Optimization &
Statistical
Inference Lab

Paper Review
An Empirical Evaluation of Thompson Sampling

Olivier Chapelle, Lihong Li

NIPS 2011

Citation=492

Issue

To solve exploration / exploitation or bandit problem

Major Algorithm

1. UCB
2. Gittins
3. Thompson Sampling

UCB

Strong theoretical guarantees on the regret
But…

Gittins

Bayes-optimal approach

TS

Idea
To randomly draw each arm according to its probability of being optimal
(Not full Bayes, namely simple)

Lack of theoretical analysis

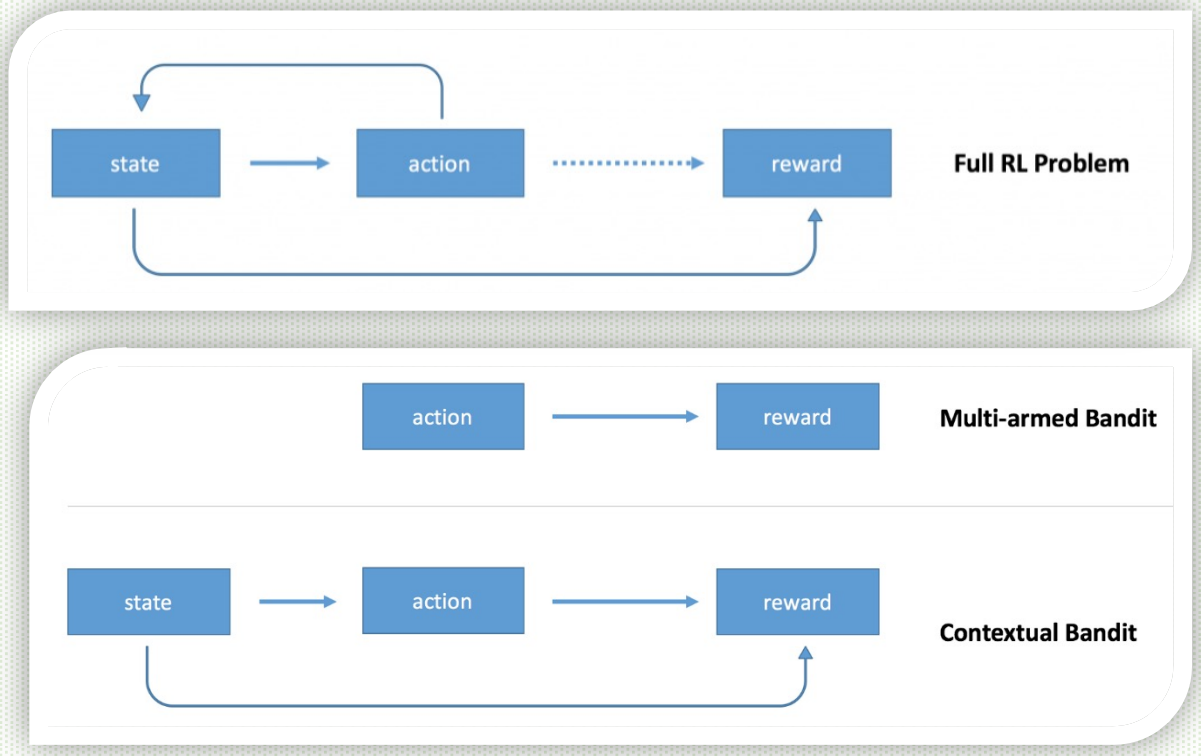
In this paper….

Algorithm description ➔ Comparison between algorithms ➔ Conclusion

Before begin…

Do you know the difference between **Multi armed bandit** and **reinforcement learning**?

Have you heard about the **contextual bandit**?



MAB deals with only action to get reward, but there is more things to think, 'state', in the contextual Bandit problem. It is different from reinforcement learning.

Source from http://pavel.surmenok.com/2017/08/26/contextual-bandits-and-reinforcement-learning/

Experiment data set : display advertisement selection, news article recommendation

Problem Setting

The contextual bandit setting is as follows. At each round we have a context x (optional) and a set of actions $\mathcal{A}$. After choosing an action $a \in \mathcal{A}$, we observe a reward r. The goal is to find a policy that selects actions such that the cumulative reward is as large as possible.

→ Contextual Bandit

Thompson sampling is best understood in a Bayesian setting as follows. The set of past observations D is made of triplets (x_i, a_i, r_i) and are modeled using a parametric likelihood function $P(r|a, x, \theta)$ depending on some parameters θ. Given some prior distribution $P(\theta)$ on these parameters, the posterior distribution of these parameters is given by the Bayes rule, $P(\theta|D) \propto \prod P(r_i|a_i, x_i, \theta)P(\theta)$.

→ Thompson Sampling

$\theta^*$is unknown.

-Just interested in maximizing the immediate reward **exploitation**, then one should choose the action that maximize below.

$$\mathbb{E}(r|a,x) = \int \mathbb{E}(r|a,x,\theta)P(\theta|D)d\theta.$$

-the probability matching heuristic consists in randomly selecting an action a according to its probability of being optimal

$$\int \mathbb{I}\left[\mathbb{E}(r|a,x,\theta) = \max_{a'} \mathbb{E}(r|a',x,\theta)\right] P(\theta|D)d\theta,$$

Not computed explicitly

Algorithm 1 Thompson sampling

$D = \emptyset$
for $t = 1, \ldots, T$ **do**
  Receive context x_t
  Draw θ^t according to $P(\theta|D)$
  Select $a_t = \arg\max_a \mathbb{E}_r(r|x_t, a, \theta^t)$
  Observe reward r_t
  $D = D \cup (x_t, a_t, r_t)$
end for

The left below figure is from "finite time analysis of the mab problem"

Each action corresponds to the choice of an arm

Deterministic policy: UCB1.
Initialization: Play each machine once.
Loop:

- Play machine j that maximizes $\bar{x}_j + \sqrt{\frac{2 \ln n}{n_j}}$, where $\bar{x}_j$ is the average reward obtained from machine j, n_j is the number of times machine j has been played so far, and n is the overall number of plays done so far.

Algorithm 2 Thompson sampling for the Bernoulli bandit

Require: α, β prior parameters of a Beta distribution
$S_i = 0, F_i = 0, \ \forall i$. {Success and failure counters}
for $t = 1, \ldots, T$ **do**
  for $i = 1, \ldots, K$ **do**
    Draw θ_i according to Beta$(S_i + \alpha, F_i + \beta)$.
  end for
  Draw arm $\hat{\imath} = \arg\max_i \theta_i$ and observe reward r
  if $r = 1$ **then**
    $S_{\hat{\imath}} = S_{\hat{\imath}} + 1$
  else
    $F_{\hat{\imath}} = F_{\hat{\imath}} + 1$
  end if
end for

Simulation

Reward Probability
Best arm = 0.5
K-1 other arms = 0.5 - epsilon

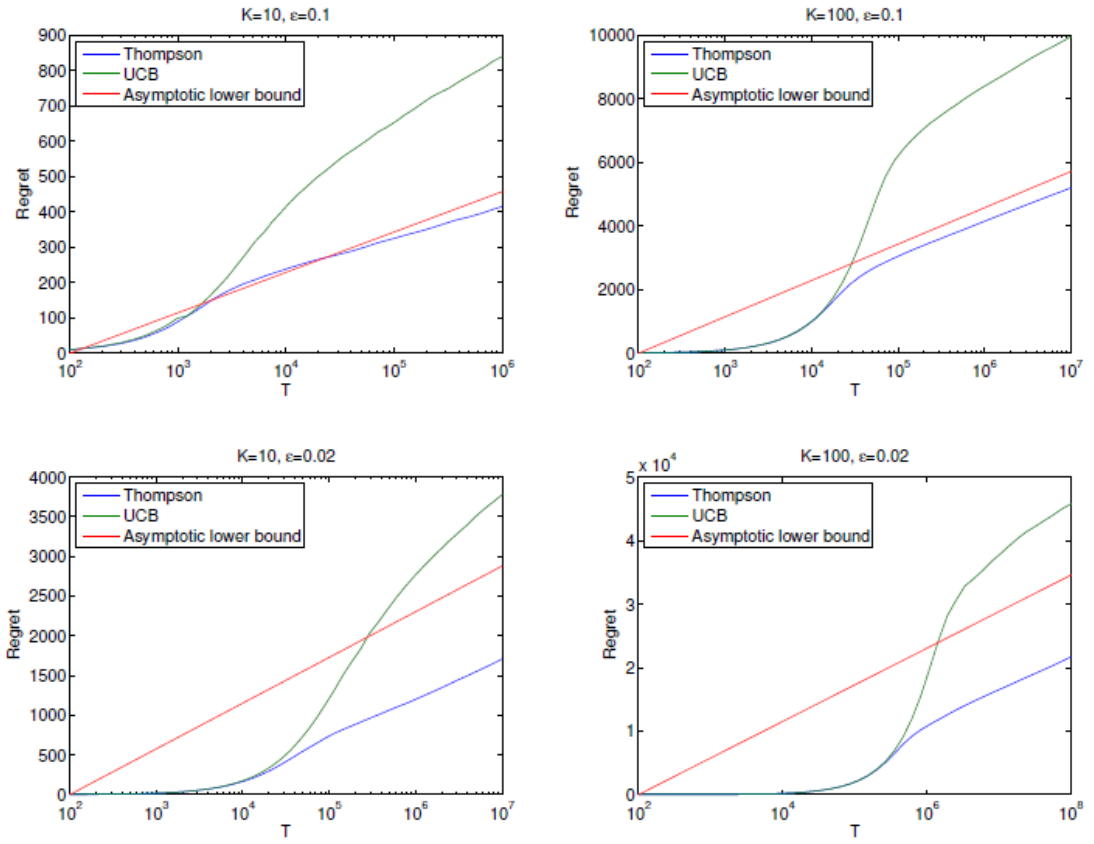
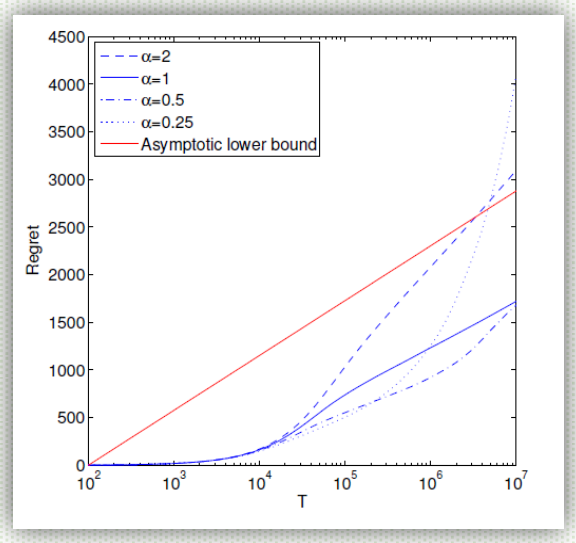
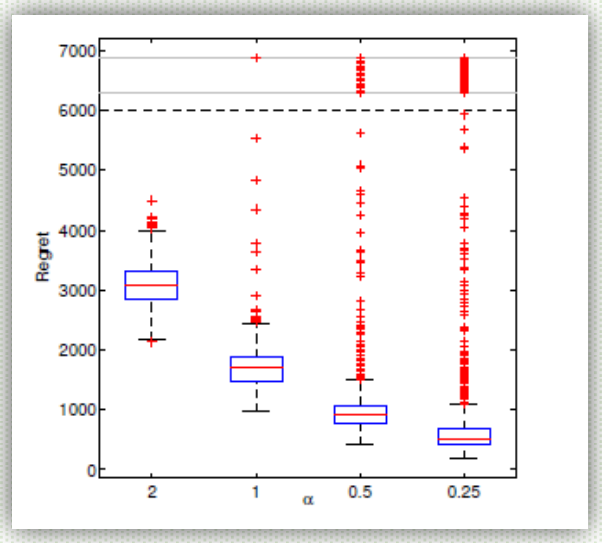


Figure 1: Cumulative regret for $K \in \{10, 100\}$ and $\varepsilon \in \{0.02, 0.1\}$. The plots are averaged over 100 repetitions. The red line is the lower bound (2) shifted such that it goes through the origin.

Analysis
The result include already some prior mismatch because the Beta prior with parameters (1,1) has a large variance while the true probabilities were selected to be close to 0.5. But…
Always better than UCB.

Right: distribution of the regret at T=10^7





Posterior reshaping

The posterior is a Beta distribution with parameters a and b, and we have tried to change it to parameters $\frac{a}{\alpha}, \frac{b}{\alpha}$. Doing so does not change the posterior mean, but multiply its variance by a factor close to α^2

$\alpha < 1$: decrease the amount of exploration , lower regret

Impact of delay

In a real world, the feedback is typically not processed immediately because of various runtime constraints. They try to quantify the impact of this delay by doing some simulations that mimic the problem of news articles recommendation.

δ	1	3	10	32	100	316	1000
UCB	24,145	24,695	25,662	28,148	37,141	77,687	226,220
TS	9,105	9,199	9,049	9,451	11,550	21,594	59,256
Ratio	2.65	2.68	2.84	2.98	3.22	3.60	3.82

It appears that Thompson sampling is more robust than UCB when the delay is long.

I think this difference comes TS's randomizing over actions (contrast to UCB – deterministic)

Display Advertising

A key element in this matching problem is the **click-through rate(CTR) estimation.** There is of course a fundamental exploration / exploitation dilemma here: in order to learn the CTR of an ad, it needs to be displayed, leading to a potential loss of short-term revenue. In this paper, we **consider standard regularized logistic regression for predicting CTR**.

Feature: identifiers of the ad, advertiser, publisher and visited page, etc. Theses are hashed and each training sample ends up being represented as sparse binary vector of dimension 2^{24}.

Posterior: Gaussian distribution with diagonal covariance matrix.

Algorithm 3 Regularized logistic regression with batch updates

Require: Regularization parameter $\lambda > 0$.

$m_i = 0,\ q_i = \lambda$. {Each weight w_i has an independent prior $\mathcal{N}(m_i, q_i^{-1})$}

for $t = 1, \ldots, T$ **do**

Get a new batch of training data $(\mathbf{x}_j, y_j),\ \ j = 1, \ldots, n$.

Find w as the minimizer of: $\frac{1}{2}\sum_{i=1}^{d} q_i(w_i - m_i)^2 + \sum_{j=1}^{n} \log(1 + \exp(-y_j \mathbf{w}^\top \mathbf{x}_j))$.

$m_i = w_i$

$q_i = q_i + \sum_{j=1}^{n} x_{ij}^2 p_j(1 - p_j),\ \ p_j = (1 + \exp(-\mathbf{w}^\top \mathbf{x}_j))^{-1}$ {Laplace approximation}

end for

The input feature vectors x are as in the real world setting, but the clicks are artificially generated with probability.

The clicks are artificially generated with probability below.

$$P(y = 1|\mathbf{x}) = (1 + \exp(-\mathbf{w}^{*\top}\mathbf{x}))^{-1}.$$

Method

Thompson sampling with Gaussian posterior approximation

LinUCB $\sum_{i=1}^{d} m_i x_i + \alpha\sqrt{\sum_{i=1}^{d} q_i^{-1} x_i^2}$

Exploit-only Select the ad with the hightest mean

Random Select the ad uniformly at random.

Epsilon greedy with epsilon probability, select a random ad; other wise, select the one with the highest mean

Table 2: CTR regrets on the display advertising data.

Method	TS			LinUCB			ε-greedy			Exploit	Random
Parameter	0.25	0.5	1	0.5	1	2	0.005	0.01	0.02		
Regret (%)	4.45	3.72	3.81	4.99	4.22	4.14	5.05	4.98	5.22	5.00	31.95

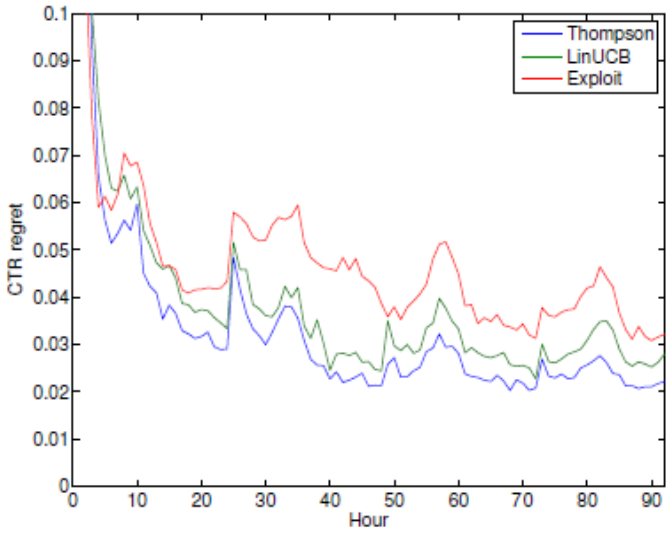


Figure 4: CTR regret over the 4 days test period for 3 algorithms: Thompson sampling with $\alpha = 0.5$, LinUCB with $\alpha = 2$, Exploit-only. The regret in the first hour is large, around 0.3, because the algorithms predict randomly (no initial model provided).

News Article Recommendation

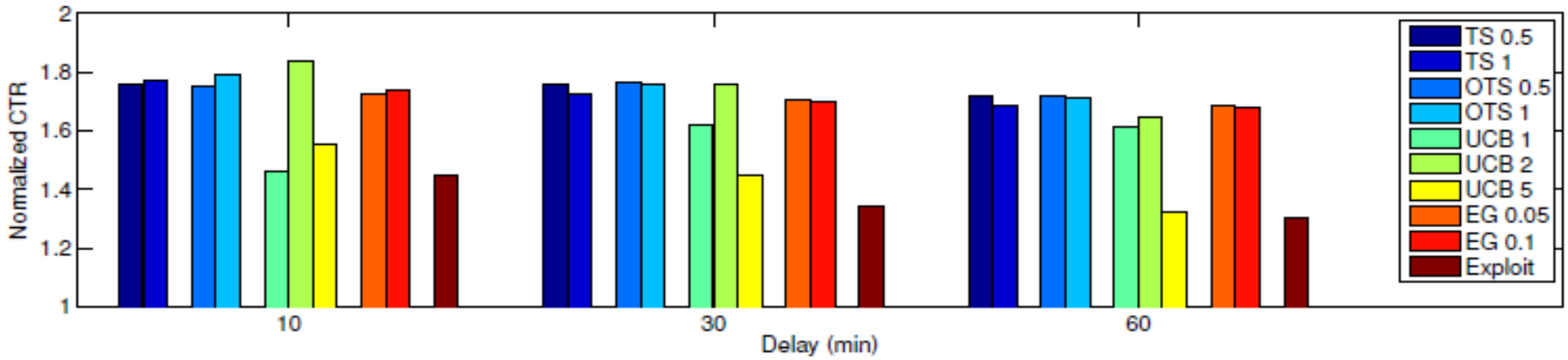


Figure 5: Normalized CTRs of various algorithm on the news article recommendation data with different update delays: $\{10, 30, 60\}$ minutes. The normalization is with respect to a random baseline.

Conclusion

1. Thompson sampling is a very effective heuristic for addressing the exploration / exploitation trade-off.
2. It does not have any parameter to tune, but their results show that tweaking the posterior to reduce exploration can be beneficial.
3. Since it is a randomized algorithm, it is robust in the case of delayed feedback.