Online Learning: Partial Information and Bandits

CS6780 – Advanced Machine Learning
Spring 2019

Thorsten Joachims
Cornell University

Reading:
http://jeremykun.com/2013/10/28/optimism-in-the-face-of-uncertainty-the-ucb1-algorithm/
http://jeremykun.com/2013/11/08/adversarial-bandits-and-the-exp3-algorithm/

Bandit Learning Model

- Setting
  - N arms named $\mathrm{H} = \{h_1, \dots, h_N\}$
  - In each round t, each arm h_i performs an action and incurs loss $\Delta_{t,i}$
  - Algorithm can select which arm to pull in each round
- Interaction Model
  - FOR t from 1 to T
    - Algorithm selects arm h_{i_t} according to strategy A_{w_t} and follows its action y
    - Arms incur losses $\Delta_{t,1} \dots \Delta_{t,N}$ (all but Δ_{t,i_t} unobserved)
    - Algorithm observes and incurs loss Δ_{t,i_t}
    - Algorithm updates w_t to w_{t+1} based on Δ_{t,i_t}

Key difference compared to Expert Model

Exponentiated Gradient Algorithm for Bandit Setting (EXP3)

- Initialize $w_1 = \left(\frac{1}{N}, \dots, \frac{1}{N}\right), \gamma = \min\left\{1, \sqrt{\frac{N \log N}{(e-1)\Delta \mathrm{T}}}\right\}$
- FOR t from 1 to T
  - Algorithm randomly picks i_t with probability $P_t(i_t) = (1-\gamma) w_{t,i} + \gamma / N$
  - Arms incur losses $\Delta_{t,1} \dots \Delta_{t,N}$
  - Algorithm observes and incurs loss Δ_{t,i_t}
  - Algorithm updates w for bandit i_t as $w_{t+1,i_t} = w_{t,i_t} \exp\left(-\eta \Delta_{t,i_t} / P(i_t)\right)$
    Then normalize w_{t+1} so that $\sum_j w_{t+1,j} = 1$.

Adversarial Bandit Regret

- Idea
  - Compare performance to best arm in hindsight
- Regret
  - Overall loss of best arm i^* in hindsight is

$$\Delta_T^* = \min_{i^* \in [1..N]} \sum_{t=1}^{T} \Delta_{t,i^*}$$

  - Expected loss of algorithm A over sequence of arm selections i_t is

$$E_A\left[\sum_{t=1}^{T} \Delta_{t,i_t}\right]$$

  - Regret is difference between expected loss of algorithm and best fixed arm in hindsight

$$ExpectedRegret(T) = E_A\left[\sum_{t=1}^{T} \Delta_{t,i_t}\right] - \min_{i^* \in [1..N]} \sum_{t=1}^{T} \Delta_{t,i^*}$$

EXP3 Regret Bound

- Theorem: For $\gamma \in]0,1]$ and stopping time T EXP3 has expected regret of at most

$$ERegret(T) \le (e-1)\gamma \left(\min_i \sum_{t=1}^{T} \Delta_{t,i}\right) + \frac{N \log N}{\gamma}$$

- Corollary: For $\Delta_{t,i} \le \Delta$, EXP3 with γ as on previous slide has expected regret of at most $ERegret(T) \le 2.63\sqrt{\Delta T N log N}$.

Stochastic Bandit Learning Model

Key difference compared to Adversarial Bandit Model

- Setting
  - N arms named $\mathrm{H} = \{h_1, \dots, h_N\}$
  - In each round t, each arm h_i performs an action and incurs loss $\Delta_{t,i}$ drawn from fixed distribution $P(\Delta|i)$ with mean μ_i.
  - Algorithm can select which ar
- Interaction Model
  - FOR t from 1 to T
    - Algorithm selects arm h_{i_t} accor action y
    - Arms incur losses $\Delta_{t,1} \dots \Delta_{t,N}$ (al
    - Algorithm observes and incurs l
    - Algorithm updates w_t to w_{t+1} b

Stochastic Bandit Regret

- Idea
  - Compare performance to arm with best expected performance
- Regret
  - Overall loss of best arm i^* is
$$\Delta_T^* = T \min_{i \in [1..N]} \mu_i = T\mu_{i^*}$$
  - Expected loss of algorithm A over sequence of arm selections i_t is
$$E_A\left[\sum_{t=1}^{T} \Delta_{t,i_t}\right]$$
  - Regret is difference between expected loss of algorithm and best fixed arm in hindsight
$$ExpectedRegret(T) = E_A\left[\sum_{t=1}^{T} \Delta_{t,i_t}\right] - T\mu_{i^*}$$

UCB1 Algorithm

- Init:
  - Play each arm i once to get initial values for $w_1 \dots w_N$.
  - $n = (1, \dots, 1)$
- For t from $(N+1)$ to T
  - Play arm $i_t = argmin_i \left\{\frac{w_i}{n_i} - \sqrt{2 \log \frac{T}{n_i}}\right\}$
  - Algorithm observes and incurs loss Δ_{t,i_t}
  - $w_i = w_i + \Delta_{t,i_t}$
  - $n_i = n_i + 1$

UCB1 Regret Bound

- Theorem: The expected regret of UCB1 is at most
$$O\left(\sum_{i \neq i^*} \frac{\log T}{\epsilon_i}\right)$$
where i^* is the best arm and $\epsilon_i = \mu_{i^*} - \mu_i$.

Other Online Learning Problems

- Contextual Bandits
- Dueling Bandits
- Coactive Learning
- Online Convex Optimization
- Partial Monitoring