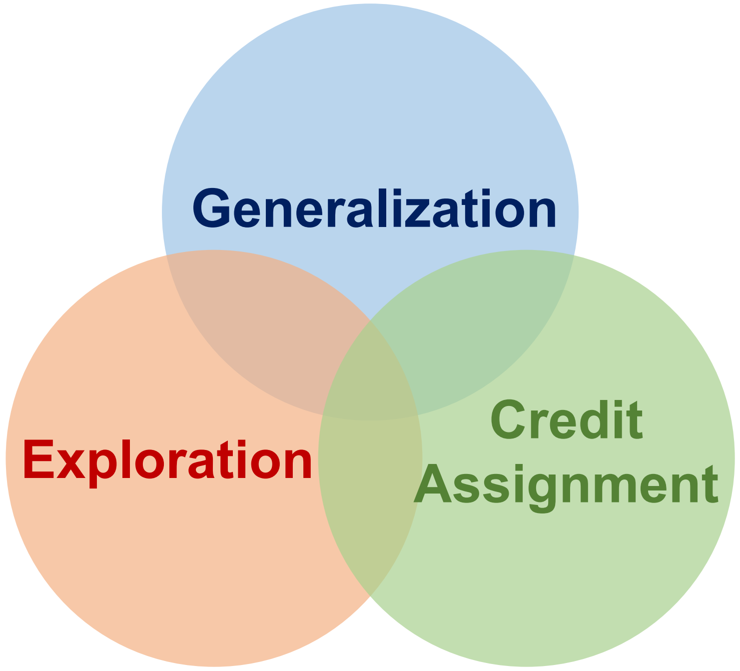
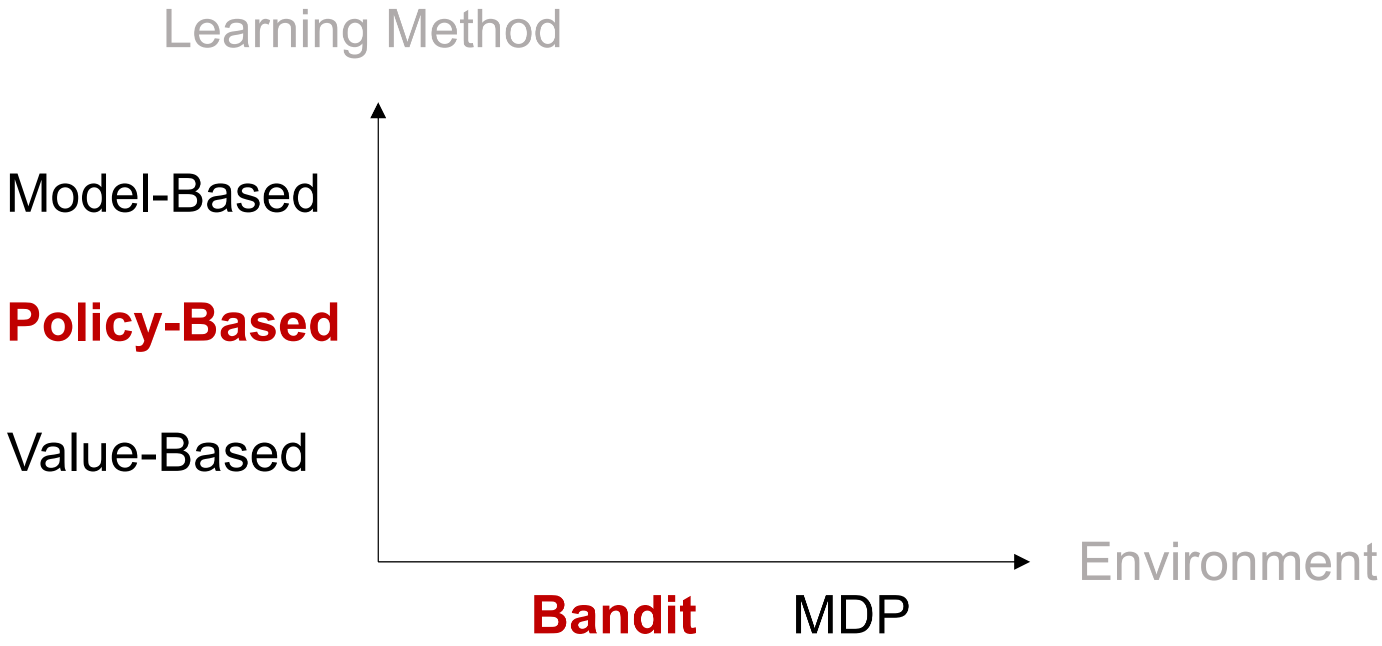
# Bandits 2

Chen-Yu Wei

# Roadmap


Generalization
Exploration
Credit Assignment

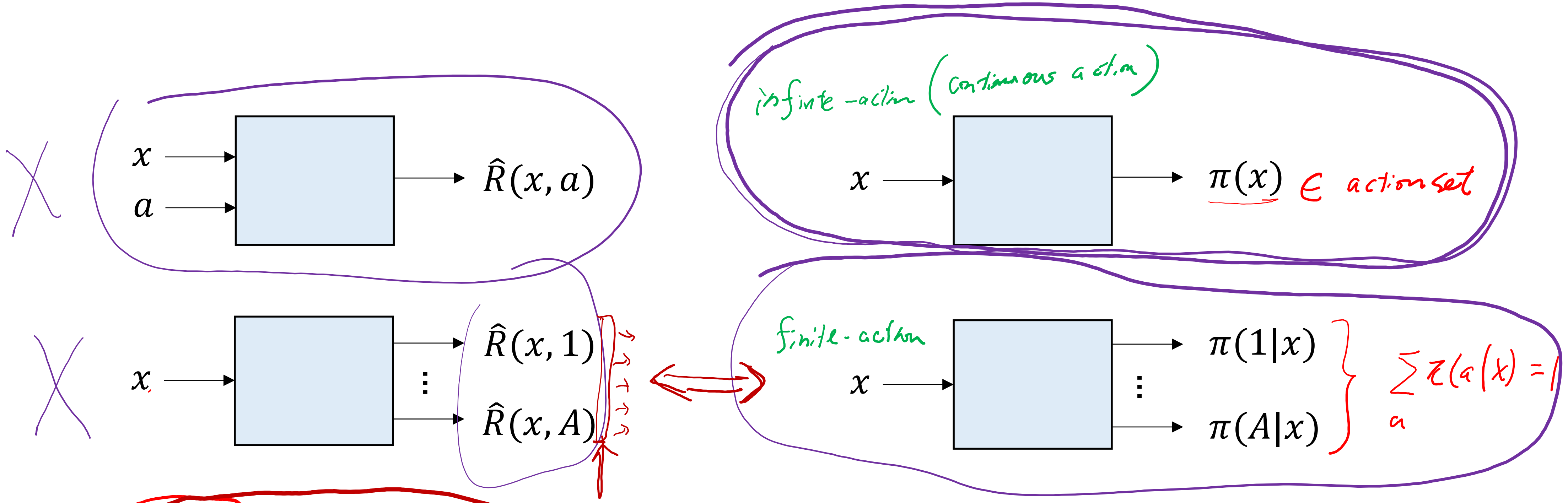

Learning Method
Model-Based
Policy-Based
Value-Based
Environment
Bandit
MDP

# Policy-Based Bandits

- Key challenges: **Exploration** and **Generalization (if there are contexts)**
- Algorithms we will discuss:
  - KL-regularized policy updates (PPO)
  - Policy gradient (REINFORCE)

# Policy-Based Bandits

$x$: context, $a$: action




$$\pi(a|x) \propto \exp\left(\lambda \hat{R}(x,a)\right)$$

softmax

**Value-based** approach

**Policy-based** approach

# Policy-Based Bandits

Why policy-based bandit algorithms?

- Actually, in finite-action contextual bandit problems, value- and policy-based approaches are almost equivalent.
- But we have to use policy-based approaches to handle **continuous action space.**
- They are also different in MDPs. (later in the course)

# The Full-Information MAB

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

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

The learner chooses an action $a_t$

Environment reveals the reward $r_t(a) = R(a) + w_t(a)$ **of all actions**

**Policy-based algorithm:** Maintain a distribution $\pi_t(a)$ and update it with feedback

Sample $a_t \sim \pi_t$

How should we update from $\pi_t$ to $\pi_{t+1}$ using $r_t(1), \dots, r_t(A)$?

# Algorithm for the Full-Information MAB

$$f(\pi) = \sum_{a=1}^{A} \pi(a)R(a)$$ ← We want to find a $\pi$ that **maximizes** this value

But we don't know $R(a)$

But we get noisy samples of $R(a)$, i.e., $r_t(a)$

# Gradient Ascent

$$f(\pi) = \sum_{a=1}^{A} \pi(a) R(a) \qquad \Rightarrow \quad \nabla_\pi f(\pi) = R$$

Gradient Ascent

For $t = 1, 2 \ldots$

$$\pi_{t+1} \leftarrow \pi_t + \eta R$$

$$\pi_{t+1} \leftarrow \Pi(\pi_{t+1})$$

Stochastic Gradient Ascent

For $t = 1, 2 \ldots$

$$\pi_{t+1} \leftarrow \pi_t + \eta r_t$$

$$\pi_{t+1} \leftarrow \Pi(\pi_{t+1})$$

# Exponential Weight Update

For $t = 1, 2 \ldots$

$$\pi_{t+1}(a) \propto \pi_t(a)\, e^{\eta r_t(a)}$$

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

Better for bandit problems (because we never get $\pi_t(a) = 0$)

# Exponential Weight Update = KL-Regularized Policy Updates

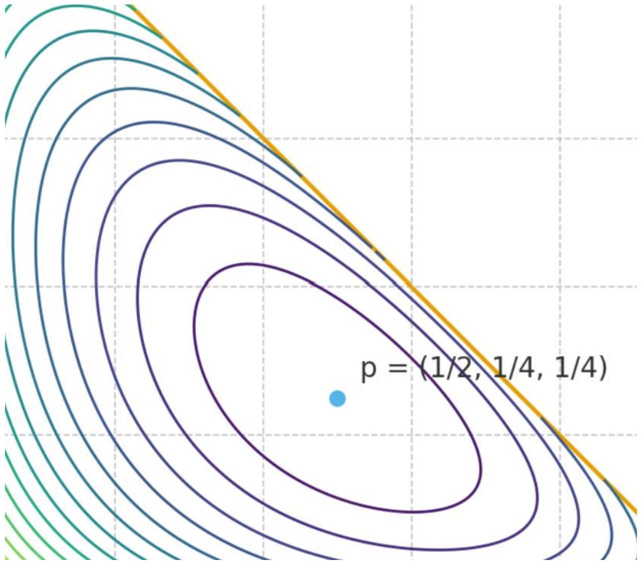
$$\pi_{t+1}(a) = \frac{\pi_t(a)\, e^{\eta r_t(a)}}{\sum_{b \in \mathcal{A}} \pi_t(b)\, e^{\eta r_t(b)}}$$

$$\Leftrightarrow$$

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

# KL Divergence – A Distance Measure for Distributions

$$\mathrm{KL}(\pi, \pi') = \sum_a \pi(a) \log \frac{\pi(a)}{\pi'(a)}$$




$\mathrm{KL}(\pi, \pi') \geq 0$

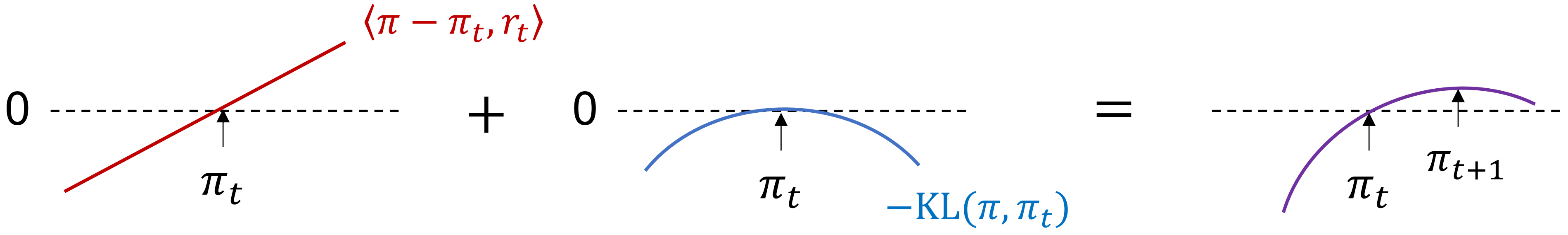
$\mathrm{KL}(\pi, \pi') = 0$ if and only if $\pi = \pi'$

$\mathrm{KL}(\pi, \pi') \neq \mathrm{KL}(\pi', \pi)$

# Regularized Policy Updates

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

$$= \underset{\pi \in \Delta(\mathcal{A})}{\operatorname{argmax}} \left\{ \underbrace{\sum_a \big(\pi(a) - \pi_t(a)\big) r_t(a)}_{\text{The Improvement of } \pi \text{ over } \pi_t \text{ on } r_t} - \frac{1}{\eta} \mathrm{KL}(\pi, \pi_t) \right\}$$



# Multi-Armed Bandits

# Adversarial Multi-Armed Bandits

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

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

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

Learner observes $r_t(a_t) = R(a_t) + w_t$

# Recall: Exponential Weight Updates

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

# Exponential Weight Updates for Bandits?

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

No longer observable

Only update the arm that we choose?

# Exponential Weight Updates for Bandits?

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

- $\hat{r}_t(a)$ 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

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

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

Inverse Propensity Weighting / Inverse Probability Weighting / Importance Weighting

# Directly Applying Exponential Weights

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

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

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

Define for all $a$:

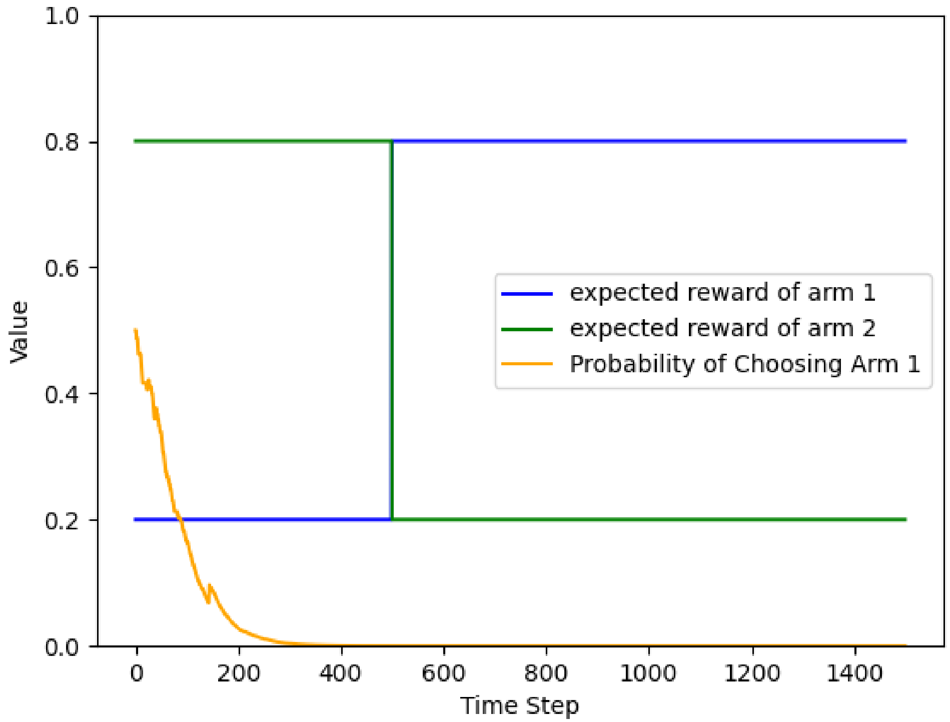
$$\hat{r}_t(a) = \frac{r_t(a)}{\pi_t(a)} \mathbb{I}\{a_t = a\}$$

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 $t \leq 500,\ r_t = [\mathrm{Bernoulli}(0.2), \mathrm{Bernoulli}(0.8)]$
- For $500 < t \leq 1500,\ r_t = [\mathrm{Bernoulli}(0.8), \mathrm{Bernoulli}(0.2)]$
- [code]



# Solution 1: Adding Extra Exploration

- **Idea:** use at least $\eta$ probability to choose each arm

- Instead of sampling $a_t$ according to $\pi_t$, 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 $\pi'_t = (1 - A\eta)\pi_t + A\eta\ \mathrm{uniform}(\mathcal{A})$, and observe $r_t(a_t)$

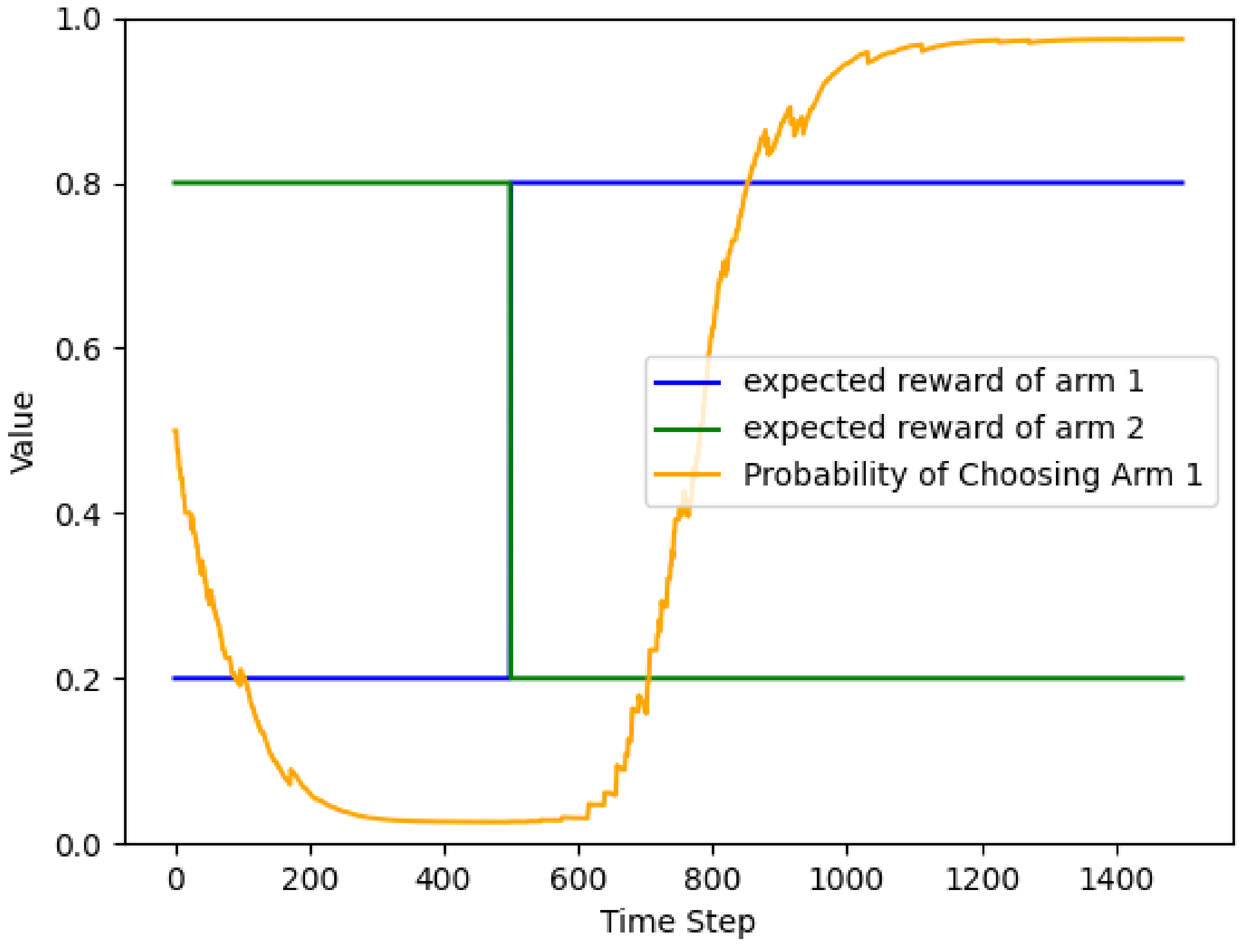
Define for all $a$:

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

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



# Solution 2: Reward Estimator with a Baseline

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

  The fact that mirror ascent **cannot handle** very positive unbiased reward estimator but **can handle** a negative one is somewhat technical in the proof.

- Still sample $a_t$ from $\pi_t$, 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 $t = 1,2,\ldots,T$:

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

Define for all $a$:

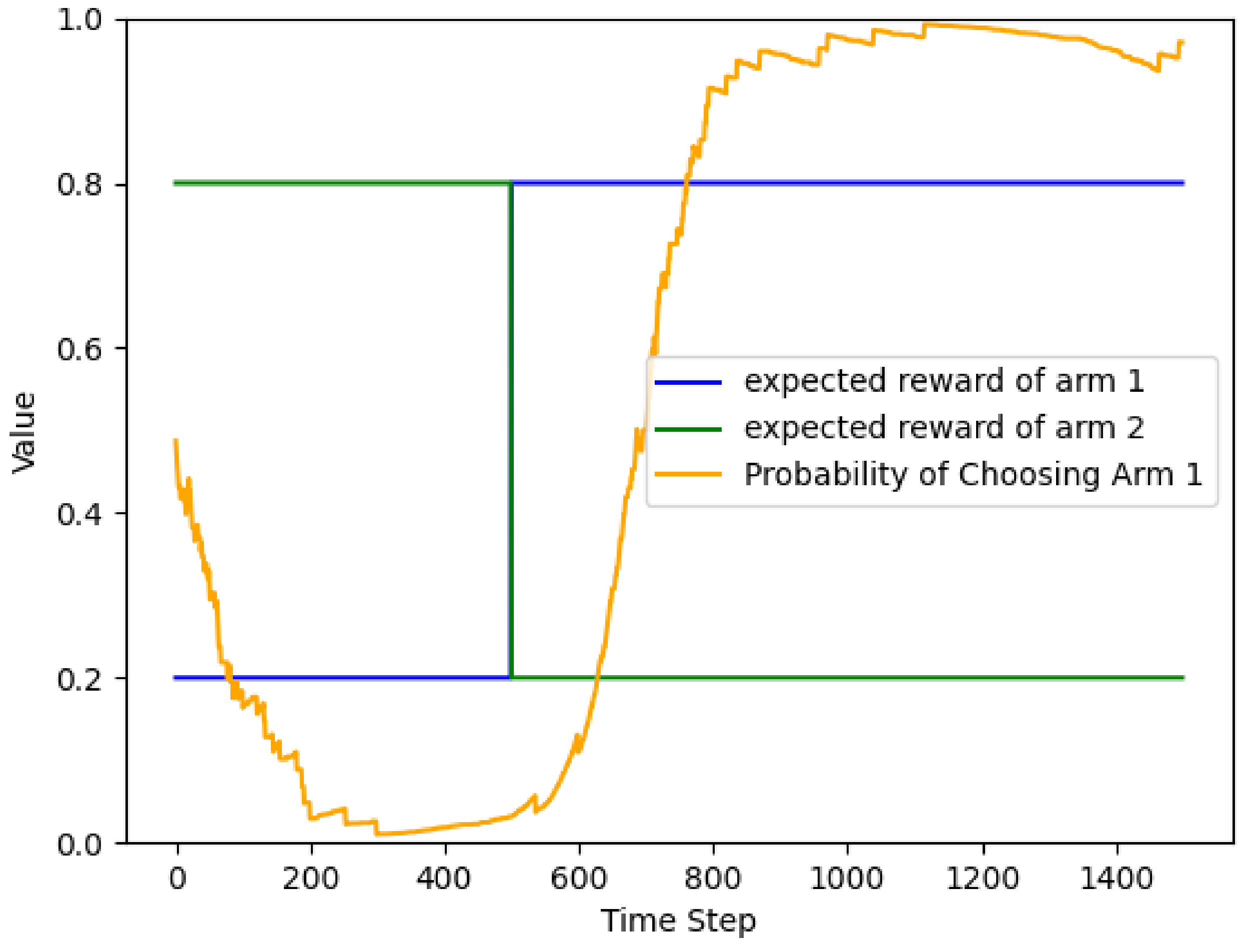
baseline

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

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



# 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.

# 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

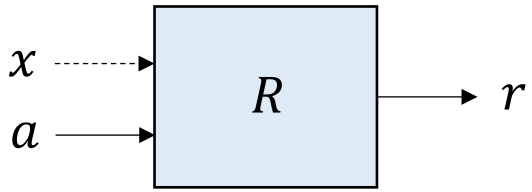
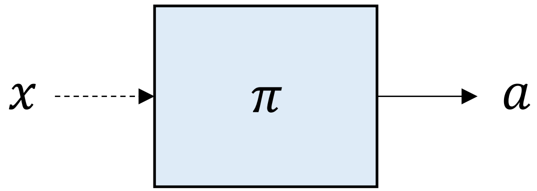
- Exponential weight update (EWU) is an effective algorithm for full-information setting. It guarantees sublinear regret even when the environment changes over time.

- 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<br><br>Uncertainty as bonus | Regression + EG, BE |
| Policy-based |   context to action distribution | KL-regularized update with reward estimators (EXP3) + baseline, uniform exploration | **Next** |

# Contextual Bandits

# Contextual Bandits

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

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

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

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

In practice, set $b_t$ as a **running average** of $r_t(a_t)$ to track the learner's own performance.

The larger $b_t$ is, the more exploration.

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

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

# 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

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

# KL-Regularized Policy Updates with Batches (PPO for CB)

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

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

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 $j = 1, \ldots, M$:

one iteration of mirror ascent

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

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

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

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

# KL-Regularized Policy Updates with Batches (PPO for CB)

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

- 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 (more on this next slide)

- Although this term can be calculated exactly, we often use samples to estimate it (so we do not need the summation 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 $a_i$

It is left as your exercise to verify this.

As we see before, the ways to construct an unbiased estimator are not unique.
This is a good one with low variance (check the link above).

We constructed unbiased reward estimators because of **lack of information**.
Here, we construct unbiased KL estimator only to **save computation**.
(replacing exact calculation by sampling)

# PPO with KL Estimator

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 $B$:

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

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

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

# Summary: PPO (Proximal Policy Optimization)

- PPO-CB can be viewed as an extension of EXP3 to contextual bandits. The central idea is KL-regularized policy updates
- PPO-CB additional uses batching, reversed KL divergence, and unbiased KL estimators for computational efficiency
- PPO is a strong algorithm for RL in MDPs
  - It is stable as it makes conservative updates in every iteration
  - It has nice theoretical guarantee in multi-armed bandits (equivalent to EXP3)
  - There is one more technique to further stabilize it: clipping the policy improvement part so that it is not overly positive --- more on this when we revisit this algorithm in MDPs.

# NPG and PG

# Recall: Two Equivalent Forms of EW / PPO

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

Regularization form

Gradient-update form

# Natural Policy Gradient

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

$\eta$ 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)**

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

$\eta$ close to zero

**(NPG)**

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

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**

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

$$\langle \pi_\theta - \pi_{\theta_t}, r_t \rangle = \sum_a \left( \pi_\theta(a) - \pi_{\theta_t}(a) \right) r_t(a)$$

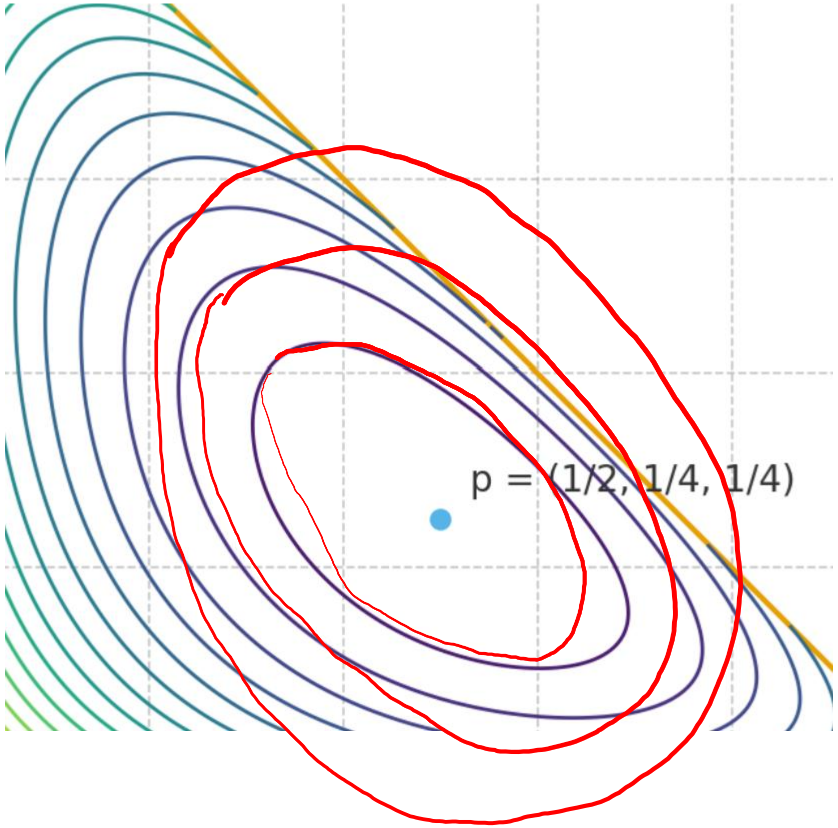
$$\approx (\theta - \theta_t)^\top \sum_a [\nabla_\theta \pi_\theta(a)]_{\theta=\theta_t} r_t(a)$$

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

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

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

$$= \theta_t + \eta F_{\theta_t}^{-1} g_t$$ **NPG**


p = (1/2, 1/4, 1/4)

# NPG vs. PG

**NPG**

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

(PPO)

$$\nabla_\theta \sum_a \pi_\theta(a)\, r_t(a)$$

**(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**

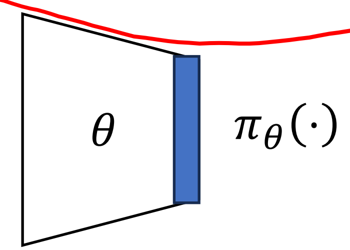
NPG

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

$\Updownarrow$

PPO

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

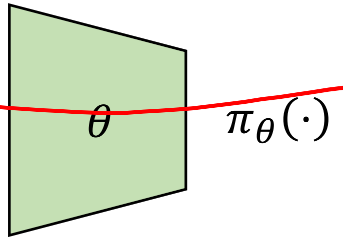



**PG**

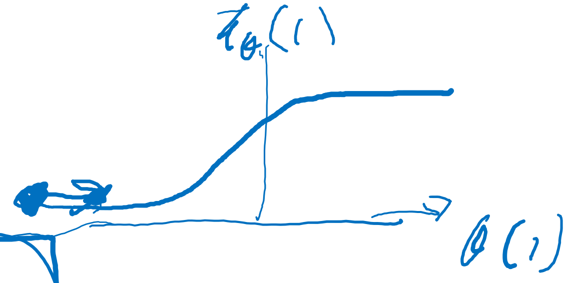
$$\theta_{t+1} = \theta_t + \eta \sum_a \nabla_\theta \pi_\theta(a)\, r_t(a) \Big|_{\theta=\theta_t}$$

$\Updownarrow$

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



# Example: NPG vs. PG with softmax policy




Consider multi-armed bandits with **softmax policy** $\boxed{\pi_\theta(a) = \frac{e^{\theta(a)}}{\sum_{a'} e^{\theta(a')}}}$

parameterized by $\theta(1), \theta(2), \dots, \theta(A)$

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

$r_t(a) - \boxed{\phantom{x}}$

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

For $t = 1,2, \dots$

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

**PG**

For $k = 1,2, \dots$

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

Check the equivalence (exercise)

$$\pi_{\theta_{t+1}}(a) \propto e^{\theta_{t+1}(a)} = e^{\theta_t(a)} e^{\eta r_t(a)} \propto \pi_{\theta_t}(a) e^{\eta r_t(a)}$$

NPG can also be written as

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

# NPG (EW) vs. PG

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

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

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

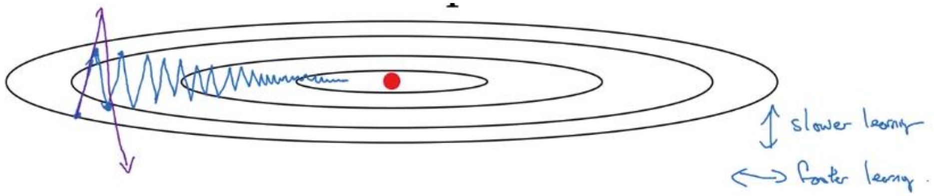
Initial policy $\pi = [0.0001, 0.9999]$

best 600

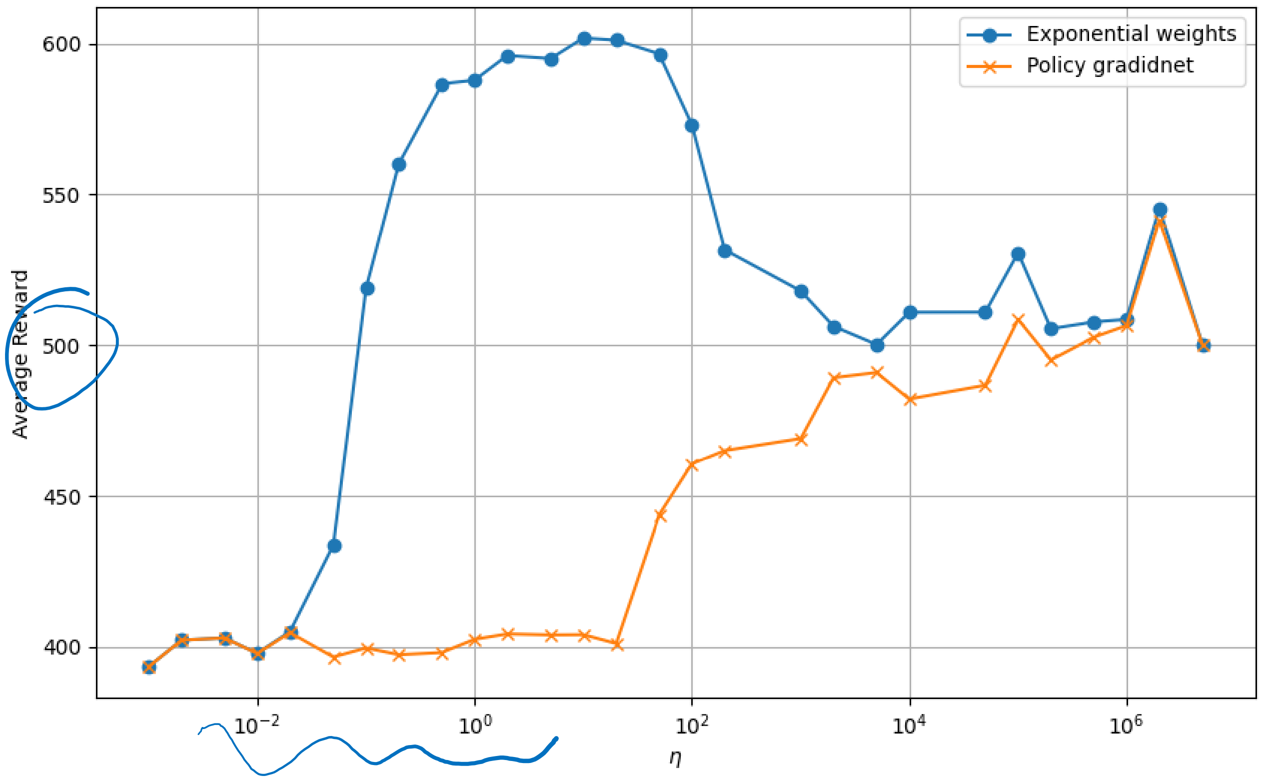
Plot total reward in 1000 rounds

worst 400

[code]




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



$\nabla_\theta \pi_{\theta_t}(a)\, \hat{r}_t(a)$

# NPG and PG with bandit feedback

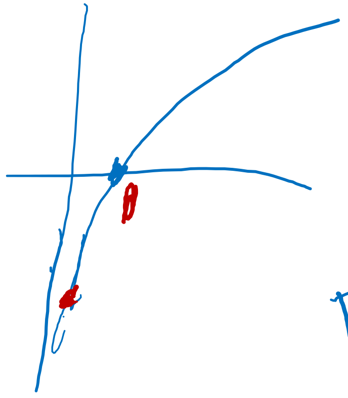
NPG

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

PG

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

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



$$g_t = \sum_a \nabla_\theta \pi_\theta(a)\, \hat{r}_t(a) = \sum_a \nabla_\theta \pi_\theta(a) \frac{r_t(a) - b_t}{\pi_{\theta_t}(a)} \mathbb{I}\{a_t = a\}$$

$$= \nabla_\theta \pi_\theta(a_t) \frac{r_t(a_t) - b_t}{\pi_{\theta_t}(a_t)}$$

$$= \nabla_\theta \log \pi_\theta(a_t)\, (r_t(a_t) - b_t)$$

$$\frac{\nabla_\theta f_\theta}{f_\theta} = \nabla_\theta \log f_\theta$$

# PG for contextual bandits

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

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

Or simply written as

$$\theta \leftarrow \theta + \eta \underbrace{\nabla_\theta \log \pi_\theta(a_t|x_t)}(r_t(x_t, a_t) - b_t(x_t))$$

Coming from inverse propensity weighting / importance weighting

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

$$g_t = \sum_a \nabla_\theta \pi_\theta(a)\left(\hat{r}_t(a) + c\right)$$

$$= \sum_a \nabla_\theta \pi_\theta(a)\, \hat{r}_t(a) + \underbrace{\sum_a \left(\nabla_\theta \pi_\theta(a)\right) \cdot c}_{\nabla_\theta \left(\underbrace{\sum_a \pi_\theta(a)}_{1}\right) \cdot c \;=\; 0}$$

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

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

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

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

# Summary: Policy-Based Algorithms in CB

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

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