Temporal Difference Learning.

To estimate $\mathbb{E}[X]$ from $x_1, \cdots x_t$.

$$\mu_t = \mu_{t-1} + \alpha_t (x_t - \mu_{t-1})$$

Theorem: $\mu = \mathbb{E}[X]$ as $t \to \infty$, with ~~approximately~~ appropriately decay learning rates.

Temporal difference (TD) prediction.

* How to evaluate policy w/o model?
  compute $V^\pi(s)$ w/o knowing $P(s'|s, \pi(s))$.

* Explore state space using policy π.

$$\underset{R(s_0)}{s_0} \xrightarrow{\pi(s_0)} \underset{R(s_1)}{s_1} \xrightarrow{\pi(s_1)} \cdots$$

* Recall Bellman equation

$$V^\pi(s) = R(s) + \gamma \cdot \sum_{s'} P(s'|s, \pi(s)) \cdot V^\pi(s')$$

* TD learning algorithm.

Initialize $V_0^\pi(s) = 0$, for all states s. (at $t=0$).

update $V_{t+1}(s_t) = V_t(s_t) + \alpha_t \left[\underbrace{R(s_t) + \gamma \cdot V_t(s_{t+1})}_{\text{random sample.}} - V_t(s_t) \right]$

learning rate.

current estimate.

with appropriately decaying learning rates, this converges to correct values.

$\boxed{\text{Q-learning}}$

* How to optimize policy π^* w/o model. $P(s'|s, a)$?
  How to compute $Q^*(s,a)$ w/o model?

* Explore state-action spaces at random.

$$\underset{R(s_0)}{s_0} \xrightarrow{a_0} \underset{R(s_1)}{s_1} \xrightarrow{a_1} \cdots$$

* Recall Bellman Optimality Equation.

$$Q^*(s,a) = R(s) + \gamma \cdot \sum_{s'} p(s'|s,a) \cdot V^*(s')$$

$$Q^*(s,a) = R(s) + \gamma \cdot \sum_{s'} p(s'|s,a) \cdot \max_{a'} Q^*(s',a')$$

* Q-learning.
  - Initialize $Q_0(s,a)$ for all states s and actions a at time $t=0$.
  - $Q_{t+1}(s_t,a_t) = Q_t(s_t,a_t) + \alpha_t \left[\underbrace{R(s_t) + \gamma \cdot \max_{a'} Q_t(s_{t+1},a')}_{\text{random sample.}} - Q_t(s_t,a_t) \right]$

With appropriately decaying learning rates (and enough time to explore the state-action space), this converges to $Q^*(s,a)$ as time $\to \infty$.

Beyond CSE150.

1). Reinforcement learning in large state spaces.
  - so far implicit assumption that we ~~have~~ can store $V^\pi(s)$ and $\pi(s)$ as lookup tables.
  - in large state spaces, we must parameterize the state value function in terms of a smaller # features.
  - tradeoff / challenges:
    - (+) generalize to unseen states
    - (−) only approximating the true value function.

2). Partially Observable MDPs (POMDP's)

POMDPs are to MDPs as HMMs are to Markov models.

Example: Robot navigation.

action : motor commands.

state : x, y coordinates.

observation : sensor measurements.

* Model for POMDPs.
  - transition matrix $P(s'|s,a)$
  - reward. $R(s)$.
  - emmission matrix $P(O_t|s)$

Stream of experience:

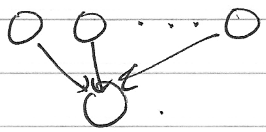
action	a_0	a_1	
	R_0	R_1	...
	O_0	O_1	

Never observe $s_0, s_1, s_2 \cdots$

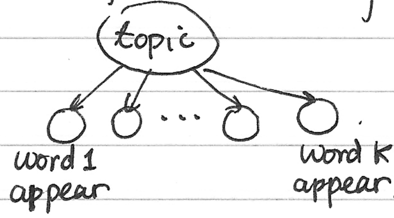
Goals of CSE 150

- How to discover compact representations of complex world?
- Balance power/expressiveness of model vs. computational tractability.

1). Noisy-OR CPT.



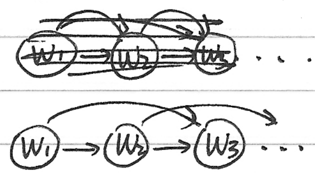
2) Naive Bayes.
(of document classification)



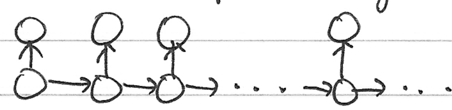
3) Markov model for language
w_ℓ = ℓth word in sentence.

bigram: $W_1 \to W_2 \ldots$

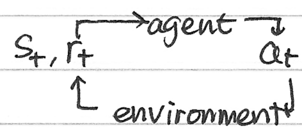
trigram:



4) HMMs for speech recognition.

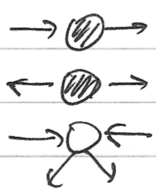


5). MDPs for planning

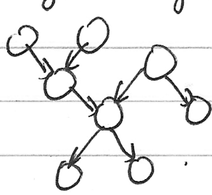


- What are efficient algorithms for automated forms of intelligence. reasoning, decision-making ....

1). conditional independence tests via d-seperation.



2) polytree algorithm for inference.



3). EM algorithm for Maximum Likelihood estimation w/ guarantees of monotonic convergence.

4) dynamic programming in HMMs.
Viterbi algorithm.
Forward/backward algorithm.

5) Algorithms in MDPs.
policy iteration
value iteration
Sampling algorithms.