687 2017-12-06 "kitchen sink" of Topics

Partial Observability

- How to handle partial observability and randomness in observations?

Function approximation still works:

$\phi(s)$ or $\phi(s,a)$

Can append a noise variable to the state:

$s = (x, y, \underset{\uparrow}{n})$ from some noise distribution

$\phi(s) = (x+n_1, y+n_2)$

MDP simply reacts in the moment

vs. POMDP - can exploit the history

Least Squares Temporal Difference Learning (LSTD)

Bradtke + Barto (1996)

* Boyen (1998) ← easier to read

- Policy evaluation
- Linear fn. approx. (or tabular)

Idea: Solve for TD(λ) fixed point given a batch of data (s_i, a_i, r_i, s_{i+1}) tuples

$$\mathbb{E}[(R_t + \hat{\gamma} w^T \phi(S_{t+1}) - w^T \phi(S_t))\phi(S_t)] = 0$$

$$\underbrace{\mathbb{E}[(\gamma \phi(S_{t+1}) - \phi(S_t))\phi(S_t)^T]}_{A} \hat{w} = \mathbb{E}[R_t \phi(S_t)]$$

$$A \cdot w = b$$

w that minimizes least square error

Least Squares Policy Improvement (LSPI)
- Same Idea applied to Policy Improvement

Hierarchical Reinforcement Learning (HRL)

Sensory inputs: eyes, ears, touch, ... → Brain → spinal cord + muscles

Goal: Play chess

High level action: Advance this pawn

Next level: Move hand to pawn; Grasp it; ...

Next level: Muscle activations

"Options Framework" Sutton et al. 1999

- One formalization is "Skills"
- Call a skill an "option":

Atuple $(\pi, \beta, \mathcal{I})$

π: policy

β: termination condition

$\beta: \mathcal{S} \to [0,1]$ Prob. of termination is a given state

$\mathcal{I}$: initiation set, $\subset \mathcal{S}$ (when can you start this skill?)

A meta-agent can select among primitive actions or options at any step. (Or possibly only options.)

Can this be done automatically?

Not solved yet!

Some methods define sub-goals.

Another: Option-Critic looks at using policy gradient to set β.

Semi-MDP: actions can vary in length - affects reward discounting.

Optimality relative to skills available

Shaping Rewards Ng, Harada, Russell (1999)

Real reward may be just at a goal, but want to encourage movement toward better parts of state space.

How do we correct inappropriate sensitivity to rewards?

Let $R'_t = R_t + F_t$. Then:

R'_t induces the same optimal policies iff F_t is a "potential-based" shaping fn, i.e. $F_t(S_t, A_t, R_t) = \gamma \Phi(S_{t+1}) - \Phi(S_t)$

for some $\Phi : \mathcal{S} \Rightarrow \mathbb{R}$

A later result: for Q learning + SARSA, the F_t is equivalent to setting starting value fns.

Experience Replay: Lin (1992)
"Self-Improving Reactive ...."

Idea: Q-learning uses samples only once - wasteful - samples may be rare or costly to obtain.

- Store "experiences" (s, a, r, s') + repeatedly present them to a learning agent.
- Experiences may be off current policy, so not great for on-policy methods like SARSA (but can discount probability of replay of older experiences).
- Does work well with eligibility traces, e.g., as in TD(λ).