5.1-5.3 Online learning and online convex optimization

1. Introduction

Online learning setting:

Learner

$x_t \uparrow \ \downarrow p_t \uparrow y_t \ \cdots$

Nature

where $x_t \in \mathcal{X}$ (input), $p_t \in \mathcal{Y}$ (prediction), $y_t \in \mathcal{Y}$

Learner is a function A s.t. $p_{t+1} = A(x_{1:t}, p_{1:t}, y_{1:t}, x_{t+1})$

[Definition 24] regret

(i) The regret of a learner w.r.t. an expert h:

$$\text{Regret}(h) := \sum_{t=1}^{T} [\ell(y_t, p_t) - \ell(y_t, h(x_t))]$$

(ii) The regret w.r.t. a class of experts $\mathcal{H}$ is:

$$\text{Regret} := \max_{h \in \mathcal{H}} \text{Regret}(h) = \sum_{t=1}^{T} \ell(y_t, p_t) - \min_{h \in \mathcal{H}} \sum_{t=1}^{T} \ell(y_t, h(x_t))$$

---

2. Warm-up

[Example 24] negative result (Not suitable for online learning)

$\mathcal{Y} = \{-1, +1\}$, $\ell(y_t, p_t) = \mathbb{1}\{y_t \neq p_t\}$. Assume A is deterministic.

Claim: for all A, $\exists\ \mathcal{H}$, $x_{1:T}$, $y_{1:T}$ s.t. Regret $\geq \frac{T}{2}$.

Key point: adversary can choose y_t to be always different from p_t.

Then $\sum_{t=1}^{T} \ell(y_t, p_t) = T$.

Consider two experts $\mathcal{H} = \{h_{-1}, h_{+1}\}$, where h_y always predicts y.

$\Rightarrow \ell(y_t, h_{-1}(x_t)) + \ell(y_t, h_{+1}(x_t)) = 1$,

therefore, $\min_{h \in \mathcal{H}} \sum_{t=1}^{T} \ell(y_t, h(x_t)) \leq \frac{T}{2}$

$\Rightarrow$ Regret $\geq \frac{T}{2}$.

[Example 29] positive result (learn with expert advice)

$\ell(y_t, p_t) = \mathbb{1}\{y_t \neq p_t\}$

[Assumption 3] relizable.

best $h^* \in H$ s.t. $\ell(y_t, h^*(x_t)) = 0$ for all $t = 1, \cdots, T$.

[Algorithm 1] majority algorithm

(1) Maintain a set $V_t \subseteq H$ of valid experts

(2) On each t, predict p_t to be the majority vote over $\{h(x_t) : h \in V_t\}$

(3) Keep experts which were correct $V_{t+1} = \{h \in V_t : y_t = h(x_t)\}$

Analysis:

(i) On each mistake, at least half of the experts are eliminated.

So $1 \leq |V_{T+1}| \leq |H| 2^{-M}$, where M is the number of mistakes.

(ii) M is the exactly the regret here.

(iii) Take logs : Regret $\leq \log_2 |H|$

Note: Realizability is too strong.

---

3. Online convex optimization. S : convex set

[Definition 25] convexity

A function $f: S \to \mathbb{R}$ is convex iff for all points $w \in S$, there is some vector $z \in \mathbb{R}^d$ s.t.

$$f(u) \geq f(w) + z \cdot (u - w) \quad \text{for all } u \in S. \quad (1)$$

[Definition 26] subgradient

For each $w \in S$, the set of all z satisfying (1) are known as subgradients at w:

$$\partial f(w) := \{z : f(u) \geq f(w) + z \cdot (u - w) \text{ for all } u \in S\}$$

(i) If f is differentiable at w, $\partial f(w) = \{\nabla f(w)\}$

(ii) If $0 \in \partial f(w)$, w is a global minimum of f.

Convex functions:

(i) $f(w) = w \cdot z$, $\forall z \in \mathbb{R}^d$

(ii) $f(w) = w^T A w$, $\forall$ positive semidefinite $A \in \mathbb{R}^{d \times d}$

(iii) Negative entropy: $f(w) = \sum_{i=1}^{d} w_i \log w_i$ on $w \in \Delta_d$

(iv) Sum: $f+g$ if f and g are convex.

(v) Scale: cf, $\forall c > 0$, f convex

(vi) Supremum: $\sup_{f \in F} f$ where F is a family of convex functions.

Setup for online convex optimization:

- Iteration $t = 1, \cdots, T$

(i) Learner chooses $w_t \in S$

(ii) Nature chooses convex loss function $f_t : S \to \mathbb{R}$

Formally: $w_{t+1} = A(w_{1:t}, f_{1:t})$

$$\text{Regret}(u) := \sum_{t=1}^{T} [f_t(w_t) - f_t(u)] \quad \text{for } u \in S$$

$$\text{Regret} := \max_{u \in S} \text{Regret}(u).$$

Now we show some examples of reducing online learning to online convex learning. [OL: online learner, OCO: online convex optimizer.]

[Example 26] linear regression

$\ell(y_t, p_t) = (p_t - y_t)^2$

On each iteration $t = 1, \cdots, T$:

OL receives $x_t \in \mathbb{R}^d$ from nature.

OL asks OCO for a weight vector $w_t \in \mathbb{R}^d$.

OL sends $p_t = w_t \cdot x_t$ to nature.

OL receives y_t from nature.

OL relays feedback to OCO via $f_t(w) = (w \cdot x_t - y_t)^2$

[Example 27] learning with expert advice.

Assume $\mathcal{H}$ is finite. Let $\mathcal{H} = \{h_1, \cdots, h_d\}$, $\ell(y_t, p_t) = \mathbb{1}\{y_t \neq p_t\}$.

Then neither ℓ or $\mathcal{H}$ is convex.

Nonetheless, we can convexify the problem via randomization.

* Formally, the reduction is as follows:

(i) OL receives x_t.

(ii) OL asks OCO for $w_t \in \Delta_d$

(iii) OL samples an expert $j_t \sim w_t$ and send $p_t = h_{j_t}(x_t)$

(iv) OL receives y_t

(v) OL relays the feed back to OCO via $f_t(w) = w \cdot z_t$,

where $z_t = [\ell(y_t, h_1(x_t)), \cdots, \ell(y_t, h_d(x_t))] \in \{0,1\}^d$

Note: (i) The expected regret of OL is the same as the regret of OCO.

(ii) Another way is to use upper bound (hinge loss or logistic loss upper bound zero-one loss). However, minimising the convex upper bound doesn't guarantee minimizing the 0-1 loss.