Hidden Markov Model (II)

©Dr. Min Chi
mchi@ncsu.edu

AI Academy

Puzzles Regarding the Dishonest Casino

GIVEN: A sequence of rolls by the casino player

1245526462146146136136661664661636616366163616515615115146123562344

- Question 1: State Estimation

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

  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

Most probable path given observations

What's most probable path given $O_1 O_2 \ldots O_T$, i.e.

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

Slow, stupid answer:

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

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

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

Efficient Solution

We're going to 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 with the maximum chance of doing all these things:

...OCCURING

and

...ENDING UP IN STATE S_i

and

...PRODUCING OUTPUT $O_1 ... O_t$

DEFINE: $mpp_t(i)$ = that path

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

The Viterbi Algorithm

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

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

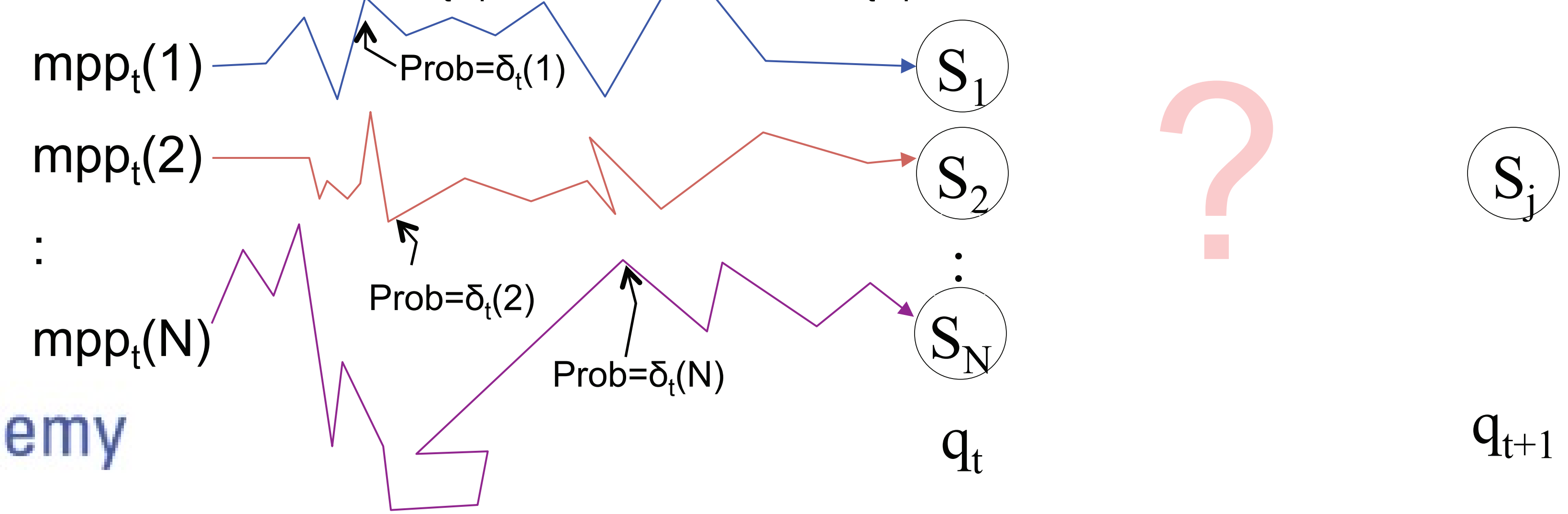
$$\delta_1(i) = \max \ \mathrm{P}\left(q_1 = S_i \wedge O_1\right)$$

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

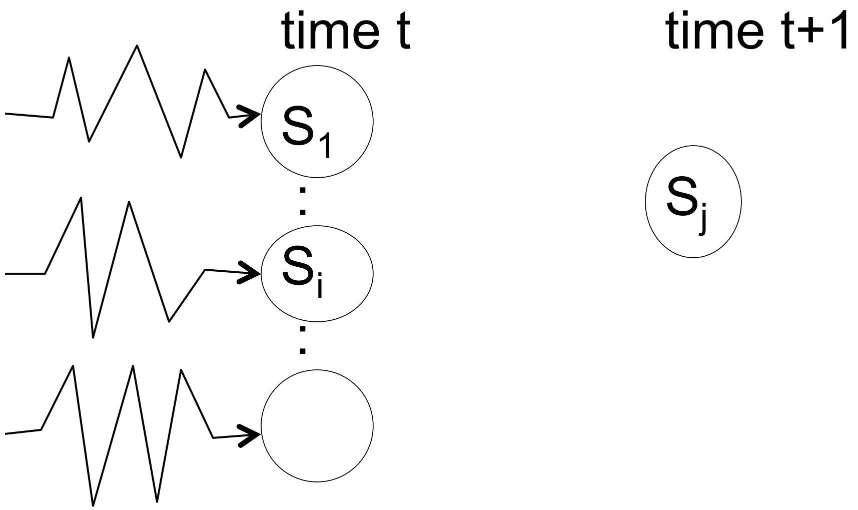
$$= \pi_i b_i\left(O_1\right)$$

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 $mpp_{t+1}(j)$?



The Viterbi Algorithm

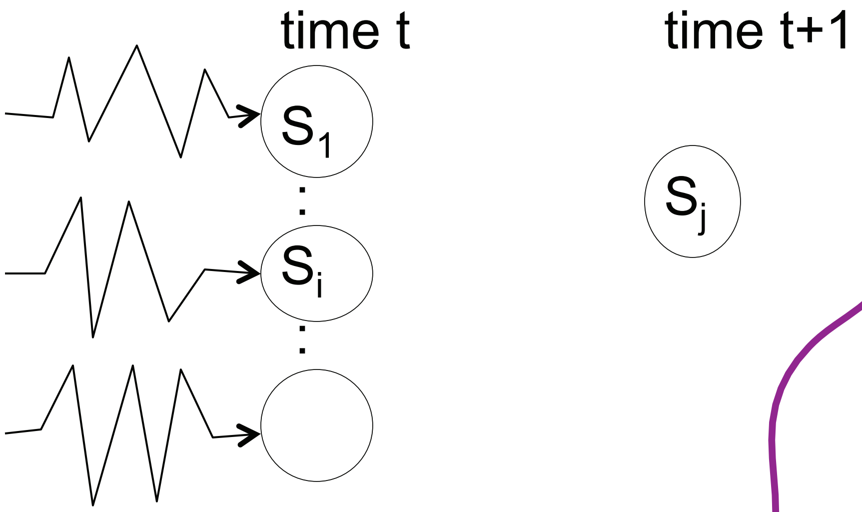


The most prob path with last two states S_i S_j

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?

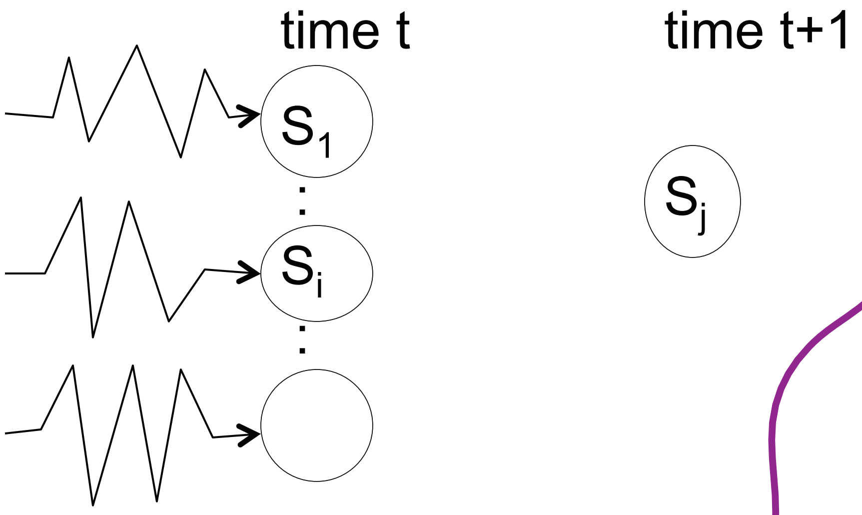
$$\delta_t(i) \times P(S_i \rightarrow S_j \wedge O_{t+1} \mid \lambda)$$

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

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



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?

$\delta_t(i) \times P(S_i \rightarrow S_j \wedge O$ ... $| \lambda)$

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

SO The most probable ...

S_{i^*} as its penultimate state

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

Summary:

$\left.\begin{array}{l} \delta_{t+1}(j) \;=\; \delta_t(i^*)\, a_{ij}\, b_j\, (O_{t+1}) \\ mpp_{t+1}(j) \;=\; mpp_{t+1}(i^*) S_{i^*} \end{array}\right\}$ with i* defined to the left

What's Viterbi 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.

Puzzles Regarding the Dishonest Casino

GIVEN: A sequence of rolls by the casino player

1245526462146146136136661664661636616366163616515615115 146123562344

- 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

Learning HMMs

- Until now we assumed that the emission and transition probabilities are known
- This is usually not the case
  -

While we will discuss learning the transition and emission models, we will not discuss selecting the states.

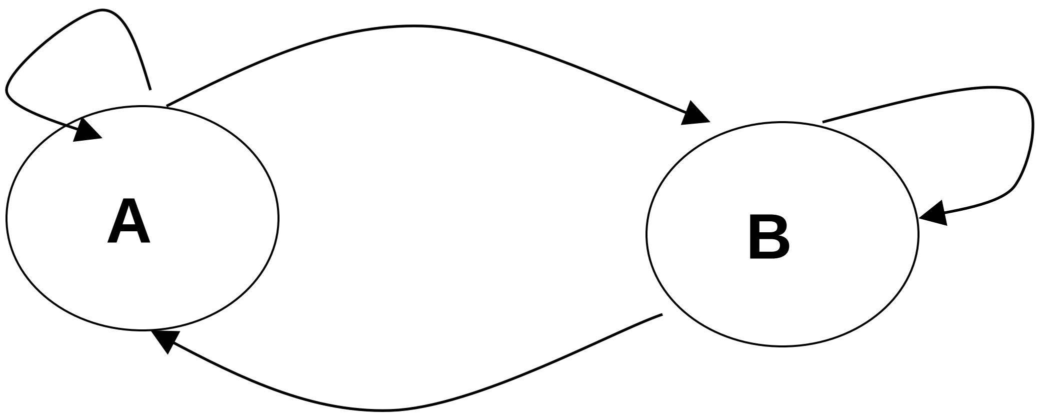
This is usually a function of domain knowledge.

Example

- Assume the model below
- We also observe the following sequence:

1,2,2,5,6,5,1,2,3,3,5,3,3,2 …..

- How can we determine the initial, transition and emission probabilities?



Initial probabilities

Q: assume we can observe the following sets of states:

AAABBAA
AABBBBB
BAABBAB

how can we learn the initial probabilities?

A: Maximum likelihood estimation

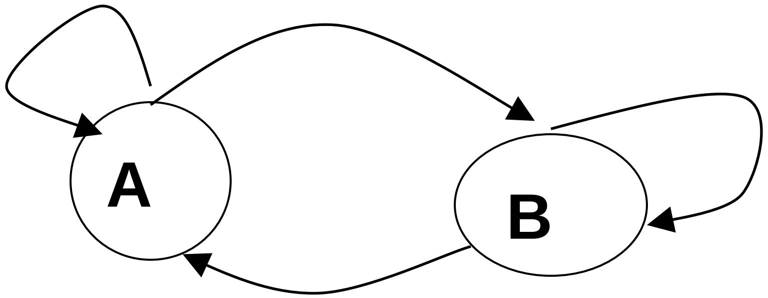
Find the initial probabilities π such that

k is the number of sequences avialable for training

$$\pi^* = \arg\max_\pi \prod_k \pi(q_1) \prod_{t=2}^{T} p(q_t \mid q_{t-1}) \Rightarrow$$

$$\pi^* = \arg\max_\pi \prod_k \pi(q_1)$$

$$\pi_A = \#A / (\#A + \#B)$$



AI Academy

NC STATE

Transition probabilities

Q: assume we can observe the set of states:

AAABBAAAABBBBBAAAABBBB

how can we learn the transition probabilities? A:

Maximum likelihood estimation

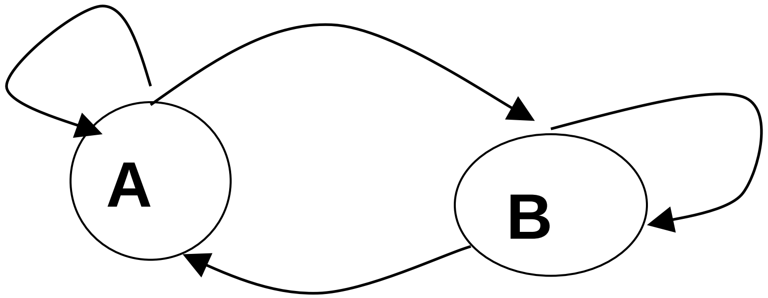
Find a transition matrix *a* such that

remember that we defined $a_{i,j}=p(q_t=s_j|q_{t-1}=s_i)$

$$a^* = \operatorname{argmax}_a \prod_k \pi(q_1) \prod_{t=2}^{T} p(q_t \mid q_{t-1}) \Rightarrow$$

$$a^* = \operatorname{argmax}_a \prod_{t=2}^{T} p(q_t \mid q_{t-1})$$

$$a_{A,B} = \#AB \,/\, (\#AB + \#AA)$$



Emission probabilities

Q: assume we can observe the set of states:

A A A B B A A A A B B B B B A A

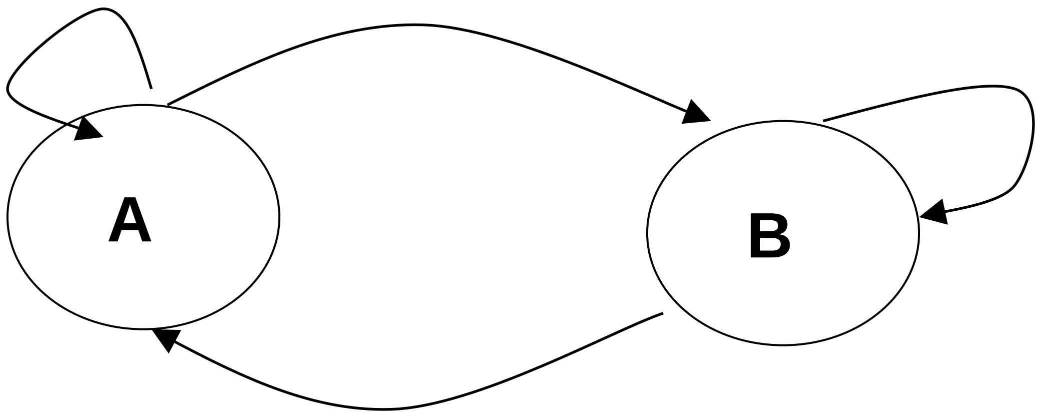
and the set of dice values

1 2 3 5 6 3 2 1 1 3 4 5 6 5 2 3

how can we learn the emission probabilities? A: 5

Maximum likelihood estimation

$$b_A(5) = \#A5 \,/\, (\#A1 + \#A2 + \ldots + \#A6)$$



AI Academy

NC STATE

Learning HMMs

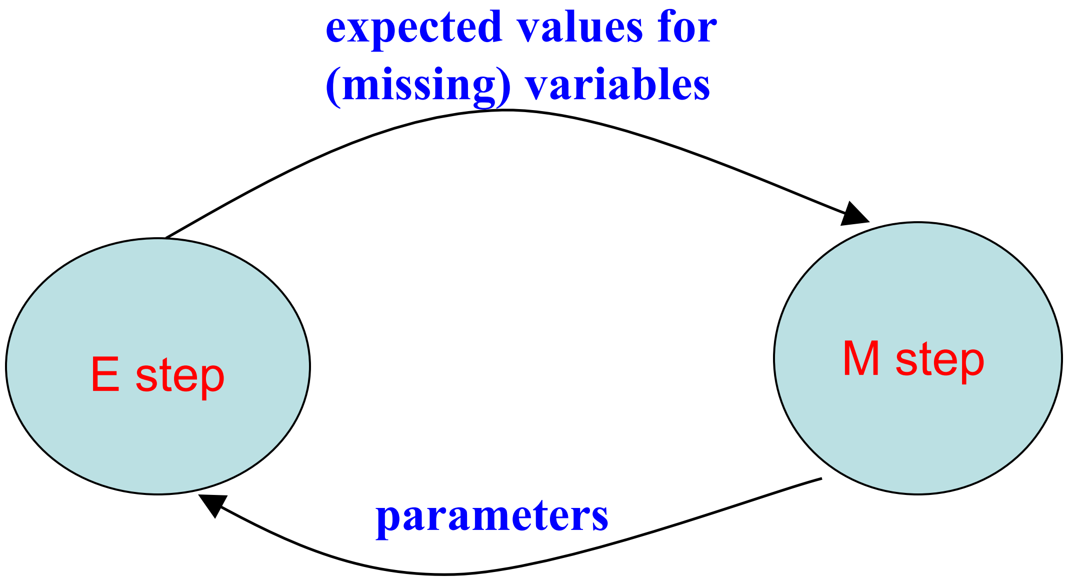
- In most case we do not know what states generated each of the outputs (fully unsupervised)
- … but had we known, it would be very easy to determine an emission and transition model!
- On the other hand, if we had such a model we could determine the set of states using the inference methods we discussed

Expectation Maximization (EM)

- Appropriate for problems with ‘missing values’ for the variables.
- For example, in HMMs we usually do not observe the states

Expectation Maximization (EM): Quick reminder

- Two steps
  - E step: Fill in the expected values for the missing variables
  - M step: Regular maximum likelihood estimation (MLE) using the values computed in the E step and the values of the other variables
- Guaranteed to converge (though only to a local minima).



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…

Max Likelihood HMM Estimation

Define

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

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

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

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

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

Forward-Backward

- We already defined a *forward* looking variable

$$\alpha_t(i) = P(O_1 \ldots O_t \wedge q_t = s_i)$$

- We also need to define a *backward* looking variable

$$\beta_t(i) = P(O_{t+1}, \cdots O_T | q_t = s_i)$$

Forward-Backward

- We already defined a *forward* looking variable

$$\alpha_t(i) = P(O_1 \ldots O_t \wedge q_t = s_i)$$

- We also need to define a *backward* looking variable

$$\beta_t(i) = P(O_{t+1}, \cdots, O_T \mid q_t = s_i) =$$
$$\sum_j a_{i,j} b_j(O_{t+1}) \beta_{t+1}(j)$$

Forward-Backward

- We already defined a *forward* looking variable

$$\alpha_t(i) = P(O_1 \ldots O_t \wedge q_t = s_i)$$

- We also need to define a *backward* looking variable

$$\beta_t(i) = P(O_{t+1}, \cdots, O_T \mid q_t = s_i)$$

- Using these two definitions we can show

P(A|B)=P(A,B)/P(B)

$$P(q_t = s_i \mid O_1, \cdots, O_T) = \frac{\alpha_t(i)\beta_t(i)}{\sum_j \alpha_t(j)\beta_t(j)} \stackrel{def}{=} S_t(i)$$

State and transition probabilities

- Probability of a state

$$P(q_t = s_i \mid O_1, \cdots, O_T) = \frac{\alpha_t(i)\beta_t(i)}{\sum_j \alpha_t(j)\beta_t(j)} \overset{def}{=} S_t(i)$$

- We can also derive a transition probability

$$P(q_t = s_i, q_{t+1} = s_j \mid o_1, \cdots, o_T) = S_t(i, j)$$

$$P(q_t = s_i, q_{t+1} = s_j \mid o_1, \cdots, o_T) =$$
$$= \frac{\alpha_t(i) P(q_{t+1} = s_j \mid q_t = s_i) P(o_{t+1} \mid q_{t+1} = s_j) \beta_{t+1}(j)}{\sum_j \alpha_t(j)\beta_t(j)} \overset{def}{=} S_t(i, j)$$

E step

- Compute $S_t(i)$ and $S_t(i,j)$ for all t, i, and j ($1 \le t \le T$, $1 \le i \le N$, $1 \le j \le N$)

$$P(q_t = s_i \mid O_1, \cdots, O_T) = S_t(i)$$

$$P(q_t = s_i, q_{t+1} = s_j \mid o_1, \cdots, o_T) = S_t(i, j)$$

M step (1)

Compute transition probabilities:

$$a_{i,j} = \frac{\hat{n}(i,j)}{\sum_{k} \hat{n}(i,k)}$$

where

$$\hat{n}(i,j) = \sum_{t} S_t(i,j)$$

M step (2)

Compute emission probabilities (here we assume a multinomial distribution):

define:

$$B_k(j) = \sum_{t|o_t=j} S_t(k)$$

then

$$b_k(j) = \frac{B_k(j)}{\sum_i B_k(i)}$$

AI Academy

NC STATE

Complete EM algorithm for learning the parameters of HMMs (Baum-Welch)

- Inputs: 1 .Observations $O_1 \ldots O_T$

  2. Number of states, model

1. Guess initial transition and emission parameters
2. Compute E step: $S_t(i)$ and $S_t(i,j)$
3. Compute M step
4. Convergence? No
5. Output complete model

We did not discuss initial probability estimation. These can be deduced from multiple sets of observation (for example, several recorded customers for speech processing)

HMM

- 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

Stay Connected

Dr. Min Chi

Associate Professor
mchi@ncsu.edu
(919) 515-7825

AI Academy

AI Academy

go.ncsu.edu/aiacademy

NC STATE UNIVERSITY