Lecture 12: Fast Reinforcement Learning [1]

Emma Brunskill

CS234 Reinforcement Learning

Winter 2020

[1] With some slides derived from David Silver

Refresh Your Understanding: Multi-armed Bandits

a1 a2

- Select all that are true:

F 1. Up to slight variations in constants, UCB selects the arm with $\arg\max_a \hat{Q}_t(a) + \sqrt{\frac{1}{N_t(a)} \log(1/\delta)}$ log t is missing

T 2. Over an infinite trajectory, UCB will sample all arms an infinite number of times true

T 3. UCB still would learn to pull the optimal arm more than other arms if we instead used $\arg\max_a \hat{Q}_t(a) + \sqrt{\frac{1}{\sqrt{N_t(a)}} \log(t/\delta)}$ true

T 4. UCB uses $\arg\max_a \hat{Q}_t(a) + b$ where b is a bonus term. Consider $b = 5$. This will make the algorithm optimistic with respect to the empirical rewards but it may still cause such an algorithm to suffer linear regret.

T 5. Algorithms that minimize regret also maximize reward

6. Not sure

Class Structure

- Last time: Fast Learning (Bandits and regret)
- **This time: Fast Learning (Bayesian bandits)**
- Next time: Fast Learning and Exploration

Recall Motivation

customers patient robot

- Fast learning is important when our decisions impact the real world

Settings, Frameworks & Approaches

- Over next couple lectures will consider 2 settings, multiple frameworks, and approaches
- Settings: Bandits (single decisions), MDPs
- Frameworks: evaluation criteria for formally assessing the quality of a RL algorithm. So far seen empirical evaluations, asymptotic convergence, regret
- Approaches: Classes of algorithms for achieving particular evaluation criteria in a certain set. So far for exploration seen: greedy, $\epsilon-$greedy, optimism

Table of Contents

Recall: Multiarmed Bandits

- Multi-armed bandit is a tuple of $(\mathcal{A}, \mathcal{R})$
- $\mathcal{A}$: known set of m actions (arms)
- $\mathcal{R}^a(r) = \mathbb{P}[r \mid a]$ is an unknown probability distribution over rewards
- At each step t the agent selects an action $a_t \in \mathcal{A}$
- The environment generates a reward $r_t \sim \mathcal{R}^{a_t}$
- Goal: Maximize cumulative reward $\sum_{\tau=1}^{t} r_\tau$
- **Regret** is the opportunity loss for one step

$$l_t = \mathbb{E}[V^* - Q(a_t)]$$

- **Total Regret** is the total opportunity loss

$$L_t = \mathbb{E}[\sum_{\tau=1}^{t} V^* - Q(a_\tau)]$$

Table of Contents

Approach: Optimism Under Uncertainty

- Estimate an upper confidence $U_t(a)$ for each action value, such that $Q(a) \leq U_t(a)$ with high probability
- This depends on the number of times $N_t(a)$ action a has been selected
- Select action maximizing Upper Confidence Bound (UCB)
- UCB1 algorithm

$$a_t = \arg\max_{a \in \mathcal{A}} [\hat{Q}_t(a) + \sqrt{\frac{2 \log t}{N_t(a)}}]$$

Hoeffding
reward bounded

- Theorem: The UCB algorithm achieves logarithmic asymptotic total regret

$$\lim_{t \to \infty} L_t \leq 8 \log t \sum_{a|\Delta_a > 0} \Delta_a$$

$E[r(a^*)]$
$- E[r(a)]$

Simpler Optimism?

- Do we need to formally model uncertainty to get the "right" level of optimism?

Greedy Bandit Algorithms and Optimistic Initialization

- Simple optimism under uncertainty approach
  - Pretend already observed one pull of each arm, and saw some optimistic reward
  - Average these fake pulls and rewards in when computing average empirical reward

Greedy Bandit Algorithms and Optimistic Initialization

- Simple optimism under uncertainty approach
  - Pretend already observed one pull of each arm, and saw some optimistic reward
  - Average these fake pulls and rewards in when computing average empirical reward
- Comparing regret results:
- **Greedy**: Linear total regret
- **Constant ϵ-greedy**: Linear total regret
- **Decaying ϵ-greedy**: Sublinear regret if can use right schedule for decaying ϵ, but that requires knowledge of gaps, which are unknown
- **Optimistic initialization**: Sublinear regret if initialize values sufficiently optimistically, else linear regret

exact $\sim \frac{R_{max}}{(1-\frac{1}{n})^{T}}$

Table of Contents

Bayesian Bandits

domain knowledge

- So far we have made no assumptions about the reward distribution $\mathcal{R}$
  - Except bounds on rewards
- **Bayesian bandits** exploit prior knowledge of rewards, $p[\mathcal{R}]$
- They compute posterior distribution of rewards $p[\mathcal{R} \mid h_t]$, where $h_t = (a_1, r_1, \ldots, a_{t-1}, r_{t-1})$
- Use posterior to guide exploration
  - Upper confidence bounds (Bayesian UCB)
  - Probability matching (Thompson Sampling)
- Better performance if prior knowledge is accurate

Short Refresher / Review on Bayesian Inference

- In Bayesian view, we start with a prior over the unknown parameters
  - Here the unknown distribution over the rewards for each arm
- Given observations / data about that parameter, update our uncertainty over the unknown parameters using Bayes Rule

Short Refresher / Review on Bayesian Inference

- In Bayesian view, we start with a prior over the unknown parameters
  - Here the unknown distribution over the rewards for each arm
- Given observations / data about that parameter, update our uncertainty over the unknown parameters using Bayes Rule
- For example, let the reward of arm i be a probability distribution that depends on parameter ϕ_i (unknown)
- Initial prior over ϕ_i is $p(\phi_i)$
- Pull arm i and observe reward r_{i1}
- Use Bayes rule to update estimate over ϕ_i:

$$p(\phi_i | r_{i1}) = \frac{p(r_{i1} | \phi_i) \, p(\phi_i)}{\int p(r_{i1} | \phi_i) p(\phi_i) d\phi_i}$$

Short Refresher / Review on Bayesian Inference

- In Bayesian view, we start with a prior over the unknown parameters
  - Here the unknown distribution over the rewards for each arm
- Given observations / data about that parameter, update our uncertainty over the unknown parameters using Bayes Rule
- For example, let the reward of arm i be a probability distribution that depends on parameter ϕ_i
- Initial prior over ϕ_i is $p(\phi_i)$
- Pull arm i and observe reward r_{i1}
- Use Bayes rule to update estimate over ϕ_i:

$$p(\phi_i|r_{i1}) = \frac{p(r_{i1}|\phi_i)p(\phi_i)}{p(r_{i1})} = \frac{p(r_{i1}|\phi_i)p(\phi_i)}{\int_{\phi_i} p(r_{i1}|\phi_i)p(\phi_i)d\phi_i}$$

Short Refresher / Review on Bayesian Inference II

- In Bayesian view, we start with a prior over the unknown parameters
- Given observations / data about that parameter, update our uncertainty over the unknown parameters using Bayes Rule

$$p(\phi_i|r_{i1}) = \frac{\overbrace{p(r_{i1}|\phi_i)}^{\text{data likelihood}}p(\phi_i)}{\int_{\phi_i} p(r_{i1}|\phi_i)p(\phi_i)d\phi_i}$$

- In general computing this update may be tricky to do exactly with no additional structure on the form of the prior and data likelihood

Short Refresher / Review on Bayesian Inference: Conjugate

- In Bayesian view, we start with a prior over the unknown parameters
- Given observations / data about that parameter, update our uncertainty over the unknown parameters using Bayes Rule

$$p(\phi_i|r_{i1}) = \frac{p(r_{i1}|\phi_i)p(\phi_i)}{\int_{\phi_i} p(r_{i1}|\phi_i)p(\phi_i)d\phi_i}$$

- In general computing this update may be tricky
- But sometimes can be done analytically
- If the parametric representation of the prior and posterior is the same, the prior and model are called **conjugate**.
- For example, exponential families have conjugate priors

Short Refresher / Review on Bayesian Inference: Bernoulli

θ Bernoulli param   unbiased coin $\theta = .5$

- Consider a bandit problem where the reward of an arm is a binary outcome $\{0, 1\}$ sampled from a Bernoulli with parameter θ
  - E.g. Advertisement click through rate, patient treatment succeeds/fails, ...

$p(0)$
$p(1)$

- The Beta distribution *Beta*(α, β) is conjugate for the Bernoulli distribution

$$p(\theta|\alpha, \beta) = \theta^{\alpha-1}(1-\theta)^{\beta-1}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}$$

where $\Gamma(x)$ is the Gamma family.

Short Refresher / Review on Bayesian Inference: Bernoulli

- Consider a bandit problem where the reward of an arm is a binary outcome $\{0, 1\}$ sampled from a Bernoulli with parameter θ
  - E.g. Advertisement click through rate, patient treatment succeeds/fails, ...
- The Beta distribution $Beta(\alpha, \beta)$ is conjugate for the Bernoulli distribution

$$p(\theta|\alpha,\beta) = \theta^{\alpha-1}(1-\theta)^{\beta-1}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}$$

  where $\Gamma(x)$ is the Gamma family.
- Assume the prior over θ is a $Beta(\alpha, \beta)$ as above
- Then after observed a reward $r \in \{0, 1\}$ then updated posterior over θ is $Beta(r + \alpha, 1 - r + \beta)$ # of 0s

  # times we've seen a 1

Bayesian Inference for Decision Making

- Maintain distribution over reward parameters
- Use this to inform action selection

Thompson Sampling

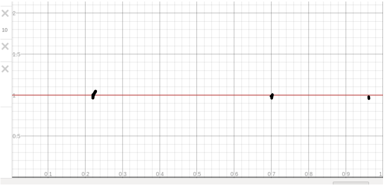
~1919 or 1929

Initialize prior over each arm a, $p(\mathcal{R}_a)$
loop
    For each arm a **sample** a reward distribution $\mathcal{R}_a$ from posterior
    Compute action-value function $Q(a) = \mathbb{E}[\mathcal{R}_a]$
    $a_t = \arg\max_{a \in \mathcal{A}} Q(a)$
    Observe reward r
    Update posterior $p(\mathcal{R}_a | r)$ using Bayes law
end loop

Toy Example: Ways to Treat Broken Toes, Thompson Sampling

- True (unknown) Bernoulli parameters for each arm/action
- Surgery: $\theta_1 = .95$ / Taping: $\theta_2 = .9$ / Nothing: $\theta_3 = .1$
- Thompson sampling:
- Place a prior over each arm's parameter. Here choose Beta(1,1) (Uniform)
  1. Sample a Bernoulli parameter given current prior over each arm Beta(1,1), Beta(1,1), Beta(1,1):

$p(\theta)$



0 θ 1

$\theta \in 0, 1$

uniform distrib

Toy Example: Ways to Treat Broken Toes, Thompson Sampling[1]

- True (unknown) Bernoulli parameters for each arm/action
- Surgery: $\theta_1 = .95$ / Taping: $\theta_2 = .9$ / Nothing: $\theta_3 = .1$
- Thompson sampling:
- Place a prior over each arm's parameter. Here choose Beta(1,1)
  1. Sample a Bernoulli parameter given current prior over each arm Beta(1,1), Beta(1,1), Beta(1,1): 0.3 0.5 0.6
  2. Select $a = \arg\max_{a \in A} Q(a) = \arg\max_{a \in A} \theta(a) = 3$

sample $\theta_1 = .3$

$\theta_2 = .5$

$\theta_3 = .6$

[1] Note: This is a made up example. This is not the actual expected efficacies of the various treatment options for a broken toe

Toy Example: Ways to Treat Broken Toes, Thompson Sampling

- True (unknown) Bernoulli parameters for each arm/action
- Surgery: $\theta_1 = .95$ / Taping: $\theta_2 = .9$ / Nothing: $\theta_3 = .1$
- Thompson sampling:
- Place a prior over each arm's parameter. Here choose θ_i ~Beta(1,1)
  1. Per arm, sample a Bernoulli θ given prior: 0.3 0.5 0.6
  2. Select $a_t = \arg\max_{a \in A} Q(a) = \arg\max_{a \in A} \theta(a) = 3$
  3. Observe the patient outcome's outcome: 0 $r \sim \theta_3$ true
  4. Update the posterior over the $Q(a_t) = Q(a^3)$ value for the arm pulled

Toy Example: Ways to Treat Broken Toes, Thompson Sampling

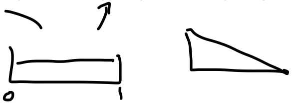
- True (unknown) Bernoulli parameters for each arm/action
- Surgery: $\theta_1 = .95$ / Taping: $\theta_2 = .9$ / Nothing: $\theta_3 = .1$
- Thompson sampling:
- Place a prior over each arm's parameter. Here choose $\theta_i \sim$Beta(1,1)
  1. Sample a Bernoulli parameter given current prior over each arm Beta(1,1), Beta(1,1), Beta(1,1): 0.3 0.5 0.6
  2. Select $a_t = \arg\max_{a \in A} Q(a) = \arg\max_{a \in A} \theta(a) = 3$
  3. Observe the patient outcome's outcome: 0
  4. Update the posterior over the $Q(a_t) = Q(a^3)$ value for the arm pulled
     - Beta(c_1, c_2) is the conjugate distribution for Bernoulli
     - If observe 1, $c_1 + 1$ else if observe 0 $c_2 + 1$
  5. New posterior over Q value for arm pulled is:
  6. New posterior $p(Q(a^3)) = p(\theta(a^3)) = \underline{Beta(1, 2)}$

Toy Example: Ways to Treat Broken Toes, Thompson Sampling

- True (unknown) Bernoulli parameters for each arm/action
- Surgery: $\theta_1 = .95$ / Taping: $\theta_2 = .9$ / Nothing: $\theta_3 = .1$
- Thompson sampling:
- Place a prior over each arm's parameter. Here choose $\theta_i \sim$Beta(1,1)
  1. Sample a Bernoulli parameter given current prior over each arm Beta(1,1), Beta(1,1), Beta(1,1): 0.3 0.5 0.6
  2. Select $a_t = \arg\max_{a \in A} Q(a) = \arg\max_{a \in A} \theta(a) = 3$
  3. Observe the patient outcome's outcome: 0
  4. New posterior $p(Q(a^3)) = p(\theta(a_3) = Beta(1,2)$

Toy Example: Ways to Treat Broken Toes, Thompson Sampling

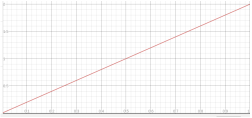
- True (unknown) Bernoulli parameters for each arm/action
- Surgery: $\theta_1 = .95$ / Taping: $\theta_2 = .9$ / Nothing: $\theta_3 = .1$
- Thompson sampling:
- Place a prior over each arm's parameter. Here choose $\theta_i \sim$Beta(1,1)
  1. Sample a Bernoulli parameter given current prior over each arm Beta(1,1), Beta(1,1), Beta(1,2): 0.7, 0.5, 0.3



Toy Example: Ways to Treat Broken Toes, Thompson Sampling

- True (unknown) Bernoulli parameters for each arm/action
- Surgery: $\theta_1 = .95$ / Taping: $\theta_2 = .9$ / Nothing: $\theta_3 = .1$
- Thompson sampling:
- Place a prior over each arm's parameter. Here choose $\theta_i \sim$Beta(1,1)
  1. Sample a Bernoulli parameter given current prior over each arm Beta(1,1), Beta(1,1), Beta(1,2): 0.7, 0.5, 0.3
  2. Select $a_t = \arg\max_{a \in A} Q(a) = \arg\max_{a \in A} \theta(a) = 1$
  3. Observe the patient outcome's outcome: 1 $\sim \theta_1$
  4. New posterior $p(Q(a^1)) = p(\theta(a^1) = Beta(2,1)$

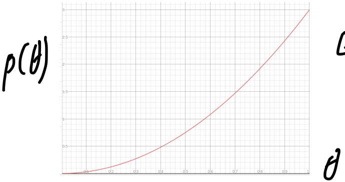
$p(\theta)$



θ

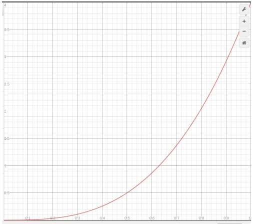
Toy Example: Ways to Treat Broken Toes, Thompson Sampling

- True (unknown) Bernoulli parameters for each arm/action
- Surgery: $\theta_1 = .95$ / Taping: $\theta_2 = .9$ / Nothing: $\theta_3 = .1$
- Thompson sampling:
- Place a prior over each arm's parameter. Here choose $\theta_i \sim$Beta(1,1)
  1. Sample a Bernoulli parameter given current prior over each arm Beta(2,1), Beta(1,1), Beta(1,2): 0.71, 0.65, 0.1
  2. Select $a_t = \arg\max_{a \in A} Q(a) = \arg\max_{a \in A} \theta(a) = 1$
  3. Observe the patient outcome's outcome: 1 θ_1 true
  4. New posterior $p(Q(a^1)) = p(\theta(a^1) = Beta(3,1)$



Toy Example: Ways to Treat Broken Toes, Thompson Sampling

- True (unknown) Bernoulli parameters for each arm/action
- Surgery: $\theta_1 = .95$ / Taping: $\theta_2 = .9$ / Nothing: $\theta_3 = .1$
- Thompson sampling:
- Place a prior over each arm's parameter. Here choose $\theta_i \sim$Beta(1,1)
  1. Sample a Bernoulli parameter given current prior over each arm Beta(2,1), Beta(1,1), Beta(1,2): 0.75, 0.45, 0.4
  2. Select $a_t = \arg\max_{a \in A} Q(a) = \arg\max_{a \in A} \theta(a) = 1$
  3. Observe the patient outcome's outcome: 1
  4. New posterior $p(Q(a^1)) = p(\theta(a^1) = Beta(4, 1)$



Toy Example: Ways to Treat Broken Toes, Thompson Sampling vs Optimism

- Surgery: $\theta_1 = .95$ / Taping: $\theta_2 = .9$ / Nothing: $\theta_3 = .1$
- How does the sequence of arm pulls compare in this example so far?

start up

Optimism	TS	Optimal
a^1	a^3	a1
a^2	a^1	a1
a^3	a^1	a1
a^1	a^1	a1
a^2	a^1	a1

UCB

Toy Example: Ways to Treat Broken Toes, Thompson Sampling vs Optimism

- Surgery: $\theta_1 = .95$ / Taping: $\theta_2 = .9$ / Nothing: $\theta_3 = .1$
- Incurred (frequentist) regret?

Optimism	TS	Optimal	Regret Optimism	Regret TS
a^1	a^3	a^1	0	0.85
a^2	a^1	a^1	0.05	0
a^3	a^1	a^1	0.85	0
a^1	a^1	a^1	0	0
a^2	a^1	a^1	0.05	0

- Now we will see how Thompson sampling works in general, and what it is doing

Table of Contents

Approach: Probability Matching

- Assume we have a parametric distribution over rewards for each arm
- **Probability matching** selects action *a* according to probability that *a* is the optimal action

$$\pi(a \mid h_t) = \mathbb{P}[Q(a) > Q(a'), \forall a' \neq a \mid h_t]$$

- Probability matching is optimistic in the face of uncertainty
    - Uncertain actions have higher probability of being max
- Can be difficult to compute probability that an action is optimal analytically from posterior
- Somewhat incredibly, a simple approach implements probability matching

Thompson Sampling

$.9 \quad .1$

$E = .9 \quad .1$

$Q(a) = E[\theta]$
$= \theta$

Initialize prior over each arm a, $p(\mathcal{R}_a)$
loop
    For each arm a **sample** a reward distribution $\mathcal{R}_a$ from posterior
    Compute action-value function $Q(a) = \mathbb{E}[\mathcal{R}_a]$
    $a_t = \arg\max_{a \in \mathcal{A}} Q(a)$
    Observe reward r
    Update posterior $p(\mathcal{R}_a|r)$ using Bayes law
end loop

Thompson Sampling Implements Probability Matching

$$
\begin{aligned}
\pi(a \mid h_t) &= \mathbb{P}[Q(a) > Q(a'), \forall a' \neq a \mid h_t] \\
&= \underbrace{\mathbb{E}_{\mathcal{R}|h_t}}_{} \left[\mathbb{1}(a = \arg\max_{a \in \mathcal{A}} Q(a)) \right]
\end{aligned}
$$

Framework: Regret and Bayesian Regret

- How do we evaluate performance in the Bayesian setting?
- Frequentist regret assumes a true (unknown) set of parameters

$$Regret(\mathcal{A}, T; \theta) = \sum_{t=1}^{T} \mathbb{E}\left[Q(a^*) - Q(a_t)\right]$$

- Bayesian regret assumes there is a prior over parameters

$$BayesRegret(\mathcal{A}, T; \theta) = \mathbb{E}_{\theta \sim p_\theta}\left[\sum_{t=1}^{T} \mathbb{E}\left[Q(a^*) - Q(a_t)|\theta\right]\right]$$

Bayesian Regret Bounds for Thompson Sampling

- Regret(UCB,T)

$$BayesRegret(TS, T) = E_{\theta \sim p_\theta}\left[\sum_{t=1}^{T} Q(a^*) - Q(a_t)|\theta\right]$$

bandit book ~ 36.1

- Posterior sampling has the same (ignoring constants) regret bounds as UCB

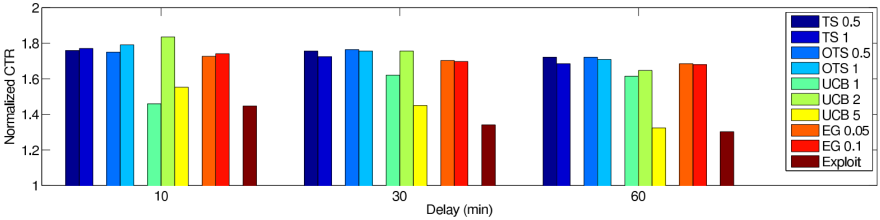
Thompson sampling implements probability matching

- Thompson sampling(1929) achieves Lai and Robbins lower bound
- Bounds for optimism are tighter than for Thompson sampling
- But empirically Thompson sampling can be extremely effective

Thompson Sampling for News Article Recommendation (Chapelle and Li, 2010)

s $\quad s \to a \quad r \quad s \sim p(s)$

- Contextual bandit: input context which impacts reward of each arm, context sampled iid each step
- Arms = articles
- Reward = click (+1) on article ($Q(a)$=click through rate)
- TS did extremely well! Lead to a big resurgence of interest in Thomspon sampling.



Check Your Understanding: Thompson Sampling and Optimism

- Consider an online news website with thousands of people logging on each second. Frequently a new person will come online before we see whether the last person has clicked (or not). Select all that are true:

T
1. Thompson sampling would be better than optimism here, because optimism algorithms are deterministic and would select the same action until we get feedback (click or not).

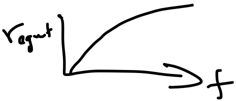
F
2. Optimism algorithms would be better than TS here, because they have stronger regret bounds

T
3. Thompson sampling could cause much worse performance than optimism if the initial prior is very misleading.
4. Not sure

Consider prior Beta(100,1) for a Bernoulli arm with parameter 0.1. Then the prior puts large weight on high values of theta for a long time.

Table of Contents

Framework: Probably Approximately Correct



- Theoretical regret bounds specify how regret grows with T
- Could be making lots of little mistakes or infrequent large ones
- May care about bounding the number of non-small errors
- More formally, probably approximately correct (PAC) results state that the algorithm will choose an action a whose value is ϵ-optimal ($Q(a) \geq Q(a^*) - \epsilon$) with probability at least $1 - \delta$ on all but a polynomial number of steps
- Polynomial in the problem parameters (# actions, ϵ, δ, etc)
- Most PAC algorithms based on optimism or Thompson sampling

bandits
regret

MDPs
regret
PAC

Toy Example: Probably Approximately Correct and Regret

- Surgery: $\theta_1 = .95$ / Taping: $\theta_2 = .9$ / Nothing: $\theta_3 = .1$
- Let $\epsilon = 0.05$.
- O = Optimism, TS = Thompson Sampling: W/in $\epsilon = \mathbb{I}(Q(a_t) \geq Q(a^*) - \epsilon)$

O	TS	Optimal	O Regret	O W/in ϵ	TS Regret	TS W/in ϵ
a^1	a^3	a^1	0	Y	0.85	N
a^2	a^1	a^1	0.05	Y	0	Y
a^3	a^1	a^1	0.85	N	0	Y
a^1	a^1	a^1	0	Y	0	Y
a^2	a^1	a^1	0.05	Y	0	Y

Toy Example: Probably Approximately Correct and Regret

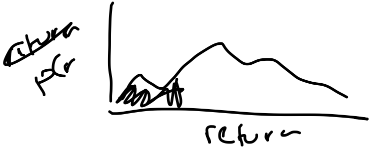
- Surgery: $\theta_1 = .95$ / Taping: $\theta_2 = .9$ / Nothing: $\theta_3 = .1$
- Let $\epsilon = 0.05$.
- O = Optimism, TS = Thompson Sampling: W/in $\epsilon = \mathbb{I}(Q(a_t) \geq Q(a^*) - \epsilon)$

O	TS	Optimal	O Regret	O W/in ϵ	TS Regret	TS W/in ϵ
a^1	a^3	a^1	0	Y	0.85	N
a^2	a^1	a^1	0.05	Y	0	Y
a^3	a^1	a^1	0.85	N	0	Y
a^1	a^1	a^1	0	Y	0	Y
a^2	a^1	a^1	0.05	Y	0	Y

Table of Contents

Fast RL in Markov Decision Processes

- Very similar set of frameworks and approaches are relevant for fast learning in reinforcement learning
- Frameworks
  - Regret
  - Bayesian regret
  - Probably approximately correct (PAC)
- Approaches
  - Optimism under uncertainty
  - Probability matching / Thompson sampling
- Framework: Probably approximately correct



Fast RL in Markov Decision Processes

- Very similar set of frameworks and approaches are relevant for fast learning in reinforcement learning
- Frameworks
  - Regret
  - Bayesian regret
  - Probably approximately correct (PAC)
- Approaches
  - **Optimism under uncertainty**
  - Probability matching / Thompson sampling
- Framework: Probably approximately correct

Optimistic Initialization: Model-Free RL

- Initialize action-value function Q(s,a) optimistically (for ex. $\frac{r_{max}}{1-\gamma}$)
  - where $r_{max} = \max_a \max_s R(s, a)$
  - Check your understanding: why is that value guaranteed to be optimistic?
- Run favorite model-free RL algorithm
  - Monte-Carlo control
  - Sarsa
  - Q-learning ...
- Encourages systematic exploration of states and actions

Optimistic Initialization: Model-Free RL

- Initialize action-value function Q(s,a) optimistically (for ex. $\frac{r_{max}}{1-\gamma}$)
  - where $r_{max} = \max_a \max_s R(s, a)$
- Run model-free RL algorithm: MC control, Sarsa, Q-learning ...
- In general the above have no guarantees on performance, but may work better than greedy or ϵ-greedy approaches

$\frac{r_{max}}{1-\gamma}$

- Even-Dar and Mansour (NeurIPS 2002) proved that
  - If run Q-learning with learning rates a_i on time step i,
  - If initialize $V(s) = \frac{r_{max}}{(1-\gamma)\prod_{i=1}^{T} \alpha_i}$ where α_i is the learning rate on step i and T is the number of samples need to learn a near optimal Q
  - Then greedy-only Q-learning is PAC
- Recent work (Jin, Allen-Zhu, Bubeck, Jordan NeurIPS 2018) proved that (much less) optimistically initialized Q-learning has good (though not tightest) regret bounds

tabular

Approaches to Model-based Optimism for Provably Efficient RL

1. Be very optimistic until confident that empirical estimates are close to true (dynamics/reward) parameters (Brafman & Tennenholtz JMLR 2002)
2. Be optimistic given the information have
   - Compute confidence sets on dynamics and reward models, or
   - Add reward bonuses that depend on experience / data] deep RL

- We will focus on the last class of approaches

Summary so Far: Settings, Frameworks & Approaches

[Note: in class didn't go through slides 55-61]

- Over 3 lectures will consider 2 settings, multiple frameworks, and approaches
- Settings: Bandits (single decisions), MDPs
- Frameworks: evaluation criteria for formally assessing the quality of a RL algorithm. So far seen empirical evaluations, asymptotic convergence, regret, probably approximately correct (PAC)
- Approaches: Classes of algorithms for achieving particular evaluation criteria in a certain set. So far for exploration seen: greedy, $\epsilon-$greedy, optimism, Thompson sampling