UVA CS 6316
– Fall 2015 Graduate:
Machine Learning

Lecture 26:
Hidden Markov models (HMMs)

Dr. Yanjun Qi

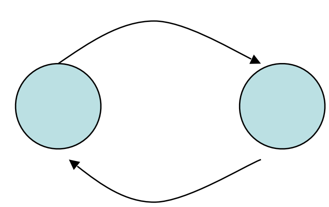
University of Virginia

Department of
Computer Science

What's wrong with Bayesian networks

- Bayesian networks are very useful for modeling joint distributions
- But they have their limitations:
  - Cannot account for temporal / sequence models
  - DAG's (no self or any other loops)

This is not a valid Bayesian network!



Hidden Markov models

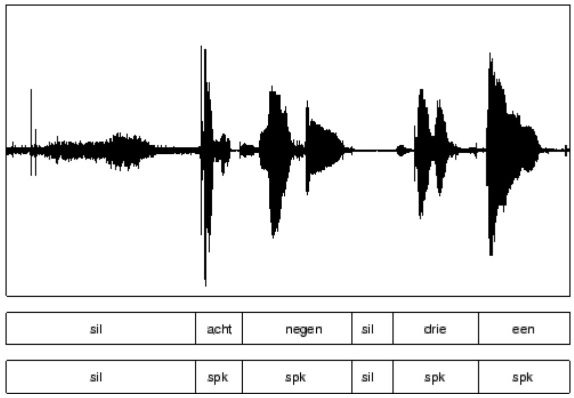
- Model a set of observation with a set of hidden states
  - Robot movement
    Observations: range sensor, visual sensor
    Hidden states: location (on a map)
  - Speech processing
    Observations: sound signals
    Hidden states: parts of speech, words
  - Biology
    Observations: DNA base pairs
    Hidden states: Genes

Hidden Markov models

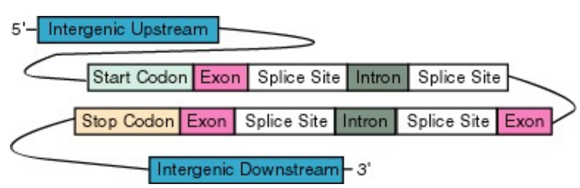
- Model a set of observation with a set of hidden states
  - Robot movement
    Observations: range sensor, visual sensor
    Hidden states: location (on a map)

1. Hidden states generate observations
2. Hidden states transition to other hidden states

Examples: Speech processing



Example: Biological data

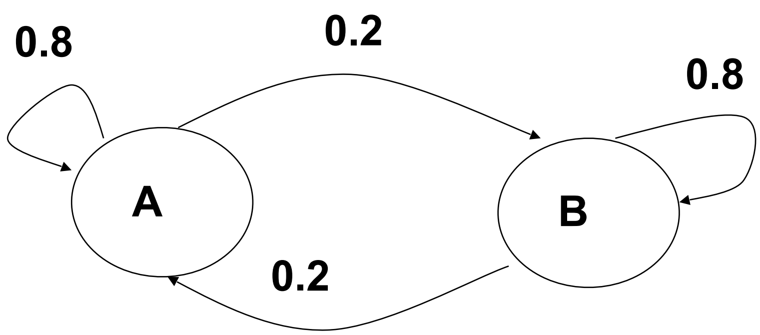


ATGAAGCTACTGTCTTCTATCGAACAAGCATGCGA
TATTTGCCGACTTAAAAAGCTCAAG
TGCTCCAAAGAAAAACCGAAGTGCGCCAAGTGTC
TGAAGAACAACTGGGAGTGTCGCTAC
TCTCCCAAAACCAAAAGGTCTCCGCTGACTAGGG
CACATCTGACAGAAGTGGAATCAAGG
CTAGAAAGACTGGAACAGCTATTTCTACTGATTTTT
CCTCGAGAAGACCTTGACATGATT

Contents

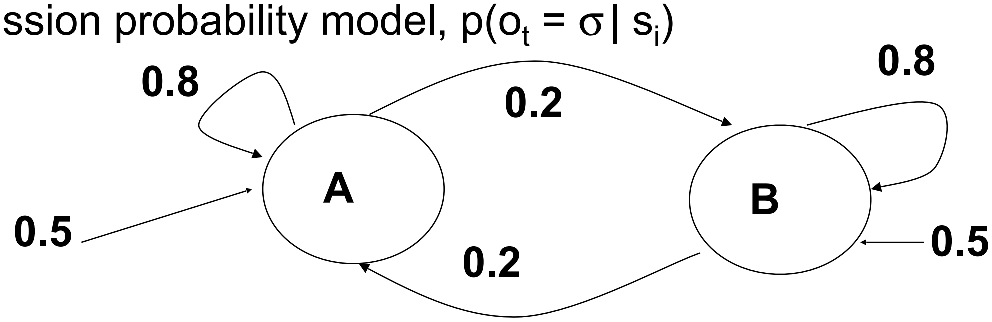
Example: Gambling on dice outcome

- Two dices, both skewed (output model).
- Can either stay with the same dice or switch to the second dice (transition mode).



A Hidden Markov model

- A set of states $\{s_1 \ldots s_n\}$
  - In each time point we are in exactly one of these states denoted by q_t
- Π_i, the probability that we *start* at state s_i
- A transition probability model, $P(q_t = s_i \mid q_{t-1} = s_j)$
- A set of possible outputs Σ
  - At time t we emit a symbol $\sigma \in \Sigma$
- An emission probability model, $p(o_t = \sigma \mid s_i)$



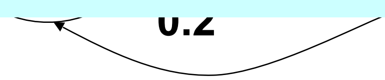
The Markov property

- A set of states $\{s_1 \dots s_n\}$

  - In each time point we are in exactly one of these states denoted by q_t
- Π_i, the probability that we start at state s_i
- A transition probability model, $P(q_t = s_i \mid q_{t-1} = s_j)$

An important aspect of this definitions is the Markov property: q_{t+1} is conditionally independent of q_{t-1} (and any earlier time points) given q_t

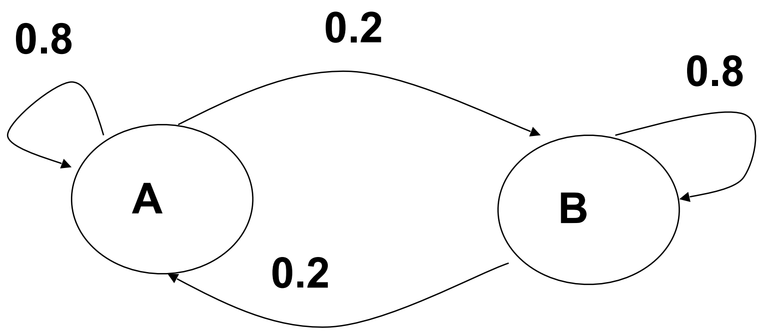
More formally $P(q_{t+1} = s_i \mid q_t = s_j) = P(q_{t+1} = s_i \mid q_t = s_j, q_{t-1} = s_j)$



What can we ask when using a HMM?

A few examples:

- "What dice is currently being used?"
- "What is the probability of a 6 in the next role?"
- "What is the probability of 6 in any of the next 3 roles?"

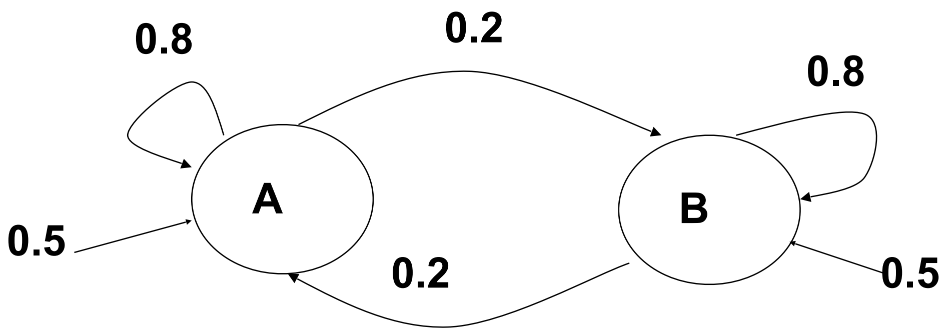


Inference in HMMs

- Computing P(Q) and $P(q_t = s_i)$
  - If we cannot look at observations
- Computing P(Q | O) and $P(q_t = s_i | O)$
  - When we have observation and care about the last state only
- Computing $\text{argmax}_Q P(Q | O)$
  - When we care about the entire path

What dice is currently being used?

- We played *t* rounds so far
- We want to determine $P(q_t = A)$
- Lets assume for now that we cannot observe any outputs (we are blind folded)
- How can we compute this?

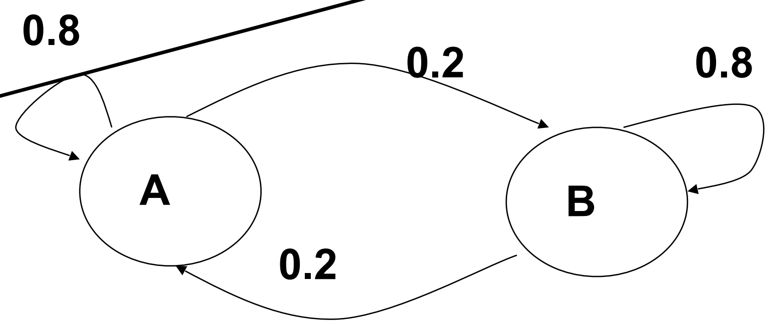


$P(q_t = A)$?

- Simple answer:
  Lets determine P(Q) where Q is any path that ends in A
  $Q = q_1, \dots q_{t-1}, A$
  $P(Q) = P(q_1, \dots q_{t-1}, A) = P(A \mid q_1, \dots q_{t-1})\, P(q_1, \dots q_{t-1}) = P(A \mid q_{t-1})\, P(q_1, \dots q_{t-1}) = \dots = P(A \mid q_{t-1}) \dots P(q_2 \mid q_1)\, P(q_1)$

Markov property!

Initial probability



$P(q_t = A)$?

- Simple answer:
  1. Lets determine P(Q) where Q is any path that ends in A
  $Q = q_1, \dots q_{t-1}, A$
  $P(Q) = P(q_1, \dots q_{t-1}, A) = P(A \mid q_1, \dots q_{t-1})\, P(q_1, \dots q_{t-1}) = P(A \mid q_{t-1})\, P(q_1, \dots q_{t-1}) = \dots = P(A \mid q_{t-1}) \dots P(q_2 \mid q_1)\, P(q_1)$

  2. $P(q_t = A) = \Sigma P(Q)$
  where the sum is over all sets of t states that end in A

$P(q_t = A)$?

- Simple answer:

  1. Lets determine $P(Q)$ where Q is any path that ends in A

  $Q = q_1, \ldots q_{t-1}, A$

  $P(Q) = P(q_1, \ldots q_{t-1}, A) = P(A \mid q_1, \ldots q_{t-1})\, P(q_1, \ldots q_{t-1}) = P(A \mid q_{t-1})\, P(q_1, \ldots q_{t-1}) = \ldots = P(A \mid q_{t-1}) \ldots P(q_2 \mid q_1)\, P(q_1)$

  2. $P(q_t = A) = \Sigma P(Q)$

  where the sum is over all sets of t sates that end in A

Q: How many sets Q are there?

A: A lot! (2^{t-1})

Not a feasible solution

$P(q_t = A)$, the smart way

- Lets define $p_t(i)$ as the probability of being in state *i* at time t: $p_t(i) = p(q_t = s_i)$
- We can determine $p_t(i)$ by induction

  1. $p_1(i) = \Pi_i$

  2. $p_t(i) = ?$

$P(q_t = A)$, the smart way

- Lets define $p_t(i)$ = probability state i at time t = $p(q_t = s_i)$
- We can determine $p_t(i)$ by induction

1. $p_1(i) = \Pi_i$
2. $p_t(i) = \Sigma_j \, p(q_t = s_i \mid q_{t-1} = s_j)p_{t-1}(j)$

$P(q_t = A)$, the smart way

- Lets define $p_t(i)$ = probability state i at time t = $p(q_t = s_i)$
- We can determine $p_t(i)$ by induction

1. $p_1(i) = \Pi_i$
2. $p_t(i) = \Sigma_j \, p(q_t = s_i \mid q_{t-1} = s_j)p_{t-1}(j)$

This type of computation is called dynamic programming

Complexity: $O(n^2 {*} t)$

Number of states in our HMM

Time / state	t1	t2	t3
s1	.3		
s2	.7		

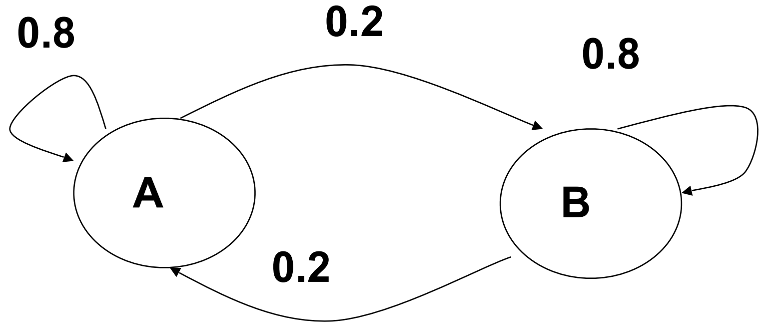
Inference in HMMs

- Computing P(Q) and $P(q_t = s_i)$ √
- Computing P(Q | O) and $P(q_t = s_i | O)$
- Computing $\text{argmax}_Q P(Q)$

But what if we observe outputs?

- So far, we assumed that we could not observe the outputs
- In reality, we almost always can.

v	P(v |A)	P(v |B)
1	.3	.1
2	.2	.1
3	.2	.1
4	.1	.2
5	.1	.2
6	.1	.3



But what if we observe outputs?

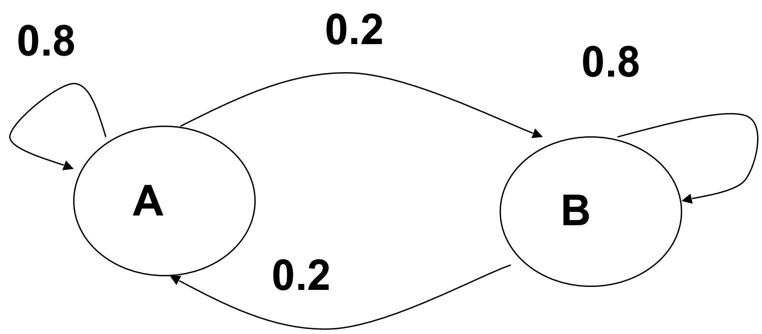
- So far, we assumed that we could not observe the outputs
- In reality, we almost always can.

Does observing the sequence

5, 6, 4, 5, 6, 6

Change our belief about the state?

v	P(v |A)	P(v |B)
1	.3	.1
2	.2	.1
3	.2	.1
4	.1	.2
5	.1	.2
6	.1	.3



$P(q_t = A)$ when outputs are observed

- We want to compute $P(q_t = A \mid O_1 \ldots O_t)$
- For ease of writing we will use the following notations (commonly used in the literature)
- $a_{j,i} = P(q_t = s_i \mid q_{t-1} = s_j)$
- $b_i(o_t) = P(o_t \mid s_i)$

Transition probability

Emission probability

$P(q_t = A)$ when outputs are observed

- We want to compute $P(q_t = A \mid O_1 \ldots O_t)$
- Lets start with a simpler question. Given a sequence of states Q, what is $P(Q \mid O_1 \ldots O_t) = P(Q \mid O)$?
  - It is pretty simple to move from P(Q) to $P(q_t = A)$
  - In some cases P(Q) is the more important question
    - Speech processing
    - NLP

P(Q | O)

- We can use Bayes rule:

$$P(Q \mid O) = \frac{P(O \mid Q)\,P(Q)}{P(O)}$$

Easy, $P(O \mid Q) = P(o_1 \mid q_1)\, P(o_2 \mid q_2) \ldots P(o_t \mid q_t)$

P(Q | O)

- We can use Bayes rule:

$$P(Q|O) = \frac{P(O|Q)P(Q)}{P(O)}$$

Easy, $P(Q) = P(q_1)\,P(q_2 \mid q_1) \ldots P(q_t \mid q_{t-1})$

P(Q | O)

- We can use Bayes rule:

$$P(Q|O) = \frac{P(O|Q)P(Q)}{P(O)}$$

Hard!

P(O)

- What is the probability of seeing a set of observations:
  - An important question in it own rights, for example classification using two HMMs
- Define $\alpha_t(i) = P(o_1, o_2 \ldots, o_t \wedge q_t = s_i)$
- $\alpha_t(i)$ is the probability that we:

  1. Observe $o_1, o_2 \ldots, o_t$
  2. End up at state i

How do we compute $\alpha_t(i)$?

Computing $\alpha_t(i)$

- $\alpha_1(i) = P(o_1 \wedge q_t = i) = P(o_1 \mid q_t = s_i)\Pi_I$

We must be at a state in time t

chain rule

Markov property

Example: Computing $\alpha_3(B)$

- We observed 2,3,6

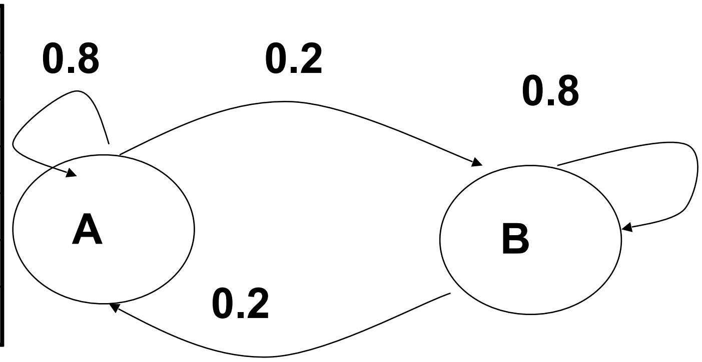
$\alpha_1(A) = P(2 \wedge q_1 = A) = P(2 \mid q_1 = A)\Pi_A = .2*.7 = .14$, $\alpha_1(B) = .1*.3 = .03$

$\alpha_2(A) = \Sigma_{j=A,,B} b_A(3) a_{j,A} \alpha_1(j) = .2*.8*.14+.2*.2*.03 = 0.0236$, $\alpha_2(B) = 0.0052$

$\alpha_3(B) = \Sigma_{j=A,,B} b_B(6) a_{j,B} \alpha_2(j) = .3*.2*.0236+.3*.8*.0052 = 0.00264$

$\Pi_A = 0.7$
$\Pi_b = 0.3$

v	P(v |A)	P(v |B)
1	.3	.1
2	.2	.1
3	.2	.1
4	.1	.2
5	.1	.2
6	.1	.3



Where we are

- We want to compute P(Q | O)
- For this, we only need to compute P(O)
- We know how to compute $\alpha_t(i)$

From now its easy

$\alpha_t(i) = P(o_1, o_2 \ldots, o_t \wedge q_t = s_i)$

so

$P(O) = P(o_1, o_2 \ldots, o_t) = \Sigma_i P(o_1, o_2 \ldots, o_t \wedge q_t = s_i) = \Sigma_i \alpha_t(i)$

note that

$$p(q_t = s_i \mid o_1, o_2 \ldots, o_t) = \frac{\alpha_t(i)}{\sum_j \alpha_t(j)}$$

P(A | B) = P(A $\wedge$ B) / P(B)

Complexity

- How long does it take to compute $P(Q \mid O)$?
- $P(Q)$: $O(n)$
- $P(O|Q)$: $O(n)$
- $P(O)$: $O(n^2 t)$

Inference in HMMs

- Computing $P(Q)$ and $P(q_t = s_i)$ $\surd$
- Computing $P(Q \mid O)$ and $P(q_t = s_i \mid O)$ $\surd$
- Computing $\text{argmax}_Q P(Q)$

Most probable path

- We are almost done …
- One final question remains

  How do we find the most probable path, that is Q* such that

$$P(Q^* \mid O) = \mathrm{argmax}_Q P(Q|O)?$$

- This is an important path
  - The words in speech processing
  - The set of genes in the genome
  - etc.

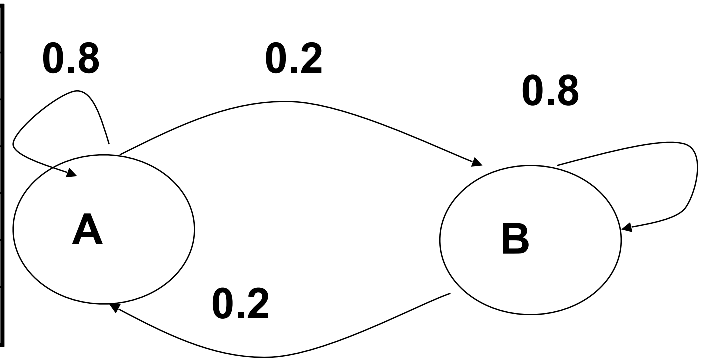
Example

- What is the most probable set of states leading to the sequence:

1,2,2,5,6,5,1,2,3 ?

Π_A=0.7
Π_b=0.3

v	P(v |A)	P(v |B)
1	.3	.1
2	.2	.1
3	.2	.1
4	.1	.2
5	.1	.2
6	.1	.3



Most probable path

$$\arg\max_Q P(Q \mid O) = \arg\max_Q \frac{P(O \mid Q)P(Q)}{P(O)}$$
$$= \arg\max_Q P(O \mid Q)P(Q)$$

We will use the following definition:

$$\delta_t(i) = \max_{q_1 \ldots q_{t-1}} p(q_1 \ldots q_{t-1} \wedge q_t = s_i \wedge O_1 \ldots O_t)$$

In other words we are interested in the most likely path from 1 to t that:

1. Ends in S_i
2. Produces outputs $O_1 \ldots O_t$

Computing $\delta_t(i)$

$$\delta_1(i) = p(q_1 = s_i \wedge O_1)$$
$$= p(q_1 = s_i)\, p(O_1 \mid q_1 = s_i)$$
$$= \pi_i b_i(O_1)$$

$$\delta_t(i) = \max_{q_1 \ldots q_{t-1}} p(q_1 \ldots q_{t-1} \wedge q_t = s_i \wedge O_1 \ldots O_t)$$

Q: Given $\delta_t(i)$, how can we compute $\delta_{t+1}(i)$?

A: To get from $\delta_t(i)$ to $\delta_{t+1}(i)$ we need to

1. Add an emission for time t+1 (O_{t+1})
2. Transition to state s_i

$$\delta_{t+1}(i) = \max_{q_1 \ldots q_t} p(q_1 \ldots q_t \wedge q_{t+1} = s_i \wedge O_1 \ldots O_{t+1})$$
$$= \max_j \delta_t(j)\, p(q_{t+1} = s_i \mid q_t = s_j)\, p(O_{t+1} \mid q_{t+1} = s_i)$$
$$= \max_j \delta_t(j)\, a_{j,i} b_i(O_{t+1})$$

The Viterbi algorithm

$$\delta_{t+1}(i) = \max_{q_1 \ldots q_t} p(q_1 \ldots q_t \wedge q_{t+1} = s_i \wedge O_1 \ldots O_{t+1})$$
$$= \max_j \delta_t(j) p(q_{t+1} = s_i \mid q_t = s_j) p(O_{t+1} \mid q_{t+1} = s_i)$$
$$= \max_j \delta_t(j) a_{j,i} b_i(O_{t+1})$$

- Once again we use dynamic programming for solving $\delta_t(i)$
- Once we have $\delta_t(i)$, we can solve for our P(Q*|O)

By:

P(Q* | O) = argmax_QP(Q|O) =

path defined by $\text{argmax}_j\ \delta_t(j)$,

Inference in HMMs

- Computing P(Q) and P(q_t = s_i) √
- Computing P(Q | O) and P(q_t = s_i |O) √
- Computing argmax_QP(Q) √

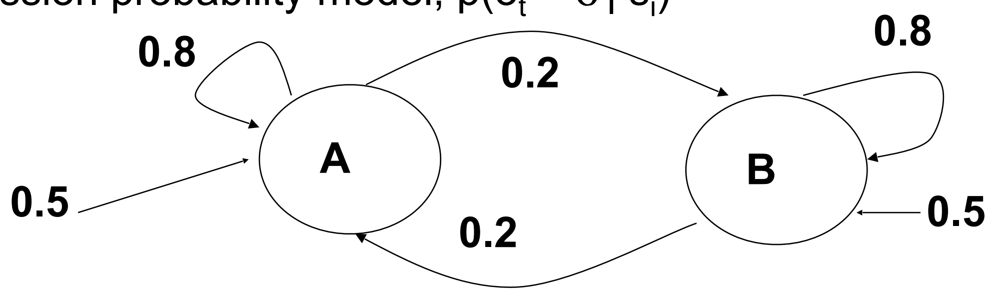
Learning in HMMs

A Hidden Markov model

- A set of states $\{s_1 \dots s_n\}$

  - In each time point we are in exactly one of these states denoted by q_t
- Π_i, the probability that we *start* at state s_i
- A transition probability model, $P(q_t = s_i \mid q_{t-1} = s_j)$
- A set of possible outputs Σ

  - At time *t* we emit a symbol $\sigma \in \Sigma$
- An emission probability model, $p(o_t = \sigma \mid s_i)$



Inference in HMMs

- Computing $P(Q)$ and $P(q_t = s_i)$ √
- Computing $P(Q \mid O)$ and $P(q_t = s_i \mid O)$ √
- Computing $\text{argmax}_Q P(Q)$ √

Learning HMMs

- Until now we assumed that the emission and transition probabilities are known
- This is usually not the case
  - How is "AI" pronounced by different individuals?
  - What is the probability of hearing "class" after "AI"?

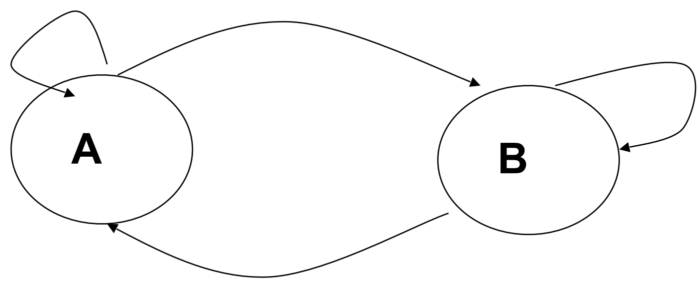
While we will discuss learning the transition and emission models, we will not discuss selecting the states.

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

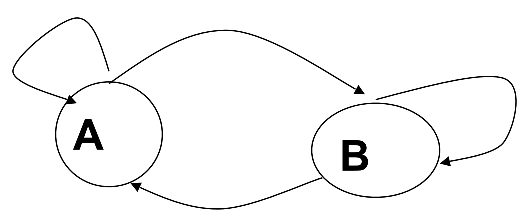
Find the initial probabilities π such that

k is the number of sequences avialable for training

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

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

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



Transition probabilities

Q: assume we can observe the set of states:

AAABBAAAABBBBBAAAABBBB

how can we learn the transition probabilities?

A: Maximum likelihood estimation

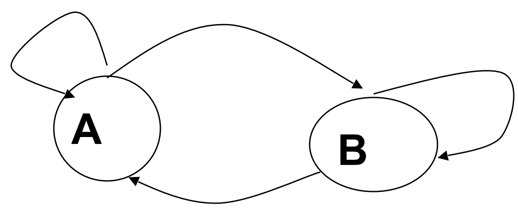
Find a transition matrix *a* such that

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

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

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

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



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

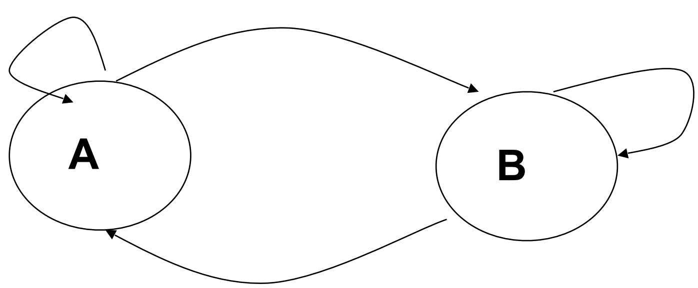
and the set of dice values

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

how can we learn the emission probabilities?

A: Maximum likelihood estimation

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



Learning HMMs

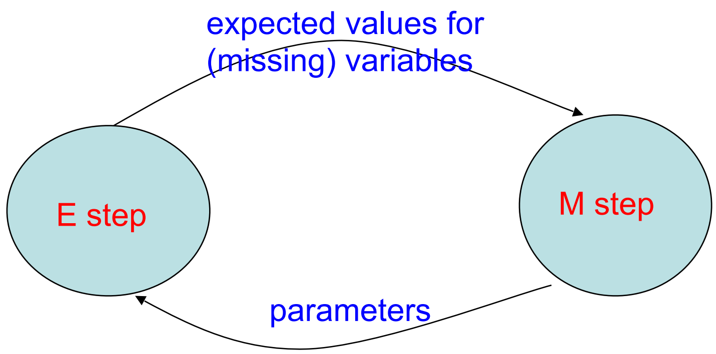
- In most case we do not know what states generated each of the outputs (fully unsupervised)
- … but had we known, it would be very easy to determine an emission and transition model!
- On the other hand, if we had such a model we could determine the set of states using the inference methods we discussed

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).



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

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

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)

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

State and transition probabilities

- Probability of a state

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

- We can also derive a transition probability

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

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

E step

- Compute $S_t(i)$ and $S_t(i,j)$ for all t, i, and j ($1 \le t \le n$, $1 \le i \le k$, $2 \le j \le k$)

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

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

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

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

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

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

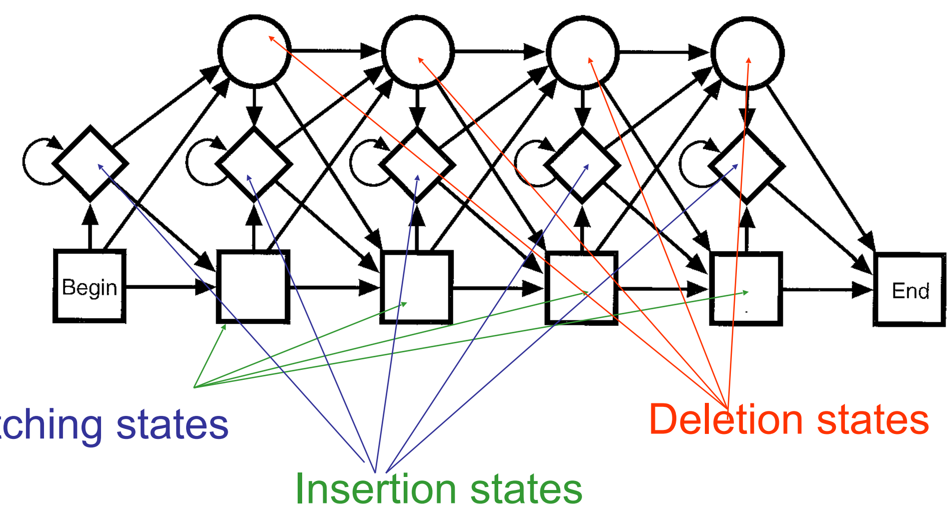
  2. Number of states, model

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?
5. Output complete model

No

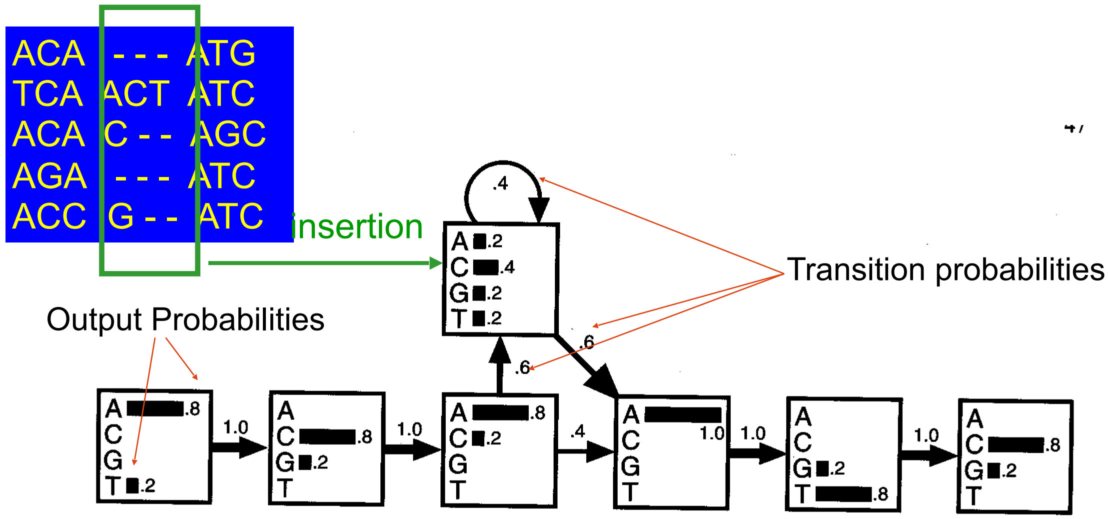
We did not discuss initial probability estimation. These can be deduced from multiple sets of observation (for example, several recorded customers for speech processing)

Building HMMs–*Topology*



No of matching states = average sequence length in the family
PFAM Database - of Protein families
(**http://pfam.wustl.edu**)

Building – *from an existing alignment*



A HMM model for a DNA motif alignments, The transitions are shown with arrows whose thickness indicate their probability. In each state, the histogram shows the probabilities of the four bases.

What you should know

- Why HMMs? Which applications are suitable?
- Inference in HMMs
  - No observations
  - Probability of next state w. observations
  - Maximum scoring path (Viterbi)

References

- Big thanks to Prof. Ziv Bar-Joseph @ CMU for allowing me to reuse some of his slides