CSE 573: Artificial Intelligence
Winter 2019

Hidden Markov Models

Hanna Hajishirzi

Many slides adapted from Pieter Abbeel, Dan Klein, Dan Weld, Stuart Russell, Andrew Moore & Luke Zettlemoyer
Today

- HMMs
  - Particle filters
  - Demos!

- Applications:
  - Robot localization / mapping

- Bayes Nets
Recap: Reasoning Over Time

- **Markov models**

\[ P(X_1) \quad P(X|X_{-1}) \]

- **Hidden Markov models**

\[
\begin{array}{c}
X_1 \\
E_1
\end{array}
\begin{array}{c}
X_2 \\
E_2
\end{array}
\begin{array}{c}
X_3 \\
E_3
\end{array}
\begin{array}{c}
X_4 \\
E_4
\end{array}
\]

\[
\begin{array}{c}
\text{rain} \\
0.9
\end{array}
\begin{array}{c}
\text{umbrella}
\end{array}
\]

\[
\begin{array}{c}
\text{sun} \\
0.1
\end{array}
\begin{array}{c}
\text{no umbrella}
\end{array}
\]

\[
\begin{array}{c}
\text{sun} \\
0.2
\end{array}
\begin{array}{c}
\text{umbrella}
\end{array}
\]

\[
\begin{array}{c}
\text{sun} \\
0.8
\end{array}
\begin{array}{c}
\text{no umbrella}
\end{array}
\]

\[
\begin{array}{c}
\text{rain} \\
0.7
\end{array}
\begin{array}{c}
\text{umbrella}
\end{array}
\]

\[
\begin{array}{c}
\text{sun} \\
0.3
\end{array}
\begin{array}{c}
\text{no umbrella}
\end{array}
\]
Inference: Base Cases

\[ P(X_1|e_1) \]

\[
P(x_1|e_1) = \frac{P(x_1, e_1)}{P(e_1)}
\]
\[
\propto_{x_1} P(x_1, e_1)
\]
\[
= P(x_1)P(e_1|x_1)
\]

\[ P(X_2) \]

\[
P(x_2) = \sum_{x_1} P(x_1, x_2)
\]
\[
= \sum_{x_1} P(x_1)P(x_2|x_1)\]
Passage of Time

- Assume we have current belief $P(X \mid \text{evidence to date})$
  $$B(X_t) = P(X_t \mid e_{1:t})$$

- Then, after one time step passes:
  $$P(X_{t+1} \mid e_{1:t}) = \sum_{x_t} P(X_{t+1}, x_t \mid e_{1:t})$$
  $$= \sum_{x_t} P(X_{t+1} \mid x_t, e_{1:t}) P(x_t \mid e_{1:t})$$
  $$= \sum_{x_t} P(X_{t+1} \mid x_t) P(x_t \mid e_{1:t})$$

- Basic idea: beliefs get “pushed” through the transitions
  - With the “B” notation, we have to be careful about what time step $t$ the belief is about, and what evidence it includes

- Or compactly:
  $$B'(X_{t+1}) = \sum_{x_t} P(X' \mid x_t) B(x_t)$$
Assume we have current belief $P(X | \text{previous evidence})$:

$$B'(X_{t+1}) = P(X_{t+1}|e_{1:t})$$

Then, after evidence comes in:

$$P(X_{t+1}|e_{1:t+1}) = \frac{P(X_{t+1}, e_{t+1}|e_{1:t})}{P(e_{t+1}|e_{1:t})}$$

$$\propto P(X_{t+1}, e_{t+1}|e_{1:t})$$

$$= P(e_{t+1}|e_{1:t}, X_{t+1}) P(X_{t+1}|e_{1:t})$$

$$= P(e_{t+1}|X_{t+1}) P(X_{t+1}|e_{1:t})$$

Or, compactly:

$$B(X_{t+1}) \propto X_{t+1} P(e_{t+1}|X_{t+1}) B'(X_{t+1})$$

Basic idea: beliefs “rewighted” by likelihood of evidence

Unlike passage of time, we have to renormalize
Filtering

**Elapse time:** compute $P(X_t \mid e_{1:t-1})$

\[ P(x_t \mid e_{1:t-1}) = \sum_{x_{t-1}} P(x_{t-1} \mid e_{1:t-1}) \cdot P(x_t \mid x_{t-1}) \]

**Observe:** compute $P(X_t \mid e_{1:t})$

\[ P(x_t \mid e_{1:t}) \propto P(x_t \mid e_{1:t-1}) \cdot P(e_t \mid x_t) \]

\[ \begin{array}{cccccccc}
<0.01 & <0.01 & <0.01 & <0.01 & <0.01 & <0.01 \\
0.01 & <0.01 & 0.06 & <0.01 & <0.01 & <0.01 \\
<0.01 & 0.76 & 0.06 & 0.06 & <0.01 & <0.01 \\
<0.01 & <0.01 & 0.06 & <0.01 & <0.01 & <0.01
\end{array} \]

**Belief:** $<P(\text{rain}), P(\text{sun})>$

- $P(X_1)$: $<0.5, 0.5>$  
  *Prior on $X_1$*

- $P(X_1 \mid E_1 = \text{umbrella})$: $<0.82, 0.18>$  
  *Observe*

- $P(X_2 \mid E_1 = \text{umbrella})$: $<0.63, 0.37>$  
  *Elapsed time*

- $P(X_2 \mid E_1 = \text{umb}, E_2 = \text{umb})$: $<0.88, 0.12>$  
  *Observe*
Example: Weather HMM

\[
\begin{align*}
B(+r) &= 0.5 \\
B(-r) &= 0.5
\end{align*}
\]

\[
\begin{align*}
B'(+r) &= 0.5 \\
B'(-r) &= 0.5
\end{align*}
\]

\[
\begin{align*}
B(+r) &= 0.818 \\
B(-r) &= 0.182
\end{align*}
\]

\[
\begin{align*}
B'(+r) &= 0.627 \\
B'(-r) &= 0.373
\end{align*}
\]

\[
\begin{align*}
B(+r) &= 0.883 \\
B(-r) &= 0.117
\end{align*}
\]

\[
\begin{array}{c|c|c}
R_t & R_{t+1} & P(R_{t+1} | R_t) \\
\hline
+r & +r & 0.7 \\
+r & -r & 0.3 \\
-r & +r & 0.3 \\
-r & -r & 0.7
\end{array}
\]

\[
\begin{array}{c|c|c}
R_t & U_t & P(U_t | R_t) \\
\hline
+r & +u & 0.9 \\
+r & -u & 0.1 \\
-r & +u & 0.2 \\
-r & -u & 0.8
\end{array}
\]
Approximate Inference

- Sometimes $|X|$ is too big for exact inference
  - $|X|$ may be too big to even store $B(X)$
  - E.g. when $X$ is continuous
  - $|X|^2$ may be too big to do updates

- Solution: approximate inference by sampling
- How robot localization works in practice
What is Sampling?

- Goal: Approximate the original distribution:

- Approximate with Gaussian distribution

- Draw samples from a distribution close enough to the original distribution
Approximate Solution: Perfect Sampling

Robot path till time $n$

Particle 1  \( x_{0:n}^1 \)

Particle $N$  \( x_{0:n}^N \)

Assume we can sample from the original distribution

\[ p(x_{0:n} \mid y_{0:n}) \]

\[
P(x_{0:n} \mid y_{0:n}) = \frac{1}{N} \quad \text{Number of samples that match with query}
\]

Converges to the exact value for large $N$
Particle Filtering

- Filtering: approximate solution
- Sometimes \(|X|\) is too big to use exact inference
  - \(|X|\) may be too big to even store \(B(X)\)
  - E.g. \(X\) is continuous
- Solution: approximate inference
  - Track samples of \(X\), not all values
  - Samples are called particles
  - Time per step is linear in the number of samples
  - But: number needed may be large
  - In memory: list of particles, not states
- Particle is just new name for sample
Our representation of $P(X)$ is now a list of $N$ particles (samples)
  - Generally, $N << |X|$

$P(x)$ approximated by number of particles with value $x$
  - So, many $x$ may have $P(x) = 0!$
  - More particles, more accuracy

For now, all particles have a weight of 1
Particle Filtering: Elapse Time

- Each particle is moved by sampling its next position from the transition model:
  \[ x' = \text{sample}(P(X'|x)) \]

  - This is like prior sampling – samples’ frequencies reflect the transition probabilities.
  - Here, most samples move clockwise, but some move in another direction or stay in place.

- This captures the passage of time:
  - If enough samples, close to exact values before and after (consistent)
**Particle Filtering: Observe**

- **Slightly trickier:**
  - Don’t sample observation, fix it
  - Downweight samples based on the evidence

\[ w(x) = P(e|x) \]

\[ B(X) \propto P(e|X)B'(X) \]

- As before, the probabilities don’t sum to one, since all have been downweighted (in fact they now sum to (N times) an approximation of \( P(e) \))
Rather than tracking weighted samples, we resample.

N times, we choose from our weighted sample distribution (i.e. draw with replacement).

This is equivalent to renormalizing the distribution.

Now the update is complete for this time step, continue with the next one.
Recap: Particle Filtering

- Particles: track samples of states rather than an explicit distribution
Particle Filtering Summary

- Represent current belief $P(X | \text{evidence to date})$ as set of $n$ samples (actual assignments $X=x$)
- For each new observation $e$:
  1. Sample transition, once for each current particle $x$
     \[ x' = \text{sample}(P(X'|x)) \]
  2. For each new sample $x'$, compute importance weights for the new evidence $e$:
     \[ w(x') = P(e|x') \]
  3. Finally, normalize by resampling the importance weights to create $N$ new particles
HMM Examples & Applications
P4: Ghostbusters

- **Plot:** Pacman's grandfather, Grandpac, learned to hunt ghosts for sport.

- He was blinded by his power, but could hear the ghosts’ banging and clanging.

- **Transition Model:** All ghosts move randomly, but are sometimes biased.

- **Emission Model:** Pacman knows a “noisy” distance to each ghost.

Noisy distance prob
True distance = 8
Which Algorithm?

Particle filter, uniform initial beliefs, 25 particles
Which Algorithm?

Exact filter, uniform initial beliefs
Which Algorithm?

Particle filter, uniform initial beliefs, 300 particles
In robot localization:

- We know the map, but not the robot’s position
- Observations may be vectors of range finder readings
- State space and readings are typically continuous (works basically like a very fine grid) and so we cannot store $B(X)$
- Particle filtering is a main technique
Particle Filter Localization (Sonar)

Global localization with sonar sensors

[Video: global-sonar-uw-annotated.avi]
Particle Filter Localization (Laser)

[Video: global-floor.gif]
Robot Mapping

- **SLAM: Simultaneous Localization And Mapping**
  - We do not know the map or our location
  - State consists of position AND map!
  - Main techniques: Kalman filtering (Gaussian HMMs) and particle methods

[Demo: PARTICLES-SLAM-mapping1-new.avi]