Online Learning under Full and Bandit Information

Artem Sokolov
Computerlinguistik
Universität Heidelberg

Real world online learning tasks

- advertising (which ad to display)
- medical treatment (which drug to prescribe)
- design/functionality rollouts (works or not)
- spam/malware filtering (filter or keep)
- stock market (sell or acquire bonds)
- network routing (which path to take)
- compression, weather, etc.

in every task there is a decision to be made under missing information

Goals

1. introduce online learning
2. explain distinction between full and partial information tasks
3. introduce the notion of regret
4. present basic algorithms for those cases

Batch learning

many i.i.d examples $\mathcal{D} = \{x_i, y_i\}_{i=1}^N$

define some loss $\ell(\mathcal{D})$ (e.g. negative log-likelihood, square error)

learn a model by $\ell(\mathcal{D}) \to \min$

deploy on a test set

Online learning

one example x_t

predict $\hat{y}_t$

get feedback

suffer some penalty $\ell_t(x_t, \hat{y}_t)$

improve the model

repeat

- note: no training/testing set distinction

input $x_t \in \mathcal{X}$ input space

truth $y_t \in \mathcal{Y}$ truth space

prediction $\hat{y}_t \in \mathcal{P}$ decision space

	$\mathcal{X}$	$\mathcal{Y}$	$\mathcal{P}$	penalty/loss
online regression	$\mathbb{R}^d$	$\mathbb{R}$	$\mathbb{R}$	$\lvert y_t - \hat{y}_t \rvert$
online classification	$\mathbb{R}^d$	$\{1, \dots, K\}$	$\{1, \dots, K\}$	$[\![y_t \neq \hat{y}_t]\!]$
expert advice	$\mathbb{R}^N$	$\mathbb{R}^d$	$\{1, \dots, N\}$	$y_t[\hat{y}_t]$
structured prediction	K^m	K^m	K^m	$\sum_{i=1}^m [\![y_t^i \neq \hat{y}_t^i]\!]$

- **early days 50-70s:** online learning is a requirement
  - ➡ first computers, very low memory, very slow CPUs
  - ➡ perceptron from 1957 is originally an online algorithm!
- **later 70-90s:** batch learning became possible
  - ➡ reasonable CPU power, reasonable memory
  - ➡ great convergence guarantees!
- **2000s-now:**
  - ➡ computers are very powerful, memory is cheap ☺
  - ➡ batch algorithms explode memory and time ☹
    - easy access to data made datasets practically infinite
    - discarding data is a bad idea, we want it all!
    - some people say that "data acquisition outpaced the Moore's law"

effectively are back into the 50s

not only a question of resources:

- the larger the data, the harder it is to guarantee stationarity
  - ➡ cannot to be cramped into a fixed size dataset
- hence algorithms need to be adaptive
- and frequent re-training is not an option (because resources...)

Online learning

one example x_t

predict $\hat{y}_t$

get feedback

suffer some penalty $\ell_t(x_t, \hat{y}_t)$

improve the model

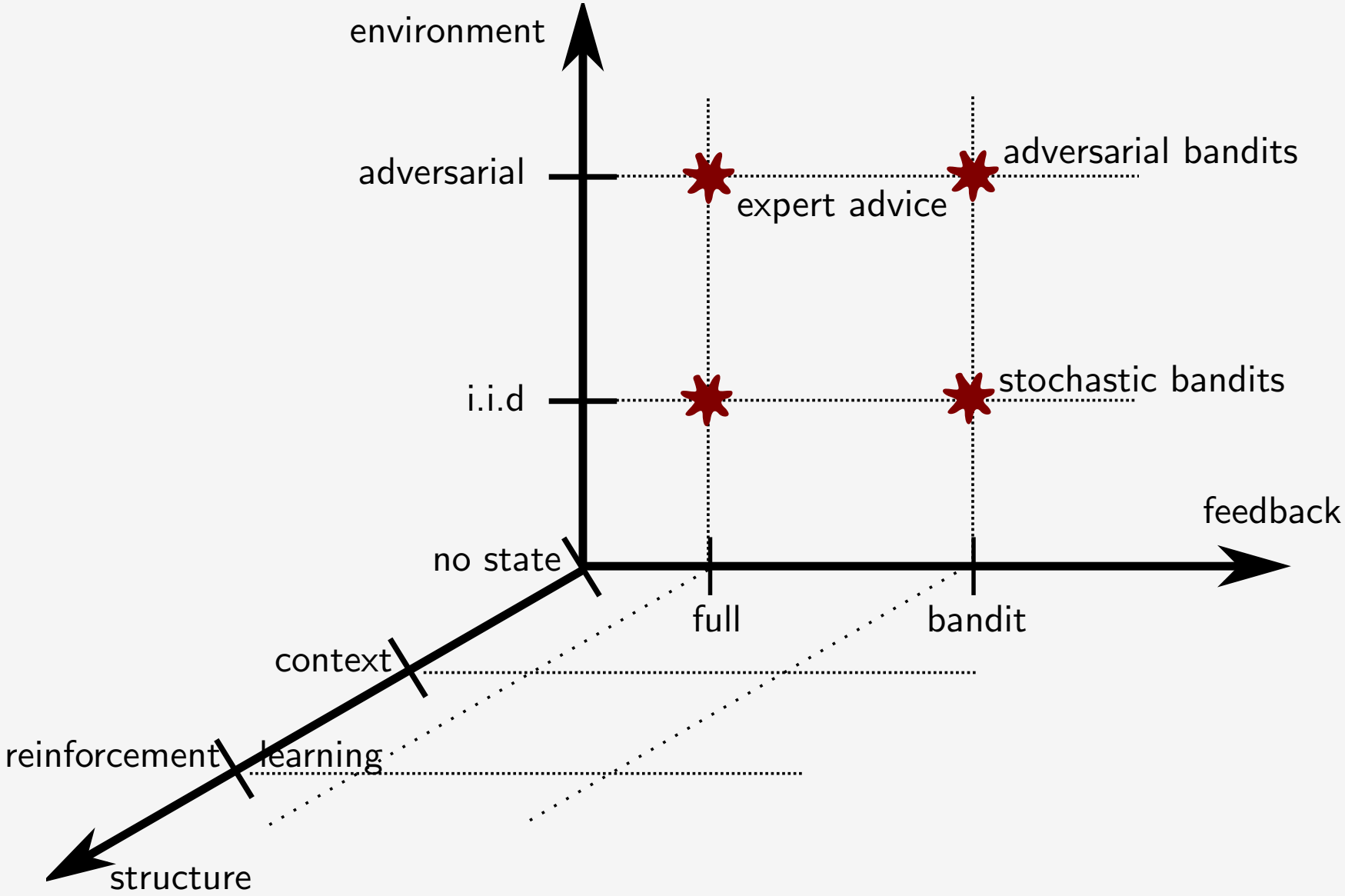
repeat

- small memory footprint
- faster updates
- faster adaptation
- better test performance (in certain sense)

Domain of online learning

- environment
  - ➡ i.i.d assumption is convenient
  - ➡ often cannot be guaranteed or is obviously violated
  - ➡ sometimes we assume nothing about distribution: 'adversarial case'
- feedback
  - ➡ full information is best
  - ➡ but correct labels are expensive and slow to get
  - ➡ often partial feedback is all you have: 'bandit case'
- structure
  - ➡ no state (important but rare case)
  - ➡ usually there is some state or context
  - ➡ structured spaces (actions change the environment)
- resources
  - ➡ batch learning is costly
  - ➡ saving everything is impractical
  - ➡ learn from one x and discard

The space of online learning algorithms



[Seldin'15]

Adversarial Environment with Full Information

(just means there are no statistical assumptions)

Online learning protocol

1: **for** $t = 0, \ldots$ **do**
2: observe x_t (if available)
3: predict $\hat{y}_t$
4: suffer loss $\ell_t(\hat{y}_t)$
5: update

- ℓ_t are arbitrary (e.g., does not mean there are uniformly distributed)
- could be random or non-random, depend on previous history
- we want algorithms that work in any case

What about the goal?

- no training set, so cannot minimize loss over training set
- even if we could, does not always make sense as ℓ_t can be anything
- ➡ **measure of success has to be calculated w.r.t. to the whole interaction, not just some end objective**

What do we want to achieve?

- in principle we want to minimize our total loss
- still not ideal, because ℓ_t can scale arbitrary
- so we need a **relative** measure
  - ➡ e.g., with respect to some fixed (but unknown) arm-pulling strategy $h = h_t$
  - ➡ or with respect to the best arm-pulling strategy from a set $\mathcal{H}$
  - ➡ note: the larger is $\mathcal{H}$ the harder it is the task
- we measure a 'cost of ignorance' or 'regret for not following that strategy'

$$R_T = \sum_{t=1}^{T} \ell_t(\hat{y}_t) - \sum_{t=1}^{T} \ell_t(h_t)$$

$$R_T = \sum_{t=1}^{T} \ell_t(\hat{y}_t) - \min_{h \in \mathcal{H}} \sum_{t=1}^{T} \ell_t(h)$$

Our ultimate goal:

- average regret $R_T/T \to 0$
- as fast as possible
- as the learning goes on, our loss is less and less different from the alternative one

Why do we need different definitions of regret?

- on the one hand, it's a tool to analyze a problem, to test it under different assumptions
- on the other, w/o any restrictions online learning is too hard (or impossible)
- need to restrict the power of adversary and vary R_T accordingly
- different definitions than reflect our knowledge about the environment:
  1. if we believe that true data is generated by some fixed function h^*, $y_t = h^*(x_t)$, it's reasonable to minimize R_T w.r.t. to that function

$$R_T(h^*) = \sum_{t=1}^{T} \ell_t(w_t) - \sum_{t=1}^{T} \ell_t(h^*)$$

  2. if not, the adversary must not at least change his mind at will, i.e. has to commit to some y_t before seeing $\hat{y}_t$; then it makes sense to optimize R_T w.r.t. to the best function from some set $\mathcal{H}$:

$$R_T(\mathcal{H}) = \sum_{t=1}^{T} \ell_t(w_t) - \min_{h \in \mathcal{H}} \sum_{t=1}^{T} \ell_t(h)$$

What happens if we don't have the commitment requirement?

Example: online classification

1: **for** $t = 0, \ldots$ **do**
2: observe x_t
3: predict $\hat{y}_t \in \{0, 1\}$
4: receive true y_t
5: suffer loss $\ell_t(\hat{y}_t) = |y_t - \hat{y}_t|$
6: update w_{t+1}

$$R_T = \sum_{t=1}^{T} \ell(\hat{y}_t) - \min_{h \in \mathcal{H}} \sum_{t=1}^{T} \ell_t(h(x_t)) R_T = \sum_{t=1}^{T} |y_t - \hat{y}_t| - \min_{h \in \mathcal{H}} \sum_{t=1}^{T} |y_t - h(x_t)|$$

- take simplest $\mathcal{H} = \{h_0, h_1\}$, where $h_a \equiv a$ (constant function)
- **Exercise 1:** can you make the learner always lose? [Shalev-Shwartz'12]
- wait until $\hat{y}_t$ and set $y_t = 1 - \hat{y}_t$

Realizability assumption: $\exists h^* \in \mathcal{H}$ s.t. $\forall t\ y_t = h^*(x_t)$. Also $|\mathcal{H}| < \infty$

Consistent

1: Initialize $V_0 = \mathcal{H}$
2: **for** $t = 0, \ldots$ **do**
3: observe x_t
4: choose **any** $h \in V_t$
5: predict $\hat{y}_t = h(x_t)$
6: receive true $y_t = h^*(x_t)$
7: update $V_{t+1} = \{h \in V_t : h(x_t) = y_t\}$

Analysis:

- $\forall t$ at least one h is removed if there was an error (and none if not)
- $1 \leq |V_t| \leq |\mathcal{H}| - \#\text{errors}$
- $R_T = \#\text{errors} - 0 = \#\text{errors} \leq |\mathcal{H}| - 1$
- can we do better? hint: purge hypotheses faster

Realizability assumption: $\exists h^* \in \mathcal{H}$ s.t. $\forall t\ y_t = h^*(x_t)$. Also $|\mathcal{H}| < \infty$

Halving

1: Initialize $V_0 = \mathcal{H}$
2: **for** $t = 0, \ldots$ **do**
3: observe x_t
4: choose **by majority vote** $h = \arg\max_{r \in \{0,1\}} |h \in V_t : h(x_t) = r|$
5: predict $\hat{y}_t = h(x_t)$
6: receive true $y_t = h^*(x_t)$
7: update $V_{t+1} = \{h \in V_t : h(x_t) = y_t\}$

Analysis:

- $\forall t$ at least one half of V_t is removed if there was an error
- $1 \leq |V_t| \leq |\mathcal{H}|/2^{\#\text{errors}}$
- $R_T(h^*) = \#\text{errors} \leq \log_2 |\mathcal{H}|$

Finiteness of $\mathcal{H}$ is crucial

Example

- real line $\mathcal{X} = (0, 1)$, thresholds $\mathcal{H} = \{h_\theta : (0, 1) \to \{0, 1\}\}$
- $h_\theta(x) = sign(\theta - x)$
- $\exists$ a sequence of x_t, y_t generated by some θ on which the Halving will have $R_T = T$

Exercise 2: construct such a sequence [Shalev-Shwartz'12] **Solution:**

- maintain L_t (left) and R_t (right)
- $L_0 = 0$, $R_0 = 1$
- pick a random $x_t \in (L_t, R_t)$
- receive $\hat{y}_t$
- report $y_t = 1 - \hat{y}_t$
- $R_{t+1} = x_t y_t + R_t \hat{y}_t$
- $L_{t+1} = x_t \hat{y}_t + L_t y_t$
- $\forall t \; R_t - L_t > 0$

- realizability assumption may be too harsh for our application
- add an element of surprise to our predictions!
  - ➡ remember to require the adversary to commit to y_t before seeing $\hat{y}_t$
  - ➡ will change lines 4 and 5 in the Consistent

Randomized

1: Initialize $V_0 = \mathcal{H}$
2: **for** $t = 0, \ldots$ **do**
3: observe x_t
4: choose probability p_t
5: predict $\hat{y}_t(w_t) = 1$ with prob. p_t
6: receive true $y_t = h^*(x_t)$
7: update $V_{t+1} = \{h \in V_t : h(x_t) = y_t\}$

$$R_T = \sum_{t=1}^{T} \mathbb{E}_{p_t}[\![\hat{y}_t \neq y_t]\!] - \min_{h \in \mathcal{H}} \sum_{t=1}^{T} [\![h(x_t) \neq y_t]\!] \qquad \leftarrow \text{note regret changed again}$$

$$= \sum_{t=1}^{T} |p_t - y_t| - \min_{h \in \mathcal{H}} \sum_{t=1}^{T} |h(x_t) - y_t| \leq \sqrt{2T \ln |\mathcal{H}|} \qquad \text{proof later}$$

- so far we had full information (i.e., we received the true y_t)
- different adversary restrictions help to get regret bounds
  - ➡ realizability + finiteness
    - Consistent $R_T \leq |\mathcal{H}| - 1$
    - Halving $R_T \leq \log_2 |\mathcal{H}|$
  - ➡ commitment
    - Randomized $R_T \leq \sqrt{2T \ln |\mathcal{H}|}$

Learning with Experts' Advice

- imagine horse-races
- you know nothing about horses ☹
- luckily you have knowledgeable friends willing to give you advices ☺
- apportion a fixed sum of money between them
- ➡ goal: minimize losses / maximize profit

I actually make a lot more money as a bookmaker than I ever did as a race horse...

- stateless case
  (you have friend's identity, but not horses' breakfast menu or expert history)
- N friends
- loss vector $\ell_t \in [0,1]^N$ e.g., $\ell_t[i] = 0.3$ if ith friend lost 30 cents
- prediction $p_t \in [0,1]^N$, $\sum_{i=1}^N p_t[i] = 1$ your distribution of money
- loss $\sum_{i=1}^N p_t[i]\ell_t[i] = \langle p_t, \ell_t \rangle$
- goal

$$R_T = \sum_{t=1}^{T} \langle p_t, \ell_t \rangle - \underbrace{\min_{i=1,\dots,N} \sum_{t=1}^{T} \ell_t[i]}_{\text{loss of the best friend}} \to \min$$

- note: you don't know how good your friends are
- note: horses/friends can conspire against you
- but in the limit you can do as good as the best friend in hindsight!
- (in terms of average loss per race)

- if one of the friends is perfect can get $\leq \log_2 N$ mistakes with Halving
- but making a mistake does not necessarily mean we should disqualify a friend

Hedge

1: init vector $w_1 \in \mathbb{R}_+^N$ s.t. $\sum_{i=1}^N w_1[i] = 1$, learning rate $\mu > 0$
2: **for** $t = 1, \ldots$ **do**
3: compute $p_t = \frac{w_t}{\sum_{i=1}^N w_t[i]}$
4: receive loss ℓ_t
5: update $w_{t+1}[i] = w_t[i] e^{-\mu \ell_t[i]}$ $\leftarrow$ "soft disqualification"

Theorem

For any $\ell^1, \ldots, \ell^T$ and any $i \in \{1, \ldots, N\}$

$$R_T = \sum_{t=1}^T \langle p_t, \ell_t \rangle - \min_j \sum_{t=1}^T \ell_t[j] \leq \sqrt{2T \ln N} + \ln N$$

We will upper- and lower-bounds the quantity $\sum_{i=1}^{N} w_{t+1}[i]$

Upper bound

$$\sum_{i=1}^{N} w_{t+1}[i] = \sum_{i=1}^{N} w_t[i] e^{-\mu \ell_t[i]} \qquad \text{use } e^{-\alpha x} \le 1 - (1 - e^{-\alpha})x$$

$$\le \sum_{i=1}^{N} w_t[i](1 - (1 - e^{-\mu})\ell_t[i])$$

$$= \Big(\sum_{i=1}^{N} w_t[i]\Big)(1 - (1 - e^{-\mu})\langle p_t, \ell_t \rangle)$$

$$\ln \sum_{i=1}^{N} w_{t+1}[i] \le \ln \Big(\sum_{i=1}^{N} w_t[i]\Big) + \ln \left(1 - (1 - e^{-\mu})\langle p_t, \ell_t \rangle\right) \qquad \text{use } \ln(1-x) \le -x$$

$$\le \ln \Big(\sum_{i=1}^{N} w_t[i]\Big) - (1 - e^{-\mu})\langle p_t, \ell_t \rangle \qquad \text{telescope}$$

$$\ln \sum_{i=1}^{N} w_{T+1}[i] \le \ln \Big(\sum_{i=1}^{N} w_1[i]\Big) - (1 - e^{-\mu}) \sum_{t=1}^{T} \langle p_t, \ell_t \rangle$$

Lower bound

for **any** $j \in 1, \ldots, N$

$$\sum_{i=1}^{N} w_{t+1}[i] \geq w_{t+1}[j] \geq w_1[j] e^{-\mu \sum_{s=1}^{t} \ell_s[j]}$$

Combining

$$\ln w_1[j] - \mu \sum_{t=1}^{T} \ell_t[j] \leq \ln \sum_{i=1}^{N} w_{T+1}[i] \leq \overbrace{\ln(\sum_{i=1}^{N} w_1[i])}^{=\ln(1)=0} - (1 - e^{-\mu}) \sum_{t=1}^{T} \langle p_t, \ell_t \rangle$$

rearranging

$$\sum_{t=1}^{T} \langle p_t, \ell_t \rangle \leq \frac{-\ln w_1[j] + \mu \sum_{t=1}^{T} \ell_t[j]}{1 - e^{-\mu}}$$

remember j was arbitrary, and let $w_1[i] = 1/N$

$$\sum_{t=1}^{T} \langle p_t, \ell_t \rangle \leq \frac{\ln N + \mu \min_j \sum_{t=1}^{T} \ell_t[j]}{1 - e^{-\mu}}$$

- suppose we can upper bound the minimal loss (we are sure about at least one of our friends)

$$\min_j \sum_{t=1}^{T} \ell_t[j] \leq \tilde{L}$$

- if we set $\mu = \ln\left(1 + \sqrt{\frac{2\ln N}{\tilde{L}}}\right)$ then

$$\sum_{t=1}^{T} \langle p_t, \ell_t \rangle \leq \frac{\ln N + \mu \min_j \sum_{t=1}^{T} \ell_t[j]}{1 - e^{-\mu}}$$

$$\leq \min_j \sum_{t=1}^{T} \ell_t[j] + \sqrt{2\tilde{L}\ln N} + \ln N$$

- after rearranging we get a regret bound:

$$\sum_{t=1}^{T} \langle p_t, \ell_t \rangle - \min_j \sum_{t=1}^{T} \ell_t[j] \leq \sqrt{2\tilde{L}\ln N} + \ln N$$

- trivially: $\tilde{L} \leq T$ (yes, very loose bound)

$$\sum_{t=1}^{T} \langle p_t, \ell_t \rangle - \min_j \sum_{t=1}^{T} \ell_t[j] \leq \sqrt{2T \ln N} + \ln N$$

- much better if $\tilde{L} \ll T$
  (e.g., there is a friend that almost never errs, $\tilde{L} \simeq 0$)

$$\sum_{t=1}^{T} \langle p_t, \ell_t \rangle - \min_j \sum_{t=1}^{T} \ell_t[j] \lesssim \ln N$$

- not surprisingly looks similar to the Halving bound (realizability & finiteness hold)

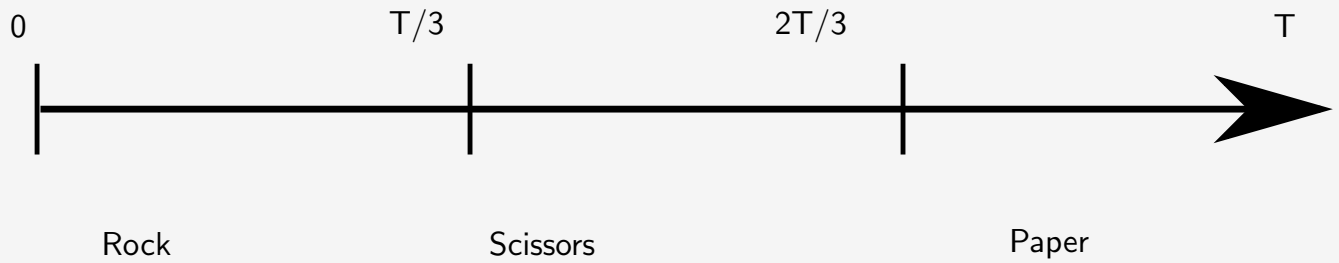
Hedge

1: init vector $w_1 \in \mathbb{R}_+^N$ s.t. $\sum_{i=1}^N w_1[i] = 1$, learning rate $\mu > 0$
2: **for** $t = 1, \ldots$ **do**
3: compute $p_t = \frac{w_t}{\sum_{i=1}^N w_t[i]}$
4: receive loss ℓ_t
5: update $w_{t+1}[i] = w_t[i] e^{-\mu \ell_t[i]}$

Exercise 3:

[Marchetti-Spaccamela'11]

- 3 experts: 1st playing always **Rock**, 2nd – **Scissors**, and 3rd – **Paper**
- your opponent plays Rock $T/3$ times, then Scissors $T/3$ times and then Paper $T/3$ times



- loss: -1 if won, $+1$ if lost, 0 if tie
- describe roughly 1) the most probable strategies played by Hedge, 2) when they switch and 3) the final distribution

- Hedge inspired Boosting – a powerful concept of combining weak algorithms into a strong one
- idea:
  - ➡ treat your training examples as experts
  - ➡ changing weights focuses attention on difficult examples
- ★ Gödel Prize 2003

Adversarial Multi-Armed Bandits

(how to use only one friend at a time in horse races)

One-armed bandits

- you are in a casino with slot-machines ("one-armed bandits")

- you have to find a machine that gives you most money
- can try one machine per time

- on the one hand, you should play the best machine so far → **exploitation**
- on the other hand, there might be better machines, not yet tried → **exploration**

"Considered by Allied scientists in WW2, it proved so intractable that the problem was proposed to be dropped over Germany so that German scientists "could also waste their time on it" *[Peter'79]*

- similar setup to expert's advice
- you can to bid on only one of the friends (experts) at a time
- consequently, no full loss vector ℓ_t is received (you don't know how much lost your other friends)
- you get only the loss $\ell_t[i_t]$ of the chosen friend i_t

Slight change to the Hedge algorithm:

Hedge

1: init w_1 s.t. $\sum_{i=1}^N w_1[i] = 1$, $\mu > 0$
2: **for** $t = 1, \ldots$ **do**
3: 'play all friends'

$$p_t = \frac{w_t}{\sum_{i=1}^N w_t[i]}$$

4: receive ℓ_t
5:

$$w_{t+1}[i] = w_t[i] e^{-\mu \ell_t[i]}$$

Exp3

1: init $w_1[i] = 1$, $\gamma \in (0, 1]$
2: **for** $t = 1, \ldots$ **do**
3: draw a friend i_t acc. to

$$p_t[i] = (1-\gamma) \frac{w_t}{\sum_{i=1}^N w_t[i]} + \frac{\gamma}{N}$$

4: receive $\ell_t[i_t]$
5:

$$w_{t+1}[i] = \begin{cases} w_t[i] e^{-\gamma \frac{\ell_t[i]}{p_t[i]}}, & \text{if } i = i_t \\ w_t[i], & \text{else} \end{cases}$$

- random surprise actions added
- "Exponential-weight algorithm for Exploration and Exploitation" $\Rightarrow$ "Exp3"

Exp3

1: init $w_1[i] = 1$, $\gamma \in (0, 1]$
2: **for** $t = 1, \ldots$ **do**
3: draw a friend i_t acc. to

$$p_t[i] = (1-\gamma)\frac{w_t}{\sum_{i=1}^{N} w_t[i]} + \frac{\gamma}{N}$$

4: receive $\ell_t[i_t]$
5: $w_{t+1}[i] = \begin{cases} w_t[i]e^{-\gamma \frac{\ell_t[i]}{p_t[i]}}, & \text{if } i = i_t \\ w_t[i], & \text{else} \end{cases}$
6: $w_{t+1}[i] = w_t[i]e^{-\gamma \tilde{\ell}_t[i]}$

- denote $\tilde{\ell}_t[i] = \begin{cases} \ell_t[i]/p_t[i] & \text{if } i = i_t \\ 0 & \text{otherwise} \end{cases}$
- then

$$\mathbb{E}\left[\tilde{\ell}_t[i] \mid i_1, \ldots, i_{t-1}\right] = p_t[i]\frac{\ell_t[i]}{p_t[i]} + (1 - p_t[i]) \cdot 0 = \ell_t[i]$$

Theorem

For any $\gamma \in (0, 1]$, $N > 0$ and any sequence of $\ell_1, \ldots, \ell_T$

$$\mathbb{E}\Big[\sum_{t=1}^{T} \ell_t[i_t]\Big] - \min_i \sum_{t=1}^{T} \ell_t[i] \leq 2\sqrt{e-1}\sqrt{TN \ln N}$$

Comparison to the full-information

- Hedge: $O(\sqrt{T \ln N})$
- Exp3: $O(\sqrt{TN \ln N})$ the price of bandit info

- very similar proof to Hedge
- see appendix

Exercise 4 (optional):

- explain why is the exploration necessary?
  - ➡ point where the proof will fail if

$$p_t[i] = w_t[i] / \sum_{i=1}^{N} w_t[i]$$

Stochastic Bandits

- restrict somewhat the adversarial setting
- as we have seen, restrictions lead to nicer regret (possibly under a different definition)
- N arms as before
- this time arm's loss $\ell_t[i]$ is sampled i.i.d. from $\mathcal{D}_i$ (unknown and fixed)
  - ➡ $\ell_t[i]$ and $\ell_t[j]$ are independent for $i \neq j$
  - ➡ mean loss $\mu_i = \mathbb{E}[\ell_t[i]]$
  - ➡ $\mu^* = \min_i \mu_i$
  - ➡ $i^* = \arg\min_i \mu_i$
- suffered loss $\ell_t[i_t]$

note: $\mathcal{D}_{i,t}$'s may change, keeping $\mu_{i,t}$ fixed

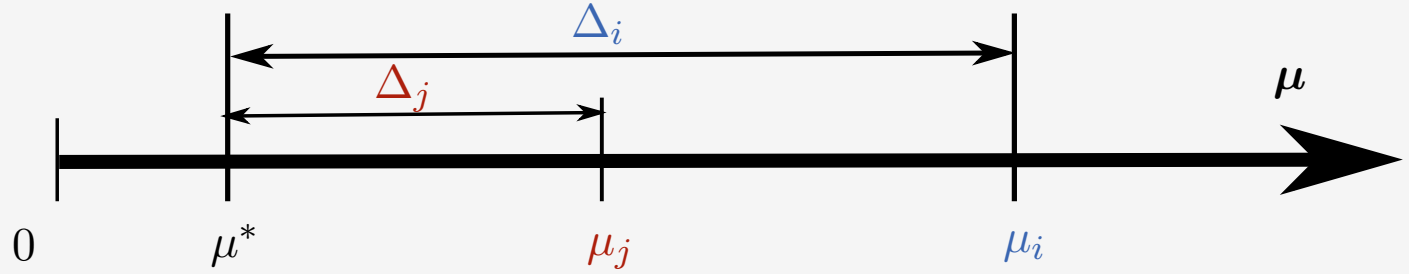
- mean loss $\mu_i = \mathbb{E}[\ell_t[i]]$
- $\mu^* = \min_i \mu_i$
- $n_i(T)$ – number of pulls of i'th arm over first T plays

Goal:

- now it makes sense to speak of expected regret
- regret:

$$
\begin{aligned}
R_T &= \mathbb{E}\Big[\sum_{t=1}^{T} \ell_t[i_t]\Big] - \min_i \mathbb{E}\Big[\sum_{t=1}^{T} \ell_t[i]\Big] \\
&= \mathbb{E}\Big[\sum_{t=1}^{T}\sum_{i=1}^{N} \ell_t[i]\llbracket i = i_t \rrbracket\Big] - \min_i T\mu_i \\
&= \sum_{i=1}^{N} \mu_i \mathbb{E}\Big[\sum_{t=1}^{T} \llbracket i = i_t \rrbracket\Big] - T\mu^* \\
&= \sum_{i=1}^{N} \mu_i \mathbb{E}[n_i(T)] - T\mu^* \qquad \to \min
\end{aligned}
$$

Gaps: $\Delta_i = \mu_i - \mu^* \geq 0$



$$R_T = \sum_{i=1}^{N} \mu_i \mathbb{E}[n_i(T)] - T\mu^* = \sum_{i=1}^{N} \Delta_i \mathbb{E}[n_i(T)]$$

Next Theorem(s)

$$R_T \leq C \sum_{i=2}^{N} \frac{\ln T}{\Delta_i} + o(\ln T)$$

- smaller gaps $\Rightarrow$ bigger regret
- intuition: takes more time to distinguish between close arms
- compare to Exp3: $O(\sqrt{NT \ln N})$

$$R_T \leq C \sum_{i=2}^{N} \frac{\ln T}{\Delta_i} + o(\ln T)$$

Why $\frac{1}{\Delta_i}$?

Intuitive example:

- imagine Bernoulli losses with p_i
- mean estimates $\bar{\mu}_i = \frac{1}{T} \sum_{t=1}^{T} \ell_t[i]$ have variance $\sigma^2 = p_i(1-p_i)/T$
- if $T \simeq \frac{1}{\Delta_i^\alpha}$ then $\sigma^2 \simeq \Delta_i^\alpha p_i(1-p_i)$
- if $\alpha \simeq 1$, hard to say if there is a real difference between μ^* and μ_i or it's just variance
- so we need rather $\alpha \simeq 2$
- as on every pull we loose about Δ_i, for $\alpha \simeq 2$ we have $\Delta_i \frac{1}{\Delta_i^\alpha} \simeq \frac{1}{\Delta_i}$

Simplest strategy:

ϵ-greedy

1: $\epsilon > 0$, $\bar{\mu}_i = 0$ empirical means of rewards
2: **for** $t = 1, \ldots$ **do**
3: with prob. $1 - \epsilon$ play current best arm $i_t = \arg\min_i \bar{\mu}_i$
4: with prob. ϵ play a random arm
5: receive $\ell_t[i_t]$
6: update empirical means ($\bar{\mu}_{i_t} = \frac{\bar{\mu}_{i_t} \cdot n_{i_t} + \ell_t[i_t]}{n_{i_t}+1}$)

Regret:

- because of the constant ϵ, $R_T \sim \epsilon T$
- need to let ϵ to to zero at a certain rate

Modified simplest strategy:

ϵ_t-decreasing

$c > 0$, $\delta > 0$, $\bar{\mu}_i = 0$
for $t = 1, \ldots$ **do**
    $\epsilon_t = \min\{1, \frac{cN}{\delta^2 t}\}$
    with prob. $1 - \epsilon_t$ play current best arm $i_t = \arg\min_i \bar{\mu}_i$
    with prob. ϵ_t play a random arm
    receive $\ell_t[i_t]$
    update means

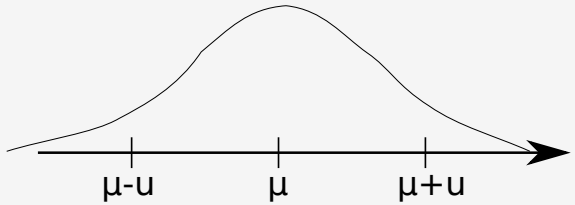
Theorem

If $0 < \delta \leq \min_{i:\mu_i<\mu^*} \Delta_i < 1$ and $T \geq \frac{cN}{\delta}$

$$P\{i_t \neq i^*\} \leq \frac{c}{\delta^2 t} + o(\frac{1}{t})$$
$$\Rightarrow R_T = O(\ln T)$$

UCB-style strategies:

- pull arms and maintain empirical averages of their losses
- calculate also confidence intervals (a region around our estimate so that the true value is within with high prob.)



- repeated plays shrink the confidence bound, the average is becoming more reliable
- now stick to the principle: "optimism in the face of uncertainty"
- play the arm whose mean loss combined with confidence bound promises the least loss
- eventually the most optimistic arm will change because
  - ➡ either that is really better
  - ➡ or it wasn't sampled often enough
- deterministic algorithm unlike ϵ-greedy

UCB

1: play each arm once
2: **for** $t = 1, \ldots$ **do**
3: play the arm $i_t = \arg\min_i \left(\bar{\mu}_i - \sqrt{\frac{2 \ln t}{n_i(t)}} \right)$
4: receive $\ell_t[i_t]$
5: update averages

Theorem 1

For $N > 0$, $T > 0$ and arbitrary distributions $\mathcal{D}_{i,t}$ with fixed means μ_i

$$R_T = \left[8 \sum_{i:\mu_i > \mu^*} \frac{\ln T}{\Delta_i} \right] + (1 + \frac{\pi^2}{3}) \sum_{i=1}^{N} \Delta_i$$

- lot's of applications where traditional batch learning may fail
- goals are formulated in terms of various regrets
- the slower the upper bound on regret grows with T the better
- full information
  - ➡ realizability
    - Consistent $R_T \leq |\mathcal{H}| - 1$
    - Halving $R_T \leq \log_2 |\mathcal{H}|$
  - ➡ commitment + surprise
    - Randomized $R_T \leq \sqrt{2T \ln |\mathcal{H}|}$
    - Hedge $R_T = O(\sqrt{T \ln N})$
- adversarial bandits
  - ➡ EXP3 $R_T = O(\sqrt{NT \ln N})$
- stochastic bandits
  - ➡ ϵ-decreasing $R_T = O(\ln T/\delta)$
  - ➡ UCB $R_T = O(\ln T/\Delta_2)$

- barely scratched the surface with most important algorithms
- advanced algorithms use EXP3, UCB as building blocks

Material for this lecture

1. S. Shalev-Shwartz. "Online Learning and Online Convex Optimization", 2012
2. Y. Seldin. "The Space of Online Learning Algorithms", 2015
3. P. Auer et al. "The nonstochastic multiarmed bandit problem", 2002
4. P. Auer et al. "Finite-time Analysis of the Multiarmed Bandit Problem", 2002

Advanced Contextual Algorithms

1. EXP4: P. Auer et al. "The nonstochastic multiarmed bandit problem", 2002
2. LinUCB: Li et al. "A contextual-bandit approach to personalized news article recommendation", 2010
3. Epoch-greedy: J. Langford "The epoch-greedy algorithm for multi-armed bandits with side information", 2003

Appendix

- similar to the Hedge algorithm
- idea: lower- and upper-bound $\sum_{i=1}^{N} w_{t+1}[i] / \sum_{i=1}^{N} w_t[i]$
- denote $\hat{\ell}_t[i] = \begin{cases} \ell_t[i]/p_t[i] & \text{if } i = i_t \\ 0 & \text{otherwise} \end{cases}$

Upper bound

$$\frac{\sum_{i=1}^N w_{t+1}[i]}{\sum_{i=1}^N w_t[i]} = \sum_{i=1}^N w_t[i] e^{-\frac{\gamma \hat{\ell}_t[i]}{N}} \qquad \text{(use } w_{t+1}[i] \text{ definition; note the bound } \hat{\ell}_t[i] \le \frac{\gamma}{N}\text{)}$$

$$= \sum_{i=1}^N \frac{p_t[i] - \frac{\gamma}{N}}{1-\gamma} e^{-\frac{\gamma \hat{\ell}_t[i]}{N}} \qquad \text{(using } e^x \le 1 + x + (e-2)x^2 \text{ for } |x| \le 1\text{)}$$

$$\le \sum_{i=1}^N \frac{p_t[i] - \frac{\gamma}{N}}{1-\gamma} \left[1 - \frac{\gamma \hat{\ell}_t[i]}{N} + (e-2)\left(\frac{\gamma \hat{\ell}_t[i]}{N}\right)^2\right]$$

$$\le 1 - \frac{\frac{\gamma}{N}}{1-\gamma} \sum_{i=1}^N p_t[i]\hat{\ell}_t[i] + \frac{(e-2)(\frac{\gamma}{N})^2}{(1-\gamma)} \sum_{i=1}^N p_t[i](\hat{\ell}_t[i])^2$$

$$\le 1 - \frac{\frac{\gamma}{N}}{1-\gamma} \ell_t[i_t] + \frac{(e-2)(\frac{\gamma}{N})^2}{(1-\gamma)} \sum_{i=1}^N \hat{\ell}_t[i] \qquad \text{(use } \ln(1+x) \le x\text{)}$$

$$\ln \frac{\sum_{i=1}^N w_{t+1}[i]}{\sum_{i=1}^N w_t[i]} \le -\frac{\frac{\gamma}{N}}{1-\gamma} \ell_t[i_t] + \frac{(e-2)(\frac{\gamma}{N})^2}{(1-\gamma)} \sum_{i=1}^N \hat{\ell}_t[i]$$

Upper bound (cont.)

sum over $t = 1, \ldots, T$

$$\ln \frac{\sum_{i=1}^N w_{T+1}[i]}{\sum_{i=1}^N w_T[i]} \leq -\frac{\frac{\gamma}{N}}{1-\gamma} \sum_{t=1}^T \ell_t[i_t] + \frac{(e-2)(\frac{\gamma}{N})^2}{(1-\gamma)} \sum_{t=1}^T \sum_{i=1}^N \hat{\ell}_t[i]$$

Lower bound

for any $j = 1, \ldots, N$

$$\ln \frac{\sum_{i=1}^N w_{T+1}[i]}{\sum_{i=1}^N w_T[i]} \geq \ln \frac{w_{T+1}[j]}{\sum_{i=1}^N w_T[i]} = -\frac{\gamma}{N} \sum_{t=1}^T \hat{\ell}_t[j] - \ln N$$

Combining (and simplifying)

$$\sum_{t=1}^T \ell_t[i_t] \leq (1-\gamma) \sum_{t=1}^T \hat{\ell}_t[j] + \frac{N \ln N}{\gamma} + (e-2)\frac{\gamma}{N} \sum_{t=1}^T \sum_{i=1}^N \hat{\ell}_t[i]$$

expectation to rescue:

$$\mathbb{E}\big[\hat{\ell}_t[i] \;\big|\; i_1, \ldots, i_{t-1}\big] = p_t[i]\frac{\ell_t[i]}{p_t[i]} + (1 - p_t[i]) \cdot 0] = \ell_t[i]$$

Take expectation

$$\mathbb{E}\big[\sum_{t=1}^{T} \ell_t[i_t]\big] \leq (1-\gamma)\sum_{t=1}^{T} \ell_t[j] + \frac{N \ln N}{\gamma} + (e-2)\frac{\gamma}{N}\sum_{t=1}^{T}\sum_{i=1}^{N} \ell_t[i]$$

since j was arbitrary and by assumption $\sum_{t=1}^{T}\sum_{i=1}^{N} \ell_t[i] \leq N\tilde{L}$

$$\leq (1-\gamma)\min_j \sum_{t=1}^{T} \ell_t[j] + \frac{N \ln N}{\gamma} + (e-2)\frac{\gamma}{N} N\tilde{L}$$

choose $\gamma = \min\big\{1, \sqrt{\frac{N \ln N}{(e-1)\tilde{L}}}\big\}$

$$\leq 2\sqrt{e-1}\sqrt{\tilde{L} N \ln N} \qquad \square$$