Survey of Different Varients of Thompson Sampling

IE 613 Course Project

Vinay Chourasiya
Karan Patel
Manoj Kumar
Under the guidance of Prof. Manjesh Hanawal

Indian Institute of Technology Bombay

May 6, 2018

Outline

Introduction

- In artificial intelligence, Thompson sampling, named after William R. Thompson, is a heuristic for choosing actions that addresses the exploration-exploitation dilemma in the multi-armed bandit (MAB) problem.
- The multi-armed bandit problem is a problem in which a fixed limited set of resources must be allocated between alternative choices in a way that maximizes their expected gain.
- The name comes from imagining a gambler at a row of slot machines, who has to decide:
  - Which machines to play?
  - How many times to play each machine?
  - In which order to play them? and
  - Whether to continue the current machine or try a different machine.
- In the problem, each machine provides a reward associated with them from some stochastic process. The objective of the gambler is to maximize the sum of rewards.

Figure 1 : Slot machines

- The crucial trade-off that the gambler faces at each trial is between "exploitation" of the machine that has the highest expected payoff and "exploration" to get more information about the expected payoffs of the other machines.

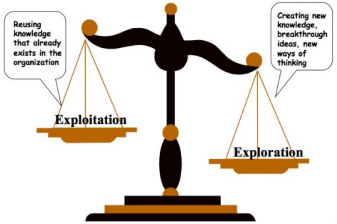


Figure 2 : Meaning of exploitation and exploration

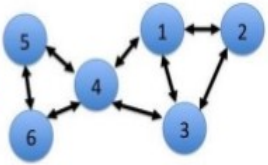
- In exploitation there is chance that we can loose optimal arm.
- In exploration we always explore hindsight arm in search of optimal arm the optimal arm.

Applications of MAB model

Clinical Trials

Ad Placement

Adaptive Routing



Stock Investment

Description of Thompson Sampling

- Consider a set of actions A and rewards in $\mathbb{R}$.
- In each round player plays an action $a \in A$ based on some prior distribution and receives a reward $r \in \mathbb{R}$ according to some stochastic distribution associated with action played.
- Aim of the player is to maximize cumulative rewards by playing appropriate action in each round.
- As, the posterior distribution is updated after every round. Hence, the rule of playing optimal action $a^* \in A$ is implemented by sampling using updated posterior distribution.

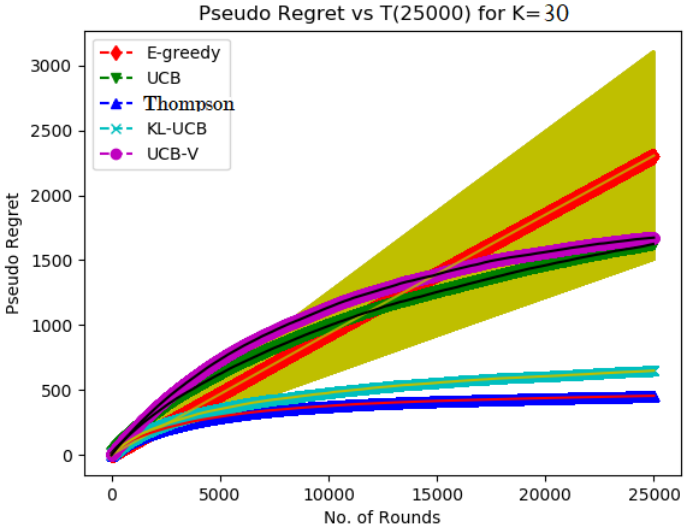
Thompson Sampling for Bernoulli Bandit setting

- Suppose there are K arm, and when any arm among them is played it yields either a success (reward $= 1$) or failure (reward $= 0$).
- Let any arm $\in \{1, 2, ..., K\}$ produces a reward of one with probability $\mu_k \in [0, 1]$ and a reward of zero with probability 1-μ_k.
- The success probabilities or mean rewards corresponding to different arms $\mu = (\mu_1, \mu_2..., \mu_k)$ may be different and are unknown, but fixed over time. Hence it can be learned by experimentation.
- Let in the first round, an arm k_1 is pulled and a reward $r_1 \in \{0, 1\}$ is generated with success probability $\mathbb{P}(r_1 = 1|k_1, \mu) = \mu_{k_1}$. After observing r_1, the agent will play another arm k_2, observes reward r_2, and this process continues.

- Also let player begin with an independent prior belief over each μ_k.
- In Bernoulli setting case, we consider these priors as beta-distributed with parameters $S = (S_1, S_2, ..., S_k)$ and $F = (F_1, F_2, ..., F_k)$. Where S is the number of success and F is Number of failure.
- Beta distribution is convenient because of it is an conjugate distribution. Hence each arm's posterior distribution will also be same as prior but with updated parameters.
- The update rule is for selected arm k: $(S_k, F_k) = \begin{cases} (S_k, F_k + 1), & \text{if } r_t = 0 \\ (S_k + 1, F_k), & \text{if } r_t = 1 \end{cases}$

Algorithm 1 Thomson Sampling with Bernoulli setting

Input: No. of arms (K), $S_k(1) = 0 \text{and} F_k(1) = 0$.
for t= 1,2,...**do**:
    for k = 1, 2,...,K **do**:
        Sample $\theta_i(t) \sim \text{Beta}(S_k + 1, F_k + 1)$
    end for
    Sample and select arm:
    $k_t \leftarrow argmax_k\ \theta_i(t)$
    Select arm k_t and observe reward r_t
    Update posterior distribution:
    If $r_t = 1$,
        $S_{k_t} = S_{k_t} + 1$
    else
        $F_{k_t} = F_{k_t} + 1$
end for


Pseudo Regret vs T(25000) for K= 30
E-greedy
UCB
Thompson
KL-UCB
UCB-V
Pseudo Regret
No. of Rounds
0
500
1000
1500
2000
2500
3000
0
5000
10000
15000
20000
25000

Thompson Sampling for General Setting

- Beyond Bernoulli setting, Thompson sampling can be applied successfully to many online decision problems.
- Lets understand genaral setting case:
- Let there is a finite set of arms K and the player pulls arms $k_1, k_2, \ldots$.
- After pulling arm k_t in t^{th} round the player observe an outcome y_t which is randomly generated by system according to some conditional probability measure.

- The player enjoys a reward of $\tilde{r}_t = r(y_t)$, where r is a known function also let $\tilde{r}_t$ is such that its value lies in interval [0,1].
- Initially, the player is uncertain about reward and represents his uncertainty by using a prior distribution (here also taken as Beta distribution).
- Now, Bernoulli trial is performed with success probability equal to observed reward ($\tilde{r}_t$) and output is observed in terms of 0 or 1, and posterior distribution is updated accordingly (same as in case of Bernoulli setting).

Algorithm 2 Thomson Sampling with General setting

Input: No. of arms (K), $S_k(1) = 0 \text{and} F_k(1) = 0$.
for t= 1,2,...**do**:
  for $k = 1, 2,...,$K **do**:
    Sample $\theta_i(t) \sim$ Beta($S_k + 1$,$F_k + 1$) distribution.
  end for
  Sample and select arm:
  $k_t \leftarrow argmax_k\ \theta_i(t)$
  Select arm k_t and observe reward $\tilde{r}_t$.
  Now by taking success probability as $\tilde{r}_t$ and get output r_t
  by using Bernoulli distribution.
  Update posterior distribution:
  If $r_t = 1$,
    $S_{k_t} = S_{k_t} + 1$
  else
    $F_{k_t} = F_{k_t} + 1$
end for

Contextual MAB

Contextual Multi Arms Bandits

1. Each arm has the side information, called context of arm.
2. On the basis of context and history player has to decide which arm play in current round.

Example : News Article Recommendation

News article pages
Different article has different appearance and news

User searching news (with some interest, he Click on some news)

News provider wants the more click from user, so he display news on the basis of user interest

Contextual MAB

Contextual MAB setting

There are N arm, at time t context vector associated with arm i is $b_i(t) \in R^d$. for all arm context vector $b_i(t)$ revealed , using context vector $b_i(t)$, and History of all previous round $1, 2..t-1$, player has to decide which arm choose for round t.Reward generate through unknown distribution with mean $b_i(t)^T \mu$ for each arm i. Given $b_i(t)$ at round t, the reward of arm i generate from an unknown distribution with mean $b_i(t)^T \mu$, where $\mu \in R^d$ is parameter, that is learn by the algorithm.let $a^*(t)$ denote optimal arm and $a(t)$ is the arm choose by the algorithm at time t,

$$a^*(t) = \arg\max_i (b_i(t)\mu)$$

Thompson Sampling

Define,

$$B(t) = I_d + \sum_{\tau=1}^{t-1} b_{a(\tau)}(\tau) b_{a(\tau)}(\tau)^T$$

$$\hat{\mu}(t) = B(t)^{-1} \left(\sum_{\tau=1}^{t-1} b_{a(\tau)}(\tau) b_{a(\tau)}(\tau)^T \right)$$

In Thompson sampling algorithm we sample $\tilde{\mu}(t)$ from distribution $N(b_i(t)^T \hat{\mu}, v^2)$ and play arm $a(\tau)$ s.t.

$$a(\tau) = \arg\max_i (b_i(t)^T \tilde{\mu(t)})$$

where, $v = R\sqrt{\frac{24}{\epsilon} d \log(\frac{1}{\delta})}$

Contextual MAB

Regret Bound

Goyal and Agrawal [AG13] showed that for stochastic contextual MAB with linear payoff function, with probability $1 - \delta$, the total regret R(T) for Thompson sampling is bounded by $O\left(\frac{d^2}{\epsilon}\sqrt{T^{1+\epsilon}(ln(Td)ln(\frac{1}{\delta}))}\right)$, for any $0 < \epsilon < 1$ and $0 < \delta < 1$.

Contextual MAB

Generalized Thompson Sampling For Contextual MAB

- Generalized Thompson Sampling proposed by the *Lihong Li*[Li13] ,based on the connection between the Thompson sampling and exponential update, he propose the family algorithm called Generalized Thompson algorithm in the expert-learning framework.

Problem Setting

1) Let set of arm $N = \{1, 2, ...K\}$
2) player observe context $x_t \in \chi$,and x_t chosen by the adversary
3) reward $r_t \in \{0, 1\}$ of arm $a \in N$ generate from the mean $\mu(x_t, a_t)$
4) Leaner has the set of experts $E \in \{E_1, ...E_n\}$, and each expert make prediction about the average reward $\mu(x, a)$.

Contextual MAB

5) Learner choose arm according the expert advise **i.e.** $\max_{a \in N} f_i(x, a)$, where $f_i(x, a)$ is the predication of E_i.
6) Prior $P = (p_1, p_2, ...p_n) \in R_+^n$ where $\|P\|_1 = 1$

Algorithm

The algorithm start with, $\eta > 0, \gamma > 0$ and the first posterior as w_1 $= (w_{1,1}, w_{2,1}, ..., w_{n,1})$, where $w_{i,1} = p_i$, and $W_1 = \|w_1\|$
Receive context $x_t \in \chi$
Select arm a_t according to the mixture probabilities: $\forall a \in N$.

$$\Pr(a) = (1 - \gamma) \sum_{i=1}^{N} \frac{w_{i,t} \mathbb{1}(E_i(x_t) = a)}{W_t} + \frac{\gamma}{K}$$

Contextual MAB

Algorithm

Observe reward r_t and update weights:

$$\forall i : w_{i,t+1} \leftarrow w_{i,t}.exp(-\eta.l(f_i(x_t, a_t), r_t)); W_{t+1} \leftarrow \| w_{t+1} \|_1 = \sum_i w_{i,t+1}$$

where $l(f, r)$ is the negative log-likelihood.

Regret Bound

For some constant $\beta \in \mathbb{R}_+$, T round and the $p1$ is the probability of choosing the expert 1 E_1, regret bound for Generalized Thompson Sampling is bounded by

$$O\left(K^{\frac{2}{3}}\beta^{\frac{1}{3}}\sqrt{ln\frac{1}{p1}}\right)$$

Multi-Play MAB

1. In Multi-play or Combinatorial MAB (MP-MAB) problem player play several arms (say $L < K$ arm) from given number of arms K in each round.
2. In many scenario MP-MAB is applicable *e.g.* Online advertising on website.
3. Thompson algorithm for MP-MAB proposed by *Komiyama et al.*[KHN15].

Problem Setting

Let set of arms $N = \{1, 2, 3..K\}$, each arm has reward distribution with some parameter that is unknown to player and player selects $L < K$ arms in each round and observe the rewards associated with L arms.Let $N_i(t)$ be the number of draws of arm i upto time t.

Multi-Play MAB

1. MP-MAB setting is different from the General MAB.
2. In general MAB setting, regret analysis is only depend only on the number of sub-optimal arm pull by the algorithm, but for MP-MAB problem regret analysis has a different structure.*Komiyama et al.*[KHN15] presented a Example: as Let K=4,L=2 and $\mu_1 = .10, \mu_2 = .09, \mu_3 = .08$ and $\mu_4 = .07$. 1st and 2nd are optimal and 3rd and 4th are sub-optimal arms.

Rounds	Game-2	Game-2
t=1	I(1)={1,2} r(1)=0	I(1)={1,3} r(1)=0.01
t=2	I(1)={3,4} r(1)=.04	I(1)={1,4} r(1)=0.02
TotalRegret	R(2)=0.04	R(2)=0.03

In both the game, number of sub-optimal arm pull is same but regret different.

Optimal Regret for MP-MAB

Anantharam et al. [AVW87] prove that for any strongly consistent algorithm and sub-optimal arm i, the number of arm i draws is lower-bounded as

$$E[N_i(T+1)] \geq \left(\frac{1-o(1)}{d(\mu_i, \mu_L)}\right) logT$$

And regret for T round is,

$$E[R(T)] \geq \sum_{i \in [K] \backslash [L]} \left(\frac{1-o(1)\triangle_{i,L}}{d(\mu_i, \mu_L)}\right) logT$$

where, μ_L is the mean of arm L^{th} top arm among the K arms.and $\triangle_{i,L}$ is the difference mean (μ_i) of arm $i \in [K] \backslash [L]$ from the L^{th} mean (μ_L).

Multi-Play MAB

Algorithm 5 Multiple-play Thomson Sampling (MP-TS) for binary rewards

Input: Number of arms (K), number of selection (L)
for i= 1,2,...Kdo:
    $A_i, B_i = 1, 1$
end for
$t \leftarrow 1$
for t = 1,2,...,Tdo
    for i= 1,2,...,Kdo
        $\theta_i(t) \sim Beta(A_i, B_i)$
    end for
    I_t = top-L arms ranked by $\theta_i(t)$.
    for $i \in I_t$ **do**
        if $X_i(t) = 1$ **then**
            $A_i \leftarrow A_i + 1$
        else
            $B_i \leftarrow B_i + 1$
        end if
    end for
end for

Multi-Play MAB

MP-TS Algorithm

Optimal Regret Bound : for any small $\epsilon_1 > 0$ and $\epsilon_2 > 0$ the regret of MP-TS is upper bounded as

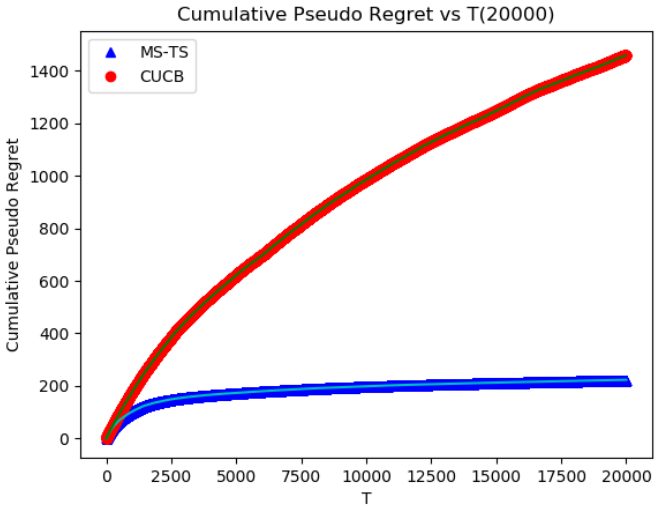
$$E[R(T)] \leq \sum_{i \in [K] \setminus [L]} \left(\frac{1 - o(1) \triangle_{i,L}}{d(\mu_i, \mu_L)} \right) logT + O((logT)^{\frac{2}{3}})$$

Experiment Analysis of MP-TS

[KHN15] We compare MP-TS with *CUCB* algorithm [CWY13] and found performance of MP-TS is batter than the *CUCB*

we consider 20-armed bandits,the simulations include 20 Bernoulli arms with $\mu_1 = 0.15$, $\mu_2 = 0.12$, $\mu_3 = 0.10$, $\mu_i = 0.05$ for $i \in \{4, 5..., 12\}$, $\mu_i = 0.03$ for $i \in \{13, 14, ..., 20\}$, and L $= 3$.

Multi-Play MAB



Budgeted MAB

1. In budgeted MAB,reward and cost associated with each arm, and cost and reward distribution of each arm is unknown to player.
2. Whenever the player pull the arm, he observe the cost and reward of the selected arm.
3. Player has the fixed budget B, hence player can't pull arm after he run-out of budget.
4. player want play optimal policy,so he can accumulate maximum reward with given budget B.
5. The application of BMAB is many scenario like on-spot instance bidding of cloud computing, This new setting models many Internet applications (e.g., ad exchange, sponsored search, and cloud computing)

Budgeted MAB

Problem setting

In budgeted MAB, we consider a slot machine with K arms,At round t, a player pulls an arm $i \in [K]$, receives a random reward $r_i(t)$, and pays a random cost $c_i(t)$, player pull arm until he runs out of his budget B.Both the reward $r_i(t)$ and the cost $c_i(t)$ are supported on [01].

Let expected reward and cost of arm i is μ_i^r and μ_i^c respectively, also we denote arm 1 as optimal arm ($\arg\max_{i\in[K]} \frac{\mu_i^r}{\mu_i^c} = 1$). define pseudo-regret [XLQ+15] .

$$\text{Regret} = R^* - E\left[\sum_{t=1}^{T_B} r_t\right]$$

where R^* is is the expected reward of the optimal policy,r_t is the reward received by an

Budgeted MAB

Algorithm 6 Budgeted Thomson Sampling (BTS)

For each arm $i \in [K]$, set $S_i^r(1) \leftarrow 0$, $F_i^r(1) \leftarrow 0$, $S_i^c \leftarrow 0$ and $F_i^c \leftarrow 0$;
Set $B_1 \leftarrow B; t \leftarrow 1$;
while $B_t > 0$ **do**
    For each arm $i \in [K]$,
    sample $\theta_i^r(t) \sim Beta(S_i^r(t) + 1, F_i^r(t) + 1$, and
    sample $\theta_i^c(t) \sim Beta(S_i^c(t) + 1, F_i^c(t) + 1$;
    Pull arm $I_t = argmax_{i \in [K]} \frac{\theta_i^r(t)}{\theta_i^c(t)}$; receive reward r_t;
    pay cost c_t; update $B_{t+1} \leftarrow B_t - c_t$;
    For Bernoulli bandits, $\tilde{r} \leftarrow r_t$, $\tilde{c} \leftarrow c_t$,
    for general bandit, sample $\tilde{r}$ from $B(r_t)$ and sample $\tilde{c}$ from $B(c_t)$;
    $S_{I_t}^r(t+1) \leftarrow S_{I_t}^r(t) + \tilde{r}$, $F_{I_t}^r(t+1) \leftarrow F_{I_t}^r(t) + 1 - \tilde{r}$;
    $S_{I_t}^c(t+1) \leftarrow S_{I_t}^c(t) + \tilde{c}$, $F_{I_t}^c(t+1) \leftarrow F_{I_t}^c(t) + 1 - \tilde{c}$;
    $\forall j \neq I_t, S_j^r(t+1) \leftarrow S_j^r(t), F_j^r(t+1) \leftarrow F_j^r(t)$,
    $S_j^c(t+1) \leftarrow S_j^c(t), F_j^c(t+1) \leftarrow F_j^c(t)$;
    Set $t \leftarrow t + 1$.
end while

Regret

Xia-Li et al.[XLQ+15] *prove that for* $\forall \gamma \in (0,1)$*, for both Bernoulli bandits and general bandits, the regret of the BTS algorithm can be upper bounded as below.*

$$\text{Regret} \leq \sum_{i=2}^{K} \left\{ \frac{2\log(B)}{\gamma^2 \mu_i^c \Delta_i} \left(\frac{\mu_1^r}{\mu_1^c} + 1 \right)^2 + \Phi_i(\gamma) \right\} + O\left(\frac{K}{\gamma^2} \right)$$

where, we consider the arm 1 is optimal one. and $\Delta_i = \frac{\mu_1^r}{\mu_1^c} - \frac{\mu_i^r}{\mu_i^c} \qquad \forall i \geq 2.$

MOMAB

1. What is Multi-Objective Multi Armed Bandits (MOMABs)?
   - presence of multiple objective.
   - rewards are vector valued.
2. What are the motivation behind MOMAB?
   - consideration of users preferences.
   - Bayesian learning approach.
3. How Multi Objective MAB different from classical MAB?
   - There may be multiple optimal arms w.r.t to preferences of users.
   - Utility function $u(\mu, w)$ is linear.
4. We are minimizing the user regret, i.e., amount of utility lost due to playing non-optimal arms.

Utility MAP-UCB

0: **Input**: A parameter prior on the distribution of w.
0: $C \rightarrow \emptyset$, // previous comparisons
0: $\bar{x}_a \rightarrow$ initialise with single pull, r_a, for each a
0: $n_a \rightarrow 1$ for each a
for $t = \mid \mathbb{A} \mid, ..., T$ **do**
$\bar{w} \rightarrow \mathbb{P}_{simplex}(MAP(w \mid C))$
$\bar{a}^* \rightarrow \text{argmax}_a \bar{w}.\bar{x}_a$
$a(t) \rightarrow \text{argmax}_a(\bar{w}.\bar{x}_a + c(\bar{w}, \bar{x}_a, n_a, t))$
$r(t) \rightarrow$ play$a(t)$and observe reward
$\bar{x}_{a(t)} \rightarrow \frac{n_{a(t)}\bar{x}_a + r(t)}{n_{a(t)}+1}$
$n_{a(t)} + +$
if $\bar{a}^* \neq a(t)$ **then**
perform user comparison for $\bar{x}_{\bar{a}_*}$ ant $\bar{x}_{a(t)}$
and add result $((\bar{x}_{\bar{a}^*} \succ \bar{x}_{a(t)}))$ or $((\bar{x}_{a(t)} \succ \bar{x}_{\bar{a}^*}))$
to C
end if

Interactive Thompson Sampling

0: **Input**: Parameter priors on reward distributions, and on w distribution.
0: C $\rightarrow \emptyset$; // previous comparisons
0: D $\rightarrow \emptyset$; // observed reward data
for $t = 1, ..., T$ **do**
$\eta_1^t, \eta_2^t \rightarrow$ draw 2 samples from $P(\eta^t|C)$
$\theta_1^t, \theta_2^t \rightarrow$ draw 2 samples from $P(\theta^t|D)$
$a_1(t) \rightarrow \text{argmax}_a \mathbb{E}_{P(r,w|a,\theta_1^t,\eta_1^t)}[w.r]$
$a_2(t) \rightarrow \text{argmax}_a \mathbb{E}_{P(r,w|a,\theta_2^t,\eta_2^t)}[w.r]$
r(t)$\rightarrow$ play $a_1(t)$ and observe reward append $(r(t), a_1(t))$ to D
if $a_1(t) \neq a_2(t)$ **then**
$\tilde{\mu}_{1,a_1(t)} \rightarrow \mathbb{E}_{P(r|a_1(t),\theta_1^t)}[r]$
$\tilde{\mu}_{2,a_2(t)} \rightarrow \mathbb{E}_{P(r|a_2(t),\theta_2^t)}[r]$
perform user comparison for $\tilde{\mu}_{1,a_1(t)}$ and $\tilde{\mu}_{2,a_2(t)}$
and add result $((\tilde{\mu}_{1,a_1(t)} \succ \tilde{\mu}_{2,a_2(t)}))$ or $((\tilde{\mu}_{2,a_2(t)} \succ \tilde{\mu}_{1,a_1(t)}))$ to C
end if
end for

Double Thompson Sampling in Dueling Bandit

1. Dueling bandit problem is a variant of classical MAB problem.
   - feedback comes in the form of pairwise comparisons.
2. Two types of dueling bandits :-
   - Condorcet dueling bandit.
   - Copeland dueling bandit.
     - ⋆ Winner is defined as the arm which beats maximum number of other arms.
3. What is double Thompson Sampling?
   - it selects both first and second candidate according to sample drawn independently from posterior distribution.
4. We assume that user preferences are stationary over time.

D-TS model for Dueling bandit

We consider a dueling bandit problem with K arms, where $K \geq 2$, denoted by $A = \{1, 2, ..., K\}$. At every time $t > 0$, user get a pair of arm $(a_t^{(1)}, a_t^{(2)})$ and a noisy comparison outcome w_t is obtained in such a way, that if user prefer $a_t^{(1)}$ over $a_t^{(2)}$ the $w_t = 1$, else $w_t = 2$. The distribution of noisy comparison outcomes is described by the preference matrix $P = [p_{ij}]_{K \times K}$, where $p_{ij} = \mathbb{P}\{i \succ j\}, i, j = 1, 2, ..K$,
$p_{ij} + p_{ji} = 1 \& p_{ii} = \frac{1}{2}$. We say that arm i beats arm j if $p_{ij} > \frac{1}{2}$.
the normalized Copeland score is defined as $\zeta_i = \frac{1}{K-1} \sum_{j \neq i} 1(p_{ij} > \frac{1}{2})$, where $1(.)$ is the indicator function. Let $\zeta_i^* = \max_{1 \leq i \leq K} \zeta_i$. We can define the Copeland winner(s) in term of ζ^* as $\mathcal{C}^* = \{i : 1 \leq i \leq K, \zeta_i = \zeta^*\}$. We can say that Condorcet winner is a special cse of Copeland winner with $\mathcal{C}^* = 1$.

Let us define, a filtration $\mathcal{H}_{t-1}$ are the history observed by dueling bandit algorithm before time t, i.e., $\mathcal{H}_{t-1} = \{a^{(1)_\tau}, a_\tau^{(2)}, w_\tau, \tau = 1, 2..., t-1\}$. Then the performance of a dueling bandit algorithm algorithm Γ is measured by its expected cumulative regret, which is defiend as

$$\mathcal{R}_\Gamma(T) = \zeta^* T - \frac{1}{2} \sum_{t=1}^{T} \mathrm{E}[\zeta_{a^{(1)}t} + \zeta_{a^{(2)}t}] \tag{1}$$

The Objective of dueling bandit algorithm Γ is to minimize $\mathcal{R}_\Gamma(T)$.

D-TS

Init : B$\leftarrow 0_{K\times K}$; // B_{ij} is the number of time slots that the user prefer arm i over j.

for $t = 1$ **to** T **do**

// Phase 1: Choose the first candidate $a^{(1)}$

$U := [u_{ij}], L := [l_{ij}]$, where $u_{ij} = \frac{B_{ij}}{B_{ij}+B+ji} + \sqrt{\frac{\alpha logt}{B_{ij}+B_{ji}}}$ if $i \neq j$

and $u_{ij} = l_{ij} = \frac{1}{2} \quad \forall i \quad ; //\frac{x}{0} := 1$for any x $\hat{\zeta}_i \leftarrow \frac{1}{K-1}\sum_{j\neq i} 1(u_{ij} > \frac{1}{2})$; // upper bound of the normalized Copeland score.

$\mathcal{C} \leftarrow \{i : \hat{\zeta}_i = \max_j \hat{\zeta}_j\}$;

for $i, j = 1, ..K$ with $i < j$ **do**

Sample $\theta_{ij}^{(1)} \sim Beta(B_{ij} + 1, B_{ji} + 1)$;

$\theta_{ij}^{(1)} \leftarrow 1 - \theta_{ij}^{(1)}$;

end for

$a^{(1)} \leftarrow \text{argmax}_i \in \mathcal{C} \sum_{j\neq i} 1(\theta_{ij}^{(1)} > \frac{1}{2})$; // Choosing from $\mathcal{C}$ to eliminate likely non winner

arms; Ties are broken randomly.

end for

0: continue.

for $t = 1$ **to** T **do**

// Phase 2: Choose the second candidate $a^{(2)}$

Sample $\theta^{(2)}_{ia^{(1)}} \sim Beta(B_{ia^{(1)}} + 1, B_{a^{(1)}i} + 1)\ \forall i \neq a^{(2)}$, and let $\theta^{(2)}_{a^{(1)}a^{(2)}} = \frac{1}{2}$;

$a^{(2)} \leftarrow \text{argmax}_{i:I_{ia^{(1)}} \leq \frac{1}{2}} \theta^{(2)}_{ia^{(1)}}$; // choosing only from uncertain pairs.

// Compare and update

Compare pair $(a^{(1)}, a^{(2)})$ and observe the result w;

Update B: $B_{a^{(1)}a^{(2)}} \leftarrow B_{a^{(1)}a^{(2)}} + 1$ if $w = 1$, or $B_{a^{(2)}a^{(1)}} \leftarrow B_{a^{(2)}a^{(1)}} + 1$ if $w = 2$;

end for

1. Agent interact with user and environment. Umap-UCB and ITS both algorithms pose pairwise comparison queries to the user and elaborate the Bayesian logistics regression to learn about the preferences of users. Performance of both the algorithms is approximately near to the regret of UCB1 and regular Thompson Sampling provided with the ground truth utility function of the user. Umap-utility performs better than UCB1 having access to ground truth. ITS outperforms Umap-UCB both in terms of regret and the number of queries posed to the user[CD10].

2. Regret for D-TS is bounded by $O(K^2 log T)$[WL16].

References I

[AG13] Shipra Agrawal and Navin Goyal, *Thompson sampling for contextual bandits with linear payoffs*, International Conference on Machine Learning, 2013, pp. 127–135.

[AVW87] Venkatachalam Anantharam, Pravin Varaiya, and Jean Walrand, *Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays-part i: Iid rewards*, IEEE Transactions on Automatic Control **32** (1987), no. 11, 968–976.

[CD10] Shamik Chaudhuri and Kalyanmoy Deb, *An interactive evolutionary multi-objective optimization and decision making procedure*, Applied Soft Computing **10** (2010), no. 2, 496–511.

References II

[CWY13] Wei Chen, Yajun Wang, and Yang Yuan, *Combinatorial multi-armed bandit: General framework and applications*, International Conference on Machine Learning, 2013, pp. 151–159.

[KHN15] Junpei Komiyama, Junya Honda, and Hiroshi Nakagawa, *Optimal regret analysis of thompson sampling in stochastic multi-armed bandit problem with multiple plays*, arXiv preprint arXiv:1506.00779 (2015).

[Li13] Lihong Li, *Generalized thompson sampling for contextual bandits*, arXiv preprint arXiv:1310.7163 (2013).

[WL16] Huasen Wu and Xin Liu, *Double thompson sampling for dueling bandits*, Advances in Neural Information Processing Systems, 2016, pp. 649–657.

References III

[XLQ+15] Yingce Xia, Haifang Li, Tao Qin, Nenghai Yu, and Tie-Yan Liu, *Thompson sampling for budgeted multi-armed bandits.*, IJCAI, 2015, pp. 3960–3966.