CSE 513T Theory of Artificial Intelligence & Machine Learning Nov 28, 2016

Lecture 24

Lecturer: Brendan Juba *Scribe: Li Xia(scribed)*

Recall

1. deterministic envieronment

   complete information

   - suffices to give α sequence of

2. Markov Decision Process 'planning'

   Stochastic environment

   Complete information

   -Must react to environment's random choices using a 'policy'

Partially observed Markov Decision Porcesses (POMDD)

Stochastic environment
Partial information (very general - we assume that the states are factored into attributes and partial information means some of these attributes get hidden)
Still must react to enviroment's random choices; must use 'reasoning' to cope with missing information

Define A POMDD:

is an interactive process given by sets of states X, actions A, and observations O along with distributions
$\forall x \in X$, $\lambda \in A$
$D(x, \alpha)$ over next states (from X)
and for each $x \in X$, $\alpha \in A$,
P(x, α) over observation from θ
(We'll also have a distirbution D_0 over initial states from X)
We assume X=$0, 1^n$,
O is a masking process on X×A taht does not hide the action :
See figure 1
As before we'll consider goals of the form, reach x_T s.t G(x_T)=1 for some prodicte G that is given as input:
Again we will seek to design policies that cope with "faults" or random choices of the environment using observations
We will continue to look at rank-k decison tree policies with the convention that branches are laveled with a literal if it is satisified we take one branch is failfilled or unknown, we take the other branch
All we hope to achieve with our policies is that G is (1-ϵ) - valid when π terminates

As we have the problem: Robot find the tube in n rooms:

Informally goal is put the box in room 2 don't be in room 2 i.e.
only reveals the attributes for the room i s.t: $\partial t_{b,2} \wedge \neg \partial t_{r,2}$
suppose O
$\partial t_{r,i} = 1$
In the final steps of our plan:

	$\partial t_{r,0}$	$\partial t_{r,1}$	$\partial t_{r,2}$	$\partial t_{b,0}$	$\partial t_{b,1}$	$\partial t_{b,2}$	holds		
t=5	0	0	1	$*$	$*$	1	0	0	→G(X_5)=0

t=6 1 0 0 0 * * 0 0 →G(X_6)=0

Need to use a frame axion to infer $\partial t_{b,2}(x_6)$

Notice to use a noisy version of the environment in which, say, the frame axioms might fail up γ, we can only conclude Pr[G(x_L)] $\geq 1 - \gamma$

now we wish to learn the environment representation (why? knowledge engineering is tedious error-prove you may not think to include rules that seems "obvious", ctc)

Pitfalls in learning:

unknown domains

obvious 1st attempt given access to a POMDD capturing a noisy STRIPS domina. find STRIPs rules captureing the domain

Difficulty 1 "Exploration"

Consider the following combination lock" domain we have attributes x1,...xn, C1, ..., Cn and open

There are two actions '0' and '1' Exactly one Ci is 1 at any time.

After taking an action when Ci=1, we enter a state where Ci+1 is 1, Initially Ci=1

The action 'b' when Ci=1 has the effect of setting xi=b

Finally when Cn=1, if (x1 - xn) = (S1-Sn) there is a conditional effect that sets open =0)

Let's suppose effect that we choose (S1,...Sn) $\in \{0,1\}^n$ uniformly at random.

Consider the variant of the environment that does not have the effect taht sets open=1

Fix a sequence of actions. We only see open=1 if the sequence of actions was S1, ..., Sn. Since S1, ..., Sn was chosen at random, find Pr[S1, ..., Sn= $\alpha_1, ..., \alpha_n$] = $1/2^n$

Over T steps/trials:

We can take a union bound

Pr[open=1 in any of the T trials] $\leq T/2^n$

So to get prob $\geq 1/2$ of seeing open=1

we need $\geq 2^n/2trials$

Corollary distinguishing S $\neq$ S' also requires $\Omega(2^n)$ of trials

Fix, only aim to generate plans w.r.t portions of the environment that have been "explored well" in the training examples / past experience/

Difficulty 2: Explicit representations

In general: a Nousy STRIPS environment poses an agnostic learning problem conditional effects. For example, are at least as hard to learn as conjunctions. As we discussed agnostic learning is mostly out of reach for the state-of-the-art.

Fix Don't try to generate a complete explicit environment representation. We'll only seek to learn the environment representation. we'll only seek to learn the environment implicity and we'll constract a policy using queries against this implicit KB

Learning from example histories

We assume that there is some (arbitrary) reference "exploration policy" Π_0 that was used to generate histories for car training data

For example,

-Π_0 might just pick random actions

-Π_0 is a uniformly random past policy

-Π_0 could be be a more intelligent exploration policy(tree to reach new station)

We will assume we're given access to example histories of Π_0 acting in our environbment for T steps. Motivated by HW4 q3, we'll further assume taht each step includes the probability with which Π_0 chose it's action.

Def: We say that a sequence of action $\alpha_1, ...\alpha_t$ is M-typical for $\mu \in$ [0,1] for the reference policy Π_0 and the POMDD if the probability that Π_0

Note in order to guarantee we see a μ -typical sequence of actions we need $1/\mu$ examples

Our task: given that there is a policy (rank-k decision tree) that only uses typical sequences of actions on any branch and such that under the conditional distribution reached on each branch, some KB that is (1-σ) - testible allow us to prove G is satisfied on the branch.

Find such a policy using access to histories generated by Π_0

Surprising thing we'll get a poly-time algorithm in terms of $1/\mu, 1/\epsilon, 1/\theta, n, T$ and exponential in k,s (space for a resolution proof)

Figure 1:

