POMDPs

3-30-16
Lab 8: MCTS revisited

Part 1: Q-learning with UCB exploration
- selection, expansion from MCTS; value updates from Q-learning

Part 2: Offline MCTS: repeated runs from the start state
- selection, expansion, backpropagation

Part 3: Online MCTS: repeated runs from each state you encounter
- selection, expansion, simulation, backpropagation
Reading Quiz

Which of these MDP algorithms did the reading discuss extending to POMDPs?

a) Value iteration
b) Q-learning
c) Approximate Q-learning
d) MCTS
Why are they called “Markov” decision processes?

Markov assumption:

Outcomes are conditionally independent of history given the state.

- Transition probabilities only depend on what state you’re in.
- It doesn’t matter how you got to this state.

This assumption is saying that the state summarizes all the information we might want to know. If something other than the state affects transitions, the world is non-Markovian.
MDPs vs POMDPs

In an MDP, the agent always knows its state.

In a POMDP, the state is **partially observable**.

The agent believes some probability distribution over what state it’s in. eg:

\[
P(S_0, S_1, S_2) = \langle 0.45, 0.55, 0.0 \rangle
\]
Optimal policy in a POMDP

In an MDP, if we know the value of every state, the optimal policy picks the best action in expectation:

$$\arg \max_a \left[ R(s, a) + \sum_{s'} P(s'|s, a)V(s') \right]$$

In a POMDP, we need to extend the EV calculation to our uncertainty over states:

$$\arg \max_a \left[ \sum_s P(s) \left( R(s, a) + \sum_{s'} P(s'|s, a)V(s') \right) \right]$$
Exercise: compute the EV of each action

\[ R(s_0, a_0) = 0 \]
\[ R(s_0, a_1) = 1 \]
\[ R(s_1, a_0) = 2 \]
\[ R(s_1, a_1) = -1 \]

\[ \Pr(s_0) = 0.25 \quad \Pr(s_1) = 0.75 \]

\[ V(s_2) = 3 \]
\[ V(s_3) = 4 \]

\[ \arg \max_a \left[ \sum_s \Pr(s) \left( R(s, a) + \sum_{s'} \Pr(s'|s, a)V(s') \right) \right] \]
How do we come up with the values?

Value iteration in an MDP:

1) Initialize each state’s value to 0.
2) Compute the optimal horizon-1 policy for each state.
3) Update the value of each state based on the optimal policy.
4) Goto step 2; repeat until converged.

In a POMDP, we need to compute values and policies for beliefs, not states.
We can compute value as a function of belief

We already computed the value of each action at one specific belief: \( P(s1) = 0.75 \)

The value for each action is a linear function of the belief. If we have two states, we can draw them as lines (more states: hyperplanes).

\[
\arg\max_a \left[ \sum_s P(s) \left( R(s, a) + \sum_{s'} P(s'|s, a) V(s') \right) \right]
\]
We also get observations

So far, our beliefs are just based on the transition model.

We may also need to update our beliefs based on observations.

For example, if we see a the blue ghost down a corridor, all states where the blue ghost is elsewhere now have probability 0.
Exercise: what is the belief state distribution?

Initial distribution: \( \langle 0.4, 0.3, 0.3 \rangle \)

Action: a0

Observation: not in \( S_1 \)
Value iteration for POMDPs

1) Compute the optimal horizon-1 policy as a function of beliefs.
2) Update the value function for each action based on the optimal policy.
   a) This includes summing over possible observations.
3) Goto step 1; repeat until converged.
Discussion question

What if we don’t know the transition model in advance?

For MDPs, we switched from value iteration to Q-learning or MCTS.

Can we apply these experiential learning methods to POMDPs? If so, what modifications do we need to make?