CSE 573: Artificial Intelligence
Autumn 2012

Reasoning about Uncertainty
& Hidden Markov Models

Daniel Weld

Many slides adapted from Dan Klein, Stuart Russell, Andrew Moore & Luke Zettlemoyer

Outline

- Overview
- Probability review
  - Random Variables and Events
  - Joint / Marginal / Conditional Distributions
  - Product Rule, Chain Rule, Bayes' Rule
- Probabilistic inference
  - Enumeration of Joint Distribution
  - Bayesian Networks – Preview
- Probabilistic sequence models (and inference)
  - Markov Chains
  - Hidden Markov Models
  - Particle Filters

Agent

Static vs. Dynamic

Environment

Fully vs. Partially Observable

Perfect vs. Noisy

What action next?

Percepts Actions

Partial Observability

Deterministic vs. Stochastic

Instantaneous vs. Durative

Markov Decision Process (MDP)

<table>
<thead>
<tr>
<th>S: set of states</th>
<th>A: set of actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pr(s'</td>
<td>s,a): transition model</td>
</tr>
<tr>
<td>γ: discount factor</td>
<td>s0: start state</td>
</tr>
</tbody>
</table>

Objective of a Fully Observable MDP

- Find a policy π: S → A
- which maximizes expected discounted reward
  - given an infinite horizon
  - assuming full observability
Partially-Observable MDP

- **S**: set of states
- **A**: set of actions
- **Pr(s'|s,a)**: transition model
- **R(s,a,s')**: reward model
- **γ**: discount factor
- **s₀**: start state
- **E**: set of possible pieces of evidence
- **Pr(e|s)**: observation model

**Ghostbusters Observations**

- A ghost is in the grid somewhere
- Model 1: Sensor readings tell distance to ghost:
  - On top of pacman: red
  - 1 or 2 away: orange
  - 3 or 4 away: yellow
  - 5+ away: green

  \[
P(\text{red} | 3) \quad P(\text{orange} | 3) \quad P(\text{yellow} | 3) \quad P(\text{green} | 3)\]

  \[
  0.05 \quad 0.15 \quad 0.5 \quad 0.3
  \]

- Model 2: Sensors are noisy, but we know \(P\text{(Color} \mid \text{Distance)}\)

**Objective of a POMDP**

- Find a policy \(\pi : \text{BeliefStates}(S) \rightarrow A\)
  - A belief state is a **probability distribution** over states

- which maximizes expected discounted reward
  - given an infinite horizon
  - assuming full observability

**Particle Filtering**

**Planning in HW 4**

- Map Estimate

**Projects**

- You choose...
- Default 1
  - Extend Pacman reinforcement learning, eg UCT
- Default 2
  - Extend Pacman to real POMDP
Random Variables

- A random variable is some aspect of the world about which we (may) have uncertainty
  - R = is it raining?
  - D = How long will it take to drive to work?
  - L = Where am I?

- We denote random variables with capital letters

Random variables have domains
  - R in \{true, false\}
  - D in \([0, 1)\)
  - L in possible locations, maybe \((0,0), (0,1), \ldots\)

Joint Distributions

- A joint distribution over a set of random variables: \(X_1, X_2, \ldots, X_n\) specifies a real number for each outcome (i.e., each assignment):
  \[
P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n)
  \]
  \[
P(x_1, x_2, \ldots, x_n)
  \]
- Must obey:
  \[
P(x_1, x_2, \ldots, x_n) \geq 0
  \]
  \[
  \sum_{(x_1, x_2, \ldots, x_n)} P(x_1, x_2, \ldots, x_n) = 1
  \]
- Size of distribution if \(n\) variables with domain sizes \(d\)?
- A probabilistic model is a joint distribution over variables of interest
- For all but the smallest distributions, impractical to write out

Terminology

- Joint Probability
  \[
p(X = x, Y = y) = \frac{n_{ij}}{N}
  \]
- Marginal Probability
  \[
p(X = x_i) = \frac{c_i}{N}
  \]
- Conditional Probability
  \[
p(Y = y_i | X = x_i) = \frac{n_{ij}}{c_i}
  \]

Independence

- Conditional Independence
  \[
P(A \land B) = P(A)P(B)
  \]

A, B Conditionally Independent Given C

- \(P(A \land B | C) = P(A | C)\)
  \[
P(A \land C) = \frac{.25 + .5}{2} = .375
  \]
  \[
P(B) = .75
  \]
  \[
P(A \land B | C) = \frac{.25 + .25 + .5}{3} = .3333
  \]

- \(P(A \land \neg C) = .5\)
  \[
P(A \land B \land \neg C) = .5
  \]
  \[
P(A | \neg C) = .5
  \]
  \[
P(A | B, \neg C) = .5
  \]
Probabilistic Inference

- **Probabilistic inference**: compute a desired probability from other known probabilities (e.g., conditional from joint)
- We generally compute conditional probabilities
  - \( P(\text{on time} | \text{no reported accidents}) = 0.90 \)
  - These represent the agent’s beliefs given the evidence
- Probabilities change with new evidence:
  - \( P(\text{on time} | \text{no accidents, 5 a.m.}) = 0.95 \)
  - \( P(\text{on time} | \text{no accidents, 5 a.m., raining}) = 0.80 \)
  - Observing new evidence causes beliefs to be updated

Inference by Enumeration

- General case:
  - Evidence variables: \( E_1 \ldots E_k = c_1 \ldots c_k \)
  - Query* variable: \( Q \)
  - Hidden variables: \( H_1 \ldots H_r \)
  - We want: \( P(Q | e_1 \ldots e_k) \)
  - First, select the entries consistent with the evidence
  - Second, sum out \( H \) to get joint of Query and evidence:
    \[
    P(Q, e_1 \ldots e_k) = \sum_{h_1, \ldots, h_r} P(Q, h_1 \ldots h_r, e_1 \ldots e_k) / P(h_1 \ldots h_r)
    \]
  - Finally, normalize the remaining entries to conditionalize
- Obvious problems:
  - Worst-case time complexity \( O(d^n) \)
  - Space complexity \( O(d^n) \) to store the joint distribution

Bayes’ Nets: Big Picture

- Two problems with using full joint distribution tables as our probabilistic models:
  - Unless there are only a few variables, the joint is WAY too big to represent explicitly
  - Hard to learn (estimate) anything empirically about more than a few variables at a time
- **Bayes’ nets**: a technique for describing complex joint distributions (models) using simple, local distributions (conditional probabilities)
  - More properly called graphical models
  - We describe how variables locally interact
  - Local interactions chain together to give global, indirect interactions

Bayes’ Net Semantics

- Let’s formalize the semantics of a Bayes’ net
  - A set of nodes, one per variable \( X \)
  - A directed, acyclic graph
  - A conditional distribution for each node
    - A collection of distributions over \( X \), one for each combination of parents’ values
    \[
    P(X | A_1 \ldots A_n)
    \]
  - CPT: conditional probability table

A Bayes net = Topology (graph) + Local Conditional Probabilities

Example Bayes’ Net: Car

[Diagram of a Bayes’ net for a car with various nodes and edges]
The Chain Rule

More generally, can always write any joint distribution as an incremental product of conditional distributions:

\[ P(x_1, x_2, x_3) = P(x_1)P(x_2|x_1)P(x_3|x_1, x_2) \]

\[ P(x_1, x_2, \ldots, x_n) = \prod_i P(x_i|x_1 \ldots x_{i-1}) \]

Bayes' Rule

Two ways to factor a joint distribution over two variables:

\[ P(x, y) = P(x|y)P(y) = P(y|x)P(x) \]

Dividing, we get:

\[ P(x|y) = \frac{P(y|x)P(x)}{P(y)} \]

Why is this at all helpful?

- Lets us build a conditional from its reverse
- Often one conditional is tricky but the other one is simple
- Foundation of many systems we’ll see later
- In the running for most important AI equation!

Inference with Bayes’ Rule

Example: Diagnostic probability from causal probability:

\[ P(\text{Cause}|\text{Effect}) = \frac{P(\text{Effect}|\text{Cause})P(\text{Cause})}{P(\text{Effect})} \]

Example:

- m is meningitis, s is stiff neck
- \( P(s|m) = 0.8 \)
- \( P(m) = 0.0001 \)
- \( P(s) = 0.1 \)

\[ P(m|s) = \frac{P(s|m)P(m)}{P(s)} = \frac{0.8 \times 0.0001}{0.1} = 0.0008 \]

Ghostbusters, Revisited

Let’s say we have two distributions:

- Prior distribution over ghost location: \( P(G) \)
- Let’s say this is uniform
- Sensor reading model: \( P(R|G) \)
- Given: we know what our sensors do
- \( R \) = reading color measured at \((1,1)\)
- E.g. \( P(R = \text{yellow} | G=(1,1)) = 0.1 \)

We can calculate the posterior distribution \( P(G|r) \) over ghost locations given a reading using Bayes’ rule:

\[ P(y|r) \propto P(r|y)P(y) \]

Markov Models (Markov Chains)

A Markov model includes:
- Random variables \( X_t \) for all time steps \( t \) (the state)
- Parameters: called transition probabilities or dynamics, specify how the state evolves over time (also, initial probs)

\[ P(X_1) \text{ and } P(X_t|X_{t-1}) \]

Later we’ll see that a Markov model is just a chain-structured Bayesian Network (BN)

Conditional Independence

Basic conditional independence:
- Each time step only depends on the previous
- Future conditionally independent of past given the present
- This is called the (first order) Markov property

This chain is just a (growing) BN
- We could use generic BN reasoning on it if we truncate the chain at a fixed length
Markov Models (Markov Chains)

- A Markov model defines
  - a joint probability distribution:
    \[ P(X_1, \ldots, X_n) = P(X_1) \prod_{t=2}^{N} P(X_t|X_{t-1}) \]
  - One common inference problem:
    - Compute marginals \( P(X_t) \) for some time step, \( t \)

Example: Markov Chain

- Weather:
  - States = \{rain, sun\}
  - Transitions:
    - Initial distribution: 1.0 sun
    - What's the probability distribution after one step?
      \[ P(X_2 = \text{sun}) = P(X_2 = \text{sun}|X_1 = \text{sun})P(X_1 = \text{sun}) + P(X_2 = \text{sun}|X_1 = \text{rain})P(X_1 = \text{rain}) \]
      \[ = 0.9 \cdot 1.0 + 0.1 \cdot 0.0 = 0.9 \]

Mini-Forward Algorithm

- Question: What's \( P(X) \) on some day \( t \)?
  - We don't need to enumerate all \( 2^t \) sequences!
  \[
P(x_t) = \sum_{x_{t-1}} P(x_t|x_{t-1})P(x_{t-1})
\]
  \[ P(x_1) = \text{known} \]

Example

- From initial observation of sun
  \[
  \begin{bmatrix}
  0.9 & 0.02 & 0.5 \\
  0.1 & 0.9 & 0.5 \\
  \end{bmatrix}
  \]

- From initial observation of rain
  \[
  \begin{bmatrix}
  0.0 & 0.1 & 0.9 \\
  0.1 & 0.9 & 0.5 \\
  \end{bmatrix}
  \]
Stationary Distributions

- If we simulate the chain long enough:
  - What happens?
  - Uncertainty accumulates
  - Eventually, we have no idea what the state is!

- Stationary distributions:
  - For most chains, the distribution we end up in is independent of the initial distribution
  - Called the stationary distribution of the chain
  - Usually, can only predict a short time out

Web Link Analysis

- PageRank over a web graph
  - Each web page is a state
  - Initial distribution: uniform over pages
  - Transitions:
    - With prob. c, follow a random outlink (solid lines)
    - With prob. 1-c, uniform jump to a random page (dotted lines, not all shown)
  - Stationary distribution
    - Will spend more time on highly reachable pages
    - E.g. many ways to get to the Acrobat Reader download page
    - Somewhat robust to link spam
    - Google 1.0 returned the set of pages containing all your keywords in decreasing rank, now all search engines use link analysis along with many other factors (rank actually getting less important over time)

Hidden Markov Models

- Markov chains not so useful for most agents
  - Eventually you don’t know anything anymore
  - Need observations to update your beliefs

- Hidden Markov models (HMMs)
  - Underlying Markov chain over states S
  - You observe outputs (effects) at each time step
  - As a Bayes’ net:

Example

- An HMM is defined by:
  - Initial distribution: $P(X_1)$
  - Transitions: $P(X_t | X_{t-1})$
  - Emissions: $P(E | X)$

Example

- An HMM is defined by:
  - Initial distribution: $P(X_1)$
  - Transitions: $P(X_t | X_{t-1})$
  - Emissions: $P(E | X)$
Hidden Markov Models

- Defines a joint probability distribution:
  \[ P(X_1, \ldots, X_n, E_1, \ldots, E_n) = P(X_1)P(E_1|X_1) \prod_{i=2}^nP(X_i|X_{i-1})P(E_i|X_i) \]

Ghostbusters HMM

- \( P(X_1) = \) uniform
- \( P(X|X) = \) usually move clockwise, but sometimes move in a random direction or stay in place
- \( P(E|X) = \) same sensor model as before: red means close, green means far away.

HMM Computations

- Given
  - joint \( P(X_1, E_1) \)
  - evidence \( E_1 = e_1 \)

- Inference problems include:
  - Filtering, find \( P(X_t|e_1) \) for all \( t \)
  - Smoothing, find \( P(X_t|e_1) \) for all \( t \)
  - Most probable explanation, find \( x^*_{1:n} = \arg\max_{x_{1:n}} P(x_{1:n}|e_{1:n}) \)