Hidden Markov Models

Manuela Veloso

Co-instructor: Pradeep Ravikumar

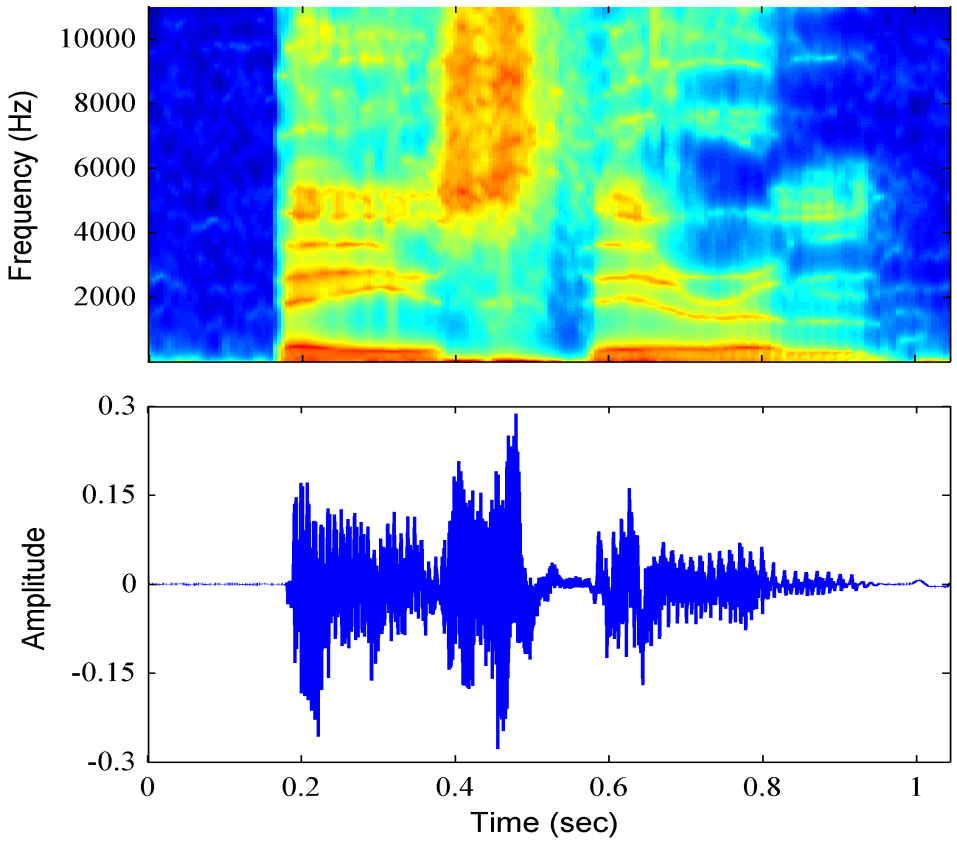
Machine Learning 10-701

Some slides courtesy of Andrew Moore and Eric Xing

Sequential data

- So far we assumed independent, identically distributed data $\{X_i\}_{i=1}^n \overset{iid}{\sim} p(\mathbf{X})$
- Sequential data
  - Time-series data

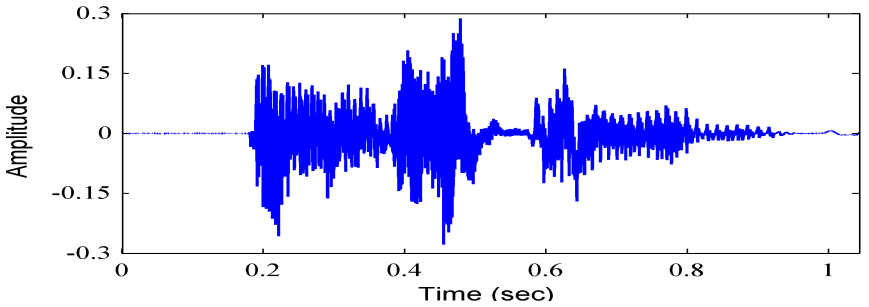
    E.g. Speech



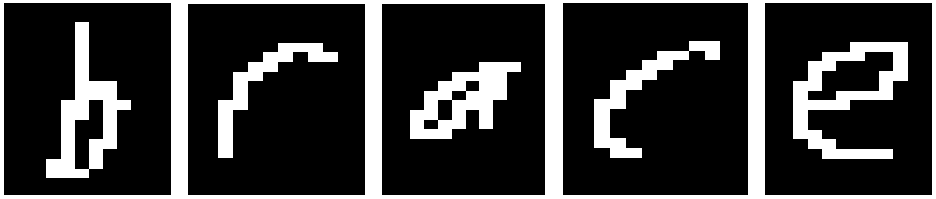
i.i.d. to sequential data

- So far we assumed independent, identically distributed data $\{X_i\}_{i=1}^n \overset{iid}{\sim} p(\mathbf{X})$
- Sequential data
  - Time-series data

    E.g. Speech



  - Characters in a sentence

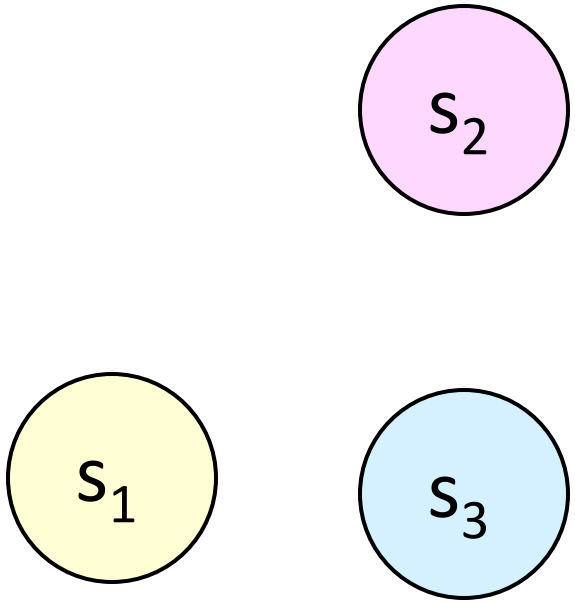


  - Mobile robots

A Markov System

N states, called s_1, s_2 .. s_N

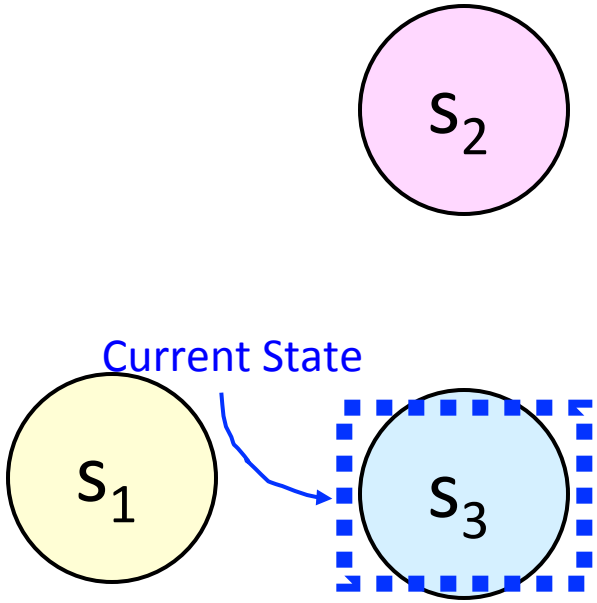
Discrete timesteps, *t=0, t=1, …*



N = 3

t=0

A Markov System



N = 3

t=0

$q_t=q_0=s_3$

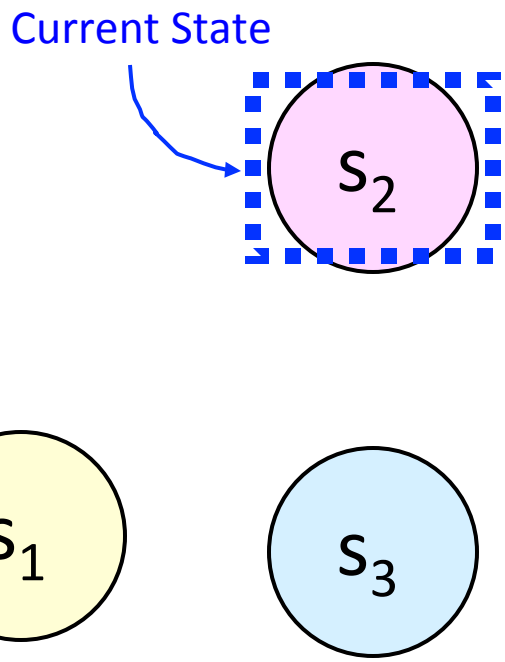
N states, called s_1, s_2 .. s_N

Discrete timesteps, *t=0, t=1, …*

On timestep t, 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



N states, called s_1, s_2 .. s_N

Discrete timesteps, *t=0, t=1, …*

On timestep t, the system is in exactly one of the available states. Call it q_t

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

After a timestep, the next state is chosen randomly according to a transition probability.

N = 3

t=1

$q_t=q_1=s_2$

A Markov System

N states, called s_1, s_2 .. s_N

Discrete timesteps, *t=0, t=1, …*

On timestep t, the system is in exactly one of the available states. Call it q_t

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

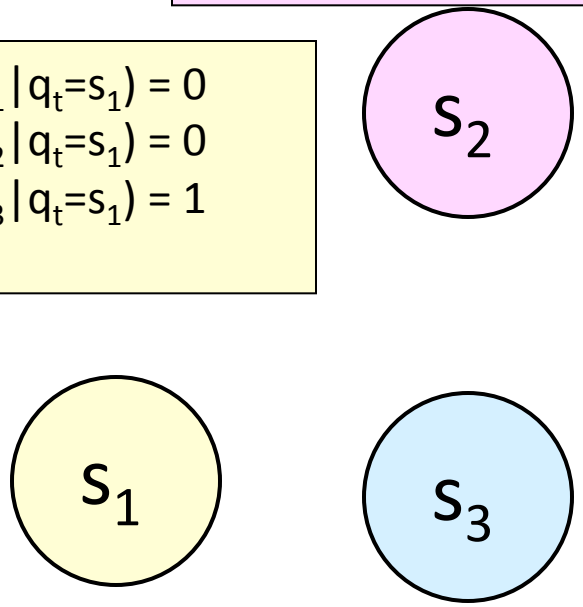
After a timestep, the next state is chosen randomly according to a transition probability.

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

$P(q_{t+1}=s_1 | q_t=s_2) = 1/2$
$P(q_{t+1}=s_2 | q_t=s_2) = 1/2$
$P(q_{t+1}=s_3 | q_t=s_2) = 0$

$P(q_{t+1}=s_1 | q_t=s_1) = 0$
$P(q_{t+1}=s_2 | q_t=s_1) = 0$
$P(q_{t+1}=s_3 | q_t=s_1) = 1$

$P(q_{t+1}=s_1 | q_t=s_3) = 1/3$
$P(q_{t+1}=s_2 | q_t=s_3) = 2/3$
$P(q_{t+1}=s_3 | q_t=s_3) = 0$

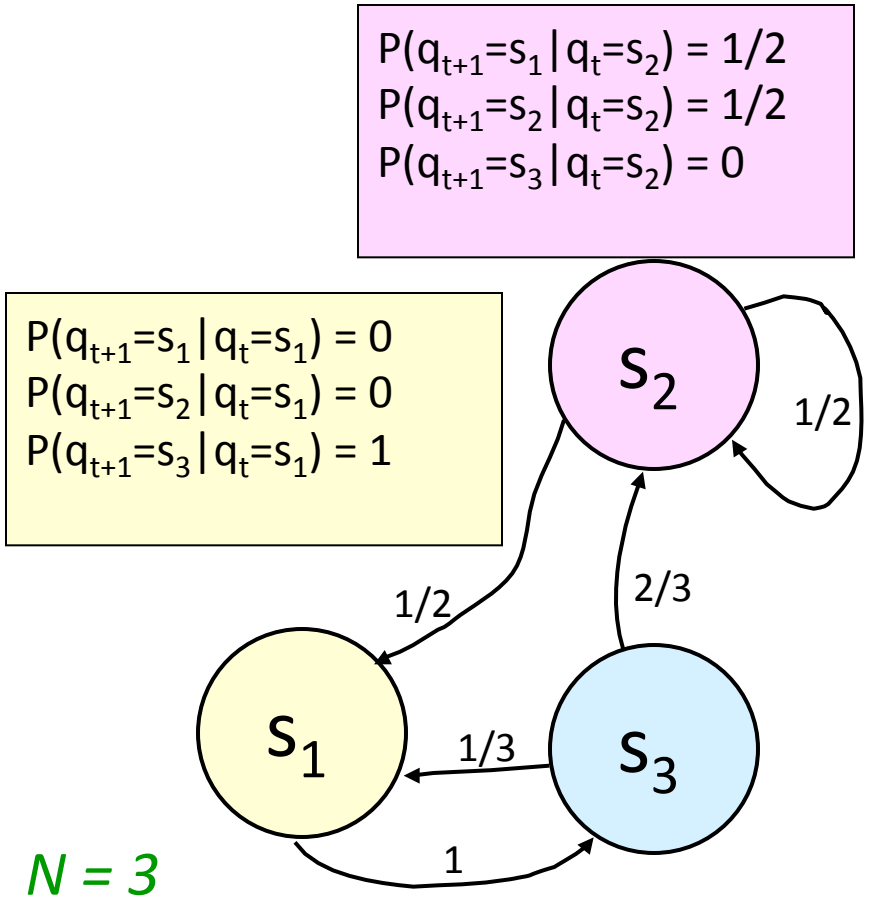


N = 3

t=1

$q_t=q_1=s_2$

A Markov System



N states, called s_1, s_2 .. s_N

Discrete timesteps, *t=0, t=1, …*

On timestep t, the system is in exactly one of the available states. Call it q_t

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

After a timestep, the next state is chosen randomly according to a transition probability.

N = 3

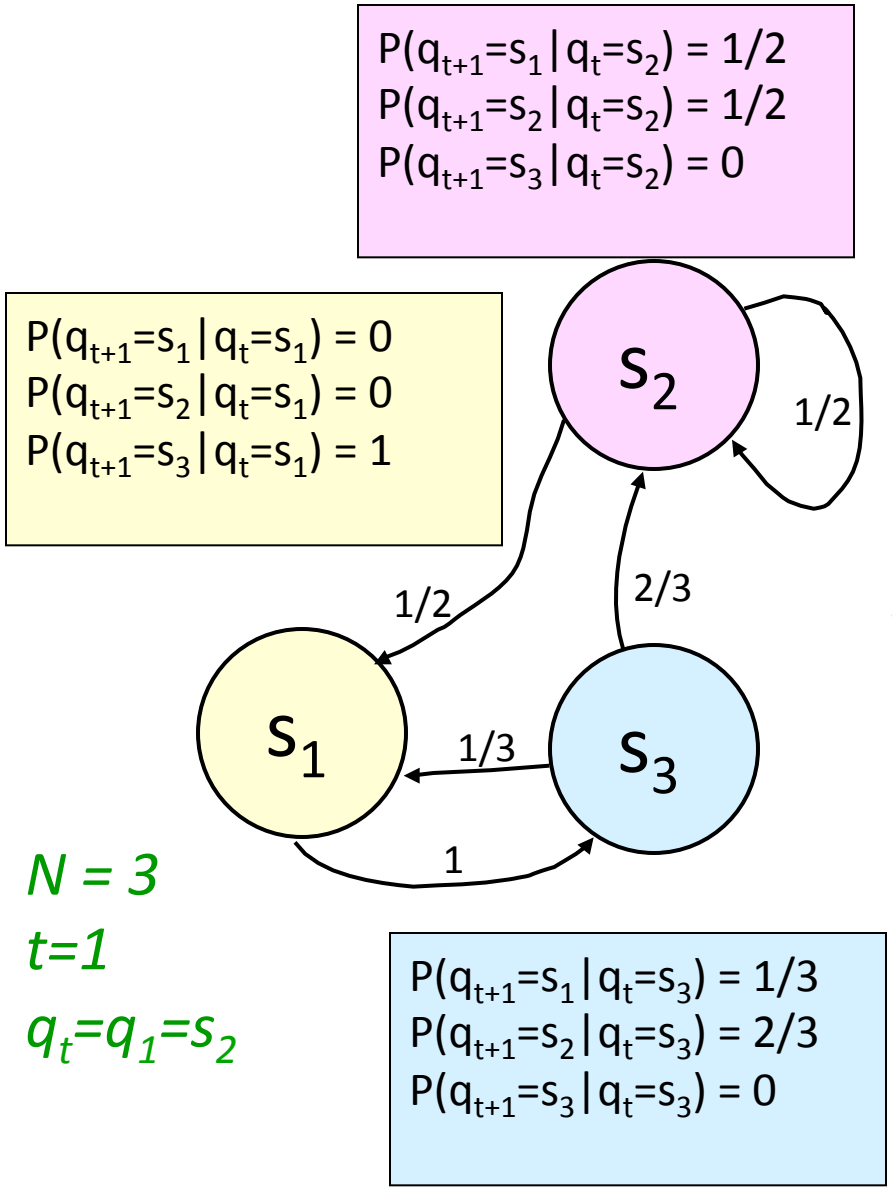
t=1

$q_t=q_1=s_2$

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

Often notated with arcs between states

Markov Property



N = 3

t=1

$q_t=q_1=s_2$

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

In other words:

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

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

Andrei Andreyevich Markov

Born	14 June 1856 N.S. Ryazan, Russian Empire
Died	20 July 1922 (aged 66) Petrograd, Russian SFSR
Residence	Russia
Nationality	Russian
Known for	Markov chains; Markov processes; stochastic processes
Scientific career	
Fields	Mathematician
Institutions	St. Petersburg University
Doctoral advisor	Pafnuty Chebyshev

Pafnuty Chebyshev

Pafnuty Lvovich Chebyshev

Born	May 16, 1821 Akatovo, Kaluga Governorate, Russian Empire
Died	December 8, 1894 (aged 73) St. Petersburg, Russian Empire

As always, thanks to wikipedia

Markov Models

- Joint Distribution

$$
\begin{aligned}
p(\mathbf{X}) &= p(X_1, X_2, \ldots, X_n) \\
&= p(X_1)p(X_2|X_1)p(X_3|X_2, X_1)\ldots p(X_n|X_{n-1}, ..., X_1) \\
&= \prod_{i=1}^{n} p(X_n|X_{n-1}, \ldots, X_1) \qquad \text{Chain rule}
\end{aligned}
$$

- Markov Assumption (m^{th} order)

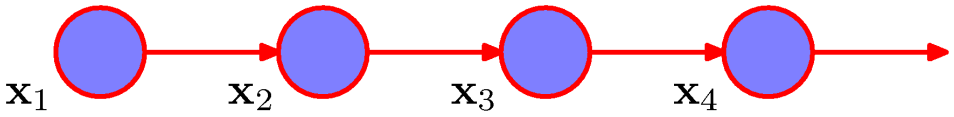
$$
p(\mathbf{X}) = \prod_{i=1}^{n} p(X_n|X_{n-1}, \ldots, X_{n-m})
$$

Current observation only depends on past m observations

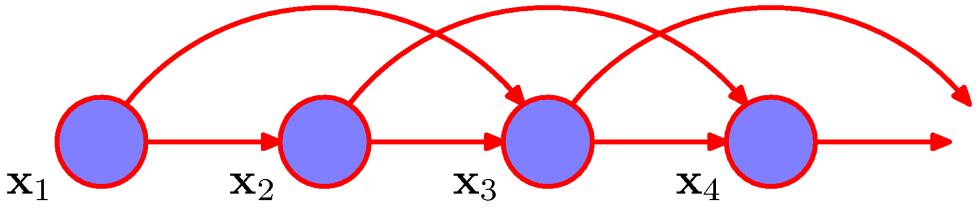
Markov Models

- Markov Assumption

1st order $$p(\mathbf{X}) = \prod_{i=1}^{n} p(X_n|X_{n-1})$$



2nd order $$p(\mathbf{X}) = \prod_{i=1}^{n} p(X_n|X_{n-1}, X_{n-2})$$



Markov Models

parameters in stationary model K-ary variables

- Markov Assumption

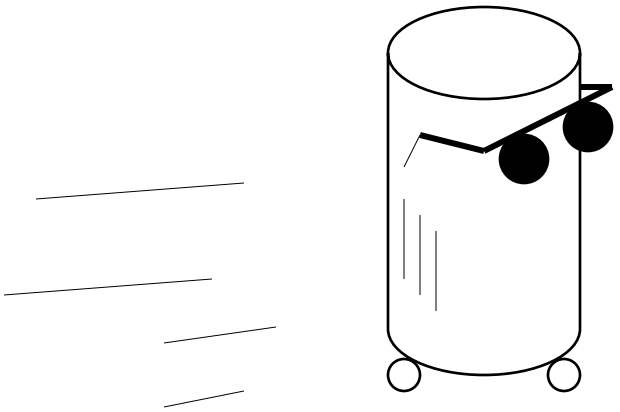
1st order $$p(\mathbf{X}) = \prod_{i=1}^{n} p(X_n|X_{n-1})$$ $O(K^2)$

mth order $$p(\mathbf{X}) = \prod_{i=1}^{n} p(X_n|X_{n-1},\ldots,X_{n-m})$$ $O(K^{m+1})$

n-1th order $$p(\mathbf{X}) = \prod_{i=1}^{n} p(X_n|X_{n-1},\ldots,X_1)$$ $O(K^n)$

≡ no assumptions – complete (but directed) graph

A Blind Robot



A Human and a Robot wander around randomly on a grid…

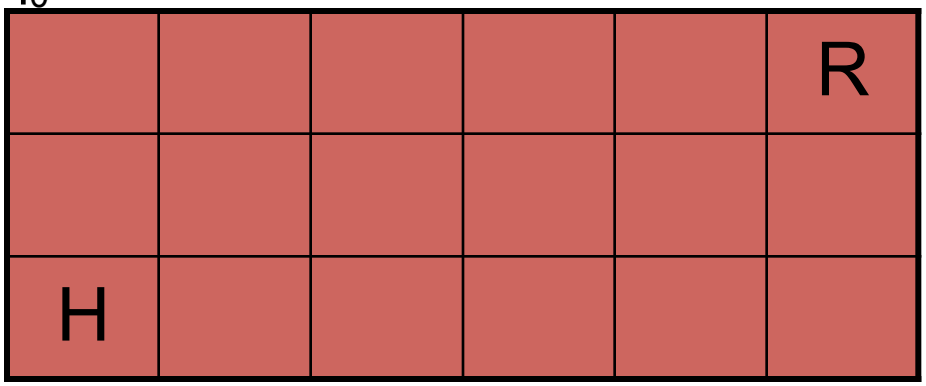
			R		
	H				

STATE q = ⟨ Location of Robot, Location of Human ⟩

Note: N (num. states) = 18 * 18 = 324 (H and R)

Dynamics of System

q_0 =



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

Typical Questions:

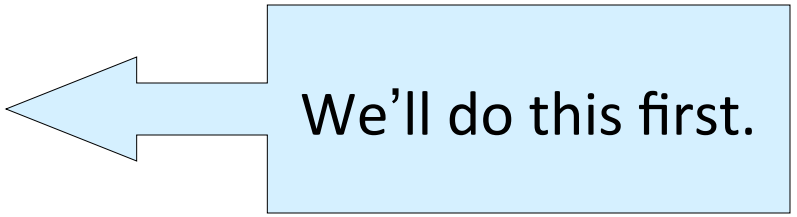
- "What is the expected time until the Human is met by the Robot?"
- "What is the probability that the Robot will hit the left wall before it meets the Human?"
- "What is the probability the Robot meets the Human on next time step?"

Example Question

"It's currently time t, and human has not been met yet. What's the probability of being met by the robot at time t + 1?"

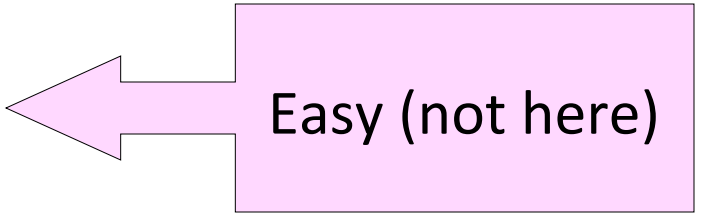
If robot is blind:

We can compute this in advance.



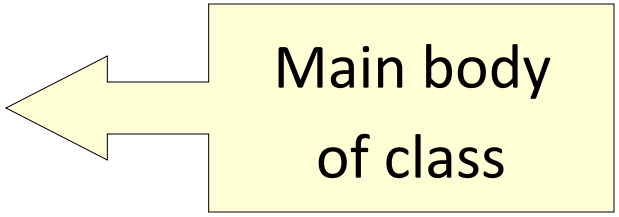
If robot is omnipotent:

(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 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., $P(q_1)=1$)

$$P(q_1\ q_2 \ldots q_t) = P(q_1\ q_2 \ldots q_{t-1})\ P(q_t \mid q_1\ q_2 \ldots q_{t-1})$$
$$= P(q_1\ q_2 \ldots q_{t-1})\ P(q_t \mid q_{t-1}) \quad \textit{Markov property}$$
$$= P(q_2 \mid q_1) P(q_3 \mid q_2) \ldots P(q_t \mid q_{t-1})$$

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

What is P(q_t =s) ? efficient answer

- For each state s_i, define

  $p_t(i)$ = Prob. state is s_i 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 $P(q_t = s)$? efficient answer

- For each state s_i, define

  $p_t(i)$ = Prob. state is s_i 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 $P(q_t = s)$? efficient answer

- For each state s_i, define

  $p_t(i)$ = Prob. state is s_i at time t

  $= P(q_t = s_i)$

- Easy to do inductive definition

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

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

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

What is $P(q_t = s)$? efficient answer

- For each state s_i, define

  $p_t(i)$ = Prob. state is s_i at time t

  $= P(q_t = s_i)$

- Easy to do inductive definition

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

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

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

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

State Transition Probability

$$a_{ij} = P(q_{t+1} = s_j \mid q_t = s_i)$$

What is $P(q_t = s)$? efficient answer

- For each state s_i, define

  $p_t(i)$ = Prob. state is s_i at time t

  $= P(q_t = s_i)$

- Easy to do inductive definition

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

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

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

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

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

t	$p_t(1)$	$p_t(2)$	…	$p_t(N)$
0	0	1		0
1				
:				
t_{final}				

What is $P(q_t = s)$? efficient answer

- For each state s_i, define

  $p_t(i)$ = Prob. state is s_i at time t

  $= P(q_t = s_i)$

- Easy to do inductive definition

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

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

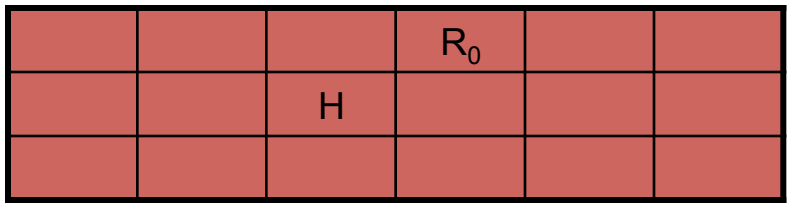
$$\sum_{i=1}^{N} P(q_{t+1} = s_j \wedge q_t = s_i) =$$

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

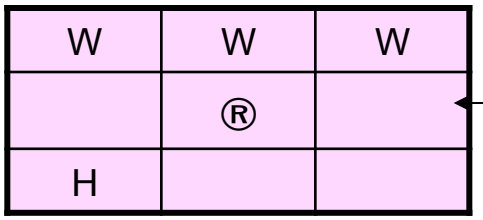
- Cost of computing $P_t(i)$ for all states S_i is now $O(t\ N^2)$
- The slow way was $O(N^t)$
- This was a simple example, as *Dynamic Programming,* used by HMMs algorithms.

Hidden State

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



True state q_t

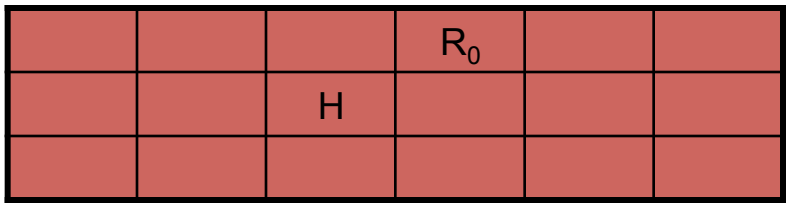


What the robot sees:
Observation O_t

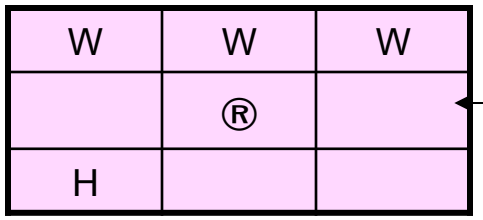
W denotes "WALL"

Noisy Hidden State

- Realistic: Noisy Proximity sensors. (unreliable contents of the 8 adjacent squares)

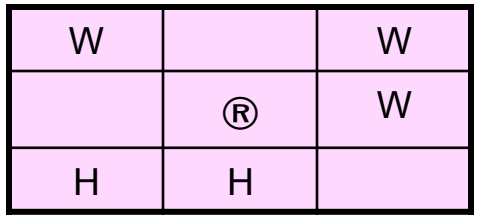


True state q_t



W denotes "WALL"

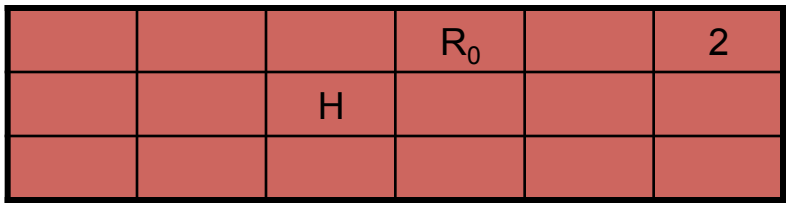
Uncorrupted Observation



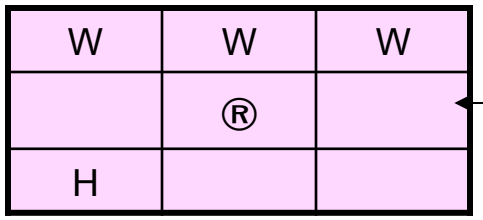
What the robot sees: Observation O_t

Noisy Hidden State

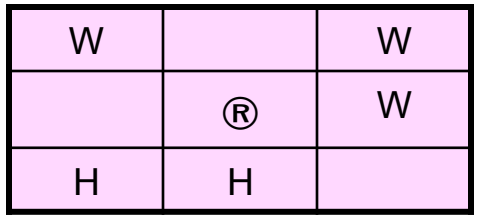
- Example: Noisy Proximity sensors. (unreliably tell us the contents of the 8 adjacent squares)



True state q_t



Uncorrupted Observation



What the robot sees: Observation O_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.

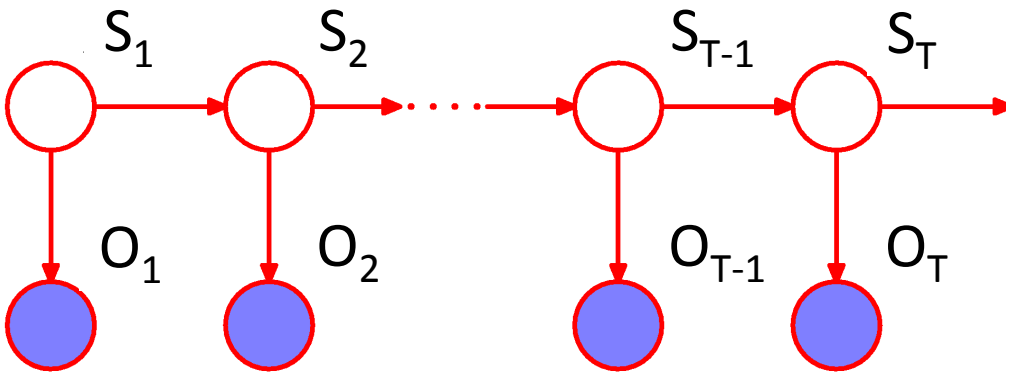
I.e.:

$P(O_t = X \mid q_t = s_i) =$

$P(O_t = X \mid q_t = s_i$, any earlier history)

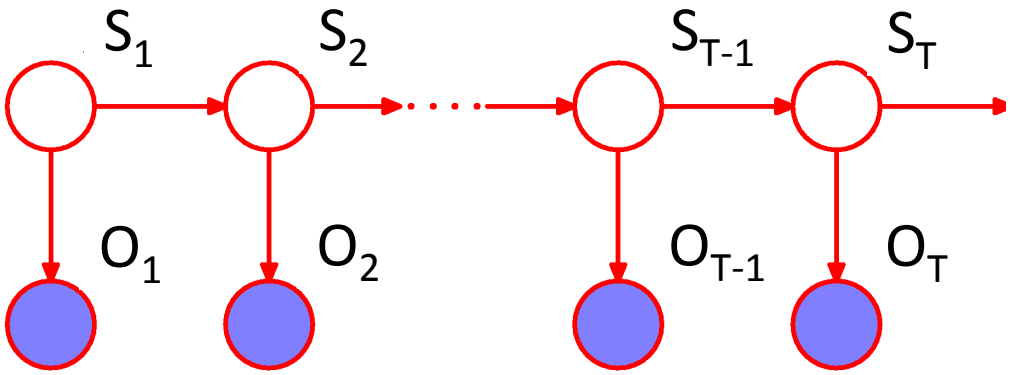
Hidden Markov Models

- Distributions that characterize sequential data with few parameters but are not limited by strong Markov assumptions.



Observation space	$O_t \in \{y_1, y_2, \ldots, y_K\}$
Hidden states	$S_t \in \{1, \ldots, I\}$

Hidden Markov Models



$$p(S_1, \ldots, S_T, O_1, \ldots, O_T) = \prod_{t=1}^{T} p(O_t|S_t) \prod_{t=1}^{T} p(S_t|S_{t-1})$$

1st order Markov assumption on hidden states $\{S_t\}$ t = 1, …, T
(can be extended to higher order).

Hidden Markov Models

- Parameters – stationary/homogeneous Markov model (independent of time t)

Initial probabilities

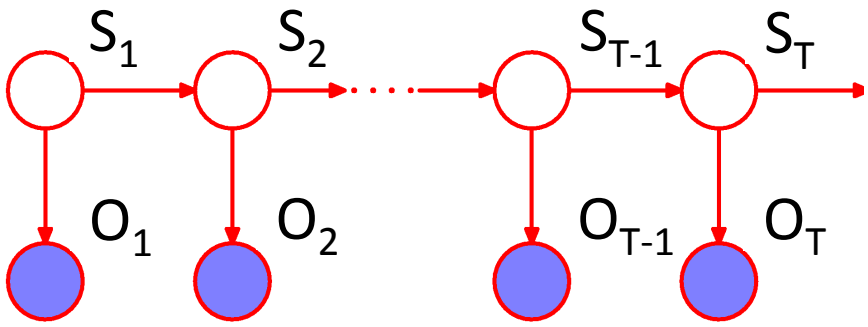
$p(S_1 = i) = \pi_i$

Transition probabilities

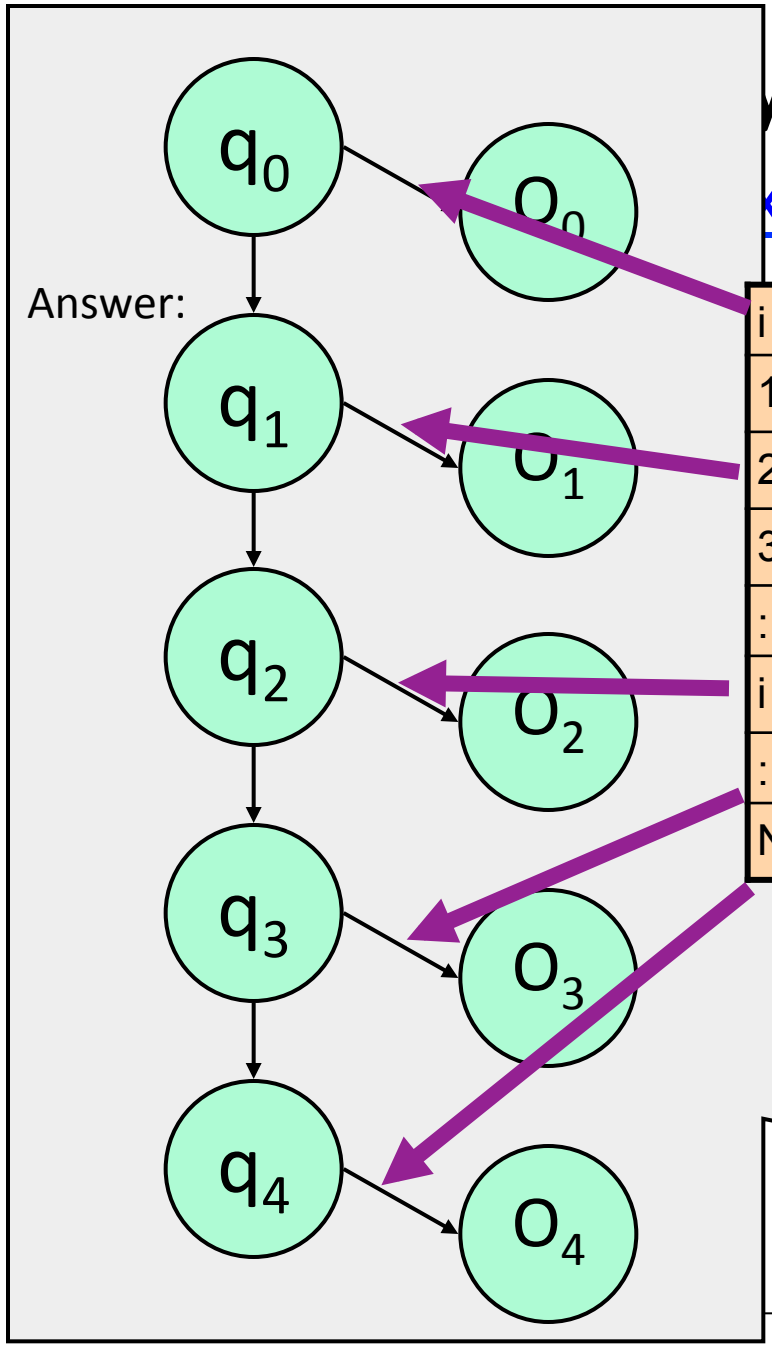
$p(S_t = j | S_{t-1} = i) = p_{ij}$

Emission probabilities

$p(O_t = y | S_t = i) = q_i^y$



$$p(\{S_t\}_{t=1}^T, \{O_t\}_{t=1}^T) = p(S_1) \prod_{t=2}^{T} p(S_t|S_{t-1}) \prod_{t=1}^{T} p(O_t|S_t)$$



y Hidden

kimity sens

s

Notation:

$$b_i(k) = P(O_t = k \mid q_t = s_i)$$

i	$P(O_t=1 \mid q_t=s_i)$	$P(O_t=2 \mid q_t=s_i)$	…	$P(O_t=k \mid q_t=s_i)$	…	$P(O_t=M \mid q_t=s_i)$
1	$b_1(1)$	$b_1(2)$	…	$b_1(k)$	…	$b_1(M)$
2	$b_2(1)$	$b_2(2)$	…	$b_2(k)$	…	$b_2(M)$
3	$b_3(1)$	$b_3(2)$	…	$b_3(k)$	…	$b_3(M)$
:	:	:	:	:	:	:
i	$b_i(1)$	$b_i(2)$	…	$b_i(k)$	…	$b_i(M)$
:	:	:	:	:	:	:
N	$b_N(1)$	$b_N(2)$	…	$b_N(k)$	…	$b_N(M)$

H	H	

What the robot sees:
Observation O_t

Hidden Markov Models

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

- Question 1: State Estimation

  What is $P(q_T=S_i \mid O_1O_2\ldots O_T)$

  It will turn out that a new cute D.P. trick will get this for us.

- Question 2: Most Probable Path

  Given $O_1O_2\ldots O_T$, 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 $O_1O_2\ldots O_T$, what is the maximum likelihood HMM that could have produced this string of observations?

  Very very useful. Uses the E.M. Algorithm

Some Famous HMM Tasks

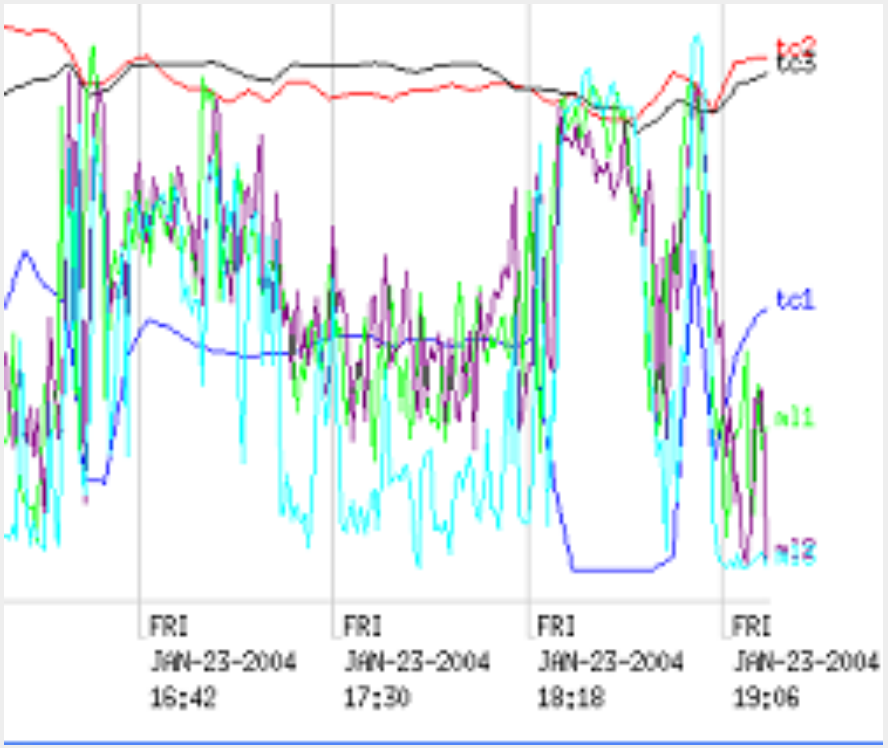
Question 1: State Estimation

What is $P(q_T=S_i \mid O_1 O_2 \ldots O_t)$

Some Famous HMM Tasks

Question 1: State Estimation

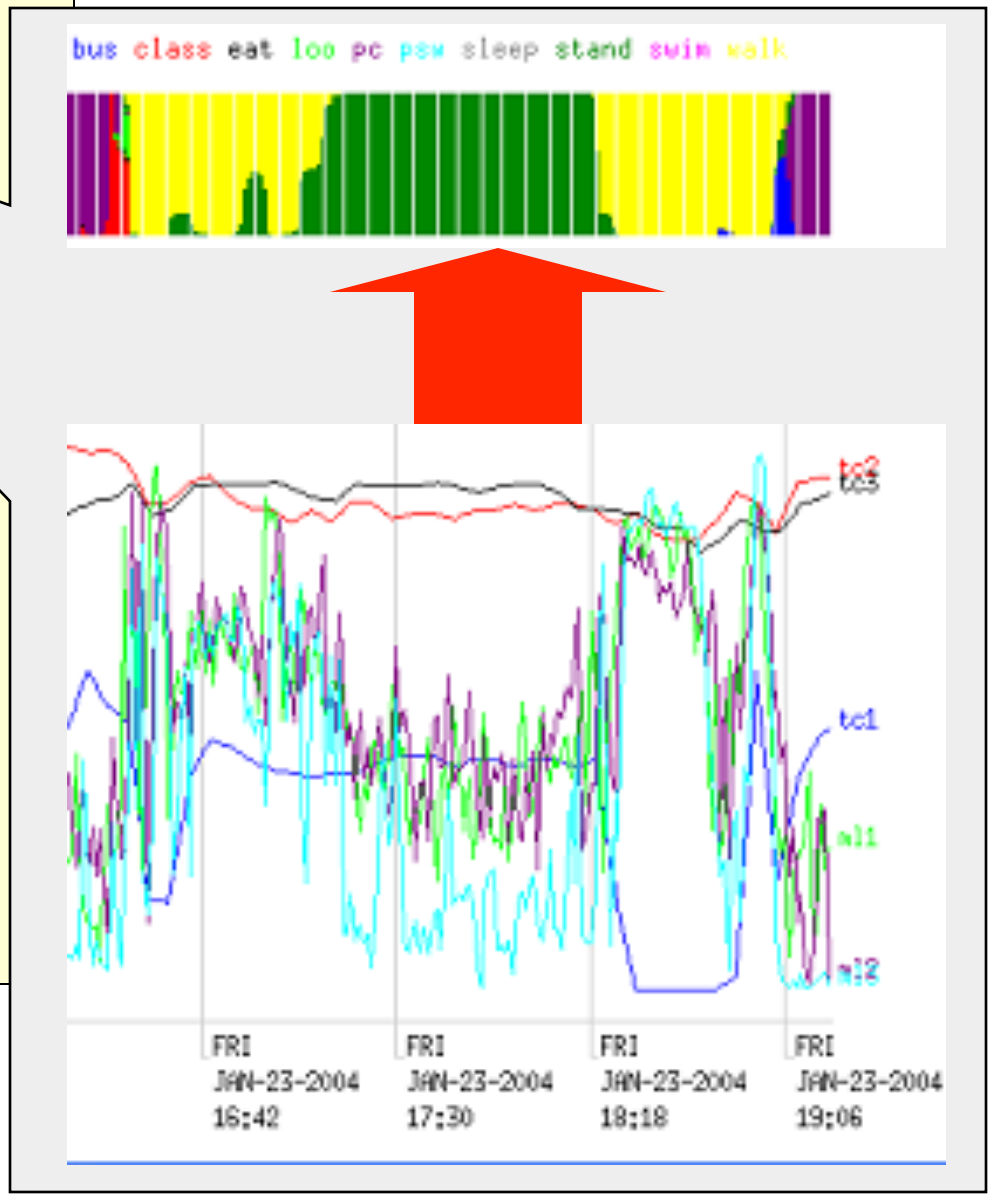
What is $P(q_T=S_i | O_1 O_2 \ldots O_t)$



Some Famous HMM Tasks

Question 1: State Estimation

What is $P(q_T=S_i | O_1 O_2 \ldots O_t)$



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?

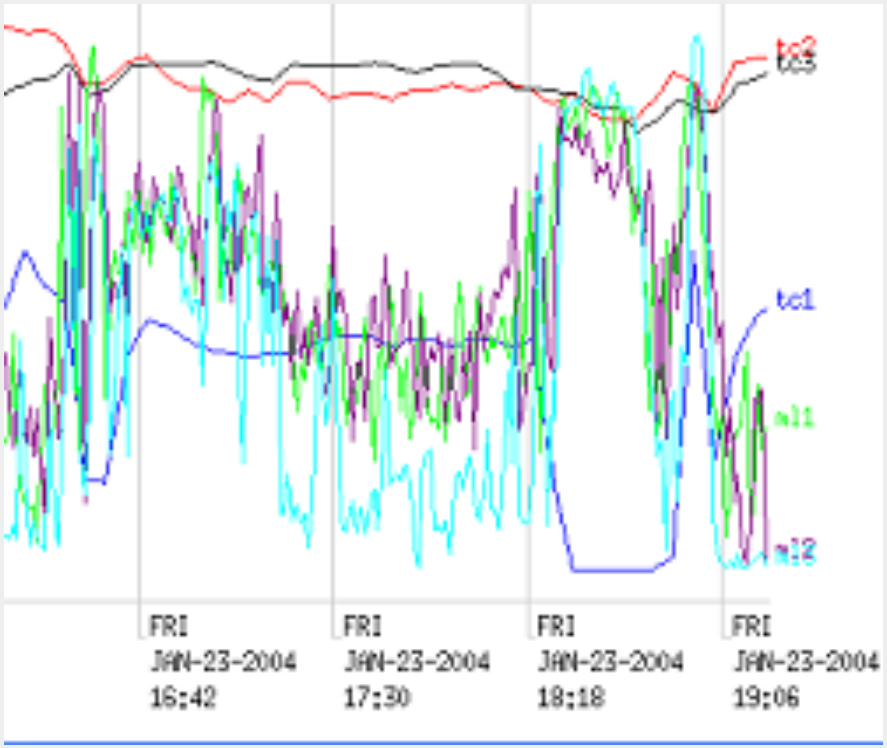
Some Famous HMM Tasks

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

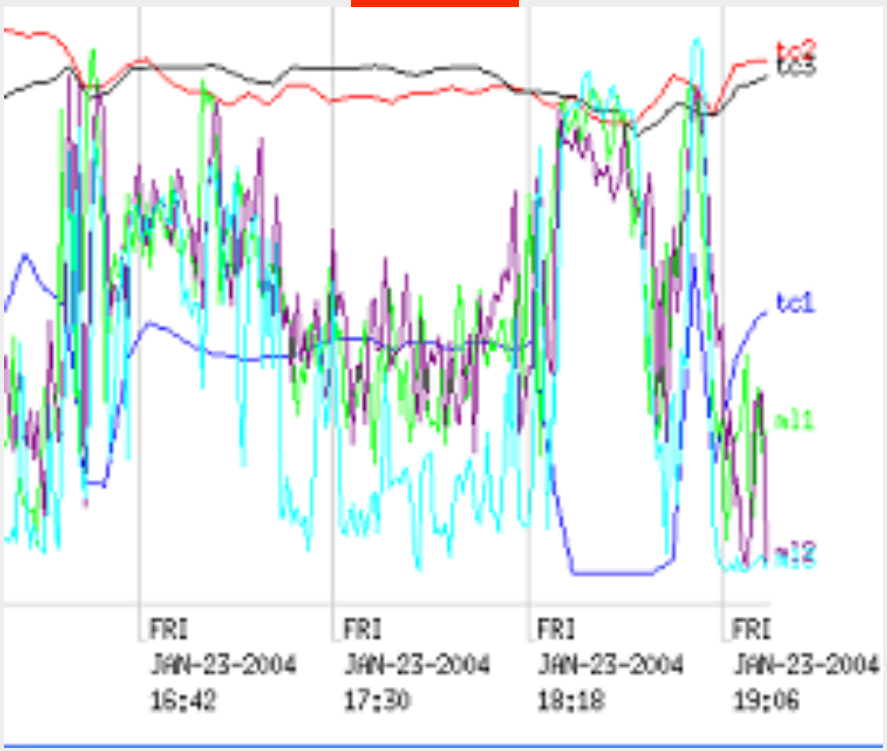
Question 1: State Estimation

What is $P(q_T=S_i \mid O_1O_2\ldots O_t)$

Question 2: Most Probable Path

Given $O_1O_2\ldots O_T$, what is the most probable path that I took?

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



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?

Question 3: Learning HMMs:

Given $O_1O_2\ldots O_T$, what is the maximum likelihood HMM that could have produced this string of observations?

Some Famou

Question 1: State Estimation

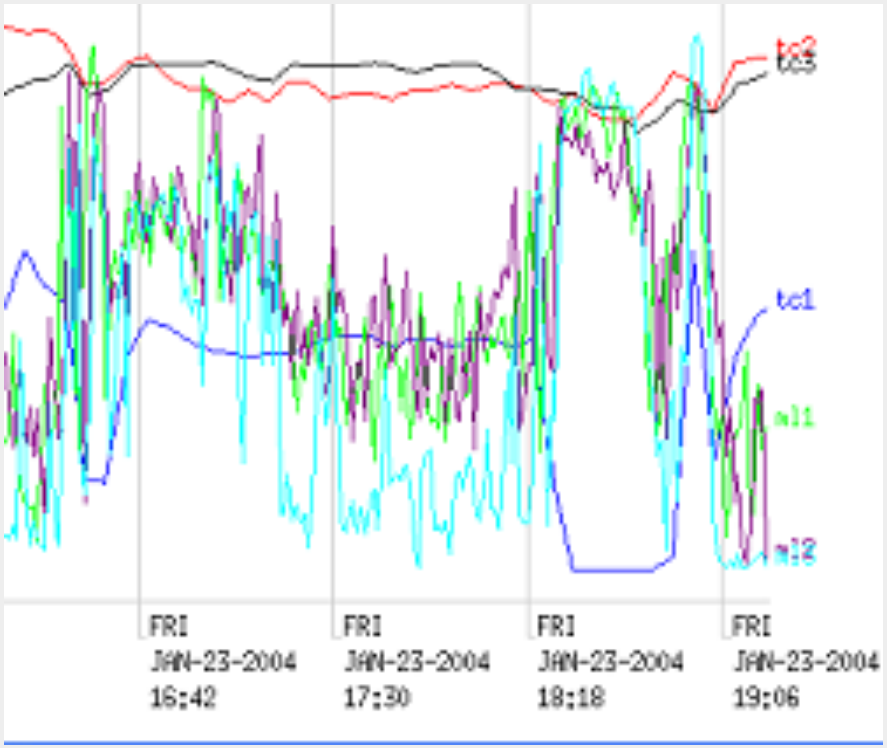
What is $P(q_T=S_i \mid O_1O_2\ldots O_T)$

Question 2: Most Probable Path

Given $O_1O_2\ldots O_T$, what is the most probable path that took?

Question 3: Learning HMMs:

Given $O_1O_2\ldots O_T$, what is the maximum likelihood HMM that could have produced this string of observations?



Some Famou

Question 1: State Estimation

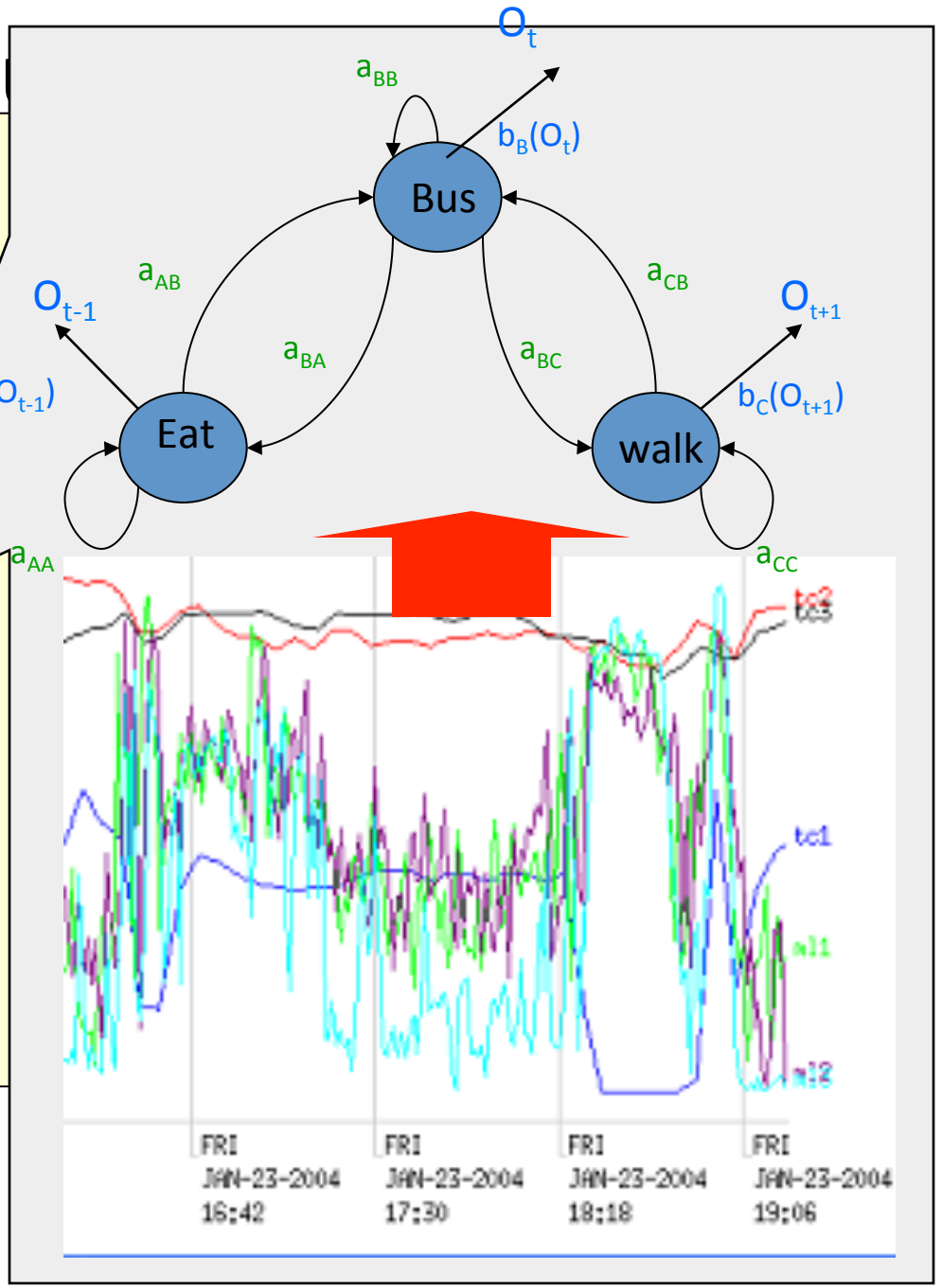
What is $P(q_T=S_i \mid O_1O_2\ldots O_T)$

Question 2: Most Probable Path

Given $O_1O_2\ldots O_T$, what is the most probable path that took?

Question 3: Learning HMMs:

Given $O_1O_2\ldots O_T$, what is the maximum likelihood HMM that could have produced this string of observations?



HMM Notation (from Rabiner's Survey)

*L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," Proc. of the IEEE, Vol.77, No.2, pp. 257--286, 1989.
Available from
http://ieeexplore.ieee.org/iel5/5/698/00018626.pdf?arnumber=18626

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

For a particular trial….

Let T be the number of observations

T also the number of states passed through

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

$Q = q_1\ q_2\ ..\ q_T$ 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

HMM Formal Definition

An HMM, λ, 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

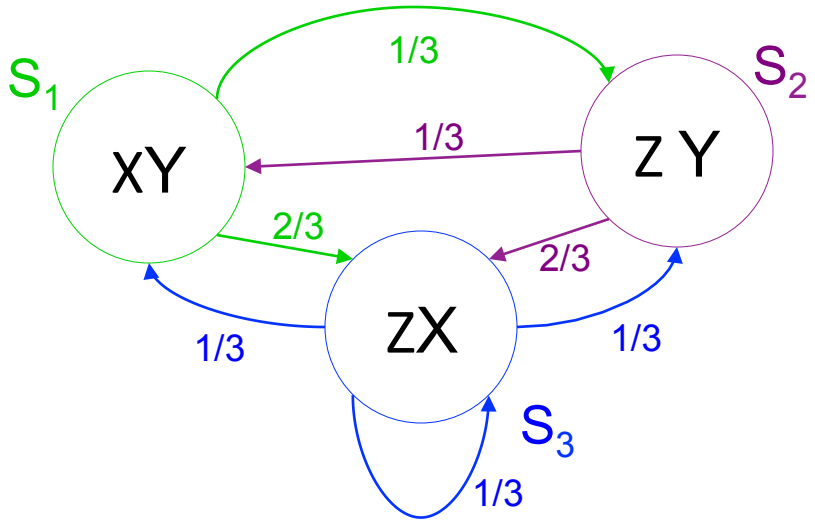
  $P(q_{t+1}=S_j \mid q_t=S_i)=a_{ij}$
- $b_1(1)b_1(2)$ … $b_1(M)$
  $b_2(1)b_2(2)$ … $b_2(M)$
  : : :
  $b_N(1)$ $b_N(2)$ … $b_N(M)$

  The observation probabilities

  $P(O_t=k \mid q_t=S_i)=b_i(k)$

HMM Example

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



N = 3

M = 3

$\pi_1 = ½$ $\pi_2 = ½$ $\pi_3 = 0$

$a_{11} = 0$ $a_{12} = ⅓$ $a_{13} = ⅔$

$a_{12} = ⅓$ $a_{22} = 0$ $a_{13} = ⅔$

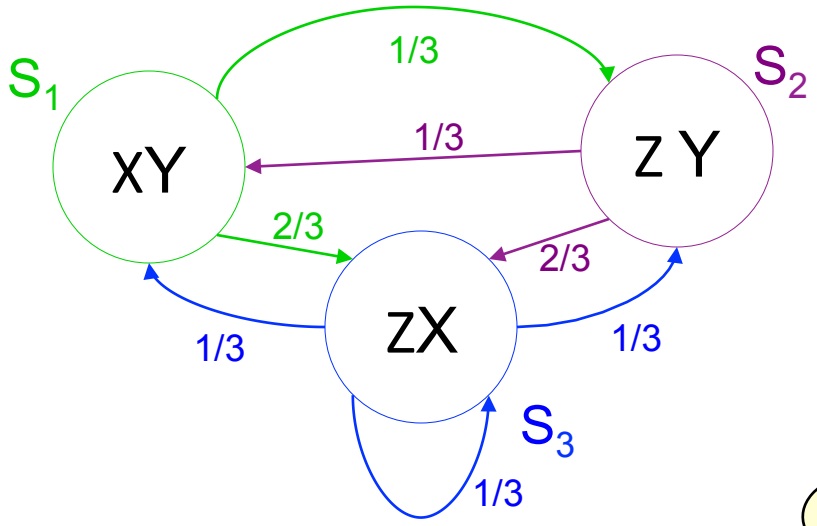
$a_{13} = ⅓$ $a_{32} = ⅓$ $a_{13} = ⅓$

$b_1(X) = ½$ $b_1(Y) = ½$ $b_1(Z) = 0$

$b_2(X) = 0$ $b_2(Y) = ½$ $b_2(Z) = ½$

$b_3(X) = ½$ $b_3(Y) = 0$ $b_3(Z) = ½$

HMM Example



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:

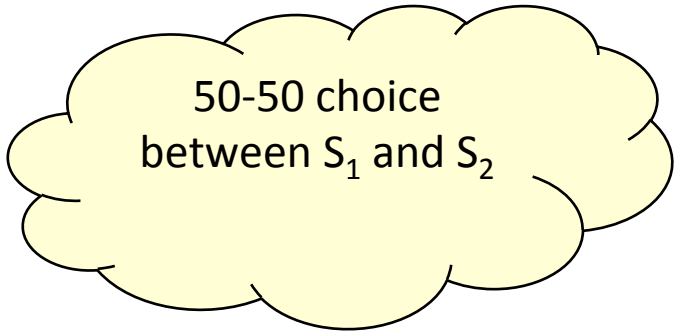
N = 3

M = 3

$\pi_1 = ½$ $\pi_2 = ½$ $\pi_3 = 0$

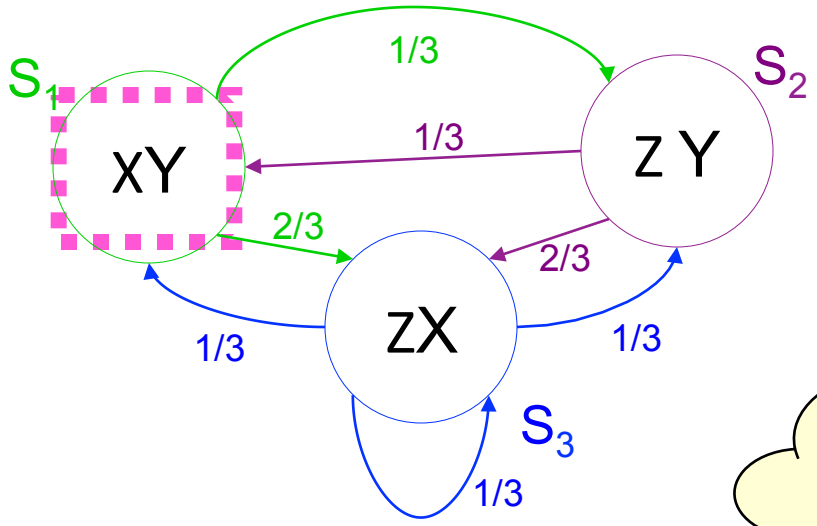
$a_{11} = 0$	$a_{12} = ⅓$	$a_{13} = ⅔$
$a_{12} = ⅓$	$a_{22} = 0$	$a_{13} = ⅔$
$a_{13} = ⅓$	$a_{32} = ⅓$	$a_{13} = ⅓$

$b_1(X) = ½$ $b_1(Y) = ½$	$b_1(Z) = 0$
$b_2(X) = 0$ $b_2(Y) = ½$	$b_2(Z) = ½$
$b_3(X) = ½$ $b_3(Y) = 0$	$b_3(Z) = ½$



$q_0=$	___	$O_0=$	___
$q_1=$	___	$O_1=$	___
$q_2=$	___	$O_2=$	___

HMM Example



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

$\pi_1 = ½$ $\pi_2 = ½$ $\pi_3 = 0$

$a_{11} = 0$	$a_{12} = ⅓$	$a_{13} = ⅔$
$a_{12} = ⅓$	$a_{22} = 0$	$a_{13} = ⅔$
$a_{13} = ⅓$	$a_{32} = ⅓$	$a_{13} = ⅓$

$b_1(X) = ½$ $b_1(Y) = ½$ $b_1(Z) = 0$

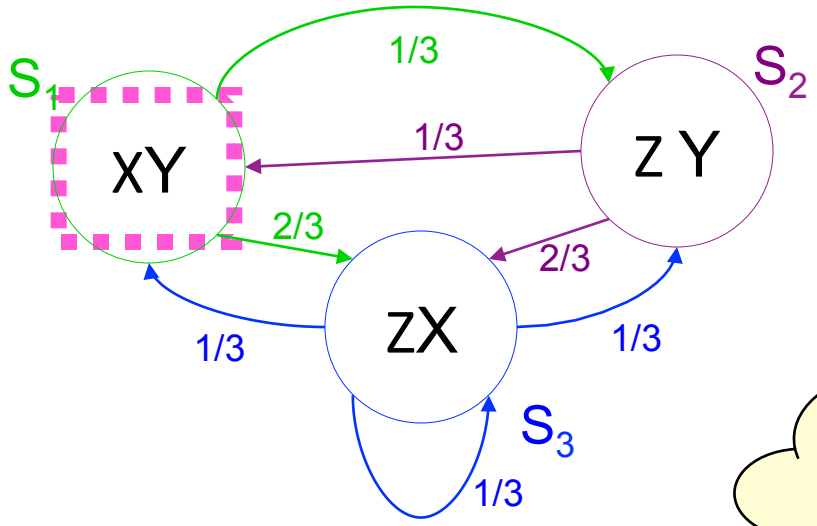
$b_2(X) = 0$ $b_2(Y) = ½$ $b_2(Z) = ½$

$b_3(X) = ½$ $b_3(Y) = 0$ $b_3(Z) = ½$

50-50 choice between X and Y

$q_0=$	S_1	$O_0=$	°
$q_1=$	___	$O_1=$	___
$q_2=$	___	$O_2=$	___

HMM Example



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

$\pi_1 = ½$ $\pi_2 = ½$ $\pi_3 = 0$

$a_{11} = 0$	$a_{12} = ⅓$	$a_{13} = ⅔$
$a_{12} = ⅓$	$a_{22} = 0$	$a_{13} = ⅔$
$a_{13} = ⅓$	$a_{32} = ⅓$	$a_{13} = ⅓$

$b_1(X) = ½$ $b_1(Y) = ½$ $b_1(Z) = 0$

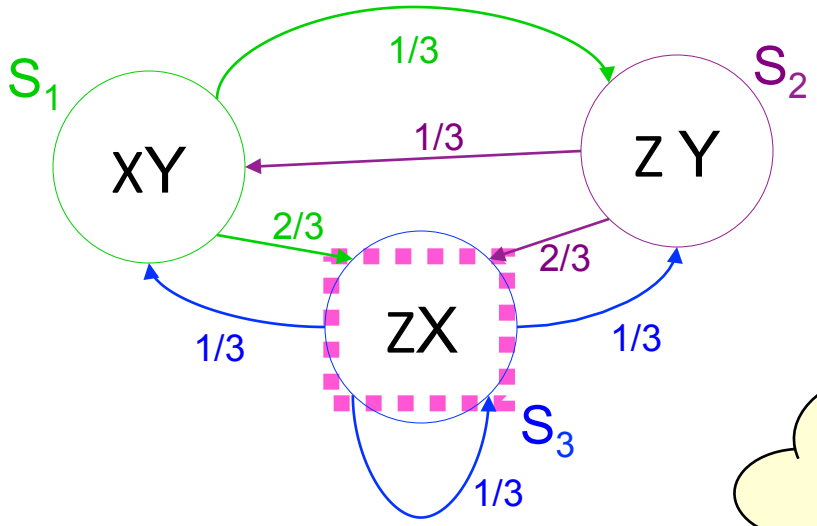
$b_2(X) = 0$ $b_2(Y) = ½$ $b_2(Z) = ½$

$b_3(X) = ½$ $b_3(Y) = 0$ $b_3(Z) = ½$

Goto S_3 with probability 2/3 or S_2 with prob. 1/3

q_0=	S_1	O_0=	X
q_1=	___	O_1=	___
q_2=	___	O_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:

N = 3

M = 3

$\pi_1 = ½$ $\pi_2 = ½$ $\pi_3 = 0$

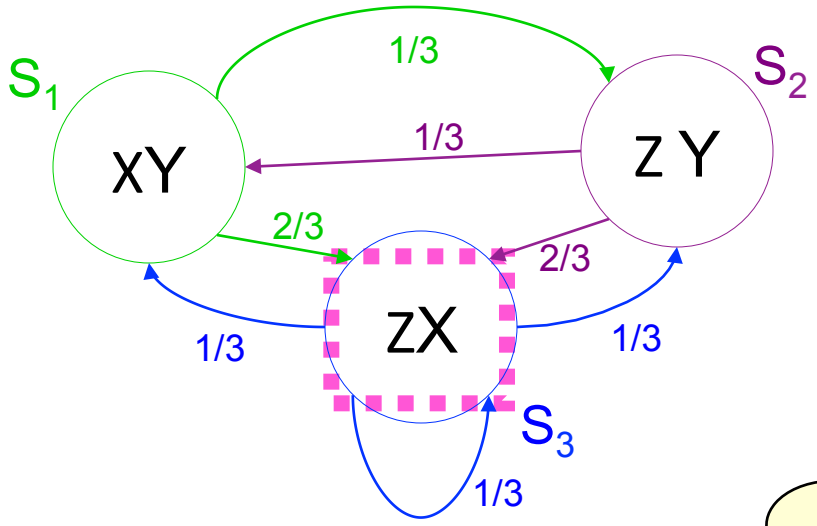
$a_{11} = 0$	$a_{12} = ⅓$	$a_{13} = ⅔$
$a_{12} = ⅓$	$a_{22} = 0$	$a_{13} = ⅔$
$a_{13} = ⅓$	$a_{32} = ⅓$	$a_{13} = ⅓$

$b_1(X) = ½$ $b_1(Y) = ½$	$b_1(Z) = 0$
$b_2(X) = 0$ $b_2(Y) = ½$	$b_2(Z) = ½$
$b_3(X) = ½$ $b_3(Y) = 0$	$b_3(Z) = ½$

50-50 choice between Z and X

q_0=	S_1	O_0=	X
q_1=	S_3	O_1=	°
q_2=	___	O_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:

N = 3

M = 3

$\pi_1 = ½$ $\pi_2 = ½$ $\pi_3 = 0$

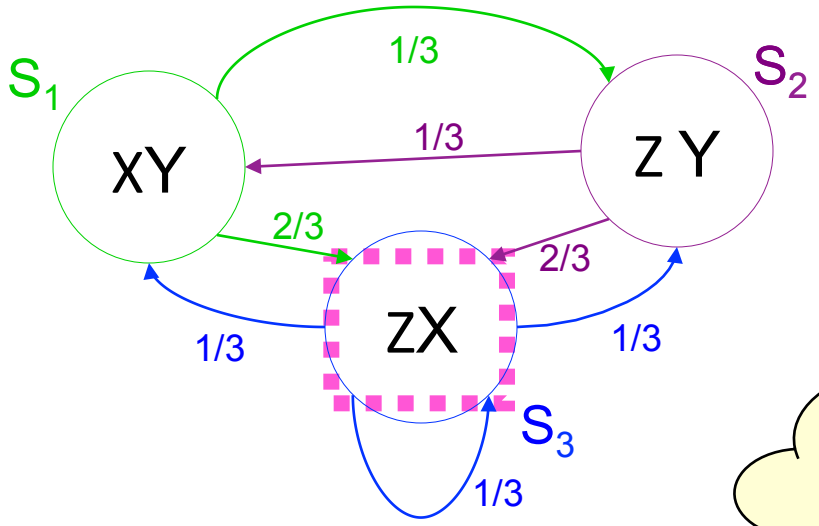
$a_{11} = 0$	$a_{12} = ⅓$	$a_{13} = ⅔$
$a_{12} = ⅓$	$a_{22} = 0$	$a_{13} = ⅔$
$a_{13} = ⅓$	$a_{32} = ⅓$	$a_{13} = ⅓$

$b_1(X) = ½$	$b_1(Y) = ½$	$b_1(Z) = 0$
$b_2(X) = 0$	$b_2(Y) = ½$	$b_2(Z) = ½$
$b_3(X) = ½$	$b_3(Y) = 0$	$b_3(Z) = ½$

Each of the three next states is equally likely

q_0=	S_1	O_0=	X
q_1=	S_3	O_1=	X
q_2=	___	O_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:

N = 3

M = 3

$\pi_1 = ½$ $\pi_2 = ½$ $\pi_3 = 0$

$a_{11} = 0$ $a_{12} = ⅓$ $a_{13} = ⅔$

$a_{12} = ⅓$ $a_{22} = 0$ $a_{13} = ⅔$

$a_{13} = ⅓$ $a_{32} = ⅓$ $a_{13} = ⅓$

$b_1(X) = ½$ $b_1(Y) = ½$ $b_1(Z) = 0$

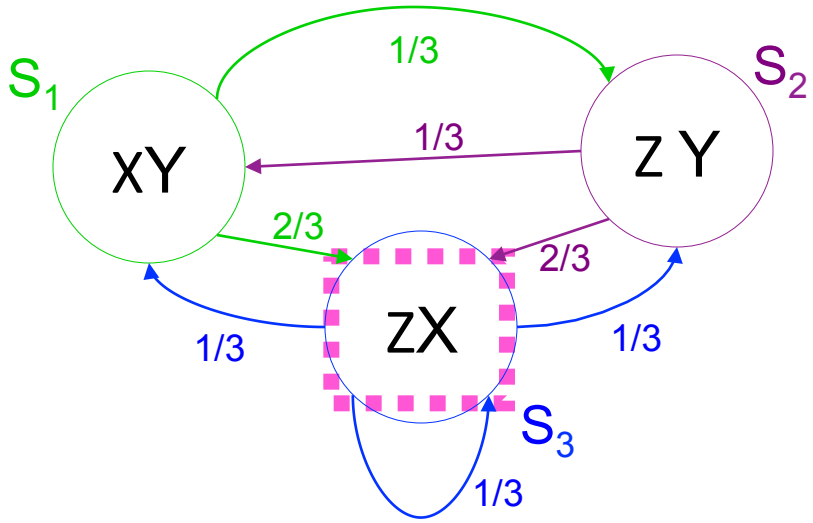
$b_2(X) = 0$ $b_2(Y) = ½$ $b_2(Z) = ½$

$b_3(X) = ½$ $b_3(Y) = 0$ $b_3(Z) = ½$

50-50 choice between Z and X

$q_0=$	S_1	$O_0=$	X
$q_1=$	S_3	$O_1=$	X
$q_2=$	S_3	$O_2=$	° ___

HMM Example



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

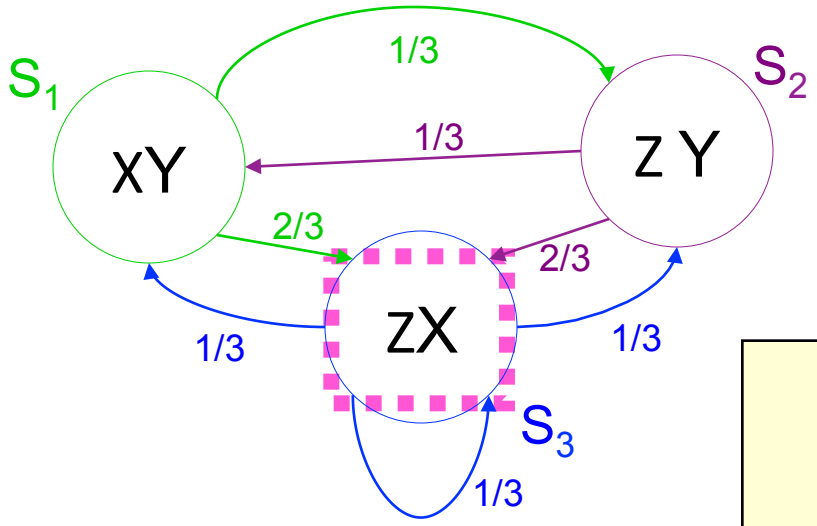
$\pi_1 = ½$ $\pi_2 = ½$ $\pi_3 = 0$

$a_{11} = 0$	$a_{12} = ⅓$	$a_{13} = ⅔$
$a_{12} = ⅓$	$a_{22} = 0$	$a_{13} = ⅔$
$a_{13} = ⅓$	$a_{32} = ⅓$	$a_{13} = ⅓$

$b_1(X) = ½$ $b_1(Y) = ½$	$b_1(Z) = 0$
$b_2(X) = 0$ $b_2(Y) = ½$	$b_2(Z) = ½$
$b_3(X) = ½$ $b_3(Y) = 0$	$b_3(Z) = ½$

q_0=	S_1	O_0=	X
q_1=	S_3	O_1=	X
q_2=	S_3	O_2=	Z

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:

N = 3

M = 3

$\pi_1 = ½$ $\pi_2 = ½$ $\pi_3 = 0$

$a_{11} = 0$ $a_{12} = ⅓$ $a_{13} = ⅔$

$a_{12} = ⅓$ $a_{22} = 0$ $a_{13} = ⅔$

$a_{13} = ⅓$ $a_{32} = ⅓$ $a_{13} = ⅓$

$b_1(X) = ½$ $b_1(Y) = ½$ $b_1(Z) = 0$

$b_2(X) = 0$ $b_2(Y) = ½$ $b_2(Z) = ½$

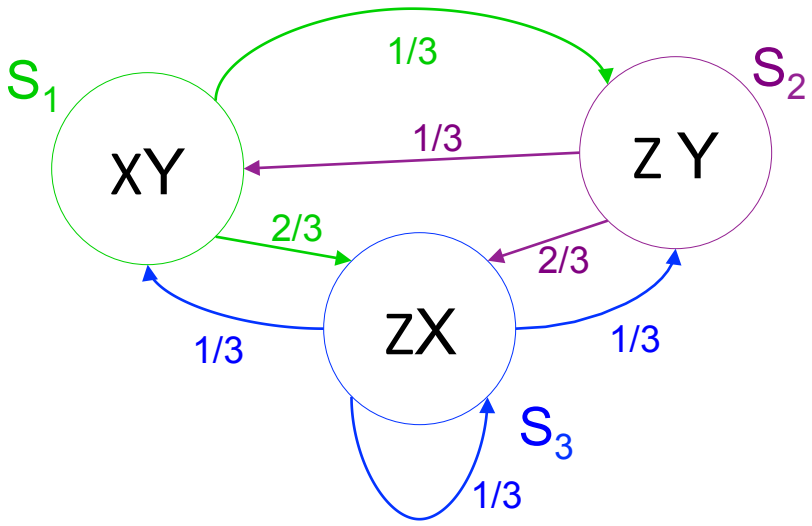
$b_3(X) = ½$ $b_3(Y) = 0$ $b_3(Z) = ½$

This is what the observer has to work with…

$q_0=$	?	$O_0=$	X
$q_1=$	?	$O_1=$	X
$q_2=$	?	$O_2=$	Z

Prob. of a series of observations

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



Slow way:

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

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

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 ^ O_2 = X ^ O_3 = Z)?

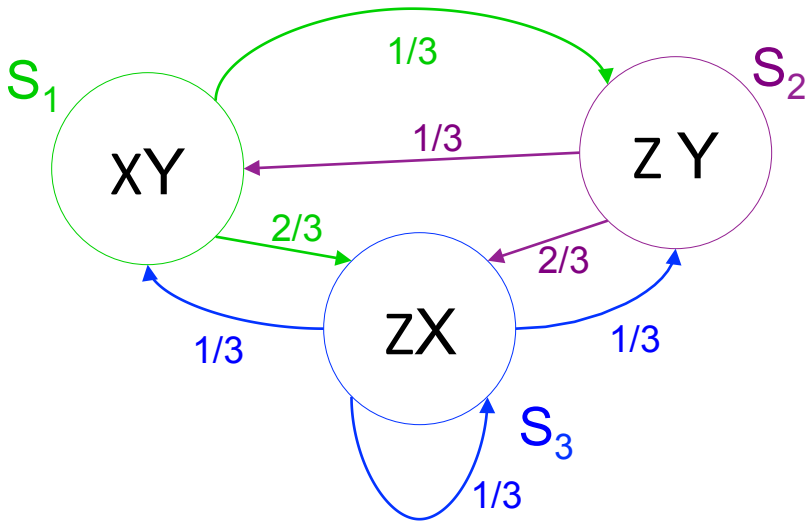
Slow way:

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

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

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

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



P(Q)= P(q_1,q_2,q_3)

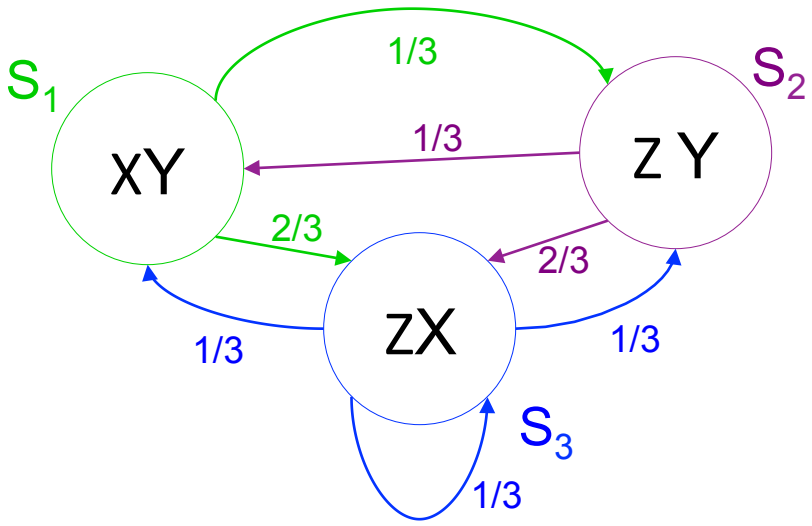
=P(q_1) P(q_2,q_3|q_1) (chain rule)

=P(q_1) P(q_2|q_1) P(q_3| q_2,q_1) (chain)

=P(q_1) P(q_2|q_1) P(q_3| q_2) (why?)
Example in the case Q = S_1 S_3 S_3:
=1/2 * 2/3 * 1/3 = 1/9

Probability of a series of observations

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



Slow way:

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

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

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)

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 * 1/2 * 1/2 = 1/8

Probability of a series of observations

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

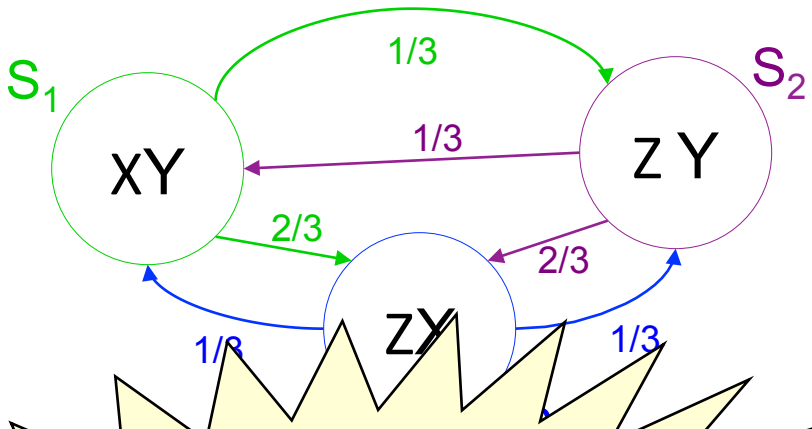
Slow way:

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

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

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

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



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 billion computations and 3.5 billion P(O|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

$$\alpha_t(i) = P(O_1\ O_2\ \ldots\ O_t \wedge q_t = S_i \mid \lambda) \qquad \text{where } 1 \le t \le T$$

$\alpha_t(i)$ = Probability that, in a random trial, we would have

- seen the first t observations
- and ended up in S_i as the state visited at time t

$\alpha_t(i)$: easy to define recursively

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

$$\alpha_1(i) = \mathrm{P}(O_1 \wedge q_1 = S_i)$$
$$= \mathrm{P}(q_1 = S_i)\mathrm{P}(O_1 \mid q_1 = S_i)$$
$$= b_i(O_1)\pi_i$$
$$\alpha_{t+1}(j) = \mathrm{P}(O_1 O_2 \ldots O_t O_{t+1} \wedge q_{t+1} = S_j)$$
$$=$$

$\alpha_t(i)$: easy to define recursively

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

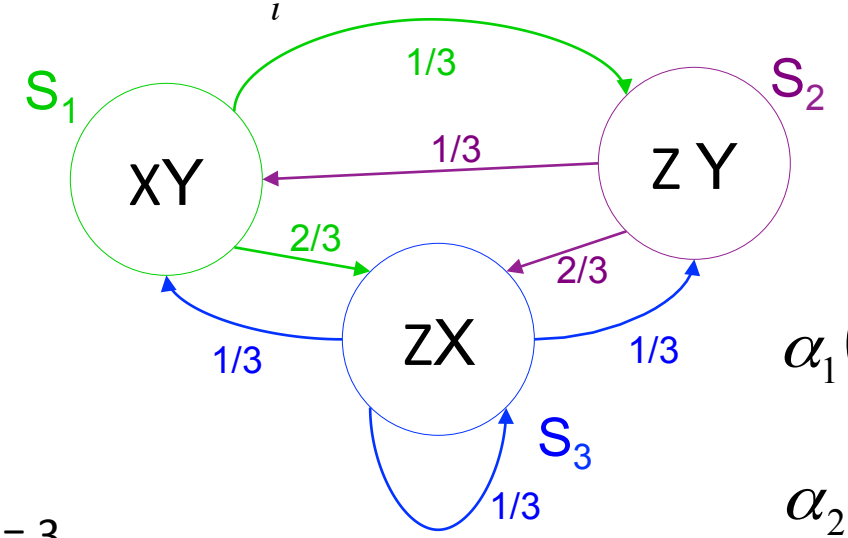
$$
\begin{aligned}
\alpha_1(i) &= \mathrm{P}(O_1 \wedge q_1 = S_i) \\
&= \mathrm{P}(q_1 = S_i)\mathrm{P}(O_1 \mid q_1 = S_i) \\
&= b_i(O_1)\pi_i \\
\alpha_{t+1}(j) &= \mathrm{P}(O_1 O_2 \ldots O_t O_{t+1} \wedge q_{t+1} = S_j) \\
&= \sum_{i=1}^{N} \mathrm{P}(O_1 O_2 \ldots O_t \wedge q_t = S_i \wedge O_{t+1} \wedge q_{t+1} = S_j) \\
&= \sum_{i=1}^{N} \mathrm{P}(O_{t+1}, q_{t+1} = S_j \mid O_1 O_2 \ldots O_t \wedge q_t = S_i)\mathrm{P}(O_1 O_2 \ldots O_t \wedge q_t = S_i) \\
&= \sum_{i} \mathrm{P}(O_{t+1}, q_{t+1} = S_j \mid q_t = S_i)\alpha_t(i) \\
&= \sum_{i} \mathrm{P}(q_{t+1} = S_j \mid q_t = S_i)\mathrm{P}(O_{t+1} \mid q_{t+1} = S_j)\alpha_t(i) \\
&= \sum_{i} a_{ij} b_j(O_{t+1})\alpha_t(i)
\end{aligned}
$$

$$\alpha_t(i) = \mathrm{P}\left(O_1 O_2 .. O_t \wedge q_t = S_i \middle| \lambda\right)$$

$$\alpha_1(i) = b_i(O_1)\pi_i$$

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

Assume $O_1\ O_2\ O_3$ = **X X Z**



N = 3

M = 3

π_1 = ½ π_2 = ½ π_3 = 0

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

$$\alpha_1(1) = \frac{1}{4} \qquad \alpha_1(2) = 0 \qquad \alpha_1(3) = 0$$

$$\alpha_2(1) = 0 \qquad \alpha_2(2) = 0 \qquad \alpha_2(3) = \frac{1}{12}$$

$$\alpha_3(1) = 0 \qquad \alpha_3(2) = \frac{1}{72} \qquad \alpha_3(3) = \frac{1}{72}$$

Easy Question

We can cheaply compute

$$\alpha_t(i)=P(O_1O_2\ldots O_t \wedge q_t=S_i)$$

(How) can we cheaply compute

$$P(O_1O_2\ldots O_t) \quad ?$$

(How) can we cheaply compute

$$P(q_t=S_i|O_1O_2\ldots O_t)$$

Following Easy Questions

We can cheaply compute

$$\alpha_t(i)=P(O_1O_2\ldots O_t \wedge q_t=S_i)$$

(How) can we cheaply compute

$$P(O_1O_2\ldots O_t) \quad ?$$

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

(How) can we cheaply compute

$$P(q_t=S_i|O_1O_2\ldots O_t)$$

$$\frac{\alpha_t(i)}{\sum_{j=1}^{N} \alpha_t(j)}$$

New question: Most probable **path** given observations

What is the most probable path given $O_1 O_2 ... O_T$, i.e.

What is $\underset{Q}{\text{argmax}}\ \mathrm{P}\left(Q \middle| O_1 O_2 ... O_T\right)$?

Slow answer:

$$\underset{Q}{\text{argmax}}\ \mathrm{P}\left(Q \middle| O_1 O_2 ... O_T\right)$$

$$= \underset{Q}{\text{argmax}}\ \frac{\mathrm{P}\left(O_1 O_2 ... O_T \middle| Q\right)\mathrm{P}(Q)}{\mathrm{P}\left(O_1 O_2 ... O_T\right)}$$

$$= \underset{Q}{\text{argmax}}\ \mathrm{P}\left(O_1 O_2 ... O_T \middle| Q\right)\mathrm{P}\left(Q\right)$$

Efficient MPP computation

We compute the following variables:

$$\delta_t(i) = \max_{q_1 q_2 .. q_{t-1}} P(q_1\ q_2\ ..\ q_{t-1} \wedge q_t = S_i \wedge O_1\ ..\ O_t)$$

= The probability of the PATH of length t-1 with the maximum chance of doing all these things:

…the path ITSELF OCCURING
and
…the path ENDING UP IN STATE S_i
and
…the path PRODUCING OUTPUT $O_1 \ldots O_t$

DEFINE: $mpp_t(i)$ = that most probable path

So: $\delta_t(i) = \text{Prob}(mpp_t(i))$

Andrew J. Viterbi[1]

Born	Andrea Giacomo Viterbi March 9, 1935 (age 83) Bergamo, Italy
Citizenship	Italian, American
Education	Massachusetts Institute of Technology (BS, MS) University of Southern California (PhD)

The Viterbi Algorithm

$$\delta_t(i) = \max_{q_1 q_2 \ldots q_{t-1}} \mathrm{P}\left(q_1 q_2 \ldots q_{t-1} \wedge q_t = S_i \wedge O_1 O_2 .. O_t\right)$$

$$mpp_t(i) = \arg\max_{q_1 q_2 \ldots q_{t-1}} \mathrm{P}\left(q_1 q_2 \ldots q_{t-1} \wedge q_t = S_i \wedge O_1 O_2 .. O_t\right)$$

$$\begin{aligned}
\delta_1(i) &= \max_{\text{one choice}} \mathrm{P}\left(q_1 = S_i \wedge O_1\right) \\
&= \mathrm{P}\left(q_1 = S_i\right)\mathrm{P}\left(O_1 \middle| q_1 = S_i\right) \\
&= \pi_i b_i\left(O_1\right)
\end{aligned}$$

The Viterbi Algorithm

$$\delta_t(i) = \max_{q_1 q_2 \ldots q_{t-1}} \mathrm{P}\left(q_1 q_2 \ldots q_{t-1} \wedge q_t = S_i \wedge O_1 O_2 .. O_t\right)$$

$$mpp_t(i) = \arg\max_{q_1 q_2 \ldots q_{t-1}} \mathrm{P}\left(q_1 q_2 \ldots q_{t-1} \wedge q_t = S_i \wedge O_1 O_2 .. O_t\right)$$

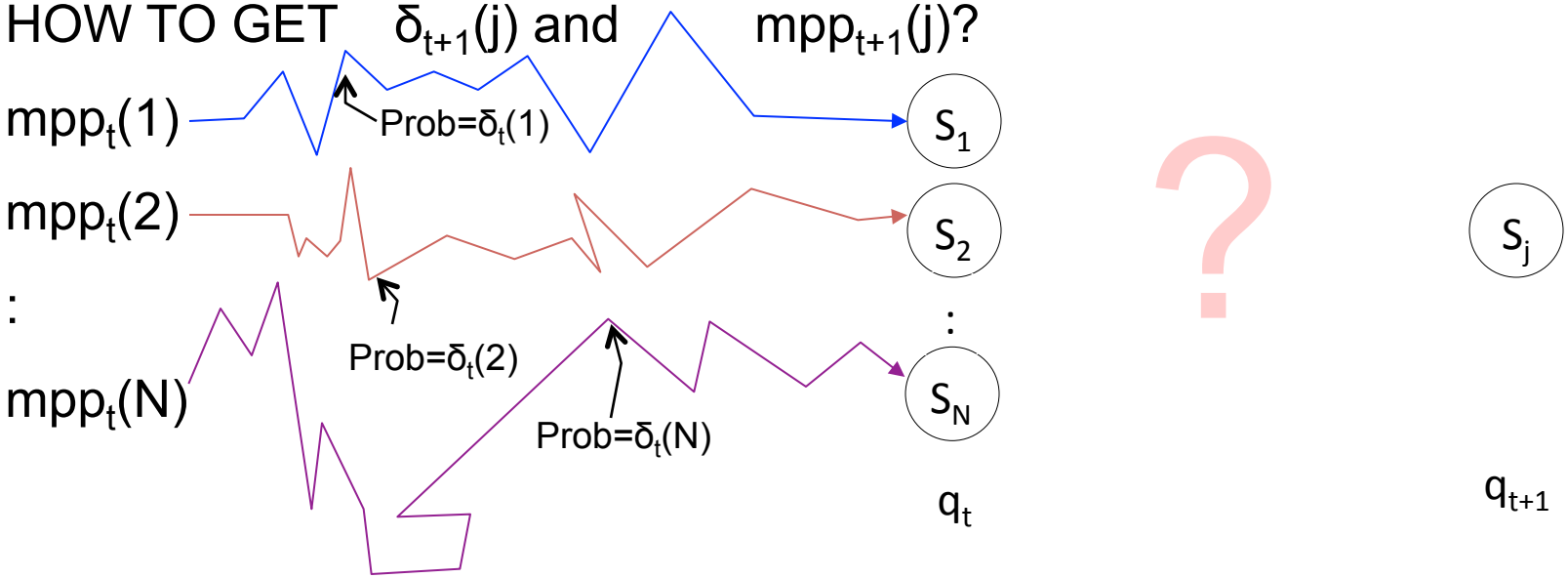
$$\delta_1(i) = \max_{\text{one choice}} \mathrm{P}\left(q_1 = S_i \wedge O_1\right)$$

$$= \mathrm{P}\left(q_1 = S_i\right)\mathrm{P}\left(O_1 \middle| q_1 = S_i\right)$$

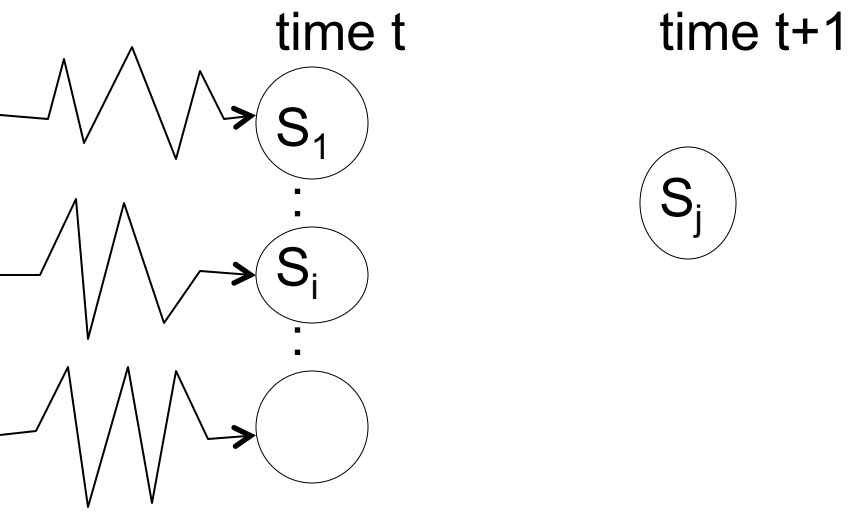
$$= \pi_i b_i\left(O_1\right)$$

If we have all the $\delta_t(i)$'s and $mpp_t(i)$'s for all i.

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



The Viterbi Algorithm

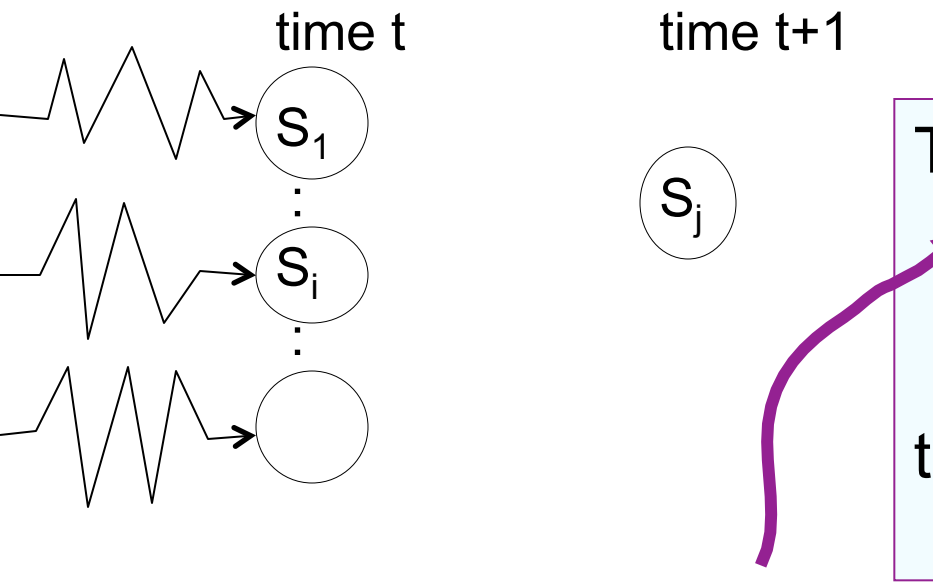


The most prob path with last two states S_i S_j

is

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

The Viterbi Algorithm



The most prob path with last two states S_i S_j

is

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

What is the prob of that path?

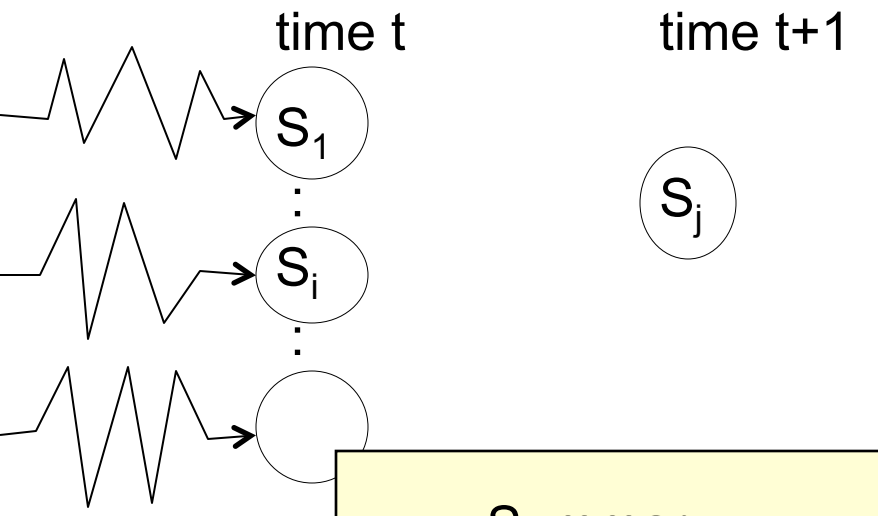
$$\delta_t(i) \times P(S_i \rightarrow S_j \wedge O_{t+1} \mid \lambda)$$

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

SO The most probable path to S_j has S_{i^*} as its penultimate state

where $i^* = \underset{i}{\operatorname{argmax}}\, \delta_t(i)\, a_{ij}\, b_j(O_{t+1})$

The Viterbi Algorithm



Summary:

$$\delta_{t+1}(j) = \delta_t(i^*)\, a_{ij}\, b_j\,(O_{t+1})$$

$$mpp_{t+1}(j) = mpp_{t+1}(i^*)S_{i^*}$$

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

What is Viterbi Algorithm used for?

Classic Example

Speech recognition:

Signal → words

HMM → observable is signal

→ Hidden state is part of word formation

What is the most probable word given this signal?

UTTERLY GROSS SIMPLIFICATION

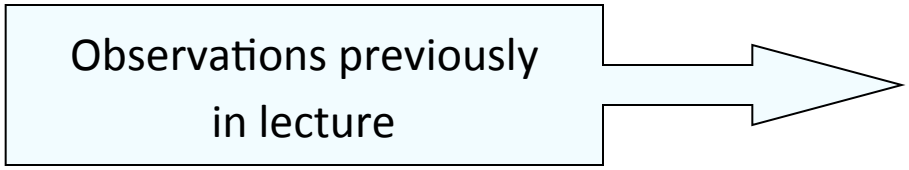
In practice: many levels of inference; not one big jump.

HMMs are used and useful

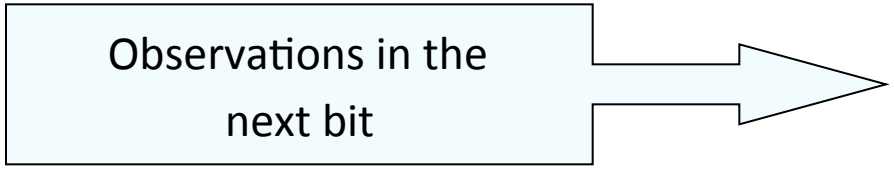
But how do you design an HMM?

Occasionally, (e.g. in our robot example) it is reasonable to deduce the HMM from first principles.

But usually, especially in Speech or Genetics, it is better to infer it from large amounts of data. $O_1\ O_2\ ..\ O_T$ with a big "T".



$O_1\ O_2\ ..\ O_T$



$O_1\ O_2\ ..\ O_{\mathbf{T}}$

Inferring an HMM

Remember, we have been doing things like

$$P(O_1\ O_2\ ..\ O_T \mid \lambda)$$

That "λ" is the notation for our HMM parameters.

Now We have some observations and we want to estimate λ from them.

AS USUAL: We could use

(i) MAX LIKELIHOOD $\lambda = \underset{\lambda'}{\text{argmax}}\ P(O_1\ ..\ O_T \mid \lambda')$

(ii) BAYES Work out $P(\lambda \mid O_1\ ..\ O_T)$

and then take $E[\lambda]$ or $\max_{\lambda} P(\lambda \mid O_1\ ..\ O_T)$

Max likelihood HMM estimation

Define

$\gamma_t(i) = P(q_t = S_i \mid O_1 O_2 \ldots O_T , \lambda)$

$\varepsilon_t(i,j) = P(q_t = S_i \wedge q_{t+1} = S_j \mid O_1 O_2 \ldots O_T , \lambda)$

$\gamma_t(i)$ and $\varepsilon_t(i,j)$ can be computed efficiently $\forall i,j,t$

(Details in Rabiner paper)

$$\sum_{t=1}^{T-1} \gamma_t(i) =$$ Expected number of transitions out of state i during the path

$$\sum_{t=1}^{T-1} \varepsilon_t(i,j) =$$ Expected number of transitions from state i to state j during the path

$$\gamma_t(i) = \mathrm{P}\left(q_t = S_i \middle| O_1 O_2 .. O_T, \lambda\right)$$

$$\varepsilon_t(i,j) = \mathrm{P}\left(q_t = S_i \wedge q_{t+1} = S_j \middle| O_1 O_2 .. O_T, \lambda\right)$$

$$\sum_{t=1}^{T-1} \gamma_t(i) = \text{expected number of transitions out of state i during path}$$

$$\sum_{t=1}^{T-1} \varepsilon_t(i,j) = \text{expected number of transitions out of i and into j during path}$$

HMM estimation

$$\text{Notice } \frac{\sum_{t=1}^{T-1} \varepsilon_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)} = \frac{\begin{pmatrix}\text{expected frequency} \\ \text{i} \to \text{j}\end{pmatrix}}{\begin{pmatrix}\text{expected frequency} \\ \text{i}\end{pmatrix}}$$

$$= \text{Estimate of } \mathrm{Prob}\left(\text{Next state } \mathrm{S_j} \middle| \text{This state } \mathrm{S_i}\right)$$

We can re-estimate

$$\mathrm{a_{ij}} \leftarrow \frac{\sum \varepsilon_t(i,j)}{\sum \gamma_t(i)}$$

We can also re-estimate

$$\mathrm{b_j}(O_k) \leftarrow \cdots \qquad \text{(See Rabiner)}$$

We want a_{ij}^{new} = new estimate of $P(q_{t+1} = s_j \mid q_t = s_i)$

We want a_{ij}^{new} = new estimate of $P(q_{t+1} = s_j \mid q_t = s_i)$

$$= \frac{\text{Expected \# transitions } i \rightarrow j \mid \lambda^{old}, O_1, O_2, \cdots O_T}{\sum_{k=1}^{N} \text{Expected \# transitions } i \rightarrow k \mid \lambda^{old}, O_1, O_2, \cdots O_T}$$

We want a_{ij}^{new} = new estimate of $P(q_{t+1} = s_j \mid q_t = s_i)$

$$= \frac{\text{Expected \# transitions } i \to j \mid \lambda^{old}, O_1, O_2, \cdots O_T}{\sum_{k=1}^{N} \text{Expected \# transitions } i \to k \mid \lambda^{old}, O_1, O_2, \cdots O_T}$$

$$= \frac{\sum_{t=1}^{T} P(q_{t+1} = s_j, q_t = s_i \mid \lambda^{\text{old}}, O_1, O_2, \cdots O_T)}{\sum_{k=1}^{N} \sum_{t=1}^{T} P(q_{t+1} = s_k, q_t = s_i \mid \lambda^{\text{old}}, O_1, O_2, \cdots O_T)}$$

We want a_{ij}^{new} = new estimate of $P(q_{t+1} = s_j \mid q_t = s_i)$

$$= \frac{\text{Expected \# transitions } i \to j \mid \lambda^{old}, O_1, O_2, \cdots O_T}{\sum_{k=1}^{N} \text{Expected \# transitions } i \to k \mid \lambda^{old}, O_1, O_2, \cdots O_T}$$

$$= \frac{\sum_{t=1}^{T} P(q_{t+1} = s_j, q_t = s_i \mid \lambda^{\text{old}}, O_1, O_2, \cdots O_T)}{\sum_{k=1}^{N} \sum_{t=1}^{T} P(q_{t+1} = s_k, q_t = s_i \mid \lambda^{\text{old}}, O_1, O_2, \cdots O_T)}$$

$$= \frac{S_{ij}}{\sum_{k=1}^{N} S_{ik}} \text{ where } S_{ij} = \sum_{t=1}^{T} P(q_{t+1} = s_j, q_t = s_i, O_1, \cdots O_T \mid \lambda^{\text{old}})$$

$$= \text{What?}$$

We want a_{ij}^{new} = new estimate of $P(q_{t+1} = s_j \mid q_t = s_i)$

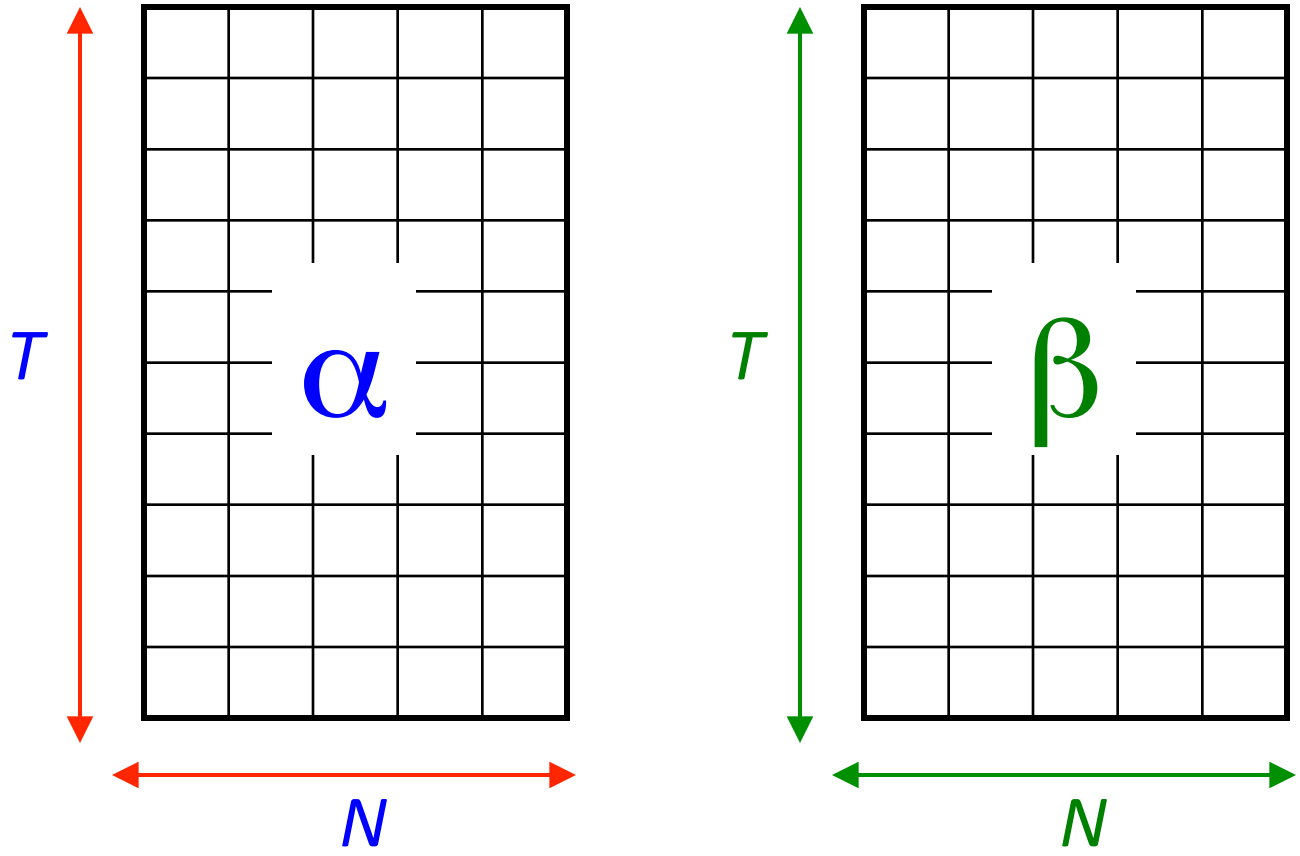
$$= \frac{\text{Expected \# transitions } i \to j \mid \lambda^{old}, O_1, O_2, \cdots O_T}{\sum_{k=1}^{N} \text{Expected \# transitions } i \to k \mid \lambda^{old}, O_1, O_2, \cdots O_T}$$

$$= \frac{\sum_{t=1}^{T} P(q_{t+1} = s_j, q_t = s_i \mid \lambda^{\text{old}}, O_1, O_2, \cdots O_T)}{\sum_{k=1}^{N} \sum_{t=1}^{T} P(q_{t+1} = s_k, q_t = s_i \mid \lambda^{\text{old}}, O_1, O_2, \cdots O_T)}$$

$$= \frac{S_{ij}}{\sum_{k=1}^{N} S_{ik}} \text{ where } S_{ij} = \sum_{t=1}^{T} P(q_{t+1} = s_j, q_t = s_i, O_1, \cdots O_T \mid \lambda^{\text{old}})$$

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

We want $a_{ij}^{\text{new}} = S_{ij} \Big/ \sum_{k=1}^{N} S_{ik}$ where $S_{ij} = a_{ij} \sum_{t=1}^{T} \alpha_t(i)\beta_{t+1}(j)b_j(O_{t+1})$



EM for HMMs

If we knew λ we could estimate EXPECTATIONS of quantities such as

Expected number of times in state i

Expected number of transitions i $\rightarrow$ j

If we knew the quantities such as

Expected number of times in state i

Expected number of transitions i $\rightarrow$ j

We could compute the MAX LIKELIHOOD estimate of

$$\lambda = \langle \{a_{ij}\}, \{b_i(j)\}, \pi_i \rangle$$

Roll on the EM Algorithm…

EM for HMMs

1. Get your observations $O_1 \ldots O_T$
2. Guess your first λ estimate $\lambda(0)$, k=0
3. k = k+1
4. Given $O_1 \ldots O_T$, $\lambda(k)$ compute
   $\gamma_t(i)$, $\varepsilon_t(i,j)$ $\forall 1 \le t \le T$, $\forall 1 \le i \le N$, $\forall 1 \le j \le N$
5. Compute expected freq. of state i, and expected freq. i→j
6. Compute new estimates of a_{ij}, $b_j(k)$, π_i accordingly. Call them $\lambda(k+1)$
7. Goto 3, unless converged.

- **Also known (for the HMM case) as the BAUM-WELCH algorithm.**

Baum-Welch

- **Leonard Esau Baum** (August 23, 1931 – August 14, 2017) was an American mathematician. He graduated from Harvard University in 1953, and earned a Ph.D. in mathematics from Harvard in 1958, with a dissertation entitled *Derivations in Commutative Semi-Simple Banach Algebras*. At the time he wrote his work with Welch, he was working for the Institute for Defense Analyses in Princeton, New Jersey.
- **Lloyd Richard Welch** (born September 28, 1927) is an American information theorist and applied mathematician, and co-inventor of the Baum–Welch algorithm and the Berlekamp–Welch algorithm, also known as the Welch–Berlekamp algorithm. Welch received his B.S. in mathematics from the University of Illinois, 1951, and Ph.D. in mathematics from the California Institute of Technology, 1958.

Bad News

- There are lots of local minima

Good News

- The local minima are usually adequate models of the data.

Notice

- EM does not estimate the number of states. That must be given.
- Often, HMMs are forced to have some links with zero probability. This is done by setting $a_{ij}=0$ in initial estimate $\lambda(0)$
- Easy extension of everything seen today: HMMs with real valued outputs

Basic Operations in HMMs

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

Problem	Algorithm	Complexity
Evaluation: Calculating $P(q_t=S_i \mid O_1 O_2 \ldots O_t)$	Forward-Backward	$O(TN^2)$
Inference: Computing $Q^* = \mathit{argmax}_Q P(Q\|O)$	Viterbi Decoding	$O(TN^2)$
Learning: Computing $\lambda^* = \mathrm{arg}\mathit{max}_\lambda P(O\|\lambda)$	Baum-Welch (EM)	$O(TN^2)$

T = # timesteps, N = # states

What You Should Know

- What is an HMM ?
- Computing (and defining) $\alpha_t(i)$
- The Viterbi algorithm
- Outline of the EM algorithm
- Good: fairly thorough reading of Rabiner* from page 257 to page 266* [Up to but not including "IV. Types of HMMs"].

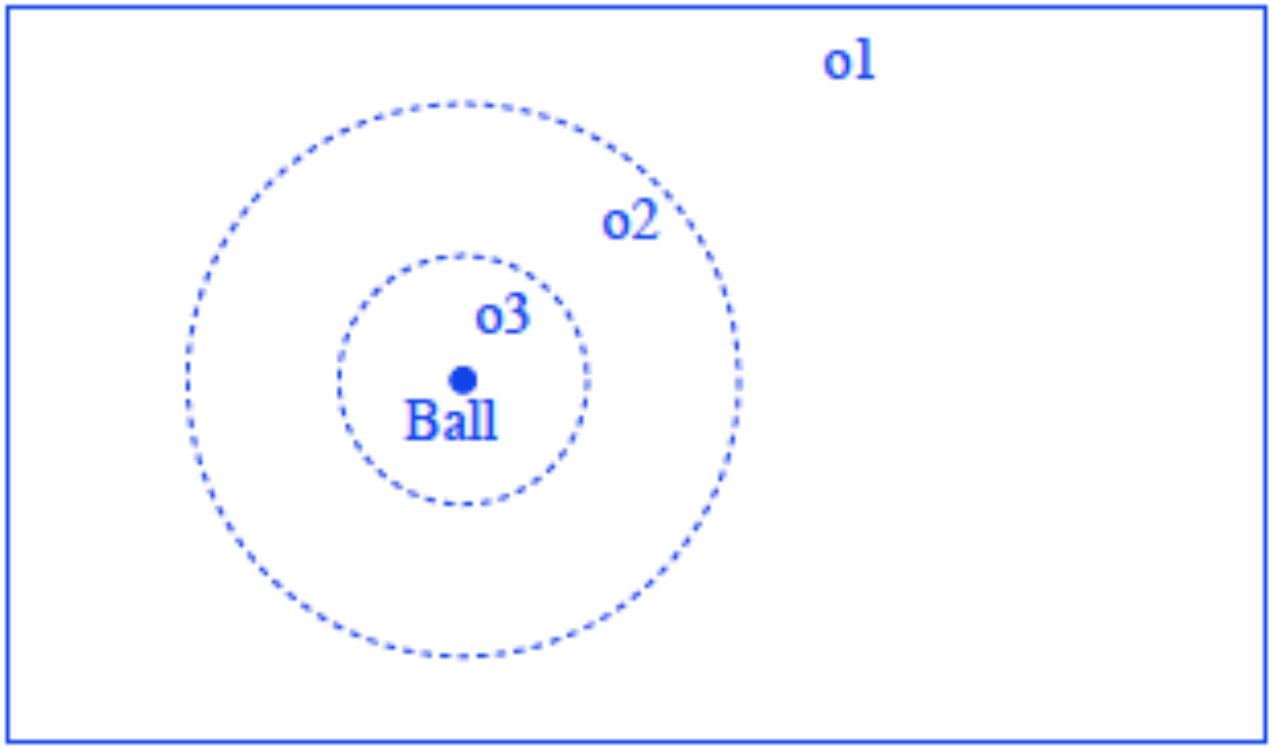
*L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," Proc. of the IEEE, Vol.77, No.2, pp. 257--286, 1989.

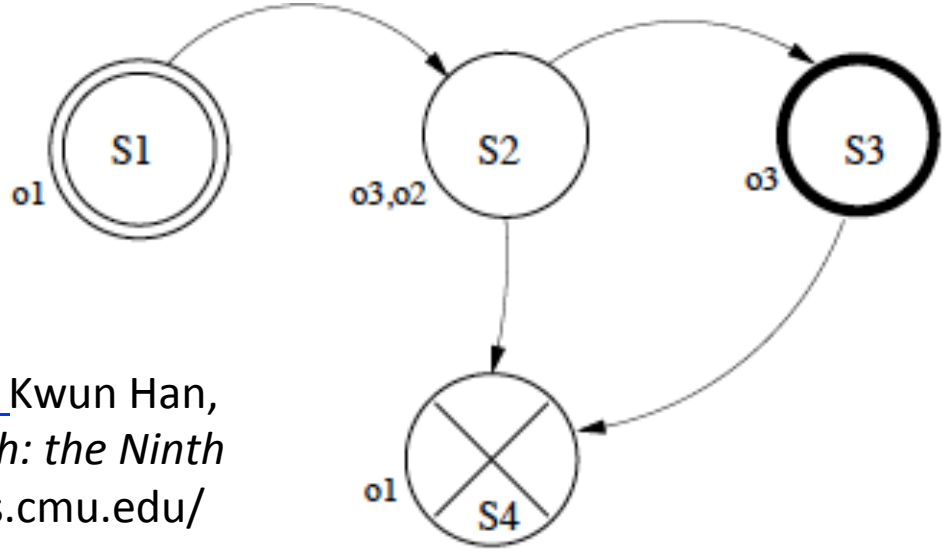
http://ieeexplore.ieee.org/iel5/5/698/00018626.pdf?arnumber=18626

Applications of **HMMs**

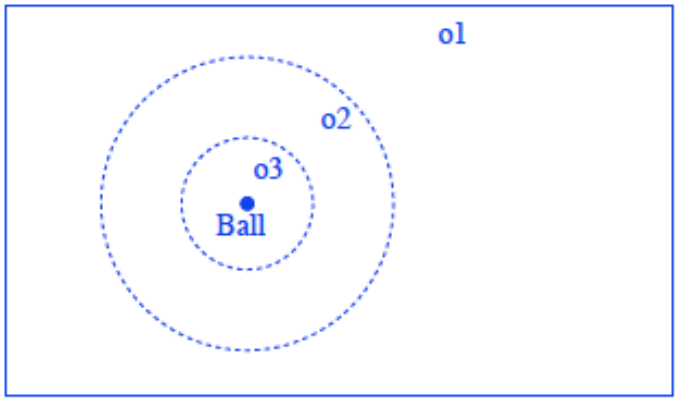
Speech Recognition

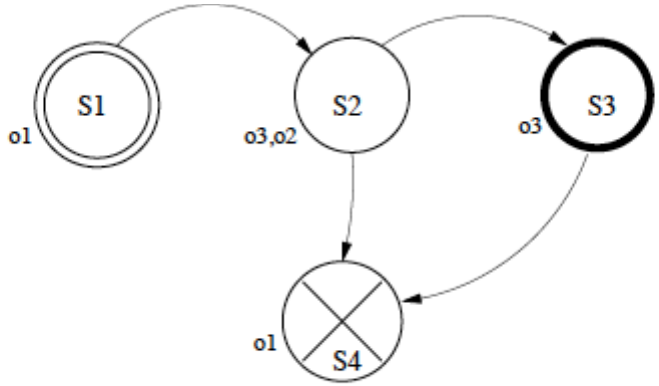
- Observations: features from speech segment
- States: word being spoken
- State Transition Model: Language Model on word sequences
- Observation Model: Acoustic Model on speech sounds for specific words

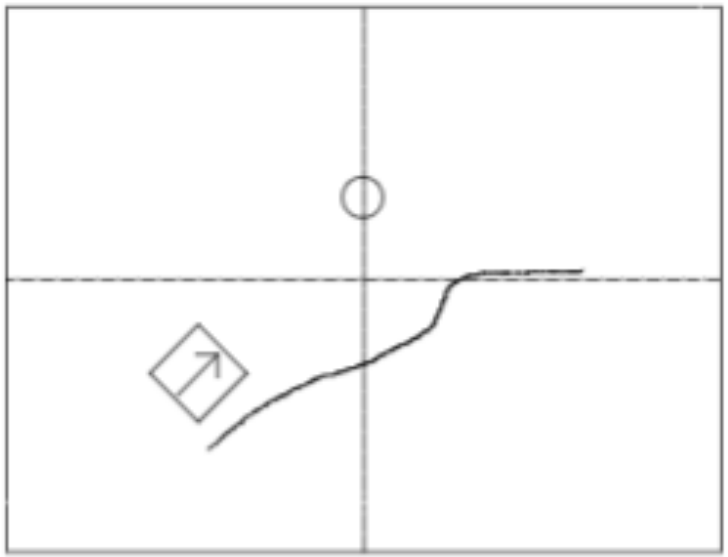


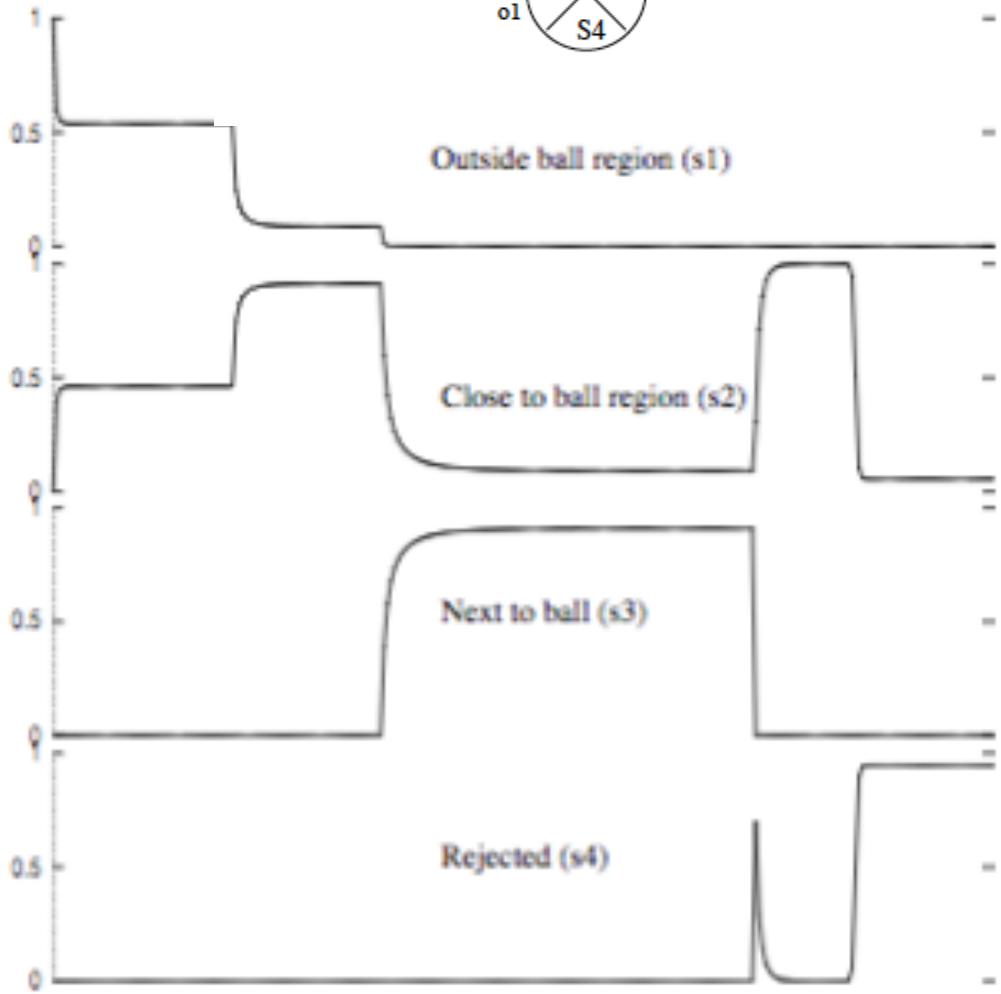


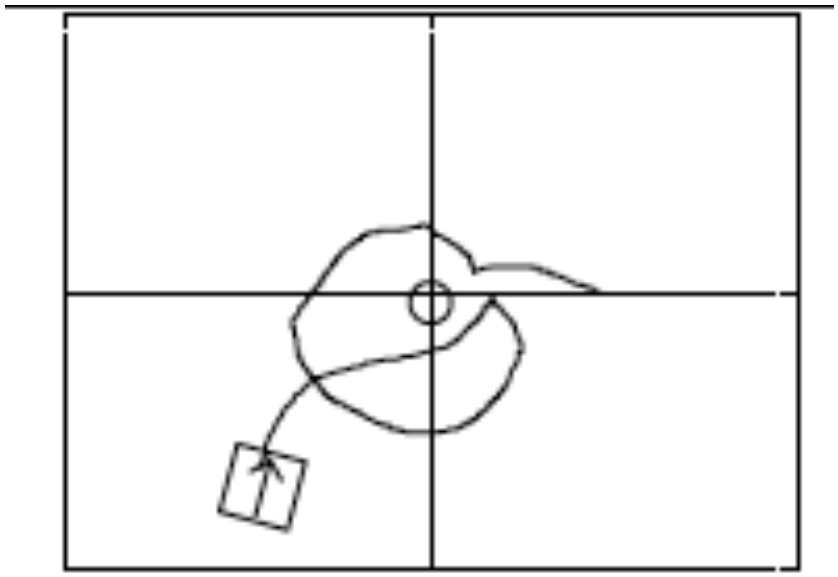
Automated Robot Behavior Recognition, Kwun Han, and Manuela Veloso In *Robotics Research: the Ninth International Symposium*, http://www.cs.cmu.edu/~mmv/papers/ijcai99-kwun.pdf

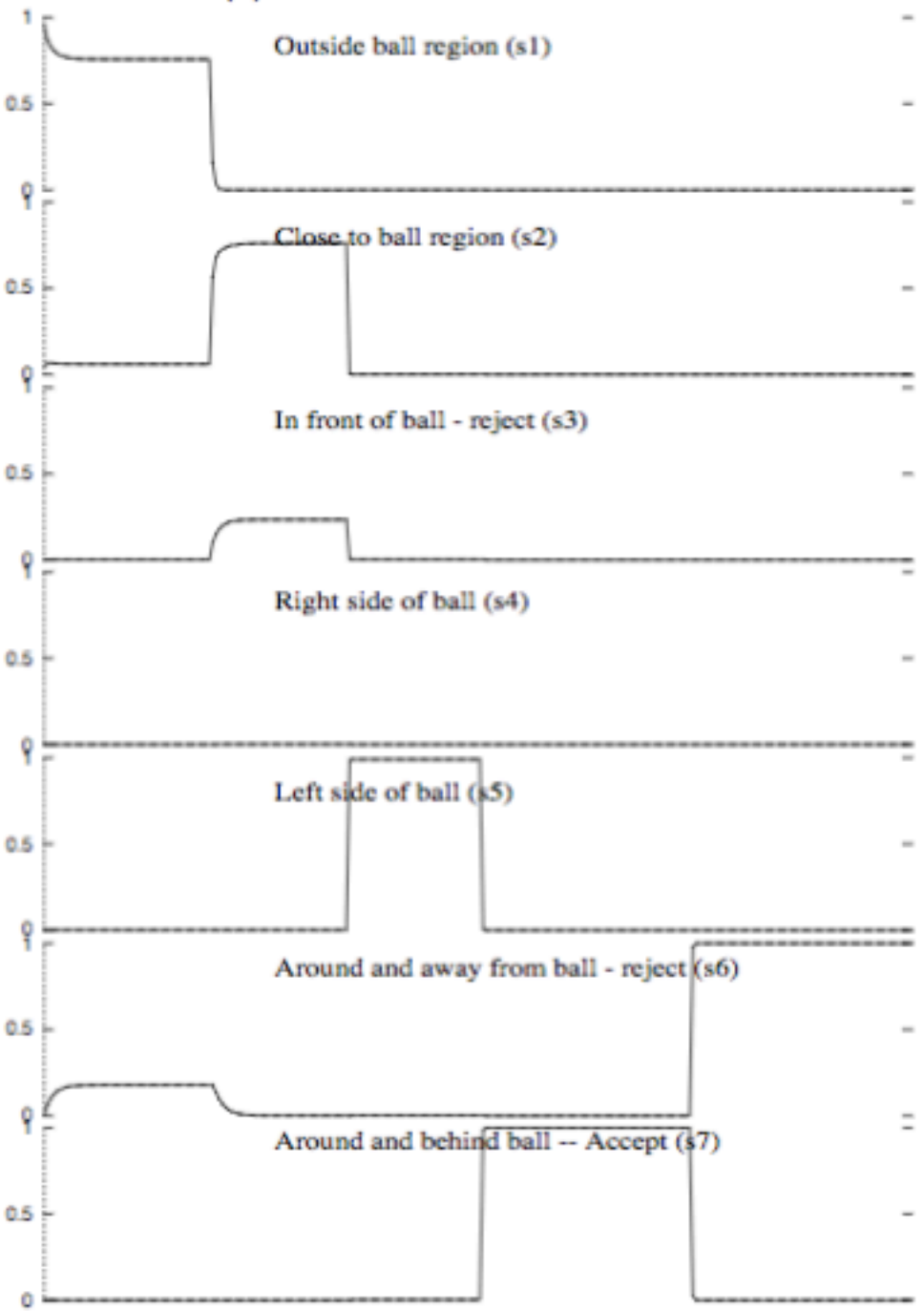

o1
o2
o3
Ball


S1
o1
S2
o3,o2
S3
o3
S4
o1




1
0.5
0
Outside ball region (s1)
Close to ball region (s2)
Next to ball (s3)
Rejected (s4)

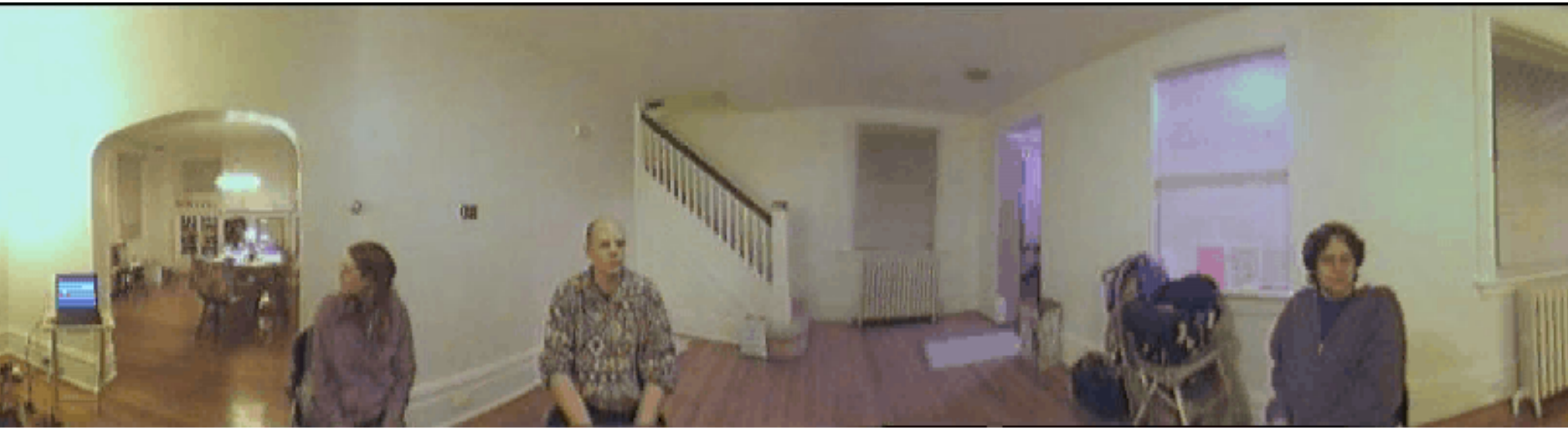




(a) Go-Behind-Ball Behavior

Activity Recognition from Videos

- Observations: image features: position displacement
- States: Type of activity: stand, standing, walk left, walk right, sit, sitting



FOCUS: A Generalized Method for Object Discovery for Robots that Observe and Interact with Humans, Manuela Veloso, Paul Rybski, and Felix von Hundelshausen
In HRI 2006, http://www.cs.cmu.edu/~mmv/papers/06hri-focus.pdf

Part of Speech Tagging

- Observations: Words
- States: Part of Speech (noun, verb, adjective, etc.)
- State Transition Model: Grammar Model on POS sequences
- Observation Model: Language Model on Word | POS