Algorithms for Adaptive Experiments that Trade-off Statistical Analysis with Reward: Combining Uniform Random Assignment and Reward Maximization

Authors: Tong Li, Jacob Nogas, Haochen Song, Bingcheng Wang, Huayin Luo, Arghavan Modiri, Nina Deliu, Ben Prystawski, Sofía S. Villar, Audrey Durand, Anna Rafferty, Joseph J. Williams

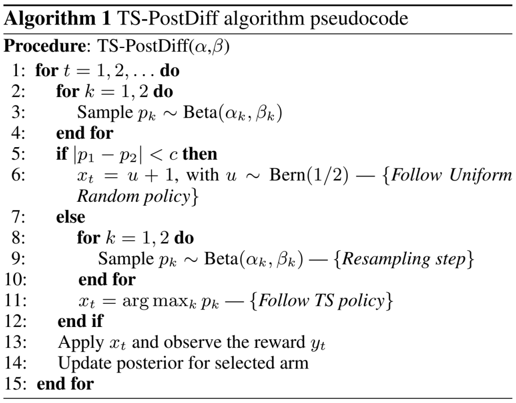
Introduction

Adaptive algorithms are traditionally used for increasing user experience and avoiding harmful actions in randomized controlled trials. However, such algorithms often collect data with measurement bias[1,2]. In particular, Thompson Sampling (TS) exhibits both lowered power (probability of correctly detecting a difference in arms) and increased False Positive Rates (FPR, probability of incorrectly concluding a difference exists)[3,4]. This paper focused on changing the adaptive algorithm to balance competing goals of minimizing regret and obtaining data that is useful for drawing conclusions.

We present a novel algorithm, called TS-PostDiff that takes a Bayesian approach to mixing TS and Uniform Random (UR): the probability a participant is assigned using UR allocation is the posterior probability that the difference between two arms is 'small' (below a certain threshold). This allows for more UR exploration when there is little or no reward to be gained and thus achieves better trade-off between FPR, statistical power, and reward.

Algorithm

In TS-PostDiff, we define ϕ as the posterior probability (after t steps) that the difference in reward between the 2 arms is less than a threshold 'c', where 'c' is a pre-defined threshold parameter. Then, with probability ϕ, choose action uniformly at random and with probability 1 - ϕ, choose action according to an adaptive allocation strategy. An illustration of the algorithm is as follows:



Algorithm 1 TS-PostDiff algorithm pseudocode

```
Procedure: TS-PostDiff($\alpha$,$\beta$)
for $t = 1, 2, \ldots$ do
    for $k = 1, 2$ do
        Sample $p_k \sim \text{Beta}(\alpha_k, \beta_k)$
    end for
    if $|p_1 - p_2| < c$ then
        $x_t = u + 1$, with $u \sim \text{Bern}(1/2)$ — {Follow Uniform Random policy}
    else
        for $k = 1, 2$ do
            Sample $p_k \sim \text{Beta}(\alpha_k, \beta_k)$ — {Resampling step}
        end for
        $x_t = \arg\max_k p_k$ — {Follow TS policy}
    end if
    Apply $x_t$ and observe the reward $y_t$
    Update posterior for selected arm
end for
```

TS-PostDiff was designed to balance between exploration and exploitation:
(1) TS-PostDiff uses UR allocation when there is little evidence for a difference in arm means, reducing the chances of getting a false positive with no or minimal-cost to reward; (2) TS-PostDiff doing TS reward maximization when there seems to be a difference in arm means that is large relative to the parameter for small differences below which maximizing reward is not a high priority (maximizing reward while potentially reducing power).

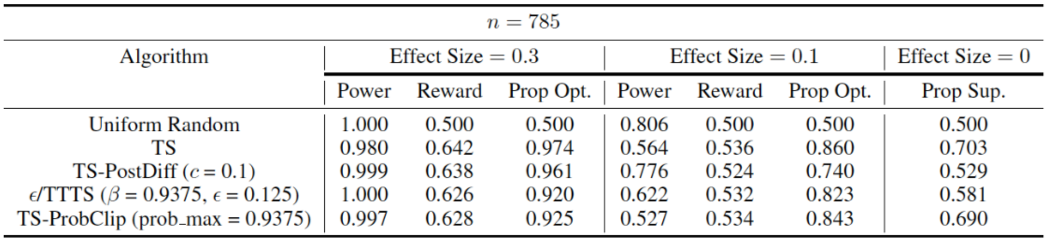
In evaluating the TS-PostDiff algorithm, we compare it with other algorithms that mix UR with Multi-Armed Bandits (MAB) allocation to balance between the FPR and power. Epsilon/Top-Two Thompson Sampling $\epsilon/TTTS$[6] adds fixed amounts of UR allocation; it allocates with UR with probability ϵ and with TS with probability $1-\epsilon$, and TS-Probability Clipping[5] defines a minimum amount of exploration, which ensures the algorithm always has a chance to explore the seemingly worse arm.

Results

We conducted 10 real-world educational experiments during 2021, where we collected data side by side using both UR (traditional RCT/experiment) and TS-PostDiff (adaptive experiment). The different arms tested alternative interventions to support students in doing better in online programming homework. We then used the effect sizes (arm differences) and sample sizes of these real-world experiments to guide our choice of comprehensive simulations, where each algorithm (TS-PostDiff, $\epsilon/TTTS$, TS-Probability Clipping, TS, and UR) was simulated over 10,000 experiments, and we compared their reward, FPR and power.

We compared the three algorithms under the parameterization that have similar FPR (0.06) . We simulated 10,000 times with 1,171 sample size. The results exhibited that when the effect size was 0.045, the rewards from all three algorithms were all around 0.524, but TS-PostDiff provided power as 0.458, which was higher than $\epsilon/TTTS(0.392)$ and TS-Prob Clipping(0.365). This matched the design goal of the TS-PostDiff algorithm: when the true effect size is small, TS-PostDiff will explore more and put higher priority on statistical inference (in which case there won't be much loss in the reward).

We further simulated a combination of settings where effect size varied from 0 to 0.1, and 0.3; sample size went from 100 to 1000; 40 different values of parameters for each of the three algorithms under their respective range. In our simulations, we considered a 2-armed Bernoulli Bandit setting with arm means p_1 and p_2. Due to limitation of space, we only presented results under the setting: a sample size of 785, an effect size of 0(p_1=0.5, p_2=0.5), 0.1 (p_1=0.55, p_2=0.45) and 0.3 (p_1=0.65, p_2=0.35). We chose c = 0.1 for TS-PostDiff, β = 0.9375 for $\epsilon/TTTS$, $prob_{max}$= 0.9375 for TS-Prob Clipping. These parameters all provided FPR equal to 0.08 when $p_1 = p_2 = 0.5$ (for fair comparison). Under those settings, we saw again that TS-PostDiff achieved better power/statistical inference when effect size was low, and better reward when effect size was large, compared to the other two algorithms.



$n = 785$							
Algorithm	Effect Size = 0.3			Effect Size = 0.1			Effect Size = 0
	Power	Reward	Prop Opt.	Power	Reward	Prop Opt.	Prop Sup.
Uniform Random	1.000	0.500	0.500	0.806	0.500	0.500	0.500
TS	0.980	0.642	0.974	0.564	0.536	0.860	0.703
TS-PostDiff ($c = 0.1$)	0.999	0.638	0.961	0.776	0.524	0.740	0.529
ϵ/TTTS ($\beta = 0.9375$, $\epsilon = 0.125$)	1.000	0.626	0.920	0.622	0.532	0.823	0.581
TS-ProbClip (prob_max = 0.9375)	0.997	0.628	0.925	0.527	0.534	0.843	0.690

Figure 1: Comparison of power, mean reward, proportion of optimal allocation, and proportion of superior allocation for UR, TS, TS-PostDiff ($c = 0.1$, $\epsilon/TTTS(\beta = 0.9375,)\epsilon = 0.125$, and TS-Prob Clipping ($prob_{max} = 0.9375$). Results were shown for effect size 0.3, 0.1 and 0 in the first, second and third blocks respectively.

Discussion

The research presents an innovative algorithm to adapt different sample sizes and effect sizes, with a good trade-off between reward, power and false positive rate. In comparison with other state of the art techniques, it maximizes reward when effect size is large while maintaining similar power, and it prioritizes and achieves higher power when effect size is small and there is little reward to be gained. This is particularly valuable for TS induced algorithms, as they have an internal bias due to its reward maximization nature. Such bias could potentially be further reduced for a dynamic threshold balancing the proportion of UR and TS.

References

1. Nie, X., Tian, X., Taylor, J., Zou, J. (2018, March). Why adaptively collected data have negative bias and how to correct for it. In International Conference on Artificial Intelligence and Statistics (pp. 1261-1269). PMLR.

2. Erraqabi, A., Lazaric, A., Valko, M., Brunskill, E., Liu, Y. E. (2017, April). Trading off rewards and errors in multi-armed bandits. In Artificial Intelligence and Statistics (pp. 709-717). PMLR.

3. Rafferty, A., Ying, H., Williams, J. (2019). Statistical consequences of using multi-armed bandits to conduct adaptive educational experiments. Journal of Educational Data Mining, 11(1), 47-79.

4. Villar, S. S., Bowden, J., Wason, J. (2015). Multi-armed bandit models for the optimal design of clinical trials: benefits and challenges. Statistical science: a review journal of the Institute of Mathematical Statistics, 30(2), 199.

5. Zhang, K., Janson, L., Murphy, S. (2020). Inference for batched bandits. Advances in neural information processing systems, 33, 9818-9829.

6. Russo, D. (2016, June). Simple Bayesian algorithms for best arm identification. In Conference on Learning Theory (pp. 1417-1418). PMLR.