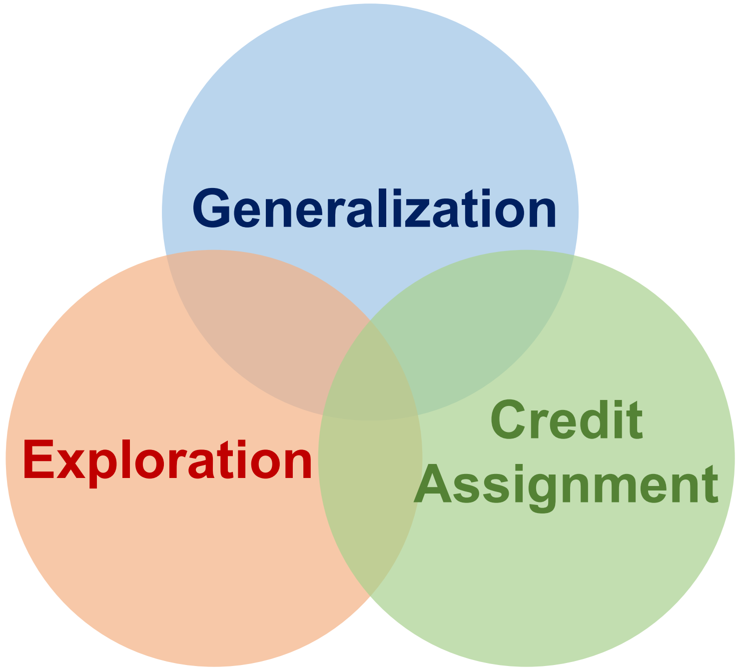
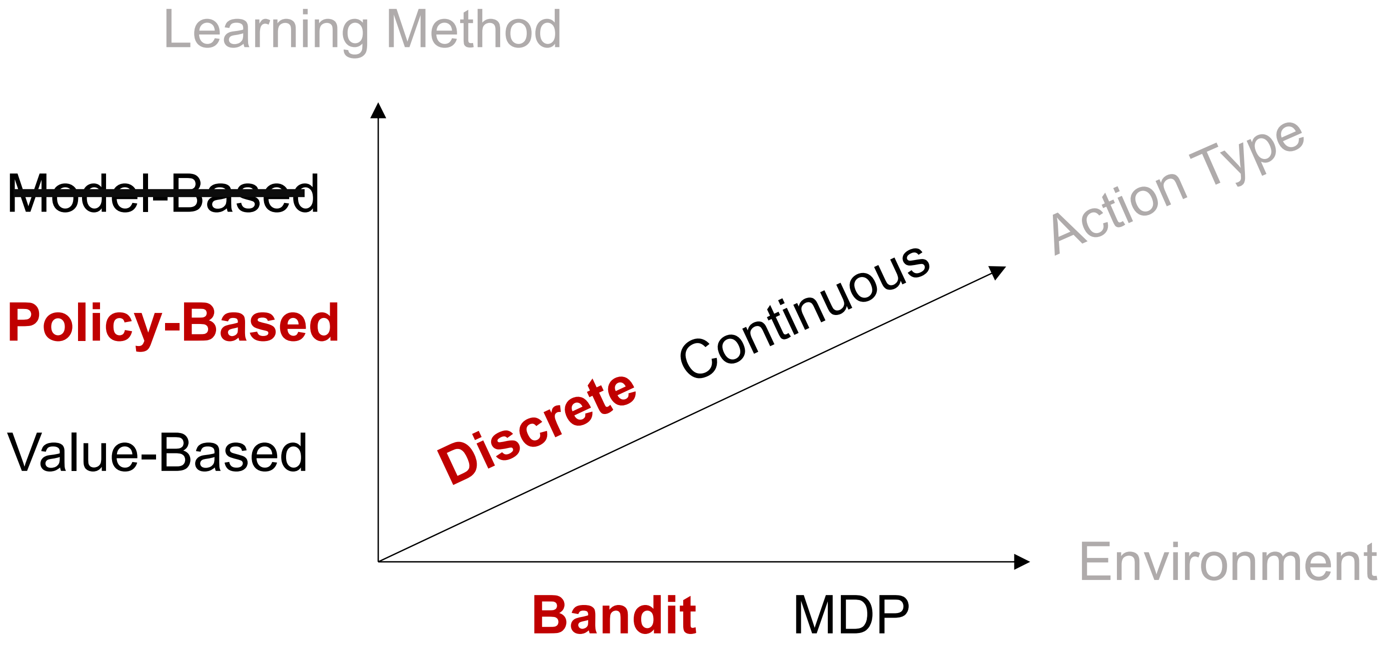
Bandits 2

Chen-Yu Wei

Roadmap


Generalization
Exploration
Credit
Assignment


Learning Method
~~Model-Based~~
Policy-Based
Value-Based
Action Type
Discrete
Continuous
Environment
Bandit
MDP

Policy-Based Bandits

- Key challenges: **Exploration** and **Generalization (if there are contexts)**
- Algorithms we will discuss:
  - KL-regularized policy updates (PPO, NPG)
  - Policy gradient (REINFORCE)
- We will add a little discussion on “time-varying” reward functions to motivate the algorithm design

The Full-Information MAB

Given: set of actions $\mathcal{A} = \{1, \dots, A\}$

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

Environment decides the reward of all actions $r_t(1), r_t(2), \dots, r_t(A)$ without revealing

The learner chooses an action a_t

Environment reveals the reward $r_t(a)$ **of all actions**

$$\text{Regret} = \max_a \sum_{t=1}^{T} r_t(a) - \sum_{t=1}^{T} r_t(a_t)$$

KL-Regularized Policy Updates

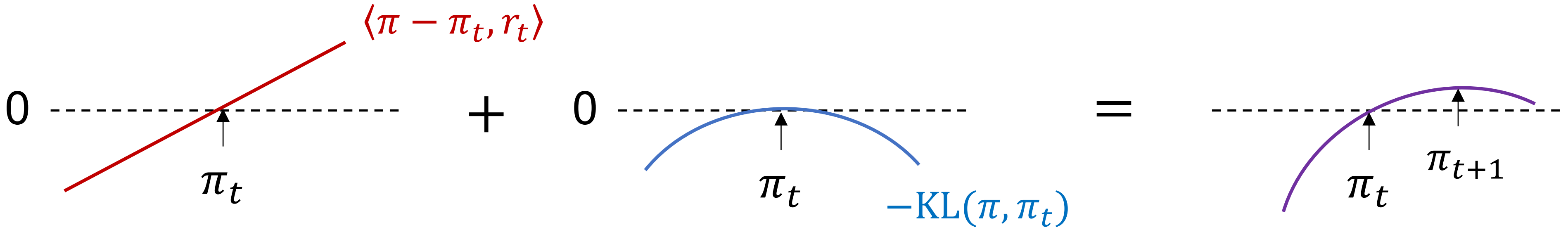
$\pi_t \quad r_t$

$$r_t = \begin{bmatrix} r_t(1) \\ \vdots \\ r_t(A) \end{bmatrix} \qquad \pi = \begin{bmatrix} \pi(1) \\ \vdots \\ \pi(A) \end{bmatrix}$$

$$\boxed{\langle \pi, r_t \rangle}$$

$$\pi_{t+1} = \operatorname*{argmax}_{\pi \in \Delta(\mathcal{A})} \left\{ \boxed{\langle \pi - \pi_t, r_t \rangle} - \boxed{\frac{1}{\eta} \mathrm{KL}(\pi, \pi_t)} \right\}$$

$$= \operatorname*{argmax}_{\pi \in \Delta(\mathcal{A})} \left\{ \underbrace{\sum_a \big(\pi(a) - \pi_t(a)\big) r_t(a)}_{\text{The Improvement of } \pi \text{ over } \pi_t \text{ on } r_t} - \underbrace{\frac{1}{\eta} \boxed{\sum_a \pi(a) \log \frac{\pi(a)}{\pi_t(a)}}}_{\text{Distance between } \pi \text{ and } \pi_t} \right\}$$



Why regularizing the update?

Why KL-Regularized Policy Updates?

1. Maintaining **stability** for adversarial environments

Time	1	2	3	4	5	6	…
$R_t(1)$	0.5	0	1	0	1	0	…
$R_t(2)$	0	1	0	1	0	1	…

Follow the leader: $a_t = \max_{a \in \mathcal{A}} \left\{ \sum_{i=1}^{t-1} r_i(a) \right\}$

2. When combining the algorithm with function approximation, the gradient only approximates the **local** reward landscape.

KL-Regularized Policy Updates

$$\pi_{t+1} = \underset{\pi \in \Delta(\mathcal{A})}{\operatorname{argmax}} \left\{ \langle \pi - \pi_t, r_t \rangle - \frac{1}{\eta} \mathrm{KL}(\pi, \pi_t) \right\}$$

$\Longleftrightarrow$

Exponential weight updates

$$\pi_{t+1}(a) = \frac{\pi_t(a)\, e^{\eta r_t(a)}}{\sum_{b \in \mathcal{A}} \pi_t(b)\, e^{\eta r_t(b)}}$$

Solving the optimization

Regret Bound for Exponential Weight Updates

Theorem. Will be proven in HW2

Assume that $\eta r_t(a) \le 1$ for all t, a. Then EWU

$$\pi_{t+1} = \underset{\pi \in \Delta(\mathcal{A})}{\operatorname{argmax}} \left\{ \langle \pi - \pi_t, r_t \rangle - \frac{1}{\eta} \mathrm{KL}(\pi, \pi_t) \right\}$$

ensures for any $a^\star \in \mathcal{A}$,

$$\sum_{t=1}^{T} \left(r_t(a^\star) - \underbrace{\langle \pi_t, r_t \rangle}_{\mathbb{E}[r_t(a_t)]} \right) \le \frac{\log A}{\eta} + \eta \sum_{t=1}^{T} \overbrace{\sum_{a=1}^{A} \pi_t(a)}^{1} \overbrace{r_t(a)^2}^{\le 1}$$

If $|r_t(a)| \le 1$ and $\eta \le 1 \Rightarrow \quad \sum_{t=1}^{T} \left(r_t(a^\star) - r_t(a_t) \right) \le \frac{\log A}{\eta} + \eta T \quad \approx \sqrt{(\log A) T}$

Questions and Discussions

- Why do we care about regret against a **fixed** action when the reward function is changing?
  - Environments where reward function is manipulated by an adversary
  - For MDPs, the “long-term reward” changes over time (because the learner’s policy in later steps changes over time).
  - A lot of other applications: game theory, constrained optimization, boosting, etc.

Exponential Weight Update ∈ Mirror Ascent

General form of **Mirror Ascent**:

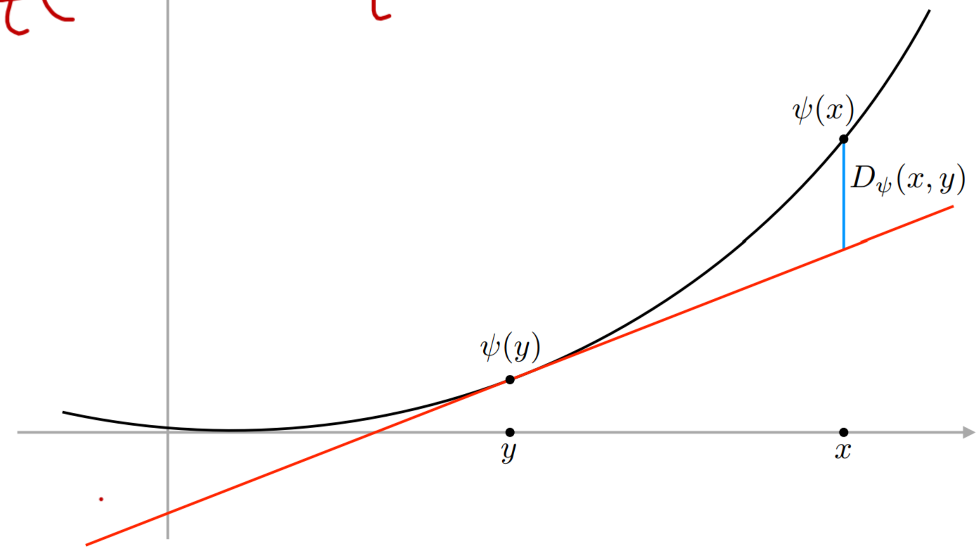
$f_t(\pi) = \langle \pi, r_t \rangle$

$\nabla f_t(\pi) = r_t$

$$x_{t+1} = \underset{x \in \Omega}{\operatorname{argmax}} \left\{ \langle x - x_t, r_t \rangle - \frac{1}{\eta} \underbrace{D_\psi(x, x_t)} \right\}$$

Usually, $r_t = \nabla f_t(x_t)$ for some function f_t that we want to maximize

Bregman divergence with respect to a convex function ψ



$$D_\psi(x, y) = \psi(x) - \psi(y) - \langle \nabla\psi(y), x - y \rangle$$

$\sum_a X(a) \log \frac{1}{X(a)}$

$P_\Omega(x) = \underset{y\in\Omega}{\text{argmin}} \|x-y\|_2$

Exponential Weight Update ∈ Mirror Ascent

Ω y x

Special cases of **Mirror Ascent**: $x_{t+1} = \underset{x\in\Omega}{\text{argmax}} \left\{ \langle x - x_t, r_t \rangle - \frac{1}{\eta} D_\psi(x, x_t) \right\}$

$\psi(x)$	$D_\psi(x,y)$	Update Rule $\nabla f_t(x_t)$	Ω
$\frac{1}{2}\|x\|_2^2$	$\frac{1}{2}\|x-y\|_2^2$	$x_{t+1} = \mathcal{P}_\Omega(x_t + \eta r_t)$ (Projected) Gradient Ascent	Any convex set
$\sum_a x(a) \log x(a)$ Negative entropy	$\sum_a x(a) \log \frac{x(a)}{y(a)}$	$x_{t+1}(a) = \frac{x_t(a) e^{\eta r_t(a)}}{\sum_b x_t(b)\, e^{\eta r_t(b)}}$	Probability space
$\sum_a \log \frac{1}{x(a)}$	$\sum_a \left(\frac{x(a)}{y(a)} - \log \frac{x(a)}{y(a)} - 1\right)$	$\frac{1}{x_{t+1}(a)} = \frac{1}{x_t(a)} - \eta r_t(a) + \gamma_t$	Probability space
$-2\sum_a \sqrt{x(a)}$	$\sum_a \frac{\left(\sqrt{x(a)} - \sqrt{y(a)}\right)^2}{2\sqrt{y(a)}}$	$\frac{1}{\sqrt{x_{t+1}(a)}} = \frac{1}{\sqrt{x_t(a)}} - \eta r_t(a) + \gamma_t$	Probability space
$\frac{1}{2}\|x\|_M^2 = \frac{1}{2}x^\top M x$	$\frac{1}{2}\|x-y\|_M^2$	$x_{t+1} = \mathcal{P}_\Omega(x_t + \eta M^{-1} r_t)$	Any convex set

Multi-Armed Bandits

Adversarial Multi-Armed Bandits

Given: set of arms $\mathcal{A} = \{1, \dots, A\}$

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

Environment decides the reward vector $r_t = (r_t(1), \dots, r_t(A))$ (not revealing)

Learner chooses an arm $a_t \in \mathcal{A}$

Learner observes $r_t(a_t)$

$$\mathrm{Regret} = \max_{a \in \mathcal{A}} \sum_{t=1}^{T} r_t(a) - \sum_{t=1}^{T} r_t(a_t)$$

Recall: Exponential Weight Updates

$$\pi_{t+1} = \underset{\pi \in \Delta(\mathcal{A})}{\operatorname{argmax}} \left\{ \langle \pi - \pi_t, r_t \rangle - \frac{1}{\eta} \mathrm{KL}(\pi, \pi_t) \right\} \Longleftrightarrow \pi_{t+1}(a) = \frac{\pi_t(a)\, e^{\eta r_t(a)}}{\sum_{b \in \mathcal{A}} \pi_t(b)\, e^{\eta r_t(b)}}$$

Exponential Weight Updates for Bandits?

$$\pi_{t+1} = \operatorname*{argmax}_{\pi \in \Delta(\mathcal{A})} \left\{ \langle \pi - \pi_t, r_t \rangle - \frac{1}{\eta} \mathrm{KL}(\pi, \pi_t) \right\} \Longleftrightarrow \pi_{t+1}(a) = \frac{\pi_t(a)\, e^{\eta r_t(a)}}{\sum_{b \in \mathcal{A}} \pi_t(b)\, e^{\eta r_t(b)}}$$

No longer observable

Only update the arm that we choose?

Exponential Weight Updates for Bandits?

$$\pi_{t+1} = \underset{\pi \in \Delta(\mathcal{A})}{\operatorname{argmax}} \left\{ \langle \pi - \pi_t, \hat{r}_t \rangle - \frac{1}{\eta} \mathrm{KL}(\pi, \pi_t) \right\} \iff \pi_{t+1}(a) = \frac{\pi_t(a)\, e^{\eta \hat{r}_t(a)}}{\sum_{b \in \mathcal{A}} \pi_t(b)\, e^{\eta \widehat{r}_t(b)}}$$

- $\hat{r}_t(a)$ is an **"estimator"** for $r_t(a)$
- But we can only observe the reward of one arm
- Furthermore, $r_t(a)$ is different in every round (If we do not sample arm a in round t, we'll never be able to estimate $r_t(a)$ in the future)

Unbiased Reward / Gradient Estimator

$\hookrightarrow \mathbb{E}_{a_t}\left[\hat{r}_t(a)\right] = r_t(a) \quad \forall a$

proof: Fix an a. $\quad \mathbb{E}_{a_t}\left[\hat{r}_t(a)\right] = \underbrace{\pi_t(a)}_{Pr(\text{choosing } a)} \cdot \frac{r_t(a)}{\pi_t(a)} + \underbrace{(1-\pi_t(a))}_{Pr(\text{not choosing } a)} \cdot 0 = r_t(a)$

Weight a sample by **the inverse of the probability we observe it**

$$\hat{r}_t(a) = \frac{r_t(a)}{\pi_t(a)} \mathbb{I}\{a_t = a\} = \begin{cases} \dfrac{r_t(a)}{\pi_t(a)} & \text{if } a_t = a \quad ✓ \\ 0 & \text{otherwise} \quad ✓ \end{cases}$$

Inverse Propensity Weighting / Inverse Probability Weighting / Importance Weighting

Directly Applying Exponential Weights

$\pi_1(a) = 1/A$ for all a

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

Sample $a_t \sim \pi_t$, and observe $r_t(a_t)$

Define for all a:

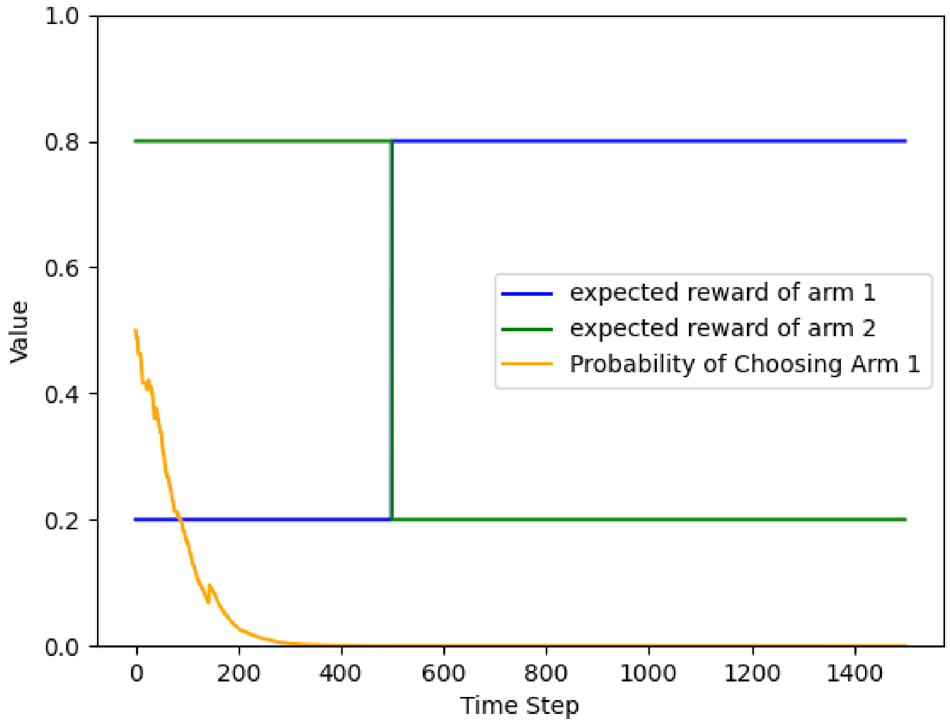
$$\hat{r}_t(a) = \frac{r_t(a)}{\pi_t(a)} \mathbb{I}\{a_t = a\}$$

Update policy:

$$\pi_{t+1}(a) = \frac{\pi_t(a) \exp(\eta \hat{r}_t(a))}{\sum_{a' \in \mathcal{A}} \pi_t(a') \exp(\eta \hat{r}_t(a'))}$$

Simple Experiment

- $A = 2,\ T = 1500,\ \eta = 1/\sqrt{T}$
- For $t \le 500,\ r_t = [\text{Bernoulli}(0.2), \text{Bernoulli}(0.8)]$
- For $500 < t \le 1500,\ r_t = [\text{Bernoulli}(0.8), \text{Bernoulli}(0.2)]$



Recall the Theorem

$$\frac{\ln A}{\eta} + 2AT = \sqrt{AT \ln A}$$

$$= \tilde{O}(\sqrt{AT})$$

Theorem. Does this still hold? ✖

Assume that $\boxed{\eta \hat{r}_t(a) \le 1}$ for all t, a. Then EWU

$$\pi_{t+1}(a) = \frac{\pi_t(a)\exp(\eta \hat{r}_t(a))}{\sum_{a' \in \mathcal{A}} \pi_t(a') \exp(\eta \hat{r}_t(a'))}$$

$\mathbb{E}[\sim] \le AT$

ensures for any $a^\star$,

$$\underbrace{\sum_{t=1}^{T} (\hat{r}_t(a^\star) - \langle \pi_t, \hat{r}_t \rangle)}_{\text{Is this still related to regret?}} \le \frac{\ln A}{\eta} + \eta \underbrace{\sum_{t=1}^{T} \sum_{a=1}^{A} \pi_t(a) \hat{r}_t(a)^2}_{\text{Is this still well-bounded?}}$$

(cf: $\mathbb{E}[\sim] \le T$ in full-info case)

Is this still related to regret? ○

Is this still well-bounded? ○

Solution 1: Adding Extra Exploration

- **Idea:** use at least η probability to choose each arm

- Instead of sampling a_t according to π_t, use

$$\pi'_t(a) = (1 - A\eta)\pi_t(a) + \eta$$

Then the unbiased reward estimator becomes

$$\hat{r}_t(a) = \frac{r_t(a)}{\pi'_t(a)} \mathbb{I}\{a_t = a\} = \frac{r_t(a)}{(1 - A\eta)\pi_t(a) + \eta} \mathbb{I}\{a_t = a\}$$

Applying Solution 1

$\pi_1(a) = 1/A$ for all a

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

Sample a_t from $\pi'_t = (1 - A\eta)\pi_t + A\eta \text{ uniform}(\mathcal{A})$, and observe $r_t(a_t)$

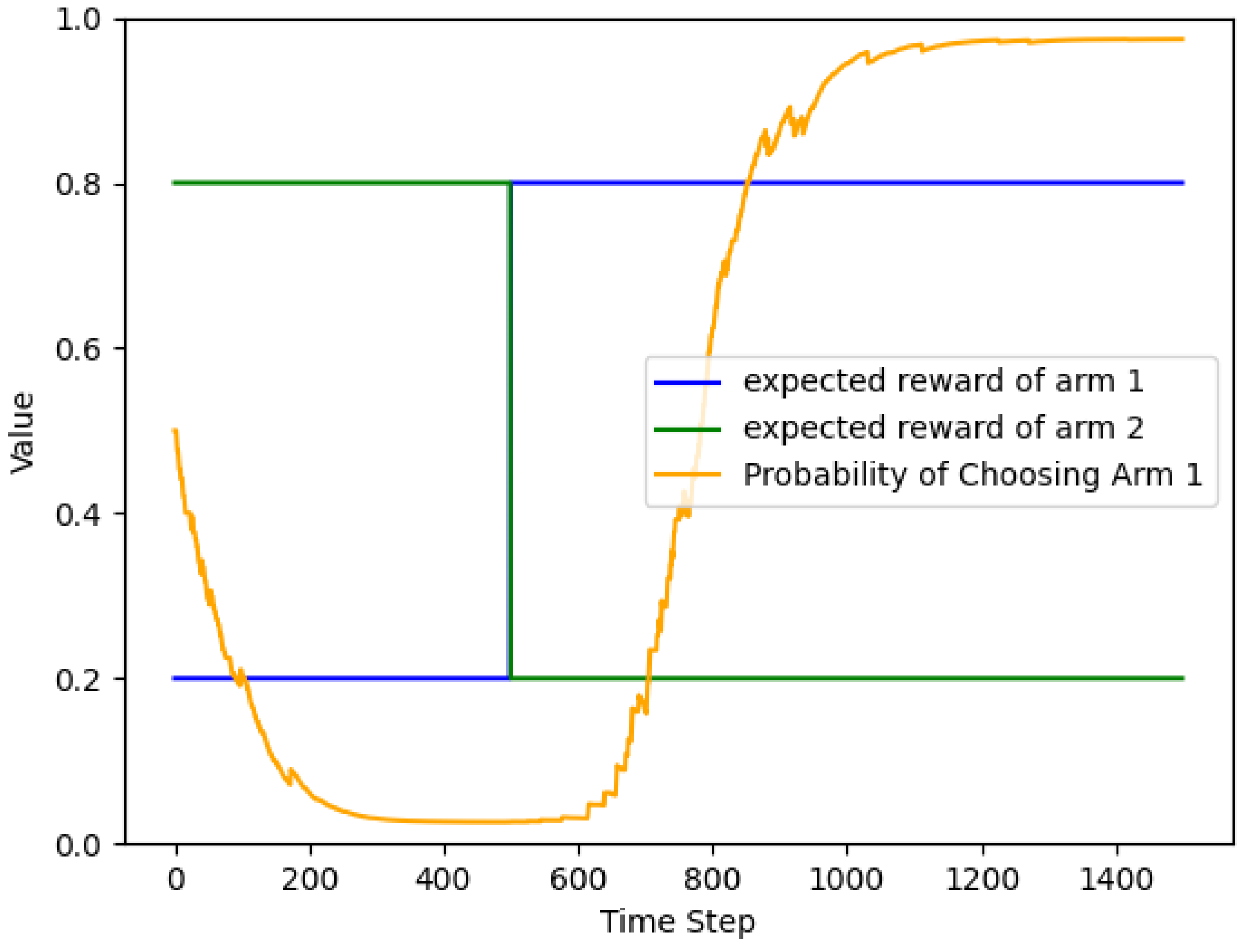
Define for all a:

$$\hat{r}_t(a) = \frac{r_t(a)}{\pi'_t(a)} \mathbb{I}\{a_t = a\}$$

Update policy:

$$\pi_{t+1}(a) = \frac{\pi_t(a) \exp(\eta \hat{r}_t(a))}{\sum_{a' \in \mathcal{A}} \pi_t(a') \exp(\eta \hat{r}_t(a'))}$$

Solution 1: Adding Extra Exploration



Regret Bound for Solution 1

Theorem. Exponential weights with Solution 1 ensures

$$\max_{a^\star} \mathbb{E}\left[\sum_{t=1}^{T}(r_t(a^\star) - r_t(a_t))\right] \le O\left(\frac{\ln A}{\eta} + \eta A T\right)$$

Solution 2: Reward Estimator with a Baseline

- The condition is only $\eta\hat{r}_t(a) \leq 1$. The reward estimator is allowed to be **very negative**!

- Still sample a_t from π_t, but construct the reward estimator as

$$\hat{r}_t(a) = \frac{r_t(a) - 1}{\pi_t(a)} \mathbb{I}\{a_t = a\} + 1$$

- Why this resolves the issue?

Applying Solution 2

$\pi_1(a) = 1/A$ for all a

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

Sample a_t from π_t, and observe $r_t(a_t)$

Define for all a:

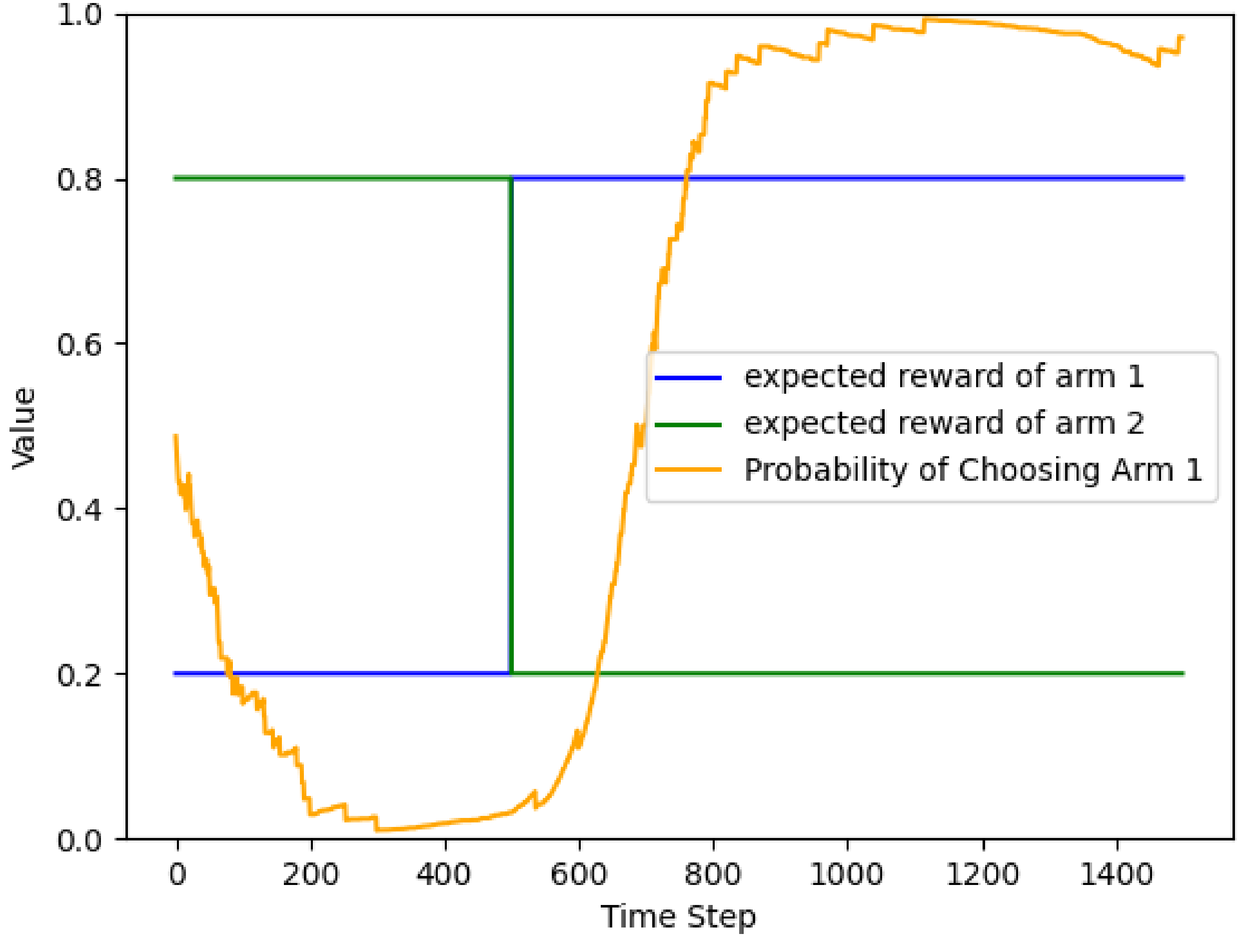
baseline

$$\hat{r}_t(a) = \frac{r_t(a) - 1}{\pi_t(a)} \mathbb{I}\{a_t = a\} + 1 \quad \text{or equivalently} \quad \hat{r}_t(a) = \frac{r_t(a) - 1}{\pi_t(a)} \mathbb{I}\{a_t = a\}$$

Update policy:

$$\pi_{t+1}(a) = \frac{\pi_t(a) \exp(\eta \hat{r}_t(a))}{\sum_{a' \in \mathcal{A}} \pi_t(a') \exp(\eta \hat{r}_t(a'))}$$

Solution 2: Reward Estimator with a Baseline



Regret Bound for Solution 2

Theorem. Exponential weights with Solution 2 ensures

$$\max_{a^\star} \mathbb{E}\left[\sum_{t=1}^{T}\left(r_t(a^\star) - r_t(a_t)\right)\right] \le O\left(\frac{\ln A}{\eta} + \eta A T\right)$$

EXP3 Algorithm

“**Exp**onential weight algorithm for **Exp**loration and **Exp**loitation”

- Exponential weights + either of the two solutions

Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, Robert Schapire.
The Nonstochastic Multiarmed Bandit Problem. 2002.

EXP3-IX

$\pi_1(a) = 1/A$ for all a

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

Sample a_t from π_t and observe $r_t(a_t)$

Define for all a:

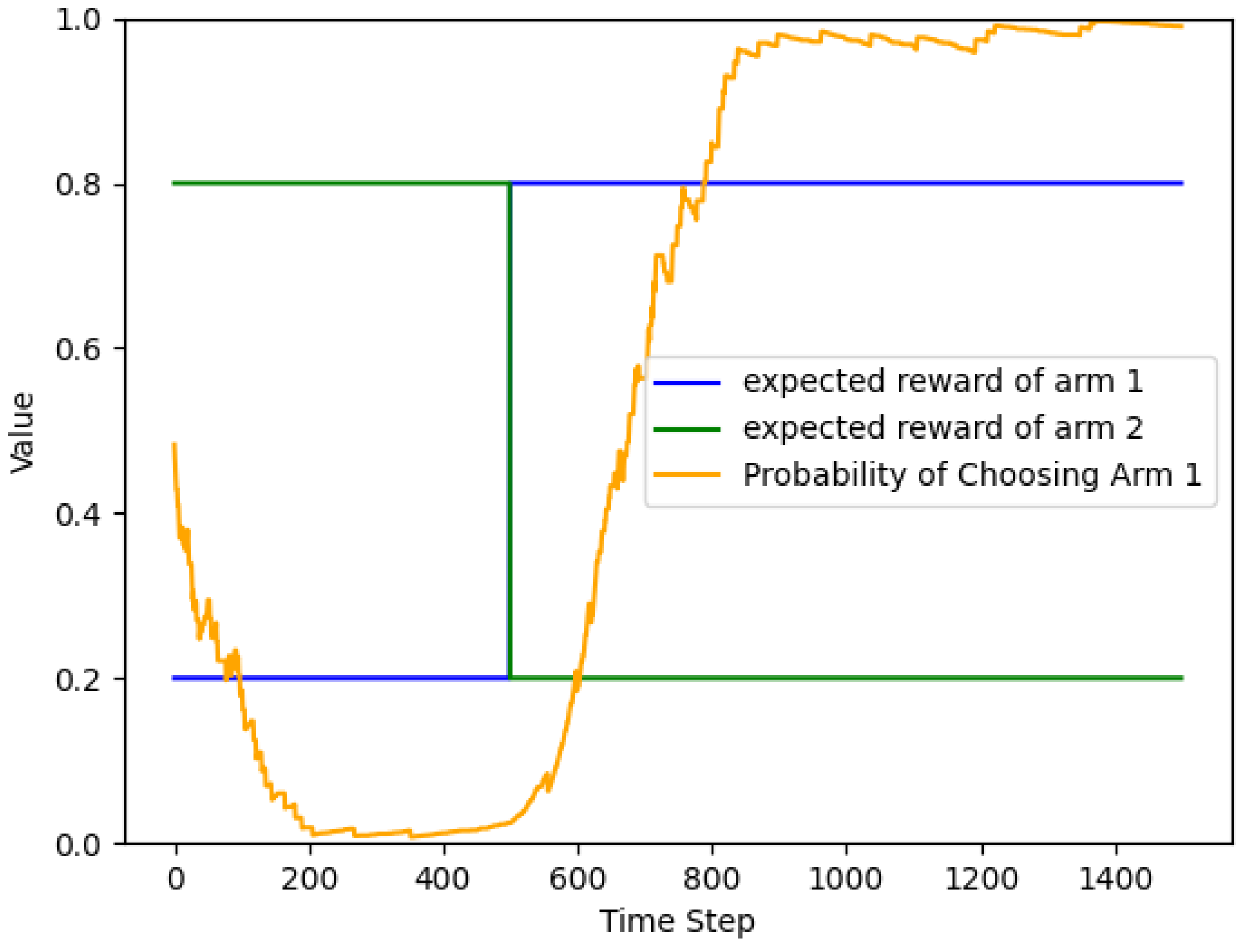
$$\hat{r}_t(a) = \frac{r_t(a) - 1}{\pi_t(a) + \eta}\mathbb{I}\{a_t = a\}$$

Update policy:

$$\pi_{t+1}(a) = \frac{\pi_t(a)\exp(\eta\hat{r}_t(a))}{\sum_{a'\in\mathcal{A}}\pi_t(a')\exp(\eta\hat{r}_t(a'))}$$

EXP3-IX

$$\hat{r}_t(a) = \frac{r_t(a) - 1}{\pi_t(a) + \eta} \mathbb{I}\{a_t = a\}$$



Regret Bound for EXP3-IX

Theorem. EXP3-IX ensures with high probability,

$$\max_{a^\star} \sum_{t=1}^{T} \left(r_t(a^\star) - r_t(a_t)\right) \le \tilde{O}\left(\frac{\ln A}{\eta} + \eta A T\right)$$

Gergely Neu. Explore no more: Improved high-probability regret bounds for non-stochastic bandits. 2015.

The Role of Baseline

$$\hat{r}_t(a) = \frac{r_t(a) - b_t}{\pi_t(a)} \mathbb{I}\{a_t = a\}$$

$$\pi_{t+1}(a) = \frac{\pi_t(a)\exp(\eta \hat{r}_t(a))}{\sum_{a' \in \mathcal{A}} \pi_t(a')\exp(\eta \hat{r}_t(a'))} \quad \text{or} \quad \pi_{t+1} = \operatorname*{argmax}_{\pi \in \Delta(\mathcal{A})} \left\{ \langle \pi, \hat{r}_t \rangle - \frac{1}{\eta} \mathrm{KL}(\pi, \pi_t) \right\}$$

Larger b_t: More exploratory (tends to decrease the probability of the action just chosen)

– needed to detect changes in the environment.

In fixed reward function setting (non-adversarial), we usually set b_t to be close to the recent performance level of the learner itself

- When finding an action better than the learner itself, increase its probability
- Otherwise, decrease its probability

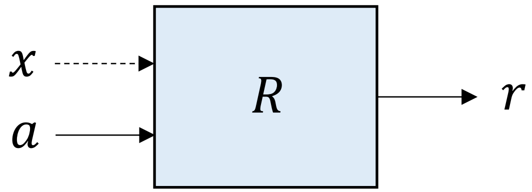
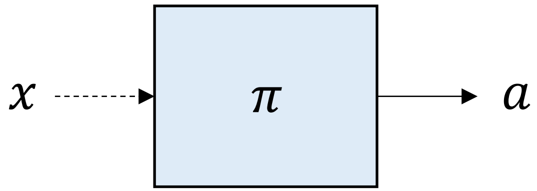
Summary

- Exponential weight update (EWU) is an effective algorithm for full-information setting. It guarantees sublinear regret even when the environment changes over time.

- Extending EWU to bandit with naïve unbiased reward estimator does not work (lack of exploration). Two ways to fix it:
  - Adding **extra uniform exploration** with probability $\geq A\eta$
  - Adding a **baseline** in the reward estimator to encourage exploration

- High-probability bounds can be achieved by adding **baseline** and **bias** (EXP3-IX).

Review: Exploration Strategies for Bandits

x: context, a: action, r: reward		MAB	CB
Value-based	 (context, action) to reward	Mean estimation + EG, BE, IGW Uncertainty as bonus	Regression + EG, BE, IGW
Policy-based	 context to action distribution	KL-regularized update with reward estimators (EXP3) + baseline, bias, uniform exploration	**Next**

Contextual Bandits

Contextual Bandits

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

Environment generates a context $x_t \in \mathcal{X}$

Learner chooses an action $a_t \in \mathcal{A}$

Learner observes $r_t(x_t, a_t)$

KL-Regularized Policy Updates

$$\pi_{t+1} = \operatorname*{argmax}_{\pi \in \Delta(\mathcal{A})} \left\{ \sum_a \pi(a)\hat{r}_t(a) - \frac{1}{\eta} \sum_a \pi(a) \log \frac{\pi(a)}{\pi_t(a)} \right\}$$

$$\hat{r}_t(a) = \frac{r_t(a) - b_t}{\pi_t(a)} \, \mathbb{I}\{a_t = a\}$$

$$\theta_{t+1} = \operatorname*{argmax}_{\theta} \left\{ \sum_a \pi_\theta(a|x_t) \, \hat{r}_t(x_t, a) - \frac{1}{\eta} \sum_a \pi_\theta(a|x_t) \log \frac{\pi_\theta(a|x_t)}{\pi_{\theta_t}(a|x_t)} \right\}$$

$$\hat{r}_t(x_t, a) = \frac{r_t(x_t, a) - b_t(x_t)}{\pi_{\theta_t}(a|x_t)} \, \mathbb{I}\{a_t = a\}$$

KL-Regularized Policy Updates

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

Receive context x_t

Take action $a_t \sim \pi_{\theta_t}(\cdot|x_t)$ and receive reward $r_t(x_t, a_t)$

Create reward estimator $\hat{r}_t(x_t, a) = \frac{r_t(x_t,a) - b_t(x_t)}{\pi_{\theta_t}(a|x_t)} \mathbb{I}\{a_t = a\}$

Update

$$\theta_{t+1} = \underset{\theta}{\operatorname{argmax}} \left\{ \sum_a \pi_\theta(a|x_t)\, \hat{r}_t(x_t, a) - \frac{1}{\eta} \sum_a \pi_\theta(a|x_t) \log \frac{\pi_\theta(a|x_t)}{\pi_{\theta_t}(a|x_t)} \right\}$$

Proximal Policy Optimization (PPO) for CB

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

For $i = 1, \dots, N$:

Receive context x_i

Take action $a_i \sim \pi_{\theta_t}(\cdot|x_i)$ and receive reward $r_i(x_i, a_i)$

Create reward estimator $\hat{r}_i(x_i, a) = \frac{r_i(x_i,a) - b_t(x_i)}{\pi_{\theta_t}(a|x_i)} \mathbb{I}\{a_i = a\}$

For $j = 1, \dots, M$:

one iteration of mirror ascent

For minibatch $\mathcal{B} \subset \{1, 2, \dots, N\}$ of size B:

$$\theta \leftarrow \theta + \nabla_\theta \frac{1}{B} \sum_{i \in \mathcal{B}} \left(\sum_a \pi_\theta(a|x_i)\, \hat{r}_i(x_i, a) - \frac{1}{\eta} \sum_a \pi_\theta(a|x_i) \log \frac{\pi_\theta(a|x_i)}{\pi_{\theta_t}(a|x_i)} \right)$$

$$= \theta + \nabla_\theta \frac{1}{B} \sum_{i \in \mathcal{B}} \left(\frac{\pi_\theta(a_i|x_i)}{\pi_{\theta_t}(a_i|x_i)} \left(r_i(x_i, a_i) - b_t(x_i)\right) - \frac{1}{\eta} \sum_a \pi_\theta(a|x_i) \log \frac{\pi_\theta(a|x_i)}{\pi_{\theta_t}(a|x_i)} \right)$$

$\theta_{t+1} \leftarrow \theta$

Proximal Policy Optimization (PPO) for CB

$$\theta \leftarrow \theta + \nabla_\theta \frac{1}{B} \sum_{i \in \mathcal{B}} \left(\frac{\pi_\theta(a_i|x_i)}{\pi_{\theta_t}(a_i|x_i)} (r_i(x_i, a_i) - b_t(x_i)) - \frac{1}{\eta} \underbrace{\sum_a \pi_\theta(a|x_i) \log \frac{\pi_\theta(a|x_i)}{\pi_{\theta_t}(a|x_i)}}_{\mathrm{KL}\left(\pi_\theta(\cdot|x_i), \pi_{\theta_t}(\cdot|x_i)\right)} \right)$$

- May replace $\mathrm{KL}\left(\pi_\theta(\cdot|x_i), \pi_{\theta_t}(\cdot|x_i)\right)$ by $\mathrm{KL}\left(\pi_{\theta_t}(\cdot|x_i), \pi_\theta(\cdot|x_i)\right)$. The latter is easier to construct unbiased estimator.

- Although this term can be calculated exactly, we often use samples to estimate it (so we do not need to sum over a)

Estimating KL by Samples

http://joschu.net/blog/kl-approx.html

Sample $a_i \sim \pi_{\theta_t}(\cdot\,|x_i)$ and define $kl_i(\theta_t, \theta) = \boxed{\frac{\pi_\theta(a_i|x_i)}{\pi_{\theta_t}(a_i|x_i)} - 1 - \log \frac{\pi_\theta(a_i|x_i)}{\pi_{\theta_t}(a_i|x_i)}}$

Then $\mathbb{E}_{a_i \sim \pi_{\theta_t}(\cdot|x_i)}[kl_i(\theta_t, \theta)] = \mathrm{KL}\left(\pi_{\theta_t}(\cdot\,|x_i), \pi_\theta(\cdot\,|x_i)\right)$

Just need one sample of a_i

As we see before, the ways to construct an unbiased estimator are not unique.

This is a good one with low variance.

PPO with KL Estimator

$$kl_i(\theta_t, \theta) = \frac{\pi_\theta(a_i|x_i)}{\pi_{\theta_t}(a_i|x_i)} - 1 - \log \frac{\pi_\theta(a_i|x_i)}{\pi_{\theta_t}(a_i|x_i)}$$

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

For $i = 1, \dots, N$:

Receive context x_i

Take action $a_i \sim \pi_{\theta_t}(\cdot|x_i)$ and receive reward $r_i(x_i, a_i)$

Create reward estimator $\hat{r}_i(x_i, a) = \frac{r_i(x_i,a) - b_t(x_i)}{\pi_{\theta_t}(a|x_i)} \mathbb{I}\{a_i = a\}$

For $j = 1, \dots, M$:

For minibatch $\mathcal{B} \subset \{1, 2, \dots, N\}$ of size B:

$$\theta \leftarrow \theta + \nabla_\theta \frac{1}{B} \sum_{i \in \mathcal{B}} \left(\frac{\pi_\theta(a_i|x_i)}{\pi_{\theta_t}(a_i|x_i)} \left(r_i(x_i, a_i) - b_t(x_i)\right) - \frac{1}{\eta} kl_i(\theta_t, \theta) \right)$$

$\theta_{t+1} \leftarrow \theta$

Summary: PPO

- PPO-CB can be viewed as an extension of EXP3 to contextual bandits. The central idea is KL-regularized policy updates
- Common techniques: baselines, avoiding **overly positive** reward estimator. These techniques prevent over exploitation
- PPO additional uses batching, reversed KL divergence, and KL estimators for computational efficiency

NPG and PG

Natural Policy Gradient

(PPO) $\theta_{t+1} = \underset{\theta}{\operatorname{argmax}} \ \mathbb{E}_x \left[\sum_a \left(\pi_\theta(a|x) - \pi_{\theta_t}(a|x) \right) \hat{r}_t(x,a) - \frac{1}{\eta} \sum_a \pi_\theta(a|x) \log \frac{\pi_\theta(a|x)}{\pi_{\theta_t}(a|x)} \right]$

η close to zero

(NPG) $\theta_{t+1} = \theta_t + \eta F_t^{-1} \ \mathbb{E}_x \left[\sum_a \nabla_\theta \pi_\theta(a|x) \, \hat{r}_t(x,a) \right] \Bigg|_{\theta=\theta_t}$

where $F_{\theta_t} = \mathbb{E}_x \mathbb{E}_{a \sim \pi_{\theta_t}(\cdot|x)} \left[\left(\nabla_\theta \log \pi_\theta(a|x) \right) \left(\nabla_\theta \log \pi_\theta(a|x) \right)^\top \right] \Big|_{\theta=\theta_t}$ **Fisher information matrix**

Natural Policy Gradient (w/o context + full-info)

(PPO)

$$\theta_{t+1} = \operatorname*{argmax}_{\theta} \sum_a \left(\pi_\theta(a) - \pi_{\theta_t}(a)\right) r_t(a) - \frac{1}{\eta} \sum_a \pi_\theta(a) \log \frac{\pi_\theta(a)}{\pi_{\theta_t}(a)}$$

η close to zero

(NPG)

$$\theta_{t+1} = \theta_t + \eta F_{\theta_t}^{-1} \sum_a \nabla_\theta \pi_\theta(a)\, r_t(a) \Big|_{\theta=\theta_t}$$

where $F_{\theta_t} = \mathbb{E}_{a\sim\pi_{\theta_t}}[(\nabla_\theta \log \pi_\theta(a))(\nabla_\theta \log \pi_\theta(a))^\top]\Big|_{\theta=\theta_t}$ **Fisher information matrix**

Proof Sketch

$$f(\theta) \approx f(\theta_t) + (\theta - \theta_t)^\top [\nabla_\theta f(\theta)]_{\theta=\theta_t} + \frac{1}{2}(\theta - \theta_t)^\top \left[\nabla_\theta^2 f(\theta)\right]_{\theta=\theta_t} (\theta - \theta_t)$$

PPO

$$\theta_{t+1} = \operatorname*{argmax}_\theta \left\{ \langle \pi_\theta - \pi_{\theta_t}, r_t \rangle - \frac{1}{\eta} \mathrm{KL}(\pi_\theta, \pi_{\theta_t}) \right\}$$

$$\begin{aligned} \langle \pi_\theta - \pi_{\theta_t}, r_t \rangle &= \sum_a \left(\pi_\theta(a) - \pi_{\theta_t}(a) \right) r_t(a) \\ &\approx (\theta - \theta_t)^\top \sum_a [\nabla_\theta \pi_\theta(a)]_{\theta=\theta_t} r_t(a) \end{aligned}$$

$$F_{\theta_t} = \left[\nabla_\theta^2 \mathrm{KL}\left(\pi_\theta, \pi_{\theta_t}\right)\right]_{\theta=\theta_t}$$ **(exercise)**

$$\mathrm{KL}\left(\pi_\theta, \pi_{\theta_t}\right) \approx \frac{1}{2} (\theta - \theta_t)^\top F_{\theta_t} (\theta - \theta_t)$$

$$\begin{aligned} \theta_{t+1} &\approx \operatorname*{argmax}_\theta \left\{ (\theta - \theta_t)^\top g_t - \frac{1}{2\eta} (\theta - \theta_t)^\top F_{\theta_t} (\theta - \theta_t) \right\} \\ &= \theta_t + \eta F_{\theta_t}^{-1} g_t \end{aligned}$$ **NPG**

NPG vs. PG

NPG

$$\theta_{t+1} = \theta_t + \eta F_t^{-1} \sum_a \nabla_\theta \pi_\theta(a)\, r_t(a) \Big|_{\theta=\theta_t}$$

(Vanilla) PG

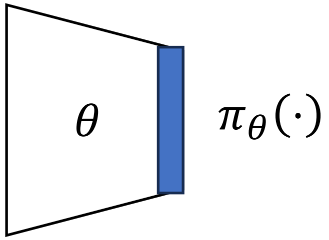
$$\theta_{t+1} = \theta_t + \eta \sum_a \nabla_\theta \pi_\theta(a)\, r_t(a) \Big|_{\theta=\theta_t}$$

NPG vs. PG

NPG

$$\theta_{t+1} = \theta_t + \eta F_t^{-1} \sum_a \nabla_\theta \pi_\theta(a) r_t(a) \Big|_{\theta=\theta_t}$$

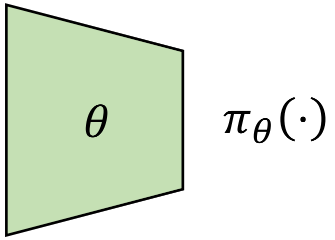
$$\theta_{t+1} = \underset{\theta}{\operatorname{argmax}} \langle \pi_\theta - \pi_{\theta_t}, r_t \rangle - \frac{1}{\eta} \mathrm{KL}(\pi_\theta, \pi_{\theta_t})$$



PG

$$\theta_{t+1} = \theta_t + \eta \sum_a \nabla_\theta \pi_\theta(a)\, r_t(a) \Big|_{\theta=\theta_t}$$

$$\theta_{t+1} = \underset{\theta}{\operatorname{argmax}} \langle \pi_\theta - \pi_{\theta_t}, r_t \rangle - \frac{1}{2\eta} \|\theta - \theta_t\|^2$$



Example: NPG vs. PG with softmax policy

Consider multi-armed bandits with **softmax policy** $\pi_\theta(a) = \frac{e^{\theta(a)}}{\sum_{a'} e^{\theta(a')}}$ parameterized by $\theta(1), \theta(2), \dots, \theta(A)$

$$\tilde{r}_t(a) = r_t(a) - \sum_{a'} \pi_{\theta_t}(a') r_t(a')$$

NPG (= Exponential Weight, without requiring $\eta \approx 0$ assumption)

For $t = 1,2, \dots$

$$\theta_{t+1}(a) \leftarrow \theta_t(a) + \eta r_t(a)$$

Check the equivalence (exercise)

NPG can also be written as

$$\theta_{t+1}(a) \leftarrow \theta_t(a) + \eta \tilde{r}_t(a)$$

PG

For $k = 1,2, \dots$

$$\theta_{t+1}(a) \leftarrow \theta_t(a) + \eta \pi_{\theta_t}(a) \tilde{r}_t(a)$$

NPG (EW) vs. PG

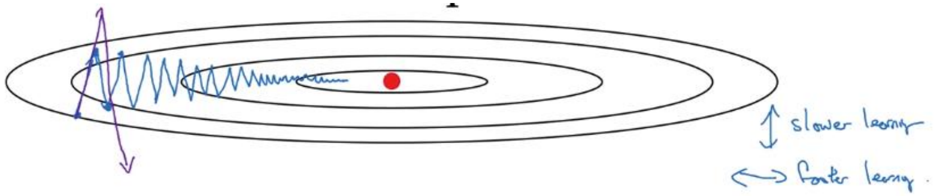
EW: $\theta_{t+1}(a) \leftarrow \theta_t(a) + \eta \tilde{r}_t(a)$

PG: $\theta_{t+1}(a) \leftarrow \theta_t(a) + \eta \pi_{\theta_t}(a) \tilde{r}_t(a)$

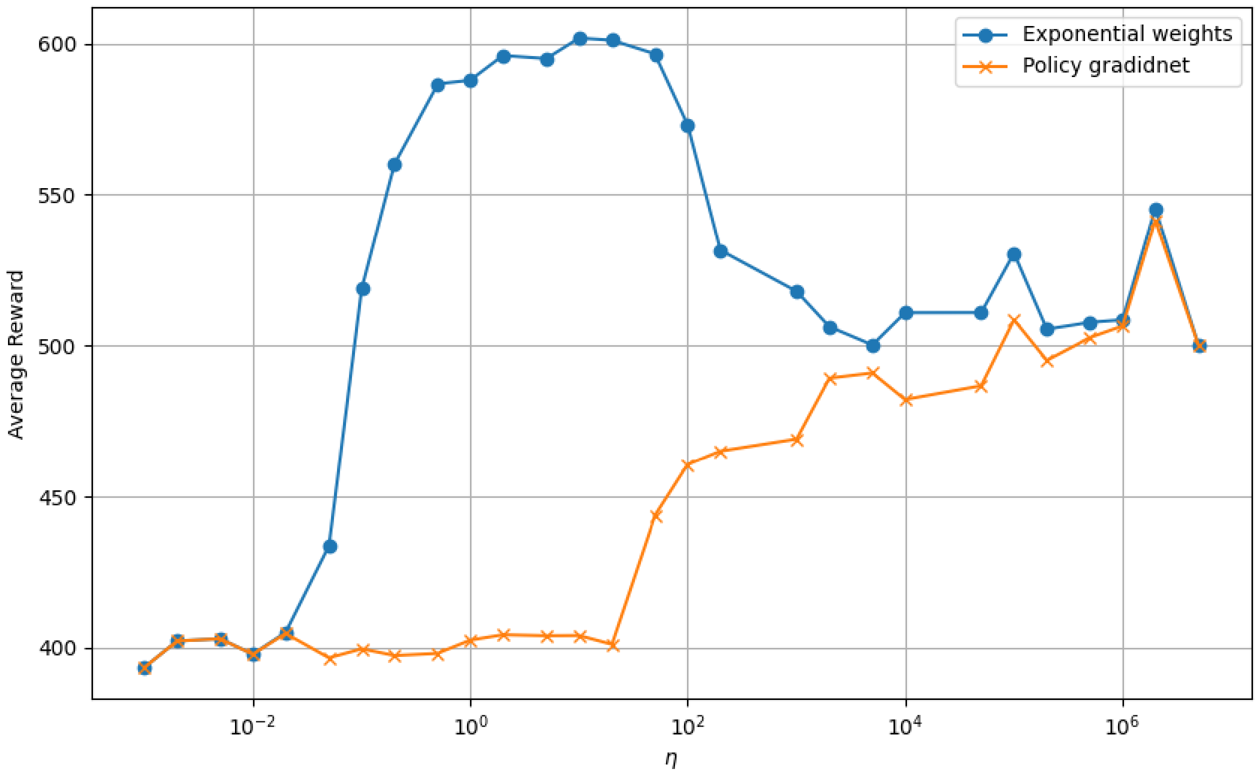
Reward = $[\mathrm{Ber}(0.6), \mathrm{Ber}(0.4)]$

Initial policy $\pi = [0.0001, 0.9999]$

Plot total reward in 1000 rounds



https://math.stackexchange.com/questions/2285282/relating-condition-number-of-hessian-to-the-rate-of-convergence



NPG and PG with bandit feedback

$$\theta_{t+1} = \theta_t + \eta F_t^{-1} \sum_a \nabla_\theta \pi_\theta(a) \hat{r}_t(a) \Big|_{\theta=\theta_t}$$

$$\theta_{t+1} = \theta_t + \eta \sum_a \nabla_\theta \pi_\theta(a)\, \hat{r}_t(a) \Big|_{\theta=\theta_t}$$

PG (REINFORCE) for contextual bandits

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

Receive context x_t

Take action $a_t \sim \pi_{\theta_t}(\cdot|x_t)$ and receive reward $r_t(x_t, a_t)$

Update

$$\theta_{t+1} \leftarrow \theta_t + \eta \left[\nabla_\theta \log \pi_\theta(a_t|x_t)\right]_{\theta=\theta_t} \left(r_t(x_t, a_t) - b_t(x_t)\right)$$

Or simply written as

$$\theta \leftarrow \theta + \eta \underbrace{\nabla_\theta \log \pi_\theta(a_t|x_t)}_{\text{Coming from inverse propensity weighting / importance weighting}}\left(r_t(x_t, a_t) - b_t(x_t)\right)$$

Verify (again) that reward offset does not affect the algorithm

Natural Policy Gradient

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

Receive context x_t

Take action $a_t \sim \pi_{\theta_t}(\cdot|x_t)$ and receive reward $r_t(x_t, a_t)$

Update

$$\theta_{t+1} \leftarrow \theta_t + \eta F_{\theta_t}^{-1} \left[\nabla_\theta \log \pi_\theta(a_t|x_t)\right]_{\theta=\theta_t} \left(r_t(x_t, a_t) - b_t(x_t)\right)$$

A naïve calculation of $F_{\theta_t}^{-1}$ will take $O(d^3)$ time

Sample-Based NPG*

A naïve calculation of $F_{\theta_t}^{-1}$ will take $O(d^3)$ time

But we can actually view $h_t \coloneqq F_{\theta_t}^{-1} g_t$ as a solution of a linear regression problem

$$\theta_{t+1} = \theta_t + \eta F_{\theta_t}^{-1} \mathbb{E}_{a\sim\pi_{\theta_t}}\left[\left(\nabla_\theta \log \pi_{\theta_t}(a)\right) r_t(a)\right]$$

where $F_{\theta_t} = \mathbb{E}_{a\sim\pi_{\theta_t}}\left[\left(\nabla_\theta \log \pi_{\theta_t}(a)\right)\left(\nabla_\theta \log \pi_{\theta_t}(a)\right)^\top\right]$

$$h_t = \left(\mathbb{E}_{a\sim\pi_{\theta_t}}[\phi_t(a)\phi_t(a)]\right)^{-1} \mathbb{E}_{a\sim\pi_{\theta_t}}[\phi_t(a) r_t(a)]$$

$$\phi_t(a) = \nabla_\theta \log \pi_{\theta_t}(a)$$

$$= \operatorname*{argmin}_h \mathbb{E}_{a\sim\pi_{\theta_t}}[(\phi_t(a)^\top h - r_t(a))^2]$$

Summary: Policy Learning in Bandits

PG	PPO / NPG
$\theta_{t+1} = \underset{\theta}{\mathrm{argmax}} \langle \pi_\theta - \pi_{\theta_t}, \hat{r}_t \rangle - \frac{1}{2\eta} \|\theta - \theta_t\|^2$	$\theta_{t+1} = \underset{\theta}{\mathrm{argmax}} \langle \pi_\theta - \pi_{\theta_t}, \hat{r}_t \rangle - \frac{1}{\eta} \mathrm{KL}(\pi_\theta, \pi_{\theta_t})$
$\theta \leftarrow \theta + \eta \nabla_\theta \langle \pi_\theta, \hat{r}_t \rangle$	$\theta \leftarrow \theta + \eta F_\theta^{-1} \nabla_\theta \langle \pi_\theta, \hat{r}_t \rangle$
$\hat{r}_t(a) = \frac{r_t(a) - b_t}{\pi_{\theta_t}(a)} \mathbb{I}\{a = a_t\}$	
$\theta \leftarrow \theta + \eta \nabla_\theta \log \pi_\theta(a_t)\ (r_t(a_t) - b_t)$	$\theta \leftarrow \theta + \eta F_\theta^{-1}\ \nabla_\theta \log \pi_\theta(a_t)\ (r_t(a_t) - b_t)$

$$F_\theta = \mathbb{E}_{a \sim \pi_\theta}[(\nabla_\theta \log \pi_\theta(a))(\nabla_\theta \log \pi_\theta(a))^\top]$$