Hidden Markov Models

CPSC 322 – Uncertainty 7

Textbook §10.5

Lecture Overview

1. Recap
2. Hidden Markov Models

Variable elimination algorithm

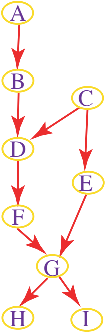
To compute $P(Q|Y_1 = v_1 \wedge \ldots \wedge Y_j = v_j)$:

- Construct a factor for each conditional probability.
- Set the observed variables to their observed values.
- For each of the other variables $Z_i \in \{Z_1, \ldots, Z_k\}$, sum out Z_i
- Multiply the remaining factors.
- Normalize by dividing the resulting factor $f(Q)$ by $\sum_Q f(Q)$.

Variable elimination example

Compute $P(G|H = h_1)$. Elimination order: A, C, E, H, I, B, D, F

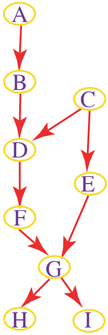
- $P(G, H) = \sum_{A,B,C,D,E,F,I} P(A, B, C, D, E, F, G, H, I)$
- $P(G, H) = \sum_{A,B,C,D,E,F,I} P(A) \cdot P(B|A) \cdot P(C) \cdot P(D|B, C) \cdot P(E|C) \cdot P(F|D) \cdot P(G|F, E) \cdot P(H|G) \cdot P(I|G)$



Variable elimination example

Compute $P(G|H=h_1)$. Elimination order: A, C, E, H, I, B, D, F

- $P(G,H) = \sum_{A,B,C,D,E,F,I} P(A) \cdot P(B|A) \cdot P(C) \cdot P(D|B,C) \cdot P(E|C) \cdot P(F|D) \cdot P(G|F,E) \cdot P(H|G) \cdot P(I|G)$
- Eliminate A: $P(G,H) = \sum_{B,C,D,E,F,I} f_1(B) \cdot P(C) \cdot P(D|B,C) \cdot P(E|C) \cdot P(F|D) \cdot P(G|F,E) \cdot P(H|G) \cdot P(I|G)$

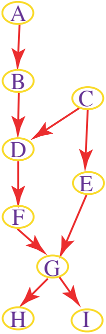


- $f_1(B) := \sum_{a \in dom(A)} P(A=a) \cdot P(B|A=a)$

Variable elimination example

Compute $P(G|H = h_1)$. Elimination order: A, C, E, H, I, B, D, F

- $P(G, H) = \sum_{B,C,D,E,F,I} f_1(B) \cdot P(C) \cdot P(D|B, C) \cdot P(E|C) \cdot P(F|D) \cdot P(G|F, E) \cdot P(H|G) \cdot P(I|G)$
- Eliminate C: $P(G, H) =$
  $\sum_{B,D,E,F,I} f_1(B) \cdot f_2(B, D, E) \cdot P(F|D) \cdot P(G|F, E) \cdot P(H|G) \cdot P(I|G)$



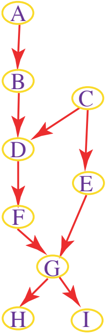
- $f_1(B) := \sum_{a \in dom(A)} P(A = a) \cdot P(B|A = a)$
- $f_2(B, D, E) := \sum_{c \in dom(C)} P(C = c) \cdot P(D|B, C = c) \cdot P(E|C = c)$

Variable elimination example

Compute $P(G|H=h_1)$. Elimination order: A, C, E, H, I, B, D, F

- $P(G,H) = \sum_{B,D,E,F,I} f_1(B) \cdot f_2(B,D,E) \cdot P(F|D) \cdot P(G|F,E) \cdot P(H|G) \cdot P(I|G)$
- Eliminate E:

  $P(G,H) = \sum_{B,D,F,I} f_1(B) \cdot f_3(B,D,F,G) \cdot P(F|D) \cdot P(H|G) \cdot P(I|G)$



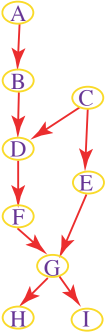
- $f_1(B) := \sum_{a \in dom(A)} P(A=a) \cdot P(B|A=a)$
- $f_2(B,D,E) := \sum_{c \in dom(C)} P(C=c) \cdot P(D|B,C=c) \cdot P(E|C=c)$
- $f_3(B,D,F,G) := \sum_{e \in dom(E)} f_2(B,D,E=e) \cdot P(G|F,E=e)$

Variable elimination example

Compute $P(G|H=h_1)$. Elimination order: A, C, E, H, I, B, D, F

- $P(G,H) = \sum_{B,D,F,I} f_1(B) \cdot f_3(B,D,F,G) \cdot P(F|D) \cdot P(H|G) \cdot P(I|G)$
- Observe $H=h_1$:

  $P(G,H=h_1) = \sum_{B,D,F,I} f_1(B) \cdot f_3(B,D,F,G) \cdot P(F|D) \cdot f_4(G) \cdot P(I|G)$



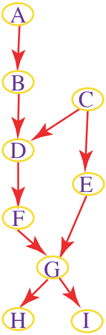
- $f_1(B) := \sum_{a \in dom(A)} P(A=a) \cdot P(B|A=a)$
- $f_2(B,D,E) := \sum_{c \in dom(C)} P(C=c) \cdot P(D|B,C=c) \cdot P(E|C=c)$
- $f_3(B,D,F,G) := \sum_{e \in dom(E)} f_2(B,D,E=e) \cdot P(G|F,E=e)$
- $f_4(G) := P(H=h_1|G)$

Variable elimination example

Compute $P(G|H=h_1)$. Elimination order: A, C, E, H, I, B, D, F

- $P(G, H=h_1) = \sum_{B,D,F,I} f_1(B) \cdot f_3(B,D,F,G) \cdot P(F|D) \cdot f_4(G) \cdot P(I|G)$
- Eliminate I:

  $P(G, H=h_1) = \sum_{B,D,F} f_1(B) \cdot f_3(B,D,F,G) \cdot P(F|D) \cdot f_4(G) \cdot f_5(G)$

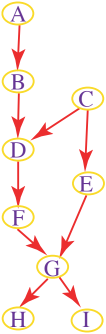


- $f_1(B) := \sum_{a \in dom(A)} P(A=a) \cdot P(B|A=a)$
- $f_2(B,D,E) := \sum_{c \in dom(C)} P(C=c) \cdot P(D|B,C=c) \cdot P(E|C=c)$
- $f_3(B,D,F,G) := \sum_{e \in dom(E)} f_2(B,D,E=e) \cdot P(G|F,E=e)$
- $f_4(G) := P(H=h_1|G)$
- $f_5(G) := \sum_{i \in dom(I)} P(I=i|G)$

Variable elimination example

Compute $P(G|H = h_1)$. Elimination order: A, C, E, H, I, B, D, F

- $P(G, H = h_1) = \sum_{B,D,F} f_1(B) \cdot f_3(B, D, F, G) \cdot P(F|D) \cdot f_4(G) \cdot f_5(G)$
- Eliminate B:
  $P(G, H = h_1) = \sum_{D,F} f_6(D, F, G) \cdot P(F|D) \cdot f_4(G) \cdot f_5(G)$

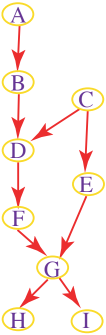


- $f_1(B) := \sum_{a \in dom(A)} P(A = a) \cdot P(B|A = a)$
- $f_2(B, D, E) := \sum_{c \in dom(C)} P(C = c) \cdot P(D|B, C = c) \cdot P(E|C = c)$
- $f_3(B, D, F, G) := \sum_{e \in dom(E)} f_2(B, D, E = e) \cdot P(G|F, E = e)$
- $f_4(G) := P(H = h_1|G)$
- $f_5(G) := \sum_{i \in dom(I)} P(I = i|G)$
- $f_6(D, F, G) := \sum_{b \in dom(B)} f_1(B = b) \cdot f_3(B = b, D, F, G)$

Variable elimination example

Compute $P(G|H=h_1)$. Elimination order: A, C, E, H, I, B, D, F

- $P(G, H=h_1) = \sum_{D,F} f_6(D,F,G) \cdot P(F|D) \cdot f_4(G) \cdot f_5(G)$
- Eliminate D: $P(G, H=h_1) = \sum_{F} f_7(F,G) \cdot f_4(G) \cdot f_5(G)$

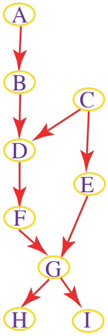


- $f_1(B) := \sum_{a \in dom(A)} P(A=a) \cdot P(B|A=a)$
- $f_2(B,D,E) := \sum_{c \in dom(C)} P(C=c) \cdot P(D|B,C=c) \cdot P(E|C=c)$
- $f_3(B,D,F,G) := \sum_{e \in dom(E)} f_2(B,D,E=e) \cdot P(G|F,E=e)$
- $f_4(G) := P(H=h_1|G)$
- $f_5(G) := \sum_{i \in dom(I)} P(I=i|G)$
- $f_6(D,F,G) := \sum_{b \in dom(B)} f_1(B=b) \cdot f_3(B=b,D,F,G)$
- $f_7(F,G) := \sum_{d \in dom(D)} f_6(D=d,F,G) \cdot P(F|D=d)$

Variable elimination example

Compute $P(G|H = h_1)$. Elimination order: A, C, E, H, I, B, D, F

- $P(G, H = h_1) = \sum_F f_7(F, G) \cdot f_4(G) \cdot f_5(G)$
- Eliminate F: $P(G, H = h_1) = f_8(G) \cdot f_4(G) \cdot f_5(G)$

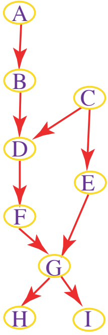


- $f_1(B) := \sum_{a \in dom(A)} P(A = a) \cdot P(B|A = a)$
- $f_2(B, D, E) := \sum_{c \in dom(C)} P(C = c) \cdot P(D|B, C = c) \cdot P(E|C = c)$
- $f_3(B, D, F, G) := \sum_{e \in dom(E)} f_2(B, D, E = e) \cdot P(G|F, E = e)$
- $f_4(G) := P(H = h_1|G)$
- $f_5(G) := \sum_{i \in dom(I)} P(I = i|G)$
- $f_6(D, F, G) := \sum_{b \in dom(B)} f_1(B = b) \cdot f_3(B = b, D, F, G)$
- $f_7(F, G) := \sum_{d \in dom(D)} f_6(D = d, F, G) \cdot P(F|D = d)$
- $f_8(G) := \sum_{f \in dom(F)} f_7(F = f, G)$

Variable elimination example

Compute $P(G|H = h_1)$. Elimination order: A, C, E, H, I, B, D, F

- $P(G, H = h_1) = f_8(G) \cdot f_4(G) \cdot f_5(G)$
- Normalize: $P(G|H = h_1) = \frac{P(G, H = h_1)}{\sum_{g \in dom(G)} P(G, H = h_1)}$



- $f_1(B) := \sum_{a \in dom(A)} P(A = a) \cdot P(B|A = a)$
- $f_2(B, D, E) := \sum_{c \in dom(C)} P(C = c) \cdot P(D|B, C = c) \cdot P(E|C = c)$
- $f_3(B, D, F, G) := \sum_{e \in dom(E)} f_2(B, D, E = e) \cdot P(G|F, E = e)$
- $f_4(G) := P(H = h_1|G)$
- $f_5(G) := \sum_{i \in dom(I)} P(I = i|G)$
- $f_6(D, F, G) := \sum_{b \in dom(B)} f_1(B = b) \cdot f_3(B = b, D, F, G)$
- $f_7(F, G) := \sum_{d \in dom(D)} f_6(D = d, F, G) \cdot P(F|D = d)$
- $f_8(G) := \sum_{f \in dom(F)} f_7(F = f, G)$

What good was Conditional Independence?

- That's great... but it looks incredibly painful for large graphs.
- And... why did we bother learning conditional independence? Does it help us at all?

What good was Conditional Independence?

- That's great. . . but it looks incredibly painful for large graphs.
- And. . . why did we bother learning conditional independence? Does it help us at all?
    - yes—we use the chain rule decomposition right at the beginning

What good was Conditional Independence?

- That's great... but it looks incredibly painful for large graphs.
- And... why did we bother learning conditional independence? Does it help us at all?
  - yes—we use the chain rule decomposition right at the beginning
- Can we use our knowledge of conditional independence to make this calculation even simpler?

What good was Conditional Independence?

- That's great... but it looks incredibly painful for large graphs.
- And... why did we bother learning conditional independence? Does it help us at all?
  - yes—we use the chain rule decomposition right at the beginning

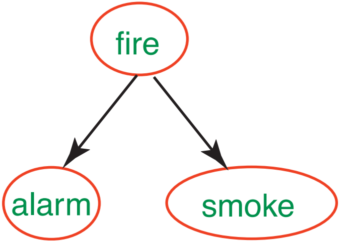
- Can we use our knowledge of conditional independence to make this calculation even simpler?
  - yes—there are some variables that we don't have to sum out
  - intuitively, they're the ones that are "pre-summed-out" in our tables
  - example: summing out I on the previous slide

One Last Trick

One last trick to simplify calculations: we can repeatedly eliminate all leaf nodes that are neither observed nor queried, until we reach a fixed point.

One Last Trick

One last trick to simplify calculations: we can repeatedly eliminate all leaf nodes that are neither observed nor queried, until we reach a fixed point.

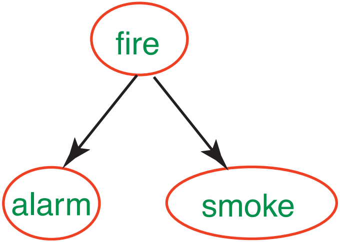


Can we justify that on a three-node graph—Fire, Alarm, and Smoke—when we ask for:

- $P(Fire)$?

One Last Trick

One last trick to simplify calculations: we can repeatedly eliminate all leaf nodes that are neither observed nor queried, until we reach a fixed point.



Can we justify that on a three-node graph—Fire, Alarm, and Smoke—when we ask for:

- $P(Fire)$?
- $P(Fire \mid Alarm)$?

Lecture Overview

1. Recap
2. Hidden Markov Models

Markov chain

- A Markov chain is a special sort of belief network:

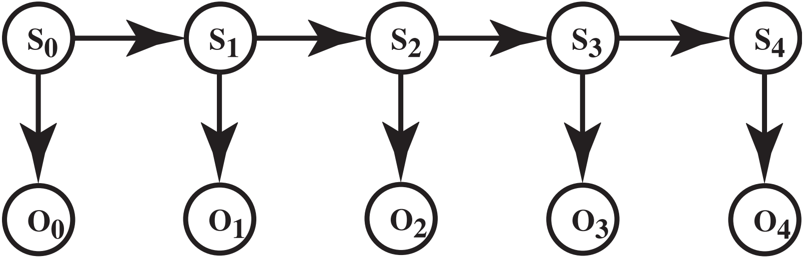
- Thus $P(S_{t+1}|S_0, \ldots, S_t) = P(S_{t+1}|S_t)$.
- Often S_t represents the state at time t. Intuitively S_t conveys all of the information about the history that can affect the future states.
- "The past is independent of the future given the present."

Stationary Markov chain

- A stationary Markov chain is when for all $t > 0$, $t' > 0$, $P(S_{t+1}|S_t) = P(S_{t'+1}|S_{t'})$.
- We specify $P(S_0)$ and $P(S_{t+1}|S_t)$.
  - Simple model, easy to specify
  - Often the natural model
  - The network can extend indefinitely

Hidden Markov Model

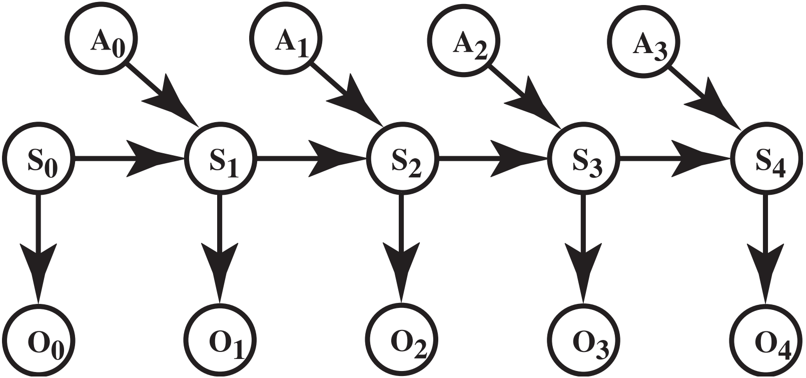
- A Hidden Markov Model (HMM) starts with a Markov chain, and adds a noisy observation about the state at each time step:



- $P(S_0)$ specifies initial conditions
- $P(S_{t+1}|S_t)$ specifies the dynamics
- $P(O_t|S_t)$ specifies the sensor model

Example: localization

- Suppose a robot wants to determine its location based on its actions and its sensor readings: Localization
- This can be represented by the augmented HMM:



Example localization domain

- Circular corridor, with 16 locations:

- Doors at positions: 2, 4, 7, 11.
- Noisy Sensors
- Stochastic Dynamics
- Robot starts at an unknown location and must determine where it is.

Example Sensor Model

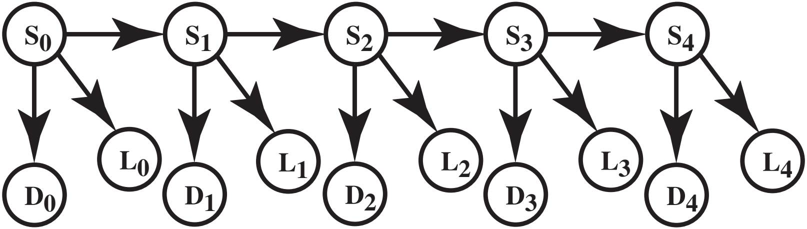
- $P(Observe\ Door \mid At\ Door) = 0.8$
- $P(Observe\ Door \mid Not\ At\ Door) = 0.1$

Example Dynamics Model

- $P(loc_{t+1} = L | action_t = goRight \wedge loc_t = L) = 0.1$
- $P(loc_{t+1} = L + 1 | action_t = goRight \wedge loc_t = L) = 0.8$
- $P(loc_{t+1} = L + 2 | action_t = goRight \wedge loc_t = L) = 0.074$
- $P(loc_{t+1} = L' | action_t = goRight \wedge loc_t = L) = 0.002$ for any other location L'.
  - All location arithmetic is modulo 16.
  - The action $goLeft$ works the same but to the left.

Combining sensor information

- Example: we can combine information from a light sensor and the door sensor: "Sensor Fusion"



- S_t: robot location at time t
- D_t: door sensor value at time t
- L_t: light sensor value at time t

Localization demo

- http://www.cs.ubc.ca/spider/poole/demos/localization/localization.html