Partially Observable Markov Decision Processes

- Who has already heard of prior to this class?
- If notation is unclear just ask

POMDP tuple

\[ < S, A, E, p(s'|s, a), p(e|s, a), r(s, a), \gamma, b_0 > \]

\[ \begin{align*}
S & : \text{set of states} \\
A & : \text{set of actions} \\
E & : \text{set of observations} \quad (\text{also often called O or Z})
\end{align*} \]

\[ p(s'|s, a) \quad \text{transition model (like MDP)} \]

\[ p(e|s, a) \quad \text{observation model} \]

\[ r(s, a) \quad \text{reward} \]

\[ \gamma \quad \text{discount factor} \]

In general won’t get direct access to states
get observations that provide some information about underlying state
still (like MDP) want to max expected \( \mathbb{E} \) of future rewards
want to compute a policy to achieve this
can think of policy as a function of either of 2 things
(i turns out equiv)

\[ \Pi : \text{history} \rightarrow a \]

\[ \text{history} = < a_1, z_1, a_2, z_2, \ldots, a_i, z_i > \]

\[ \Pi : b \rightarrow a \]

\( b \) is a belief state
vector of length \( |S| \)
\( b(s) \) represents probability of being in each state \( s \)

\[ \sum_s b(s) = 1 \quad \text{so belief lies in (}|S| - 1|\)-dim simplex \]

typically start with an initial \( b_0 \)
represents prior knowledge
could be uniform (card game, hidden cards from shuffled deck)
could be highly specific (prior prob of topics people)
could even be dull a function (robot having, may know
start but perceptual ambiguity
student knowing 3)
Problem w/history is gets longer w/each step belief state, when updated using Bayes filter, is a sufficient statistic for history

Let \( b_{t+1}(s) = p(s \mid h_{t+1}) \)

\( h_t = [h_{t-1}, a_t, e_t] \)

\( b_t(s^t) = p(s^t \mid h_t) \)

\[ = p(s^t \mid h_{t-1}, a_t, e_t) \]

\[ = \frac{p(s^t, e_t \mid h_{t-1}, a_t)}{p(e_t \mid h_{t-1}, a_t)} \quad \text{Bayes rule} \]

\[ = \sum_{s'} p(s', e_t, s \mid h_{t-1}, a_t) \frac{p(s \mid h_{t-1}, a_t)}{p(e_t \mid h_{t-1}, a_t)} \]

\[ = \sum_{s'} p(s', s \mid h_{t-1}, a_t) p(e_t \mid s', s, h_{t-1}, a_t) p(s \mid h_{t-1}, a_t) / p(e_t \mid h_{t-1}, a_t) \]

Markov property: observation depends only on state of interest

\[ = p(s' \mid s, a_t) \sum_{s'} p(s' \mid s, a_t) b_{t-1}(s) \]

\[ p(e_t \mid h_{t-1}, a_t) \rightarrow \text{numerator, summed over } s' \]

So to compute belief state after history \( h_t \) sufficient to calculate using transition, observation models & prior belief \( \rightarrow b \) sufficient stat for history

Anstrom 1965 showed \( b \) also sufficient for optimal control

So objective is to compute \( \Pi : b \rightarrow 0 \) to max expected \( \Pi \) future rwd

\( a \) Policies

If only have 1 time step to act, only 1A1 different policies, one for each action

What is the value of each of these policies?

\( V_P(s) = r(s, a(s)) \) reward for state \( s \) & action of policy \( P \)

Define the \( \alpha \)-vector associated w/policy \( P \) as a 1A1 length vector where each index \( i \) specifies the value of taking that policy in state \( s_i \)

the value of a belief \( b \) under a policy is the expected value over the distribution over states specified by \( b \)

\[ V_P(b) = \sum_{s} b(s) \alpha_P(s) \]

\[ = b \cdot \alpha_P \quad \text{dot product} \]

implies expected value of a particular fixed policy varies linearly with \( b \) (hyperplane in belief space)
the value function over all beliefs \( b \) is the supremum over the set of \( \alpha \)-vectors at each \( b \) will choose the policy of the \( \alpha \) which maximizes \( \text{bf}_{\alpha} \).

Implies value function is PWLC each \( \alpha \)-vector is hyperplane -> convex supremum of a set of convex functions is convex

*Note that \( \alpha \)-vectors/fixed policies effectively partition the belief space into regions that share the same best policy (cool, important to \( b \) finite conditional policies)

What about acting for 2 steps?

Consider potential conditional policy trees

Choose action \( \alpha_{i} \rightarrow \alpha_{i}' \) and choose a 1-step policy to follow out of set of 1-step policies for each possible observation

? How many CPTs does this create?

\( t=1 \) \( |A| \)
\( t=2 \) \( |A| \times |A| \)

Computing the value of a 2-step CPT \( P \)

\[
\alpha_{P}(s) = r(s, \alpha(p)) + \gamma \sum_{s'} \sum_{\alpha(p)} p(c|s', \alpha(p)) \alpha_{P}(c) p(c|s', \alpha(p))
\]

Can use this equation for any \( T \)-step CPT (building up from \( T=1 \) to \( T \))

At each time step value function is the supremum over all \( \alpha \in \Pi \).

Example (simplified)

Doctor diagnosing whether a patient has cancer state: has cancer or doesn't have cancer
actions: run a test (colonoscopy etc.) diagnose cancer diagnose no cancer
observations: null, positive test, neg test
transition model: if run a test, assume state says the same
if diagnose, assume patient leaves of a new patient comes in whose probability of cancer is \( b(cancer) \)
observation model

\[
\begin{align*}
p(\text{pos} | \text{test, } s=cancer) &= 0.8 \\
p(\text{neg} | \text{test, } s= \text{no cancer}) &= 0.9 \\
p(\text{null} | \text{either diagnosis}) &= 1.0
\end{align*}
\]
reward

\[ r(\text{test}, s=\text{cancer}/\text{no cancer}) = -1 \]
\[ r(\text{diag cancer}, s=\text{cancer}) = -100 \]
\[ r(\text{""}, s=\text{no cancer}) = -10 \]
\[ r(\text{diag no cancer}, s=\text{cancer}) = -250 \]
\[ r(\text{""}, s=\text{no cancer}) = 0 \]

\[ \gamma = 1.99 \]

1 time step to act

\[ \alpha_{\text{test}} = [-1, -1] \]
\[ \alpha_{\text{diag NC}} = [0, -250] \]
\[ \alpha_{\text{diag C}} = [-10, -100] \]

note that \( \alpha_{\text{diag NC}} \) is dominated: no beliefs at which it is best policy

2 time steps

how many CPTs?

potentially 81: 3^3
some observations are impossible for some actions
\[ 1A^2 + 21A = 15 \]
\[ \text{test} \]
\[ \text{neg} \quad \text{pos} \]
\[ \text{no cancer} \quad \text{test} \]

\[ V_p(\text{no cancer}) = -1 + \gamma \left( p(\text{pos|no cancer}) \alpha_{\text{test}}(\text{no cancer}) + p(\text{neg|no cancer}) \alpha_{\text{diag NC}}(\text{no cancer}) \right) \]
\[ = -1 + \gamma \left( .1, -1 + .9, 0 \right) \]
\[ = -1 + \gamma \left( -1 \right) = -1.099 \]

\[ V_p(\text{cancer}) = -1 + \gamma \left( p(\text{pos|cancer}) \alpha_{\text{test}}(\text{cancer}) + p(\text{neg|cancer}) \alpha_{\text{diag NC}}(\text{cancer}) \right) \]
\[ = -1 + \gamma \left( .9, -1 + .2, -250 \right) \]
\[ = -51.292 \]

\[ \alpha_p = [-1.099, -51.292] \]

Value function backups

can think of computing value function as enumerating all t+1 step conditional policy trees from prior set of t-step policy trees

can also think of computing best (t+1)-step policy tree for a particular belief b (\( t \) repeating this for all b)
let \( \Gamma \) be the set of \( t \)-step \( \alpha \)-vectors that make up the \( t \)-step value function (PWLC) 

back up for belief \( b \) is 

\[
V(b) = \max_a \left[ \sum_s r(s, a) b(s) + \gamma \sum_{s'} \max_{\alpha} \sum_{s} \sum_{a} p(s' | s, a) p(\epsilon | s', a) \alpha(s') b(s) \right]
\]

choose the \( t \)-step policy to 
follow that max expected future reward 
given started in \( b(s) \), took \( a \), \( \epsilon \) saw 

to see in a different way, multiply & divide inner term 
by \( p(\epsilon | b, \alpha) \) 

\[
V(b) = \max_a \left[ \sum_s r(s, a) b(s) + \gamma \sum_{s'} \max_{\alpha} p(\epsilon | b, \alpha) \sum_s \alpha(s') p(s' | s, a) b(s) \right] \\
= b^{\text{ac}}(s')
\]

just select \( t \)-step policy tree at 
each \( \epsilon \) branch that maximizes 
expected value of \( b^{\text{ac}}(s') \)

note that can view this backup as constructing the best \((t+1)\)-step 
policy for belief \( b \)

\[
\alpha^{\text{ac}}_b = r(s, a) + \gamma \sum_{s'} p(\epsilon | b, \alpha) \arg \max_{\alpha} \left[ \alpha(s') b(s) \right]
\]

\[
\alpha = \arg \max_{\alpha} \alpha^{\text{ac}}_s, b \quad \text{(still linear, value func sup norm over linear, convex)}
\]

if do this for all possible beliefs, not really efficient 
only a finite \# of \( m \)-step conditional policy trees for any \( m \) \in \( \mathbb{N} \)

infinite \# of beliefs \((|s| - 1)\)-dim simplex

So might as well enumerate conditioned policy trees & then 
prune those which are dominated 
again, recall \( \alpha \)-partition 
belief space

but later see that if not computing a policy over all 
beliefs, could be helpful

\[
? \text{ is MDP solution an upper or lower bound to POMDP } V_v^* \text{?}
\]

upper bound: 

\[
\max_{\alpha} V(b) = \sum_e b(\epsilon) V^{\max}(\epsilon)
\]

intuitively: full state knowledge can never hurt you

mathematically 

\[
V(b) = \max_a \left[ \sum_s r(s, a) b(s) + \gamma \sum_{s'} \max_{\alpha} \sum_s \sum_{a} p(s' | s, a) p(\epsilon | s', a) \alpha(s') b(s) \right] \\
\leq \max_a \left[ \sum_s r(s, a) b(s) + \gamma \sum_s \sum_{a} \max_{\alpha} \sum_{s'} p(s' | s, a) p(\epsilon | s', a) \alpha(s') b(s) \right] \\
= \max_a \left[ \sum_s r(s, a) b(s) + \gamma \sum_{s'} \sum_s b(s) \max_{\alpha} \sum_{s'} p(s' | s, a) \alpha(s') \right] \\
= \max_a \sum_s b(s) V^{\max}(s)
\]
What does P-WLC nature of value function imply about uncertainty 
often worse to be uncertain, never better to be uncertain vs. max of rewards if known.

Why can't we deduce ourselves (just believe we won the lottery) because belief state represents accurate probability that in each state possible states using real observation & transition models for state estimation.

In cancer example actions were fairly clearly separated into information gathering (diagnostic test) vs. reward max (decide cancer/nocancer). What is an example where actions might move along the continuum? (Or a domain)

Ex: robot navigation/coastal navigation/Minerva practicalities: intractable to enumerate all CPTs # grows as |A|^O(|E|^2)^n expensive to prune

Many of these may be irrelevant, given initial belief only a subset of beliefs may be reachable.

In work in 1990s (Kaelbling, Littman, Cassandra, Heussieck) most problems very small 4-12 states. This might be sufficient for some problems breast cancer & colon cancer screening have both been formulated as small POMDPs. Most work within AI community has focused on approximate solutions.

Is t-step value function upper or lower bound for (4t+1) step value func?

Answer: depends! If all positive, lower bound negative, upper bound.

If interested in infinite horizon policy can start by initializing starting x-vectors as a lower bound (ex. \( r_{min} / (1-\gamma) \)).

maybe do motiv exs before break