CSE 150

HW 5 - due **Tues**day

Hidden Markov Models (HMMs)

* Random Variables

  $S_t \in \{1, 2, \ldots, n\}$ states at time t

  $O_t \in \{1, 2, \ldots, m\}$ observation at time t
  "noisy" reflection of hidden state S_t

* Ex: toilet training

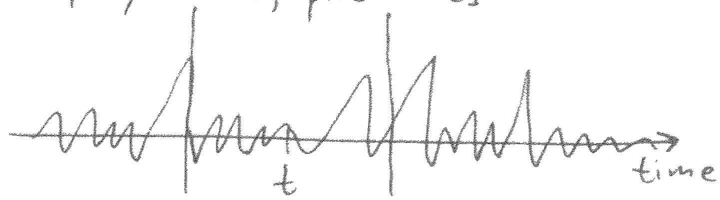
  S = {have-to-go, dont havetogo, went}

  O = {smile, intense concentration, walking funny, squat}

* Ex: speech recognition

  S_t = units of language: words, syllables, phonemes

  O_t = acoustic measurements



* Ex: robotics

  S_t = location

  O_t = sensor readings

* Markov Assumptions

  - finite context

  $$P(S_t \mid S_{t-1}, S_{t-2}, \ldots, S_1) = P(S_t \mid S_{t-1})$$

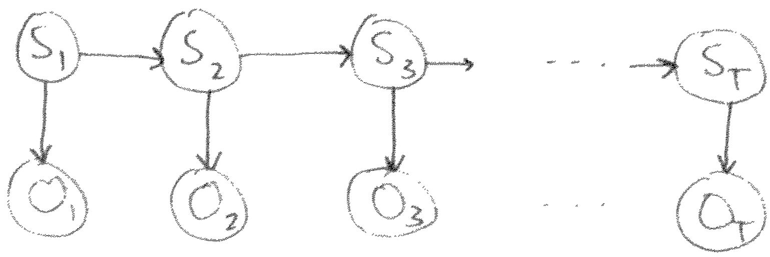
  $$P(O_t \mid S_1, S_2, \ldots, S_{t-1}, S_t, S_{t+1}, \ldots, S_T) = P(O_t \mid S_t)$$

  - shared CPTs

  $$P(S_{t+1} = s' \mid S_t = s) = P(S_t = s' \mid S_{t-1} = s)$$

  $$P(O_t = o \mid S_t = s) = P(O_{t+1} = o \mid S_{t+1} = s)$$

* Belief Network



hidden states

observations

Is this a polytree? <u>Yes</u>

Joint Distribution

$$P(\vec{S}, \vec{O}) = P(S_1) \cdot \left\{ \prod_{t=2}^{T} P(S_t | S_{t-1}) \right\} \cdot \left\{ \prod_{t=1}^{T} P(O_t | S_t) \right\}$$

$\vec{S} = (S_1, S_2, \ldots, S_T)$

$\vec{O} = (O_1, O_2, \ldots, O_T)$

$P(S_1)$: initial state

* Parameters

$\pi_i = P(S_1 = i)$ initial state distribution

$a_{ij} = P(S_{t+1} = j | S_t = i)$ transition matrix $(i = 1, \ldots, n$ & $j = 1, \ldots, n)$

$b_{ik} = P(O_t = k | S_t = i)$ emission matrix $(i = 1, \ldots, n$ & $k = 1, \ldots, m)$

For clarity: $b_{ik} = b_i(k)$ alternate notation

Ex: isolated word speech recognizer

e.g. word = CAT

use HMM with $n = 5$ states, with initial and final states representing silence

$$\pi_i = \begin{array}{cl} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} & \begin{array}{l} \text{initial silence} \\ \text{"C"} \\ \text{"A"} \\ \text{"T"} \\ \text{final silence} \end{array} \end{array} \qquad a_{ij} = \begin{pmatrix} 0.9 & 0.1 & \emptyset & \emptyset & \emptyset \\ \emptyset & 0.9 & 0.1 & \emptyset & \emptyset \\ \emptyset & \emptyset & 0.9 & 0.1 & \emptyset \\ \emptyset & \emptyset & \emptyset & 0.9 & 0.1 \\ \emptyset & \emptyset & \emptyset & \emptyset & 1 \end{pmatrix}$$

left to right HMM

* Key Questions for HMMs

1) How to compute $P(O_1, O_2, \dots, O_T)$ given $\{\pi_i, a_{ij}, b_{ik}\}$

2) How to compute most likely state sequence?

$$(s_1^*, s_2^*, \dots, s_T^*) = \underset{s_1, s_2, \dots, s_T}{\operatorname{argmax}} \left[P(\underbrace{s_1, s_2, \dots, s_T}_{n^T \text{ possible settings}} \mid O_1, O_2, \dots, O_T) \right]$$

sequence of states ~~that maximizes~~
posterior probability

3) How to estimate parameters $\{\pi_i, a_{ij}, b_{ik}\}$ that maximize likelihood $P(O_1, O_2, \dots, O_T)$?

We can use <u>EM</u> algorithm

Of these Questions: ① and ② are inference.
③ is learning.

(1) Computing Likelihood

$$P(O_1, O_2, \dots, O_T) = \sum_{\vec{s}} P(\overbrace{s_1, s_2, \dots, s_T}^{\text{hidden}}, \overbrace{O_1, O_2, \dots, O_T}^{\text{observed}})$$

$\vec{s}$ ← sum over n^T hidden configurations

$$= \sum_{\vec{s}} P(s_1) \left[\prod_{t=2}^{T} P(s_t \mid s_{t-1}) \right] \left[\prod_{t=1}^{T} P(O_t \mid S_t) \right]$$

* Efficient Recursion

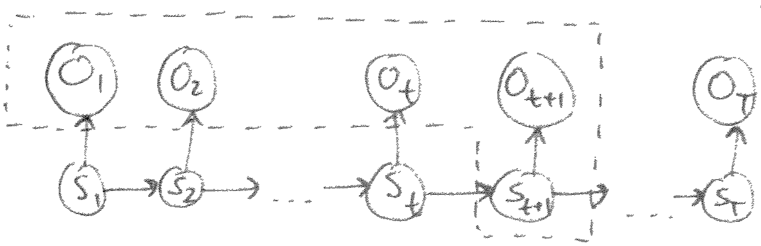
$$P(O_1, O_2, \dots, O_t, O_{t+1}, S_{t+1}=j) = \sum_{i=1}^{n} P(O_1, \dots, O_t, O_{t+1}, S_{t+1}=j, S_t=i) \quad \text{maginalization}$$

$$= \sum_{i=1}^{n} P(O_1, \dots, O_t, S_t=i) \quad \text{product rule}$$

$$\times P(S_{t+1}=j, O_{t+1} \mid S_t=i, O_1, \dots, O_t)$$

$$= \sum_{i=1}^{n} P(O_1, \dots, O_t, S_t=i) P(S_{t+1}=j, O_{t+1} \mid S_t=i)$$

conditional independence



$$= \sum_{i=1}^{n} P(O_1, O_2, \ldots, O_t, S_t = i)\, P(S_{t+1} = j \mid S_t = i)\, P(O_{t+1} \mid S_{t+1} = j, S_t = i)$$ product rule

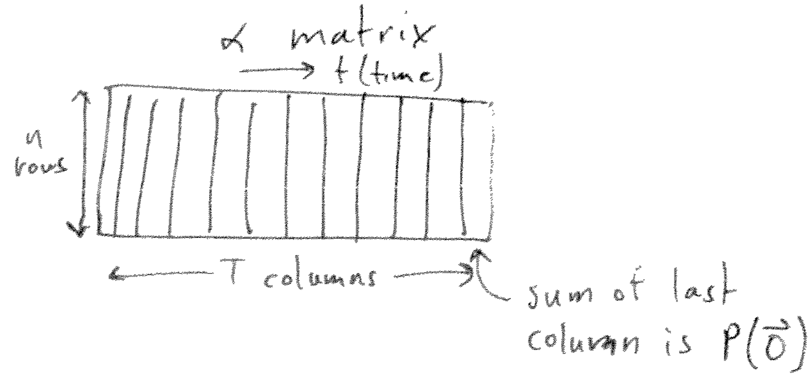
$$= \sum_{i=1}^{n} \underbrace{P(O_1, O_2, \ldots, O_t, S_t = i)}_{\text{recursive instance}} \underbrace{P(S_{t+1} = j \mid S_t = i)\, P(O_{t+1} \mid S_{t+1} = j)}_{\text{CPTs}}$$ conditional independence

* short hand notation

$$\alpha_{it} = P(O_1, O_2, \ldots, O_t, S_t = i)$$

$i = 1, \ldots, n$ # hidden states

$t = 1, \ldots, T$ sequence length



* forward Algorithm
  - recursive step

$$\alpha_{j\,t+1} = \sum_{i=1}^{n} \alpha_{it}\, a_{ij}\, b_j(O_{t+1})$$

* initial condition (column of α)

$$\alpha_{i1} = P(O_1, S_1 = i) = P(S_1 = i)\, P(O_1 \mid S_1 = i) = \pi_i\, b_i(O_1) \quad \text{for } i = 1, \ldots, n$$

* Back to likelihood Computation

$$P(O_1, O_2, \ldots, O_T) = \sum_{i=1}^{n} P(O_1, O_2, \ldots, O_T, S_T = i) \quad \text{marginalization}$$

$$= \sum_{i=1}^{n} \alpha_{iT}$$

* Scales as $O(n^2 T)$

  linear, not exponential, in sequence length
  quadratic in # of states

* Warning: naive calculations will underflow for long sequences because $P(O_1, O_2, \ldots, O_T) \ll 1$