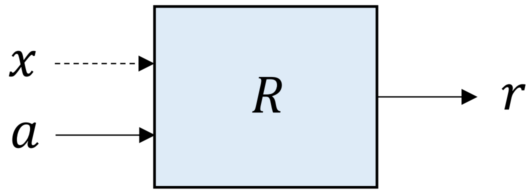
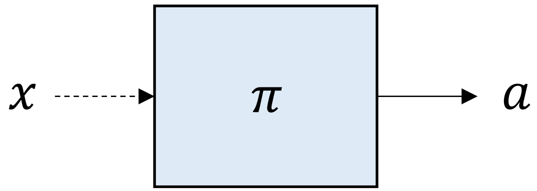
Review: Bandit Techniques

x: context, a: action, r: reward	MAB	CB
Value-based  (context, action) to reward	Mean estimation + EG, BE, IGW	Regression + EG, BE, IGW
Policy-based  context to action distribution	KL-regularized update with reward estimators (EXP3) + baseline, bias, or uniform exploration	PPO/NPG PG + baseline, bias, uniform exploration, clipping

Are we done with bandits?

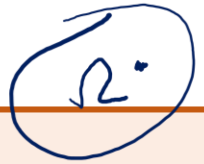
- Almost, but we have a last important topic: how to deal with continuous action sets? (#actions could be infinite)
- We will go over the 4 regimes once again to deal with continuous actions

	MAB	CB
VB		
PB		

Dealing with Continuous Action Set

Continuous Action Set

Full-information feedback



Given: Action set $\Omega \subseteq \mathbb{R}^d$

For time $t = 1, 2, \dots, T$:

Learner chooses a point $a_t \in \Omega$

Environment reveals a reward function $r_t\colon\ \Omega \to \mathbb{R}$

Bandit feedback

Given: Action set $\Omega \subseteq \mathbb{R}^d$

For time $t = 1, 2, \dots, T$:

Learner chooses a point $a_t \in \Omega$

Environment reveals a reward value $r_t(a_t)$

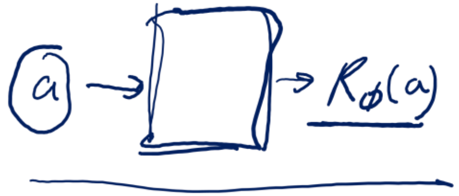
Continuous Multi-Armed Bandits

With a mean estimator

	MAB	CB
VB	•	
PB		

Value-Based Approach (mean estimation)

- Use supervised learning to learn a reward function $R_\phi(a)$



- How to perform the exploration strategies (like ϵ-Greedy)?
  - How to find $\text{argmax}_a R_\phi(a)$?
  - Usually, there needs to be another **policy learning procedure** that helps to find $\text{argmax}_a R_\phi(a)$
  - Then we can explore as $a_t = \text{argmax}_a R_\phi(a) + \sigma \mathcal{N}(0, I)$

Full-Information Policy learning Procedure

Gradient Ascent

For $t = 1, 2, \dots, T$:

Choose action a_t

Receive reward function r_t: $\Omega \rightarrow \mathbb{R}$

Update action $a_{t+1} \leftarrow \mathcal{P}_\Omega(a_t + \eta \nabla r_t(a_t))$

When $\pi_\theta = \mathcal{N}(\mu_\theta, \sigma^2 I)$, the KL-regularized policy update

$$\theta_{t+1} = \underset{\theta}{\operatorname{argmax}} \left\{ \int \left(\pi_\theta(a) - \pi_{\theta_t}(a)\right) r_t(a)\, \mathrm{d}a - \frac{1}{\eta} \mathrm{KL}\left(\pi_\theta, \pi_{\theta_t}\right) \right\}$$

is close to $\mu_{\theta_{t+1}} \leftarrow \mu_{\theta_t} + \eta\sigma \nabla r_t(\mu_{\theta_t})$

Regret Bound of Gradient Ascent

Theorem. If Ω is convex and all reward functions r_t are concave, then Gradient Ascent ensures

$$\text{Regret} = \max_{a^\star \in \Omega} \sum_{t=1}^{T} r_t(a^\star) - r_t(a_t) \leq \frac{\max_{a \in \Omega} \|a\|_2^2}{\eta} + \eta \sum_{t=1}^{T} \|\nabla r_t\|_2^2$$

This can also be applied to the finite-action setting, but only ensures a $\sqrt{AT}$ regret bound (using exponential weights we get $\sqrt{(\log A)T}$)

Combining with Mean Estimator

The mean estimator R_ϕ essentially gives us a full-information reward function

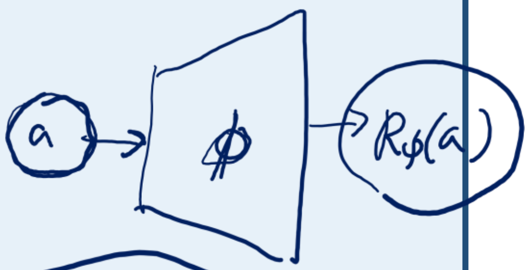
For $t = 1,2,\ldots,T$:

Take action $\tilde{a}_t = \mathcal{P}_\Omega(a_t + \sigma\,\mathcal{N}(0,I))$

Receive $r_t(\tilde{a}_t)$

Update the mean estimator:

$$\phi \leftarrow \phi - \lambda \nabla_\phi \left[\left(\underline{R_\phi(\tilde{a}_t)} - \underline{r_t(\tilde{a}_t)}\right)^2\right]$$



Update policy:

$$a_{t+1} = \mathcal{P}_\Omega\left(a_t + \eta \nabla_a R_\phi(a_t)\right) \longleftarrow \phi \quad \arg\max_a R_\phi(a)$$

Think of this as a continuous-action counterpart of ϵ-Greedy

Continuous Contextual Bandits

With a regression oracle

	MAB	CB
VB		•
PB		

Combining with Regression Oracle (a bandit version of DDPG)

Assume policy parametrization
$\pi_\theta(\cdot\,|x) = \mathcal{N}(\mu_\theta(x), \sigma^2 I)$

For $t = 1,2,\ldots,T$:

Receive context x_t

Take action $a_t = \mathcal{P}_\Omega(\mu_\theta(x_t) + \sigma\,\mathcal{N}(0, I))$

Receive $r_t(x_t, a_t)$

Update the mean estimator:

$$\phi \leftarrow \phi - \lambda \nabla_\phi \left[\left(R_\phi(x_t, a_t) - r_t(x_t, a_t) \right)^2 \right]$$

Update policy:

$$\theta \leftarrow \theta + \eta \nabla_\theta R_\phi(\mu_\theta(x_t))$$

Continuous Multi-Armed Bandits

Pure policy-based algorithms

	MAB	CB
VB		
PB	•	

Pure Policy-Based Approach

Gradient Ascent

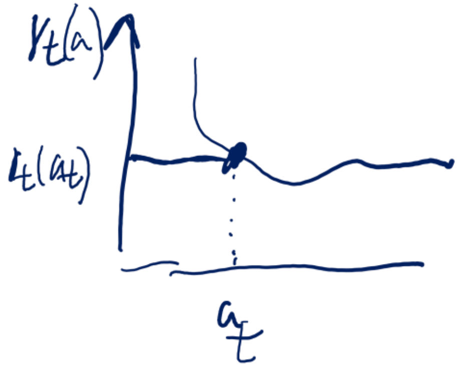
For $t = 1, 2, \dots, T$:

Choose action a_t

Receive reward function $r_t\colon\ \Omega \to \mathbb{R}$

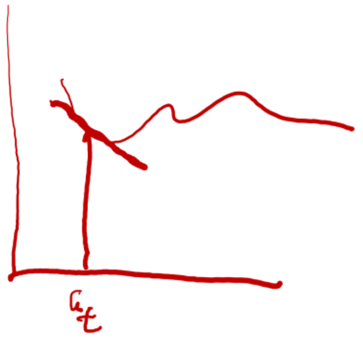
Update action $a_{t+1} \leftarrow \mathcal{P}_\Omega(a_t + \eta \boxed{\nabla r_t(a_t)})$

We face a similar problem as in EXP3: if we only observe $r_t(a_t)$, how can we estimate the **gradient**?

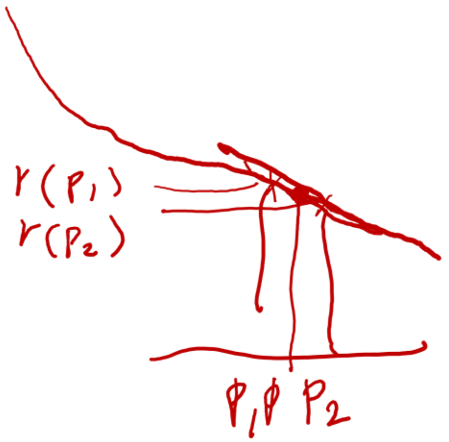


(Nearly) Unbiased Gradient Estimator

Goal: construct $g_t \in \mathbb{R}^d$ such that $\mathbb{E}[g_t] \approx \nabla r_t(a_t)$ with only $r_t(a_t)$ feedback



$\underline{p_1, p_2} \longrightarrow r(p_1), r(p_2)$



$$\boxed{\nabla r(p) \approx \frac{r(p_2) - r(p_1)}{p_2 - p_1}}$$

(Nearly) Unbiased Gradient Estimator (1/3)

Uniformly randomly choose a direction $i_t \in \{1, 2, \ldots, d\}$

Uniformly randomly choose $\beta_t \in \{1, -1\}$

Sample $\tilde{a}_t = a_t + \delta \beta_t \mathrm{e}_{i_t}$

Observe $r_t(\tilde{a}_t)$

Define $g_t = \frac{d r_t(\tilde{a}_t)}{\delta} \beta_t \mathrm{e}_{i_t}$

(Nearly) Unbiased Gradient Estimator (2/3)

Uniformly randomly choose s_t from the unit sphere $\mathbb{S}_d = \{s \in \mathbb{R}^d\colon \ \|s\|_2 = 1\}$

Sample $\tilde{a}_t = a_t + \delta s_t$

Observe $r_t(\tilde{a}_t)$

Define $g_t = \frac{d r_t(\tilde{a}_t)}{\delta} s_t$

(Nearly) Unbiased Gradient Estimator (3/3)

Choose $s_t \sim \mathcal{D}$ with $\mathbb{E}_{s\sim\mathcal{D}}[s] = 0$

Sample $\tilde{a}_t = a_t + s_t$

Observe $r_t(\tilde{a}_t)$

Define $g_t = r_t(\tilde{a}_t) H_t^{-1} s_t$ where $H_t \coloneqq \mathbb{E}_{s\sim\mathcal{D}}[s s^\top]$

Gradient Ascent with Gradient Estimator

Assume the feasible set Ω contains a ball of radius δ

Define $\Omega' = \{a \in \Omega\colon\ \mathcal{B}(a, \delta) \subset \Omega\}$

Arbitrarily pick $a_1 \in \Omega'$

For $t = 1,2, \dots, T$:

Let $\tilde{a}_t = a_t + s_t$ where $s_t \sim \mathcal{D}$ (assume that $\|s_t\| \le \delta$ always holds)

Receive $r_t(\tilde{a}_t)$

Define

$$g_t = (r_t(\tilde{a}_t) - b_t) H_t^{-1} s_t \qquad \text{where } H_t \coloneqq \mathbb{E}_{s\sim\mathcal{D}}[ss^\top]$$

Update policy:

$$a_{t+1} = \Pi_{\Omega'}\,(a_t + \eta g_t)$$

Continuous Contextual Bandits

Pure policy-based algorithms

	MAB	CB
VB		
PB		•