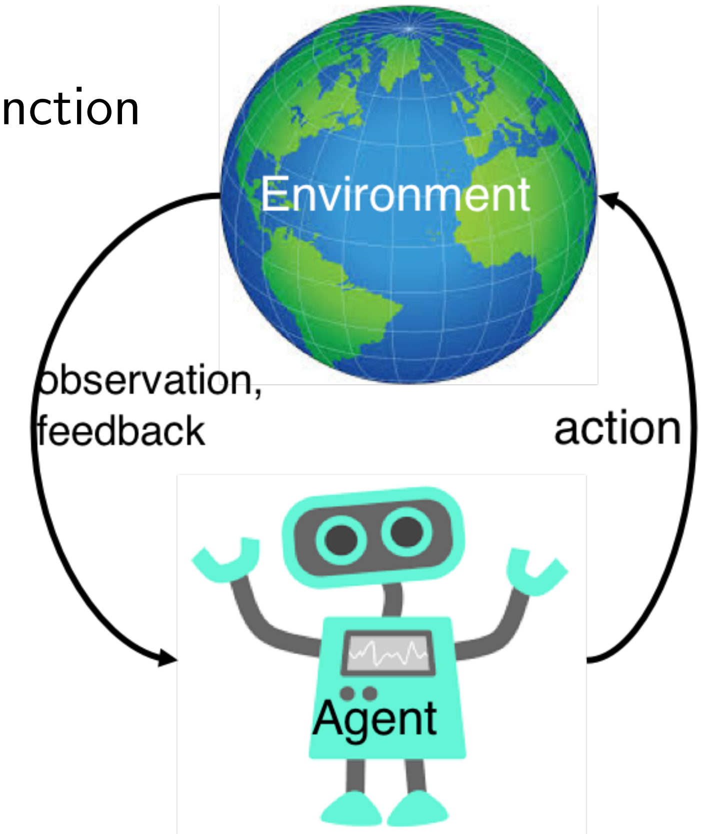
Lecture 9: Stochastic Bandits

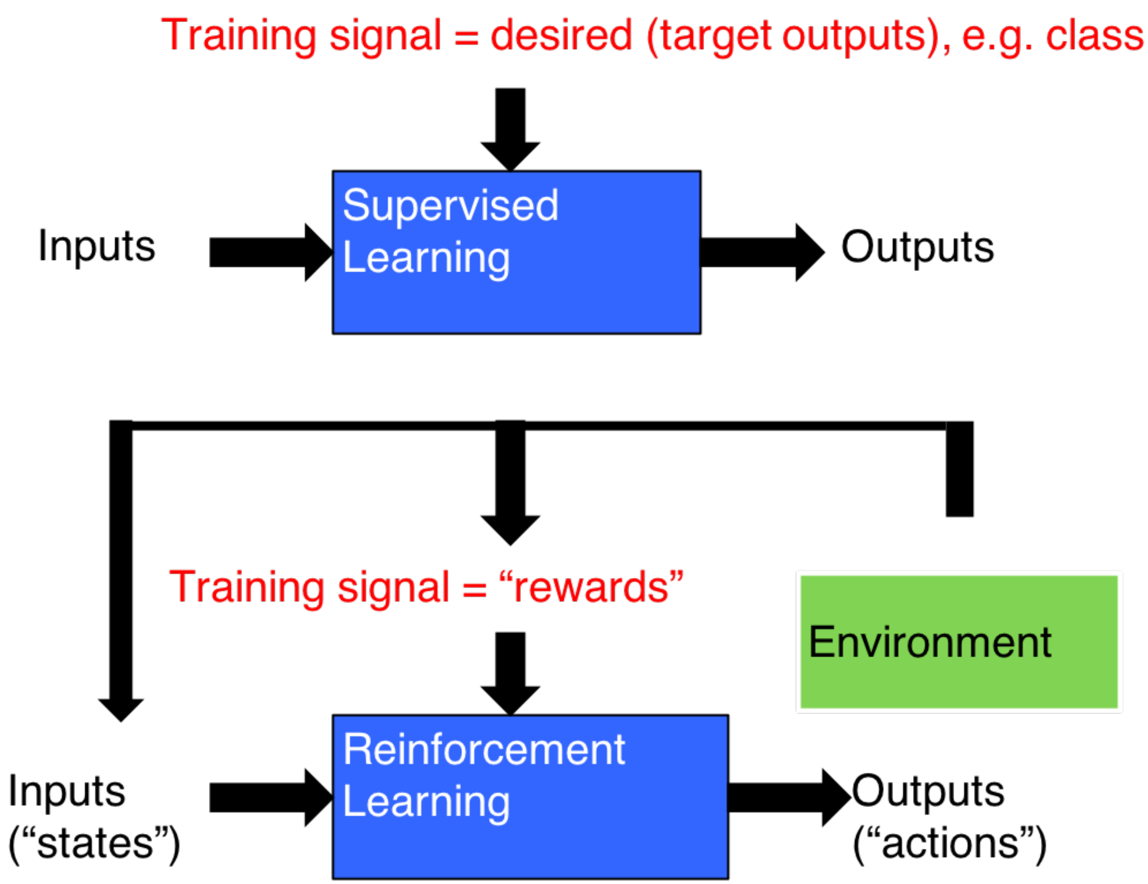
- Multi-armed bandits
- ε-greedy
- Upper Confidence Bounds
- Thompson Sampling

Reinforcement Learning (RL)

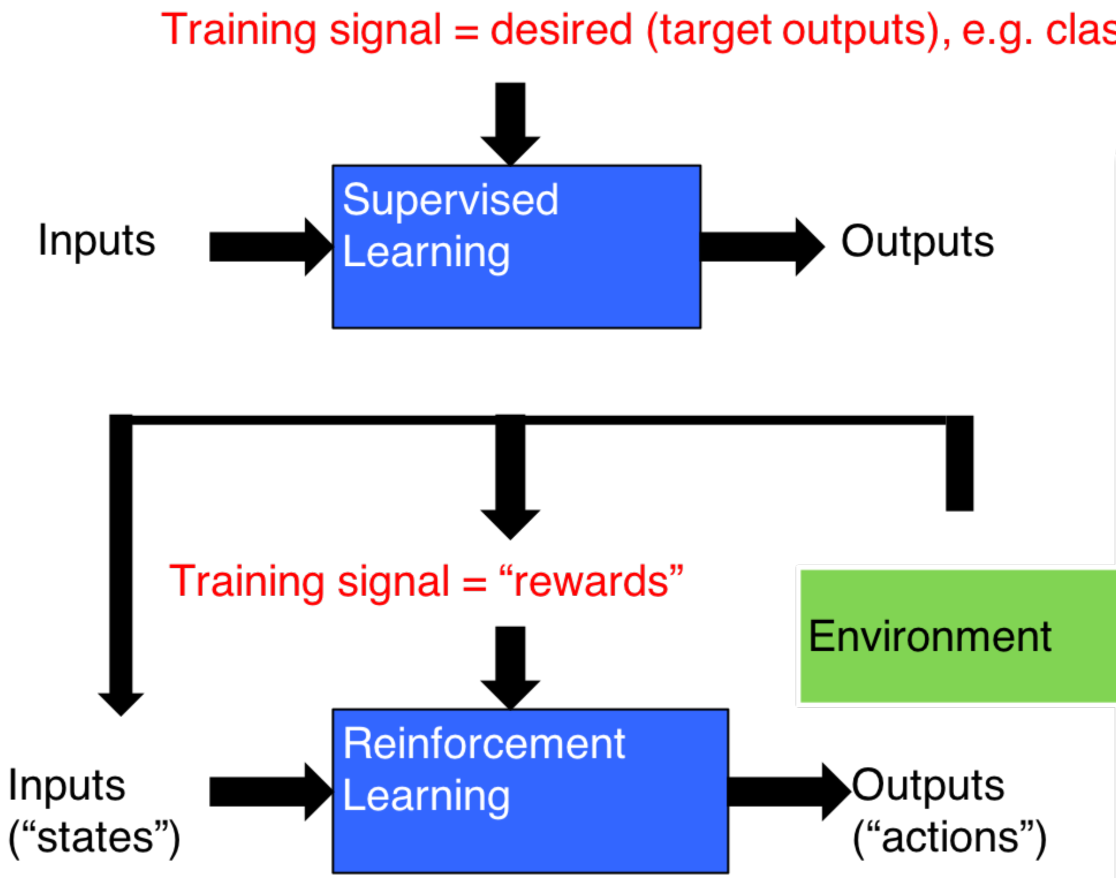
- Learning by trial-and-error, in real time
- Improve with experience
- Inspired by psychology
  - Agent + Environment
  - Agent selects actions to maximize utility function



RL vs supervised learning



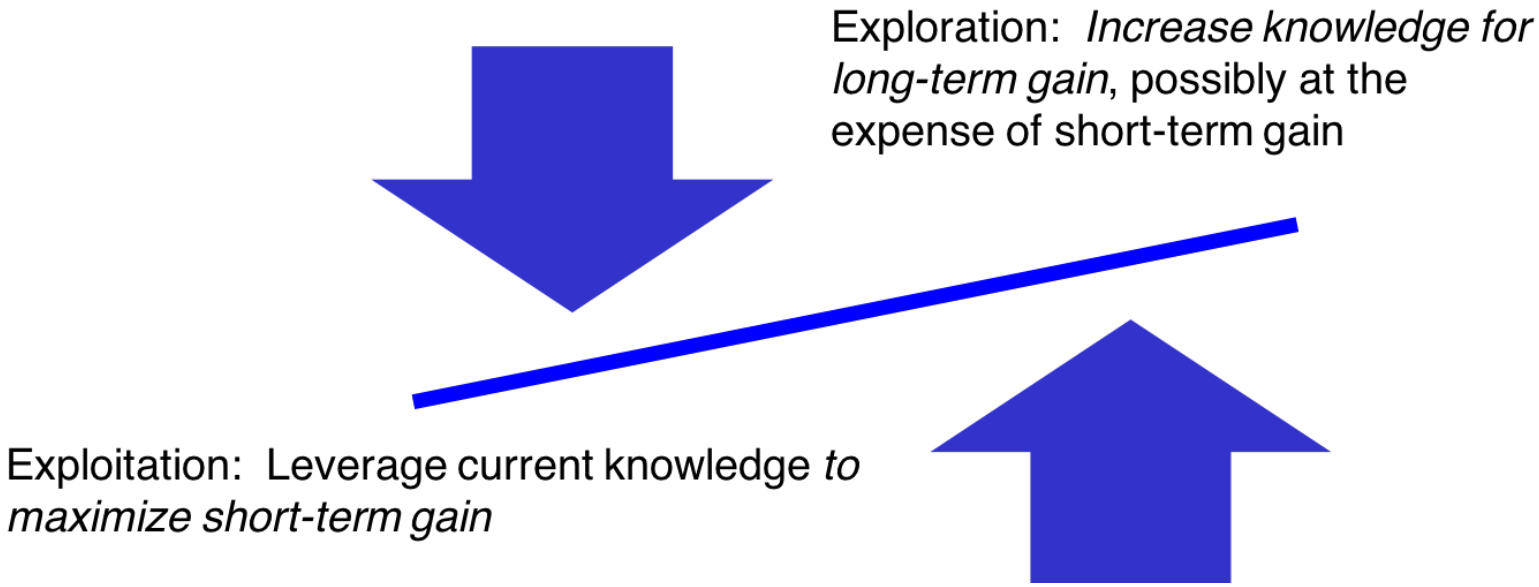
RL vs supervised learning



Practical and technical challenges:

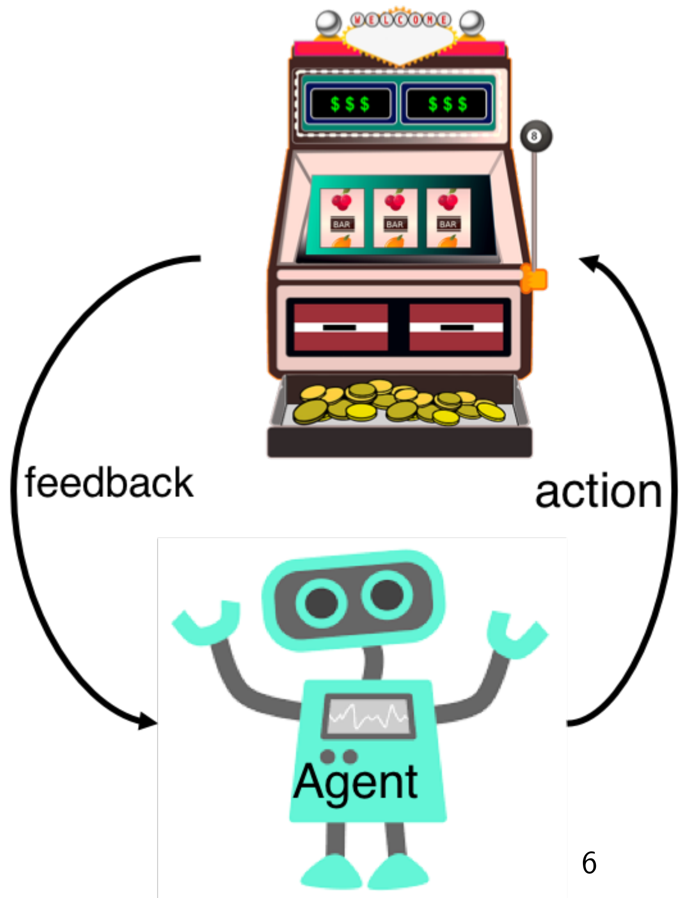
1. Need access to the environment
2. Jointly learning AND planning from **correlated** samples
3. Data distribution changes with actions choice

Exploration/Exploitation



Multi-armed bandit

- Named after the original name of slot machines
- Simplified setting to focus on exploration/exploitation tradeoff
- Remove the notion of “states”
- Focus on the actions/rewards dynamic



Bandit setting (Robbins 1952)

- Set $\mathcal{K} = \{1, 2, \ldots, K\}$ of K actions (arms, machines)
- You are facing a tuple of distributions $\nu = (P_1, P_2, \ldots, P_K)$

P_1 $\quad P_2 \quad \ldots \quad P_K$

- Parameters of distributions are unknown ahead of time
- The best action must be determined by interacting with the environment

Playing a bandit

- You can choose repeatedly among the K arms; each choice is called a play
- After each play of machine k_t, the machine gives a reward $r_t \sim P_{k_t}$
- The value of action k is its expected reward: $\mathbb{E}[r_t]$ when $k_t = k$

Objective: Choose actions in a way that maximizes the reward obtained in the long run (e.g. over 1000 rounds)

Application #1: Internet advertising

- A large Internet company is interested in selling advertising on their website
- It receives money when a company places an ad on the website and that ad gets clicked by a visitor to the website
- What are the bandit's arms?

Application #1: Internet advertising

- A large Internet company is interested in selling advertising on their website
- It receives money when a company places an ad on the website and that ad gets clicked by a visitor to the website
- On a webpage, you can choose to display any of K possible ads
  - Each ad is as an action, with an unknown probability of click rate
  - If the add is clicked, there is a reward, otherwise none

Q: What is the best advertisement strategy to maximize return?
Note that this does not require knowledge of the user, the ad content, the webpage content, etc.

Application #2: Network server selection

- Suppose you can choose to send a job from a user to be processed on one of several servers
- The servers have different processing speed (e.g. due to geographic location, load, etc.
- What are the bandit's arms?

Application #2: Network server selection

- Suppose you can choose to send a job from a user to be processed on one of several servers
- The servers have different processing speed (e.g. due to geographic location, load, etc.
- Each server can be viewed as an action (arm)
- Over time, you want to learn what is the best action to select
- This is used in routing, DNS server selection, cloud computing, etc.

Making decisions

- Policy:

$$\pi = (\pi_1, \pi_2, \dots)$$

- Probability of playing each action at decision time t:

$$\begin{aligned}&\pi_t(k_1, r_1, k_2, r_2, \dots, k_{t-1}, r_{t-1}) \\ &\quad = \Pr[k_t = k | k_1, r_1, k_2, r_2, \dots, k_{t-1}, r_{t-1}] \quad \text{for each } k \in \mathcal{K}\end{aligned}$$

Maximize payoff/minimizing regret

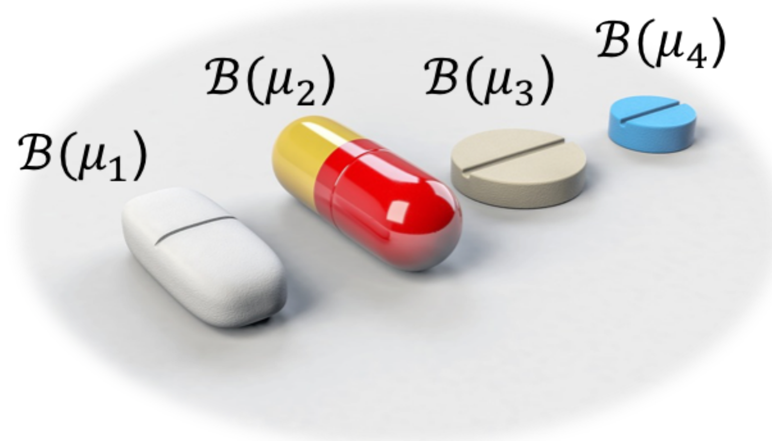
- Maximize cumulative (expected) rewards: $\mathbb{E}[r_1 + r_2 + \cdots + r_T]$
- Define $\mu_k(\nu)$ as the expectation of P_k in configuration ν
- Optimal action: $k_\star(\nu) = \arg\max_{k \in \mathcal{K}} \mu_k(\nu)$
- Optimal expected reward: $\mu_\star(\nu) = \max_{k \in \mathcal{K}} \mu_k(\nu)$
- Minimize regret:

$$\begin{aligned} R_T(\pi, \nu) &= T\mu_\star(\nu) - \mathbb{E}\left[\sum_{t=1}^{T} r_t\right] \\ &= T\mu_\star(\nu) - \sum_{t=1}^{T} \mu_{k_t}(\nu) \end{aligned}$$

$\rightarrow$ Comparison of the algorithm with a gold standard

Example of motivation: Clinical trials (Thompson 1933)

- Available treatments $\mathcal{K}$
- For each patient t:
  - Recommend treatment k_t
  - Observe treatment response r_t
- $r_t \in \{0, 1\}$: "not cured" vs "cured"
- P_k is a different Bernoulli distribution for each treatment $k \in \mathcal{K}$
- Goal: Maximize the number of patients cured



The regret

$$R_T(\pi, \nu) = T\mu_\star(\nu) - \sum_{t=1}^{T} \mu_{k_t}(\nu)$$

Lemma 1. *Let ν be a stochastic bandit environment. Then,*

1. $R_T(\pi, \nu) \geq 0$
2. *Choosing $k_t = k_\star$ for all $t = 1, \ldots, T$ satisfies $R_T(\pi, \nu) = 0$*
3. *If $R_T(\pi, \nu) = 0$, then k_t is optimal with probability 1:* $\Pr[k_t = k_\star] = 1$

- Point 3 is only achievable if we know $k_\star$ in ν
- In general, we only know that $\nu \in \mathcal{E}$ for some environment class $\mathcal{E}$
- Relatively weak objective: find a policy π with sublinear regret

$$\forall \nu \in \mathcal{E}, \quad \lim_{T \to \infty} \frac{R_T(\pi, \nu)}{T} = 0$$

Decomposing the regret

- Suboptimality gap: $\Delta_k(\nu) = \mu_\star(\nu) - \mu_k(\nu)$
- Number of plays of action k up to time t: $N_k(t) = \sum_{s=1}^{t} \mathbb{I}\{k_s = k\}$

$\rightarrow$ In general, $N_k(t)$ is random: policy π is based on random observations

Lemma 2. *For any policy π and K-armed stochastic bandit environment ν and horizon $T \in \mathbb{N}$, the regret R_T of policy π in ν satisfies*

$$R_T = \sum_{k \in \mathcal{K}} \Delta_k \, \mathbb{E}[N_k(T)]$$

Proof. Using that $\sum_{k \in \mathcal{K}} \mathbb{I}\{k_t = k\} = 1$ we have

$$
\begin{aligned}
R_T = T\mu_\star - \mathbb{E}\left[\sum_{t=1}^{T} r_t\right] &= \sum_{t=1}^{T} \mathbb{E}\left[\mu_\star - r_t | k_t\right] \\
&= \sum_{t=1}^{T} \sum_{k \in \mathcal{K}} \mathbb{E}\left[(\mu_\star - r_t)\mathbb{I}\{k_t = k\} | k_t\right] \\
&= \sum_{t=1}^{T} \sum_{k \in \mathcal{K}} (\mu_\star - \mu_{k_t})\, \mathbb{E}\left[\mathbb{I}\{k_t = k\} | k_t\right] \\
&= \sum_{t=1}^{T} \sum_{k \in \mathcal{K}} (\mu_\star - \mu_k)\, \mathbb{E}\left[\mathbb{I}\{k_t = k\} | k_t\right] \\
&= \sum_{k \in \mathcal{K}} \Delta_k \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{I}\{k_t = k\} | k_t\right]
\end{aligned}
$$

□

Estimating action values

- Then we can estimate the value of the action as the sample average of the rewards obtained:

$$\hat{\mu}_k(t) = \frac{\sum_{s=1}^{t} r_s \mathbb{I}\{k_s = k\}}{N_k(t)}$$

- By law of large numbers: $\lim_{N_k(t)\to\infty} \hat{\mu}_k(t) = \mu_k$

Exploration/Exploitation tradeoff

- On the one hand, you need to explore actions, to figure out which one is best (which means some amount of random choice)
- On the other hand, you want to exploit the knowledge you already have, which means picking the greedy action:

$$k_t = \arg\max_{k \in \mathcal{K}} \hat{\mu}_k(t-1)$$

- Exploration only: Choose each action T/K times

$\rightarrow$ Optimal action is played as much as worst action

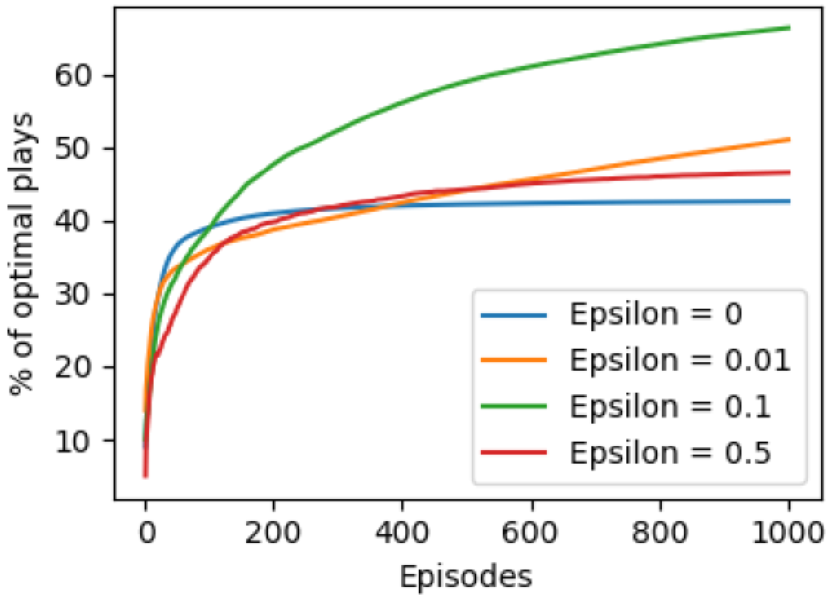
- Always choose action $k_t = \arg\max_{k \in \mathcal{K}} \hat{\mu}_k(t-1)$

$\rightarrow$ Risk of converging to a suboptimal action

So how and when should we explore?

ε-greedy: A simple strategy

- Pick exploration constant $\varepsilon \in [0,1]$, usually small (e.g. $\varepsilon = 0.1$)
- Explore with probability ε: select action k_t uniformely at random
- Exploit with probability ε: selection action $k_t = \arg\max_{k \in \mathcal{K}} \hat{\mu}_k(t-1)$
- Worst action is played (on expectation) $T\varepsilon/K$ times after T rounds



Weakness: Linear regret!

ε-greedy: Sublinear regret

Theorem 1 (Auer at al. 2002). *Solution: Decrease ε with time*

- *Consider reward distributions with bounded support in* $[0, 1]$
- *Let* $\Delta = \min_{k\in\mathcal{K}, k\neq k_\star} \Delta_k$

For $\varepsilon_t = \min\left\{\frac{6K}{\Delta^2 T}, 1\right\}$, *there exists a constant* $C > 0$ *such that the probability of playing a suboptimal action is bounded by* $\frac{C}{\Delta^2 t}$. *As a consequence, for any suboptimal action* k, *it holds that*

$$\mathbb{E}[N_k(T)] \leq \sum_{t=1}^{T} \frac{C}{\Delta^2 t} \leq \frac{C}{\Delta^2} \ln T$$

and thus

$$R_T \leq \sum_{k\in\mathcal{K}} \Delta_k \frac{C}{\Delta^2} \ln T$$

ε-greedy: Weaknesses

- Theoretical guarantee requires the knowledge of Δ

$\rightarrow$ In practice: Decrease ε with time (e.g. $1/t$, $1/\sqrt{t}$, . . .)
What is the good rate?

- Exploration makes suboptimal choices: It explores all actions equally

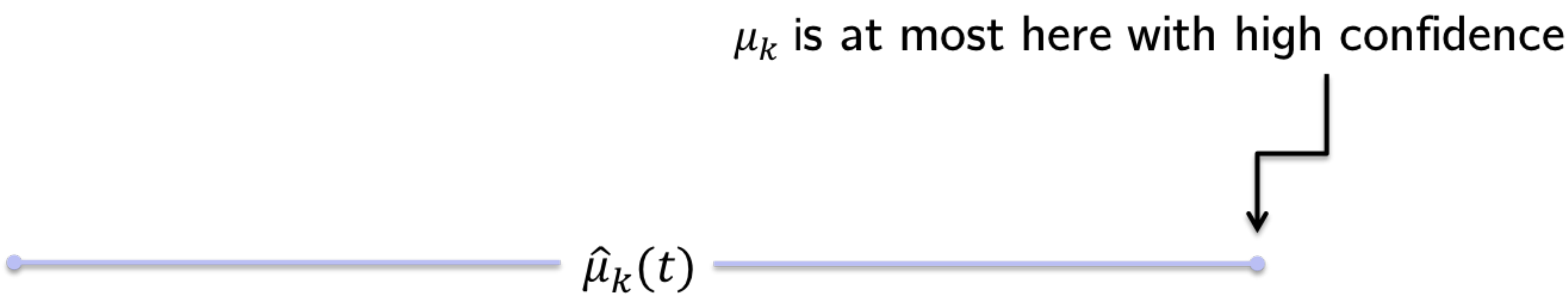
$\rightarrow$ Problem especially important for $K > 2$

Idea: Explore based on information needed

Confidence intervals

E.g. Chernoff-Hœffding
For $r_t \in [0, 1]$:

$$\Pr\left[\mu_k - \hat{\mu}_k(t) \geq \varepsilon\right] \leq \exp\left(-\frac{N_k(t)\varepsilon^2}{2}\right) \quad \text{or} \quad \Pr\left[\mu_k \geq \hat{\mu}_k(t) + \sqrt{\frac{2\ln(1/\delta)}{N_k(t)}}\right] \leq \delta$$



$\rightarrow$ Confidence interval reduces as we gain information about the action

Upper Confidence Bounds (UCB)

$$\mathrm{UCB}_k(t,\delta) = \hat{\mu}_k(t) + \sqrt{\frac{2\ln(1/\delta)}{N_k(t)}}$$

- Input: Number of actions K, confidence level δ
- Play each action once
- For episodes $t > K$: Select action $k_t = \arg\max_{k \in \mathcal{K}} \mathrm{UCB}_k(t-1,\delta)$

Key observation: After the initial plays of each action, action k can only be selected at time $t+1$ if $\mathrm{UCB}_k(t,\delta) \geq \mathrm{UCB}_\star(t,\delta)$. This requires at least one of the following:

a) $\mathrm{UCB}_k(t,\delta) \geq \mu_\star$

b) $\mathrm{UCB}_\star(t,\delta) \leq \mu_\star$

UCB: Bounding suboptimal plays

$$
\begin{aligned}
\mathbb{E}[N_k(T)] &\leq 1 + \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{I}\{\mathrm{UCB}_k(t-1) \geq \mu_\star\}\right] + \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{I}\{\mathrm{UCB}_\star(t-1) \leq \mu_\star\}\right] \\
&\leq 1 + \sum_{t=1}^{T} \Pr\left[\mathrm{UCB}_k(t-1) \geq \mu_\star\right] + \sum_{t=1}^{T} \Pr\left[\mathrm{UCB}_\star(t-1) \leq \mu_\star\right]
\end{aligned}
$$

UCB: $\mathrm{UCB}_k(t,\delta) \geq \mu_\star$

$$\hat{\mu}_k(t) + \sqrt{\frac{2\ln\frac{1}{\delta}}{N_k(t)}} \geq \mu_\star$$

$$\hat{\mu}_k(t) + \sqrt{\frac{2\ln\frac{1}{\delta}}{N_k(t)}} - \mu_k \geq \underbrace{\mu_\star - \mu_k}_{\Delta_k}$$

$$\hat{\mu}_k(t) - \mu_k \geq \underbrace{\Delta_k - \sqrt{\frac{2\ln\frac{1}{\delta}}{N_k(t)}}}_{c\Delta_k} \Rightarrow N_k(t) = \frac{2\ln\frac{1}{\delta}}{\Delta_k^2(1-c)^2}$$

$$\Pr\left[\hat{\mu}_k(t) - \mu_k \geq \Delta_k - \sqrt{\frac{2\ln\frac{1}{\delta}}{N_k(t)}}\right] \leq \exp\left(-\frac{N_k(t)c^2\Delta_k^2}{2}\right) = \delta^{c^2/(1-c)^2}$$

for $N_k(t) \geq \dfrac{2\ln\frac{1}{\delta}}{\Delta_k^2(1-c)^2}$

UCB: $\mathrm{UCB}_\star(t,\delta) \le \mu_\star$

- $\Pr\left[\hat{\mu}_\star(t) + \sqrt{\frac{2\ln\frac{1}{\delta}}{N_\star(t)}} \le \mu_\star\right]$ depends on $N_\star(t)$ (explicitely and in $\hat{\mu}_\star(t)$)
- Explicit $\hat{\mu}_{\star,u}(t) = \hat{\mu}_\star(t)$ s.t. $N_\star(t) = u$
- $N_\star(t)$ is a random variable that could take any value $s \in \{1,\ldots,t\}$

Look at

$$\Pr\left[\min_{s=1\ldots t} \hat{\mu}_{\star,s}(t) + \sqrt{\frac{2\ln\frac{1}{\delta}}{s}} \le \mu_\star\right] \le \Pr\left[\bigcup_{s=1\ldots t} \hat{\mu}_{\star,s}(t) + \sqrt{\frac{2\ln\frac{1}{\delta}}{s}} \le \mu_\star\right]$$

$$\le \sum_{s=1}^{t} \Pr\left[\hat{\mu}_{\star,s}(t) + \sqrt{\frac{2\ln\frac{1}{\delta}}{s}} \le \mu_\star\right]$$

$$\le t\delta$$

UCB: Bounding suboptimal plays (cont'd)

$$\begin{aligned}
\mathbb{E}[N_k(T)] &\le 1 + \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{I}\{\mathrm{UCB}_k(t-1) \ge \mu_\star\}\right] + \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{I}\{\mathrm{UCB}_\star(t-1) \le \mu_\star\}\right] \\
&\le 1 + \sum_{t=1}^{T} \Pr\left[\mathrm{UCB}_k(t-1) \ge \mu_\star\right] + \sum_{t=1}^{T} \Pr\left[\mathrm{UCB}_\star(t-1) \le \mu_\star\right] \\
&\le 1 + \left\lceil \frac{2\ln\frac{1}{\delta}}{\Delta_k^2(1-c)^2} \right\rceil + \sum_{t=1}^{T} \delta^{c^2/(1-c)^2} + \sum_{t=1}^{T} t\delta
\end{aligned}$$

$\Rightarrow$ $\sum_{t=1}^{T} t\delta \le 1$ for $\delta = 1/T^2$

$\Rightarrow$ $\sum_{t=1}^{T} \delta^{c^2/(1-c)^2} = \sum_{t=1}^{T} T^{-2c^2/(1-c)^2} = T^{1-2c^2/(1-c)^2}$

UCB: Bounding suboptimal plays (cont'd)

$$\mathbb{E}[N_k(T)] \leq 1 + \left\lceil \frac{2 \ln T^2}{\Delta_k^2 (1-c)^2} \right\rceil + T^{1-2c^2/(1-c)^2} + 1$$

$\Rightarrow$ Polynomial dependence on T unless $2c^2/(1-c)^2 \geq 1$
$\Rightarrow$ First term blows up if $c \to 1$
$\Rightarrow$ Arbitrarily pick $c = 1/2$

Theorem 2. *Consider UCB on a stochastic K-armed 1-subgaussian bandit problem. For any horizon T, if $\delta = 1/T^2$ then*

$$\mathbb{E}[N_k(T)] \leq \frac{16 \ln T}{\Delta_k^2} + 3$$

$$R_T \leq \sum_{k \in \mathcal{K}, k \neq k_\star} \frac{16 \ln T}{\Delta_k} + 3 \sum_{k \in \mathcal{K}} \Delta_k$$

Thompson Sampling: A Bayesian intuition

- Select next action based on its probability of being optimal
- Observations obtained with action k: $X_{k,1}, X_{k,2}, \ldots, X_{k,N_k(t)}$

$$\overbrace{\Pr\left[\mu_k | X_{k,1}, X_{k,2}, \ldots, X_{k,N_k(t)}\right]}^{\text{posterior}} = \frac{\overbrace{\Pr\left[X_{k,1}, X_{k,2}, \ldots, X_{k,N_k(t)|\mu_k}\right]}^{\text{likelihood of observations}} \overbrace{\Pr[\mu_k]}^{\text{prior}}}{\int \Pr\left[X_{k,1}, X_{k,2}, \ldots, X_{k,N_k(t)} | \mu'\right] \Pr[\mu'] d\mu'}$$

Conjugate priors

- When prior and posterior have the same form
- Provides a closed-form expression for the posterior
- All members from the exponential family have conjugate priors:
  - Gaussian: Gaussian
  - Bernoulli: Beta
  - Poisson: Gamma
  - Multinomial: Dirichlet

Example: Beta-Bernoulli

- Bernoulli e.g. head/tail
- Parameter p: probability of success (1)
  - $(1-p)$: probability of failure
  - $\mu = p$
- Posterior on μ after N observations:

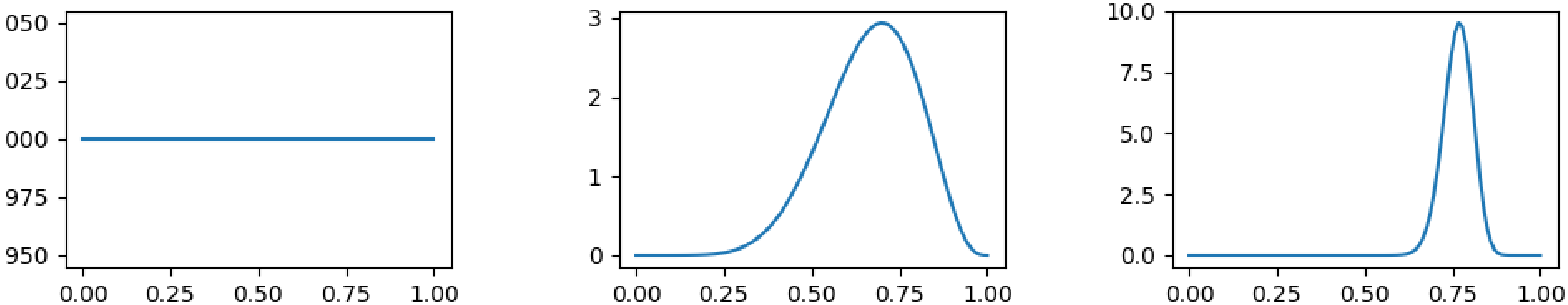
$$\mathrm{Beta}\left(\underbrace{\alpha_0 + \sum_{n=1}^{N} X_n}_{\text{number of successes}}, \underbrace{\beta_0 + N - \sum_{n=1}^{N} X_n}_{\text{number of failures}} \right)$$

- α_0 and β_0 are priors

Example: Beta-Bernoulli posterior evolution

- Typical priors: $\alpha_0 = \beta_0 = 1$: Uniform priors
- Example: $\mu = 0.1$

$N = 0$ $\qquad$ $N = 10$ $\qquad$ $N = 100$

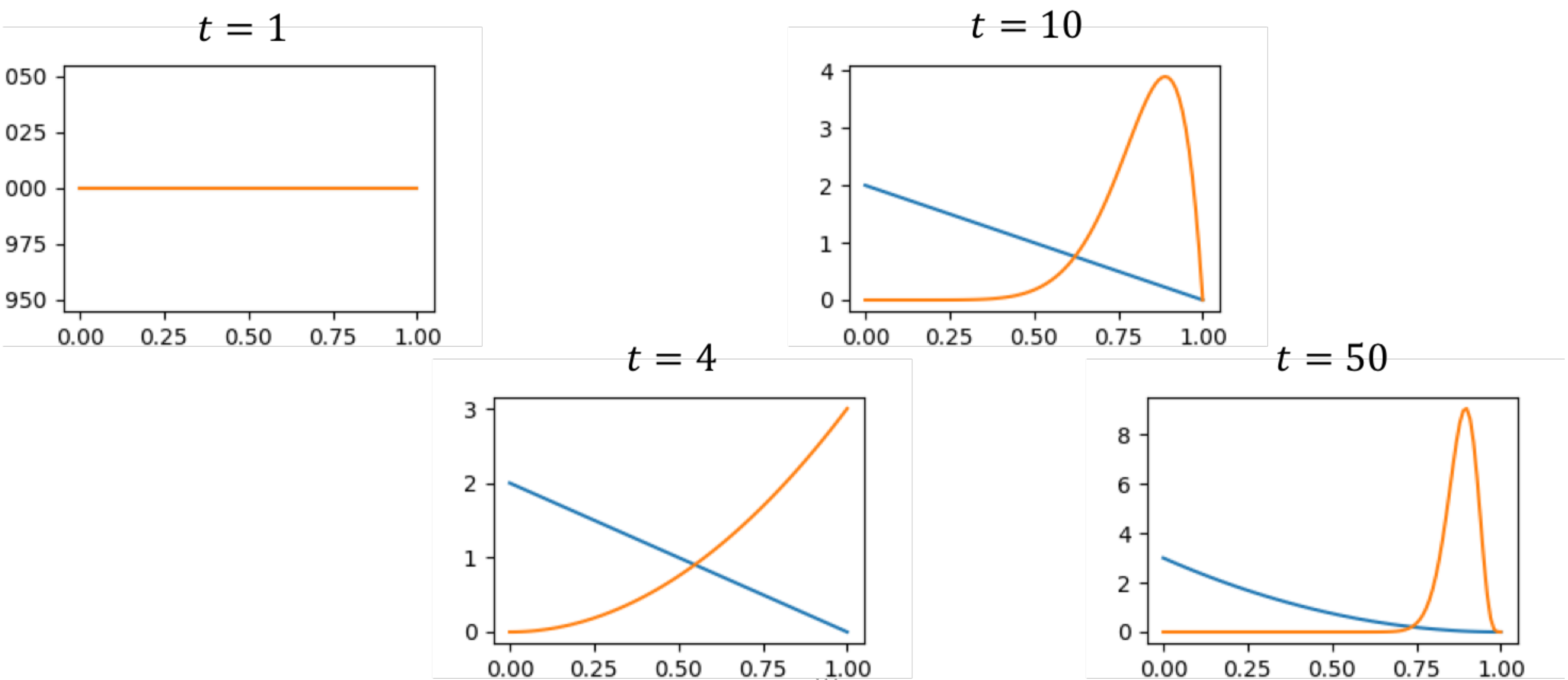


Thompson Sampling

- Maintain one posterior $\pi_t^{(k)}$ for each action $k \in \mathcal{K}$
- At time t:
  - Sample one value $\theta_k \sim \pi_t^{(k)}$ for each action $k \in \mathcal{K}$
  - Select action $k_t = \arg\max_{k \in \mathcal{K}} \theta_k$
- Stochastic policy
  - Good for parallel runs
  - Good for delayed feedback
- Exploration reduces as posterior tightens

Example: Thompson Sampling on 2-arms

- Bernoulli rewards (head/tails, win/loss) with $\mu_1 = 0.9 \qquad \mu_2 = 0.1$



Thompson Sampling analysis

- Frequentist analysis is much more technical than UCB
- Frequentist analysis depends on prior

Theorem 3. *If Thompson Sampling is run on a Gaussian bandit $\nu \in \mathcal{E}_{\mathcal{N}}^{K}$, then*

$$R_T \leq C \sum_{k \in \mathcal{K}} \Delta_k + \sum_{k \in \mathcal{K}, k \neq k_\star} C \frac{\ln T}{\Delta_k}$$

for some universal constant $C > 0$.

- In practice, TS has better regret than *standard* UCB
- There exists variants of UCB that have better regret and less variance (in practice) than TS and

Summary

- Many algorithms, many *similar* regret bounds (in terms of order)
- Similar regret bounds do not necessarily mean similar performance in practice
- UCB depends on the tightness of confidence intervals
- TS depends on how fast the posterior converges
- Stochasticity in the policy can be good or bad
- Both UCB and TS have been extended to different bandits variants

Synthetic experiments

```
import numpy as np

means = np.array([0.1, 0.9])
regrets = np.max(means) - means
K = len(means)

alg = TS_Bernoulli(nb_arms=K, a0=1, b0=1)

cumul_regrets = [0]
for t in range(1000):
    k_t = alg.select()
    r_t = np.random.rand() < means[k_t]
    alg.update(k_t, r_t)
    cumul_regrets.append(cumul_regrets[-1]+regrets[k_t])
...
```