Machine Learning 10-701

Tom M. Mitchell
Machine Learning Department
Carnegie Mellon University

April 28, 2011

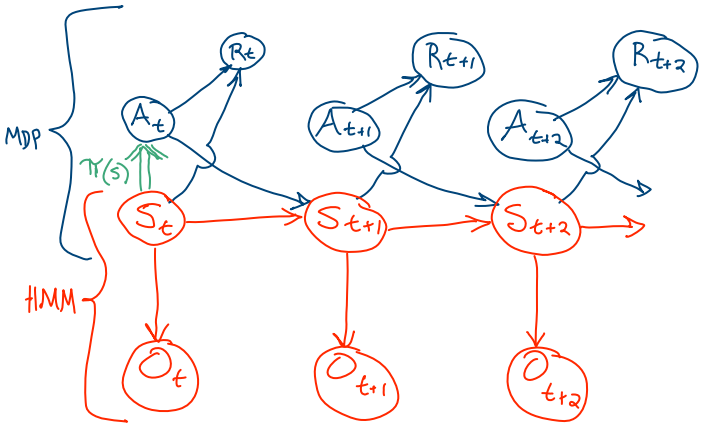
Today:

- Learning of control policies
- TD(λ)
- Animal learning from rewards

Readings:

- Mitchell, chapter 13
- Kaelbling, et al., *Reinforcement Learning: A Survey*

HMM, Markov Process, Markov Decision Process



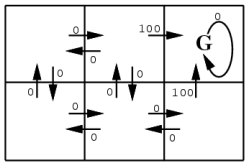
Immediate rewards r(s,a)

State values V*(s)

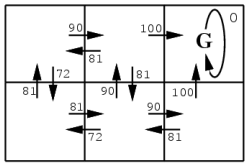
State-action values Q*(s,a)

$$V^*(s) = E[r(s, \pi^*(s))] + \gamma E_{s'|s,\pi^*(s)}[V^*(s')]$$

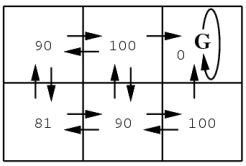
Bellman equation.



$r(s,a)$ (immediate reward) values

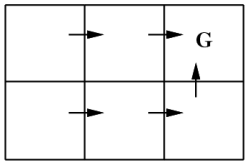


$Q(s,a)$ values



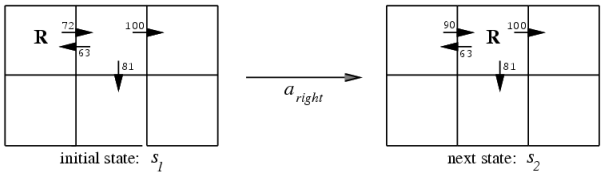
$V^*(s)$ values

Consider first the case where P(s'| s,a) is deterministic



One optimal policy

Updating $\hat{Q}$



$$\begin{aligned}\hat{Q}(s_1, a_{right}) &\leftarrow r + \gamma \max_{a'} \hat{Q}(s_2, a') \\ &\leftarrow 0 + 0.9 \max\{63, 81, 100\} \\ &\leftarrow 90\end{aligned}$$

notice if rewards non-negative, then

$$(\forall s, a, n) \quad \hat{Q}_{n+1}(s,a) \geq \hat{Q}_n(s,a)$$

and

$$(\forall s, a, n) \quad 0 \leq \hat{Q}_n(s,a) \leq Q(s,a)$$

Nondeterministic Case

Q learning generalizes to nondeterministic worlds

Alter training rule to

$$\hat{Q}_n(s,a) \leftarrow (1-\alpha_n)\hat{Q}_{n-1}(s,a)+\alpha_n[r+\max_{a'}\hat{Q}_{n-1}(s',a')]$$

where

$$\alpha_n = \frac{1}{1+visits_n(s,a)}$$

Can still prove convergence of $\hat{Q}$ to Q [Watkins and Dayan, 1992]

Temporal Difference Learning

Q learning: reduce discrepancy between successive Q estimates

One step time difference:

$$Q^{(1)}(s_t,a_t) \equiv r_t + \gamma \max_a \hat{Q}(s_{t+1},a)$$

Why not two steps?

$$Q^{(2)}(s_t,a_t) \equiv r_t + \gamma r_{t+1} + \gamma^2 \max_a \hat{Q}(s_{t+2},a)$$

Or n?

$$Q^{(n)}(s_t,a_t) \equiv r_t+\gamma r_{t+1}+\cdots+\gamma^{(n-1)} r_{t+n-1}+\gamma^n \max_a \hat{Q}(s_{t+n},a)$$

Blend all of these:

$$Q^{\lambda}(s_t,a_t) \equiv (1-\lambda)\left[Q^{(1)}(s_t,a_t) + \lambda Q^{(2)}(s_t,a_t) + \lambda^2 Q^{(3)}(s_t,a_t)\right.$$

Temporal Difference Learning

$$Q^{\lambda}(s_t, a_t) \equiv (1-\lambda)\left[Q^{(1)}(s_t, a_t) + \lambda Q^{(2)}(s_t, a_t) + \lambda^2 Q^{(3)}(s_t, a_t)\right.$$

Equivalent expression:

$$Q^{\lambda}(s_t, a_t) = r_t + \gamma[\ (1-\lambda)\max_a \hat{Q}(s_t, a_t) + \lambda\ Q^{\lambda}(s_{t+1}, a_{t+1})]$$

TD(λ) algorithm uses above training rule

- Sometimes converges faster than Q learning
- converges for learning V^* for any $0 \leq \lambda \leq 1$ (Dayan, 1992)
- Tesauro's TD-Gammon uses this algorithm

MDP's and RL: What You Should Know

- Learning to choose optimal actions A
- From *delayed reward*
- By learning evaluation functions like V(S), Q(S,A)

Key ideas:

- If next state function S_t x A_t → S_{t+1} is known
  - can use dynamic programming to learn V*(S)
  - or, learn it by sampling <s,a> pairs and applying our update rule
  - once learned, choose action A_t that maximizes $V^*(S_{t+1})$
- If next state function S_t x A_t → S_{t+1} **un**known
  - learn $Q(S_t,A_t) = E[V^*(S_{t+1})]$
  - to learn, sample <s,a> pairs by executing actions in actual world
  - once learned, choose action A_t that maximizes $Q(S_t,A_t)$

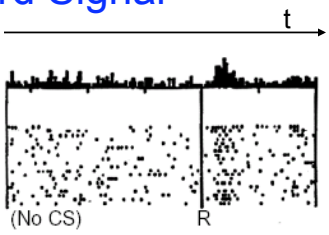
MDPs and Reinforcement Learning: Further Issues

- What strategy for choosing actions will optimize
  - learning rate? (*explore* uninvestigated states)
  - obtained reward? (*exploit* what you know so far)
- *Partially observable* Markov Decision Processes
  - state is not fully observable
  - maintain probability distribution over possible states you're in
- Convergence guarantee with function approximators?
  - our proof assumed a table representation for Q, V
  - some types of function approximators still converge (e.g., nearest neighbor) [Gordon, 1999]
- Correspondence to human learning?

Reinforcement Learning in Animals?

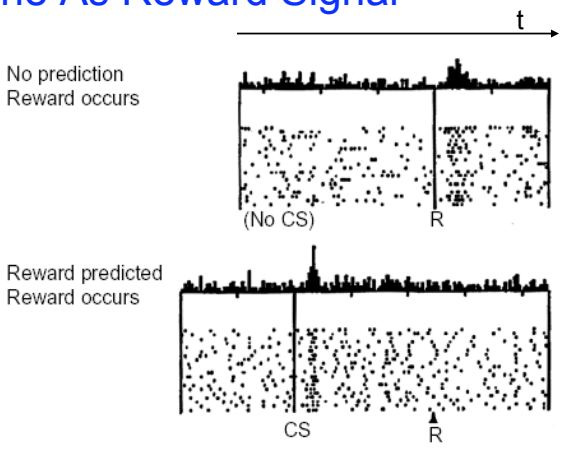
Dopamine As Reward Signal

[Schultz et al., *Science*, 1997]



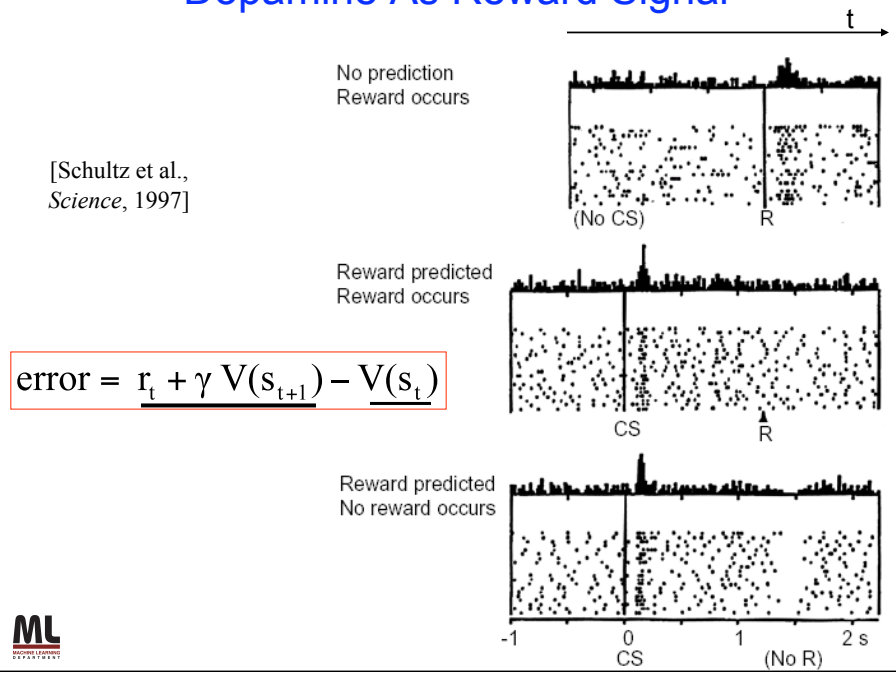
Dopamine As Reward Signal

[Schultz et al., *Science*, 1997]



Dopamine As Reward Signal

[Schultz et al., *Science*, 1997]



$$\text{error} = r_t + \gamma V(s_{t+1}) - V(s_t)$$

RL Models for Human Learning

[Seymore et al., Nature 2004]

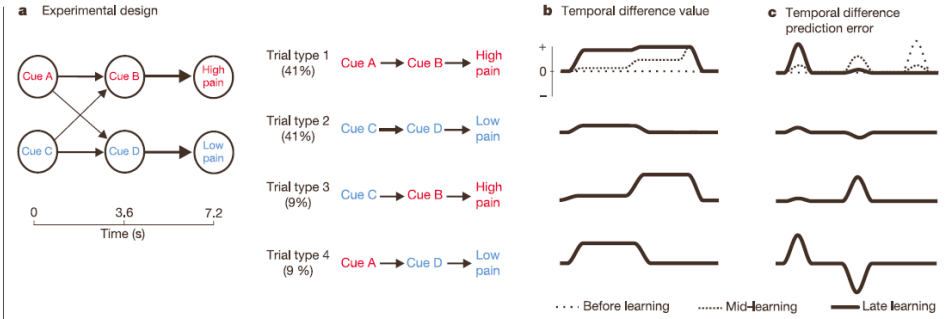
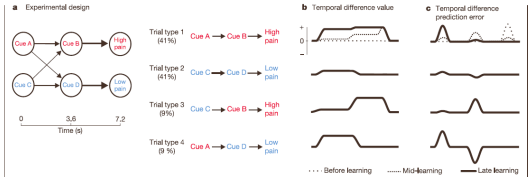


Figure 1 Experimental design and temporal difference model. **a**, The experimental design expressed as a Markov chain, giving four separate trial types. **b**, Temporal difference value. As learning proceeds, earlier cues learn to make accurate value predictions (that is, weighted averages of the final expected pain). **c**, Temporal difference prediction error, during learning the prediction error is transferred to earlier cues as they acquire the ability to make predictions. In trial types 3 and 4, the substantial change in prediction elicits a large positive or negative prediction error. (For clarity, before and mid-learning are shown only for trial type 1.)



[Seymore et al., Nature 2004]

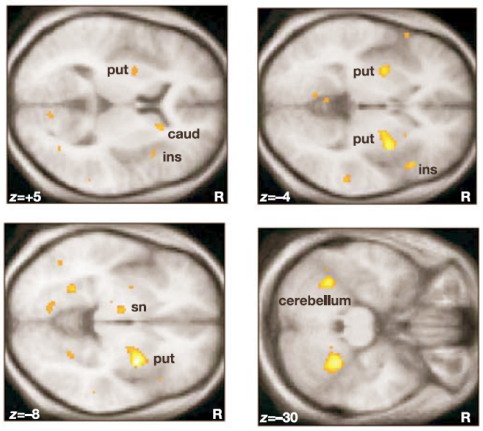


Figure 2 Temporal difference prediction error (statistical parametric maps). Areas coloured yellow/orange show significant correlation with the temporal difference

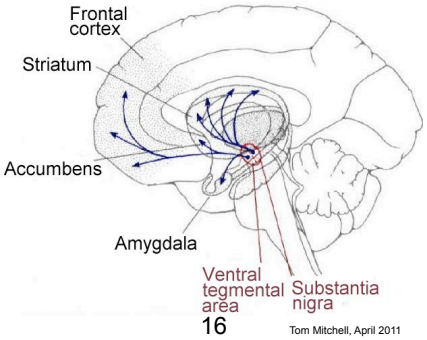
ril 2011

One Theory of RL in the Brain

from [Nieuwenhuis et al.]

- Basal ganglia monitor events, predict future rewards
- When prediction revised upward (downward), causes increase (decrease) in activity of midbrain dopaminergic neurons, influencing ACC

- This dopamine-based activation somehow results in revising the reward prediction function. Possibly through direct influence on Basal ganglia, and via prefrontal cortex



16

Tom Mitchell, April 2011

Summary: Temporal Difference ML Model Predicts Dopaminergic Neuron Acitivity during Learning

- Evidence now of neural reward signals from
  - Direct neural recordings in monkeys
  - fMRI in humans (1 mm spatial resolution)
  - EEG in humans (1-10 msec temporal resolution)
- Dopaminergic responses encode Bellman error
- Some differences, and efforts to refine the model
  - How/where is the value function encoded in the brain?
  - Study timing (e.g., basal ganglia learns faster than PFC ?)
  - Role of prior knowledge, rehearsal of experience, multi-task learning?

Tom Mitchell, April 2011