

Exploration-Exploitation

DAVIDE BACCIU – BACCIU@DI.UNIPI.IT



UNIVERSITÀ DI PISA

Outline

- ✓ Introduction
- ✓ Exploration and Exploitation
 - ✓ Simple naïve exploration (ϵ -greedy)
 - ✓ Optimistic approaches
 - ✓ Probability matching & Information Value
- ✓ Bandits
 - ✓ Multi-armed
 - ✓ Contextual
- ✓ Back to MDPs (briefly)

Introduction

Exploration-Exploitation Dilemma

- ✓ Online decision-making involves a fundamental choice:
 - ✓ **Exploitation** - Make the best decision given current information
 - ✓ **Exploration** - Gather more information
- ✓ The best long-term strategy may involve short-term sacrifices
- ✓ Gather enough information to make the best overall decisions

Examples

- ✓ Restaurant Selection
 - ✓ **Exploitation** - Go to your favourite restaurant
 - ✓ **Exploration** - Try a new restaurant
- ✓ Holiday planning
 - ✓ **Exploitation** – The camping site you go to since you are born
 - ✓ **Exploration** – Hitchhike and follow the flow
- ✓ Game Playing
 - ✓ **Exploitation** - Play the move you believe is best
 - ✓ **Exploration** - Play an experimental move

Principles

- ✓ Random Exploration

- ✓ Add noise to greedy policy (e.g. ϵ -greedy)

- ✓ Optimism in the Face of Uncertainty

- ✓ Estimate uncertainty on value
- ✓ Prefer to explore states/actions with highest uncertainty

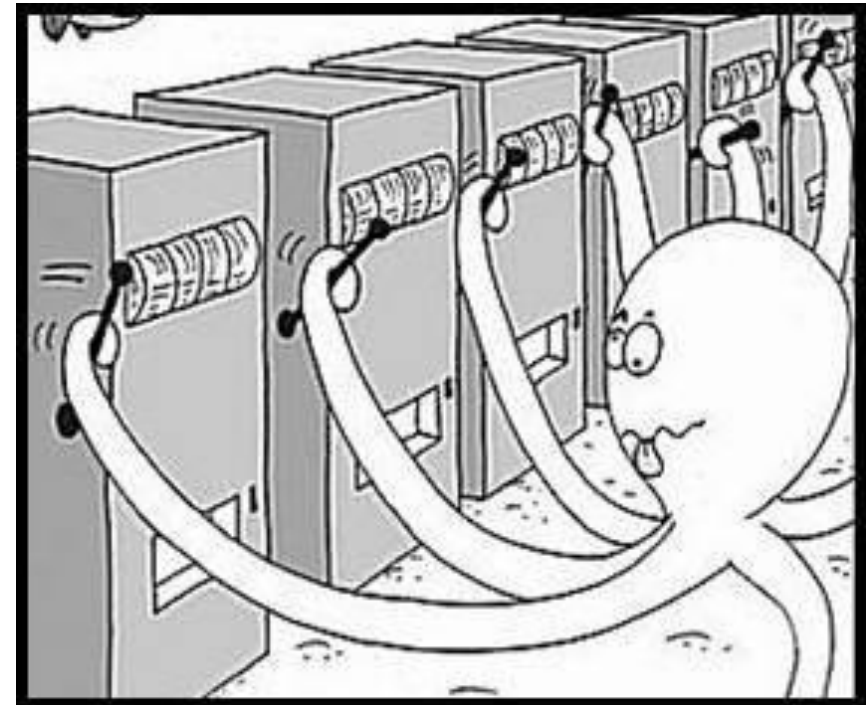
- ✓ Information State Search

- ✓ Consider agent's information as part of its state
- ✓ Lookahead to see how information helps rewards

Bandits

Multi-Armed Bandit

- ✓ A multi-armed bandit is a tuple $\langle \mathcal{A}, \mathcal{R} \rangle$
 - ✓ \mathcal{A} is a known set of m actions (or “arms”)
 - ✓ $\mathcal{R}^a(r) = P(r|a)$ 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 $r_t \sim \mathcal{R}^{a_t}$
- ✓ The goal is to maximise cumulative reward $\sum_{\tau=1}^t r_{\tau}$



Regret

- ✓ The **action-value** is the mean reward for action a

$$Q(a) = \mathbb{E}[r|a]$$

- ✓ The **optimal value** V^* is

$$V^* = Q(a^*) = \max_{a \in \mathcal{A}} Q(a)$$

- ✓ The **regret** is the opportunity loss for one step

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

- ✓ The total regret is the total opportunity loss

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

- ✓ Maximise cumulative reward \equiv minimise total regret

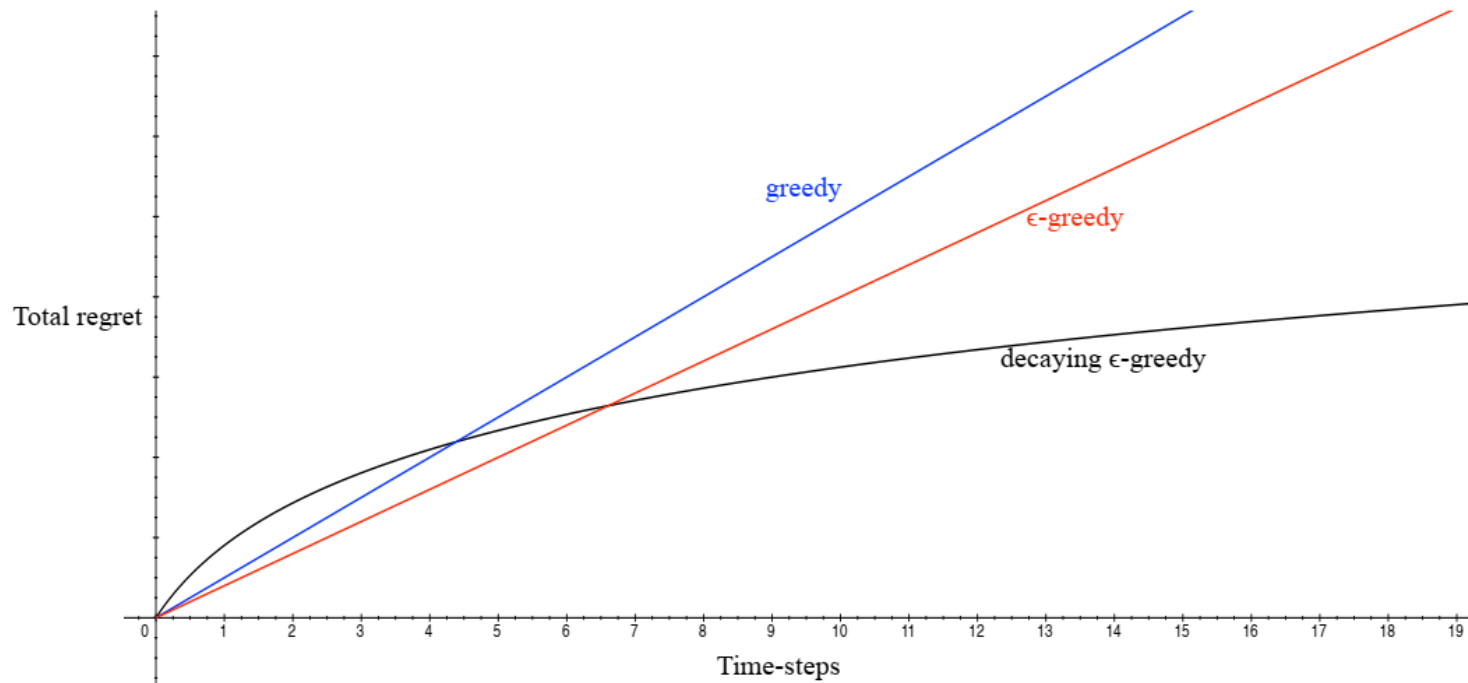
Counting Regret

- ✓ The count $N_t(a)$ is **expected number of selections for action a**
- ✓ The gap $\Delta_a = V^* - Q(a)$ is the **difference in value between action a and optimal action a^***
- ✓ Regret is a function of gaps and the counts

$$L_t = \mathbb{E} \left[\sum_{\tau=1}^t V^* - Q(a_\tau) \right] = \sum_{a \in \mathcal{A}} \mathbb{E}[N_t(a)] (V^* - Q(a)) = \sum_{a \in \mathcal{A}} \mathbb{E}[N_t(a)] \Delta_a$$

- ✓ A good algorithm ensures **small counts for large gaps**
- ✓ **Problem:** gaps are not known!

Linear or Sublinear Regret



- ✓ If an algorithm **forever explores** it will have linear total regret
- ✓ If an algorithm **never explores** it will have linear total regret
- ✓ Is it possible to achieve sublinear total regret?

Exploration Strategies

Greedy Algorithms

- ✓ We consider algorithms that estimate $\hat{Q}_t(a) \approx Q(a)$
- ✓ Estimate the value of each action by Monte-Carlo evaluation

$$\hat{Q}_t(a) = \frac{1}{N_t(a)} \sum_{\tau} r_{\tau} \mathbf{1}(a_{\tau}; a)$$

- ✓ The greedy algorithm selects action with highest value

$$a_t^* = \arg \max_{a \in \mathcal{A}} \hat{Q}_t(a)$$

- ✓ Greedy can lock onto a suboptimal action forever

Greedy has linear total regret

ϵ -greedy Algorithms

- ✓ The ϵ -greedy algorithm continues to explore forever
 - ✓ With probability $1 - \epsilon$ select $a = \max_{a \in \mathcal{A}} \hat{Q}(a)$
 - ✓ With probability ϵ select a random action
- ✓ Constant ϵ ensures minimum regret

$$L_t \geq \frac{\epsilon}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \Delta_a$$

ϵ -greedy has linear total regret

Decaying ϵ_t -greedy Algorithms

- ✓ Pick a decay schedule for $\epsilon_1, \epsilon_2, \dots$
- ✓ Consider the following schedule

$$c > 0$$
$$d = \min_{a|\Delta_a > 0} \Delta_i$$
$$\epsilon_t = \min \left\{ 1, \frac{c|\mathcal{A}|}{d^2 t} \right\}$$

- ✓ Decaying ϵ_t -greedy has **logarithmic asymptotic total regret**
- ✓ Unfortunately, schedule requires **advance knowledge of gaps**
- ✓ **Goal:** find an algorithm with sublinear regret for any multi-armed bandit (without knowledge of R)

Lower Bound

- ✓ The performance of any algorithm is **determined by similarity between optimal arm and other arms**
- ✓ Hard problems have similar-looking arms with different means
- ✓ This is described formally by the gap Δ_a and the similarity in distributions $KL(\mathcal{R}^a || \mathcal{R}^{a^*})$

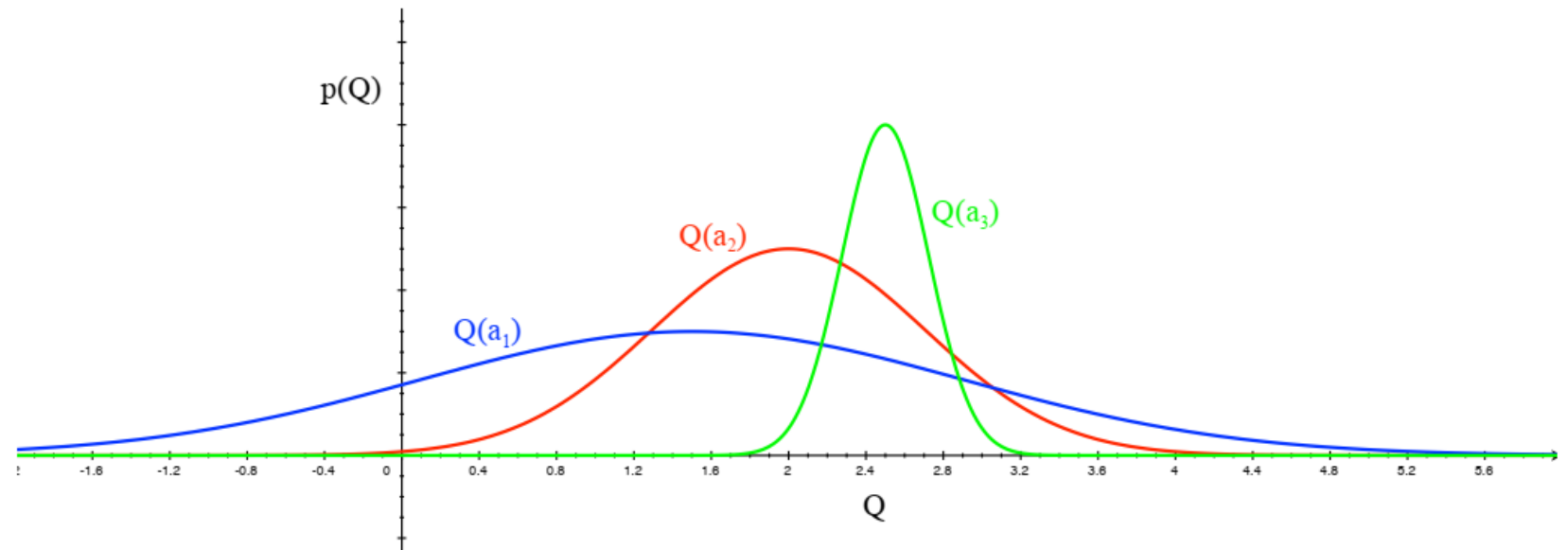
Theorem (Lai and Robbins)

Asymptotic total regret is at least logarithmic in the number of steps

$$\lim_{t \rightarrow \infty} L_t \geq \log t \sum_{a | \Delta_a > 0} \frac{\Delta_a}{KL(\mathcal{R}^a || \mathcal{R}^{a^*})}$$

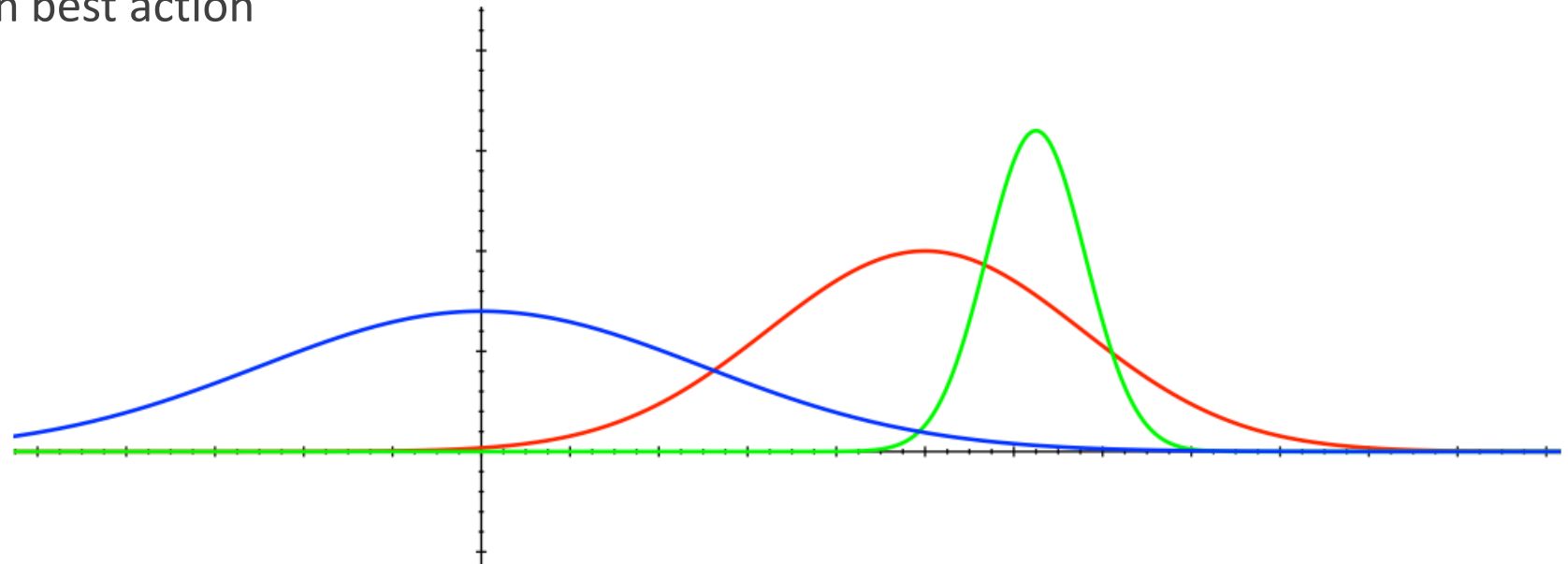
Optimism in the Face of Uncertainty (I)

- ✓ Which action should we pick?
- ✓ The more uncertain we are about an action-value
- ✓ The more important it is to explore that action
- ✓ It could turn out to be the best action



Optimism in the Face of Uncertainty (II)

- ✓ After picking blue action
- ✓ We are less uncertain about the value
- ✓ And more likely to pick another action
- ✓ Until we home in on best action



Upper Confidence Bounds

- ✓ Estimate an **upper confidence** $\hat{U}_t(a)$ for each action value
- ✓ Such that $Q(a) \leq \hat{Q}_t(a) + \hat{U}_t(a)$ with high probability
- ✓ This depends on the number of times $N(a)$ has been selected
 - ✓ Small $N_t(a) \Rightarrow$ large $\hat{U}_t(a)$ (estimated value is **uncertain**)
 - ✓ Large $N_t(a) \Rightarrow$ small $\hat{U}_t(a)$ (estimated value is **accurate**)
- ✓ Select action maximising **Upper Confidence Bound (UCB)**

$$a_t = \arg \max_{a \in \mathcal{A}} \hat{Q}_t(a) + \hat{U}_t(a)$$

Hoeffding's Inequality

Theorem (Hoeffding's Inequality)

Let X_1, \dots, X_t be i.i.d random variables in $[0,1]$ and let $\bar{X}_t = \frac{1}{t} \sum_{\tau=1}^t x_\tau$ be the sample mean. Then

$$P(\mathbb{E}[X] > \bar{X}_t + u) \leq e^{-2tu^2}$$

We will apply Hoeffding's Inequality to rewards of the bandit conditioned on selecting action a

$$P(Q(a) > \hat{Q}_t(a) + \hat{U}_t(a)) \leq e^{-2N_t(a)\hat{U}_t(a)^2}$$

Calculating Upper Confidence Bounds

- ✓ Pick a probability p that true value exceeds UCB
- ✓ Now solve for $U_t(a)$

$$e^{-2N_t(a)U_t(a)^2} = p$$
$$U_t(a) = \sqrt{\frac{-\log p}{2N_t(a)}}$$

- ✓ Reduce p as we observe more rewards, e.g. $p = t^{-4}$
- ✓ Ensures we select optimal action as $t \rightarrow \infty$

$$U_t(a) = \sqrt{\frac{2\log t}{N_t(a)}}$$

UCB1

- ✓ This leads to the **UCB1 algorithm**

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

- ✓ The UCB algorithm achieves **logarithmic asymptotic total regret**

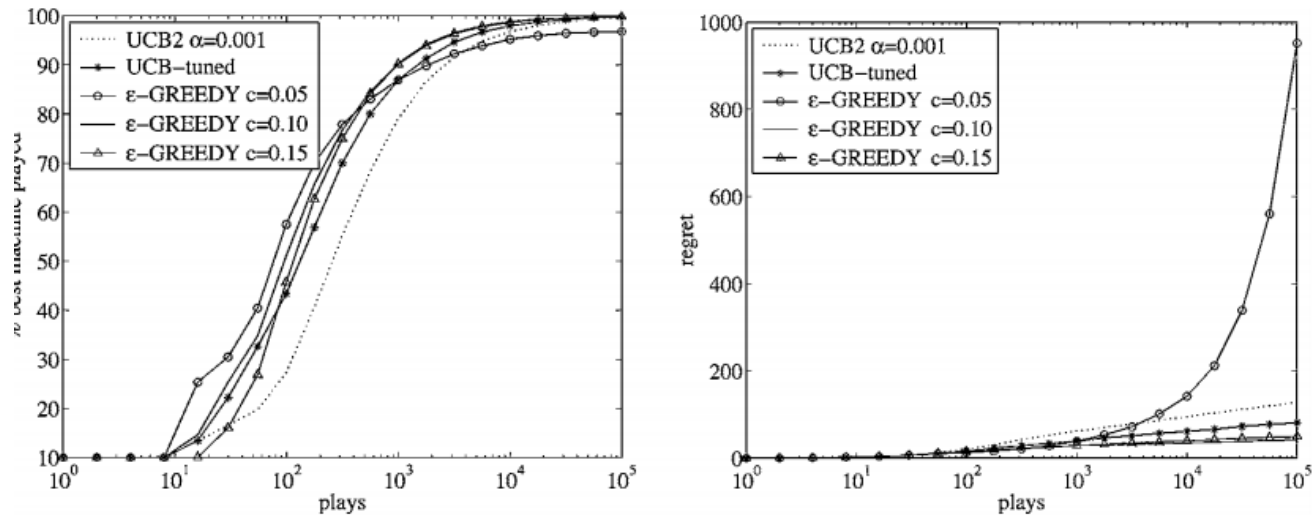
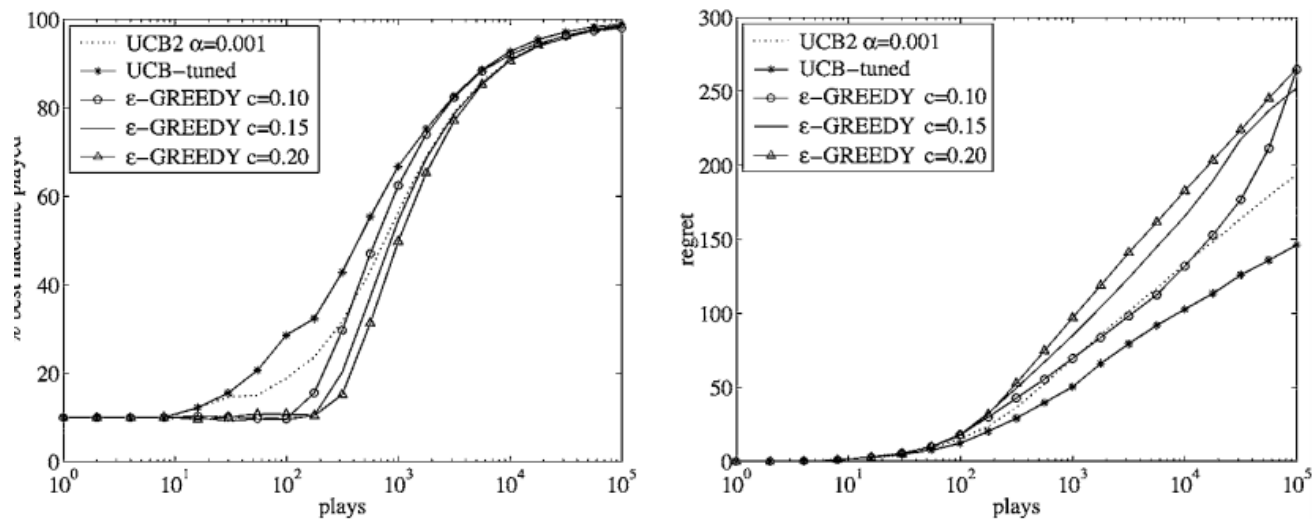


Figure 9. Comparison on distribution 11 (10 machines with parameters 0.9, 0.6, ..., 0.6).



UCB vs. ϵ -greedy on 10-armed Bandit

Information State

Value of Information

- ✓ Exploration is useful because it gains information
- ✓ Can we quantify the value of information?
 - ✓ How much reward a decision-maker would be prepared to pay in order to have that information, prior to making a decision
 - ✓ Long-term reward after getting information - immediate reward
- ✓ Information gain is higher in uncertain situations
- ✓ Therefore it makes sense to explore uncertain situations more
- ✓ If we know value of information, we can trade-off exploration and exploitation optimally

Information State Space

- ✓ We have viewed bandits as one-step decision-making problems
- ✓ Can also **view as sequential decision-making** problems
- ✓ At each step there is an information state \tilde{s}
 - ✓ \tilde{s} is a statistic of the history, i.e. $\tilde{s} = f(h_t)$
 - ✓ summarizes all information accumulated so far
- ✓ Each **action a causes a transition to a new information state \tilde{s}'** (and adds information) with probability $\tilde{P}_{\tilde{s}, \tilde{s}'}^a$
- ✓ Defines an **MDP $\tilde{\mathcal{M}}$ in augmented information state space**

$$\tilde{\mathcal{M}} = \langle \tilde{\mathcal{S}}, \mathcal{A}, \tilde{\mathcal{P}}, \mathcal{R}, \gamma \rangle$$

Example - Bernoulli Bandits

- ✓ Consider a Bernoulli bandit, such that $\mathcal{R}_a = \mathcal{B}(\mu_a)$ (e.g. win or lose a game with probability μ_a)
- ✓ Want to find which arm has the highest μ_a
- ✓ The information state is $\tilde{s} = \langle \alpha, \beta \rangle$
 - ✓ α_a counts the pulls of arm a where reward was 0
 - ✓ β_a counts the pulls of arm a where reward was 1

Solving Information State Space Bandits

- ✓ We now have an infinite MDP over information states that can be solved by reinforcement learning
- ✓ **Model-free** reinforcement learning
 - ✓ e.g. Q-learning (Duff, 1994)
- ✓ **Bayesian model-based** reinforcement learning
 - ✓ e.g. Gittins indices (Gittins, 1979)
 - ✓ This approach is known as **Bayes-adaptive RL**
 - ✓ Finds **Bayes-optimal exploration/exploitation trade-off** with respect to prior distribution

Contextual Bandits

Contextual Bandits

- ✓ A contextual bandit is a tuple $\langle \mathcal{A}, \mathcal{S}, \mathcal{R} \rangle$
- ✓ $\mathcal{S} = P(s)$ is an unknown distribution over states (contexts)
- ✓ $\mathcal{R}_s^a(r) = P(r|s, a)$ is an unknown probability distribution over rewards
- ✓ At each step t
 - ✓ Environment generates state $s_t \sim \mathcal{S}$
 - ✓ Agent selects action $a_t \in \mathcal{A}$
 - ✓ Environment generates reward $r_t \sim \mathcal{R}_{s_t}^{a_t}$



Linear Regression

- ✓ Action-value function is expected reward for state s and action a

$$Q(s, a) = \mathbb{E}[r|s, a]$$

- ✓ Estimate value function with a linear function approximator

$$Q_\theta(s, a) = \phi(s, a)^T \theta \approx Q(s, a)$$

- ✓ Estimate parameters by least squares regression

$$A_t = \sum_{\tau=1}^t \phi(s_\tau, a_\tau) \phi(s_\tau, a_\tau)^T$$

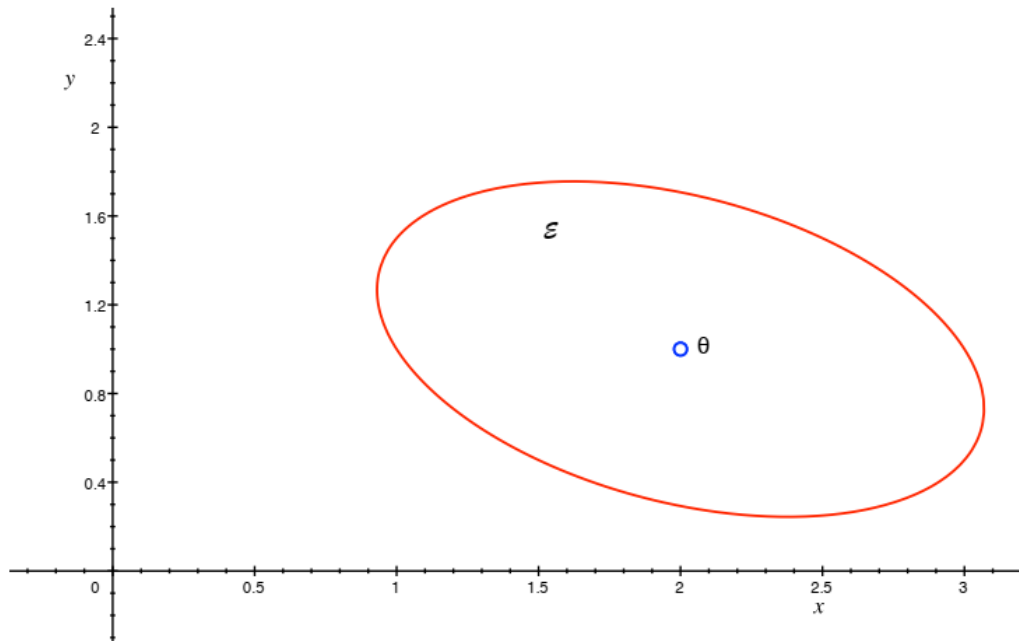
$$b_t = \sum_{\tau=1}^t \phi(s_\tau, a_\tau) r_\tau$$

$$\theta_t = A_t^{-1} b_t$$

Linear Upper Confidence Bounds

- ✓ Least squares regression estimates the mean action-value $Q_{\theta}(s, a)$
- ✓ But it **can also estimate the variance** of the action-value $\sigma_{\theta}^2(s, a)$
 - ✓ i.e. the uncertainty due to parameter estimation error
- ✓ Add on a **bonus for uncertainty**, $U_{\theta}(s, a) = c\sigma$
 - ✓ i.e. define UCB to be c standard deviations above the mean

Geometric Interpretation



- ✓ Define confidence ellipsoid ε_t around parameters θ_t
- ✓ Such that ε_t includes true parameters θ^* with high probability
- ✓ Use this ellipsoid to estimate the uncertainty of action values
- ✓ Pick parameters within ellipsoid that maximise action value

$$\arg \max_{\theta \in \varepsilon} Q_{\theta}(s, a)$$

Calculating Linear Upper Confidence Bounds (LinUCB)

✓ For least squares regression, parameter covariance is A^{-1}

✓ Action-value is linear in features

$$Q_{\theta}(s, a) = \phi(s, a)^T \theta$$

✓ So action-value **variance is quadratic**

$$\sigma_{\theta}^2(s, a) = \phi(s, a)^T A^{-1} \phi(s, a)$$

✓ Upper **confidence bound** is

$$Q_{\theta}(s, a) + c \sqrt{\phi(s, a)^T A^{-1} \phi(s, a)}$$

✓ Select **action maximising upper confidence bound**

$$a_t = \arg \max_{a \in \mathcal{A}} Q_{\theta}(s_t, a) + c \sqrt{\phi(s_t, a)^T A^{-1} \phi(s_t, a)}$$

Exploration-Exploitation in MDPs

Applying Exploration/Exploitation to MDPs

The same principles for exploration/exploitation apply to MDPs

- ✓ Naive Exploration
- ✓ Optimism in the Face of Uncertainty
- ✓ Probability Matching
- ✓ Information State Search

Upper Confidence Bounds - Model-Free RL

- ✓ Maximise **UCB on action-value function** $Q^\pi(s, a)$

$$a_t = \arg \max_{a \in \mathcal{A}} Q(s_t, a) + U(s_t, a)$$

- ✓ Estimate uncertainty in policy evaluation
- ✓ Ignore uncertainty from policy improvement

- ✓ Maximise **UCB on optimal action-value function** $Q^*(s, a)$

$$a_t = \arg \max_{a \in \mathcal{A}} Q(s_t, a) + U_1(s_t, a) + U_2(s_t, a)$$

- ✓ Estimate **uncertainty in policy evaluation**
- ✓ Estimate **uncertainty from policy improvement**

Information State Search in MDPs

✓ MDPs can be augmented to include information state

✓ Now the augmented state is $\langle s, \tilde{s} \rangle$

✓ s is original state within MDP

✓ \tilde{s} is a statistic of the history

✓ Each action a causes a transition

✓ to a new state s' with probability $\mathcal{P}_{s,s'}^a$

✓ to a new information state \tilde{s}'

✓ Defines MDP $\tilde{\mathcal{M}}$ in augmented information state space

$$\tilde{\mathcal{M}} = \langle \tilde{\mathcal{S}}, \mathcal{A}, \tilde{\mathcal{P}}, \mathcal{R}, \gamma \rangle$$

✓ Posterior distribution over MDP model is an information state

$$\tilde{s} = P(\mathcal{P}, \mathcal{R} | h_t)$$

✓ Solve this MDP to find optimal exploration/exploitation trade-off (with respect to prior)

Wrap-up

Take home messages

- ✓ A selection of principles for exploration/exploitation
 - ✓ Naive methods (ϵ -greedy)
 - ✓ Upper confidence bounds
 - ✓ Information state search
- ✓ Principles developed in bandit setting but also apply to MDP setting

Coming up

Imitation Learning

- ✓ Demonstration techniques
- ✓ Inverse reinforcement learning
- ✓ Reinforcement learning with generative models