

Machine Learning

10-701/15-781, Fall 2011

Hidden Markov Model

Eric Xing

Lecture 11, October 17, 2011



Reading: Chap. 13 CB

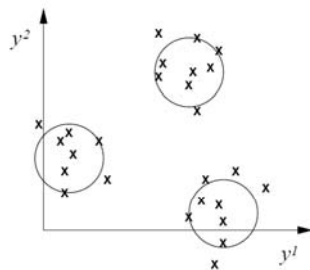
© Eric Xing @ CMU, 2006-2011

1



Motivation

- Clustering:



© Eric Xing @ CMU, 2006-2011

2



A somewhat similar problem



An experience in a casino

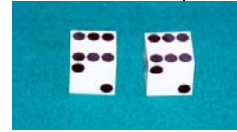
Game:

1. You bet \$1
2. You roll (always with a fair die)
3. Casino player rolls (maybe with fair die, maybe with loaded die)
4. Highest number wins \$2

Question:

1245526462146146136136661664661636
616366163616515615115146123562344

Which die is being used in each play?



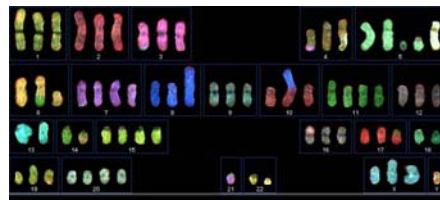
© Eric Xing @ CMU, 2006-2011

3

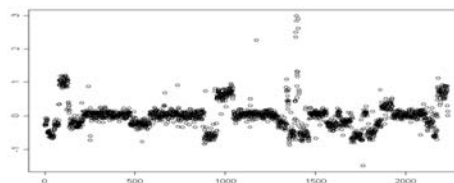
Question:



- Naturally, data points arrive one at a time
 - Does the ordering index carry (additional) clustering information besides the data value itself?
 - Example:
Chromosomes of tumor cell:



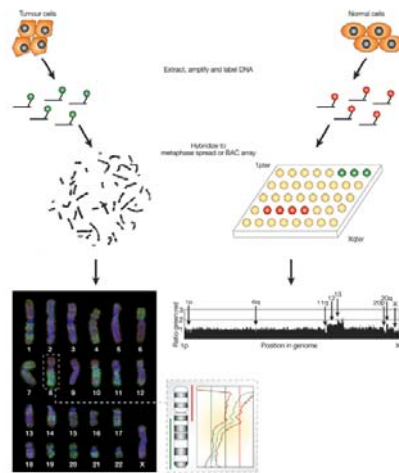
Copy number measurements
(known as CGH)



© Eric Xing @ CMU, 2006-2011

4

Array CGH (comparative genomic hybridization)

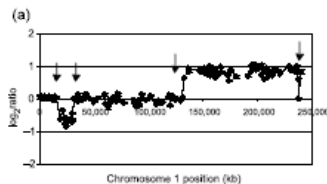


- The basic assumption of a CGH experiment is that the ratio of the binding of test and control DNA is proportional to the ratio of the copy numbers of sequences in the two samples.
- But various kinds of noises make the true observations less easy to interpret ...

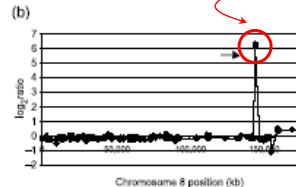
© Eric Xing @ CMU, 2006-2011

5

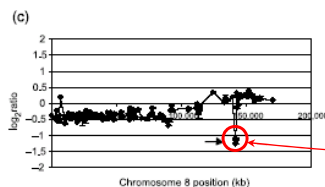
DNA Copy number aberration types in breast cancer



Copy number profile for chromosome 1 from 600 MPE cell line



Copy number profile for chromosome 8 from COLO320 cell line



Copy number profile for chromosome 8 in MDA-MB-231 cell line

60-70 fold amplification of CMYC region

deletion

© Eric Xing @ CMU, 2006-2011

6

Question:

- Sometimes, just data value by itself is hardly clusterable!



- Unlike continuous vectors, which can take different values in an "infinite" space, and often naturally settle to different cluster just due to value differences, entities with discrete attributes often can not manifest their labels by a one time snapshot of their discrete values alone, sometime additional information is needed ...

- e.g.,

1245526462146146136136661664661636616366163616515615115146123562344

© Eric Xing @ CMU, 2006-2011

7

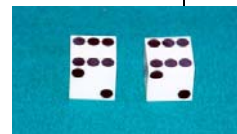
Suppose you were told about the following story before heading to Vegas...

The Dishonest Casino !!!

A casino has two dice:

- Fair die
 $P(1) = P(2) = P(3) = P(5) = P(6) = 1/6$
- Loaded die
 $P(1) = P(2) = P(3) = P(5) = 1/10$
 $P(6) = 1/2$

Casino player switches back-&-forth between fair and loaded die once every 20 turns



© Eric Xing @ CMU, 2006-2011

8

Puzzles Regarding the Dishonest Casino



GIVEN: A sequence of rolls by the casino player

1245526462146146136136661664661636616366163616515615115146123562344
???

QUESTION

- How likely is this sequence, given our model of how the casino works?
 - This is the **EVALUATION** problem
- What portion of the sequence was generated with the fair die, and what portion with the loaded die?
 - This is the **DECODING** question
- How “loaded” is the loaded die? How “fair” is the fair die? How often does the casino player change from fair to loaded, and back?
 - This is the **LEARNING** question

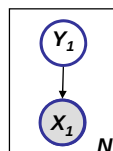
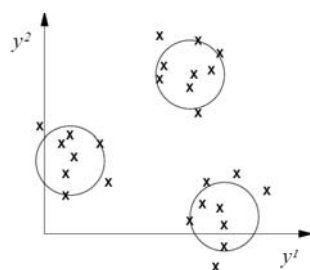
© Eric Xing @ CMU, 2006-2011

9

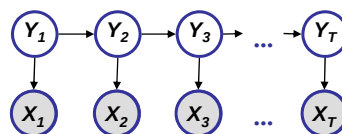
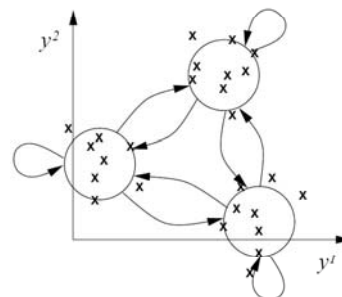
From static to dynamic mixture models



Static mixture



Dynamic mixture



© Eric Xing @ CMU, 2006-2011

10

Hidden Markov Models

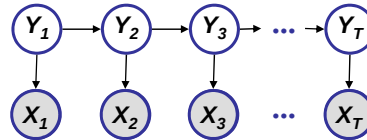


The underlying source:

genomic entities,
dice,

The sequence:

CGH signal,
sequence of rolls,

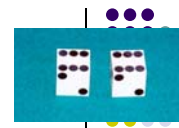


Markov property:

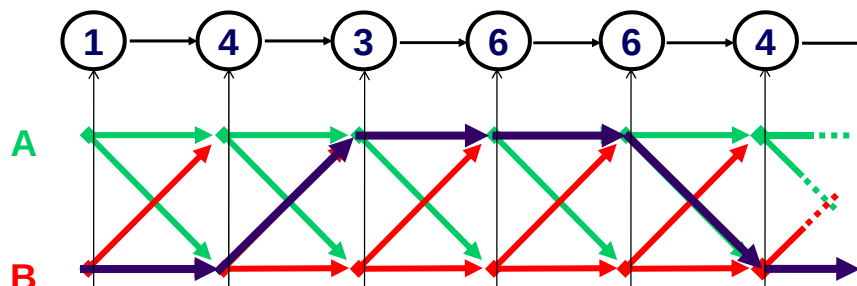
© Eric Xing @ CMU, 2006-2011

11

An HMM is a Stochastic Generative Model



- Observed sequence:



- Hidden sequence (a parse or segmentation):

© Eric Xing @ CMU, 2006-2011

12

Definition (of HMM)

- **Observation space**

Alphabetic set:

$$C = \{c_1, c_2, \dots, c_K\}$$

Euclidean space:

$$\mathbb{R}^d$$

- **Index set of hidden states**

$$I = \{1, 2, \dots, M\}$$

- **Transition probabilities between any two states**

$$p(y_t^j = 1 | y_{t-1}^i = 1) = a_{i,j},$$

or $p(y_t | y_{t-1}^i = 1) \sim \text{Multinomial}(a_{i,1}, a_{i,2}, \dots, a_{i,M}), \forall i \in I.$

- **Start probabilities**

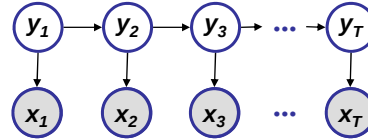
$$p(y_1) \sim \text{Multinomial}(\pi_1, \pi_2, \dots, \pi_M).$$

- **Emission probabilities associated with each state**

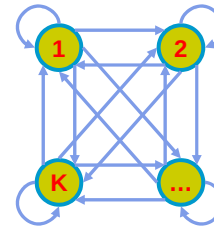
$$p(x_t | y_t^i = 1) \sim \text{Multinomial}(b_{i,1}, b_{i,2}, \dots, b_{i,K}), \forall i \in I.$$

or in general:

$$p(x_t | y_t^i = 1) \sim f(\cdot | \theta_i), \forall i \in I.$$



Graphical model

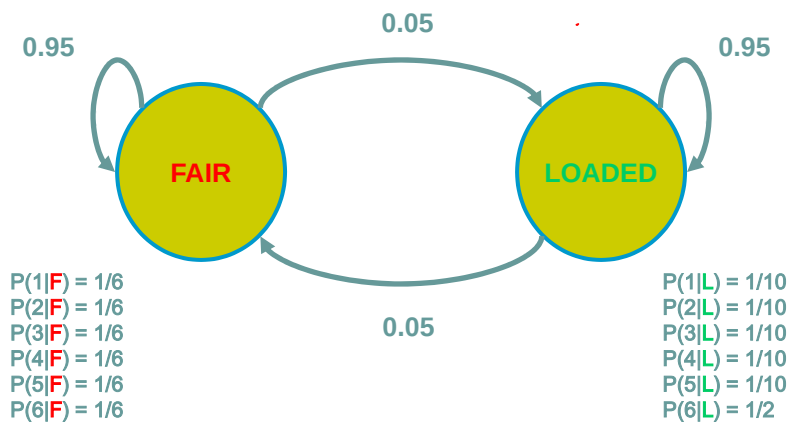


State automata

© Eric Xing @ CMU, 2006-2011

13

The Dishonest Casino Model



© Eric Xing @ CMU, 2006-2011

14

Three Main Questions on HMMs



1. Evaluation

GIVEN an HMM M , and a sequence x ,
FIND Prob ($x | M$)
ALGO. Forward

2. Decoding

GIVEN an HMM M , and a sequence x ,
FIND the sequence y of states that maximizes, e.g., $P(y | x, M)$,
or the most probable subsequence of states
ALGO. Viterbi, Forward-backward

3. Learning

GIVEN an HMM M , with unspecified transition/emission probs.,
and a sequence x ,
FIND parameters $\theta = (\pi_i, a_{ij}, \eta_{ik})$ that maximize $P(x | \theta)$
ALGO. Baum-Welch (EM)

© Eric Xing @ CMU, 2006-2011

15

Joint Probability



1245526462146146136136661664661636616366163616515615115146123562344
FF

- When the state-labeling is known, this is easy ...

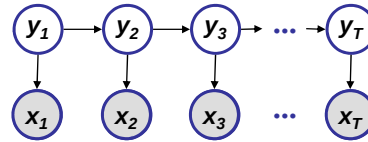
$$P(\mathbf{X}, \mathbf{Y}) \quad ?$$

© Eric Xing @ CMU, 2006-2011

16

Probability of a Parse

- Given a sequence $\mathbf{x} = x_1, \dots, x_T$ and a parse $\mathbf{y} = y_1, \dots, y_T$,
- To find how likely is the parse:
(given our HMM and the sequence)



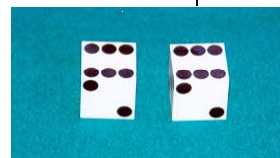
$$\begin{aligned}
 p(\mathbf{x}, \mathbf{y}) &= p(x_1, \dots, x_T, y_1, \dots, y_T) && \text{(Joint probability)} \\
 &= p(y_1) p(x_1 | y_1) p(y_2 | y_1) p(x_2 | y_2) \dots p(y_T | y_{T-1}) p(x_T | y_T) \\
 &= p(y_1) P(y_2 | y_1) \dots p(y_T | y_{T-1}) \times p(x_1 | y_1) p(x_2 | y_2) \dots p(x_T | y_T)
 \end{aligned}$$

© Eric Xing @ CMU, 2006-2011

17

Example: the Dishonest Casino

- Let the sequence of rolls be:
 - $\mathbf{x} = 1, 2, 1, 5, 6, 2, 1, 6, 2, 4$
- Then, what is the likelihood of
 - $\mathbf{y} = \text{Fair, Fair, Fair, Fair, Fair, Fair, Fair, Fair, Fair, Fair?}$
(say initial probs $a_{0\text{Fair}} = \frac{1}{2}$, $a_{0\text{Loaded}} = \frac{1}{2}$)



$$\frac{1}{2} \times P(1 | \text{Fair}) P(\text{Fair} | \text{Fair}) P(2 | \text{Fair}) P(\text{Fair} | \text{Fair}) \dots P(4 | \text{Fair}) =$$

$$\frac{1}{2} \times (\frac{1}{6})^{10} \times (0.95)^9 = .00000000521158647211 = 5.21 \times 10^{-9}$$

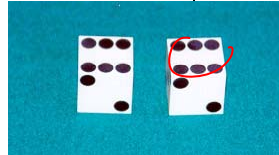
© Eric Xing @ CMU, 2006-2011

18

Example: the Dishonest Casino



- So, the likelihood the die is fair in all this run is just 5.21×10^{-9}



- OK, but what is the likelihood of
 - π = Loaded, Loaded, Loaded, Loaded, Loaded, Loaded, Loaded, Loaded, Loaded, Loaded, Loaded?

$$\frac{1}{2} \times P(1 \mid \text{Loaded}) P(\text{Loaded} \mid \text{Loaded}) \dots P(4 \mid \text{Loaded}) =$$

$$\frac{1}{2} \times (1/10)^8 \times (1/2)^2 (0.95)^9 = .00000000078781176215 = 0.79 \times 10^{-9}$$

- Therefore, it is after all 6.59 times more likely that the die is fair all the way, than that it is loaded all the way

© Eric Xing @ CMU, 2006-2011

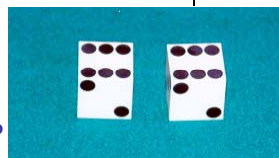
19

Example: the Dishonest Casino



- Let the sequence of rolls be:

- $x = 1, 6, 6, 5, 6, 2, 6, 6, 3, 6$



- Now, what is the likelihood $\pi = F, F, \dots, F$?

- $\frac{1}{2} \times (1/6)^{10} \times (0.95)^9 = 0.5 \times 10^{-9}$, same as before

- What is the likelihood $y = L, L, \dots, L$?

$$\frac{1}{2} \times (1/10)^4 \times (1/2)^6 (0.95)^9 = .00000049238235134735 = 5 \times 10^{-7}$$

- So, it is 100 times more likely the die is loaded

© Eric Xing @ CMU, 2006-2011

20

Marginal Probability



1245526462146146136136661664661636616366163616515615115146123562344

FF

- What if state-labeling Y is not observed

$$P(\mathbf{X}) \quad ?$$

The Forward Algorithm



- We want to calculate $P(\mathbf{x})$, the likelihood of $\mathbf{x}$, given the HMM
 - Sum over all possible ways of generating $\mathbf{x}$:

$$p(\mathbf{x}) = \sum_{\mathbf{y}} p(\mathbf{x}, \mathbf{y}) = \sum_{y_1} \sum_{y_2} \cdots \sum_{y_N} \pi_{y_1} \prod_{t=2}^T a_{y_{t-1}, y_t} \prod_{t=1}^T p(x_t | y_t)$$

- To avoid summing over an exponential number of paths $\mathbf{y}$, define

$$\alpha(y_t^k = 1) = \alpha_t^k \stackrel{\text{def}}{=} P(x_1, \dots, x_t, y_t^k = 1) \quad (\text{the forward probability})$$

- The recursion:

$$\alpha_t^k = p(x_t | y_t^k = 1) \sum_i \alpha_{t-1}^i a_{i,k}$$

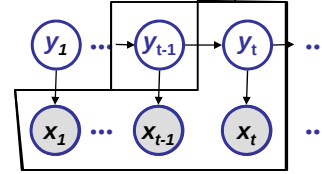
$$P(\mathbf{x}) = \sum_k \alpha_T^k$$

The Forward Algorithm – derivation



- Compute the forward probability:

$$\alpha_t^k = P(x_1, \dots, x_{t-1}, x_t, y_t^k = 1)$$



$$\begin{aligned} &= \sum_{y_{t-1}} P(x_1, \dots, x_{t-1}, y_{t-1}) P(y_t^k = 1 | y_{t-1}, x_1, \dots, x_{t-1}) P(x_t | y_t^k = 1, x_1, \dots, x_{t-1}, y_{t-1}) \\ &= \sum_{y_{t-1}} P(x_1, \dots, x_{t-1}, y_{t-1}) P(y_t^k = 1 | y_{t-1}) P(x_t | y_t^k = 1) \\ &= P(x_t | y_t^k = 1) \sum_i P(x_1, \dots, x_{t-1}, y_{t-1}^i = 1) P(y_t^k = 1 | y_{t-1}^i = 1) \\ &= P(x_t | y_t^k = 1) \sum_i \alpha_{t-1}^i a_{i,k} \end{aligned}$$

Chain rule: $P(A, B, C) = P(A)P(B | A)P(C | A, B)$

© Eric Xing @ CMU, 2006-2011

23

The Forward Algorithm



- We can compute α_t^k for all k, t , using dynamic programming!

Initialization:

$$\alpha_1^k = P(x_1 | y_1^k = 1) \pi_k$$

$$\begin{aligned} \alpha_1^k &= P(x_1, y_1^k = 1) \\ &= P(x_1 | y_1^k = 1) P(y_1^k = 1) \\ &= P(x_1 | y_1^k = 1) \pi_k \end{aligned}$$

Iteration:

$$\alpha_t^k = P(x_t | y_t^k = 1) \sum_i \alpha_{t-1}^i a_{i,k}$$

Termination:

$$P(\mathbf{x}) = \sum_k \alpha_T^k$$

© Eric Xing @ CMU, 2006-2011

24

Three Main Questions on HMMs

1. Evaluation

GIVEN an HMM M , and a sequence x ,
 FIND Prob ($x | M$)
 ALGO. **Forward**

2. Decoding

GIVEN an HMM M , and a sequence x ,
 FIND the sequence y of states that maximizes, e.g., $P(y | x, M)$,
 or the most probable subsequence of states
 ALGO. **Viterbi, Forward-backward**

3. Learning

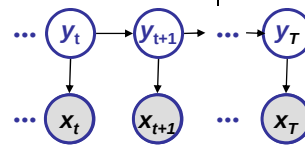
GIVEN an HMM M , with unspecified transition/emission probs.,
 and a sequence x ,
 FIND parameters $\theta = (\pi_i, a_{ij}, \eta_{ik})$ that maximize $P(x | \theta)$
 ALGO. **Baum-Welch (EM)**

© Eric Xing @ CMU, 2006-2011

25

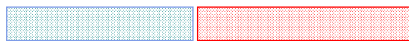
The Backward Algorithm

- We want to compute $P(y_t^k = 1 | x)$,
 the posterior probability distribution on the t^{th} position, given x



- We start by computing

$$\begin{aligned} P(y_t^k = 1, x) &= P(x_1, \dots, x_t, y_t^k = 1, x_{t+1}, \dots, x_T) \\ &= P(x_1, \dots, x_t, y_t^k = 1) P(x_{t+1}, \dots, x_T | x_1, \dots, x_t, y_t^k = 1) \\ &= P(x_1 \dots x_t, y_t^k = 1) P(x_{t+1} \dots x_T | y_t^k = 1) \end{aligned}$$



- The recursion: Forward, α_t^k Backward, $\beta_t^k = P(x_{t+1}, \dots, x_T | y_t^k = 1)$

$$\beta_t^k = \sum_i a_{k,i} p(x_{t+1} | y_{t+1}^i = 1) \beta_{t+1}^i$$

© Eric Xing @ CMU, 2006-2011

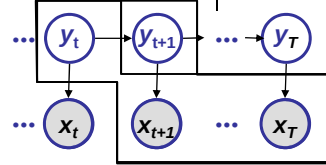
26

The Backward Algorithm – derivation



- Define the backward probability:

$$\begin{aligned}
 \beta_t^k &= P(x_{t+1}, \dots, x_T | y_t^k = 1) \\
 &= \sum_{y_{t+1}} P(y_{t+1}, x_{t+1}, \dots, x_T | y_t^k = 1) \\
 &= \sum_i P(y_{t+1}^i = 1 | y_t^k = 1) p(x_{t+1} | y_{t+1}^i = 1, y_t^k = 1) P(x_{t+2}, \dots, x_T | x_{t+1}, y_{t+1}^i = 1, y_t^k = 1) \\
 &= \sum_i P(y_{t+1}^i = 1 | y_t^k = 1) p(x_{t+1} | y_{t+1}^i = 1) P(x_{t+2}, \dots, x_T | y_{t+1}^i = 1) \\
 &= \sum_i a_{k,i} p(x_{t+1} | y_{t+1}^i = 1) \beta_{t+1}^i
 \end{aligned}$$



Chain rule: $P(A, B, C | \alpha) = P(A | \alpha) P(B | A, \alpha) P(C | A, B, \alpha)$

© Eric Xing @ CMU, 2006-2011

27

The Backward Algorithm



- We can compute β_t^k for all k, t , using dynamic programming!

Initialization:

$$\beta_T^k = 1, \forall k$$

Iteration:

$$\beta_t^k = \sum_i a_{k,i} P(x_{t+1} | y_{t+1}^i = 1) \beta_{t+1}^i$$

Termination:

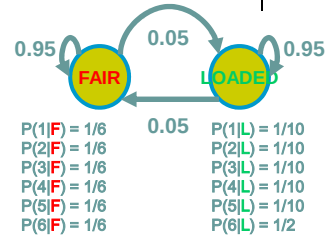
$$P(\mathbf{x}) = \sum_k \alpha_1^k \beta_1^k$$

© Eric Xing @ CMU, 2006-2011

28

Example:

$x = 1, 2, 1, 5, 6, 2, 1, 6, 2, 4$



$$\alpha_t^k = P(x_t | y_t^k = 1) \sum_i \alpha_{t-1}^i a_{i,k}$$

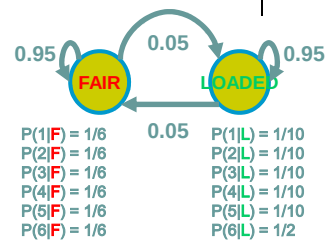
$$\beta_t^k = \sum_i a_{k,i} P(x_{t+1} | y_{t+1}^i = 1) \beta_{t+1}^i$$

© Eric Xing @ CMU, 2006-2011

29

$x = 1, 2, 1, 5, 6, 2, 1, 6, 2, 4$

Alpha (actual)		Beta (actual)	
0.0833	0.0500	0.0000	0.0000
0.0136	0.0052	0.0000	0.0000
0.0022	0.0006	0.0000	0.0000
0.0004	0.0001	0.0000	0.0000
0.0001	0.0000	0.0001	0.0001
0.0000	0.0000	0.0007	0.0006
0.0000	0.0000	0.0045	0.0055
0.0000	0.0000	0.0264	0.0112
0.0000	0.0000	0.1633	0.1033
0.0000	0.0000	1.0000	1.0000



$$\alpha_t^k = P(x_t | y_t^k = 1) \sum_i \alpha_{t-1}^i a_{i,k}$$

$$\beta_t^k = \sum_i a_{k,i} P(x_{t+1} | y_{t+1}^i = 1) \beta_{t+1}^i$$

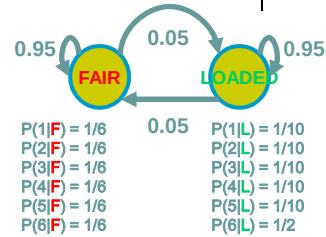
© Eric Xing @ CMU, 2006-2011

30



$x = 1, 2, 1, 5, 6, 2, 1, 6, 2, 4$

Alpha (logs)		Beta (logs)	
-2.4849	-2.9957	-16.2439	-17.2014
-4.2969	-5.2655	-14.4185	-14.9922
-6.1201	-7.4896	-12.6028	-12.7337
-7.9499	-9.6553	-10.8042	-10.4389
-9.7834	-10.1454	-9.0373	-9.7289
-11.5905	-12.4264	-7.2181	-7.4833
-13.4110	-14.6657	-5.4135	-5.1977
-15.2391	-15.2407	-3.6352	-4.4938
-17.0310	-17.5432	-1.8120	-2.2698
-18.8430	-19.8129	0	0



$$\alpha_t^k = P(x_t | y_t^k = 1) \sum_i \alpha_{t-1}^i a_{i,k}$$

$$\beta_t^k = \sum_i a_{k,i} P(x_{t+1} | y_{t+1}^i = 1) \beta_{t+1}^i$$

What is the probability of a hidden state prediction?



$x = 1, 2, 1, 5, 6, 2, 1, 6, 2, 4$

Alpha (logs)		Beta (logs)	
-2.4849	-2.9957	-16.2439	-17.2014
-4.2969	-5.2655	-14.4185	-14.9922
-6.1201	-7.4896	-12.6028	-12.7337
-7.9499	-9.6553	-10.8042	-10.4389
-9.7834	-10.1454	-9.0373	-9.7289
-11.5905	-12.4264	-7.2181	-7.4833
-13.4110	-14.6657	-5.4135	-5.1977
-15.2391	-15.2407	-3.6352	-4.4938
-17.0310	-17.5432	-1.8120	-2.2698
-18.8430	-19.8129	0	0

What is the probability of a hidden state prediction?



- A single state:

$$P(y_t | \mathbf{X})$$

- What about a hidden state sequence ?

$$P(y_1, \dots, y_T | \mathbf{X})$$

© Eric Xing @ CMU, 2006-2011

33

Posterior decoding



- We can now calculate

$$P(y_t^k = 1 | \mathbf{x}) = \frac{P(y_t^k = 1, \mathbf{x})}{P(\mathbf{x})} = \frac{\alpha_t^k \beta_t^k}{P(\mathbf{x})}$$

- Then, we can ask

- What is the most likely state at position t of sequence $\mathbf{x}$:

$$k_t^* = \arg \max_k P(y_t^k = 1 | \mathbf{x})$$

- Note that this is an MPA of a **single** hidden state, what if we want to a MPA of a whole hidden state sequence?

- Posterior Decoding: $\{y_t^{k_t^*} = 1 : t = 1 \dots T\}$

- This is different from MPA of a **whole** sequence of hidden states

- This can be understood as **bit error rate** vs. **word error rate**

Example:
MPA of X ?
MPA of (X, Y) ?

x	y	$P(x, y)$
0	0	0.35
0	1	0.05
1	0	0.3
1	1	0.3

© Eric Xing @ CMU, 2006-2011

34

Viterbi decoding

- GIVEN $\mathbf{x} = x_1, \dots, x_T$, we want to find $\mathbf{y} = y_1, \dots, y_T$, such that $P(\mathbf{y}|\mathbf{x})$ is maximized:

$$\mathbf{y}^* = \operatorname{argmax}_{\mathbf{y}} P(\mathbf{y}|\mathbf{x}) = \operatorname{argmax}_{\pi} P(\mathbf{y}, \mathbf{x})$$

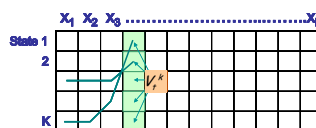
- Let

$$V_t^k = \max_{\{y_1, \dots, y_{t-1}\}} P(x_1, \dots, x_{t-1}, y_1, \dots, y_{t-1}, x_t, y_t^k = 1)$$

= Probability of most likely sequence of states ending at state $y_t = k$

- The recursion:

$$V_t^k = p(x_t | y_t^k = 1) \max_i a_{i,k} V_{t-1}^i$$



- Underflows are a significant problem

$$p(x_1, \dots, x_t, y_1, \dots, y_t) = \pi_{y_1} a_{y_1, y_2} \dots a_{y_{t-1}, y_t} b_{y_1, x_1} \dots b_{y_t, x_t}$$

- These numbers become extremely small – underflow
- Solution: Take the logs of all values: $V_t^k = \log p(x_t | y_t^k = 1) + \max_i (\log(a_{i,k}) + V_{t-1}^i)$

The Viterbi Algorithm – derivation

- Define the viterbi probability:

$$\begin{aligned} V_{t+1}^k &= \max_{\{y_1, \dots, y_t\}} P(x_1, \dots, x_t, y_1, \dots, y_t, x_{t+1}, y_{t+1}^k = 1) \\ &= \max_{\{y_1, \dots, y_t\}} P(x_{t+1}, y_{t+1}^k = 1 | x_1, \dots, x_t, y_1, \dots, y_t) P(x_1, \dots, x_t, y_1, \dots, y_t) \\ &= \max_{\{y_1, \dots, y_t\}} P(x_{t+1}, y_{t+1}^k = 1 | y_t) P(x_1, \dots, x_{t-1}, y_1, \dots, y_{t-1}, x_t, y_t) \\ &= \max_i P(x_{t+1}, y_{t+1}^k = 1 | y_t^i = 1) \max_{\{y_1, \dots, y_{t-1}\}} P(x_1, \dots, x_{t-1}, y_1, \dots, y_{t-1}, x_t, y_t^i = 1) \\ &= \max_i P(x_{t+1}, | y_{t+1}^k = 1) a_{i,k} V_t^i \\ &= P(x_{t+1}, | y_{t+1}^k = 1) \max_i a_{i,k} V_t^i \end{aligned}$$

The Viterbi Algorithm

- Input: $\mathbf{x} = x_1, \dots, x_T$

Initialization:

$$V_1^k = P(x_1 | y_1^k = 1) \pi_k$$

Iteration:

$$V_t^k = P(x_t | y_t^k = 1) \max_i a_{i,k} V_{t-1}^i$$

$$\text{Ptr}(k, t) = \arg \max_i a_{i,k} V_{t-1}^i$$

Termination:

$$P(\mathbf{x}, \mathbf{y}^*) = \max_k V_T^k$$

TraceBack:

$$y_T^* = \arg \max_k V_T^k$$

$$y_{t-1}^* = \text{Ptr}(y_t^*, t)$$

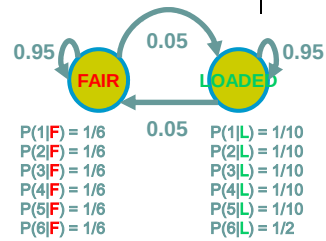
© Eric Xing @ CMU, 2006-2011

37

$\mathbf{x} = 1, 2, 1, 5, 6, 2, 1, 6, 2, 4$

V_t^k

0.6250	0.3750
0.7353	0.2647
0.8224	0.1776
0.8853	0.1147
0.7200	0.2800
0.8108	0.1892
0.8772	0.1228
0.7043	0.2957
0.7988	0.2012
0.8687	0.1313



$$\alpha_t^k = P(x_t | y_t^k = 1) \sum_i \alpha_{t-1}^i a_{i,k}$$

$$\beta_t^k = \sum_i a_{k,i} P(x_{t+1} | y_{t+1}^i = 1) \beta_{t+1}^i$$

$$V_t^k = p(x_t | y_t^k = 1) \max_i a_{i,k} V_{t-1}^i$$

© Eric Xing @ CMU, 2006-2011

38

Viterbi Vs. Posterior Decoding (individual)



$x = 1, 2, 1, 5, 6, 2, 1, 6, 2, 4$

V_t^k	$ptr(k, t)$	Veterbi	PD	$p(y_t^k = 1 x)$
0.6250 0.3750	N/A	1	1	0.8128 0.1872
0.7353 0.2647	1 2	1	1	0.8238 0.1762
0.8224 0.1776	1 2	1	1	0.8176 0.1824
0.8853 0.1147	1 2	1	1	0.7925 0.2075
0.7200 0.2800	1 2	1	1	0.7415 0.2585
0.8108 0.1892	1 2	1	1	0.7505 0.2495
0.8772 0.1228	1 2	1	1	0.7386 0.2614
0.7043 0.2957	1 2	1	1	0.7027 0.2973
0.7988 0.2012	1 2	1	1	0.7251 0.2749
0.8687 0.1313	1 2	1	1	0.7251 0.2749

© Eric Xing @ CMU, 2006-2011

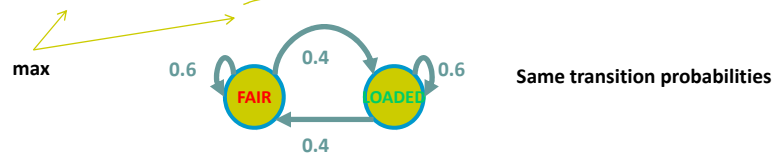
39

Another Example



$x = 6, 2, 3, 5, 6, 2, 6, 3, 6, 6$

V_t^k	$ptr(k, t)$	Veterbi	PD	$p(y_t^k = 1 x)$
0.2500 0.7500	N/A	2	2	0.2733 0.7267
0.5263 0.4737	2 2	1	1	0.6040 0.3960
0.6494 0.3506	1 2	1	1	0.6538 0.3462
0.7143 0.2857	1 1	1	1	0.6062 0.3938
0.3333 0.6667	1 1	2	2	0.2861 0.7139
0.5263 0.4737	2 2	2	1	0.5342 0.4658
0.2703 0.7297	1 2	2	2	0.2734 0.7266
0.5263 0.4737	2 2	2	1	0.5226 0.4774
0.2703 0.7297	1 2	2	2	0.2252 0.7748
0.1818 0.8182	2 2	2	2	0.2159 0.7841



© Eric Xing @ CMU, 2006-2011

40

Computational Complexity and implementation details



- What is the running time, and space required, for Forward, and Backward?

$$\alpha_t^k = p(x_t | y_t^k = 1) \sum_i \alpha_{t-1}^i a_{i,k}$$

$$\beta_t^k = \sum_i a_{k,i} p(x_{t+1} | y_{t+1}^i = 1) \beta_{t+1}^i$$

$$V_t^k = p(x_t | y_t^k = 1) \max_i a_{i,k} V_{t-1}^i$$

Time: $O(K^2M)$;

Space: $O(KM)$.

- Useful implementation technique to avoid underflows
 - Viterbi: sum of logs
 - Forward/Backward: rescaling at each position by multiplying by a constant

Three Main Questions on HMMs



1. Evaluation

GIVEN an HMM M , and a sequence x ,
 FIND Prob ($x | M$)
 ALGO. Forward

2. Decoding

GIVEN an HMM M , and a sequence x ,
 FIND the sequence y of states that maximizes, e.g., $P(y | x, M)$,
 or the most probable subsequence of states
 ALGO. Viterbi, Forward-backward

3. Learning

GIVEN an HMM M , with unspecified transition/emission probs.,
 and a sequence x ,
 FIND parameters $\theta = (\pi_i, a_{ij}, \eta_{ik})$ that maximize $P(x | \theta)$
 ALGO. Baum-Welch (EM)

Learning HMM



- Next Lecture

Summary: the HMM algorithms



Questions:

- **Evaluation:** What is the probability of the observed sequence? **Forward**
- **Decoding:** What is the probability that the state of the 3rd roll is loaded, given the observed sequence? **Forward-Backward**
- **Decoding:** What is the most likely die sequence? **Viterbi**
- **Learning:** Under what parameterization are the observed sequences most probable? **Baum-Welch (EM)**