Online Sign Identification: Minimization of the Number of Errors in Thresholding Bandits

Reda Ouhamma*, Rémy Degenne*, Pierre Gaillard†, Vianney Perchet‡

* Univ. Lille, Inria, CNRS, Centrale Lille, UMR 9189 CRIStAL, F-59000 Lille, France. † Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, 38000 Grenoble, France. ‡ Crest, Ensae & Criteo AI Lab

NEURAL INFORMATION PROCESSING SYSTEMS

Setting and contributions

Online sign identification

- Multi-armed bandit: Arm $k \sim \nu_k$, mean μ_k, variance σ^2
- predict $s_k = \text{sign}(\mu_k) \in \{-1, 1\}$.
- Given weights $(a_k)_{1 \le k \le K} \in \mathbb{R}$ and budget of T samples, minimize

$$L_T = \sum_{k=1}^K a_k \mathbb{I}\{\hat{s}_k \neq s_k\}$$

Contributions: We investigate the thresholding bandit problem with a weighted number of errors loss. Our contributions are:

1. A generic method to design algorithms, with generic proof and good performance on the weighted number of errors loss
2. The class of algorithms we analyze includes both LSA and APT, we improve the exponential decay rate in their bound: by a factor of 4005 for LSA and 8 for APT
3. Lower-bounds and counter-intuitive results regarding adaptivity

Index-based algorithms

We consider **Index-based algorithms:** at time $t+1$ pull:

$$i_{t+1} \in \arg\min_{k \in [K]} F(N_{k,t}, N_{k,t}\hat{\Delta}^2_{k,t}, a_k)$$

Algorithm 1 Index-based algorithms for thresholding bandit

1: **Input parameters:** an index function $F\colon \mathbb{N} \times \mathbb{R}_+ \times \mathbb{R}_+^* \to \mathbb{R}$, $a_1, \dots, a_K > 0, \sigma > 0$
2: **for all** $t \in [T]$ do **do**
3:   **for all** $k \in [K]$ define **do**
4:     $N_{k,t-1} = \sum_{s=1}^{t-1} \mathbb{I}\{k = i_s\}$, $\quad \hat{\mu}_{k,t-1} = \frac{1}{N_{k,t-1}} \sum_{s=1}^{t-1} \mathbb{I}\{k = i_s\} X_s$
    $\hat{\Delta}^2_{k,t-1} = \frac{1}{2\sigma^2}\hat{\mu}^2_{k,t-1}$
5:   **end for**
6:   pull $i_t \in \arg\min_{k \in [K]} F(N_{k,t-1}, N_{k,t-1}\hat{\Delta}^2_{k,t-1}, a_k)$.
7:   observe $X_t \sim \nu_{i_t}$
8: **end for**
9: Define $t_{\max} = \max_{t \in [T]} \min_{k \in [K]} F(N_{k,t}, N_{k,t}\hat{\Delta}^2_{k,t}, a_k)$
10: Return for each $k \in [K]$, $\quad \hat{s}_k = \text{sign}(\hat{\mu}_{k,t_{\max}})$

Assumption The index function $F(n, x, a)\colon \mathbb{N} \times \mathbb{R}_+ \times \mathbb{R}_+^* \to \mathbb{R}$ is non-decreasing in n and x and $\lim_{n \to +\infty} F(n, ny, a) = +\infty$ for all $y > 0, a > 0$.

Examples: we recall the following algorithms

- LSA of [1]: $i_{t+1} = \arg\min_{k \in [K]} N_{k,t}\hat{\Delta}^2_k$. Equivalent to $F(n, x) = x$.
- APT of [2]: $i_{t+1} = \arg\min_{k \in [K]} \alpha N_{k,t}\hat{\Delta}^2_{k,t} + \log N_{k,t}$. Equivalent to $F(n, x) = x + \log(n)$.

Oracle & lower bounds

Non-adaptive oracle

- If known gaps ($\Delta_k = \frac{|\mu_k|}{\sigma\sqrt{2}}$), fixed pull number $N_{k,T}$ of arm k, then using Hoeffding's inequality: $\quad \mathbb{E}[L_T] \le \sum_{k=1}^K a_k e^{-N_{k,T}\Delta_k^2}$
- Oracle: minimize the above upper bound. Assume wlog that $a_1\Delta_1^2 \le \ldots \le a_K\Delta_K^2$, then

$$\exists k_0 \in [K], \quad \mathbb{E}[L_T] \le \sum_{k<k_0} a_k + \sum_{k \ge k_0} a_k \exp\left(-\frac{T + \sum_{j \in S}\frac{1}{\Delta_j^2}\log\left(\frac{a_k\Delta_k^2}{a_j\Delta_j^2}\right)}{\sum_{j \in S}\frac{1}{\Delta_j^2}}\right)$$

Remark: This oracle doesn't pull the arms with index smaller than k_0.

Lower bound: fix $\{\Delta_k, k \in [K]\}$ and $T \ge K$. For any algorithm, there exists $\mu_k \in \{\Delta_k, -\Delta_k\}$ such that

$$\mathbb{E}[L_T] \ge \frac{1}{4}\min_{\sum_k N_k = T} \sum_{k=1}^K a_k e^{-4N_k\Delta_k^2}$$

A good algorithm pulls all arms: There exists $\mu_1, \mu_2, \mu_{1,\epsilon}, \mu_{2,\epsilon}$ that if $\max_{\tilde{\mu} \in \mu_1, \mu_2} \mathbb{E}_{\tilde{\mu}}[L_T] \le c_1 \min_{\sum_k N_k = T} \sum_k e^{-c_0 N_k \Delta_k^2}$ then

$$\max_{\mu \in \{\mu_{1,\epsilon}, \mu_{2,\epsilon}\}} \mathbb{E}_\mu\left[\sum_{k=1}^{k_0} N_{k,T}\right] = \Omega(T)$$

Remark: The latter implies that we cannot achieve the M-shaped allocation given by the non-adaptive oracle.

FWT algorithm statement, intuition

Intuition behind FWT:

1. Write the loss upper bound: $B(N_T) = \sum_{k=1}^K a_k e^{-N_{k,T}\Delta_k^2}$
2. Estimate its gradient sequentially: $\nabla B(N_t) = \left(-a_k\Delta_k^2 e^{-N_{k,t}\Delta_k^2}\right)_k$
3. Gaps must be estimated $\implies \hat{\nabla} B(N_t)_k = -a_k\hat{\Delta}_k^2 e^{-N_{k,t}\hat{\Delta}^2_{k,t}}$
4. Frank-Wolfe recommends $F_0(n, x, a_k) = x - \log x + \log(n/a_k)$
5. F_0 is decreasing in x for $x \in (0, 1)$ so we propose the modification:

$$F^{\texttt{FWT}}(n, x, a_k) = \max\{x, 1\} - \log(\max\{x, 1\}) + \log(n/a_k)$$

Deriving LSA & APT:

- APT: can be derived similarly to FWT (see above) using the upper-bound:

$$\mathbb{E}[L_T] = \mathbb{E}\left[\sum_{k=1}^K a_k \mathbb{I}\{\hat{s}_k \neq s_k\}\right] \le B(N_t) = \max_{k \in [K]} e^{-N_{k,t}\Delta_k^2}.$$

- LSA: Derived using the same upper-bound as FWT, with a different solution for the instability: Instead of step 5 (see above), they estimate:

$$\hat{\Delta}_i^{-1} \sim \sqrt{N_{i,t}}.$$

Loss analysis

Theorem: Assume that $a_k = 1$, it comes:

$$\mathbb{E}[L_T^{\text{APT}}] \le 2\sqrt{eT}\sum_{k=1}^K a_k \exp\left(-\frac{1}{4}\frac{T}{\sum_{k=1}^K 1/\Delta_k^2}\right),$$

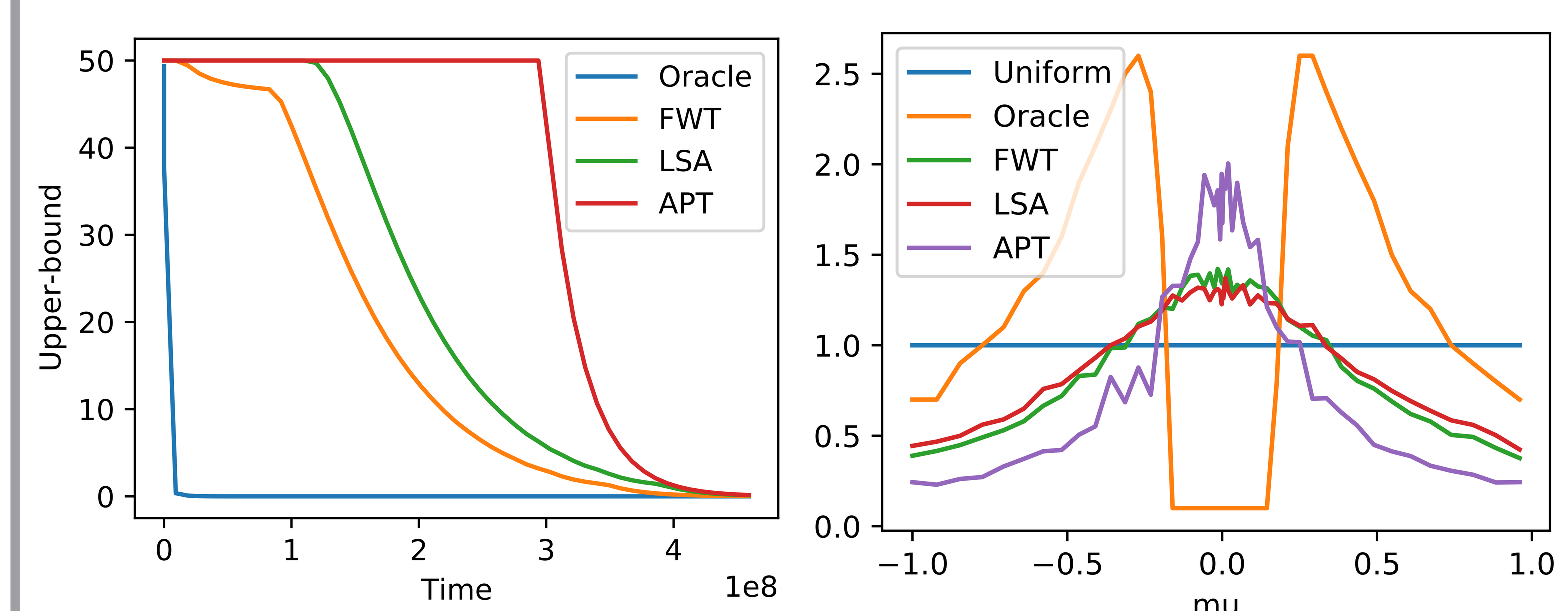
for $T \ge 2\sum_{j=1}^K \frac{1}{\Delta_j^2}\left(2 + \log\frac{a_j\Delta_j^2 \max_i a_i\Delta_i^2}{(\min_k a_k\Delta_k^2)^2} - \log\frac{T}{e^3}\right)$, it comes:

$$\mathbb{E}[L_T^{\texttt{FWT}}] \le 2\sqrt{eT}\sum_k a_k \exp\left(-\frac{1}{2}\frac{T/2 - \sum_j \frac{1}{\Delta_j^2}\log\frac{a_j\Delta_j^2}{a_k\Delta_k^2}}{\sum_j 1/\Delta_j^2}\right)$$

Remarks:

- Theorem 2 in the paper is less explicit but more general and holds for all $T > 0$.
- LSA's bound is less explicit, close to FWT's for large T.
- Our bounds improve over the original results in terms of the exponential decay rate: LSA by a factor of 4005 and APT by 8
- LSA & FWT recover the exponent of the oracle (up to factor 1/4).

Empirical illustration: Consider $\Delta_i = (i/K)^2$. In the left figure (below) we compare the loss upper bounds, in the right one we see the oracle and empirical sampling distributions with respect to μ.



Conclusion

- For thresholding bandits:
  1. We propose FWT that achieves explicit finite time loss bounds
  2. We use our proof to improve the original bound of LSA by a factor of 4005 and APT by 8.
  3. Our method, FWT, is within a factor 4 of the oracle.
- Other results in the paper:
  1. Extension to the sum-of-gaps objective
  2. Benefits of adaptivity, our algorithms surpass the optimal non-adaptive oracle empirically in certain settings.
- Future work: the paper could be complemented by a deeper theoretical analysis of adaptivity.

References

[1] Andrea Locatelli, Maurilio Gutzeit, and Alexandra Carpentier. An optimal algorithm for the thresholding bandit problem. In *International Conference on Machine Learning*, pages 1690–1698. PMLR, 2016.

[2] Chao Tao, Saúl Blanco, Jian Peng, and Yuan Zhou. Thresholding bandit with optimal aggregate regret. *arXiv preprint arXiv:1905.11046*, 2019.