Probability and Time:
Hidden Markov Models (HMMs)

Computer Science cpsc322, Lecture 32

(Textbook Chpt 6.5.2)

Nov, 25, 2013
Lecture Overview

• Recap
• Markov Models
  • Markov Chain
• Hidden Markov Models
Answering Queries under Uncertainty

Probability Theory

Static Belief Network & Variable Elimination

Dynamic Bayesian Network

Hidden Markov Models

Student Tracing in tutoring Systems

Biological Informatics

Robotics

Markov Chains

Natural Language Processing

Monitoring (e.g., credit cards)

Diagnostic Systems (e.g., medicine)

Email spam filters

Some Applications you will know

Some Applications you will know a little
A stationary Markov Chain: for all $t > 0$

- $P(S_{t+1} | S_0, \ldots, S_t) = P(S_{t+1} | S_t)$ and

- $P(S_{t+1} | S_t)$ the same $\forall t$

We only need to specify $P(S_0)^K$ and $P(S_{t+1} | S_t)$

- Simple Model, easy to specify
- Often the natural model
- The network can extend indefinitely
- Variations of SMC are at the core of most Natural Language Processing (NLP) applications!
Lecture Overview

• Recap

• Markov Models
  • Markov Chain
  • Hidden Markov Models
How can we minimally extend Markov Chains?

- Maintaining the Markov and stationary assumptions?

A useful situation to model is the one in which:

- the reasoning system does not have access to the states
- but can make observations that give some information about the current state
A. $2 \times h$
B. $h \times h$
C. $k \times h$
D. $k \times k$
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

|domain(S)| = \( k \)

|domain(O)| = \( h \)

A. \( 2 \times h \)
B. \( h \times h \)
C. \( k \times h \)
D. \( k \times k \)
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

- $|\text{domain}(S)| = k$
- $|\text{domain}(O)| = h$
Example: Localization for “Pushed around” Robot

- **Localization** (where am I?) is a fundamental problem in robotics
- Suppose a robot is in a circular corridor with 16 locations
- There are **four doors** at positions: 2, 4, 7, 11
- The Robot initially doesn’t know where it is
- The Robot is **pushed around**. After a push it can stay in the same location, move left or right.
- The Robot has a **Noisy sensor** telling whether it is in front of a door
This scenario can be represented as...

- **Example Stochastic Dynamics**: when pushed, it stays in the same location \( p = 0.2 \), moves one step left or right with equal probability.

\[
P(Loc_{t+1} \mid Loc_t)
\]

\( Loc_t = 10 \)
This scenario can be represented as...

- Example Stochastic Dynamics: when pushed, it stays in the same location $p=0.2$, moves left or right with equal probability.

$$P(Loc_{t+1} \mid Loc_t) = \frac{1}{2}$$

$$P(Loc_1) = \frac{1}{16} \frac{1}{16} \frac{1}{16}$$
This scenario can be represented as...

Example of Noisy sensor telling whether it is in front of a door.

- If it is in front of a door, \( P(O_t = T) = .8 \)
- If not in front of a door, \( P(O_t = T) = .1 \)

\[ \text{dom}(\text{Loc}_i) = \{0, 1, \ldots, 15\} \]

\[ P(O_t / \text{Loc}_i) \]

Wrong! \[ P(O_t = T) P(O_t = F) \]

16 probability distributions...
Useful inference in HMMs

- **Localization**: Robot starts at an unknown location and it is pushed around $t$ times. It wants to determine where it is:

  $$P(Loc_t | O_1 \ldots O_t)$$

- **In general**: compute the posterior distribution over the current state given all evidence to date

  $$P(S_t | O_0 \ldots O_t)$$
Example : Robot Localization

- Suppose a robot wants to determine its location based on its actions and its sensor readings
- Three actions: *goRight, goLeft, Stay*
- This can be represented by an augmented HMM
Robot Localization Sensor and Dynamics Model

- **Sample Sensor Model** (assume same as for pushed around)
- **Sample Stochastic Dynamics:**
  \[
  P(Loc_{t+1} | \text{Action}_t, Loc_t)
  \]
  \[
  P(Loc_{t+1} = L | \text{Action}_t = \text{goRight} , Loc_t = L) = 0.1
  \]
  \[
  P(Loc_{t+1} = L + 1 | \text{Action}_t = \text{goRight} , Loc_t = L) = 0.8
  \]
  \[
  P(Loc_{t+1} = L + 2 | \text{Action}_t = \text{goRight} , Loc_t = L) = 0.074
  \]
  \[
  P(Loc_{t+1} = L' | \text{Action}_t = \text{goRight} , Loc_t = L) = 0.002 \text{ for all other locations } L'
  \]

- All location arithmetic is modulo 16
- The action \textit{goLeft} works the same but to the left
Sample Stochastic Dynamics: \( P(L_{t+1} | \text{Action}_t, L_{t}) \)

\[
P(L_{t+1} = L | \text{Action}_t = \text{goRight} , L_{t} = L) = 0.1
\]

\[
P(L_{t+1} = L+1 | \text{Action}_t = \text{goRight} , L_{t} = L) = 0.8
\]

\[
P(L_{t+1} = L + 2 | \text{Action}_t = \text{goRight} , L_{t} = L) = 0.074
\]

\[
P(L_{t+1} = L' | \text{Action}_t = \text{goRight} , L_{t} = L) = 0.002 \text{ for all other locations } L'
\]
Robot Localization additional sensor

- **Additional Light Sensor**: there is light coming through an opening at location 10
  \[ P (L_t | Loc_i) \]

- Info from the two sensors is combined: "Sensor Fusion"
The Robot starts at an unknown location and must determine where it is

The model appears to be too ambiguous

- Sensors are too noisy
- Dynamics are too stochastic to infer anything

But inference actually works pretty well.

Let’s check:


You can use standard Bnet inference. However you typically take advantage of the fact that time moves forward (not in 322)
Sample scenario to explore in demo

• Keep making observations without moving. What happens?
• Then keep moving without making observations. What happens?
• Assume you are at a certain position alternate moves and observations
• ....
HMMs have many other applications….

Natural Language Processing: e.g., Speech Recognition
- **States:** phoneme \ word
- **Observations:** acoustic signal \ phoneme

Bioinformatics: Gene Finding
- **States:** coding / non-coding region
- **Observations:** DNA Sequences  \ ATC66AA

For these problems the critical inference is:
find the most likely sequence of states given a sequence of observations

\[ \text{Viterbi Algorithm} \]
Markov Models

Markov Chains

Simplest Possible Dynamic Bnet

Hidden Markov Model

Add noisy Observations about the state at time $t$

Add Actions and Values (Rewards)

Markov Decision Processes (MDPs)

PO MDP

Partially Observable
Learning Goals for today’s class

You can:

• Specify the components of an Hidden Markov Model (HMM)
• Justify and apply HMMs to Robot Localization

Clarification on second LG for last class

You can:

• Justify and apply Markov Chains to compute the probability of a Natural Language sentence (NOT to estimate the conditional probs- slide 18)
Next week

<table>
<thead>
<tr>
<th>Problem</th>
<th>Static</th>
<th>Constraint Satisfaction</th>
<th>Query</th>
<th>Sequential</th>
<th>Planning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deterministic</td>
<td>Arc Consistency</td>
<td>Vars + Constraints</td>
<td>Search</td>
<td>STRIPS</td>
<td>Search</td>
</tr>
<tr>
<td>Stochastic</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Environment**
  - Deterministic
    - Arc Consistency
    - Search
  - Stochastic
    - Belief Nets
    - Var. Elimination
    - Markov Chains and HMMs
    - Decision Nets
    - Markov Decision Processes
    - Value Iteration

**Representation**
- Reasoning
- Technique
Next Class

• One-off decisions (*TextBook 9.2*)
• Single Stage Decision networks (* 9.2.1*)
People

Instructor

• Giuseppe Carenini (carenini@cs.ubc.ca; office CICS 105)

Teaching Assistants

• Kamyar Ardekani  kamyar.ardekany@gmail.com

• Tatsuro Oya   toya@cs.ubc.ca

• Xin Ru (Nancy) Wang  nancywang1991@yahoo.ca