_{33} = \frac{1}{3} \)

\( b_1 (X) = \frac{1}{2} \)
\( b_1 (Y) = \frac{1}{2} \)
\( b_1 (Z) = 0 \)

\( b_2 (X) = 0 \)
\( b_2 (Y) = \frac{1}{2} \)
\( b_2 (Z) = \frac{1}{2} \)

\( b_3 (X) = \frac{1}{2} \)
\( b_3 (Y) = 0 \)
\( b_3 (Z) = \frac{1}{2} \)
Prob. of a series of observations

What is \( P(O) = P(O_1 O_2 O_3) \)
\( = P(O_1 = X \land O_2 = X \land O_3 = Z)? \)

Slow way: \( P(O) = \sum_{Q \in \text{Paths of length } 3} P(O \land Q) \)
\( = \sum_{Q \in \text{Paths of length } 3} P(O \mid Q)P(Q) \)

How do we compute \( P(Q) \) for an arbitrary path \( Q? \)

How do we compute \( P(O \mid Q) \) for an arbitrary path \( Q? \)

\[
P(Q) = P(q_1, q_2, q_3)
= P(q_1) P(q_2, q_3 \mid q_1) \text{ (chain rule)}
= P(q_1) P(q_2 \mid q_1) P(q_3 \mid q_2, q_1) \text{ (chain)}
= P(q_1) P(q_2 \mid q_1) P(q_3 \mid q_2)
\]
Example in the case \( Q = S_1 S_3 S_3 \):
\( = \frac{1}{2} \times \frac{2}{3} \times \frac{1}{3} = \frac{1}{9} \)
What is $P(O) = P(O_1 \, O_2 \, O_3)$  

$= P(O_1 = X \, \wedge \, O_2 = X \, \wedge \, O_3 = Z)$?

Slow way: $P(O) = \sum_{Q \in \text{Paths of length } 3} P(O \wedge Q)$  

$= \sum_{Q \in \text{Paths of length } 3} P(O \mid Q)P(Q)$

How do we compute $P(Q)$ for an arbitrary path $Q$?

How do we compute $P(O \mid Q)$ for an arbitrary path $Q$?

Example in the case $Q = S_1 S_3 S_3$:  

$= P(O_1 \mid q_1) \, P(O_2 \mid q_2) \, P(O_3 \mid q_3)$ (why?)  

$= P(X \mid S_1) \, P(X \mid S_3) \, P(Z \mid S_3) = 1/2 \times 1/2 \times 1/2 = 1/8$
Prob. of a series of observations

What is \( P(O) = P(O_1 O_2 O_3) \)
\[ = P(O_1 = X \land O_2 = X \land O_3 = Z)? \]

Slow way: \( P(O) = \sum_{Q \in \text{Paths of length } 3} P(O \land Q) \)
\[ = \sum_{Q \in \text{Paths of length } 3} P(O|Q)P(Q) \]

How do we compute \( P(Q) \) for an arbitrary path \( Q \)?

How do we compute \( P(O|Q) \) for an arbitrary path \( Q \)?

\[
\text{P(O) would need 27 P(Q) computations and 27 P(O|Q) computations}
\]

A sequence of 20 observations would need \( 3^{20} = 3.5 \text{ billion computations} \)

So let’s be smarter…
Forward-Backward Algorithm

- Given observations $O_1 \ O_2 \ ... \ O_T$
- Define
  
  $$\alpha_t(i) = P(O_1 \ O_2 \ ... \ O_t \wedge q_t = S_i \mid \lambda) \text{ where } 1 \leq t \leq T$$

- $\alpha_t(i) =$ Probability that, in a random trial,
  
  - We’d have seen the first $t$ observations
  - We’d have ended up in $S_i$ as the $t$’th state visited.
Forward-Backward Algorithm

\[ \alpha_t(i) = P(O_1 O_2 \ldots O_T \land q_t = S_i \mid \lambda) \]

\[ \alpha_1(i) = P(O_1 \land q_1 = S_i) \]
\[ = P(q_1 = S_i) P(O_1 \mid q_1 = S_i) \]
\[ = \pi_i b_i(O_1) \]

\[ \alpha_{t+1}(j) = P(O_1 O_2 \ldots O_t O_{t+1} \land q_{t+1} = S_j) \]
\[ = \sum_i a_{ij} b_j(O_{t+1}) \alpha_t(i) \]
Forward-Backward Algorithm

\[ \alpha_t(i) = P(O_1 \ O_2 \ \ldots \ O_T \ \land \ q_t = S_i \mid \lambda) \]

\[ \alpha_1(i) = P(O_1 \land q_1 = S_i) \]
\[ = P(q_1 = S_i)P(O_1 \mid q_1 = S_i) \]
\[ = \pi_i b_i(O_1) \]

\[ \alpha_{t+1}(j) = P(O_1 O_2 \ldots O_t \ O_{t+1} \land q_{t+1} = S_j) \]
\[ = \sum_{i=1}^{N} P(O_1 O_2 \ldots O_t \land q_t = S_i \land O_{t+1} \land q_{t+1} = S_j) \]
\[ = \sum_{i=1}^{N} P(O_{t+1}, q_{t+1} = S_j \mid O_1 O_2 \ldots O_t \land q_t = S_i)P(O_1 O_2 \ldots O_t \land q_t = S_i) \]
\[ = \sum_i P(O_{t+1}, q_{t+1} = S_j \mid q_t = S_i)\alpha_t(i) \]
\[ = \sum_i P(q_{t+1} = S_j \mid q_t = S_i)P(O_{t+1} \mid q_{t+1} = S_j)\alpha_t(i) \]
\[ = \sum_i a_{ij} b_j(O_{t+1})\alpha_t(i) \]
Example Revisited

\[ \alpha_t(i) = P(O_1O_2..O_t \wedge q_t = S_i | \lambda) \]

\[ \alpha_1(i) = b_i(O_1)\pi_i \]

\[ \alpha_{t+1}(j) = \sum_i a_{ij}b_j(O_{t+1})\alpha_t(i) \]

WE SAW  \( O_1 O_2 O_3 = X X Z \)

\[ \begin{align*} 
\alpha_1(1) &= \frac{1}{4} \\
\alpha_1(2) &= 0 \\
\alpha_1(3) &= 0 \\
\alpha_2(1) &= 0 \\
\alpha_2(2) &= 0 \\
\alpha_2(3) &= \frac{1}{12} \\
\alpha_3(1) &= 0 \\
\alpha_3(2) &= \frac{1}{72} \\
\alpha_3(3) &= \frac{1}{72} 
\end{align*} \]
We can cheaply compute
\[ \alpha_t(i) = P(O_1 O_2 \ldots O_t \wedge q_t = S_i) \]

(How) can we cheaply compute
\[ P(O_1 O_2 \ldots O_t) \] ?

\[ \sum_{i=1}^{N} \alpha_t(i) \]

(How) can we cheaply compute
\[ P(q_t = S_i \mid O_1 O_2 \ldots O_t) \]

\[ \frac{\alpha_t(i)}{\sum_{j=1}^{N} \alpha_t(j)} \]
Example

\[ \pi = [1 \ 0 \ 0]^T \]

\[
\begin{bmatrix}
.5 \\
.4 \\
.6 \\
.1 \\
.2 \\
.5 \\
.3 \\
.0 \\
.3 \\
.7
\end{bmatrix}
\]

\[ P(\text{RRGB} | \lambda) \]
Backward Algorithm

- **Backward probability**
  \[ \beta_t(i) = P(x_{t+1} x_{t+2} \ldots x_T \mid q_t = S_i, \lambda) \]

- **Initialization**
  \[ \beta_T(i) = 1 \quad 1 \leq i \leq N \]

- **Induction**
  \[ \beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(x_{t+1}) \beta_{t+1}(j) \quad 1 \leq i \leq N, \quad t = T - 1, T - 2, \ldots, 1 \]
**Problem 2: Most Probable Path Decoding**

- **Goal:** Find the ‘optimal’ state sequence associated with the given observation.
- **What is the optimality criteria?**

  One possible criterion: choose the states which are individually most likely.
  This maximizes the expected number of correct individual states.

- **Problem:** When the HMM has state transitions $a_{ij} = 0$ for some $i$ and $j$, the optimal state sequence may not even be a valid state sequence.

  Since the solution determines the most likely state at every instant, without regard to the probability of occurrence of sequence of states.

- Formal technique for finding the single best state sequence based on dynamic programming methods is called the **Viterbi algorithm**
Viterbi Algorithm

- **Goal:** To find the single best state sequence \( Q = \{q_1, q_2, \ldots, q_r\} \)
- **Observation:** \( O = \{O_1, O_2, \ldots, O_t\} \)

\[
\delta_t(i) = \max_{q_1, q_2, \ldots, q_{t-1}} P[q_1, q_2, \ldots, q_t = i, O_1, O_2, \ldots, O_t|\lambda]
\]

i.e., \( \delta_t(i) \) is the best score (highest probability) along a single path, at time \( t \), which accounts for the first \( t \) observations and ends in state \( S_j \). By induction we have

\[
\delta_{t+1}(j) = \max_i \delta_t(i)a_{ij} \cdot b_j(O_{t+1}). \quad (31)
\]

To actually retrieve the state sequence, we need to keep track of the argument which maximized (31), for each \( t \) and \( j \). We do this via the array \( \psi_t(j) \). The complete procedure

\[
\psi_{t+1}(j) = \arg \max_{1 \leq i \leq N} \delta_t(i)a_{ij}
\]
Viterbi Algorithm Contd.

1) Initialization:
\[ \delta_1(i) = \pi_i b_i(O_1), \quad 1 \leq i \leq N \]
\[ \psi_1(i) = 0. \]

2) Recursion:
\[ \delta_t(j) = \max_{1 \leq i \leq N} [\delta_{t-1}(i)a_{ij}]b_i(O_t), \quad \begin{array}{l} 2 \leq t \leq T \\ 1 \leq j \leq N \end{array} \]
\[ \psi_t(j) = \arg\max_{1 \leq i \leq N} [\delta_{t-1}(i)a_{ij}], \quad \begin{array}{l} 2 \leq t \leq T \\ 1 \leq j \leq N \end{array} \]

3) Termination:
\[ P^* = \max_{1 \leq i \leq N} [\delta_T(i)] \]
\[ q_T^* = \arg\max_{1 \leq i \leq N} [\delta_T(i)]. \]
Example
Problem 3: Estimate Model Parameters

- Goal: Adjust the model parameters \((A, B, \pi)\) to maximize the probability of the observation sequence given the model.
- Most difficult of the 3 problems.
- No way to analytically solve for the model.

- Solution: Choose \(\lambda = (A, B, \pi)\) such that \(P(O|\lambda)\) is locally maximized.
- Iterative procedure called Baum-Welch Method (alternately EM method) or using gradient descent techniques.

First define \(\xi_t(i,j)\), the probability of being in state \(S_i\) at time \(t\), and state \(S_j\) at time \(t+1\), given the model and the observation sequence, i.e.

\[
\xi_t(i,j) = P(q_t = S_i, q_{t+1} = S_j | O, \lambda).
\] (36)
Baum-Welch Method

The forward variable $\alpha_t(i)$ defined as

$$\alpha_t(i) = P(O_1, O_2, \ldots, O_t, q_t = S_i | \lambda)$$

In a similar manner, we can consider a backward variable $\beta_t(i)$ defined as

$$\beta_t(i) = P(O_{t+1}, O_{t+2}, \ldots, O_T | q_t = S_i, \lambda) \quad (23)$$

$$\xi_t(i, j) = \frac{\alpha_t(i) \ a_{ij} \ b_j(O_{t+1}) \ \beta_{t+1}(j)}{P(O | \lambda)}$$

$$= \frac{\alpha_t(i) \ a_{ij} \ b_j(O_{t+1}) \ \beta_{t+1}(j)}{\sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_t(i) \ a_{ij} \ b_j(O_{t+1}) \ \beta_{t+1}(j)} \quad (37)$$

where the numerator term is just $P(q_t = S_i, q_{t+1} = S_j, O | \lambda)$
We have previously defined $\gamma_t(i)$ as the probability of being in state $S_i$ at time $t$, given the observation sequence and the model; hence we can relate $\gamma_t(i)$ to $\xi_t(i, j)$ by summing over $j$, giving

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i, j).$$  \hspace{1cm} (38)

$$\sum_{t=1}^{T-1} \gamma_t(i) = \text{expected number of transitions from } S_i$$

$$\sum_{t=1}^{T-1} \xi_t(i, j) = \text{expected number of transitions from } S_i \text{ to } S_j.$$
Re-estimation Formulas

\( \bar{\pi}_i = \text{expected frequency (number of times) in state } S_i \text{ at time } (t = 1) = \gamma_1(i) \)

\( \bar{a}_{ij} = \frac{\text{expected number of transitions from state } S_i \text{ to state } S_j}{\text{expected number of transitions from state } S_i} \)

\[
\begin{align*}
\bar{a}_{ij} &= \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \\
&= \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}
\end{align*}
\]

\( \bar{b}_{j(k)} = \frac{\text{expected number of times in state } j \text{ and observing symbol } v_k}{\text{expected number of times in state } j} \)

\[
\begin{align*}
\bar{b}_{j(k)} &= \frac{\sum_{t=1}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)} \\
&= \frac{\sum_{t=1}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)}
\end{align*}
\]

reestimated model as \( \bar{\lambda} = \bar{A}, \bar{B}, \bar{\pi} \), \( P(O|\bar{\lambda}) > P(O|\lambda) \).
Re-estimation Notes

The reestimation formulas of (40a)–(40c) can be derived directly by maximizing (using standard constrained optimization techniques) Baum’s auxiliary function

$$Q(\lambda, \tilde{\lambda}) = \sum_Q P(Q|O, \lambda) \log [P(O, Q|\tilde{\lambda})]$$  

(41)

over $\tilde{\lambda}$. It has been proven by Baum and his colleagues [6], [3] that maximization of $Q(\lambda, \tilde{\lambda})$ leads to increased likelihood, i.e.

$$\max_{\tilde{\lambda}} [Q(\lambda, \tilde{\lambda})] = P(O|\tilde{\lambda}) \geq P(O|\lambda).$$  

(42)

- Alternative: Maximize $P = P(O|\hat{\lambda})$ using gradient descent techniques = EM Algorithm
Types of HMM

- Fully connected HMM
  - Every state of the model can be reached from every other state
  - All $a_{ij} > 0$

- Left-right model or Bakis Model
  - States proceed from left to right
  - Useful to model speech signals
  - Additional Constraints: Here $= 2$
    \[
    a_{ij} = 0, \quad j < i
    \]
    \[
    \pi_i = \begin{cases} 
    0, & i \neq 1 \\
    1, & i = 1
    \end{cases}
    \]

$A = \begin{bmatrix} 
    a_{11} & a_{12} & a_{13} & a_{14} \\
    a_{21} & a_{22} & a_{23} & a_{24} \\
    a_{31} & a_{32} & a_{33} & a_{34} \\
    a_{41} & a_{42} & a_{43} & a_{44} 
\end{bmatrix}$
Applications – Human Action Recognition

- Used in automatic monitoring systems
- Input: Time Sequential images $I = \{I_1, I_2, \ldots, I_T\}$
- Each Frame $I_i \rightarrow$ feature vector $f_i \in \mathbb{R}^n$
- Feature space $f_i \in \mathbb{R}^n$ divided into clusters using vector quantization
- Each codeword (cluster centers) assigned a symbol
- No. of clusters = no. of HMM output symbols
- Each feature vector $f_i \rightarrow$ nearest codeword $\rightarrow$ symbol
- Symbol Sequences are used for both learning and recognition

Transformed to Sequences $O$

Recognizing human action in time-sequential images using hidden Markov model, CVR 1992
Learning & Recognition

- One HMM is created for each category.
- Suppose there are C types of actions.
- Goal: Choose from the C HMM’s, the model that best matches the observations.
- Testing: A sequence of an unknown category is given.

\[ \lambda_i = \{A_i, B_i, \pi_i\}, i = 1 \ldots C. \]

If an unknown category is given, we calculate \( Pr(\lambda_i|O) \) for each HMM \( \lambda_i \) and select \( \lambda_{c^*} \), where

\[ c^* = \arg \max_i (Pr(\lambda_i|O)) \]  \hspace{1cm} (1)

- Forward-backward algorithm

**Learning:**

- Each HMM should be trained so that it is most likely to generate the symbol patterns for its category.
- HMM Training -> optimizing the model parameters \( (A, B, \pi) \) to maximize the probability of observation sequence.
- Baum-Welch algorithm used for training.
Goal: To recognize 6 tennis strokes:

organized were six tennis strokes: 'forehand stroke', 'backhand stroke', 'forehand volley', 'backhand volley', 'smash', and 'service'.

Figure 4: Example of tennis action (forehand volley)

Figure 5: Human area extraction
a) original, b) background, c) extracted
a) backhand volley

b) backhand stroke

c) forehand volley

d) forehand stroke

e) smash

f) service
Experiment

- Training: After VQ, 12 images of each category as codewords
- NO. of HMM symbols = 72
- No. of states = 36

The number of HMM symbols was 72; symbols 0 to 11 represent 'backhand stroke', 12 to 23 the 'backhand volley', 24 to 35 the 'service', 36 to 47 the 'smash', 48 to 59 the 'forehand stroke', and 60 to 71 the 'forehand volley'.
Experiment

- Recognition % decreases as the no. of training pattern reduces
- HMM parameters critically depend on the selection of training patterns
- Training pattern should cover the maximum test pattern scatter
- Performance decreases if train and test subjects are different, since different persons have some uniqueness in the actions