



Sequential Decision

Session 5: Exponential weights and off-policy evaluation

Igor Colin

October 2026

ESIEE Paris

Today

Part	Content	Time
Lecture	Hedge, EXP3, evaluation of a policy from logs	1 h
Tutorial 5	exercises 1 and 2: Hedge by hand, importance sampling	40 min
Lab 5	Hedge, EXP3, evaluation of recommendation policies from logs	1 h 20

So far, the rewards of each arm had a fixed distribution, and the agent learned from its own choices. Today, two assumptions are dropped:

1. **no model of the rewards**: they can be any sequence. The answer is a randomized policy, a softmax;
2. **learning from the choices of another policy**: logs. The answer is importance weighting.

Both tools come back in policy gradient and RLHF (sessions 8 and 9).

Choosing a route

Choosing a route every morning

Morning	metro	bus	bike
sunny	30	38	**28**
rainy	**32**	50	55
strike	60	45	**35**

typical travel times in minutes

Each morning you choose a route, after seeing the weather and the news: a **context** u_t.

- A **policy** maps a context to a route: bike if sunny or strike, metro if rainy
- Regret against the best route for each context: always the bike costs 23 minutes every rainy day
- Times random with a fixed distribution per context: a **contextual bandit**. Many contexts: features $x_{t,a}$ that depend on u_t, as in the track `users`, and LinUCB

Recommendations that change the traffic



Now you are Waze, and you advise thousands of drivers.

- Advise road A, the fastest, to everybody: road A jams
- The times react to your own choices: no fixed distribution, and the guarantees of UCB or Thompson sampling are lost
- Strikes, works and accidents have no reliable model either
- Goal: guarantees that hold for every sequence of travel times. To start, forget the context: one choice per day

Without a model of the rewards

Full information

	metro	bus	bike
Monday	30	40	35
Tuesday (strike)	60	45	35
Wednesday (rain)	32	50	55
Thursday	30	38	33
Total	**152**	173	158

travel times in minutes

Each morning you take a route; in the evening, a traffic app gives the time of every route.

- **Experts**: the $K = 3$ routes
- **Gain**: $g_t(a) = 1 - \text{time}/60 \in [0, 1]$
- **Full information**: $g_t(a)$ known for every route
- **No model**: any sequence of times is possible
- **Benchmark**: the best route in hindsight (metro)

Prediction with expert advice

Protocol

K experts. At each round $t = 1, \dots, T$:

1. the agent chooses a probability vector p_t over the experts;
2. the gains $g_t \in [0,1]^K$ of all the experts are revealed;
3. the agent receives $\langle p_t, g_t \rangle = \sum_a p_t(a)\, g_t(a)$.

Regret against the best expert

$$R_T = \max_a G_T(a) - \sum_{t=1}^{T} \langle p_t, g_t \rangle \ ,$$

where $G_t(a) = \sum_{s \le t} g_s(a)$ is the cumulative gain of expert a.

In the example

- p_t: how likely you are to take each route on day t
- $G_t(a)$: total gain of route a up to day t
- No assumption on the sequence $g_1, g_2, \dots$: the bound must hold for *every* sequence

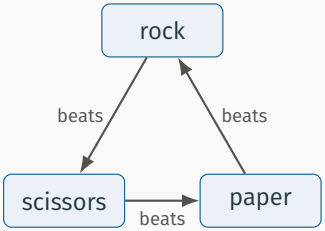
Follow the leader

Follow the leader (FTL): take the route with the largest total gain so far (the metro in case of a tie). Two routes; the red frame is the route taken by FTL:

	day 1	day 2	day 3	day 4	day 5	day 6
totals so far	0 vs 0	0.5 vs 0	0.5 vs 1	1.5 vs 1	1.5 vs 2	2.5 vs 2
gain of metro	0.5	0	1	0	1	0
gain of bus	0	1	0	1	0	1
FTL takes	metro	metro	bus	metro	bus	metro
FTL gains	0.5	0	0	0	0	0

- FTL always takes the route that has just won, and that route loses the next day: after day 1, FTL gains nothing
- Each route gains about $T/2$: $R_T \approx T/2$, linear (tutorial, exercise 1)

Randomizing is necessary



- Against an opponent who knows your rule, **any deterministic rule loses every round**: if your rule says rock, they play paper
- Playing uniformly at random wins, draws and loses with probability $1/3$ each, whatever the opponent does
- FTL is a deterministic rule, and the alternating sequence is the paper against its rock

Any deterministic strategy has a linear regret on some sequence

With 2 experts, give gain 0 to the expert the strategy is about to choose and 1 to the other: the strategy gains 0, one expert gains at least $T/2$, so $R_T \geq T/2$. Same for Waze: a deterministic advice sends every driver to the same road.

Hedge

Hedge (exponential weights)

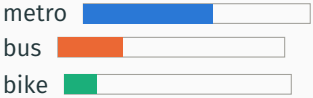
With a learning rate $\eta > 0$,

$$p_t(a) = \frac{e^{\eta G_{t-1}(a)}}{\sum_b e^{\eta G_{t-1}(b)}} \ .$$

- Softmax of the cumulative gains, temperature $1/\eta$; $\eta \to \infty$: FTL, $\eta \to 0$: uniform
- Randomized, as required: a route behind by g has $e^{\eta g}$ times less weight, but is never abandoned

Tutorial, exercise 1: three routes, $\eta = \log 2$, so the weight of route a is $2^{G_{t-1}(a)}$; gains $(1, 0, 0)$, then $(1, 1, 0)$.

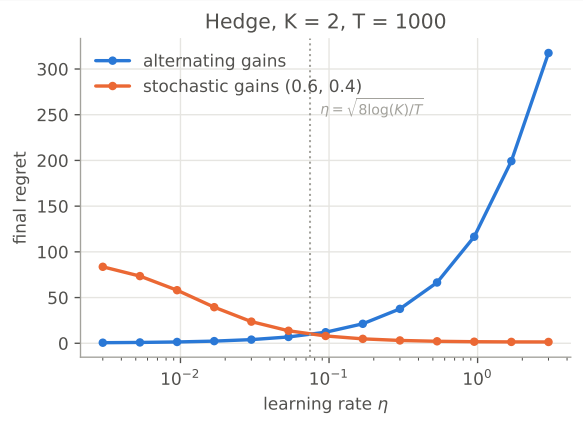
$t = 3$, $G_2 = (2, 1, 0)$



Theorem

For any sequence of gains in $[0, 1]$, $R_T \leq \frac{\log K}{\eta} + \frac{\eta T}{8}$, so $R_T \leq \sqrt{T \log(K)/2}$ for $\eta = \sqrt{8 \log(K)/T}$.

The learning rate



- Alternating gains: a small η is best, a large one behaves as FTL
- Stochastic gains: the leader is the best expert, and a large η is best
- $\eta = \sqrt{8\log(K)/T}$ is a compromise: never the best on a given sequence, never linear
- The bound depends on $\log K$ only: thousands of experts are affordable

Bandit feedback

Without the traffic app

	metro	bus	bike
Monday	**30**	?	?
Tuesday	?	?	**35**
Wednesday	**32**	?	?
Thursday	?	**38**	?

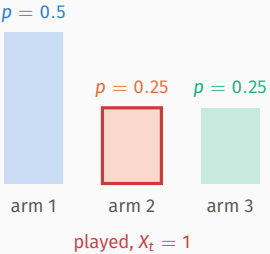
only the route taken is observed

Now you only know the time of the route you took: this is **bandit feedback**, as in sessions 1 to 4, but still without a model of the times.

- Hedge needs the gains of all the routes to update its weights
- The gains of the other routes must be estimated, from a single observation per day
- The choice is random, with known probabilities p_t: this is what makes the estimation possible

Estimating the gains that are not observed

probabilities p_t



estimates $\hat{x}_t = (0,\ 4,\ 0)$

With bandit feedback, only the reward of the arm played is observed.

Importance weighting

$$\hat{x}_t(a) = \frac{X_t \mathbf{1}\{A_t = a\}}{p_t(a)}$$

- Arm 2 is played one time out of four: its reward is counted four times
- **Unbiased**: $\mathbb{E}[\hat{x}_t(a)] = x_t(a)$ for every arm, played or not (tutorial, exercise 2)
- **Variance** $x_t(a)^2(1 - p_t(a))/p_t(a)$: large for rarely played arms

EXP3

EXP3 (Auer et al., 2002)

At each round:

1. $p_t(a) \propto \exp\big(-\eta \hat{L}_{t-1}(a)\big)$;
2. draw $A_t \sim p_t$ and observe X_t;
3. $\hat{L}_t(A_t) = \hat{L}_{t-1}(A_t) + \dfrac{1 - X_t}{p_t(A_t)}$.

Hedge on the estimated losses $1 - x_t(a)$.

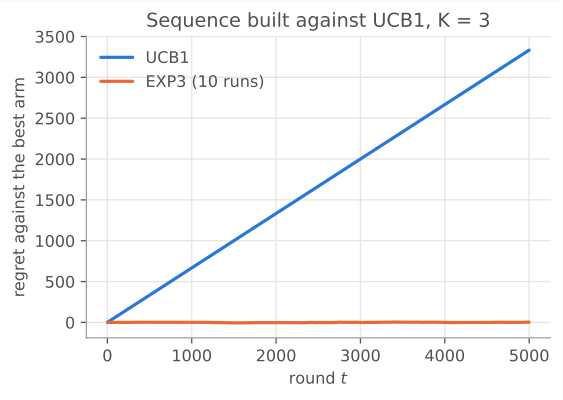
Theorem

For any sequence of rewards in $[0, 1]$, with $\eta = \sqrt{2\log(K)/(TK)}$,

$$R_T \leq \sqrt{2TK \log K} \ .$$

- An extra $\sqrt{K}$ compared with Hedge: the price of observing one reward per round
- Losses rather than gains: the estimates stay nonnegative, which keeps the analysis simple

Deterministic strategies can be trapped



UCB1 is deterministic: its next arm is a function of the past.

1. Simulate UCB1, giving reward 0 to the arm it chooses and 1 to the others
2. Replay this fixed sequence

- UCB1 receives 0 at every round; each arm pays about $2T/3$
- EXP3 cannot be anticipated: regret close to 0
- The argument of the rock-paper-scissors slide, applied to UCB1

Stochastic or adversarial

	UCB, Thompson sampling	EXP3
Stochastic rewards	$O(\log T)$, instance-dependent	$O(\sqrt{TK \log K})$
Arbitrary sequence	can be linear	$O(\sqrt{TK \log K})$
Needs T	no	yes, for η (or a decreasing η_t)
Choice of the arm	argmax (TS: argmax of a draw)	draw from a softmax

- Lab: on $(0.6, 0.5, 0.4)$, close at $T = 5000$; at $T = 20000$, UCB1 ≈ 180, EXP3 ≈ 230
- Rewards that change over time are in between: neither assumption holds (project 1)
- Softmax policies and importance ratios come back in policy gradient and RLHF (sessions 8 and 9)

Learning from logs

Learning from logs

Day	Weather	Advice	Time	Probability
1	sunny	bike	27	0.8
2	rainy	metro	33	0.9
3	sunny	metro	31	0.2
4	rainy	bike	58	0.1
5	sunny	bike	29	0.8

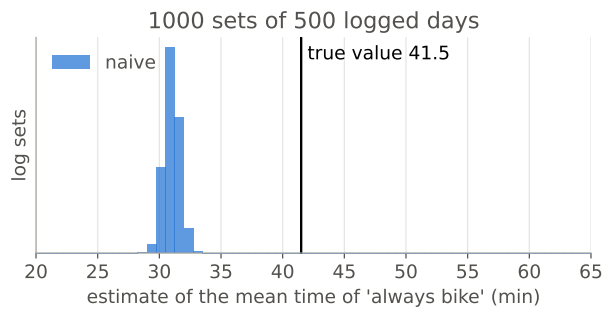
Logs of Waze. Its policy μ advises the bike with probability 0.8 on sunny days and 0.1 on rainy days.

Waze records the context, the advice, the reward, and the **propensity**: the probability with which it gave this advice.

- Question: what would the policy π = *always the bike* give, on average, without deploying it?
- True answer, unknown to Waze: half of the days are rainy, so $\frac{1}{2} \times 28 + \frac{1}{2} \times 55 = 41.5$ minutes
- Lab: the same question for a news recommender

The naive estimator

Naive estimator: keep the days where the advice is the one π would have given (the bike), and average their times. Table: days 1, 4, 5, so $(27 + 58 + 29)/3 = 38$.



- About 31 minutes over many days, far from 41.5: Waze advises the bike mostly on sunny days, so 89% of the kept days are sunny. They are not representative

Importance sampling

IPS estimator (inverse propensity scoring)

$$\hat{V}_{\mathrm{IPS}}(\pi) = \frac{1}{n} \sum_{i=1}^{n} \underbrace{\frac{\pi(a_i \mid u_i)}{\mu(a_i \mid u_i)}}_{\text{weight } w_i} r_i$$

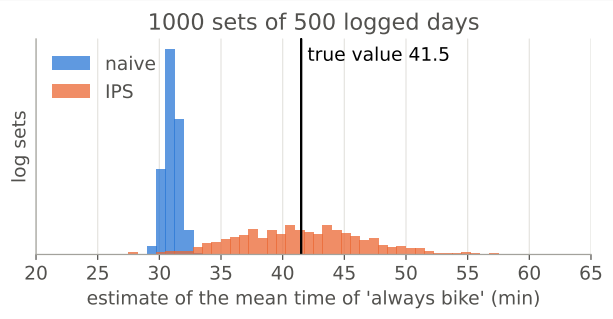
In the example, $\pi(\text{bike}) = 1$:

- metro days: weight 0
- sunny bike days: weight $1/0.8 = 1.25$
- rainy bike days: weight $1/0.1 = 10$

A rainy bike day stands for 10 rainy days. On average:
$\frac{1}{2} \cdot 0.8 \cdot 1.25 \cdot 28 + \frac{1}{2} \cdot 0.1 \cdot 10 \cdot 55 = 41.5$.

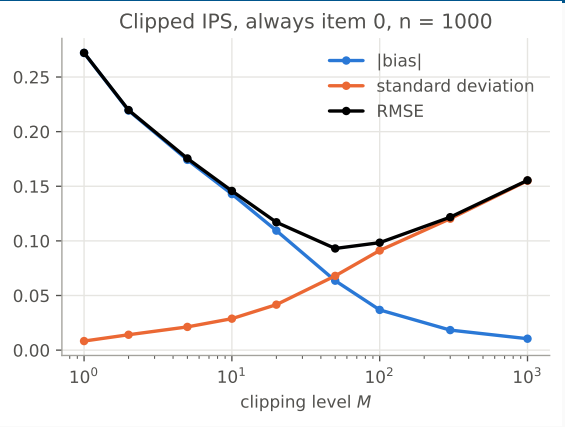
- **Unbiased** if $\mu(a \mid u) > 0$ whenever $\pi(a \mid u) > 0$ (tutorial, exercise 2)
- Also for a random π
- The weighting of EXP3, with a ratio of two policies
- Needs the logged propensities

Naive and IPS



- **Naive**: precise but wrong. **IPS**: right on average, but spread (standard deviation of 5 minutes with 500 days)
- The spread comes from the rainy bike days: about 25 out of 500, with weight 10. On the 5 days of the table, IPS gives 130 minutes
- Large variance when π plays what μ rarely plays (tutorial, exercise 3)

Clipping



Lab: news recommender, a policy that μ follows in 13% of the logs; each point summarizes 200 sets of 1000 logs

Replace each weight w_i by $\min(w_i, M)$.

- Small M: low variance, but biased, since the rare logs count less than they should
- Large M: IPS, unbiased but noisy
- RMSE, the typical error of the estimate: best in between, 0.09 at $M = 50$ against 0.21 for IPS
- PPO (session 8) clips the same ratio $\pi_{\text{new}}/\pi_{\text{old}}$, for the same reason

Summary

Setting	Method	Key point
No model, full information	Hedge	randomize: softmax of the gains
No model, bandit feedback	EXP3	estimate the unseen gains by X/p
Logs of another policy	IPS, clipped IPS	reweight by π/μ

Tutorial 5

- Hedge and FTL by hand
- Importance sampling is unbiased
- Variance and clipping

Lab 5

- Hedge against FTL, learning rate
- EXP3 against UCB1
- Off-policy evaluation of recommenders

Reading

- Lattimore and Szepesvári, *Bandit Algorithms*, chapters 11 and 18
- Li et al., WSDM 2011 (replay)