Lecture 13: Fast Reinforcement Learning

Emma Brunskill

CS234 Reinforcement Learning

Winter 2026

- With some slides from or derived from David Silver, Examples new

Refresh Your Understanding: Deterministic Bandits

- Which of the following statements is true in a bandit problem with deterministic rewards?
  1. Upper Confidence Bound (UCB) will have sublinear regret with probability one
  2. Upper Confidence Bound (UCB) will have linear regret with some strictly positive probability
  3. a division by zero will occur in the Upper Confidence Bound (UCB) algorithm and so it cannot handle deterministic environments
  4. Upper Confidence Bound (UCB) will always exhibit more regret than in a stochastic environment because it will over-explore.
  5. Not Sure

Refresh Your Understanding: Deterministic Bandits

- Which of the following statements is true in a bandit problem with deterministic rewards?
  1. Upper Confidence Bound (UCB) will have sublinear regret with probability one
  2. Upper Confidence Bound (UCB) will have linear regret with some strictly positive probability
  3. a division by zero will occur in the Upper Confidence Bound (UCB) algorithm and so it cannot handle deterministic environments
  4. Upper Confidence Bound (UCB) will always exhibit more regret than in a stochastic environment because it will over-explore.
  5. Not Sure

Class Survey Results

- Thank you to everyone that filled it in!
- The Good: Tutorials (+40), Check your understandings & lectures and math
- Things to Improve: Lectures and math (want more high level structure and conceptual understanding and examples)
- What I Will Change: Will try to emphasize conceptual ideas and provide concrete examples

Where We are

- Last time: Bandits and regret and UCB (fast learning)
- This time: Bandits, regret and Bayesian bandits (fast learning)

Today

- Bayesian bandits
- Thompson sampling
- Bayesian Regret

Multiarmed Bandits Notation Recap

- 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

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

- Maximize cumulative reward $\iff$ minimize total regret

Today

- **Bayesian bandits**
- Thompson Sampling
- Bayesian Regret

Bayesian bandits

- So far we have made no assumptions about the reward distribution $\mathcal{R}$
  - Except bounds on rewards
- **Bayesian bandits** exploit prior knowledge of rewards, $p[\mathcal{R}]$

Short Refresher on / Introduction to 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 on / Introduction to 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:

Short Refresher on / Introduction to 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 on / Introduction to Bayesian Inference

- In Bayesian view, we start with a prior over the unknown parameters
- Give 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 to do exactly with no additional structure on the form of the prior and data likelihood

Short Refresher on / Introduction to Bayesian Inference: Conjugate

- In Bayesian view, we start with a prior over the unknown parameters
- Give 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 on / Introduction to Bayesian Inferencee: 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 success/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

Short Refresher on / Introduction to 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 success/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 $Beta(\alpha, \beta)$ as above
- Then after observed a reward $r \in \{0, 1\}$ then updated posterior over θ is $Beta(r + \alpha, 1 - r + \beta)$

Bayesian Inference for Decision Making

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

Bayesian bandits Overview

- 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 $h_t = (a_1, r_1, \dots, a_{t-1}, r_{t-1})$
- Use posterior to guide exploration
    - Upper confidence bounds (Bayesian UCB)
    - Probability matching (Thompson Sampling)
- Better performance if prior knowledge is accurate

Today

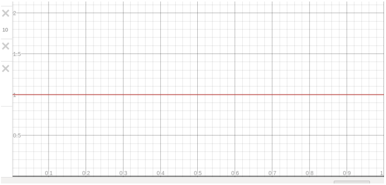
- Bandits and Probably Approximately Correct
- Bayesian bandits
- Thompson Sampling
- Bayesian Regret

Thompson Sampling

Initialize prior over each arm a, $p(\mathcal{R}_a)$
for iteration$=1, 2, \ldots$ **do**
    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 $p(\mathcal{R}_a)$ using Bayes Rule
end for

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):



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 $a = \arg\max_{a \in A} Q(a) = \arg\max_{ainA} \theta(a) =$

[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

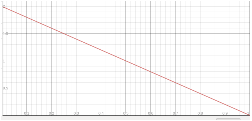
- True (unknown) Bernoulli parameters for each arm/action
- Surgery: $\theta_1 = .95$ / Taping: $\theta_2 = .9$ / Nothing: $\theta_3 = .1$
- Thompson sampling:
- Place a prior over each arm's parameter. Here choose $\theta_i \sim$Beta(1,1)
  1. Per arm, sample a Bernoulli θ given prior: 0.3 0.5 0.6
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

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 $a_t = \arg\max_{a \in A} Q(a) = \arg\max_{ainA} \theta(a) = 3$
  3. Observe the patient outcome's outcome: 0
  4. Update the posterior over the $Q(a_t) = Q(a^1)$ 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 $p(Q(a^3)) = p(\theta(a_3) = Beta(1,2)$

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 $a_t = \arg\max_{a \in A} Q(a) = \arg\max_{ainA} \theta(a) = 1$
  3. Observe the patient outcome's outcome: 0
  4. New posterior $p(Q(a^1)) = p(\theta(a_1) = Beta(1,2)$



Toy Example: Ways to Treat Broken Toes, Thompson Sampling

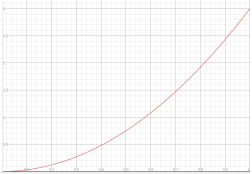
- True (unknown) Bernoulli parameters for each arm/action
- Surgery: $\theta_1 = .95$ / Taping: $\theta_2 = .9$ / Nothing: $\theta_3 = .1$
- Thompson sampling:
- Place a prior over each arm's parameter. Here choose $\theta_i \sim$Beta(1,1)
  1. Sample a Bernoulli parameter given current prior over each 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 $a_t = \arg\max_{a \in A} Q(a) = \arg\max_{ainA} \theta(a) = 1$
  3. Observe the patient outcome's outcome: 1
  4. New posterior $p(Q(a^1)) = p(\theta(a_1) = Beta(2,1)$

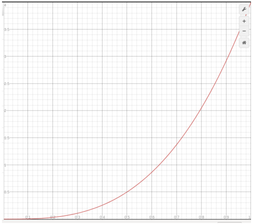
Toy Example: Ways to Treat Broken Toes, Thompson Sampling

- 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): 0.71, 0.65, 0.1
  2. Select $a_t = \arg\max_{a \in A} Q(a) = \arg\max_{ainA} \theta(a) = 1$
  3. Observe the patient outcome's outcome: 1
  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 $a_t = \arg\max_{a \in A} Q(a) = \arg\max_{ainA} \theta(a) = 1$
  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?

Optimism	TS
a^1	a^3
a^2	a^1
a^3	a^1
a^1	a^1
a^2	a^1

On to General Setting for Thompson Sampling

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

Probability Matching

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

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

- Can be difficult to compute probability that an action is optimal analytically from posterior
- Somewhat incredibly, Thompson sampling implements probability matching

Thompson sampling implements probability matching

Thompson sampling implements probability matching

- Thompson sampling:

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

Probability Matching

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

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

- Can be difficult to compute probability that an action is optimal analytically from posterior
- Somewhat incredibly, Thompson sampling implements probability matching
- Note: probability matching is often optimistic in the face of uncertainty
  - Uncertain actions have higher probability of being max

Today

- Bandits and Probably Approximately Correct
- Bayesian bandits
- Thompson Sampling
- Bayesian Regret

Framework: Regret and Bayesian Regret

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

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

where $\mathbb{E}_\tau$ denotes an expectation with respect to the history of actions taken and rewards observed given an algorithm $\mathcal{A}$.

- Bayesian regret assumes there is a prior over parameters

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

Bounding Regret Using Optimism

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

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

where $\mathbb{E}_\tau$ denotes an expectation with respect to the history of actions taken and rewards observed given an algorithm $\mathcal{A}$ (under event that U_t is an upper bound).

Thompson sampling

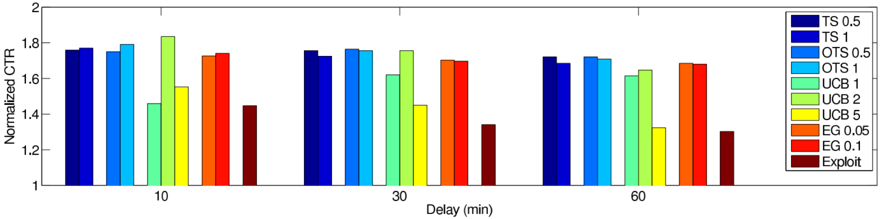
- Frequentist bounds for standard* Thompson sampling do not* (last checked) match best bounds for frequentist algorithms
- Empirically Thompson sampling can be effective, especially in contextual multi-armed bandits

Contextual Bandit Thompson Sampling

- 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)

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

- 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)



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:
  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)
  2. Optimism algorithms would be better than TS here, because they have stronger regret bounds for this setting
  3. Thompson sampling could cause much worse performance than optimism if the initial prior is very misleading.
  4. Not sure

Check Your Understanding: Thompson Sampling and Optimism **Solutions**

- 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:
  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)
  2. Optimism algorithms would be better than TS here, because they have stronger regret bounds for this setting
  3. Thompson sampling could cause much worse performance than optimism if the initial prior is very misleading.
  4. Not sure

Optimal Policy for Bayesian bandits?

- Thompson Sampling often works well, but is it optimal?
- Given prior, and known horizon, could compute decision policy that would maximize expected rewards given the available horizon
- Computational challenge: naively this would create a decision policy that is a function of the history to the next arm to pull

Gittins Index for Bayesian bandits

- Thompson Sampling often works well, but is it optimal?
- Given prior, and known horizon, could compute decision policy that would maximize expected rewards given the available horizon
- Computational challenge: naively this would create a decision policy that is a function of the history to the next arm to pull
- **Index policy**: a decision policy that computes a "real-valued index for each arm and plays the arm with the largest index," using statistics only from that arm and the horizon (definition from Lattimore and Svespari 2019 Bandit Algorithms)
- **Gittins index**: optimal policy for maximizing expected discounted reward in a Bayesian multi-armed bandit

Today

- Bandits and Probably Approximately Correct
- Bayesian bandits
- Thompson Sampling
- Bayesian Regret

What You Should Understand

- Understand how multi-armed bandits relate to MDPs
- Be able to define regret and PAC
- Be able to prove why UCB bandit algorithm has sublinear regret
- Understand (be able to give an example) why e-greedy and greedy and pessimism can result in linear regret
- Be able to implement UCB bandit algorithm
- Be able to implement Thompson Sampling for Bernoulli rewards

Where We are

- Last time: Bandits and regret and UCB (fast learning)
- This time: Bayesian bandits (fast learning)

Bayesian Regret Bounds for Thompson Sampling

- Regret(UCB,T)

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

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

Toy Example: Probably Approximately Correct and Regret

- Surgery: $\phi_1 = .95$ / Taping: $\phi_2 = .9$ / Nothing: $\phi_3 = .1$
- 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