在线学习入门十讲

Intro to Online Learning

1. Online Learning Setup
2. Evaluation of the Learner
3. Online Convex Optimization
4. Reduce Online Learning to Online Optimization
5. Reduce Online Learning to Batch Learning
6. FTL Algorithm
7. BTL Algorithm
8. FTRL Algorithm
9. Online Gradient Descent
10. Bandit Problem.

Online Learning

1. Online Learning Setup

Classical supervised learning: ① $(x^{(i)}, y^{(i)}) \overset{i.i.d}{\sim} P_{train}$

② $P_{train} = P_{test}$

In real: data could be generated in sequence

$P_{test} \neq P_{train}$

↓

domain shift 领域转移

⇓

generalization

transfer learning

Online Learning:

deal with "worst-case" domain shift

e.g. Recommendation system

↳ adaptive, dynamic

features:

1) data may be adversarial 敌对的

2) data and predictions are sequential

3) feedback is limited

4) online learning learns while testing 训练和测试相互融合

Online Learning
- ① learner / agent / algorithm / player → minimize the loss
- ② environment / nature → maximize the loss

⇔ game 博弈

Game: T rounds

Algorithm: General Online learning Problem.

for $t = 1$ to T do:

learner receives $x_t \in \mathcal{X}$ from environment.

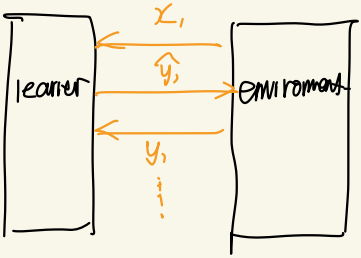
learner predicts $\hat{y}_t \in \mathcal{Y}$

may be chosen adversarially

learner receives the label y_t from environment,

$\Rightarrow$ loss $\ell(y_t, \hat{y}_t)$

"agnostic"



How to generate (x_t, y_t)?

(adversarial setting)

2. Evaluation of the Learner

How to evaluate the performance of the learner?

cumulative loss $\sum_{t=1}^{T} \ell(y_t, \hat{y}_t)$ $\qquad \ell \geq 0$

"best model" in hindsight: $h^* = \underset{h \in \mathcal{H}}{\operatorname{argmin}} \sum_{t=1}^{T} \ell(y_t, h(x_t))$

后见之明

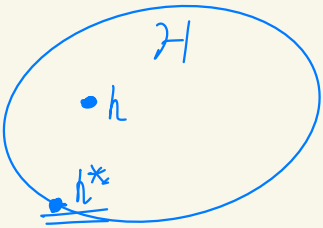
baseline

遗憾

$$\boxed{\text{Regret} = \sum_{t=1}^{T} \ell(y_t, \hat{y}_t) - \min_{h \in \mathcal{H}} \sum_{t=1}^{T} \ell(y_t, h(x_t))}$$

遗憾界

If the best model in hindsight performs well, then the learner has more responsibility to learn to predict well.



Risk $R(h) - R(h^*)$.

excess risk (estimation error)

generalization.

def (Realizable problem) An online learning problem is realizable (w.r.t. a family of predictor $\mathcal{H}$) if $\exists\, h \in \mathcal{H}$ s.t. $\forall T$: $\sum_{t=1}^{T} \ell(y_t, h(x_t)) = 0$

Example. Majority Algorithm (Halving algorithm)

binary classification $y_t \in \{0, 1\}$. $\ell(y_t, \hat{y}_t) = \mathbb{1}(y_t \neq \hat{y}_t)$

realizable: $\exists\, h^* \in \mathcal{H}$. $y_t = \hat{y}_t^* = h^*(x_t)$

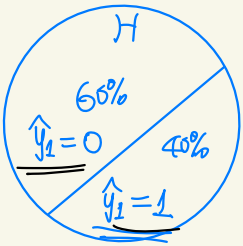
$$\text{Regret} = \sum_{t=1}^{T} \ell(y_t, h(x_t)) \triangleq M \quad (\#\text{ of mistakes in } T \text{ rounds})$$

at each round, the learner maintains a set $V_t \subseteq \mathcal{H}$

$$\forall h \in V_t: \sum_{i=1}^{t} \ell(y_i, h(x_i)) = 0$$

(初始时为此情况本犯错的假设)

$\hat{y}_t$ is simply the prediction made by majority of $h \in V_t$.



$x_1 \Rightarrow \hat{y}_1 = 0$ <

正确: $V_1 = V_0$

错误: $V_1 = \{h \in V_0 : h(x_1) = y_1\}$

(把分类错误的假设从序列中删除)

每一轮 <

正确: maintain

错误: $|V_{t+1}| \leq \frac{1}{2} \cdot |V_t|$

realizable $\Rightarrow$ $|V_t| \neq \emptyset$. $(h^* \in V_t)$

finite hypothesis space

$$1 \le |V_T| \le |\mathcal{H}| \cdot \left(\frac{1}{2}\right)^M.$$

# of Mistakes $\Rightarrow$ $M \le \log|\mathcal{H}|$

$\text{Regret} = M \le \log|\mathcal{H}|$

最多只犯 $\log|\mathcal{H}|$ 次错误，就可以把 h^* 找出！

3. Online Convex Optimization (OCO)

Online (convex) Optimization:

for $t = 1$ to T do:

the learner picks some action $w_t \in \Omega$ from action space Ω. (h)

the environment picks a (convex) function $f_t: \Omega \to [0,1]$ ($\ell(y_t, \cdot)$)

the learner suffers the loss $f_t(w_t)$ and observe the entire loss function $f_t(\cdot)$

↳ the learner: minimize the function f_t at each step

$$\text{Regret} = \sum_{t=1}^{T} f_t(w_t) - \underbrace{\min_{w \in \Omega} \sum_{t=1}^{T} f_t(w)}_{\text{best action in hindsight}}$$

It seems an impossible task! $(t=1)$

$(x_t, y_t) \overset{i.i.d}{\sim} P$

OCO 三个情况

- Stochastic setting: $f_1, \dots, f_T \overset{i.i.d}{\sim} P$ → non-adversarial
- $\boxed{\text{oblivious setting:}}$ 研究 ☆ $f_1, \dots, f_T$ are chosen arbitrarily but before the game starts → adversarial, not adaptive (x_t, y_t)
- adaptive setting (non-oblivious)

$\downarrow$

t. f_t can depend on the learner's actions $w_1, \cdots, w_t$.

$\rightarrow$ adversarial. adaptive.

learner $\begin{cases} \text{deterministic} \rightarrow \text{frequentist} \\ \text{stochastic} \rightarrow \text{Bayesian} \end{cases}$ PAC

If learner is deterministic, adaptive setting = oblivious setting

4. Reduce Online Learning to Online Optimization

$$H = \{h_w : w \in \Omega\} \Rightarrow \hat{y}_t = h_{w_t}(x_t)$$

参数更新方法

If the learner posseses oracle to solve online optimization problem, the learner can consult the oracle to obtain w_t, the parameter of the model as in online optimization problem, and then predict $\hat{y}_t = h_{w_t}(x_t)$

Then send back $\ell(y_t, \hat{y}_t)$ to the oracle for optimizing the parameter.

Example 1. Online Regression. $h_w(x) = w^T x$, $w \in \Omega$

ℓ: squared error.

for $t = 1$ to T:

the learner receives $x_t \in \mathbb{R}^d$ from the environment.

the learner predict $\hat{y}_t$.

the environment selects y_t and sends it to the learner.

the learner suffers loss $\ell(y_t, \hat{y}_t) = (y_t - \hat{y}_t)^2$.

$\Downarrow$

OCO $\begin{cases} w_t \in \Omega : \text{learner.} \\ f_t \in \mathcal{F} : \text{enviroment.} \end{cases}$

for $t=1$ to T:

the learner receives $x_t \in \mathbb{R}^d$ from the enviroment.

the learner gives x_t to the $\boxed{\text{OCO solver}}$ and obtains $w_t \in \mathbb{R}^d$

oracle.

the learner predicts $\hat{y}_t = h_{w_t}(x_t) = w_t^T x_t$.

the environment selects y_t and sends it to the learner.

the learner suffers loss $(y_t - h_{w_t}(x_t))^2$

with (x_t, y_t) observed, the learner can reconstruct the loss function

$f_t(w) = (y_t - h_w(x_t))^2$ and give it to OCO solver

f_t in OCO $\Longleftrightarrow$ squared error loss functions for (x_t, y_t)

w_t in OCO $\Longleftrightarrow$ parameters of linear model h_{w_t}

Example 2 The Expert Problem 专家问题

$\mathcal{H}$: N experts "which one should we follow ?"

每一个假设代表了一个专家 $\Rightarrow$ choose a h_t wisely at each time step

Binary $\hat{y} \in \{0,1\}$. 0/1 loss.

☆☆ 完全信息在线学习

for $t=1$ to T do:

the learner obtains predictions from N experts

the learner chooses to follow prediction of one of the experts $i_t \in [N]$

the environment gives the learner the true value.

$\Rightarrow$ the learner is able to learn the loss of each experts: $l_t \in \{0,1\}^N$

the learner suffers the loss $l_t(i_t)$

$$\Rightarrow \text{Regret} = \mathbb{E}\left[\sum_{t=1}^{T} \ell_t(i_t) - \underbrace{\min_{i\in[N]} \sum_{t=1}^{T} \ell_t(i)}_{\text{best expert in hindsight.}}\right]$$

$i_t \sim P_t$: $P_t(j) = \begin{cases} 1, & j = i_t \\ 0, & j \neq i_t \end{cases}$

- Continuous action space

Choose i_t from $\Omega = [N]$

N-dimensional simplex

$\rightarrow$ choose a distribution P_t from $\Delta(N) = \{p \in \mathbb{R}^N : \|p\|_1 = 1, p \succeq 0\}$

belief (confidence)

then sample $i_t \sim P_t$

loss: $\mathbb{E}_{i\sim P_t}[\ell_t(i)] = \sum_{i\in[N]} P_t(i)\cdot \ell_t(i) = \langle P_t, \ell_t \rangle$

for $t = 1$ to T do:

the learner obtains predictions from N experts

the learner chooses a distribution $P_t \in \Delta(N)$

the learner samples (randomness) one expert $i_t \sim P_t$.

the environment gives the learner the true value

$\Rightarrow$ the learner obtains the loss of all experts $\ell_t \in \{0,1\}^N$

the learner suffer expected loss $\langle P_t, \ell_t \rangle$

OCO

for $t = 1$ to T do:

the learner obtains predictions from N experts

the learner invokes the $\boxed{\text{OCO oracle}}$ to obtain $P_t \in \Delta(N)$

the learner choose to follow prediction of one of the experts, $i_t \in [N]$. (by sampling through p_t)

the environment gives the learner the true value.

$\Rightarrow$ the learn obtains the loss $l_t \in \{0,1\}^N$

the learner suffers the loss $\langle p_t, l_t \rangle$.

the learner can construct the loss function $f_t(p) = \langle p, l_t \rangle$

and then give it to OCO oracle.

OCO: $$\text{Regret} = \sum_{t=1}^{T} \underbrace{\langle p_t, l_t \rangle}_{f_t(p_t)} - \min_{p \in \Delta(N)} \sum_{t=1}^{T} \underbrace{\langle p, l_t \rangle}_{f_t(p)} = \sum_{t=1}^{T} f_t(p_t) - \min_{p \in \Delta(N)} \sum_{t=1}^{T} f_t(p)$$

Online learning: $$\text{Regret} = \sum_{t=1}^{T} \mathbb{E}_{i \sim p_t}[l_t(i)] - \min_{i \in [N]} \sum_{t=1}^{T} l_t(i)$$

$$\sum_{t=1}^{T} \langle p, l_t \rangle = \left\langle p, \sum_{t=1}^{T} l_t \right\rangle \geq \min_{i \in [N]} \sum_{t=1}^{T} l_t(i)$$

with equality when $\begin{cases} p(i) = 1 & \text{if } i = \arg\min_i \sum_{t=1}^{T} l_t(i) \\ p(i) = 0 & \text{otherwise} \end{cases}$

5. Reducing Online Learning to Batch Learning

在线-批量转换

Standard supervised learning : i.i.d dataset $\{(x_t, y_t)\}_{t=1}^{T}$

Risk : $R(w)$: population risk $R(w) = \mathbb{E}_{(x_t, y_t)}[\ell(y_t, h_w(x_t))] \in [0,1]$

$\hat{R}(w)$: empirical risk $\hat{R}(w) = \frac{1}{T} \sum_{t=1}^{T} \ell(y_t, h_w(x_t))$

assume: ℓ $\begin{cases} \text{bounded } |\ell(y, \hat{y})| \leq 1 \\ \text{convex (w.r.t. second argument)} \end{cases}$

excess-risk (estimation error) : $R\left(\frac{1}{T} \sum_{t=1}^{T} w_t\right) - R(w^*)$

Thm (relationship between excess risk and regret) Suppose we run an online learning algorithm on dataset $\{(x_t, y_t)\}_{t=1}^{T}$ and obtain a sequence of models: $w_1, \cdots, w_T$ and Regret R_T. Let $\bar{w} = \frac{1}{T}\sum_{t=1}^{T} w_t$, $w^* = \arg\min_{w \in \Omega} R(w)$

generalization bound:

泛化界

$$R(\bar{w}) - R(w^*) \le \frac{R_T}{T} + \tilde{O}\left(\frac{1}{\sqrt{T}}\right) \quad \text{w.h.p.}$$

learned params　　best params

O: 数学常数

$\tilde{O}$: 忽略对数

证明详见FML系列 8.3节

PAC: probably approximately correct.

$$\Pr\left(|R(\bar{w}) - R(w^*)| \ge \varepsilon\right) \le \delta$$

近似正确　概率上

① $R_T = \Omega(T) \Rightarrow$ trivial

② batch problem

generalization bound $\begin{cases} O\left(\frac{1}{T}\right) \Rightarrow \text{regret}: \tilde{O}(1) \\ O\left(\frac{1}{\sqrt{T}}\right) \Rightarrow \text{regret}: O(\sqrt{T}) \end{cases}$ ✓

most commonly

6. FTL Algorithm

Follow-The-Leader:

$$\boxed{w_t = \arg\min_{w \in \Omega} \sum_{i=1}^{t-1} f_i(w)}$$

the historical loss the learner has seen so far

FTL algorithm fails to perform in many cases.

Example. FTL on expert problem. $f_t(p) = \langle p, l_t \rangle$

$$p_t = \arg\min_{p \in \Delta(N)} \sum_{i=1}^{t-1} f_i(p) = \arg\min_{p \in \Delta(N)} \sum_{i=1}^{t-1} \langle p, l_i \rangle$$

$$= \arg\min_{p \in \Delta(N)} \left\langle p, \sum_{i=1}^{t-1} l_i \right\rangle$$

$$\Rightarrow P_t(i_t) = \begin{cases} 1, & i_t = \underset{j\in[N]}{\operatorname{argmin}} \sum_{i=1}^{t-1} l_i(j) \\ 0, & \text{o.w.} \end{cases}$$

point-mass probability $\Rightarrow$ deterministic 确定性算法!

$$\sum_{i=1}^{t-1} f_i(P_t) = \sum_{i=1}^{t-1} l_i(i_t)$$

Weather forecasting:

Day	1	2	3	4	...	...
偶数天正确 → Expert 1 's loss	1	0	1	0	...	...
奇数天正确 → Expert 2 's loss	0	1	0	1	...	...
FTL choice i_t	1	2	1	2	...	...

Cumulative loss 1: T
2: —

若存在多个minimizer
- 按照某个次序选取 → deterministic (1和2累计损失相同时，选择1)
- 按照均匀分布采样 → stochastic

$\Downarrow$

mistakes: T.

mistakes of the best expert in hindsight: $\frac{T}{2}$

$\Rightarrow$ Regret $= T - \frac{T}{2} = \boxed{\Theta(T)}$ adversarial

Every deterministic algorithm cannot reach regret $o(T)$ for all data/environment

$\Rightarrow$ Add some randomness!

7. BTL Algorithm

Be-The-Leader: At time step t, BTL strategy chooses the action that would have performed best on $f_1, \cdots, f_{t-1}$ and f_t.

$\Leftrightarrow$ BTL action at time t is $w_{t+1} = \underset{w\in\Omega}{\operatorname{argmin}} \sum_{i=1}^{t} f_i(w)$ ← illegal!

because we make choice w_{t+1} before seeing f_t !!!

在"未来"的最优选择！ 并不是一个实际可行的策略！

Proposition 1 BTL strategy has non-positive regret.

预知未来.

$$w_t = \arg\min_{w\in\Omega} \sum_{i=1}^{t-1} f_i(w), \quad \Rightarrow \text{BTL regret} = \sum_{t=1}^{T} f_t(w_{t+1}) - \min_{w\in\Omega} \sum_{t=1}^{T} f_t(w) \le 0 \quad (*)$$

FTL result

Proof: (induction) $T=1$: $f_1(w_2) - \min_{w\in\Omega} f_1(w) = f_1(w_2) - f_1(w_2) = 0$.

Suppose $(*)$ holds for all $t \le T-1$, $f_1 \cdots f_{T-1}$

$$\sum_{t=1}^{T} f_t(w_{t+1}) - \min_{w\in\Omega} \sum_{t=1}^{T} f_t(w) = \sum_{t=1}^{T} f_t(w_{t+1}) - \sum_{t=1}^{T} f_t(w_{T+1})$$

$$= \sum_{t=1}^{T-1} f_t(w_{t+1}) - \sum_{t=1}^{T-1} f_t(w_{T+1}) \quad \text{(final summands cancel)}$$

$$\le \sum_{t=1}^{T-1} f_t(w_{t+1}) - \min_{w\in\Omega} \sum_{t=1}^{T-1} f_t(w) \quad \text{(induction hypothesis)}$$

$$\le 0 \qquad \square$$

Proposition 2 (FTL regret bound)

$$\text{FTL regret} = \sum_{t=1}^{T} f_t(w_t) - \min_{w\in\Omega} \sum_{t=1}^{T} f_t(w) \le \sum_{t=1}^{T} \left(f_t(w_t) - f_t(w_{t+1}) \right)$$

Proof: $\sum_{t=1}^{T} f_t(w_t) - \min_{w\in\Omega} \sum_{t=1}^{T} f_t(w)$

FTL

$$= \underbrace{\sum_{t=1}^{T} f_t(w_{t+1}) - \min_{w\in\Omega} \sum_{t=1}^{T} f_t(w)}_{\le 0} + \sum_{t=1}^{T} \left(f_t(w_t) - f_t(w_{t+1}) \right)$$

BTL

$$\le \sum_{t=1}^{T} \left(f_t(w_t) - f_t(w_{t+1}) \right) \qquad \square$$

control ! → w_t does not change much from round to round.

$\Rightarrow$ stable policy

$O(\sqrt{T})$

8. FTRL Algorithm

Follow-The-Regularized-Leader

def (Strongly-convex) We say a differentiable function $\phi: \Omega \to \mathbb{R}$ is α-strongly-convex w.r.t. the norm $\|\cdot\|$ on Ω:

$$\phi(x) \geq \phi(y) + \langle \nabla f(y), x-y \rangle + \frac{\alpha}{2} \cdot \|x-y\|^2 \quad \forall x, y \in \Omega$$

"quadratic"

$$\Updownarrow$$

$$\nabla^2 f(y) \succeq \alpha \cdot I$$

Regularization: 正则化

SVM: $\min \frac{1}{2}\|w\|^2 + \lambda \cdot \sum_{i=1}^{m} \xi_i^p$

Ridge Regression: $\min \|Xw-y\|_2^2 + \lambda \cdot \|w\|_2^2$

Neural Network: dropout.

add noise < input, weight, output.

early stopping

data augmentation.

⋮

Given a 1-strongly-convex function $\phi(\cdot)$, FTRL strategy:

$$w_t = \underset{w \in \Omega}{\operatorname{argmin}} \sum_{i=1}^{t-1} f_i(w) + \frac{1}{\eta} \cdot \phi(w)$$

regularizer

tuning parameter

Lemma (Regularizers induce stability) Let F and f be functions taking Ω into $\mathbb{R}$.

F: α-strongly-convex (w.r.t. $\|\cdot\|$)

f: convex

$$w = \underset{z \in \Omega}{\operatorname{argmin}} F(z)$$

$$w' = \underset{z \in \Omega}{\operatorname{argmin}} \left(F(z) + f(z)\right)$$

$$\Rightarrow 0 \leq f(w) - f(w') \leq \frac{1}{\alpha} \|\nabla f(w)\|_*^2$$

$$\boxed{\|y\|_* = \sup_{\|x\| \leq 1} \langle y, x \rangle}$$

Proof: By strong convexity of F: (Ω is a convex subset of $\mathbb{R}^d$)

$$F(w') - F(w) \geq \underbrace{\langle \nabla F(w), w'-w \rangle}_{0} + \frac{\alpha}{2} \|w'-w\|^2$$

$$\geq \frac{\alpha}{2} \|w'-w\|^2$$

$F+f$: α-strongly-convex $\quad \nabla^2(F+f) = \nabla^2 F + \nabla^2 f \succeq \alpha \cdot I$

$$\big(F(w)+f(w)\big) - \big(F(w')+f(w')\big) \geq \frac{\alpha}{2} \cdot \|w-w'\|^2$$

$$\Rightarrow \quad f(w) - f(w') \geq \alpha \cdot \|w-w'\|^2 \geq 0 \qquad \text{(LHS) } \checkmark$$

$$\Rightarrow \quad \|w-w'\| \leq \sqrt{\frac{1}{\alpha}\big(f(w)-f(w')\big)}$$

f convex: $f(w') \geq f(w) + \langle \nabla f(w), w'-w \rangle$

$$\Rightarrow \quad f(w) - f(w') \leq \langle \nabla f(w), w-w' \rangle$$

$$\leq \|\nabla f(w)\|_* \cdot \|w-w'\|$$

Cauchy-Schwarz (Hölder)

$$\leq \|\nabla f(w)\|_* \cdot \sqrt{\frac{1}{\alpha}\big(f(w)-f(w')\big)}$$

$$\Rightarrow \quad f(w) - f(w') \leq \frac{1}{\alpha} \|\nabla f(w)\|_* \qquad \square$$

FTRL Algorithm: $\quad w_t = \underset{w \in \Omega}{\operatorname{argmin}} \left(\sum_{i=1}^{t-1} f_i(w) + \frac{1}{\eta} \phi(w) \right)$

Thm (Regret for FTRL) Let ϕ be a 1-strongly-convex regularizer w.r.t. norm $\|\cdot\|$.

$$\text{FTRL regret} = \sum_{t=1}^{T} f_t(w_t) - \min_{w \in \Omega} \sum_{t=1}^{T} f_t(w) \leq \frac{D}{\eta} + \eta \cdot \sum_{t=1}^{T} \|\nabla f_t(w_t)\|_*^2$$

regret bound

where $D = \max_{w \in \Omega} \phi(w) - \min_{w \in \Omega} \phi(w)$

Suppose that $\forall t, w$: uniform bound $\|\nabla f_t(w)\|_*^2 \leq G$.

$$\Rightarrow \text{FTRL regret} \leq \frac{D}{\eta} + \eta \cdot TG$$

AM-GM

By $\eta = \sqrt{D/TG}$:

$$\text{FTRL regret} \leq 2\sqrt{DTG} = O(\sqrt{T})$$

Proof: Let $f_0(x) \triangleq \frac{\phi(x)}{\eta}$ $\Rightarrow$ FTRL policy: $w_t = \underset{w\in\Omega}{\operatorname{argmin}} \sum_{i=0}^{t-1} f_i(w)$

By regret bound of FTL (start at $t=0$) FTL with additional "round" of play at time zero

$$\sum_{t=0}^{T} f_t(w_t) - \min_{w\in\Omega} \sum_{t=0}^{T} f_t(w) \le \sum_{t=0}^{T} \left(f_t(w_t) - f_t(w_{t+1})\right)$$

Let $F(w) = \sum_{i=0}^{t-1} f_i(w)$ is $\frac{1}{\eta}$-strongly-convex.

ϕ: 1-强凸

$\Rightarrow f_0$: $\frac{1}{\eta}$-强凸

$f(w) = f_t(w)$ $(t \ge 1)$

F minimizer $\to w_t$.

F+f minimizer $\to w_{t+1}$

by lemma, $f_t(w_t) - f_t(w_{t+1}) \le \eta \cdot \|\nabla f_t(w_t)\|_*^2$.

$$\Rightarrow \sum_{t=0}^{T} f_t(w_t) - \min_{w\in\Omega} \sum_{t=0}^{T} f_t(w) \le \eta \cdot \sum_{t=1}^{T} \|\nabla f_t(w_t)\|_*^2 + f_0(w_0) - f_0(w_1)$$

$$\text{LHS} = \sum_{t=0}^{T} f_t(w_t) - \min_{w\in\Omega} \sum_{t=0}^{T} f_t(w) = \sum_{t=0}^{T} f_t(w_t) - \sum_{t=0}^{T} f_t(w^*)$$

$w^* = \underset{w\in\Omega}{\operatorname{argmin}} \sum_{t=0}^{T} f_t(w)$

$$= f_0(w_0) - f_0(w^*) + \left(\sum_{t=1}^{T} f_t(w_t) - \sum_{t=1}^{T} f_t(w^*)\right)$$

$$\ge f_0(w_0) - f_0(w^*) + \left(\sum_{t=1}^{T} f_t(w_t) - \min_{w\in\Omega} \sum_{t=1}^{T} f_t(w)\right)$$

FTRL regret

$f_0 = \frac{\phi}{\eta}$

$$\sum_{t=1}^{T} f_t(w_t) - \min_{w\in\Omega} \sum_{t=1}^{T} f_t(w) \le f_0(w^*) - f_0(w_0) + \eta \cdot \sum_{t=1}^{T} \|\nabla f(w_t)\|_*^2 + f_0(w_0) - f_0(w_1)$$

$$= \eta \cdot \sum_{t=1}^{T} \|\nabla f(w_t)\|_*^2 + f_0(w^*) - f_0(w^1)$$

$D = \max\phi - \min\phi$

$$\le \eta \cdot \sum_{t=1}^{T} \|\nabla f(w_t)\|_*^2 + D/\eta \qquad \square$$

$$\text{FTRL Regret} = \sum_{t=1}^{T} f_t(w_t) - \min_{w\in\Omega} \sum_{t=1}^{T} f_t(w) \le \frac{D}{\eta} + \eta \sum_{t=1}^{T} \underbrace{\|\nabla f(w_t)\|_*^2}_{\le G}$$

Apply FTRL Algorithm

① Online Linear Regression : $\Omega = \{w : \|w\|_2 \le 1\}$ $\quad f_t(w) = \frac{1}{2}(y_t - w^T x_t)^2$

$\phi(w) = \frac{1}{2}\|w\|_2^2$ (1-strongly-convex) $\quad \|x_t\|_2 \le 1. \quad |y_t| \le 1$

$D = \max \phi(w) - \min \phi(w) = \frac{1}{2}$

$\|\nabla f(w_t)\|_*^2 = \|\nabla f(w_t)\|_2^2 = \|-(y_t - w^T x_t) x_t\|_2^2 \le \underbrace{|y_t - w^T x_t|^2}_{\le 2} \|x_t\|_2^2 \le 4 = G.$

Regret Bound $= \frac{D}{\eta} + \eta G T = \frac{1}{2\eta} + 4T\eta = 2\sqrt{2}T = O(\sqrt{T})$ $\quad (\eta = \sqrt{\frac{1}{8T}})$

$$\sum_{t=1}^{T} (y_t - w_t^T x_t)^2 - \min_{\|w\|_2 \le 1} \sum_{t=1}^{T} (y_t - w^T x_t)^2 \le O(\sqrt{T})$$

② Expert Problem $\quad \Omega = \Delta(N) = \{p \in \mathbb{R}^N : \|p\|_1 = 1. \ p \succeq 0\}$

$f_t(p) = \langle l_t, p \rangle \quad l_t \in \{0,1\}^N$

$\phi(p) = \frac{1}{2}\|p\|_2^2$ $\quad D = \max_{p \in \Delta(N)} \phi(p) - \min_{p \in \Delta(N)} \phi(p)$

$\le \max_{p \in \Delta(N)} \frac{1}{2}\|p\|_2^2 \le \max_{p \in \Delta(N)} \frac{1}{2}\|p\|_1^2 = \frac{1}{2}$

$\|x\|_2 \le \|x\|_1 \le \sqrt{n} \cdot \|x\|_2. \quad x \in \mathbb{R}^n.$

$\|\nabla_p f_t(p)\|_2^2 = \|l_t\|_2^2 \le N = G$ ✓

$\Rightarrow$ Regret Bound $O(\sqrt{GDT}) = O(\sqrt{NT})$

$\rightarrow$ optimal dependency on T, but not N!

$\phi(p) = -H(p) = \sum_{j=1}^{N} p(j) \log p(j)$: negative entropy. $\quad O(\sqrt{\log N \cdot T})$

Lemma. $\phi(p) = -H(p)$ is a 1-strongly convex function wrt ℓ_1-norm $\|\cdot\|_1$

Proof: We need to show $\phi(p) - \phi(q) - \langle \nabla\phi(q), p-q \rangle \geq \frac{1}{2}\|p-q\|_1^2$

$\forall p, q \in \Delta(N) \subseteq \mathbb{R}^N$

$$\nabla\phi(q) = \begin{pmatrix} \log q(1) + 1 \\ \vdots \\ \log q(N) + 1 \end{pmatrix}$$

$$\phi(p) - \phi(q) - \langle \nabla\phi(q), p-q \rangle$$

$$= \sum_{j=1}^{N} p(j)\log p(j) - \sum_{j=1}^{N} q(j)\log q(j) - \sum_{j=1}^{N} (1 + \log q(j)) \cdot (p(j) - q(j))$$

$$= \sum_{j=1}^{N} p(j)\log p(j) - \sum_{j=1}^{N} p(j)\log q(j) - \underbrace{\sum_{j=1}^{N} (p(j) - q(j))}_{1-1=0}$$

$$= \sum_{j=1}^{N} p(j)\log\frac{p(j)}{q(j)} = KL(p\|q) \geq \frac{1}{2}\|p-q\|_1 \quad \square$$

Total-Variance Distance

Pinsker's inequality

$D = \sup \phi(p) - \inf \phi(p) = \log N.$

$0 \leq \underbrace{H(p)}_{\sum_{j=1}^{N} p(j)\log p(j)} \leq \log N$

$G = \|\nabla f_t(p)\|_*^2 = \|\nabla f_t(p)\|_\infty^2$

$= \|\ell_t\|_\infty^2 \leq 1$

$\Rightarrow$ Regret Bound $O(\sqrt{DGT}) = O(\sqrt{\log N \cdot T})$

Different Regularizers would lead to different dependencies other than T.

$$p_t = \underset{p \in \Delta(N)}{\operatorname{argmin}} \sum_{i=1}^{t-1} \underbrace{f_i(p)}_{\langle g_i, p \rangle} - \frac{1}{\eta} H(p)$$

Can we get the exact analytical solution for p_t?

① call off-the-shelf cvx algorithm → 不高效

② KKT condition ⇒ solve equation system. → 不高效

Lemma (Gibbs variational principle) Let ν, μ be probability distribution on $[N]$

吉布斯变分原理

$$\boxed{\sup_{\nu}\left(\mathbb{E}_{\nu}[f] - KL(\nu \| \mu)\right) = \log \mathbb{E}_{\mu}[e^{f}]}$$

the optimal solution is attained at $\nu(x) \propto \mu(x) \cdot e^{f(x)}$

$$\mathbb{E}_{\nu}[f] = \mathbb{E}_{x\sim\nu}[f(x)] = \langle \nu, f \rangle$$

→ taking the supremum over distribution ν of a linear function plus the KL divergence as a regularizer will give us the same distribution as exponentiating f

μ: uniform distribution over $[N]$

Cor Let ν be a probability distribution. $\boxed{\mathbb{E}_{\nu}[f] - H(\nu)}$ is minimized at $\nu(x) \propto e^{-f(x)}$

Proof: $KL(\nu \| \mu) = \sum_{x} \nu(x) \cdot \log \frac{\nu(x)}{\mu(x)}$

$$= \log N + \sum_{x} \nu(x) \log \nu(x) = \log N + (-H(\nu))$$

$$\sup_{\nu}\left(\mathbb{E}_{\nu}[f] - KL(\nu \| \mu)\right) = \sup_{\nu}\left(\mathbb{E}_{\nu}[f] - \log N + H(\nu)\right)$$

$$= -\inf\left(\mathbb{E}_{\nu}[f] - H(\nu) + \log N\right) \quad (*)$$

$\Rightarrow \mathbb{E}_{\nu}[f] - H(\nu)$ is minimized when $\sup\left(\mathbb{E}_{\nu}[-f] - KL(\nu \| \mu)\right)$ takes the optimal value, i.e. $\nu(x) \propto e^{-f(x)}$. $\square$

$$p_t = \operatorname*{argmin}_{p \in \Delta(N)} \left(\sum_{i=1}^{t-1} f_i(p) - \frac{1}{\eta} H(p)\right) = \operatorname*{argmin}_{p \in \Delta(N)} \left(\left\langle \sum_{i=1}^{t-1} l_i, p \right\rangle - \frac{1}{\eta} H(p)\right)$$

$$= \operatorname*{argmin}_{p \in \Delta(N)} \left(\underbrace{\left\langle \eta \cdot \sum_{i=1}^{t-1} l_i, p \right\rangle}_{f} - \frac{1}{\eta} H(p)\right) \hookrightarrow \mathbb{E}_{p}[f]$$

let $f = \eta \sum_{i=1}^{t-1} l_i$

by Corollary, $P_t \propto \exp\left(-\eta \cdot \sum_{i=1}^{t-1} l_i\right)$ ☆

$$\Rightarrow \boxed{P_t = \frac{\exp(-\eta \cdot L_t(j))}{\sum_{k=1}^{N} \exp(-\eta \cdot L_t(k))}}$$

$L_t = \sum_{i=1}^{t-1} l_i$: historical loss

Softmax

Expert Problem ⇒ compute historical loss + softmax

Multiplicative Weights Update Algorithm ☆

At each time step t, for each expert $j \in [N]$:

更新 ① $\boxed{\tilde{P}_t(j) = P_{t-1}(j) \cdot \exp(-\eta \cdot l_{t-1}(j))} = \begin{cases} P_{t-1}(j) \cdot \exp(-\eta) & \text{错误} \\ P_{t-1}(j) & \text{正确} \end{cases}$

归一化 ② normalization : $P_t = \frac{\tilde{P}_t}{\|\tilde{P}_t\|_1}$

指数权重

AdaBoost

Exponential Weight Majority

9. Online Gradient Descent

loss function: linear loss (linear function of the params)

We can always assume f_t to be linear in OCO w.l.o.g.

△ Convex-to-Linear Reduction

the linear case is the hardest case

assume we have an algorithm $\mathcal{A}$ that solves the linear case

$\mathcal{A}'$:

for $t = 1, \cdots, T$:

the learner invokes $\mathcal{A}$ to get output action $w_t \in \Omega$.

the environment gives the learner the loss function $f_t(\cdot)$

the learner construct a linear function $g_t(w) = \langle \nabla f_t(w_t), w \rangle$

the learner feeds $g_t(\cdot)$ to algorithm $\mathcal{A}$ as the loss function.

☆

☆ local linear approximation

局部线性近似

Proposition. If a deterministic algorithm $\mathcal{A}$ has regret no more than $\gamma(T)$ for linear case for some function $\gamma(\cdot)$, then any algorithm $\mathcal{A}'$ stated above has regret no more than $\gamma(T)$ for convex functions.

遗憾界并会因线性改变而改变！

Proof: $\forall w \in \Omega$: regret of $\mathcal{A}$: $\sum_{t=1}^{T} g_t(w_t) - \sum_{t=1}^{T} g_t(w) \le \gamma(T)$

$$g_t(w) = \langle \nabla f_t(w_t), w \rangle$$

$$g_t(w_t) - g_t(w) = \langle \nabla f_t(w_t), w_t - w \rangle \ge f_t(w_t) - f_t(w) \quad \text{(Convexity)}$$

$$\Rightarrow \sum_{t=1}^{T} f_t(w_t) - \sum_{t=1}^{T} f_t(w) \le \gamma(T) \qquad \text{Regret of } \mathcal{A}' \le \gamma(T) \ \square$$

OGD (Online Gradient Descent) 在线梯度下降

FTRL + ℓ_2 regularization $\quad \phi(x) = \frac{1}{2}\|x\|_2^2$

online-to-linear reduction

Given any OCO, we first do online-to-linear reduction,

then use FTRL:

$$w_t = \arg\min_{w \in \Omega} \sum_{i=1}^{t-1} g_i(w) + \frac{1}{2\eta} \cdot \|w\|_2^2$$

$$= \arg\min_{w \in \Omega} \sum_{i=1}^{t-1} \langle \nabla f_i(w_i), w \rangle + \frac{1}{2\eta} \|w\|_2^2$$

Online GD:

$$= \Pi_\Omega\left(-\eta \cdot \sum_{i=1}^{t-1} \nabla f_i(w_i)\right)$$

投影

projection

?

$$\arg\min_{w \in \Omega} \langle a, w \rangle + \frac{1}{2\eta}\|w\|_2^2 = \arg\min_{w \in \Omega} \frac{1}{2\eta}\|w + \eta a\|_2^2 - \frac{\eta}{2}\|a\|_2^2$$

$$= \arg\min_{w \in \Omega} \|w + \eta a\|_2^2$$

$$= \arg\min_{w \in \Omega} \|w - (-\eta a)\|_2^2 = \Pi_\Omega(-\eta a)$$

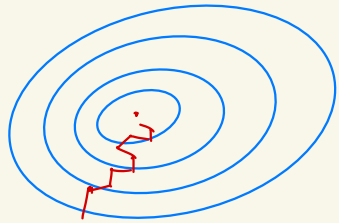
(def) $\Pi_\Omega(w) = \arg\min_{s \in \Omega} \|w - s\|^2$

"lazy projection": $u_t = u_{t-1} - \eta \cdot \nabla f_{t-1}(w_{t-1})$

$w_t = \Pi_\Omega(u_t)$

"eager projection": $u_t = w_{t-1} - \eta \cdot \nabla f_{t-1}(w_{t-1})$

$w_t = \Pi_\Omega(u_t)$



OGD: Regret Bound $O(\sqrt{T})$ by FTRL.

$$\boxed{w_{t+1} = \Pi_\Omega\left(w_t - \eta_t \cdot \nabla f_t(w_t)\right)}$$

Thm 1 (Regret Bound for OGD/OCO) Assume all f_t's are l-Lipschitz continuous,

Then for OCO, regret bound of OGD algorithm is $O(\sqrt{T})$

Proof: $w'_{t+1} = w_t - \eta_t \cdot \nabla f_t(w_t)$ $\quad \eta_t = \frac{\Gamma}{l\sqrt{t}}$ $\quad t \uparrow \eta_t \downarrow$ decay step size 待定

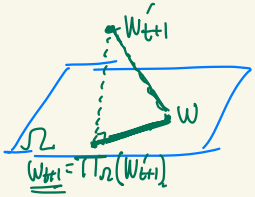
$\forall w \in \Omega \quad f_t(w_t) - f_t(w) \le \langle \nabla f_t(w_t), w_t - w \rangle$ (Convexity)

$= \frac{1}{\eta_t} \langle w_t - w'_{t+1}, w_t - w \rangle$

$2\|w_t\|^2 + \cancel{\|w\|^2} - 2\langle w_t, w\rangle - \cancel{\|w_{t+1}\|^2} - \cancel{\|w\|^2}$

$+ 2\langle w'_{t+1}, w\rangle + \cancel{\|w_t\|^2} + \cancel{\|w'_{t+1}\|^2} - 2\langle w_t, w'_{t+1}\rangle$

$= \frac{1}{2\eta_t}\left(\|w_t - w\|^2 - \|w'_{t+1} - w\|^2 + \|w_t - w'_{t+1}\|^2\right)$



$= \frac{1}{2\eta_t}\left(\|w_t - w\|^2 - \|w'_{t+1} - w\|^2\right) + \frac{\eta_t}{2} \cdot \|\nabla f_t(w_t)\|^2$

$\le \frac{1}{2\eta_t}\left(\|w_t - w\|^2 - \|w_{t+1} - w\|^2\right) + \frac{\eta_t}{2} \cdot \underbrace{\|\nabla f_t(w_t)\|^2}_{\le l}$

$|f(x) - f(x')| \le l \cdot \|x - x'\|$

$\Rightarrow \|\nabla f(x)\| \le l.$

$\le \frac{1}{2\eta_t}\left(\|w_t - w\|^2 - \|w_{t+1} - w\|^2\right) + \frac{\eta_t}{2} \cdot l^2$

$\Rightarrow$ Summation: $\sum_{t=1}^{T} f_t(w_t) - \sum_{t=1}^{T} f_t(w)$

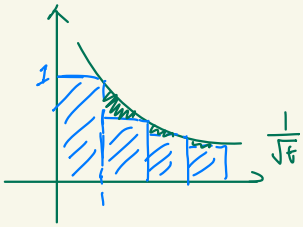
$\le \frac{1}{2\eta_1}\|w_1 - w\|^2 - \frac{1}{2\eta_T}\|w_{T+1} - w\|^2 + \frac{1}{2}\sum_{t=2}^{T}\left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right)\|w_t - w\|^2 + \frac{l^2}{2}\sum_{t=1}^{T}\eta_t$

let the diameter of Ω be Γ: $\forall u,v\in\Omega: \|u-v\|^2 \le \Gamma$

$\eta_t = \frac{\Gamma}{l\sqrt{t}}$, $\sum_{t=1}^{T} f_t(w_t) - \sum_{t=1}^{T} f_t(w) \le \frac{\Gamma^2}{2\eta_1} + \frac{\Gamma^2}{2}\cdot\sum_{t=2}^{T}\left(\frac{1}{\eta_t}-\frac{1}{\eta_{t-1}}\right) + \frac{l^2}{2}\sum_{t=1}^{T}\eta_t$

telescoping

$= \frac{\Gamma^2}{2}\cdot\frac{1}{\eta_T} + \frac{l^2}{2}\sum_{t=1}^{T}\eta_t$



$= \frac{\Gamma l\sqrt{T}}{2} + \frac{l\Gamma}{2}\cdot\sum_{t=1}^{T}\frac{1}{\sqrt{t}} \le \frac{3\Gamma l}{2}\cdot\sqrt{T} = O(\sqrt{T})$

$\le 1 + \int_1^T \frac{1}{\sqrt{t}}\,dt = 1 + 2\sqrt{t}\,\Big|_1^T = 2\sqrt{T}-1$

$\Rightarrow$ Regret $= \sum_{t=1}^{T} f_t(w_t) - \min_{w\in\Omega}\sum_{t=1}^{T} f_t(w) = O(\sqrt{T})$ □

Thm② Assume all f_t's are α-strongly-convex and l-Lipschitz.

Regret Bound for OCO / OGD is $O\left(\frac{\log T}{\alpha}\right)$ 在线强凸优化

Proof: $\forall w\in\Omega$: $f_t(w_t) - f_t(w) \le \langle \nabla f_t(w_t), w_t - w\rangle - \frac{\alpha}{2}\cdot\|w_t-w\|^2$

$\le \frac{1}{2\eta_t}\left(\|w_t-w\|^2 - \|w_{t+1}-w\|^2\right) + \frac{\eta_t}{2}\cdot l^2 - \frac{\alpha}{2}\cdot\|w_t-w\|^2$ by Strong convexity

$\Rightarrow \sum_{t=1}^{T} f_t(w_t) - \sum_{t=1}^{T} f_t(w) \le \frac{1}{2\eta_1}\cdot\|w_1-w\|^2 - \frac{1}{2\eta_T}\|w_{T+1}-w\|^2$

$+ \frac{1}{2}\sum_{t=2}^{T}\left(\frac{1}{\eta_t}-\frac{1}{\eta_{t-1}}\right)\cdot\|w_t-w\|^2 + \frac{l^2}{2}\sum_{t=1}^{T}\eta_t - \frac{1}{2}\sum_{t=1}^{T}\alpha\cdot\|w_t-w\|^2$

$\eta_t = \frac{1}{\alpha t}$

$\le \frac{\alpha}{2}\cdot\|w_1-w\|^2 - \cancel{\frac{T\alpha}{2}\|w_{T+1}-w\|^2} + \frac{1}{2}\sum_{t=2}^{T}\left(\underbrace{\alpha t - \alpha(t-1)}_{\alpha}\right)\cdot\|w_t-w\|^2$

$+ \frac{l^2}{2}\cdot\sum_{t=1}^{T}\frac{1}{\alpha t} - \frac{1}{2}\sum_{t=1}^{T}\alpha\|w_t-w\|^2$

$\le \frac{l^2}{2}\cdot\sum_{t=1}^{T}\frac{1}{\alpha t} = \frac{l^2}{2}\cdot\frac{1}{\alpha}\cdot\sum_{t=1}^{T}\frac{1}{t} = O\left(\frac{\log T}{\alpha}\right)$ □

凸优化: $O(\sqrt{T})$

强凸优化: $O(\log T)$

关于 η 的选取

10. Bandit Problem

Online Learning → OCO ← FTL, BTL, FTRL

① $w_t \in \Omega$.

② $f_t \to f_t(w_t)$

完全已知 entire function ✗

+ Online-to-linear → OGD

① Online Linear Regression

② Expert Problem

$O(\sqrt{T})$

完全信息在线学习 full-information online learning

< 赌博机在线学习 bandit online learning

we can only see the function value, not the function,

$f_t(w_t)$ $f_t(w)$

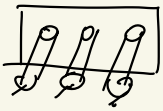
多臂问题 < Full-information → true answer

Bandit Problem → whether the prediction is true

不知道正确答案!

- stochastic
- oblivious ← FTRL
- adaptive (non-oblivious)

1. Multi-Armed Bandit (MAB) 多臂赌博机



K arm (摇臂) ⇔ K experts

During each iteration, the learner needs to select one out of K arms to pull and then receive the corresponding reward 收益 → loss 损失

Goal: maximize the cumulative reward

Stochastic: the reward of each arm is i.i.d

μ_i: 第i个摇臂的收益均值. i_t: 学习者在第t轮选择的摇臂.

$$\text{Regret} = T \max_{i\in[K]} \mu_i - \sum_{t=1}^{T} \mu_{i_t} \quad \text{in Bandit Problem.}$$

expected regret

尝试不同的摇臂　　选择最大收益的摇臂

partial informative → exploration v.s. exploitation

探索　　利用

★ a s in Reinforcement Learning (Q-learning)

① ε-Greedy　random number $r \in [0,1]$ $\begin{cases} r < \varepsilon: & \text{exploration} \\ r \geq \varepsilon: & \text{exploitation} \end{cases}$

② softmax $\Pr(a \mid s) = \dfrac{e^{Q(s,a)/\tau}}{\sum_{a'} e^{Q(s,a')/\tau}}$

Knowledge distillation

τ: temperature parameter

τ → 大: exploration; 小: exploitation

UCB (Upper Confidence Bound) 置信上界

*ELBO (Evidence Lower Bound)

Key idea: maintain an UCB $\hat{\mu}_i$ for each arm i.

which ensures the mean reward $\mu_i \leq \hat{\mu}_i$ w.h.p

对 μ_i 的上界的估计

The algorithm selects the arm with highest UCB.

↳ balance exploration & exploitation automatically

sample mean v.s mean

$[n\mu - t, n\mu + t]$

Concentration inequality

△ Hoeffding's inequality:

$X_1, \cdots, X_n \in [0,1]$. $\mu(X_i) = \mu$.

$$\begin{cases} \Pr\left(\sum_{i=1}^{n} X_i \geq n\mu + t\right) \leq \exp\left(-\frac{2t^2}{n}\right) \\ \Pr\left(\sum_{i=1}^{n} X_i \leq n\mu - t\right) \leq \exp\left(-\frac{2t^2}{n}\right) \end{cases}$$

假设第i个摇臂奖励在[0,1]之间取值 且选择了摇臂 n_i 次.

样本均值 $\bar{\mu}_i$

$$\text{UCB}: \hat{\mu}_i = \bar{\mu}_i + \sqrt{\frac{2\log\alpha}{n_i}}$$

$$\sum_{j=1}^{n} X_j \geq n_i\bar{\mu}_i + \sqrt{2n_i\log\alpha}.$$

令 $t=\sqrt{2n_i\log\alpha}$, $n=n_i$ $\Rightarrow \delta = \exp\left(-\frac{2t^2}{n}\right) = \exp\left(-\frac{4n_i\log\alpha}{n_i}\right) = \alpha^{-4}$

$\Rightarrow$ w.p. $\geq 1-\alpha^{-4}$ $\quad \mu_i \leq \hat{\mu}_i$.

UCB $\hat{\mu}_i$ $<$
- $\bar{\mu}_i$: sample mean $\rightarrow$ learner's current knowledge
- $\sqrt{\frac{2\log\alpha}{n}}$: interval length. $\rightarrow$ uncertainty of the estimate.

$\hat{\mu}_i \rightarrow$ 大 $\rightarrow$
- sample mean ↑ $\Rightarrow$ exploitation
- interval length ↑ $\Rightarrow$ exploration

tradeoff

MAB+UCB Algorithm:

for $t=1$ to K do:

the learner pulls the t-th arm

the learner observes the reward X_t^t

纯探索 (每个手臂拉一次)

X_i^t: 第t轮拉动手臂i得到的回报

initialize the # of times each arm has been selected $n_i^k = 1$, $i=1,\cdots,K$.

initialize the sample mean of each arm: $\bar{\mu}_i(n_i^k) = X_i^i$, $i=1,\cdots,K$.

---

for $t=K+1$ to T do:

the learner pulls the arm.

n_i^t: 截止到第t轮时手臂i被选择的次数

$\bar{\mu}_i(n)$: 手臂i拉动n次后得到的样本均值

☆ $$i_t = \arg\max_{i\in[K]} \bar{\mu}_i(n_i^{t-1}) + \sqrt{\frac{2\log(t-1)}{n_i^{t-1}}} \quad (*)$$

基于UCB选择摇臂

the learner observe the reward $X_{i_t}^t$

update

update the # of times each arm has been pulled:

$$n_{i_t}^t = n_{i_t}^{t-1} + 1 \quad ; \quad n_j^t = n_j^{t-1}, \quad j \neq i_t.$$

update the same mean of the arm i_t:

$$\bar{\mu}_{i_t}(n_{i_t}^t) = \frac{n_{i_t}^{t-1} \times \bar{\mu}_{i_t}(n_{i_t}^{t-1}) + X_{i_t}^t}{n_{i_t}^t}$$

Thm1 (Regret Bound of stochastic MAB) Assume that reward of each arm falls within $[0,1]$, and the reward of each arm are i.i.d (i.e. stochastic setting) then MAB + UCB algorithm satisfies

$$\text{Regret} = T \max_i \mu_i - \sum_{t=1}^{T} \mu_{i_t}$$

$$\mathbb{E}[\text{regret}] = T \cdot \max_{i \in [k]} \mu_i - \mathbb{E}\left[\sum_{t=1}^{T} \mu_{i_t}\right] = O(k \log T)$$

对算法执行过程中可能出现的所有动作路径 $i_1, \cdots, i_T$ 取期望

手臂个数

$*$ $O(k \cdot \log T)$ w.h.p.

$$\mathbb{E}[\text{regret}] \leq 8 \cdot \sum_{i \neq *} \frac{\log T}{\Delta_i} + \left(1 + \frac{\pi^2}{3}\right) \sum_{i \neq *} \Delta_i$$

$* = \arg\max_{i \in [k]} \mu_i$ $\quad \Delta_i = \mu_* - \mu_i$

最优手臂

Claim: ① $A \Rightarrow B$: $\mathbb{1}(A) \leq \mathbb{1}(B)$

② $\mathbb{1}(A_1 \text{ or } A_2 \text{ or} \cdots \text{or } A_n) \leq \mathbb{1}(A_1) + \mathbb{1}(A_2) + \cdots + \mathbb{1}(A_n)$

③ $\mathbb{1}(A) \leq \mathbb{1}(A, B) + \mathbb{1}(A, \neg B)$

Proof of Thm1. $\text{Regret} = T \max_i \mu_i - \sum_{t=1}^{T} \mu_{i_t} = T \cdot \mu_* - \sum_{t=1}^{T} \mu_{i_t}$

$$= \sum_{i \neq *} (\mu_* - \mu_i) \cdot n_i^T$$

T轮中第i个手臂被拉动的次数

$$= \sum_{i \neq *} \Delta_i n_i^T$$

$$n_i^T = 1 + \sum_{t=k+1}^{T} \mathbb{1}(i_t = i)$$

待定参数

$$\overset{③}{\leq} 1 + \sum_{t=k+1}^{T} \mathbb{1}(i_t = i,\ n_i^{t-1} < l) + \sum_{t=k+1}^{T} \mathbb{1}(i_t = i,\ n_i^{t-1} \geq l)$$

$$\leq 1 + (l-1) + \sum_{t=k+1}^{T} \mathbb{1}(i_t = i,\ n_i^{t-1} \geq l)$$

$(*)$: $i_t = i \Rightarrow \bar{\mu}_*(n_*^{t-1}) + \sqrt{\frac{2\log(t-1)}{n_*^{t-1}}} \leq \bar{\mu}_i(n_i^{t-1}) + \sqrt{\frac{2\log(t-1)}{n_i^{t-1}}}$

UCB

$$\mathbb{1}(i_t = i,\ n_i^{t-1} \geq l)$$

$$\overset{①}{\leq} \mathbb{1}\left(\bar{\mu}_*(n_*^{t-1}) + \sqrt{\frac{2\log(t-1)}{n_*^{t-1}}} \leq \bar{\mu}_i(n_i^{t-1}) + \sqrt{\frac{2\log(t-1)}{n_i^{t-1}}},\ n_i^{t-1} \geq l\right)$$

$$\overset{①}{\leq} \mathbb{1}\left(\min_{0<p<t} \bar{\mu}_*(p) + \sqrt{\frac{2\log(t-1)}{p}} \leq \max_{l\leq q<t} \bar{\mu}_i(q) + \sqrt{\frac{2\log(t-1)}{q}}\right)$$

$$\overset{②}{\leq} \sum_{p=1}^{t-1} \sum_{q=l}^{t-1} \mathbb{1}\left(\bar{\mu}_*(p) + \sqrt{\frac{2\log(t-1)}{p}} \leq \bar{\mu}_i(q) + \sqrt{\frac{2\log(t-1)}{q}}\right)$$

$$n_i^T \leq l + \sum_{t=k+1}^{T} \sum_{p=1}^{t-1} \sum_{q=l}^{t-1} \mathbb{1}\left(\bar{\mu}_*(p) + \sqrt{\frac{2\log(t-1)}{p}} \leq \bar{\mu}_i(q) + \sqrt{\frac{2\log(t-1)}{q}}\right)$$

$$\leq l + \sum_{t=1}^{T-1} \sum_{p=1}^{t-1} \sum_{q=l}^{t-1} \mathbb{1}\left(\bar{\mu}_*(p) + \sqrt{\frac{2\log t}{p}} \leq \bar{\mu}_i(q) + \sqrt{\frac{2\log t}{q}}\right)$$

$$\bar{\mu}_*(p) + \sqrt{\frac{2\log t}{p}} \leq \bar{\mu}_i(p) + \sqrt{\frac{2\log t}{p}}$$

sample mean $\Longleftrightarrow$ Mean

$$\Longleftrightarrow \bar{\mu}_*(p) + \sqrt{\frac{2\log t}{p}} \leq \mu_*. \quad \text{or.} \quad \mu_* \leq \mu_i + 2\sqrt{\frac{2\log t}{q}} \quad \text{or.} \quad \mu_i + \sqrt{\frac{2\log t}{q}} \leq \bar{\mu}_i(q)$$

otherwise: $\bar{\mu}_*(p) + \sqrt{\frac{2\log t}{p}} \geq \mu_* \geq \mu_i + 2\sqrt{\frac{2\log t}{p}} \geq \bar{\mu}_i(q) + \sqrt{\frac{2\log t}{q}}$.

contradictory!

$$n_i^T \le l + \sum_{t=1}^{T-1}\sum_{p=1}^{t-1}\sum_{q=l}^{t-1}\left(\mathbb{1}\left(\bar{\mu}_*(p)+\sqrt{\frac{2\log t}{p}} \le \mu_*\right) + \mathbb{1}\left(\mu_* \le \mu_i + 2\sqrt{\frac{2\log t}{q}}\right) + \mathbb{1}\left(\mu_i + \sqrt{\frac{2\log t}{q}} \le \bar{\mu}_i(q)\right)\right)$$

$$\mathbb{E}[n_i^T] \le l + \sum_{t=1}^{T-1}\sum_{p=1}^{t-1}\sum_{q=l}^{t-1}\left(\underbrace{\Pr\left(\bar{\mu}_*(p)+\sqrt{\frac{2\log t}{p}} \le \mu_*\right)}_{2)\ \le t^{-4}} + \Pr\left(\mu_* \le \mu_i + 2\sqrt{\frac{2\log t}{q}}\right) + \underbrace{\Pr\left(\mu_i + \sqrt{\frac{2\log t}{q}} \le \bar{\mu}_i(q)\right)}_{1)\ \le t^{-4}}\right)$$

Hoeffding's inequality: $\Pr\left(\sum_{i=1}^{n} X_i \ge n\mu + t\right) \le \exp\left(-\frac{2t^2}{n}\right)$ 1)

$\Pr\left(\sum_{i=1}^{n} X_i \le n\mu - t\right) \le \exp\left(-\frac{2t^2}{n}\right)$ 2)

$$\le l + \sum_{t=1}^{T-1}\sum_{p=1}^{t-1}\sum_{q=l}^{t-1}\left(2t^{-4} + \Pr\left(\mu_* \le \mu_i + 2\sqrt{\frac{2\log t}{q}}\right)\right)$$

预定事件

$\Delta_i = \mu_* - \mu_i \to$ 常数 $\qquad q \ge l$

$$l \le \frac{8\log t}{\Delta_i^2} \Leftarrow \Delta_i = \mu_* - \mu_i \le 2\sqrt{\frac{2\log t}{q}} \le 2\sqrt{\frac{2\log t}{l}} = \sqrt{\frac{8\log t}{l}}$$

Let $l = \left\lceil \frac{8\log t}{\Delta_i^2} \right\rceil$: $\Pr\left(\mu_* \le \mu_i + 2\sqrt{\frac{2\log t}{q}}\right) = 0, \quad q \ge l$

$$\mathbb{E}[n_i^T] \le \left\lceil \frac{8\log t}{\Delta_i^2} \right\rceil + 2\cdot\sum_{t=1}^{T-1}\underbrace{\sum_{p=1}^{t-1}}_{O(t)}\underbrace{\sum_{q=l}^{t-1}}_{O(t)} t^{-4}$$

$$\le \frac{8\log t}{\Delta_i^2} + 1 + 2\cdot\sum_{t=1}^{\infty} t^{-2} \le \frac{8\log t}{\Delta_i^2} + 1 + \frac{\pi^2}{3} \qquad \left(\sum_{t=1}^{\infty}\frac{1}{t^2} = \frac{\pi^2}{6}\right)$$

$$\mathbb{E}[\text{Regret}] = \sum_{i \ne *} \Delta_i \cdot \mathbb{E}[n_i^T] \le 8\cdot\sum_{i \ne *}\frac{\log T}{\Delta_i^2} + \left(1+\frac{\pi^2}{3}\right)\cdot\sum_{i \ne *}\Delta_i$$

$$= O(K\log T) \qquad \square$$

手臂个数

MAB+UCB

recommendation system 商品推荐

Arm 手臂 : 各个手臂之间相互独立 没有关联的 → 线性赌徒

⇓

寻找connection

reward 商品d维向量 ⇒ 手臂: d维向量

→ 可以表示成该向量的函数

2. Linear Bandit 线性赌博机

手臂 x 的奖励的均值: $\mu_x = x^T w^*$. $\quad x, w^* \in \mathbb{R}^d$

different arms share the same set of params w^*.

$\Downarrow$

build the relationship between arms

reward: $y = x^T w^* + \varepsilon$.

random noise (with zero mean)

$\mathcal{X} \subseteq \mathbb{R}^d$: the set of arms

$$\text{regret} = T \cdot \max_{x \in \mathcal{X}} x^T w^* - \sum_{t=1}^{T} x_t^T w^*$$

expected regret

如果能 w.h.p $w^* \in \mathcal{R}$ (confidence region)

则对于每一个手臂 x, 可以构造 $\boxed{UCB = \max_{w \in \mathcal{R}_t} x^T w \triangleq \hat{\mu}_x}$

且 w.h.p $\mu_x = x^T w^* \le \hat{\mu}_x$

算法可以根据 UCB $\hat{\mu}_x$ 来选择最优手臂, 得到奖励后更新置信区域

① 估计 w^* (w_t) ② 构造 w^* 的置信区域

① 估计 w^*: 前 t 轮, 选择手臂 $x_1, \cdots, x_t$, 观察奖励 $y_1, \cdots, y_t$.

$y_t = x_t^T w^* + \varepsilon_t$ $\quad$ ~~Linear~~ Ridge Regression

第 t 轮对 w^* 估计: $w_t = \arg\min_{w \in \mathbb{R}^d} \sum_{i=1}^{t} (y_i - x_i^T w)^2 + \lambda \|w\|_2^2$.

$$\Rightarrow w_t = \left(\lambda I + \sum_{i=1}^{t} x_i x_i^T\right)^{-1} \left(\sum_{i=1}^{t} y_i x_i\right)$$

(closed-form solution)

需要保存前 t 轮所有 (x_i, y_i)

by Sherman-Morrison-Woodbury: $(A + UV^T)^{-1} = A^{-1} - A^{-1}U(I + V^T A^{-1} U)^{-1} V^T A^{-1}$.

① 在线计算 w_t:

$$w_t = \left(Z_{t-1}^{-1} - \frac{Z_{t-1}^{-1} x_t x_t^T Z_{t-1}^{-1}}{1 + x_t^T Z_{t-1}^{-1} x_t} \right) \cdot \left(q_{t-1} + y_t x_t \right)$$

$$= w_{t-1} + \left(y_t - \frac{x_t^T Z_{t-1}^{-1} q_t}{1 + x_t^T Z_{t-1}^{-1} x_t} \right) Z_{t-1}^{-1} x_t$$

$$Z_{t-1} = \lambda I + \sum_{i=1}^{t-1} x_i x_i^T \longrightarrow Z_t^{-1} = Z_{t-1}^{-1} - \frac{Z_{t-1}^{-1} x_t x_t^T Z_{t-1}^{-1}}{1 + x_t^T Z_{t-1}^{-1} x_t}$$

(iterative solution)

$$q_{t-1} = \sum_{i=1}^{t-1} y_i x_i \longrightarrow q_t = q_{t-1} + y_t x_t$$

(∘)

② 构造 w_t^* 的置信区域 → concentration inequality!

Lemma $y_t = x_t^T w^* + \varepsilon_t$, where ε_t { mean: zero; conditional μ-sub-Gaussian }

$\left(\mathbb{E}_t[e^{\lambda \varepsilon_t}] \le \exp\left(\frac{\lambda^2 \mu^2}{2} \right) \right)$

Assume $\|w^*\| \le \Lambda$, $\|x_t\| \le r$, $\forall t \in [T]$.

w.p. $\ge 1-\delta$:

$$w^* \in \mathcal{R}_t = \left\{ w \in \mathbb{R}^d \;\middle|\; \|w - w_t\|_{Z_t} \le \mu \cdot \sqrt{2 \log \frac{1}{\delta} + d \log\left(1 + \frac{t \cdot r^2}{\lambda d}\right)} + \Lambda \sqrt{\lambda} \right\}$$

(△)

$Z_t = \lambda I + \sum_{i=1}^{t} x_i x_i^T$ PD

$Z_t \succ 0$: $\|x\|_{Z_t} = \sqrt{x^T Z_t x}$.

Linear Bandit + LUCB algorithm: ☆

for $t = 1$ to T do:

the learner pulls the arm by $\boxed{x_t = \arg\max_{x \in \mathcal{X}} \left(\max_{w \in \mathcal{R}_{t-1}} x^T w \right)}$ ← LUCB

the learner observes the reward y_t.

update w_t by (∘)

update $\mathcal{R}_t$ by (△)

how to optimization?

$\mathcal{X}$ infinite: NP-Hard!

consider special forms of $\mathcal{X}$ (sphere)

Thm② (Regret Bound of Linear Bandit + LUCB)

Suppose the assumption in Lemma holds.

w.h.p $\quad \text{Regret} = T \max_{x \in \mathcal{X}} x^T w^* - \sum_{t=1}^{T} x_t^T w^* = \tilde{O}(d\sqrt{T})$