Lecture 8:
A Vacuum Cleaning Robot: Sensing, and Perception
• For the trash sorting robot, we had multiple sensors, and their measurements were conditionally independent (given state).

• We could combine those measurements using Bayes to formulate state estimates.

• For the vacuum cleaning robot, we’ll use a single sensor that has only three possible outputs: not very powerful.

• We’ll take measurements at each time step, and combine these with the robot’s knowledge about its actions and the environment to make inferences about state.

• Bayes networks – and various special cases of Bayes nets – will be the key inference tool.
Trash Sorting Sensors

• Three sensors (weight, conductivity, vision-classifier).

• At any time $t$, collect measurements from the three sensors: $z_t^w, z_t^c, z_t^v$ and use Bayes to compute $P(X_t = x | z_t^w, z_t^c, z_t^v)$.

• Measurements are conditionally independent given state, which gives a nice computational simplification after applying Bayes.

➤ The passing of time was irrelevant – each new sensor measurement was for a new piece of trash:
  • Completely independent of previous measurements
  • Completely independent of previous actions
  • Completely independent of previous states

This is not the case for our vacuuming robot!
Vacuuming robot sensor

• A single sensor that detects light levels, and returns a measurement $z$:

  • Bright, $z = 2$
  • Medium, $z = 1$
  • Dark, $z = 0$

<table>
<thead>
<tr>
<th>$X_f$</th>
<th>dark</th>
<th>medium</th>
<th>light</th>
</tr>
</thead>
<tbody>
<tr>
<td>Living Room</td>
<td>0.1</td>
<td>0.1</td>
<td>0.8</td>
</tr>
<tr>
<td>Kitchen</td>
<td>0.1</td>
<td>0.1</td>
<td>0.8</td>
</tr>
<tr>
<td>Office</td>
<td>0.2</td>
<td>0.7</td>
<td>0.1</td>
</tr>
<tr>
<td>Hallway</td>
<td>0.8</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>Dining Room</td>
<td>0.1</td>
<td>0.8</td>
<td>0.1</td>
</tr>
</tbody>
</table>

• Sun is to the south, so plenty of light for living room and kitchen.
• Office and Dining room are poorly lit via windows.
• Hallway has no windows and is always dark.

• For Hallway, $(z = 0 \mid H) = 0.8$, MLE will do the job!
• For $z = 1, z = 2$, there’s really no way to uniquely identify state from one measurement.
Exploiting History

• Suppose we observe a sequence of measurements and actions:
  \[ z_1 = 0, a_1 = up, z_2 = 2 \]
  It seems likely that \( x_1 = H, x_2 = L \)

• Suppose we observe a sequence of measurements and actions:
  \[ z_1 = 1, a_1 = right \]
  \[ z_2 = 0, a_2 = right \]
  \[ z_3 = 1 \]
  It seems likely that \( x_1 = O, x_2 = H, x_3 = D \)

These examples illustrate the basic idea, but these examples are really simple.

How do we formalize/generalize this into a sensor model that accounts for actions and measurements as time sequences?
Bayes Networks

In the past, we have seen graphical models for various sorts of Markov chains:

These models are special cases of the more general Bayesian Networks (Bayes nets):

• Directed Acyclic Graph (DAG)
• For conditional probability $P(X|Y_1, ..., Y_m)$ there are directed edges from each of $Y_i$ to $X$.
• There are no other edges in the graph.
This network represents several conditional probability relationships:

- $P(W|X,Z)$
- $P(X|Y,Z)$
- $P(Y|Z)$
- $P(Z)$

Perhaps more importantly, Bayes nets explicitly encode \textit{conditional independence} relationships:

- $W$ is conditionally independent of $Y$ given $X$
The (first) Magic of Bayes Nets

For a Bayes net with variables $X_1 \ldots X_n$, the joint distribution is given by:

$$P(X_1 \ldots X_n) = \prod_i P(X_i | \mathcal{P}(X_i))$$

where $\mathcal{P}(X_i)$ denotes the set of parents of node $X_i$

For this specific network, the joint distribution is given by

The (first) Magic of Bayes Nets

We can see why this works (for this example) by expanding the chain rule for joint probability distributions:

\[
\]

But from the topology of the Bayes net, we know

\[
P(W|X, Y, Z) = P(W|X, Z)
\]

Making this substitution, we arrive to the desired result:

\[
\]
More Magic of Bayes Nets

How difficult would it be to explicitly encode the joint distributions for our vacuum cleaning robot?

Suppose we consider $X_1, \ldots X_{T+1}$, and we want to encode $P(X_1, \ldots X_{T+1})$

<table>
<thead>
<tr>
<th>$X_1$</th>
<th>$X_2$</th>
<th>$\ldots$</th>
<th>$X_T$</th>
<th>$X_{T+1}$</th>
<th>$P(X_1, \ldots X_{T+1})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>L</td>
<td></td>
<td>L</td>
<td>L</td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>L</td>
<td></td>
<td>L</td>
<td>K</td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>O</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>H</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>D</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\ddots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
</tr>
</tbody>
</table>

Number of rows = $|X|^{T+1}$

Total storage $\approx (T + 1)(|X|^2 \times |\mathcal{A}|)$

# rows in CPT = $|X|^2 \times |\mathcal{A}|$
Dynamic Bayes Nets

• Bayes nets can be used to represent systems that evolve over time.
• Our vacuum cleaning robot is an example of such a system, at any time $t$, we have $x_t$ and $a_t$ and together, these determine (probabilistically) what happens for $x_{t+1}$.
• A dynamic Bayes net has a simple structure that repeats at each time step:
Simulation

- Forward simulation is easy for Dynamic Bayes Nets (DBNs).
- Sample initial state $x_0$ from the prior $P(X_0)$
- For each $k$ generate a sample $x_{k+1}$ from the distribution $P(X_{k+1}|X_k = x_k, a_k)$
- This is sometimes called **ancestral sampling**: to generate a sample for some node, look at its immediate ancestors.
Observations

• The motivation for all of this Bayes net machinery was the idea that the history of sensor measurements was interesting. How do we encode this in a Bayes net?

• Recall our sensor model: $P(Z_t | X_t)$.

• This is easy to encode in a Bayes net!
Still More Magic of Bayes Nets

For a controlled HMM with states $X_0, \ldots, X_n$, and observations $Z_0, \ldots, Z_n$, the joint distribution is given by:

$$P(Z_0, \ldots, Z_n, X_0 \ldots X_n | a_0 \ldots a_n) = P(Z_0 | X_0)P(X_0) \prod_i P(Z_i | X_i)P(X_i | X_{i-1}, a_i)$$
Simulation Revisited

• Forward simulation is easy for Dynamic Bayes Nets (DBNs).
• Sample initial state $x_0$ from the prior $P(X_0)$
• For each $k$
  • generate a sample $z_k$ from the distribution $P(Z_k | X_k = x_k)$
  • generate a sample $x_{k+1}$ from the distribution $P(X_{k+1} | X_k = x_k, a_k)$
As before, perception is the problem of inferring things about the world given sensor information and context.

For our controlled HMM, we have

- a sequence of given measurements $Z_t = z_t$
- the known sequence of applied actions $a_1, \ldots, a_n$

and we want to infer the states, $X_1, \ldots, X_n$

There is a lot of structure in this problem, and we can exploit this structure to obtain computationally efficient inference algorithms.
Hidden Markov Models (HMMs)

• Notice that in the system shown below,
  • we know $Z_t = z_t$ for all $t$
  • We know $a_t$ for all $t$

• We do not know any of $X_0 \ldots X_1$, but we do know that the states form a Markov chain.

• We say that the states, $X_0 \ldots X_n$, are hidden.

HMMs are a good model for speech recognition systems:
• Spoken words behave like a Markov chain (if you know the current word, you know a lot about what will be the next word).
• Measurements are audio signals.

Note: If we increase the relevant history, e.g., so that state $X_t$ depends on $X_{t-1}, X_{t-2} \ldots X_{t-n}$, we have an nth order Markov chain. Larger $n$ gives better prediction.
Inference in Bayes Nets

Our perception problem is straightforward:

• Given $Z_1 = z_1 \ldots Z_n = z_n$, and the sequence of applied actions $a_1, \ldots, a_n$,
• Infer the states, $X_1, \ldots, X_n$

The description of the problem almost immediately tells us the mathematical specification:

➢ Use $P(X_1, \ldots, X_n \mid Z_1 = z_1 \ldots Z_n = z_n, a_1, \ldots, a_n)$ to determine an estimate of the state sequence.
Most Probable Explanation

- Recall the definition of conditional probability:

\[ P(A, B) = P(A|B)P(B) \]

- We want to compute \( P(X|Z, A) \):

\[ P(X|Z, A) = \frac{P(X, Z, A)}{P(Z, A)} \propto P(X, Z, A) \]

- We know how to compute \( P(X, Z, A) \)! (Bayes net magic)

\[
\begin{align*}
X &= X_1, \ldots X_n \\
Z &= Z_1, \ldots Z_n \\
A &= a_1, \ldots a_n
\end{align*}
\]
Most Probable Explanation

We are given $Z_t = z_t$, and $a_t$ for all $t$.

For every possible value of $x_0, \ldots, x_n$, compute

$$P(X, Z, A) = P(Z_0 = z_0|X_0 = x_0)P(X_0 = x_0) \prod_i P(Z_i = z_i|X_i = x_i)P(X_i = x_i|X_{i-1} = x_{i-1}, a_i)$$

Our estimate is given by

$$X^* = \arg \max_X P(X, Z, A)$$

Not the most efficient algorithm, but in principle, this gets the job done.