Hidden Markov Models

Andrew W. Moore
Professor
School of Computer Science
Carnegie Mellon University
www.cs.cmu.edu/~awm
awm@cs.cmu.edu
412-268-7599
A Markov System

Has $N$ states, called $s_1, s_2 \ldots s_N$

There are discrete timesteps, $t=0, t=1, \ldots$

$N = 3$

$t=0$
A Markov System

Has $N$ states, called $s_1$, $s_2$, .. $s_N$

There are discrete timesteps, $t=0$, $t=1$, ...

On the $t$'th timestep the system is in exactly one of the available states. Call it $q_t$

Note: $q_t \in \{s_1, s_2, .. s_N\}$
A Markov System

Has \( N \) states, called \( s_1, s_2 \ldots s_N \)

There are discrete timesteps, \( t=0, t=1, \ldots \)

On the \( t \)'th timestep the system is in exactly one of the available states. Call it \( q_t \)

Note: \( q_t \in \{s_1, s_2 \ldots s_N \} \)

Between each timestep, the next state is chosen randomly.

\( N = 3 \)

\( t=1 \)

\( q_t=q_1=s_2 \)
A Markov System

Has \( N \) states, called \( s_1, s_2 \ldots s_N \)

There are discrete timesteps, \( t=0, t=1, \ldots \)

On the \( t \)'th timestep the system is in exactly one of the available states. Call it \( q_t \)

Note: \( q_t \in \{s_1, s_2 \ldots s_N \} \)

Between each timestep, the next state is chosen randomly.

The current state determines the probability distribution for the next state.
A Markov System

Has \( N \) states, called \( s_1, s_2, \ldots, s_N \)

There are discrete timesteps, \( t=0, t=1, \ldots \)

On the \( t \)'th timestep the system is in exactly one of the available states. Call it \( q_t \)

Note: \( q_t \in \{s_1, s_2, \ldots, s_N\} \)

Between each timestep, the next state is chosen randomly.

The current state determines the probability distribution for the next state.

Often notated with arcs between states
Markov Property

$q_{t+1}$ is conditionally independent of $\{q_{t-1}, q_{t-2}, \ldots q_1, q_0\}$ given $q_t$. In other words:

$$P(q_{t+1} = s_j | q_t = s_i) =$$

$$P(q_{t+1} = s_j | q_t = s_i, \text{any earlier history})$$

Question: what would be the best Bayes Net structure to represent the Joint Distribution of $(q_0, q_1, \ldots q_3, q_4)$?

$N = 3$
$t=1$
$q_t = q_1 = s_2$
Markov Property

$q_{t+1}$ is conditionally independent of \{ $q_{t-1}$, $q_{t-2}$, \ldots $q_1$, $q_0$ \} given $q_t$.

In other words:

\[
P(q_{t+1} = s_j | q_t = s_i) = \]

\[
P(q_{t+1} = s_j | q_t = s_i, \text{any earlier history})
\]

Question: what would be the best Bayes Net structure to represent the Joint Distribution of \( q_0, q_1, q_2, q_3, q_4 \)?
Markov Property

$q_{t+1}$ is conditionally independent of $\{q_{t-1}, q_{t-2}, \ldots, q_1, q_0\}$ given $q_t$.

In other words:

$$P(q_{t+1} = s_j | q_t = s_i) = P(q_{t+1} = s_j | q_t = s_i, \text{any earlier history})$$

**Question:** what would be the best Bayes Net structure to represent the Joint Distribution of $(q_0, q_1, q_2, q_3, q_4)$?

**Answer:**

$$P(q_{t+1} = s_1 | q_t = s_2) = 1/2$$

$$P(q_{t+1} = s_2 | q_t = s_2) = 1/2$$

$$P(q_{t+1} = s_3 | q_t = s_2) = 0$$

$$P(q_{t+1} = s_1 | q_t = s_1) = 0$$

$$P(q_{t+1} = s_2 | q_t = s_1) = 0$$

$$P(q_{t+1} = s_3 | q_t = s_1) = 1$$

$$P(q_{t+1} = s_1 | q_t = s_3) = 1/3$$

$$P(q_{t+1} = s_2 | q_t = s_3) = 2/3$$

$$P(q_{t+1} = s_3 | q_t = s_3) = 0$$

$$P(q_{t+1} = s_1 | q_t = s_i) = a_{i1}$$

$$P(q_{t+1} = s_2 | q_t = s_i) = a_{i2}$$

$$\vdots$$

$$P(q_{t+1} = s_N | q_t = s_i) = a_{iN}$$

Each of these probability tables is identical.

Notation:

$$a_{ij} = P(q_{t+1} = s_j | q_t = s_i)$$
A Blind Robot

A human and a robot wander around randomly on a grid…

STATE $q =$ Location of Robot, Location of Human

Note: $N$ (num. states) = 18 * 18 = 324
Dynamics of System

Each timestep the human moves randomly to an adjacent cell. And Robot also moves randomly to an adjacent cell.

Typical Questions:

• “What’s the expected time until the human is crushed like a bug?”
• “What’s the probability that the robot will hit the left wall before it hits the human?”
• “What’s the probability Robot crushes human on next time step?”
Example Question

“It’s currently time t, and human remains uncrushed. What’s the probability of crushing occurring at time $t + 1$?”

If robot is blind:

We can compute this in advance.

If robot is omnipotent:

(I.E. If robot knows state at time $t$), can compute directly.

If robot has some sensors, but incomplete state information …

Hidden Markov Models are applicable!
What is \( P(q_t = s) \)? slow, stupid answer

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

Given we know the start state \( q_1 \) (i.e. \( P(q_1) = 1 \))

\[
P(q_1 \ q_2 \ldots \ q_t) = P(q_1 \ q_2 \ldots \ q_{t-1}) \ P(q_t|q_1 \ q_2 \ldots \ q_{t-1})
= P(q_1 \ q_2 \ldots \ q_{t-1}) \ P(q_t|q_{t-1})
= P(q_2|q_1)P(q_3|q_2) \ldots P(q_t|q_{t-1})
\]

WHY?

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 \( t \)
What is $P(q_t = s)$? Clever answer

- 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

  $\forall i \quad p_0(i) =$

  $\forall j \quad p_{t+1}(j) = P(q_{t+1} = s_j) =$
What is $P(q_t = s)$? Clever answer

- 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

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

  $\forall j \quad p_{t+1}(j) = P(q_{t+1} = s_j) =$
What is $P(q_t = s)$? Clever answer

- 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

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

  $\forall j \quad p_{t+1}(j) = P(q_{t+1} = s_j) = \sum_{i=1}^{N} P(q_{t+1} = s_j \land q_t = s_i) =$
What is $P(q_t = s)$? Clever answer

- 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

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

  $\forall j \ p_{t+1}(j) = P(q_{t+1} = s_j) =$

  $\sum_{i=1}^{N} P(q_{t+1} = s_j \land q_t = s_i) =$

  $\sum_{i=1}^{N} P(q_{t+1} = s_j \mid q_t = s_i) P(q_t = s_i) = \sum_{i=1}^{N} a_{ij} p_t(i)$

  Remember, $a_{ij} = P(q_{t+1} = s_j \mid q_t = s_i)$
**What is $P(q_t = s)$?** Clever answer

- 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>$t_{\text{final}}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
What is $P(q_t = s)$? Clever answer

- 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

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

  $$\forall j \quad p_{t+1}(j) = P(q_{t+1} = s_j) =$$

  $$\sum_{i=1}^{N} P(q_{t+1} = s_j \land q_t = s_i) =$$

  $$\sum_{i=1}^{N} P(q_{t+1} = s_j \mid q_t = s_i)P(q_t = s_i) = \sum_{i=1}^{N} a_{ij} p_t(i)$$

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

- The stupid way was $O(N^t)$

- This was a simple example

- It was meant to warm you up to this trick, called **Dynamic Programming**, because HMMs do many tricks like this.
Hidden State

“It’s currently time \( t \), and human remains uncrushed. What’s the probability of crushing occurring at time \( t + 1 \)?”

If robot is blind:

We can compute this in advance.

If robot is omnipotent:

(I.E. If robot knows state at time \( t \)),
can compute directly.

If robot has some sensors, but incomplete state information …

Hidden Markov Models are applicable!

We’ll do this first

Too Easy. We won’t do this

Main Body of Lecture
Hidden State

- The previous example tried to estimate $P(q_t = s_i)$ unconditionally (using no observed evidence).
- Suppose we can observe something that’s affected by the true state.
- Example: **Proximity sensors.** (tell us the contents of the 8 adjacent squares)

```
True state $q_t$

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>R₀</td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

```
What the robot sees: Observation $O_t$

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>W</td>
<td>W</td>
<td>W</td>
</tr>
<tr>
<td></td>
<td>®</td>
<td></td>
</tr>
<tr>
<td>H</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

W denotes “WALL”
Noisy Hidden State

- Example: **Noisy Proximity sensors.** (unreliably tell us the contents of the 8 adjacent squares)

![Diagram](image)

- True state $q_t$
- Uncorrupted Observation
- What the robot sees: Observation $O_t$
Noisy Hidden State

- Example: Noisy Proximity sensors. (unreliably tell us the contents of the 8 adjacent squares)

True state $q_t$

$O_t$ is noisily determined depending on the current state.

Assume that $O_t$ is conditionally independent of \{ $q_{t-1}, q_{t-2}, \ldots, q_1, q_0, O_{t-1}, O_{t-2}, \ldots, O_1, O_0$ \} given $q_t$.

In other words:

$P(O_t = X | q_t = s_i) =$

$P(O_t = X | q_t = s_i, \text{any earlier history})$
Noisy Hidden State

- Example: **Noisy Proximity sensors.** (unreliably tell us the contents of the 8 adjacent squares)

True state $q_t$

$O_t$ is noisily determined depending on the current state.

Assume that $O_t$ is conditionally independent of $\{q_{t-1}, q_{t-2}, ..., q_1, q_0, O_{t-1}, O_{t-2}, ..., O_1, O_0\}$ given $q_t$.

In other words:

$$P(O_t = X | q_t = s_i) =$$

$$P(O_t = X | q_t = s_i, \text{any earlier history})$$

Question: what’d be the best Bayes Net structure to represent the Joint Distribution of $(q_0, q_1, q_2, q_3, q_4, O_0, O_1, O_2, O_3, O_4)$?
Noisy Hidden State

Example: Noisy Proximity sensors. (unreliably tell us the contents of the 8 adjacent squares)

• True state \( q_t \)
• Uncorrupted Observation \( O_t \)

What the robot sees:
Observation \( O_t \) is noisily determined depending on the current state.
Assume that \( O_t \) is conditionally independent of \( \{q_{t-1}, q_{t-2}, \ldots, q_0, O_{t-1}, O_{t-2}, \ldots, O_0\} \) given \( q_t \).
In other words:

\[
P(O_t = X | q_t = s) = P(O_t = X | q_t = s, \text{any earlier history})
\]

Question: what’d be the best Bayes Net structure to represent the Joint Distribution of \( (q_0, q_1, q_2, q_3, q_4, O_0, O_1, O_2, O_3, O_4) \)?

Answer:
**Hidden State:**

- **Example:** Noisy Proximity sensors. (unreliably tell us the contents of the 8 adjacent squares)

- **Denotes** "WALL"

True state

Uncorrupted Observation

What the robot sees: Observation

**Assume that** $O_t$ **is conditionally independent of** $\{q_{t-1}, q_{t-2}, \ldots, q_1, q_0, O_{t-1}, O_{t-2}, \ldots, O_1, O_0\}$ **given** $q_t$.

In other words:

$$P(O_t = X_i | q_t = s_i) = P(O_t = X_i | q_t = s_i, \text{any earlier history})$$

**Question:** what'd be the best Bayes Net structure to represent the Joint Distribution of $(q_0, q_1, q_2, q_3, q_4, O_0, O_1, O_2, O_3, O_4)$?

**Answer:**

**Notation:**

$$b_i(k) = P(O_t = k | q_t = s_i)$$

| $i$ | $P(O_t=1|q_t=s_i)$ | $P(O_t=2|q_t=s_i)$ | $\ldots$ | $P(O_t=k|q_t=s_i)$ | $\ldots$ | $P(O_t=M|q_t=s_i)$ |
|-----|-------------------|-------------------|------|-------------------|------|-------------------|
| 1   | $b_1(1)$          | $b_1(2)$          | $\ldots$ | $b_1(k)$       | $\ldots$ | $b_1(M)$        |
| 2   | $b_2(1)$          | $b_2(2)$          | $\ldots$ | $b_2(k)$       | $\ldots$ | $b_2(M)$        |
| 3   | $b_3(1)$          | $b_3(2)$          | $\ldots$ | $b_3(k)$       | $\ldots$ | $b_3(M)$        |
| $\vdots$ | $\vdots$          | $\vdots$          | $\ddots$ | $\vdots$       | $\ddots$ | $\vdots$        |
| $i$ | $b_i(1)$          | $b_i(2)$          | $\ldots$ | $b_i(k)$       | $\ldots$ | $b_i(M)$        |
| $\vdots$ | $\vdots$          | $\vdots$          | $\ddots$ | $\vdots$       | $\ddots$ | $\vdots$        |
| $N$ | $b_N(1)$          | $b_N(2)$          | $\ldots$ | $b_N(k)$       | $\ldots$ | $b_N(M)$        |
Hidden Markov Models

Our robot with noisy sensors is a good example of an HMM

• **Question 1: State Estimation**
  What is $P(q_T = S_i | O_1 O_2 \ldots O_T)$
  It will turn out that a new cute D.P. trick will get this for us.

• **Question 2: Most Probable Path**
  Given $O_1 O_2 \ldots O_T$, what is the most probable path that I took?
  And what is that probability?
  Yet another famous D.P. trick, the VITERBI algorithm, gets this.

• **Question 3: Learning HMMs:**
  Given $O_1 O_2 \ldots O_T$, what is the maximum likelihood HMM that could have produced this string of observations?
  Very very useful. Uses the E.M. Algorithm
Are H.M.M.s Useful?

You bet !!

• Robot planning + sensing when there’s uncertainty (e.g. Reid Simmons / Sebastian Thrun / Sven Koenig)

• Speech Recognition/Understanding
  Phones $\rightarrow$ Words, Signal $\rightarrow$ phones

• Human Genome Project
  Complicated stuff your lecturer knows nothing about.

• Consumer decision modeling

• Economics & Finance.

Plus at least 5 other things I haven’t thought of.
Some Famous HMM Tasks

Question 1: State Estimation
What is $P(q_T = S_i \mid O_1 O_2 \ldots O_t)$?
Some Famous HMM Tasks

Question 1: State Estimation
What is \( P(q_T=q_i | O_1O_2...O_t) \)?
Some Famous HMM Tasks

Question 1: State Estimation
What is $P(q_T = s | O_1 O_2 ... O_t)$?
Some Famous HMM Tasks

Question 1: State Estimation
What is $P(q_T=S_i \mid O_1O_2\ldots O_t)$?

Question 2: Most Probable Path
Given $O_1O_2\ldots O_T$, what is the most probable path that I took?
Some Famous HMM Tasks

Question 1: State Estimation
What is \( P(q_T=S_i \mid O_1O_2\ldots O_t) \)?

Question 2: Most Probable Path
Given \( O_1O_2\ldots O_T \), what is the most probable path that I took?
Some Famous HMM Tasks

Question 1: State Estimation
What is $P(q_T=S_i \mid O_1 O_2 \ldots O_t)$?

Question 2: Most Probable Path
Given $O_1 O_2 \ldots O_T$, what is the most probable path that I took?

Woke up at 8.35, Got on Bus at 9.46, Sat in lecture 10.05-11.22…
Some Famous HMM Tasks

<table>
<thead>
<tr>
<th>Question 1: State Estimation</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is $P(q_T=S_i \mid O_1O_2\ldots O_t)$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question 2: Most Probable Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>Given $O_1O_2\ldots O_T$, what is the most probable path that I took?</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Question 3: Learning HMMs:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Given $O_1O_2\ldots O_T$, what is the maximum likelihood HMM that could have produced this string of observations?</td>
</tr>
</tbody>
</table>
Some Famous HMM Tasks

Question 1: State Estimation
What is $P(q_T=S_i | O_1O_2...O_T)$?

Question 2: Most Probable Path
Given $O_1O_2...O_T$, what is the most probable path that I took?

Question 3: Learning HMMs:
Given $O_1O_2...O_T$, what is the maximum likelihood HMM that could have produced this string of observations?
Some Famous HMM Tasks

**Question 1: State Estimation**
What is $P(q_T=S_i | O_1O_2…O_T)$?

**Question 2: Most Probable Path**
Given $O_1O_2…O_T$, what is the most probable path that I took?

**Question 3: Learning HMMs**
Given $O_1O_2…O_T$, what is the maximum likelihood HMM that could have produced this string of observations?
Basic Operations in HMMs

For an observation sequence $O = O_1 \ldots O_T$, the three basic HMM operations are:

<table>
<thead>
<tr>
<th>Problem</th>
<th>Algorithm</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Evaluation:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Calculating $P(q_t = S_i</td>
<td>O_1 O_2 \ldots O_t)$</td>
<td>Forward-Backward</td>
</tr>
<tr>
<td><strong>Inference:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Computing $Q^* = \arg\max_Q P(Q</td>
<td>O)$</td>
<td>Viterbi Decoding</td>
</tr>
<tr>
<td><strong>Learning:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Computing $\lambda^* = \arg\max_\lambda P(O</td>
<td>\lambda)$</td>
<td>Baum-Welch (EM)</td>
</tr>
</tbody>
</table>

$T =$ # timesteps, $N =$ # states
HMM Notation
(from Rabiner’s Survey)

The states are labeled $S_1, S_2, \ldots, S_N$.

For a particular trial….

Let $T$ be the number of observations.
$T$ is also the number of states passed through.

$O = O_1, O_2, \ldots, O_T$ is the sequence of observations.

$Q = q_1, q_2, \ldots, q_T$ is the notation for a path of states.

$\lambda = \langle N, M, \{\pi_i\}, \{a_{ij}\}, \{b_i(j)\}\rangle$ is the specification of an HMM.


Available from

HMM Formal Definition

An HMM, $\lambda$, is a 5-tuple consisting of

- $N$ the number of states
- $M$ the number of possible observations
- $\{\pi_1, \pi_2, .. \pi_N\}$ The starting state probabilities
  \[ P(q_0 = S_i) = \pi_i \]
- $a_{11}$ $a_{22}$ ... $a_{1N}$
  $a_{21}$ $a_{22}$ ... $a_{2N}$
  ... ...
  $a_{N1}$ $a_{N2}$ ... $a_{NN}$

  The state transition probabilities
  \[ P(q_{t+1} = S_j | q_t = S_i) = a_{ij} \]

- $b_1(1)$ $b_1(2)$ ... $b_1(M)$
  $b_2(1)$ $b_2(2)$ ... $b_2(M)$
  ... ...
  $b_N(1)$ $b_N(2)$ ... $b_N(M)$

  The observation probabilities
  \[ P(O_t = k | q_t = S_i) = b_i(k) \]

This is new. In our previous example, start state was deterministic.
Here’s an HMM

- **N = 3**
- **M = 3**
- $\pi_1 = 1/2$
- $\pi_2 = 1/2$
- $\pi_3 = 0$
- $a_{11} = 0$
- $a_{12} = 1/3$
- $a_{13} = 1/3$
- $a_{12} = 1/3$
- $a_{22} = 0$
- $a_{13} = 2/3$
- $a_{23} = 1/3$
- $b_1 (X) = 1/2$
- $b_1 (Y) = 1/2$
- $b_1 (Z) = 0$
- $b_2 (X) = 0$
- $b_2 (Y) = 1/2$
- $b_2 (Z) = 1/2$
- $b_3 (X) = 1/2$
- $b_3 (Y) = 0$
- $b_3 (Z) = 1/2$

Start randomly in state 1 or 2

Choose one of the output symbols in each state at random.
Here's an HMM

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:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_0$</td>
<td></td>
<td>$O_0$</td>
</tr>
<tr>
<td>$q_1$</td>
<td></td>
<td>$O_1$</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}$

$a_{22} = 0$

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

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

$b_2 (X) = 0$

$b_3 (X) = \frac{1}{2}$  
b_3 (Y) = 0  
b_3 (Z) = \frac{1}{2}$
Here’s an HMM

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

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:

\[
\begin{array}{|c|c|c|}
\hline
q_0 & S_1 & O_0 \\
\hline
q_1 & & O_1 \\
\hline
q_2 & & O_2 \\
\hline
\end{array}
\]

50-50 choice between X and Y
Here’s an HMM

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:

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

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>Symbol</th>
<th>Symbol</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_0$</td>
<td>$S_1$</td>
<td>$O_0$</td>
<td>$X$</td>
</tr>
<tr>
<td>$q_1$</td>
<td>$S_3$</td>
<td>$O_1$</td>
<td>_</td>
</tr>
<tr>
<td>$q_2$</td>
<td>_</td>
<td>$O_2$</td>
<td>_</td>
</tr>
</tbody>
</table>

$N = 3$
$M = 3$

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

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

$b_1 (X) = \frac{1}{2}$
$b_1 (Y) = \frac{1}{2}$
$b_1 (Z) = 0$
$b_2 (X) = 0$
$b_2 (Y) = \frac{1}{2}$
$b_2 (Z) = \frac{1}{2}$
$b_3 (X) = \frac{1}{2}$
$b_3 (Y) = 0$
$b_3 (Z) = \frac{1}{2}$
Here’s an HMM

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:

Each of the three next states is equally likely

<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>___</td>
<td>O_2=</td>
<td>___</td>
</tr>
</tbody>
</table>

N = 3
M = 3
π_1 = ½
π_2 = ½
π_3 = 0

a_{11} = 0
a_{12} = ⅓
a_{13} = ⅓
a_{22} = 0
a_{32} = ⅓

b_1 (X) = ½
b_1 (Y) = ½
b_1 (Z) = 0
b_2 (X) = 0
b_2 (Y) = ½
b_2 (Z) = ½
b_3 (X) = ½
b_3 (Y) = 0
b_3 (Z) = ½
Here’s an HMM

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
π₁ = ½
π₂ = ½
π₃ = 0

a₁₁ = 0
a₁₂ = ⅓
a₁₃ = ⅓
a₁₃ = ⅓
a₂₁ = ½
a₂₂ = 0
a₂₃ = ⅓
a₃₁ = ⅓
a₃₂ = ⅓
a₃₃ = ⅓

b₁ (X) = ½
b₁ (Y) = ½
b₁ (Z) = 0
b₂ (X) = 0
b₂ (Y) = ½
b₂ (Z) = ½
b₃ (X) = ½
b₃ (Y) = 0
b₃ (Z) = ½
Here’s an HMM

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

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

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

N = 3
M = 3
π₁ = ½
π₂ = ½
π₃ = 0

a₁₁ = 0
a₁₂ = ⅓
a₁₃ = ⅓
a₂₂ = 0
a₂₃ = ⅓
a₃₂ = ⅓
a₃₃ = ⅓

b₁ (X) = ½
b₁ (Y) = ½
b₁ (Z) = 0
b₂ (X) = 0
b₂ (Y) = ½
b₂ (Z) = ½
b₃ (X) = ½
b₃ (Y) = 0
b₃ (Z) = ½

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:

This is what the observer has to work with…

| q₀ = ? | O₀ = X |
| q₁ = ? | O₁ = X |
| q₂ = ? | O₂ = Z |
Prob. of a series of observations

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

Slow, stupid 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 \( Q \)?
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, stupid 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 \( Q \)?

Example in the case \( Q = S_1 S_3 S_3 \):

\[
= 1/2 \times 2/3 \times 1/3 = 1/9
\]
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, stupid way:

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

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

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

\[
P(O|Q) = P(O_1 O_2 O_3 | q_1 q_2 q_3 )
= P(O_1 | q_1 ) P(O_2 | q_2 ) P(O_3 | q_3 ) \text{ (why?)}
\]

Example in the case \( Q = S_1 S_3 S_3 \):

\[
= P(X| S_1 ) P(X| S_3 ) P(Z| S_3 ) =
=1/2 \times 1/2 \times 1/2 = 1/8
\]
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, stupid way:

\[
P(O) = \sum_{Q} P(O \land Q)
\]

\[
= \sum_{Q} P(O \mid Q)P(Q)
\]

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

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

A sequence of 20 observations would need \( 3^{20} = 3.5 \) billion computations and 3.5 billion \( P(O \mid Q) \) computations.

So let’s be smarter…
The Prob. of a given series of observations, non-exponential-cost-style

Given observations $O_1, O_2, \ldots, O_T$

Define

$$\alpha_t(i) = \Pr(O_1, O_2, \ldots, O_t \land q_t = S_i \mid \lambda) \quad \text{where} \quad 1 \leq t \leq T$$

$\alpha_t(i)$ = 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.

In our example, what is $\alpha_2(3)$?
$\alpha_t(i)$: easy to define recursively

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

($\alpha_t(i)$ can be defined stupidly by considering all paths length “$t$”. How?)

$\alpha_1(i) = P(O_1 \land q_1 = S_i)$

$\quad = P(q_1 = S_i) P(O_1 \mid q_1 = S_i)$

$\quad = \text{what?}$

$\alpha_{t+1}(j) = P(O_1 O_2 \ldots O_t O_{t+1} \land q_{t+1} = S_j)$

$\quad = \text{what?}$
\( \alpha_t(i) \): easy to define recursively

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

(\( \alpha_t(i) \) can be defined stupidly by considering all paths length "t". How?)

\[
\begin{align*}
\alpha_1(i) &= P(O_1 \land q_1 = S_i) \\
&= P(q_1 = S_i) P(O_1 \mid q_1 = S_i) \\
&= \text{what?}
\end{align*}
\]

\[
\alpha_t(j) = P(O_1 O_2 \cdots O_t O_{t+1} \land q_{t+1} = S_j)
\]

\[
= \sum_{j=1}^{N} P(O_1 O_2 \cdots O_t \land q_t = S_i \land O_{t+1} \land q_{t+1} = S_j)
\]

\[
= \sum_{j=1}^{N} P(O_{t+1}, q_{t+1} = S_j \mid O_1 O_2 \cdots O_t \land q_t = S_i) P(O_1 O_2 \cdots O_t \land q_t = S_i)
\]

\[
= \sum_{j=1}^{N} P(O_{t+1}, q_{t+1} = S_j \mid q_t = S_i) \alpha_t(i)
\]

\[
= \sum_{j} P(q_{t+1} = S_j \mid q_t = S_i) P(O_{t+1} \mid q_{t+1} = S_j) \alpha_t(i)
\]

\[
= \sum_{i} a_{ij} b_j (O_{t+1}) \alpha_t(i)
\]
in our example

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

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

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

WE SAW \( O_1O_2O_3 = X \times Z \)

\[ \alpha_1(1) = \frac{1}{4} \quad \alpha_1(2) = 0 \quad \alpha_1(3) = 0 \]

\[ \alpha_2(1) = 0 \quad \alpha_2(2) = 0 \quad \alpha_2(3) = \frac{1}{12} \]

\[ \alpha_3(1) = 0 \quad \alpha_3(2) = \frac{1}{72} \quad \alpha_3(3) = \frac{1}{72} \]
Easy Question

We can cheaply compute

$$\alpha_t(i) = P(O_1O_2...O_t \land q_t = S_i)$$

(How) can we cheaply compute

$$P(O_1O_2...O_t)$$

(How) can we cheaply compute

$$P(q_t = S_i | O_1O_2...O_t)$$
Easy Question

We can cheaply compute

$$\alpha_t(i) = P(O_1 O_2 \ldots O_t \land q_t = S_i)$$

(How) can we cheaply compute

$$P(O_1 O_2 \ldots O_t) \quad ? \quad \sum_{i=1}^{N} \alpha_t(i)$$

(How) can we cheaply compute

$$P(q_t = S_i | O_1 O_2 \ldots O_t) \quad \frac{\alpha_t(i)}{\sum_{j=1}^{N} \alpha_t(j)}$$
Most probable path given observations

What's most probable path given $O_1O_2...O_T$, i.e.

What is $\arg\max_Q P(Q|O_1O_2...O_T)$?

Slow, stupid answer:

$$\arg\max_Q P(Q|O_1O_2...O_T)$$

$$= \arg\max_Q \frac{P(O_1O_2...O_T|Q)P(Q)}{P(O_1O_2...O_T)}$$

$$= \arg\max_Q P(O_1O_2...O_T|Q)P(Q)$$
Efficient MPP computation

We’re going to compute the following variables:

\[ \delta_t(i) = \max_{q_1 q_2 \ldots q_{t-1}} P(q_1 q_2 \ldots q_{t-1} \land q_t = S_i \land O_1 \ldots O_t) \]

= The Probability of the path of Length t-1 with the maximum chance of doing all these things:

...OCCURRING

...ENDING UP IN STATE S_i

...PRODUCING OUTPUT O_1 \ldots O_t

DEFINE: \[ \text{mpp}_t(i) = \text{that path} \]

So:

\[ \delta_t(i) = \text{Prob}(\text{mpp}_t(i)) \]
The Viterbi Algorithm

\[ \delta_t(i) = \max q_1 q_2 \ldots q_{t-1} P(q_1 q_2 \ldots q_{t-1} \land q_t = S_i \land O_1 O_2 \ldots O_t) \]

\[ \text{arg max} \ mpp_t(i) = q_1 q_2 \ldots q_{t-1} P(q_1 q_2 \ldots q_{t-1} \land q_t = S_i \land O_1 O_2 \ldots O_t) \]

\[ \delta_1(i) = \text{one choice} \ P(q_1 = S_i \land O_1) \]

\[ = P(q_1 = S_i) P(O_1 | q_1 = S_i) \]

\[ = \pi_i b_i(O_1) \]

Now, suppose we have all the \( \delta_t(i) \)'s and \( mpp_t(i) \)'s for all \( i \).

**HOW TO GET** \( \delta_{t+1}(j) \) and \( mpp_{t+1}(j) \)?

\( mpp_t(1) \)

\( \text{Prob} = \delta_t(1) \)

\( S_1 \)

\( mpp_t(2) \)

\( \text{Prob} = \delta_t(2) \)

\( S_2 \)

\( \vdots \)

\( mpp_t(N) \)

\( \text{Prob} = \delta_t(N) \)

\( S_N \)

\( q_t \)

\( q_{t+1} \)

\( S_j \)
The Viterbi Algorithm

The most prob path with last two states $S_i \ S_j$

is

the most prob path to $S_i$, followed by transition $S_i \rightarrow S_j$
The Viterbi Algorithm

The most probable path with last two states $S_i \rightarrow S_j$

is the most probable path to $S_i$, followed by transition $S_i \rightarrow S_j$

What is the prob of that path?

$\delta_t(i) \times P(S_i \rightarrow S_j \land O_{t+1} | \lambda)$

$= \delta_t(i) \ a_{ij} \ b_j (O_{t+1})$

SO The most probable path to $S_j$ has $S_i^*$ as its penultimate state

where $i^* = \text{argmax}_i \delta_t(i) \ a_{ij} \ b_j (O_{t+1})$
The Viterbi Algorithm

What is the prob of that path?
\[ \delta_t(i) \times P(S_i \rightarrow S_j \land O_{t+1} \mid \lambda) \]

\[ = \delta_t(i) \ a_{ij} \ b_j(O_{t+1}) \]

SO The most probable \( S_i^* \) as its penultimate state

where \( i^* = \text{argmax} \ \delta_t(i) \ a_{ij} \ b_j(O_{t+1}) \)

The most prob path with last two states \( S_i, S_j \)
is
the most prob path to \( S_i \), followed by transition \( S_i \rightarrow S_j \)

Summary:
\[ \delta_{t+1}(j) = \delta_t(i^*) \ a_{ij} \ b_j(O_{t+1}) \]
\[ \text{mpp}_{t+1}(j) = \text{mpp}_{t+1}(i^*)S_{i^*} \] with \( i^* \) defined to the left
What’s Viterbi used for?

Classic Example

Speech recognition:

Signal $\rightarrow$ words

HMM $\rightarrow$ observable is signal

$\rightarrow$ Hidden state is part of word formation

What is the most probable word given this signal?

UTTERLY GROSS SIMPLIFICATION

In practice: many levels of inference; not one big jump.