

# **Full-Information Online Learning with Adversarial Reward**

Chen-Yu Wei

# The Expert Problem

Alternative protocol:

Environment **decides** the reward vector  $r_t$  (not revealing)

Learner chooses an expert  $a_t$

Environment reveals  $r_t$

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

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

Learner chooses a distribution over experts  $p_t \in \Delta_{\mathcal{A}}$

Environment **decides** and reveals the reward vector  $r_t = (r_t(1), \dots, r_t(A))$

**Adversarial environment:**  $r_1(a), \dots, r_T(a)$  do not have the same mean

$$\text{Regret} = \max_{a \in \mathcal{A}} \sum_{t=1}^T r_t(a) - \sum_{t=1}^T \langle p_t, r_t \rangle$$

# Strategies?

- Follow the leader

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

time	1	2	3	4	5	...
action 1	1/2	1	0	1	0	...
action 2	1	0	1	0	1	...

Learner total reward  $\leq 1$   
Total reward of best action  
 $\approx \frac{T}{2}$

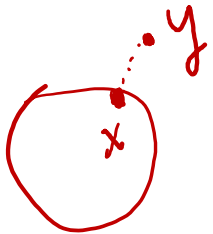
# Incremental Updates

**Projected gradient ascent:**

$$p_{t+1} = \Pi_{\Delta_{\mathcal{A}}}(p_t + \eta r_t)$$

**Exponential weight updates:**

$$p_{t+1}(a) = \frac{p_t(a) \exp(\eta r_t(a))}{\sum_{a' \in \mathcal{A}} p_t(a') \exp(\eta r_t(a'))}$$



action's expected reward =  $\langle p, r_t \rangle$   
(p)

$$\Pi_{\Delta}(y) = \operatorname{argmin}_{x \in \Delta} \|x - y\|_2$$

# Equivalent Forms of EWU

$$p_{t+1}(a) \propto p_t(a) \exp(\eta r_t(a)) \propto p_{t-1}(a) \exp(\eta r_{t-1}(a)) \exp(\eta r_t(a)) \dots$$

$$p_{t+1}(a) = \frac{p_t(a) \exp(\eta r_t(a))}{\sum_{a' \in \mathcal{A}} p_t(a') \exp(\eta r_t(a'))}$$

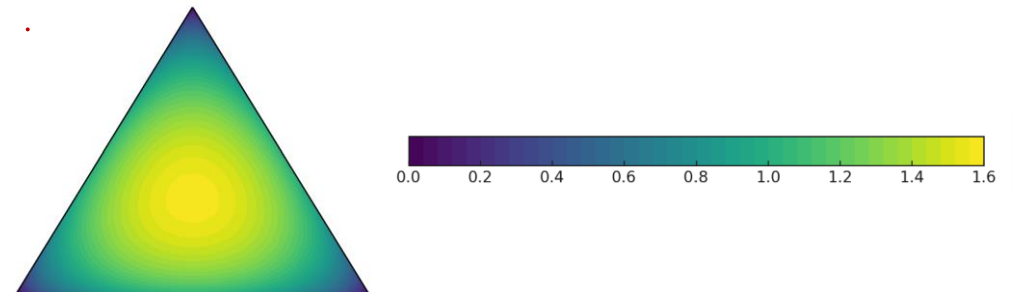
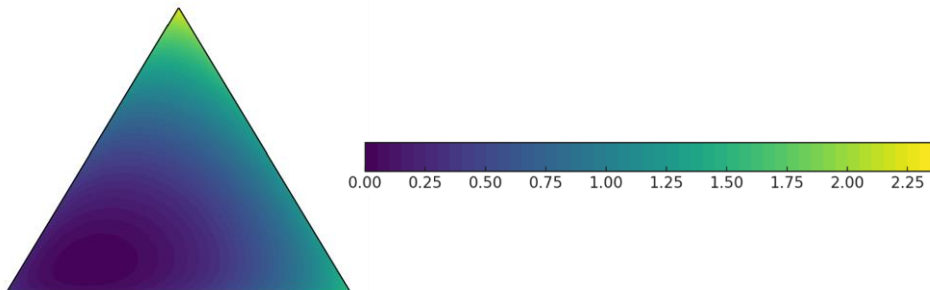
$$p_{t+1}(a) = \frac{\exp(\eta \sum_{i=1}^t r_i(a))}{\sum_{a' \in \mathcal{A}} \exp(\eta \sum_{i=1}^t r_i(a'))} \rightarrow \eta t \cdot \hat{r}_t(a)$$

$$p_{t+1} = \operatorname{argmax}_{p \in \Delta_{\mathcal{A}}} \left\{ \langle p, r_t \rangle - \frac{1}{\eta} \text{KL}(p, p_t) \right\}$$

$$\text{KL}(p, q) := \sum_{a=1}^A p(a) \ln \frac{p(a)}{q(a)} \quad (\text{KL divergence})$$

$$p_{t+1} = \operatorname{argmax}_{p \in \Delta_{\mathcal{A}}} \left\{ \left\langle p, \sum_{i=1}^t r_i \right\rangle + \frac{1}{\eta} H(p) \right\}$$

$$H(p) := \sum_{a=1}^A p(a) \ln \frac{1}{p(a)} \quad (\text{Shannon entropy})$$



# Regret Bound for Exponential Weight Updates

## Theorem.

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

$$|r_t(a)| \leq R_{\max}$$

$$p_{t+1}(a) = \frac{p_t(a) \exp(\eta r_t(a))}{\sum_{a' \in \mathcal{A}} p_t(a') \exp(\eta r_t(a'))}$$

ensures

$$\text{Regret} = \max_{a^*} \sum_{t=1}^T (r_t(a^*) - \langle p_t, r_t \rangle) \leq \frac{\ln A}{\eta} + \eta \sum_{t=1}^T \sum_{a=1}^A p_t(a) \underbrace{r_t(a)^2}_{\leq R_{\max}^2}$$

$$\begin{aligned} &\leq \frac{\ln A}{\eta} + \eta \sum_t \left( \sum_a p_t(a) \right) R_{\max}^2 \\ &= \frac{\ln A}{\eta} + \eta T \cdot R_{\max}^2 \end{aligned}$$

optimal  $\downarrow$

$$R_{\max} \sqrt{T \ln A}$$

# Regret Bound Analysis

$$p_1(a) = \frac{1}{A}$$

$$p_{t+1}(a) = \frac{p_t(a) \exp(\eta r_t(a))}{\sum_{a'} p_t(a') \exp(\eta r_t(a'))}$$

$$\Rightarrow \log \frac{p_{t+1}(a^*)}{p_t(a^*)} = \log \frac{\exp(\eta r_t(a^*))}{\sum_a p_t(a) \exp(\eta r_t(a))} = \eta r_t(a^*) - \log \left( \sum_a p_t(a) \exp(\eta r_t(a)) \right)$$

$$\Rightarrow \underline{r_t(a^*) - \langle p_t, r_t \rangle} = \frac{1}{\eta} \log \frac{p_{t+1}(a^*)}{p_t(a^*)} + \underbrace{\frac{1}{\eta} \log \left( \sum_a p_t(a) \exp(\eta r_t(a)) \right) - \langle p_t, r_t \rangle}_{\leq 1}$$

$$\Rightarrow \text{Regret} = \sum_{t=1}^T (r_t(a^*) - \langle p_t, r_t \rangle)$$

$$\begin{aligned} &\leq \frac{1}{\eta} \log \frac{p_{T+1}(a^*)}{p_1(a^*)} + \eta \sum_t \sum_a p_t(a) r_t(a)^2 \\ &\leq \frac{1}{\eta} \log A + \eta \sum_{t,a} p_t(a) r_t(a)^2 \end{aligned}$$

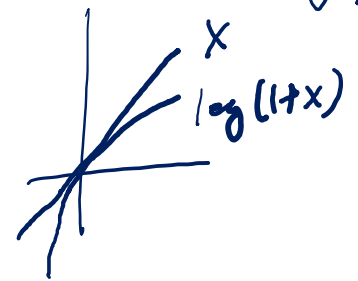
$$\frac{1}{\eta} \log \left( \sum_a P_t(a) \exp(\eta r_t(a)) \right) - \langle P_t, r_t \rangle$$

$$\leq \frac{1}{\eta} \log \left( \sum_a P_t(a) \left( 1 + \eta r_t(a) + \eta^2 r_t(a)^2 \right) \right) - \langle P_t, r_t \rangle \quad e^x \leq 1 + x + x^2 \text{ for } x \leq 1$$

$$= \frac{1}{\eta} \log \left( \underbrace{\sum_a P_t(a)}_1 + \eta \sum_a P_t(a) r_t(a) + \eta^2 \sum_a P_t(a) r_t(a)^2 \right) - \langle P_t, r_t \rangle$$

$$\leq \frac{1}{\eta} \left( \eta \underbrace{\sum_a P_t(a) r_t(a)}_{\langle P_t, r_t \rangle} + \eta^2 \sum_a P_t(a) r_t(a)^2 \right) - \langle P_t, r_t \rangle \quad \left[ \log(1+x) \leq x \right]$$

$$= \eta \sum_a P_t(a) r_t(a)^2$$





# Exponential Weight Updates

$$\Delta_{\mathcal{A}} = \left\{ x \in \mathbb{R}^A : \sum_a x(a) = 1, x(a) \geq 0 \right\}$$

Exponential Weight Updates = KL divergence regularized policy updates

$$p_{t+1}(a) = \frac{p_t(a) \exp(\eta r_t(a))}{\sum_{a' \in \mathcal{A}} p_t(a') \exp(\eta r_t(a'))}$$

$$p_{t+1} = \underset{p \in \Delta_{\mathcal{A}}}{\operatorname{argmax}} \left\{ \underbrace{\langle p, r_t \rangle}_{F(p)} - \underbrace{\frac{1}{\eta} \operatorname{KL}(p, p_t)} \right\}$$

$$\operatorname{KL}(p, p_t) = \sum_a p(a) \log \frac{p(a)}{p_t(a)}$$

$$\frac{\partial}{\partial p(a)} = \log \frac{p(a)}{p_t(a)} + 1$$

$$\mathcal{L}(p) = \underbrace{F(p)} + \lambda \left( \sum_a p(a) - 1 \right) + \sum_a \mu(a) p(a), \mu(a) \geq 0$$

$$\frac{\partial}{\partial p(a)} = r_t(a) - \frac{1}{\eta} \left( \log \frac{p(a)}{p_t(a)} + 1 \right) + \lambda + \mu(a) = 0$$

$\mu(a) p(a) = 0 \quad \forall a$

$$\Rightarrow p(a) = p_t(a) \exp \left( \eta r_t(a) + \eta \lambda + \cancel{\eta \mu(a)} \right)$$

KL divergence regularized policy updates is the basis of many RL algorithms (e.g., PPO, SAC).

# Projected Gradient Descent

Projected Gradient Descent = Euclidean norm regularized policy updates

$$p_{t+1} = \Pi_{\Delta_{\mathcal{A}}}(p_t + \eta r_t)$$

=

$$p_{t+1} = \operatorname{argmax}_{p \in \Delta_{\mathcal{A}}} \left\{ \langle p, r_t \rangle - \frac{1}{2\eta} \|p - p_t\|_2^2 \right\}$$

# Recap

- We considered **full-information, policy-based** learning algorithms for the expert problem (finite action decision-making problem).
- We showed that the **exponential weight algorithm** achieves sub-linear regret even when the reward is adversarial.
- Actually, **projected gradient descent** also works, though we haven't provided a proof (will do it today).
- Exponential weight updates = KL divergence regularized gradient updates
- Projected gradient descent = Euclidean norm regularized gradient updates

# Distance Regularized Updates

## Projected Gradient Descent

$$p_{t+1} = \Pi_{\Delta_{\mathcal{A}}}(p_t + \eta r_t)$$

$$p_{t+1} = \max_{p \in \Delta_{\mathcal{A}}} \left\{ \langle p, r_t \rangle - \frac{1}{2\eta} \|p - p_t\|_2^2 \right\}$$

## Exponential Weight Updates

$$p_{t+1}(a) \propto p_t(a) \exp(\eta r_t(a))$$

$$p_{t+1} = \max_{p \in \Delta_{\mathcal{A}}} \left\{ \langle p, r_t \rangle - \frac{1}{\eta} \text{KL}(p, p_t) \right\}$$


- Adversarial reward
- Stochastic reward
- For non-linear functions, gradient only approximates the function locally

# General Framework: Mirror Descent

## Projected Gradient Descent

$$p_{t+1} = \Pi_{\Delta_{\mathcal{A}}}(p_t + \eta r_t)$$


$$p_{t+1} = \max_{p \in \Delta_{\mathcal{A}}} \left\{ \langle p, r_t \rangle - \frac{1}{2\eta} \|p - p_t\|_2^2 \right\}$$

$$\psi(p) = \frac{1}{2} \|p\|_2^2$$


## Exponential Weight Updates

$$p_{t+1}(a) \propto p_t(a) \exp(\eta r_t(a))$$

$$p_{t+1} = \max_{p \in \Delta_{\mathcal{A}}} \left\{ \langle p, r_t \rangle - \frac{1}{\eta} \text{KL}(p, p_t) \right\}$$

$$\psi(p) = \sum_{a=1}^A p(a) \ln p(a)$$


### Mirror Descent

$$p_{t+1} = \max_{p \in \Omega} \left\{ \langle p, r_t \rangle - \frac{1}{\eta} D_{\psi}(p, p_t) \right\}$$

$$D_{\psi}(p, q) := \psi(p) - \psi(q) - \langle \nabla \psi(q), p - q \rangle$$

(Bregman divergence w.r.t. the potential function / regularizer  $\psi$ )

# Online Linear Optimization and Projected Gradient Descent

**Given:** Convex feasible set  $\Omega \subseteq \mathbb{R}^d$

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

Learner chooses a point  $w_t \in \Omega$

Environment reveals a reward vector  $r_t \in \mathbb{R}^d$

$$\text{Regret} = \max_{w \in \Omega} \sum_{t=1}^T \langle w, r_t \rangle - \sum_{t=1}^T \langle w_t, r_t \rangle$$

## Projected Gradient Descent

Arbitrary  $w_1 \in \Omega$

$$w_{t+1} = \Pi_{\Omega}(w_t + \eta r_t)$$

# Regret Bound of Projected Gradient Descent

**Theorem.** Projected Gradient Descent ensures

$$\sum_{t=1}^T \langle w^*, r_t \rangle - \sum_{t=1}^T \langle w_t, r_t \rangle \leq \frac{\|w^* - w_1\|_2^2}{2\eta} + \frac{\eta}{2} \sum_{t=1}^T \|r_t\|_2^2 \leq \frac{D^2}{2\eta} + \frac{\eta G^2 T}{2}$$

# Summary

## Projected Gradient Descent

$$w_{t+1} = \Pi_{\Omega}(w_t + \eta r_t)$$

$$w_{t+1} = \max_{w \in \Omega} \left\{ \langle w, r_t \rangle - \frac{1}{2\eta} \|w - w_t\|_2^2 \right\}$$

$$\begin{aligned} \text{Regret} &\leq O(DG\sqrt{T}) \\ &= O(\sqrt{AT}) \quad (\text{in the expert setting}) \end{aligned}$$

## Exponential Weight Updates

$$p_{t+1}(a) \propto p_t(a) \exp(\eta r_t(a))$$

$$p_{t+1} = \max_{p \in \Delta_{\mathcal{A}}} \left\{ \langle p, r_t \rangle - \frac{1}{\eta} \text{KL}(p, p_t) \right\}$$

$$\text{Regret} \leq O(\sqrt{T \log A})$$