From PAC to Instance-Optimal Sample Complexity in the Plackett-Luce Model

Aadirupa Saha [1] **Aditya Gopalan** [1]

Abstract

We consider PAC-learning a good item from k-subsetwise feedback information sampled from a Plackett-Luce probability model, with *instance-dependent* sample complexity performance. In the setting where subsets of a fixed size can be tested and top-ranked feedback is made available to the learner, we give an algorithm with optimal instance-dependent sample complexity, for PAC best arm identification, of $O\bigg(\frac{\Theta_{[k]}}{k}\sum_{i=2}^{n}\max\Big(1,\frac{1}{\Delta_i^2}\Big)\ln\frac{k}{\delta}\Big(\ln\frac{1}{\Delta_i}\Big)\bigg)$, Δ_i being the Plackett-Luce parameter gap between the best and the i^{th} best item, and $\Theta_{[k]}$ is the sum of the Plackett-Luce parameters for the top-k items. The algorithm is based on a wrapper around a PAC winner-finding algorithm with weaker performance guarantees to adapt to the hardness of the input instance. The sample complexity is also shown to be multiplicatively better depending on the length of rank-ordered feedback available in each subset-wise play. We show optimality of our algorithms with matching sample complexity lower bounds. We next address the winner-finding problem in Plackett-Luce models in the fixed-budget setting with instance dependent upper and lower bounds on the misidentification probability, of $\Omega\left(\exp(-2\tilde{\Delta}Q)\right)$ for a given budget Q, where $\tilde{\Delta}$ is an explicit instance-dependent problem complexity parameter. Numerical performance results are also reported.

1. Introduction

We consider the problem of sequentially learning the best item of a set when subsets of items can be tested but information about only their relative strengths is observed. This is a basic search problem motivated by applications in recommender systems and information retrieval (Hofmann et al., 2013; Radlinski et al., 2008), crowdsourced ranking (Chen et al., 2013), tournament design (Graepel & Herbrich, 2006), etc. It has received recent attention in the online learning community, primarily under the rubric of dueling bandits (e.g., (Yue et al., 2012) and online ranking in the Plackett-Luce (PL) discrete choice model (Chen et al., 2018; Saha & Gopalan, 2019; Ren et al., 2018).

Our focus in this paper is to study the instance-dependent complexity of learning the (near) best item in a subset-wise PL feedback model by which we mean the following. Each item has an a priori unknown PL weight parameter, and every time a subset of alternatives is selected, an item or items sampled from the PL probability distribution over the subset are observed by the learner. Given a tolerance ϵ and confidence level δ, the learner faces the task of sequentially playing subsets of items, and stopping and finding an ϵ-optimal arm, i.e., an arm i whose PL parameter satisfies $\theta_i \geq \max_j \theta_j - \epsilon$, with probability of error at most δ.

Existing work on best arm learning in PL models, e.g., (Saha & Gopalan, 2019), focuses on attaining the worst-case or *instance-independent* sample complexity of learning an approximately best item. By this, we mean that the typical goal is to design algorithms that terminate in a number of rounds bounded by a function of only ϵ, δ and the number of arms n, typically of the form $O\left(\frac{n}{\epsilon^2}\log\left(\frac{1}{\delta}\right)\right)$ rounds. Such worst-case results, though significantly novel, suffer from two weaknesses: (1) The termination time guarantees become vacuous in the setting where an exact best arm is sought ($\epsilon = 0$), and (2) The guarantees do not reflect the fact that some problem instances, in terms of their items' PL parameters, are easier than others to learn, e.g., the instance with parameters $(\theta_1, \ldots, \theta_n) = (1, 0.01, \ldots, 0.01)$ ought to be much easier than $(1, 0.99, \ldots, 0.99)$ since item 1 is a distinctly clearer winner than in the latter case. In this paper, we set ourselves the more challenging objective of quantifying and attaining sample complexity that depends on the inherent 'hardness' of the PL instance. In this context, we make the following contributions:

(1) We give the first instance-optimal algorithm for the problem of (ϵ, δ)-PAC learning a best item in a PL model when subsets of a fixed size can be tested in each round. This is accomplished by building a novel wrapper algorithm (Alg. 1) around an (ϵ, δ)-PAC learning algorithm used as

[1]Indian Institute of Science, Bangalore, India. Correspondence to: Aadirupa Saha <aadirupa@iisc.ac.in>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

a subroutine that we designed (Alg. 5). We also provide a matching instance-dependent lower bound on the sample complexity of any algorithm, to establish the optimality of our algorithm (Thm. 3,4,7).

(2) When richer, m length rank-ordered information is observed per subsetwise query, we show the optimal instance-dependent sample complexity lower bound is much smaller than just with the winner feedback case (Thm. 8). We also propose an optimal algorithm for this setting (Alg. 8) with an $\frac{1}{m}$-factor improved sample complexity guarantee which is shown to be optimal (Thm. 5).

(3) We also study the fixed-budget version of the best-item learning problem, where a learning horizon of Q rounds is specified instead of a desired confidence level δ, and the performance measure of interest is the probability of error in identifying a best arm. We give an algorithm for learning the best item of a Plackett-Luce instance under a fixed budget with general m-way ranking feedback (Alg. 8, Thm. 12), and also prove an instance-dependent lower bound for it (Thm. 11).

Our theoretical findings are also supported with numerical experiments on different datasets.

Related work. For classical multiarmed bandits setting, there is a well studied literature on PAC-arm identification problem (Even-Dar et al., 2006; Audibert & Bubeck, 2010; Kalyanakrishnan et al., 2012; Karnin et al., 2013; Jamieson et al., 2014), where the learner gets to see a noisy draw of absolute reward feedback of an arm upon playing a single arm per round. Some of the existing results on dueling bandits line of works also focuses on PAC learning from pairwise preference feedback for best arm identification problem (Yue & Joachims, 2011; Urvoy et al., 2013; Szörényi et al., 2015; Busa-Fekete et al., 2014a), or even more general problem objectives e.g. PAC top set recovery (Busa-Fekete et al., 2013; Mohajer et al., 2017; Chen et al., 2017), or PAC-ranking of items (Busa-Fekete et al., 2014b; Falahatgar et al., 2017), even in the feedback setup of noisy comparisons (Braverman & Mossel, 2008; Caragiannis et al., 2013). There are also very few recent developments that focuses on learning for subsetwise feedback in an online setup (Sui et al., 2017; Brost et al., 2016; Saha & Gopalan, 2018a; 2019; Ren et al., 2018; Chen et al., 2018). Some of the existing work also explicitly consider the Plackett-Luce parameter estimation problem with subset-wise feedback but for offline setup only (Jang et al., 2017; Khetan & Oh, 2016). While most of the above work address the (ϵ,δ)-PAC recovery problem, i.e. finding an 'ϵ-approximation' of the desired (set of) item(s) with probability at least $(1-\delta)$, few of them also focuses of instant dependent PAC recovery guarantees where the sample complexity explicitly depends of the parameters of the underlying model, e.g. for classical multiarmed bandits (Audibert & Bubeck, 2010; Karnin et al., 2013; Kalyanakrishnan et al., 2012), or even for preference based bandits (Szörényi et al., 2015; Chen et al., 2018).

2. Problem Setup

Notation. We denote by $[n]$ the set $\{1, 2, ..., n\}$. For any subset $S \subseteq [n]$, let $|S|$ denote the cardinality of S. When there is no confusion about the context, we often represent (an unordered) subset S as a vector, or ordered subset, S of size $|S|$ (according to, say, a fixed global ordering of all the items $[n]$). In this case, $S(i)$ denotes the item (member) at the ith position in subset S. For any ordered set S, $S(i:j)$ denotes the set of items from position i to j, $i<j$, $\forall i,j \in [|S|]$. We denote by $\boldsymbol{\Sigma}_S = \{\sigma \mid \sigma$ is a permutation over items of $S\}$, where for any permutation $\sigma \in \Sigma_S$, $\sigma(i)$ denotes the element at the i-th position in $\sigma, i \in [|S|]$. We also denote by $\boldsymbol{\Sigma}_S^m$ the set of permutations of any m-subset of S, for any $m \in [k]$, i.e. $\Sigma_S^m := \Sigma_{S'}$ s.t. $S' \subseteq S$, $|S'| = m$. $\mathbf{1}(\varphi)$ is generically used to denote an indicator variable that takes the value 1 if the predicate φ is true, and 0 otherwise. $x \vee y$ denotes the maximum of x and y, and $Pr(A)$ is used to denote the probability of event A, in a probability space.

Definition 1 (Plackett-Luce probability model)**.** *A Plackett-Luce probability model, specified by positive parameters $(\theta_1, \ldots, \theta_n)$, is a collection of probability distributions $\{Pr(\cdot|S) : S \subset [n], S \neq \emptyset\}$, where for each non-empty subset $S \subseteq [n]$, $Pr(i|S) = \frac{\theta_i \mathbf{1}(i\in S)}{\sum_{j\in S}\theta_j}$ $\forall 1 \le i \le n$. The indices $1, \ldots, n$ are referred to as 'items' or 'arms' .*

Since the Plackett-Luce probability model is invariant to positive scaling of its parameters $\boldsymbol{\theta} \equiv (\theta_i)_{i=1}^n$, we make the standard assumption that $\max_{i\in[n]} \theta_i = 1$.

An online learning algorithm is assumed to interact with a Plackett-Luce probability model over n items (the 'environment') as follows. At each round $t = 1, 2, \ldots$, the algorithm decides to either (a) terminate and return an item $I \in [n]$, or (b) play (test) a subset $S_t \subset [n]$ of k distinct items, upon which it receives stochastic feedback whose distribution is governed by the probability distribution $Pr(\cdot|S_t)$. We specifically consider the following structures for feedback received upon playing a subset S:

1. **Winner feedback:** The environment returns a single item J drawn independently from the probability distribution $Pr(\cdot|S)$ where $Pr(J=j|S) = \frac{\theta_j}{\sum_{k\in S}\theta_k}$ $\forall j \in S$.

2. **Top-m Ranking feedback** $(1 \le m \le k-1)$**:** Here, the environment returns an ordered list of m items sampled without replacement from the Plackett-Luce probability model on S. More formally, the environment returns a partial ranking $\boldsymbol{\sigma} \in \boldsymbol{\Sigma}_S^m$, drawn from the probability distribution $Pr(\boldsymbol{\sigma} = \sigma|S) = \prod_{i=1}^m \frac{\theta_{\sigma^{-1}(i)}}{\sum_{j\in S\setminus\sigma^{-1}(1:i-1)}\theta_j}$, $\sigma \in \boldsymbol{\Sigma}_S^m$. This can also be seen as picking an item $\boldsymbol{\sigma}^{-1}(1) \in S$ accord-

ing to *Winner feedback* from S, then picking $\boldsymbol{\sigma}^{-1}(2)$ from $S \setminus \{\boldsymbol{\sigma}^{-1}(1)\}$, and so on for m times. When $m = 1$, Top-m Ranking feedback is the same as Winner feedback.

Definition 2 ((ϵ, δ)-PAC or fixed-confidence algorithm)**.** *An online learning algorithm is said to be (ϵ, δ)-PAC with termination time bound Q if the following holds with probability at least $1 - \delta$ when it is run in a Plackett-Luce model: (a) it terminates within Q rounds (subset plays), (b) the returned item I is an ϵ-optimal item: $\theta_I \geq \max_{i \in [n]} \theta_i - \epsilon = 1 - \epsilon$. (Probability is over both the environment and the algorithm.)*

By the *sample complexity* of an (ϵ, δ)-PAC online learning algorithm $\mathcal{A}$ for a Plackett-Luce instance $\boldsymbol{\theta} \equiv (\theta_i)_{i=1}^n$ and playable subset size k, we mean the smallest possible termination time bound Q for the algorithm when run on $\boldsymbol{\theta}$. We use the notation $N_{\mathcal{A}}(\epsilon, \delta) \equiv N_{\mathcal{A}}(\epsilon, \delta, \boldsymbol{\theta}, n, k)$ to denote this sample complexity. We aim to design (ϵ, δ)-PAC algorithms with as small a value of sample complexity as possible, depending on the number of items n, the playable subset size k, approximation error ϵ, confidence δ, and most importantly, the Plackett-Luce model parameters $(\theta_i)_{i=1}^n$. We also assume item 1 is optimal: $\theta_1 = \max_{i \in [n]} \theta_i = 1$, and $\Delta_i = \theta_1 - \theta_i$ for any $i \in [n]$.

3. Instance-dependent regret for *Probably-Correct-Best-Item* problem

3.1. Prelude: An algorithm for $\epsilon = 0$

For clarity of exposition, we first describe the design of a $(0, \delta)$-PAC or *Probably-Correct-Best-Item* learning algorithm, i.e., an algorithm that attempts to learn the unique best item in a Plackett-Luce model when such an item exists[1]: $1 = \theta_1 > \max_{i \geq 2} \theta_i$. This is then generalised in the next section to an online learning algorithm that is (ϵ, δ)-PAC.

High-level idea behind algorithm design. The algorithm we propose (*PAC-Wrapper*) is based on using an (ϵ, δ)-PAC-algorithm known to have (expected) termination time bounded in terms of ϵ and δ (a 'worst' case termination guarantee not necessarily dependent on instance parameters) as an underlying black-box subroutine. The wrapper algorithm uses the black-box repeatedly, with successively more stringent values of ϵ and δ, to eliminate suboptimal arms in a phased manner. The termination analysis of the algorithm shows that any suboptimal arm $i \in [n] \setminus \{1\}$ survives for about $O\Big(\frac{1}{\Delta_i^2} \ln \frac{k}{\delta}\Big)$ rounds before being eliminated, which leads to the desired bound of $O\bigg(\sum_{i=2}^n \frac{1}{\Delta_i^2} \ln \frac{k}{\delta}\bigg)$ on algorithm's run time performance (with high probability $(1-\delta)$) (Thm. 3).

[1]When there is more than one best item the problem of finding a best item with confidence is not well-defined.

Algorithm description. The *PAC-Wrapper* algorithm we propose (Alg. 1) runs in phases indexed by $s = 1, 2, \ldots$, where each phase s comprises of the following steps.

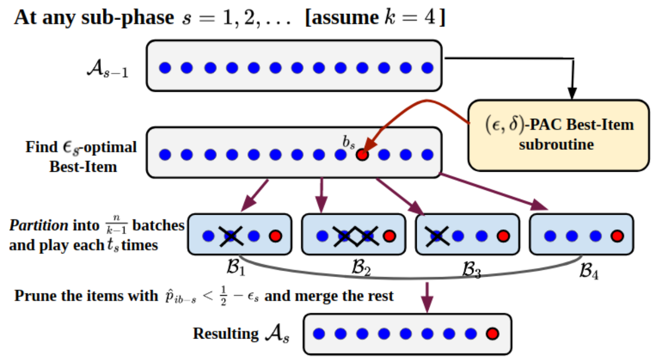

Figure 1. A sample run of Alg. 1 at any sub-phase s with the set of surviving arms $\mathcal{A}_{s-1}$: **Step 1.** The algorithm finds a (ϵ_s, δ_s)-PAC item b_s, where $\epsilon_s = \frac{1}{2^{s+2}}$ and $\delta_s = \frac{\delta}{40s^3}$. **Step 2.** It partitions $\mathcal{A}_{s-1}$ into $B_s = \left\lceil \frac{\mathcal{A}_{s-1}}{k-1} \right\rceil$ groups $\mathcal{B}_1, \ldots \mathcal{B}_{B_s}$ of size k, each containing b_s, and plays each for $t_s = \frac{2k}{\epsilon_s^2} \ln \frac{k}{\delta_s}$ times. **Step 3.** Based on the received feedback of t_s plays, the algorithm updates the empirical pairwise probability $\hat{p}_{ij}$ of each item pair (i, j) within a group $\mathcal{B}$ by applying *Rank-Breaking*, and discards any item $i \in \mathcal{B}$ with $\hat{p}_{ib_s} < \frac{1}{2} - \epsilon_s$. The rest of the surviving items are then combined to $\mathcal{A}_s$, and the algorithm recurses to $s + 1$.

Step 1: Finding a good reference item. It first calls an (ϵ_s, δ_s)-PAC subroutine (described in Sec. 3.4 for completeness) with $\epsilon_s = \frac{1}{2^{s+2}}$ and $\delta_s = \frac{\delta}{120s^3}$ to obtain a 'reasonably good item' b_s—an item that is likely within an ϵ_s margin of the Best-Item with probability at least $(1 - \delta_s)$) and thus a potential Best-Item. For this we design a new sequential elimination-based algorithm (Alg. 5 in Appendix A.3), and argue that it finds such a (ϵ_s, δ_s)-PAC 'good item' with *instance-dependent* sample complexity (Thm. 6), which is crucial in the overall analysis. This is an improvement upon the instance-agnostic Algorithm 6 of (Saha & Gopalan, 2019) whose sample complexity guarantee is not strong enough to be used along with the wrapper.

Step 2: Benchmarking items against the reference item. After obtaining a candidate good item, the algorithm divides the rest of the current surviving arms into equal-sized groups of size $k - 1$, say the groups $\mathcal{B}_1, \ldots, \mathcal{B}_{B_s}$, and 'stuffs' the good 'probe' item b_s into each group, creating $B_s = \left\lceil \frac{\mathcal{A}_{s-1}}{k-1} \right\rceil$ item groups of size k (the Partition subroutine, Algorithm 2, Appendix A.2). It then plays each group $\mathcal{B}_b$, $b \in [B_s]$ for a total of $t_s = \frac{2\hat{\Theta}_S}{\epsilon_s^2} \ln \frac{k}{\delta_s}$ rounds, where $\hat{\Theta}_S$ denotes a 'near-accurate' relative score estimate of the Plackett-Luce model for the set $\mathcal{B}_b$–we use the subroutine *Score-Estimate* for estimating $\hat{\Theta}_S$ (see Alg. 3, Thm. 13 in Appendix A.2). From the winner data obtained in this process, it updates the empirical pairwise win count w_i of each item within any batch $\mathcal{B}_b$ by applying a rank-breaking idea (see Alg. 4, Appendix A.2) .

Step 3: Discarding items weaker than the reference item. Finally, from each group $\mathcal{B}_b$, the algorithm eliminates all arms whose empirical pairwise win frequency over the probe item b_s is less than $\frac{1}{2} - \epsilon_s$ (i.e. $\forall i \in \mathcal{B}_b$ for which $\hat{p}_{ib_s} < \frac{1}{2} - \epsilon_s$, $\hat{p}_{ij}$ being the empirical pairwise preference of item i over j obtained via *Rank-Breaking*). The next phase then begins, unless there is only one surviving item left, which is output as the candidate Best-Item. Pointers to the 4 subroutines used in the overall algorithm are as below.

(1). (ϵ, δ)***-PAC Best-Item*** **subroutine:** Given $\epsilon, \delta \in (0, 1)$, this finds an (ϵ, δ)-Best-Item in $O\Big(\frac{\Theta_{[k]}}{\epsilon^2} \ln \frac{k}{\delta}\Big)$ samples, where $\Theta_{[k]} = \max\limits_{S \subseteq [n] \mid |S|=k} \sum_{i \in S} \theta_i$ (See Alg. 5, Thm. 6 in Appendix A.3).

(2). ***Rank-Breaking*** **subroutine:** This is a procedure of deriving pairwise comparisons from multiwise (subsetwise) preference information (Soufiani et al., 2014; Khetan & Oh, 2016). (See Alg. 4, Appendix A.2).

(3). ***Score-Estimate*** **subroutine:** Given a set S and a reference item $b \in [n]$, this estimates the relative Plackett-Luce scores of the set w.r.t. b (see Alg. 3, Appendix A.2).

(4). ***Partition*****:** This partitions a given set of items into equally sized batches (See Alg. 2, Appendix A.2).

Fig. 1 graphically depicts a sample run of a sub-phase s (for $k = 4$). Note that as the playable subset size is k, we need to specially treat the final few sub-phases when the number of surviving arms (i.e. $|\mathcal{A}_s|$) falls below k (Lines 22-31 in Alg. 1). The complete algorithm is given in Appendix A.1.

Theorem 3 (*PAC-Wrapper*$(0, \delta)$-PAC sample complexity bound with Winner feedback)**.** *With probability at least* $(1 - \delta)$*,* $\mathcal{A}$ *as PAC-Wrapper (Algorithm 1) returns the Best-Item with sample complexity* $N_{\mathcal{A}}(0, \delta) = O\bigg(\frac{\Theta_{[k]}}{k} \sum_{i=2}^n \max\Big(1, \frac{1}{\Delta_i^2}\Big) \ln \frac{k}{\delta} \Big(\ln \frac{1}{\Delta_i}\Big)\bigg)$, *where* $\Theta_{[k]} := \max_{S \subseteq [n] \mid |S|=k} \sum_{i \in S} \theta_i$.

Remark 1. As $\Theta_{[k]} \le k$, *PAC-Wrapper* takes $O(\frac{1}{\Delta_i^2}) \ln \frac{1}{\delta}$ rounds to eliminate all suboptimal items with confidence δ. However, the dependence of the upper bound on $\Theta_{[k]}$ implies a $1/k$ factor gain in sample complexity when the underlying instance is 'easy'. Indeed, when $\Theta_{[k]} = O(1)$, e.g., in an instance where $\theta_1 \approx 1$ and $\theta_i \approx 0 \; \forall i \neq 1$, then the algorithm just takes $O(\frac{1}{k\Delta_i^2}) \ln \frac{1}{\delta}$ time to terminate. On the other hand, if $1 = \theta_1 > \theta_i \approx 1$, then $\Theta_{[k]} = \Omega(k)$ which gives the worst case orderwise complexity.

Proof sketch The proof of Thm. 3 is based on the following claims:

Claim-1: At any sub-phase $s = 1, 2, \ldots$, the Best-Item a^* is likely to beat the (ϵ_s, δ_s)-PAC item b_s by sufficiently high margin with probability at least $(1 - \frac{\delta}{20})$, and hence is never discarded (Lem. 19).

Claim-2: Let $[n]_r := \{i \in [n] : \frac{1}{2^r} \le \Delta_i < \frac{1}{2^{r-1}}\}$, and we denote the set of surviving arms in $[n]_r$ at s^{th} sub-phase by $\mathcal{A}_{r,s}$, i.e. $\mathcal{A}_{r,s} = [n]_r \cap \mathcal{A}_s$, for any $s = 1, 2, \ldots$. Then with probability at least $(1 - \frac{19\delta}{20})$, any such set $\mathcal{A}_{r,s}$ reduces at a constant rate once $s \ge r$, $r = 1, \ldots, \log_2(\Delta_{\min})$ (Lem. 20)—this ensures that all suboptimal elements get eventually discarded after they are played sufficiently often.

Claim-3: The number of occurrences of any sub-optimal item $i \in [n] \setminus \{1\}$ before it gets discarded away is proportional to $O\Big(\frac{1}{\Delta_i^2} \ln \frac{k}{\delta}\Big)$. Combining this over all arms yields the desired sample complexity. Details of the proof is given in Appendix A.4. $\square$.

3.2. An algorithm for general $\epsilon > 0$

It is straightforward to extend the $(0, \delta)$-PAC guarantee for *PAC-Wrapper* to get a more general (ϵ, δ)-PAC algorithm for any given $\epsilon \in [0, 1]$. The idea is to simply execute the algorithm as originally specified until (and if) it reaches a phase s such that ϵ_s falls below the given tolerance ϵ (i.e. $\epsilon_s \le \epsilon$), at which point the algorithm can stop right after calling the subroutine (ϵ, δ)-*PAC Best-Item* and output the item b_s returned by it. The full algorithm is given in Appendix A.5 for the sake of brevity.

Theorem 4 (*PAC-Wrapper* (ϵ, δ)-PAC sample complexity bound with Winner feedback)**.** *For any* $\epsilon \in [0, 1]$*, with probability at least* $(1 - \delta)$*,* $\mathcal{A}$ *as PAC-Wrapper (Algorithm 1) returns the* ϵ*-Best-Item (see Defn. 2) with sample complexity* $N_{\mathcal{A}}(\epsilon, \delta) = O\bigg(\frac{\Theta_{[k]}}{k} \sum_{i=2}^n \max\Big(1, \frac{1}{\max(\Delta_i, \epsilon)^2}\Big) \ln \frac{k}{\delta} \Big(\ln \frac{1}{\max(\Delta_i, \epsilon)}\Big)\bigg)$.

Discussion. To the best of our knowledge, this is the first (ϵ, δ)-PAC learning algorithm for the Plackett-Luce model with general multi-wise comparisons with an *item-wise* instance-dependent sample complexity bound. For $\epsilon > 0$, this is order-wise stronger than the best known worst-case (instance-independent) upper bound of $O\left(\frac{n}{\epsilon^2} \log\left(\frac{k}{\delta}\right)\right)$ (Saha & Gopalan, 2019), since $\max(\Delta_i, \epsilon)^2 \ge \epsilon^2$. Thus *PAC-Wrapper* is provably able to adapt to the hardness of the Plackett-Luce instance $\boldsymbol{\theta}$ to stop early in case the instance is 'well-separated'. Note that for dueling bandits ($k = 2$), our result strictly improves order-wise upon the $\tilde{O}\Big(n \cdot \max_{i \ge 2} \frac{1}{\max(\Delta_i, \epsilon)^2}\Big)$ sample complexity[2] of the best known (ϵ, δ)-PAC algorithm (PLPAC) (Szörényi et al., 2015)—which can be worse by a factor of n for many instances. For example, consider an instance having one 'strong' suboptimal item, say $j \in [n] \setminus \{1\}$ with $\Delta_j \approx 0$, but $\Omega(n)$ many extremely 'weak' items with $\Delta_i \approx 1$; our

[2]Notation $\tilde{O}(\cdot)$ hides polylogarithmic factors in $\epsilon, \delta, \Delta_i, n, k$.

sample complexity bound is just $\tilde{O}\Big(\frac{1}{2\Delta_j^2}\ln\frac{1}{\delta} + \frac{n}{2}\ln\frac{1}{\delta}\Big)$, whereas that of PLPAC is $O\Big(\frac{n}{\Delta_j^2}\ln\frac{n}{\Delta_j\delta}\Big)$.

3.3. PAC learning in the Plackett-Luce model with Top-m Ranking feedback

Main Idea. Algorithmically, the key modification to make is in the *Rank-Breaking* subroutine of *PAC-Wrapper*, which now uses a rank-ordered list of m feedback items to output all possible rank-broken comparison pairs. The essence of the $\frac{1}{m}$ factor improvement in the sample complexity over Winner feedback lies in the fact that this naturally gives rise to $O(m)$ times additional number of pairwise preferences in comparison to Winner feedback. Hence, it turns out to be sufficient to sample any batch $\mathcal{B}_b, \forall b \in [B_s]$ for only $O(\frac{1}{m})$ times compared to the earlier case, which finally leads to the improved sample complexity of *PAC-Wrapper* for Top-m Ranking feedback. The full description of Alg. 7 is given in Appendix A.7 for the sake of brevity.

Theorem 5 (*PAC-Wrapper*: Sample Complexity for $(0,\delta)$-PAC Guarantee for Top-m Ranking feedback)**.** *With probability at least $(1-\delta)$, PAC-Wrapper (Algorithm 1) returns the Best-Item with sample complexity $N_{\mathcal{A}}(0,\delta) = O\bigg(\frac{\Theta_{[k]}}{k}\sum_{i=2}^n \max\Big(1, \frac{1}{m\Delta_i^2}\Big)\ln\frac{k}{\delta}\Big(\ln\frac{1}{\Delta_i}\Big)\bigg)$.*

Remark 2. Following the similar procedure as argued in Sec. 3.4, one can easily derive an (ϵ,δ)-PAC version of *PAC-Wrapper (for Top-m Ranking feedback)* as well, and a similar guarantee as that of Thm. 4 with a reduction factor $1/m$. We omit the details in the interest of space.

3.4. (ϵ,δ)-PAC subroutine (used in the main algorithm, *PAC-Wrapper*, i.e. in Alg. 1, 5 or 7)

We briefly describe here the core (ϵ,δ)-PAC subroutine used in algorithms 1 and 7 to find an ϵ Best-Item with high probability $(1-\delta)$ in an instance-dependent way (full details are available in Appendix A.3): The algorithm (ϵ,δ)-*PAC Best-Item* first divides the set of n items into batches of size k, then plays each group sufficiently long enough until a single item of that group stands out as the empirical winner in terms of its empirical pairwise advantage over the rest (again estimated though *Rank-Breaking*). It then just retains this empirical winner for every group and recurses on the set of surviving winners until only a single item is left, which is declared as the (ϵ,δ)-PAC item.

Theorem 6 ((ϵ,δ)-*PAC Best-Item*: Correctness and Sample Complexity with Top-m Ranking feedback)**.** *For any $\epsilon \in \big(0,\frac{1}{8}\big]$ and $\delta \in (0,1)$, with probability at least $(1-\delta)$, (ϵ,δ)-PAC Best-Item (Algorithm 5) returns an item $b_s \in [n]$ satisfying $p_{b_s 1} > \frac{1}{2}-\epsilon$ with sample complexity $O\Big(\frac{n\Theta_{[k]}}{k}\max\big(1,\frac{1}{m\epsilon^2}\big)\log\frac{k}{\delta}\Big)$, where $\Theta_{[k]} := \max_{S\subseteq[n],|S|=k}\sum_{i\in S}\theta_i$.*

Remark 3. The best item-finding subroutine we develop, along with the corresponding analysis, is an improvement over Alg. 6 of (Saha & Gopalan, 2019) which had k instead of ${\Theta_{[k]}}_{[k]} \le k$ here. The improvement is especially pronounced for instances where $\Theta_{[k]} = O(1)$ (e.g. where $\theta_{a^*} \to 1$ and for all $i \in [n]\setminus\{a^*\}$, $\theta_i \to 0$ etc.). Note that this is an artefact of the *adaptive nature* of our proposed algorithm (Alg. 5) which samples each batch adaptively for just sufficiently enough times before discarding out the weakest $(k-1)$ items (see Line 11), whereas (Saha & Gopalan, 2019) sample each batch for a fixed $O\Big(\frac{k}{\epsilon^2}\ln\frac{k}{\delta}\Big)$ times irrespective of the empirical outcomes, leading to a *worse, instance independent sample complexity*.

4. Instance-dependent lower bounds on sample complexity

We here derive information-theoretic lower bounds on sample complexity for *Probably-Correct-Best-Item* problem. We first show a lower bound of $\Omega\Big(\sum_{i=2}^n\frac{\theta_i\theta_1}{\Delta_i^2}\ln\big(\frac{1}{\delta} + \frac{n}{k}\ln\frac{1}{\delta}\big)\Big)$ with Winner feedback implying that the sample complexity of *PAC-Wrapper* (Thm. 3) is tight upto logarithmic factors. We then analyze the lower bound for Top-m Ranking feedback and show an $\frac{1}{m}$-factor improvement in the sample complexity lower bound, establishing the optimality (up to logarithmic factors) of our *PAC-Wrapper* algorithm for Top-m Ranking feedback (see Alg. 7 and Thm. 5).

4.1. Lower bound for Winner feedback

Theorem 7 (Sample complexity lower bound: $(0,\delta)$-PAC or *Probably-Correct-Best-Item* with Winner feedback)**.** *Given $\delta \in [0,1]$, suppose $\mathcal{A}$ is an online learning algorithm for Winner feedback which, when run on any Plackett-Luce instance, terminates in finite time almost surely, returning an item I satisfying $Pr(\theta_I = \max_i\theta_i) > 1-\delta$. Then, on any Plackett-Luce instance $\theta_1 > \max_{i\ge 2}\theta_i$, the expected number of rounds it takes to terminate is $\Omega\Big(\max\Big(\sum_{i=2}^n\frac{\theta_i\theta_1}{\Delta_i^2}\ln\frac{1}{\delta}, \frac{n}{k}\ln\frac{1}{\delta}\Big)\Big)$.*

Proof sketch. We employ the measure-change technique of Kaufmann et al (Kaufmann et al., 2016) (see Lem. 26, Appendix) for lower bounds on the PAC sample complexity for standard multiarmed bandits (MAB). The novelty of our proof lies in mapping their result to our setting: For our case each MAB instance corresponds to an instance of the BB-PL problem with the arm set containing all subsets of $[n]$ of size k: $A = \{S = (S(1),\ldots S(k)) \subseteq [n]\}$.

We now consider any general true $\text{PL}(n,\boldsymbol{\theta})$ problem instance $\text{PL}(n,\boldsymbol{\theta}^1) : \theta_1^1 > \theta_2^1 \ge \ldots \ge \theta_n^1$, and corresponding to each suboptimal item $a \in [n] \setminus \{1\}$, we define an alternative problem instance PL$(n, \boldsymbol{\theta}^a)$: $\theta_a^a = \theta_1^1 + \epsilon$; $\theta_i^a = \theta_i^1, \ \forall i \in [n] \setminus \{a\}$, for some $\epsilon > 0$. Then, applying Lemma 26 on every pairs of problem instances $(\boldsymbol{\theta}^1, \boldsymbol{\theta}^a)$, and suitably upper bounding the KL-divergence terms we arrive at $n-1$ constraints of the form:

$$\ln \frac{1}{2.4\delta} \leq \sum_{S \in A | a \in S} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(\tau_A)] KL(p_S^1, p_S^a)$$

$$\leq \sum_{S \in A | a \in S} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(\tau_A)] \frac{\Delta_a'^2}{\theta_S^1(\theta_1^1 + \epsilon)}, \forall a \in [n] \setminus \{1\}$$

Since the total sample complexity of $\mathcal{A}$ being $\mathcal{N}(0, \delta) = \sum_{S \in A} N_S$ (here N_S is the number of plays of subset S by $\mathcal{A}$), the problem of finding the sample complexity lower bound actually reduces to solving the (primal) linear programming (LP) problem:

$$\textbf{Primal LP (P):} \ \min_{S \in A} \sum_{S \in A} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S]$$

$$\text{such that} \sum_{S \in A | a \in S} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S] \frac{\Delta_a'^2}{\theta_S^1(\theta_1^1 + \epsilon)} \geq \ln \frac{1}{2.4\delta}, \quad \forall a \in [n] \setminus \{1\}$$

However above has $O\binom{n}{k}$ many optimization variables (precisely $\mathbf{E}_{\boldsymbol{\theta}^1}[N_S]$s), so we instead solve the *dual* LP to reach the desired bound. Lastly the $\Omega\Big(\frac{n}{k} \ln \frac{1}{\delta}\Big)$ term in the lower bound arises as any learning algorithm must at least test each item a constant number of times via k-wise subset plays before judging it optimality which is the bare minimum sample complexity the learner has to incur (Chen et al., 2018). The complete proof is given in Appendix B.1. $\square$.

4.2. Lower bound for Top-m Ranking feedback

Theorem 8 (Sample complexity Lower Bound: $(0, \delta)$-*Probably-Correct-Best-Item* with Top-m Ranking feedback)**.** *Suppose $\mathcal{A}$ is an online learning algorithm for Top-m Ranking feedback which, given $\delta \in [0, 1]$ and run on any Plackett-Luce instance, terminates in finite time almost surely, returning an item I satisfying $Pr(\theta_I = \max_i \theta_i) > 1 - \delta$. Then, on any Plackett-Luce instance $\theta_1 > \max_{i \geq 2} \theta_i$, the expected number of rounds it takes to terminate is* $\Omega\bigg(\max\Big(\frac{1}{m}\sum_{i=2}^n \frac{\theta_i \theta_1}{\Delta_i^2} \ln\big(\frac{1}{\delta}\big), \frac{n}{k} \ln \frac{1}{\delta}\Big)\bigg)$.

Proof sketch. The crucial fact used here is owning to the chain rule for KL-divergence, the KL divergence for Top-m Ranking feedback is m times larger than that of just with Winner feedback: $KL(p_S^1, p_S^a) = KL(p_S^1(\sigma_1), p_S^a(\sigma_1)) + + \sum_{i=2}^m KL(p_S^1(\sigma_i \mid \sigma(1:i-1)), p_S^a(\sigma_i \mid \sigma(1:i-1)))$, where we abbreviate $\sigma(i)$ as σ_i and $KL(P(Y \mid X), Q(Y \mid X)) := \sum_x Pr\Big(X = x\Big)\big[KL(P(Y \mid X = x), Q(Y \mid X = x))\big]$ denotes the conditional KL-divergence. Using this and the upper bound on the KL divergences for Winner feedback setup as derived for Thm. 7, we get that in this case $KL(p_S^1, p_S^a) \leq \frac{m\Delta_a'^2}{\theta_S^1(\theta_1^1+\epsilon)}, \forall a \in [n] \setminus \{1\}$, where lies the crux of the $\frac{1}{m}$-factor improvement in the sample complexity lower bound compared to Winner feedback. The lower bound now can be derived following a similar procedure that of Thm. 7. Details are given in B.1. $\square$.

5. The *Fixed-Sample-Complexity* Learning Problem

This section studies the problem of finding the Best-Item within a maximum allowed number of queries Q, with minimum possible probability of misidentification. Note the algorithms for *Probably-Correct-Best-Item* setting cannot be used here as they do not take the total sample complexity Q as input; also, simply terminating such algorithms with a suitable δ after Q runs may not necessarily be optimal. We present results for the general Top-m Ranking feedback.

5.1. Lower Bound: *Fixed-Sample-Complexity* setting

We derive an instance-dependent lower bound on error probability in which the problem complexity depends on the complexity term $\Big(\sum_{a=2}^n \frac{\theta_a}{\Delta_a^2}\Big)^{-1}$, unlike the case for our first objective (*Probably-Correct-Best-Item*), which depends on the gap parameter $\frac{1}{\Delta_a^2}$, $\forall \in [n] \setminus \{1\}$. We first define a natural consistency or 'non-trivial learning' property for any best-arm algorithm given a fixed budget of Q:

Definition 9 (*Budget-Consistent* Best-Item Identification Algorithm)**.** *An online algorithm $\mathcal{A}$, taking as input a sample complexity budget Q, terminating within Q rounds and outputting an item $I \in [n]$, is said to be Budget-Consistent if, for every Plackett-Luce instance $\boldsymbol{\theta} \equiv (\theta_i)_{i=1}^n$ with a unique best item $a^*(\boldsymbol{\theta}) := \arg\max_{i \in [n]} \theta_i$, it satisfies $Pr_{\boldsymbol{\theta}}\big(I = a^*(\boldsymbol{\theta})\big) \geq 1 - \exp(-f(\boldsymbol{\theta}) \cdot Q)$ when run on $\boldsymbol{\theta}$, where $f : [0,1]^n \mapsto \mathbb{R}_+$ is an instance-dependent function mapping every Plackett-Luce instance to a real number.*

Informally, a *Budget-Consistent* algorithm picks out the best arm in a Plackett-Luce instance with arbitrarily low error probability given enough rounds Q. We next define the notion of a *Order-Oblivious* or *label-invariant* algorithm before stating our main lower bound result.

Definition 10 (Order obliviousness or label invariance)**.** *A Budget-Consistent algorithm $\mathcal{A}$ is said to be Order-Oblivious if its output is insensitive to the specific labelling of items, i.e., if for any PL model $(\theta_1, \ldots, \theta_n)$, bijection $\phi : [n] \to [n]$ and any item $I \in [n]$, it holds that $Pr(\mathcal{A}$ outputs $I \,|\, (\theta_1, \ldots, \theta_n)) = Pr(\mathcal{A}$ outputs $I \,|\, (\theta_{\phi(1)}, \ldots, \theta_{\phi(n)}))$, where $Pr(\cdot \,|(\alpha_1, \ldots, \alpha_n))$ denotes the probability distribution on the trajectory of $\mathcal{A}$ induced by the PL model*

$(\alpha_1, \ldots, \alpha_n)$.

Theorem 11 (Confidence lower bound in fixed sample complexity Q for Top-m Ranking feedback)**.** *Let $\mathcal{A}$ be a Budget-Consistent and Order-Oblivious algorithm for identifying the Best-Item under Top-m Ranking feedback. For any Plackett-Luce instance $\boldsymbol{\theta}$ and sample size (budget) Q, its probability of error in identifying the best arm in $\boldsymbol{\theta}$ satisfies $Pr_{\boldsymbol{\theta}}\left(I \neq \arg\max_{i \in [n]} \theta_i\right) = \Omega\left(\exp\left(-2mQ\tilde{\Delta}\right)\right)$, where the complexity parameter $\tilde{\Delta} := \left(\sum_{a=2}^{n} \frac{(\theta_a)^2}{\Delta_a^2}\right)^{-1}$.*

Remark 4. As expected, the error probability reduces with increasing feedback size m and budget Q. However a more interesting tradeoff lies in the instant dependent complexity term $\tilde{\Delta}$: for 'easy' instances where most of the suboptimal item have $\theta_a \to 0$ (i.e. $\Delta_a \to 1$), $\tilde{\Delta}$ shoots up, in fact attains $\tilde{\Delta} \to \infty$ in the limiting case where $\theta_a \to 0 \, \forall i \in [n \setminus \{1\}$. On the other hand, for 'hard' instances, where there exists even one suboptimal item $a \in [n] \setminus \{1\}$ with $\theta_a \approx 1$ (i.e. $\Delta_a \approx 0$), $\tilde{\Delta} \to 0$ raising the minimum error probability significantly, which indicates the hardness of the learning problem.

5.2. Proposed Algorithm for *Fixed-Sample-Complexity* setup: *Uniform-Allocation*

Main Idea. Our proposed algorithm *Uniform-Allocation* solves the problem with a uniform budget allocation rule: Since we are allowed to play sets of size k only, we divide the items into k-sized batches and eliminate the bottom half of the winning items once each batch is played sufficiently. The important parameter to tune is how long to play the batches. Given a fixed budget Q, since one does not have an idea about which batch the Best-Item lies in, a good strategy is to allocate the budget uniformly across all sets formed during the entire run of the algorithm, which can shown to be precisely $O(\frac{n+k\log_2 k}{k})$ sets, so we allocate a budget of $Q' = O\left(\frac{kQ}{n+k\log_2 k}\right)$ samples per batch.

Algorithm description. The algorithm proceeds in rounds, where in each round it divides the set of surviving items into batches of size k and plays each $Q' = \frac{(n+k)kQ}{2n^2\log_2 k}$ times. Upon this it retains only the top half of the winning arms, eliminating the rest forever. The hope here is that with 'enough' observed samples, the Best-Item always stays in the top half and never gets eliminated. The next round recurses on the remaining items, and the algorithm finally returns the only single element is left as the potential Best-Item. The pseudocode is moved to Appendix C.2.

Theorem 12 (*Uniform-Allocation*: Confidence bound for Best-Item identification with fixed sample complexity Q)**.** *Given a budget of Q rounds, Uniform-Allocation returns the Best-Item of PL$(n, \boldsymbol{\theta})$ with probability at least $1 - O\left(\log_2 n \exp\left(-\frac{mQ\Delta_{\min}^2}{16(2n+k\log_2 k)}\right)\right)$, where $\Delta_{\min} = \min_{i=2}^{n} \Delta_i$.*

Remark 5. Thm. 12 equivalently shows that with sample complexity at most $O\left(\frac{16(2n+k\log_2 k)}{m\Delta_{\min}^2}\ln\left(\frac{\log_2 n}{\delta}\right)\right)$, *Uniform-Allocation* returns the Best-Item with probability at least $(1-\delta)$. The bound is clearly optimal in terms of m and Q (comparing with Thm. 11), however it still remains an open problem to close the gap between the complexity term $\tilde{\Delta} = \left(\sum_{a=2}^{n} \frac{(\theta_a)^2}{\Delta_a^2}\right)^{-1}$ in the lower bound, vs. the $\left(\frac{n}{\Delta_{\min}^2}\right)^{-1}$ term that we obtained.

6. Experiments

This section reports numerical results of our proposed algorithm *PAC-Wrapper* (PW) on different Plackett-Luce environments. All reported performances are averaged across 50 runs. The default values of the parameters are set to be $k = 5$, $\epsilon = 0.01$, $\delta = 0.01$, $m = 1$ unless explicitly mentioned/tuned in the specific experimental setup. We compared our algorithm with the only existing benchmark algorithm *Divide-and-Battle* (DnB) (Saha & Gopalan, 2019) (even though, as described earlier, it does not apply to instance-optimal analysis, specifically for $\epsilon = 0$; this is reflected in our experimental results as well). We use 8 different PL environments (with different $\boldsymbol{\theta}$ parameters) for the purpose, their descriptions are moved to Appendix D.

Throughout this section, by the term *sample-complexity*, we mean the average (mean) termination time of the algorithms across multiple reruns (i.e. number of subsetwise queries performed by the algorithm before termination).

6.1. Results: *Probably-Correct-Best-Item* setting

Sample-Complexity vs Error-Margin (ϵ)**.** Our first set of experiments analyses the sample complexity ($\mathcal{N}(\epsilon, \Delta)$) of *PAC-Wrapper* with varying ϵ (keeping δ fixed at 0.1). As expected, Fig. 2 shows that the sample complexity increases with decreasing ϵ for both the algorithms. However, the interesting part is, for PW the sample complexity becomes almost constant beyond a certain threshold of ϵ (precisely when ϵ falls below $\Delta_{\min}$) in every case, whereas for DnB it keeps on scaling in $O(\frac{1}{\epsilon^2})$ irrespective of the 'hardness' of the underlying PL environment due to its non-adaptive nature—this is the region where we excel out. Also, note that the harder the dataset (i.e. the smaller its $\Delta_{\min}$), the smaller this threshold is, as follows from Thm. 4, which verifies the instance-adaptive nature of our PW algorithm as it terminates as soon as ϵ falls below $\Delta_{\min}$.

Itemwise sample complexity. This experiment reveals the survival time of the items (i.e. total number plays of an item before elimination) in *PAC-Wrapper* algorithm. The results in Fig. 3 clearly shows the inverse dependency of

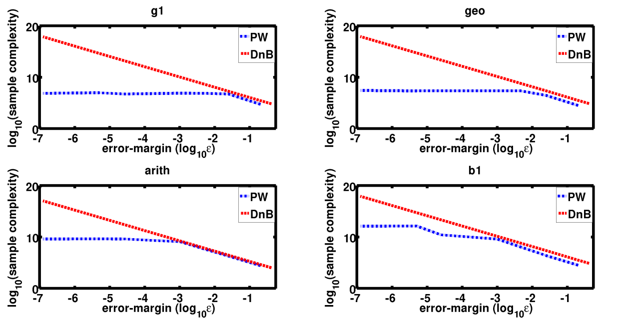

Figure 2. Sample-Complexity vs Error-Margin (ϵ) (both in log scale) of PW and DnB across 4 different problem instances.

the survival time of items w.r.t. their θ parameter, e.g. for **g4** dataset, the survival times of the items are categorized into 4 groups, highest for item 1, with items 2-6, 7-11, and 12-16 following it—justifying the $O\big(\frac{1}{\Delta_i^2}\big)$ survival times for each item i (in Thm. 3 or 5).

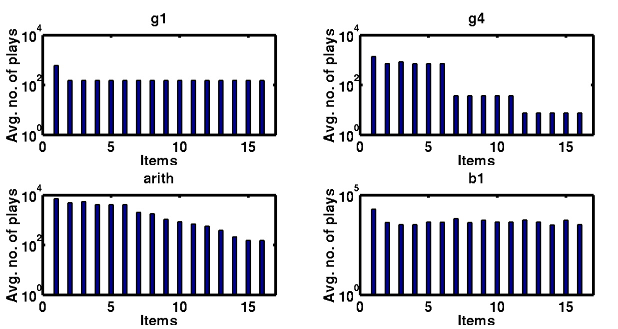

Figure 3. Survival time of different items (Itemwise sample complexity) in PW on 4 different problem instances.

Tradeoff: Sample-Complexity vs size of Top-ranking Feedback m. In this case we verified the flexibility of *PAC-Wrapper* for Top-m Ranking feedback (Alg. 7). We run it on different datasets with increasing size of top-ranking feedback (m). Again, justifying the claims of Thm. 5, Fig. 4 shows the sample complexity varies at a rate of $\frac{1}{m}$ (note that as m is doubled, sample complexity gets about halved), while rest of the parameters (i.e. k, δ, ϵ) are kept unchanged.

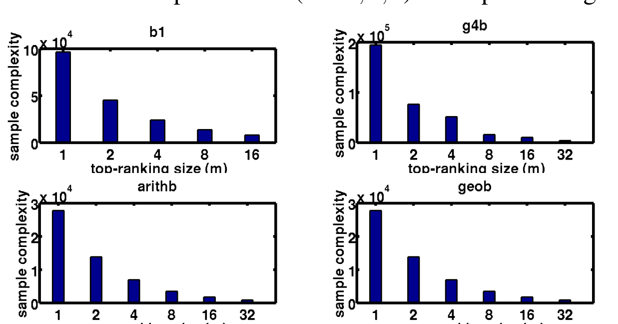

Figure 4. Sample-Complexity vs length of rank ordered feedback (m) of PW for 4 different problem instances.

6.2. Results: *Fixed-Sample-Complexity* setting

Success probability $(1-\delta)$ vs Sample-Complexity (Q). Finally we analysed the success probability $(1-\delta)$ of algorithm *Uniform-Allocation* (UA) for varying sample complexities (Q), keeping ϵ fixed at $(\Delta_{\min})/2$. Fig. 5 shows that the algorithm identifies the Best-Item with higher confidence with increasing Q—justifying its $O(\exp(-Q)$ error confidence rate as proved in Thm. 12. Note that **g4** being the easiest instance, it reaches the maximum success rate 1 at a much smaller Q, compared to the rest. By construction, DnB is not designed to operate in *Fixed-Sample-Complexity* setup, but due to lack of any other existing baseline, we still use it for comparison force terminating it if the specified sample complexity is exceeded, and as expected, here again it performs poorly in the lower sample complexity region.

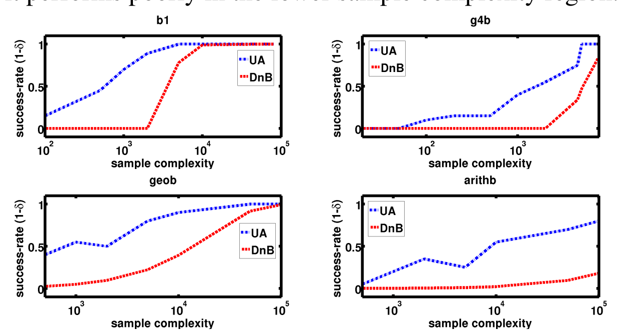

Figure 5. Comparative performances of PW and DnB in terms of Success probability $(1-\delta)$ vs Sample-Complexity (Q) across 4 different problem instances.

7. Conclusion and Future Work

Moving forward, it would be interesting to explore similar algorithmic and statistical questions in the context of other common subset choice models such as the Mallows model, Multinomial Probit, etc. It would also be of great practical interest to develop efficient algorithms for large item sets, especially when there is structure among the parameters to be exploited. One can also aim to develop instant dependent guarantees for other 'learning from relative feedback' objectives, e.g. PAC-ranking (Szörényi et al., 2015), top-set identification (Busa-Fekete et al., 2013) etc., both in fixed confidence as well as fixed budget setting.

Acknowledgements

The authors are grateful to Praneeth Netrapalli and the anonymous reviewers for valuable feedback. This work is supported by a Qualcomm Innovation Fellowship 2019-20, and the Indigenous 5G Test Bed project grant from the Dept. of Telecommunications, Government of India.

References

Audibert, J.-Y. and Bubeck, S. Best arm identification in multi-armed bandits. In *COLT-23th Conference on Learning Theory-2010*, pp. 13–p, 2010.

Boyd, S. and Vandenberghe, L. *Convex optimization*. Cambridge university press, 2004.

Braverman, M. and Mossel, E. Noisy sorting without resampling. In *Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 268–276. Society for Industrial and Applied Mathematics, 2008.

Brost, B., Seldin, Y., Cox, I. J., and Lioma, C. Multi-dueling bandits and their application to online ranker evaluation. *CoRR*, abs/1608.06253, 2016.

Busa-Fekete, R., Szorenyi, B., Cheng, W., Weng, P., and Hüllermeier, E. Top-k selection based on adaptive sampling of noisy preferences. In *International Conference on Machine Learning*, pp. 1094–1102, 2013.

Busa-Fekete, R., Hüllermeier, E., and Szörényi, B. Preference-based rank elicitation using statistical models: The case of mallows. In *Proceedings of The 31st International Conference on Machine Learning*, volume 32, 2014a.

Busa-Fekete, R., Szörényi, B., and Hüllermeier, E. Pac rank elicitation through adaptive sampling of stochastic pairwise preferences. In *AAAI*, pp. 1701–1707, 2014b.

Caragiannis, I., Procaccia, A. D., and Shah, N. When do noisy votes reveal the truth? In *Proceedings of the fourteenth ACM conference on Electronic commerce*, pp. 143–160. ACM, 2013.

Chen, X., Bennett, P. N., Collins-Thompson, K., and Horvitz, E. Pairwise ranking aggregation in a crowdsourced setting. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pp. 193–202. ACM, 2013.

Chen, X., Gopi, S., Mao, J., and Schneider, J. Competitive analysis of the top-k ranking problem. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1245–1264. SIAM, 2017.

Chen, X., Li, Y., and Mao, J. A nearly instance optimal algorithm for top-k ranking under the multinomial logit model. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 2504–2522. SIAM, 2018.

Even-Dar, E., Mannor, S., and Mansour, Y. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of machine learning research*, 7(Jun):1079–1105, 2006.

Falahatgar, M., Hao, Y., Orlitsky, A., Pichapati, V., and Ravindrakumar, V. Maxing and ranking with few assumptions. In *Advances in Neural Information Processing Systems*, pp. 7063–7073, 2017.

Freund, Y. and Schapire, R. E. Game theory, on-line prediction and boosting. In *COLT*, volume 96, pp. 325–332. Citeseer, 1996.

Graepel, T. and Herbrich, R. Ranking and matchmaking. *Game Developer Magazine*, 25:34, 2006.

Hofmann, K. et al. Fast and reliable online learning to rank for information retrieval. In *SIGIR Forum*, volume 47, pp. 140, 2013.

Jamieson, K., Malloy, M., Nowak, R., and Bubeck, S. lil' ucb : An optimal exploration algorithm for multi-armed bandits. In Balcan, M. F., Feldman, V., and Szepesvari, C. (eds.), *Proceedings of The 27th Conference on Learning Theory*, volume 35 of *Proceedings of Machine Learning Research*, pp. 423–439. PMLR, 2014.

Jang, M., Kim, S., Suh, C., and Oh, S. Optimal sample complexity of m-wise data for top-k ranking. In *Advances in Neural Information Processing Systems*, pp. 1685–1695, 2017.

Kalyanakrishnan, S., Tewari, A., Auer, P., and Stone, P. Pac subset selection in stochastic multi-armed bandits. In *ICML*, volume 12, pp. 655–662, 2012.

Karnin, Z., Koren, T., and Somekh, O. Almost optimal exploration in multi-armed bandits. In *International Conference on Machine Learning*, pp. 1238–1246, 2013.

Kaufmann, E., Cappé, O., and Garivier, A. On the complexity of best-arm identification in multi-armed bandit models. *The Journal of Machine Learning Research*, 17 (1):1–42, 2016.

Khetan, A. and Oh, S. Data-driven rank breaking for efficient rank aggregation. *Journal of Machine Learning Research*, 17(193):1–54, 2016.

Mohajer, S., Suh, C., and Elmahdy, A. Active learning for top-k rank aggregation from noisy comparisons. In

International Conference on Machine Learning, pp. 2488–2497, 2017.

Popescu, P. G., Dragomir, S., Slusanschi, E. I., and Stanasila, O. N. Bounds for Kullback-Leibler divergence. *Electronic Journal of Differential Equations*, 2016, 2016.

Radlinski, F., Kurup, M., and Joachims, T. How does clickthrough data reflect retrieval quality? In *Proceedings of the 17th ACM conference on Information and knowledge management*, pp. 43–52. ACM, 2008.

Ren, W., Liu, J., and Shroff, N. B. Pac ranking from pairwise and listwise queries: Lower bounds and upper bounds. *arXiv preprint arXiv:1806.02970*, 2018.

Saha, A. and Gopalan, A. Battle of bandits. In *Uncertainty in Artificial Intelligence*, 2018a.

Saha, A. and Gopalan, A. Active ranking with subset-wise preferences. *arXiv preprint arXiv:1810.10321*, 2018b.

Saha, A. and Gopalan, A. PAC Battling Bandits in the Plackett-Luce Model. In *Algorithmic Learning Theory*, pp. 700–737, 2019.

Soufiani, H. A., Parkes, D. C., and Xia, L. Computing parametric ranking models via rank-breaking. In *ICML*, pp. 360–368, 2014.

Sui, Y., Zhuang, V., Burdick, J. W., and Yue, Y. Multi-dueling bandits with dependent arms. *arXiv preprint arXiv:1705.00253*, 2017.

Szörényi, B., Busa-Fekete, R., Paul, A., and Hüllermeier, E. Online rank elicitation for plackett-luce: A dueling bandits approach. In *Advances in Neural Information Processing Systems*, pp. 604–612, 2015.

Urvoy, T., Clerot, F., Féraud, R., and Naamane, S. Generic exploration and k-armed voting bandits. In *International Conference on Machine Learning*, pp. 91–99, 2013.

Yue, Y. and Joachims, T. Beat the mean bandit. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 241–248, 2011.

Yue, Y., Broder, J., Kleinberg, R., and Joachims, T. The k-armed dueling bandits problem. *Journal of Computer and System Sciences*, 78(5):1538–1556, 2012.

Supplementary: From PAC to Instance Optimal Sample Complexity in the Plackett-Luce Model

A. Appendix for Sec. 3

A.1. Pseudo-code for *PAC-Wrapper*

Algorithm 1 *PAC-Wrapper* **(for** ***Probably-Correct-Best-Item*** **problem with Winner feedback)**

input: Set of items: $[n]$, Subset size: $n \geq k > 1$, Confidence term $\delta > 0$
init: $\mathcal{A}_0 \leftarrow [n]$, $s \leftarrow 1$
while $|\mathcal{A}_{s-1}| \geq k$ **do**
    Set $\epsilon_s = \frac{1}{2^{s+2}}$, $\delta_s = \frac{\delta}{120 s^3}$, $\mathcal{R}_s \leftarrow \emptyset$
    $b_s \leftarrow (\epsilon, \delta)$-*PAC Best-Item*$(\mathcal{A}_{s-1}, k, 1, \epsilon_s, \delta_s)$
    $\mathcal{B}_1, \ldots \mathcal{B}_{B_s} \leftarrow$ *Partition*$(\mathcal{A}_{s-1} \setminus \{b_s\}, k-1)$
    if $|\mathcal{B}_{B_s}| < k-1$, **then** $\mathcal{R}_s \leftarrow \mathcal{B}_{B_s}$ and $B_s = B_s - 1$
    for $b = 1, 2 \ldots B_s$ **do**
        $\hat{\Theta}_S \leftarrow$ *Score-Estimate*$(b_s, \mathcal{B}_b, \delta_s)$. Set $\hat{\Theta}_S \leftarrow \max(2\hat{\Theta}_S + 1, 2)$.
        Set $\mathcal{B}_b \leftarrow \mathcal{B}_b \cup \{b_s\}$
        Play $\mathcal{B}_b$ for $t_s := \frac{2\hat{\Theta}_S}{\epsilon_s^2} \ln \frac{k}{\delta_s}$ rounds
        Receive the winner feedback: $\sigma_1, \sigma_2, \ldots \sigma_{t_s} \in \mathbf{\Sigma}^1_{\mathcal{B}_b}$ after each respective t_s rounds.
        Update pairwise empirical win-count w_{ij} using *Rank-Breaking* on $\sigma_1 \ldots \sigma_{t_s}$, $\forall i, j \in \mathcal{B}_b$
        $\hat{p}_{ij} := \frac{w_{ij}}{w_{ij} + w_{ji}}$ for all $i, j \in \mathcal{B}_b$
        If $\exists i \in \mathcal{B}_b$ s.t. $\hat{p}_{ib_s} > \frac{1}{2} - \epsilon_s$, **then** $\mathcal{A}_s \leftarrow \mathcal{A}_s \cup \{i\}$
    end for
    $\mathcal{A}_s \leftarrow \mathcal{A}_s \cup \mathcal{R}_s$, $s \leftarrow s+1$
end while
$\mathcal{A} \leftarrow \mathcal{A}_{s-1}$,
$\mathcal{B} \leftarrow \mathcal{A}_{s-1} \cup \{(k - |\mathcal{A}_{s-1}|)$ elements from $[n] \setminus \mathcal{A}_{s-1}\}$
Pairwise empirical win-count $w_{ij} \leftarrow 0$, $\forall i, j \in \mathcal{A}$
while $|\mathcal{A}| > 1$ **do**
    Set $\epsilon_s = \frac{1}{2^{s+2}}$, and $\delta_s = \frac{\delta}{80 s^3}$
    $b_s \leftarrow (\epsilon, \delta)$-*PAC Best-Item*$(\mathcal{B}, k, m, \epsilon_s, \delta_s)$
    $\hat{\Theta}_S \leftarrow$ *Score-Estimate*$(b_s, \mathcal{A} \setminus \{b_s\}, \delta_s)$. Set $\hat{\Theta}_S \leftarrow \max(2\hat{\Theta}_S + 1, 2)$.
    Play $\mathcal{B}$ for $t_s := \frac{2\hat{\Theta}_S}{m\epsilon_s^2} \ln \frac{k}{\delta_s}$ rounds, and receive the corresponding winner feedback: $\sigma_1, \sigma_2, \ldots \sigma_{t_s} \in \mathbf{\Sigma}^m_{\mathcal{B}}$ per round.
    Update pairwise empirical win-count w_{ij} using *Rank-Breaking* on $\sigma_1 \ldots \sigma_{t_s}$, $\forall i, j \in \mathcal{A}$
    Update $\hat{p}_{ij} := \frac{w_{ij}}{w_{ij} + w_{ji}}$ for all $i, j \in \mathcal{A}$
    If $\exists i \in \mathcal{A}$ with $\hat{p}_{ib_s} < \frac{1}{2} - \epsilon_s$, **then** $\mathcal{A} \leftarrow \mathcal{A} \setminus \{i\}$
    $s \leftarrow s+1$
end while
output: The item remaining in $\mathcal{A}_s$

A.2. Subroutines used in *PAC-Wrapper* (Alg. 1)

Partition **subroutine:** Partition a given set $\mathcal{A}$ into $B = \left\lceil \frac{|\mathcal{A}|}{k} \right\rceil$ equally sized batches $\mathcal{B}_1, \mathcal{B}_2, \ldots \mathcal{B}_B$, each of size at most k.

Algorithm 2 ***Partition*** subroutine

Input: Set of items: $\mathcal{A} \subseteq [n]$, Batch size: $k \in [n]$
$B \leftarrow \left\lceil \frac{|\mathcal{A}|}{k} \right\rceil$
Divide $\mathcal{A}$ into B subsets $\mathcal{B}_1, \mathcal{B}_2, \ldots \mathcal{B}_B$ such that $\mathcal{B}_i \cap \mathcal{B}_j = \emptyset$, $\cup_{i=1}^{B} \mathcal{B}_i = \mathcal{A}$ and $|\mathcal{B}_i| = k$, $\forall i \in [B-1]$
Output: B batches $\mathcal{B}_1, \mathcal{B}_2, \ldots \mathcal{B}_B$

***Score-Estimate* subroutine:** Our proposed algorithm relies on a black-box subroutine for efficient estimation of sum of the Plackett-Luce model score parameters (θ) of any given subset $S \subseteq [n]$, which we denote by $\Theta_S := \sum_{i \in S}\theta_i$. We achieve this with the subroutine *Score-Estimate* (Alg. 3) which requires a pivot element $b \in [n]$ to estimate the sum of the score parameters of the given set S (i.e. Θ_S): The algorithm simply plays the subset $S \cup \{b\}$ sufficiently many times and estimate Θ_S based on the relative win counts of items in Θ_S w.r.t. pivot b. Under the assumption that $b \in [n]$ is a sufficiently good item such that $\theta_b > \frac{1}{2}$, Thm. 13 shows Alg. 3 successfully estimates the relative scores of any subset S (upto multiplicative constants) with high confidence $(1-\delta_s)$.

Algorithm 3 ***Score-Estimate***(S, b, δ) subroutine

Input: Set of items: $S \subseteq [n]$, pivot b, and confidence parameter δ
repeat
  Play $S \cup \{b\}$ and observe the winner.
until b wins for $d = 10 \ln \frac{4}{\delta}$ times
Let T be the total number of plays of $S \cup \{b\}$, and let $Z = T - d$.
Return: $\frac{Z}{d}$

Theorem 13 (*Score-Estimate* high probability estimation guarantee)**.** *Let $\Theta_S := \sum_{i \in S}\theta_i$. Given $\Theta_S > \theta_b$, with probability at least $(1-\delta)$:*

i. the algorithm terminates in at most $\frac{10(\theta_b + \Theta_S)}{\theta_b} \ln \frac{2}{\delta}$ rounds and,

ii. the output returned by Score-Estimate (Alg. 3) satisfies:

$$\Big| \frac{Z}{d} - \frac{\Theta_S}{\theta_b} \Big| \le \frac{1}{2} \max\left(\frac{\Theta_S}{\theta_b}, 1 \right)$$

Proof. Let X_i denotes the time iteration when b wins for the i^{th} time, $\forall i \in [d]$. Note that this implies $X_i \sim \text{Geometric}\Big(\frac{\theta_b}{\Theta_S + \theta_b}\Big)$, $\forall i \in [d]$. Then from Lem. 7 of (Saha & Gopalan, 2018b), we have for any $\eta > 0$,

$$Pr\Big(\Big| \frac{Z}{d} - \frac{\Theta_S}{\theta_b} \Big| \ge \eta \Big) \le 2\exp\left(- \frac{2d\eta^2}{\Big(1 + \frac{\Theta_S}{\theta_b}\Big)\Big(\eta + 1 + \frac{\Theta_S}{\theta_b}\Big)} \right).$$

We first want to get the right hand side $2\exp\left(- \frac{2d\eta^2}{\big(1 + \frac{\Theta_S}{\theta_b}\big)\big(\eta + 1 + \frac{\Theta_S}{\theta_b}\big)} \right) \le \frac{\delta}{2}$, which further implies to have $d \ge \frac{\big(1 + \frac{\Theta_S}{\theta_b}\big)\big(\eta + 1 + \frac{\Theta_S}{\theta_b}\big)}{2\eta^2} \ln \frac{4}{\delta}$. Towards this we now would consider two cases:

Case 1: Suppose $\frac{\Theta_S}{\theta_b} \ge 1$: Then we can set $\eta = \frac{\Theta_S}{2\theta_b}$ and thus one must have:

$$\frac{\Big(1 + \frac{\Theta_S}{\theta_b}\Big)\Big(\eta + 1 + \frac{\Theta_S}{\theta_b}\Big)}{2\eta^2} \ln \frac{4}{\delta} = \frac{\Big(\frac{\theta_b}{\Theta_S} + 1\Big)\Big(\frac{\theta_b}{\Theta_S} + \frac{3}{2}\Big)}{2(1/4)} \ln \frac{4}{\delta} \le \frac{\Big(2\Big)\Big(\frac{5}{2}\Big)}{2(1/4)} \ln \frac{4}{\delta} = 10 \ln \frac{4}{\delta} \le d$$

Case 2: Suppose $\frac{\Theta_S}{\theta_b} < 1$: In this case we may set $\eta = \frac{1}{2}$ so then it suffices to have

$$\frac{\Big(1 + \frac{\Theta_S}{\theta_b}\Big)\Big(\eta + 1 + \frac{\Theta_S}{\theta_b}\Big)}{2\eta^2} \ln \frac{4}{\delta} \le \frac{\Big(2\Big)\Big(\frac{5}{2}\Big)}{2/4} \ln \frac{4}{\delta} = 10 \ln \frac{4}{\delta}$$

Thus combining both cases we get with probability at least $(1-\delta/2)$: $\Big| \frac{Z}{d} - \frac{\Theta_S}{\theta_b} \Big| < \frac{1}{2} \max\left(\frac{\Theta_S}{\theta_b}, 1 \right)$.

So we are only left to prove the required sample complexity of *Score-Estimate* to yield $d = 10\ln\frac{4}{\delta}$ wins of θ_b. For this, note that at any round item b wins with probability $\frac{\theta_b}{\Theta_S+\theta_b}$. So for any fixed τ rounds $E[d] = \frac{\theta_b\tau}{\Theta_S+\theta_b}$. Then applying multiplicative Chernoff bounds, we know that for any $\epsilon \in (0,1)$,

$$Pr\Big(d \le (1-\epsilon)E[d]\Big) \le \exp\Big(-\frac{E[d]\epsilon^2}{2}\Big),$$

which implies whenever $\tau \ge \frac{20(\Theta_S+\theta_b)}{\theta_b\epsilon^2}\ln\frac{4}{\delta}$, $d \ge (1-\epsilon)\frac{\tau\theta_b}{\theta_b+\Theta_S}$ with probability at least $(1-\delta/4)$. Finally noting that we need $d \ge 10\ln\frac{4}{\delta}$, this implies we can easily set $\epsilon = \frac{1}{2}$ so that

$$d \ge (1-\epsilon)\frac{20(\Theta_S+\theta_b)}{\theta_b\epsilon^2}\frac{\theta_b}{\theta_b+\Theta_S}\ln\frac{4}{\delta} \ge 10\ln\frac{4}{\delta},$$

and the claim follows. □

Lemma 14. *Let us denote $\hat{\Theta}_S := \max($Score-Estimate$(S,b,\delta),1)$. Consider the notations introduced in Thm. 13, Then with probability at least $(1-\delta)$, $\max(1,\frac{\Theta_S}{2\theta_b}) \le \frac{Z}{d} \le \max(\frac{\Theta_S}{2\theta_b},\frac{3}{2})$, and the algorithm Score-Estimate terminates in at most $40(\Theta_S+\theta_b)\ln\frac{4}{\delta}$ many iterations.*

Proof. The proof directly follows from Thm. 13. □

Corollary 15. *Let $S \subseteq [n]$, $|S| = k$, and let $\Theta_{[k]} = \max_{S\subseteq[n]||S|=k}\sum_{i\in S}\theta_i$. With the notation of Lem. 14, if b is an $\frac{1}{2}$-optimal item such that $\theta_b > \theta_1 - \epsilon$ for any $\epsilon \in \Big(0,\frac{1}{2}\Big]$, then with probability at least $(1-\delta)$, $\max(1,\Theta_S/2) \le \frac{Z}{d} \le 6\Theta_{[k]}$, and the Score-Estimate algorithm terminates in at most $80\Theta_{[k]}\ln\frac{4}{\delta}$ iterations.*

Proof. The proof directly follows from Lem. 14, with noting that by definition for any $S \subseteq [n]$, $|S| = k$, $\Theta_S \le \Theta_{[k]}$, $\theta_b > \frac{1}{2}$ and $\Theta_{[k]} \ge 1$, since we assume $\theta_1 = 1$ and of course $\theta_1 \in \Theta_{[k]}$. □

Rank-Breaking Subroutine (Soufiani et al., 2014; Khetan & Oh, 2016). This is a procedure of deriving pairwise comparisons from multiwise (subsetwise) preference information. Formally, given any set $S \subseteq [n]$, $m \le |S| < n$, if $\boldsymbol{\sigma} \in \boldsymbol{\Sigma}_S^m$ denotes a possible Top-m Ranking feedback of S, *Rank-Breaking* considers each item in S to be beaten by its preceding items in $\boldsymbol{\sigma}$ in a pairwise sense and extracts out total $\sum_{i=1}^{m}(k-i) = \frac{m(2k-m-1)}{2}$ such pairwise comparisons. For instance, given a full ranking of a set of 4 elements $S = \{a,b,c,d\}$, say $b \succ a \succ c \succ d$, Rank-Breaking generates the set of 6 pairwise comparisons: $\{(b\succ a),(b\succ c),(b\succ d),(a\succ c),(a\succ d),(c\succ d)\}$.

Algorithm 4 *Rank-Breaking* subroutine

1: **Input:** Subset $S \subseteq [n]$, such that $|S| = k$ $(n \ge k)$
2: A top-m ranking $\boldsymbol{\sigma} \in \boldsymbol{\Sigma}_S^m$, for some $m \in [k-1]$
3: Pairwise win-count w_{ij} for each item pair $i,j \in S$
4: **while** $\ell = 1,2,\ldots m$ **do**
5: Update $w_{\sigma(\ell)i} \leftarrow w_{\sigma(\ell)i} + 1$, for all $i \in S \setminus \{\sigma(1),\ldots,\sigma(\ell)\}$
6: **end while**

A.3. Pseudo-code for (ϵ,δ)-*PAC Best-Item*

Algorithm description: The algorithm (ϵ,δ)-*PAC Best-Item* first divides the set of n items into batches of size k, and plays each group sufficiently long enough, until a single item of that group stands out as the empirical winner in terms of its empirical pairwise advantage over the rest (again estimated through *Rank-Breaking*). It then just retains this empirical winner for every group and recurses on the set of surviving winners, until only a single item is left behind, which is declared as the (ϵ,δ)-PAC item. The Its important to note that the sample complexity of our algorithm (see Thm. 6) offers an improved instance dependent guarantee (compared to the $O\Big(\frac{n}{m\epsilon^2}\ln\frac{k}{\delta}\Big)$ sample complexity algorithm *Divide-and-Battle* proposed by (Saha & Gopalan, 2019)), which would turn out to be crucial for the *instance-dependent sample-complexity* analyses of our main algorithms, Alg. 1 or 7, later. (See proof of Thm. 3 and 5 respectively for details.) Though our proposed algorithm proceed along a line similar to *Divide-and-Battle* of (Saha & Gopalan, 2019), the crux of our proposed algorithm lies in sampling each subset $\mathcal{G}_g$ just sufficiently enough in an adaptive way for only $O\bigg(\frac{\hat{\Theta}_{\mathcal{G}_g}}{m\epsilon^2}\ln\frac{2n}{\delta}\bigg)$ times—thanks to our sum estimation routine *Score-Estimate* (Alg. 3)—instead of sampling them blindly for $O\bigg(\frac{k}{m\epsilon^2}\ln\frac{2n}{\delta}\bigg)$ times as proposed in *Divide-and-Battle*. To find the $(1/2,\delta)$-optimal item: $b \in [n]$ required to estimate $\hat{\Theta}_{\mathcal{G}_g}$, we can use the existing algorithms like *Divide-and-Battle*. The complete algorithm is described in Alg. 5.

Algorithm 5 (ϵ,δ)-***PAC Best-Item*** (for TR feedback)

Input:
  Set of items: $[n]$, and subset size: $k > 2$ $(n \ge k \ge m)$
  Error bias: $\epsilon > 0$, and confidence parameter: $\delta > 0$
  A $(1/2,\delta)$-optimal item: $b \in [n]$, such that $\theta_b > \frac{1}{2}$
Initialize:
  $S \leftarrow [n]$, $\epsilon_0 \leftarrow \frac{\epsilon}{8}$, and $\delta_0 \leftarrow \frac{\delta}{2}$
  Divide S into $G := \lceil \frac{n}{k} \rceil$ sets $\mathcal{G}_1, \mathcal{G}_2, \cdots \mathcal{G}_G$ such that $\cup_{j=1}^{G}\mathcal{G}_j = S$ and $\mathcal{G}_j \cap \mathcal{G}_{j'} = \emptyset,\ \forall j,j' \in [G]$, $|G_j| = k, \forall j \in [G-1]$. **If** $|\mathcal{G}_G| < k$, **then** set $\mathcal{R}_1 \leftarrow \mathcal{G}_G$ and $G = G-1$, **Else** $\mathcal{R}_1 \leftarrow \emptyset$.
while $\ell = 1, 2, \ldots$ **do**
  Set $\delta_\ell \leftarrow \frac{\delta_{\ell-1}}{2}$, $\epsilon_\ell \leftarrow \frac{3}{4}\epsilon_{\ell-1}$
  for $g = 1, 2, \cdots G$ **do**
    $\hat{\Theta}_{\mathcal{G}_g} \leftarrow$ *Score-Estimate*$\big(\mathcal{G}_g, b, \delta_\ell\big)$
    Initialize pairwise (empirical) win-count $w_{ij} \leftarrow 0$, for each item pair $i,j \in \mathcal{G}_g$
    for $\tau = 1, 2, \ldots t := \left\lceil \frac{16\hat{\Theta}_{\mathcal{G}_g}}{m\epsilon_l^2}\ln\frac{2k}{\delta_\ell} \right\rceil$ **do**
      Play the set $\mathcal{G}_g$ (one round of battle)
      Receive feedback: The top-m ranking $\boldsymbol{\sigma} \in \boldsymbol{\Sigma}_{\mathcal{G}_g}^m$
      Update win-count w_{ij} of each item pair $i,j \in \mathcal{G}_g$ using *Rank-Breaking*$(\mathcal{G}_g, \boldsymbol{\sigma})$
    end for
    Estimate pairwise win probabilities: $\forall i,j \in \mathcal{G}_g$ $\hat{p}_{i,j} = \frac{w_{ij}}{w_{ij}+w_{ji}}$ if $w_{ij}+w_{ji} > 0$, $\hat{p}_{i,j} = \frac{1}{2}$ otherwise
    If $\exists i \in \mathcal{G}_g$ such that $\hat{p}_{ij} + \frac{\epsilon_\ell}{2} \ge \frac{1}{2}, \forall j \in \mathcal{G}_g$, then set $c_g \leftarrow i$, else select $c_g \leftarrow$ uniformly at random from $\mathcal{G}_g$. Set $S \leftarrow S \setminus (\mathcal{G}_g \setminus \{c_g\})$
  end for
  $S \leftarrow S \cup \mathcal{R}_\ell$
  if $(|S| == 1)$ **then**
    Break (out of the **while** loop)
  else if $|S| \le k$ **then**
    $S' \leftarrow$ Randomly sample $k - |S|$ items from $[n] \setminus S$, and set $S \leftarrow S \cup S'$, $\epsilon_\ell \leftarrow \frac{2\epsilon}{3}$, $\delta_\ell \leftarrow \delta$
  end if
  Divide S into $G := \lceil \frac{|S|}{k} \rceil$ sets $\mathcal{G}_1, \cdots \mathcal{G}_G$ such that $\cup_{j=1}^{G}\mathcal{G}_j = S$, $\mathcal{G}_j \cap \mathcal{G}_{j'} = \emptyset,\ \forall j,j' \in [G]$, $|G_j| = k, \forall j \in [G-1]$. **If** $|\mathcal{G}_G| < k$, **then** set $\mathcal{R}_{\ell+1} \leftarrow \mathcal{G}_G$ and $G = G-1$, **Else** $\mathcal{R}_1 \leftarrow \emptyset$.
end while
Output: r_*, the single item remaining in S

Theorem 6 ((ϵ,δ)-*PAC Best-Item*: Correctness and Sample Complexity with Top-m Ranking feedback)**.** *For any* $\epsilon \in \big(0, \frac{1}{8}\big]$ *and* $\delta \in (0,1)$, *with probability at least* $(1-\delta)$, (ϵ,δ)-*PAC Best-Item (Algorithm 5) returns an item* $b_s \in [n]$ *satisfying* $p_{b_s 1} > \frac{1}{2} - \epsilon$ *with sample complexity* $O\left(\frac{n\Theta_{[k]}}{k}\max\left(1, \frac{1}{m\epsilon^2}\right)\log\frac{k}{\delta}\right)$, *where* $\Theta_{[k]} := \max_{S \subseteq [n], |S|=k}\sum_{i \in S}\theta_i$.

Proof. For notational convenience we will use $\tilde{p}_{ij} = p_{ij} - \frac{1}{2}$, $\forall i,j \in [n]$.

We start by recalling a lemma from (Saha & Gopalan, 2019) which will be used crucially in the analysis:

Lemma 16. *(Saha & Gopalan, 2019) For any three items* $a,b,c \in [n]$ *such that* $\theta_a > \theta_b > \theta_c$, *if* $\tilde{p}_{ba} > -\epsilon_1$ *and* $\tilde{p}_{cb} > -\epsilon_2$, *where* $\epsilon_1, \epsilon_2 > 0$ *and* $(\epsilon_1 + \epsilon_2) < \frac{1}{2}$, *then* $\tilde{p}_{ca} > -(\epsilon_1 + \epsilon_2)$.

We first bound the sample complexity of Algorithm 5. For clarity of notation, we denote the set S at the beginning of iteration ℓ (i.e., at line 9) by S_ℓ. Note that at an iteration ℓ, any set $\mathcal{G}_g$ is played $t = \lceil \frac{16\hat{\Theta}_{\mathcal{G}_g}}{m\epsilon_l^2} \ln \frac{2k}{\delta_\ell} \rceil \le \lceil \frac{96\Theta_{[k]}}{m\epsilon_l^2} \ln \frac{2k}{\delta_\ell} \rceil$ times, where the inequality follows from Corollary 15. Also, since the algorithm discards exactly $k-1$ items from each set $\mathcal{G}_g$, the maximum number of iterations possible is $\lceil \ln_k n \rceil$. Now, at iteration ℓ, since $G = \left\lfloor \frac{|S_\ell|}{k} \right\rfloor < \frac{|S_\ell|}{k}$, the total sample complexity for iteration ℓ is at most

$$\frac{|S_\ell|}{k} t \le \frac{n}{k^\ell} \left\lceil \frac{96\Theta_{[k]}}{m\epsilon_\ell^2} \ln \frac{2k}{\delta_\ell} \right\rceil$$

using the fact that $|S_\ell| \le \frac{n}{k^{\ell-1}}$ for all $\ell \in [\lfloor \ln_k n \rfloor]$. For all iterations $\ell \in [\lfloor \ln_k n \rfloor]$ except the final one, we have $\epsilon_\ell = \frac{\epsilon}{8}\left(\frac{3}{4}\right)^{\ell-1}$ and $\delta_\ell = \frac{\delta}{2^{\ell+1}}$. Moreover, for the last iteration $\ell = \lceil \ln_k n \rceil$, the sample complexity is at most $\left\lceil \frac{96\Theta_{[k]}}{m(\epsilon/2)^2} \ln \frac{4k}{\delta} \right\rceil$ since, in this case, $\epsilon_\ell = \frac{\epsilon}{2}$, and $\delta_\ell = \frac{\delta}{2}$, and $|S| = k$.

Let us ignore, for the moment, the additional sample complexity due to the score estimation subroutine, *Score-Estimate*, in the operation of Algorithm 5. Then, the argument above implies that the sample complexity of the algorithm is at most

$$\begin{aligned}
(A) := \sum_{\ell=1}^{\lceil \ln_k n \rceil} \frac{|S_\ell|}{k} t &\le \sum_{\ell=1}^{\infty} \frac{n}{k^\ell} \left\lceil \frac{96\Theta_{[k]}}{m\left(\frac{\epsilon}{8}\left(\frac{3}{4}\right)^{\ell-1}\right)^2} \ln \frac{k2^{\ell+1}}{\delta} \right\rceil + \left\lceil \frac{96\Theta_{[k]}}{m(\epsilon/2)^2} \ln \frac{4k}{\delta} \right\rceil \\
&\le \sum_{\ell=1}^{\infty} \frac{n}{k^\ell} \left(\frac{96\Theta_{[k]}}{m\left(\frac{\epsilon}{8}\left(\frac{3}{4}\right)^{\ell-1}\right)^2} \ln \frac{k2^{\ell+1}}{\delta} + 1 \right) + \left(\frac{96\Theta_{[k]}}{m(\epsilon/2)^2} \ln \frac{4k}{\delta} + 1 \right) \\
&\le \frac{4096 n\Theta_{[k]}}{mk\epsilon^2} \sum_{\ell=1}^{\infty} \frac{16^{\ell-1}}{(9k)^{\ell-1}} \Big(\ln \frac{k}{\delta} + (\ell+1) \Big) + \frac{n}{k-1} + \frac{384\Theta_{[k]}}{m\epsilon^2} \ln \frac{4k}{\delta} + 1 \\
&\le \frac{4096 n\Theta_{[k]}}{mk\epsilon^2} \ln \frac{k}{\delta} \sum_{\ell=1}^{\infty} \frac{4^{\ell-1}}{(9k)^{\ell-1}} \Big(3\ell \Big) + \frac{384\Theta_{[k]}}{m\epsilon^2} \ln \frac{4k}{\delta} + \Big(1 + \frac{n}{k-1} \Big) \\
&= O\Big(\frac{n\Theta_{[k]}}{mk\epsilon^2} \ln \frac{k}{\delta} + \frac{n}{k} \Big) \quad \text{[for any } k > 1\text{]} \\
&= O\Big(\frac{n\Theta_{[k]}}{mk\epsilon^2} \ln \frac{k}{\delta} + \frac{n\Theta_{[k]}}{k} \ln \frac{k}{\delta} \Big) \quad \text{[since } \Theta_{[k]} \ge 1, \ln \frac{k}{\delta} \ge 1\text{]}.
\end{aligned}$$

Turning to the extra effort expended by the score estimation subroutine *Score-Estimate*$(\mathcal{G}_g, b, \delta_\ell)$, at each phase ℓ, the sample complexity of *Score-Estimate* is known by Cor. 15 to be at most $80\Theta_{[k]} \ln \frac{4}{\delta_\ell} = O\Big(\Theta_{[k]} \ln \frac{1}{\delta_\ell}\Big)$ for any subgroup $\mathcal{G}_g$. And since there are at most $G = \left\lfloor \frac{|S_\ell|}{k} \right\rfloor < \frac{|S_\ell|}{k}$ subgroups at any phase ℓ, this implies that the total sample complexity incurred at any phase owing to *Score-Estimate* is at most $\frac{80|S_\ell|\Theta_{[k]}}{k} \ln \frac{4}{\delta_\ell} \le \frac{80n\Theta_{[k]}}{k^{\ell+1}} \ln \frac{4}{\delta_\ell}$. Following the same calculations as before, the total sample complexity incurred by the *Score-Estimate* subroutine within the algorithm, over all iterations, is at most

$$(B) := \sum_{\ell=1}^{\lceil \ln_k n \rceil} \frac{80n\Theta_{[k]}}{k^{\ell+1}} \ln \frac{4}{\delta_\ell} \le \sum_{\ell=1}^{\infty} \frac{n\Theta_{[k]}}{k} \frac{80}{k^\ell} \ln \frac{82^\ell}{\delta} = O\Big(\frac{n\Theta_{[k]}}{k} \ln \frac{1}{\delta} \Big) = O\Big(\frac{n\Theta_{[k]}}{k} \ln \frac{k}{\delta} \Big).$$

Observe now that the term (B) is dominated by (A) in general unless $\frac{1}{m\epsilon^2} = O(1)$, or in other words m is so large that $m = \Omega\left(\frac{1}{\epsilon^2}\right)$. Thus taking care of the above tradeoff between term (A) and (B), the final sample complexity can be expressed as $O(\frac{n\Theta_{[k]}}{k} \max\left(1, \frac{1}{m\epsilon^2}\right) \log \frac{k}{\delta})$. This proves the sample complexity bound for Algorithm 5.

We now proceed to prove the (ϵ, δ)-PAC correctness of Algorithm 5. We start by making the following observation.

Lemma 17. *Consider any particular set $\mathcal{G}_g$ at any iteration $\ell \in \lfloor \frac{n}{k} \rfloor$, and let $q_i := \sum_{\tau=1}^{t} \mathbf{1}(i \in \mathcal{G}_g^m)$ be the number of times any item $i \in \mathcal{G}_g$ appears in the top-m rankings when $\mathcal{G}_g$ is played for t rounds. If $i_g := \arg\max_{i \in \mathcal{G}_g} \theta_i$ and $\theta_{i_g} > \frac{1}{2}$, then for any $\eta \in \big(\frac{3}{32\sqrt{2}}, 1\big]$, with probability at least $\Big(1 - \frac{\delta_\ell}{2k}\Big)$, $q_{i_g} > (1-\eta)\frac{mt}{k}$.*

Proof. Define $i^\tau := \mathbf{1}(i \in \mathcal{G}_g^m)$ as the indicator of the event that the i^{th} element appears in the top-m ranking at iteration $\tau \in [t]$. Using the definition of the top-m ranking feedback model, we get $\mathbf{E}[i_g^\tau] = Pr(\{i_g \in \mathcal{G}_g^m\}) = Pr\big(\exists j \in [m] \mid \sigma(j) = i_g\big) = \sum_{j=1}^{m} Pr\big(\sigma(j) = i_g\big) > \sum_{j=0}^{m-1} \frac{\theta_{i_g}}{\hat{\Theta}_{\mathcal{G}_g}} \geq \frac{m\theta_{i_g}}{\hat{\Theta}_{\mathcal{G}_g}}$, as $Pr(\{i_g|S\}) = \frac{\theta_{i_g}}{\sum_{j \in S} \theta_j} \geq \frac{\theta_{i_g}}{\hat{\Theta}_{\mathcal{G}_g}}$ for any $S \subseteq [\mathcal{G}_g]$, $i \in \mathcal{G}_g$, as $i_g := \arg\max_{i \in \mathcal{G}_g} \theta_i$ is the best item of set $\mathcal{G}_g$. Hence $\mathbf{E}[q_{i_g}] = \sum_{\tau=1}^{t} \mathbf{E}[i_g^\tau] \geq \frac{mt\theta_{i_g}}{\Theta_{\mathcal{G}_g}} > \frac{mt}{2\Theta_{\mathcal{G}_g}}$.

Applying the Chernoff-Hoeffding concentration inequality for w_{i_g}, we get that for any $\eta \in (\frac{3}{32}, 1]$,

$$
\begin{aligned}
Pr\Big(q_{i_g} \leq (1-\eta)\mathbf{E}[q_{i_g}]\Big) &\leq \exp\left(-\frac{\mathbf{E}[q_{i_g}]\eta^2}{2}\right) \leq \exp\left(-\frac{mt\eta^2}{4\Theta_{\mathcal{G}_g}}\right) \\
&= \exp\left(-\frac{2\eta^2}{\epsilon_\ell^2}\ln\left(\frac{2k}{\delta_\ell}\right)\right) = \exp\left(-\frac{(\sqrt{2}\eta)^2}{\epsilon_\ell^2}\ln\left(\frac{2k}{\delta_\ell}\right)\right) \\
&\leq \exp\left(-\ln\left(\frac{2k}{\delta_\ell}\right)\right) \leq \frac{\delta_\ell}{2k},
\end{aligned}
$$

where the second last inequality holds as $\eta \geq \frac{3}{32\sqrt{2}}$ and $\epsilon_\ell \leq \frac{3}{32}$, for any iteration $\ell \in \lceil \ln n \rceil$; in other words for any $\eta \geq \frac{3}{32\sqrt{2}}$, we have $\frac{\sqrt{2}\eta}{\epsilon_\ell} \geq 1$ which leads to the second last inequality. Thus, we get that with probability at least $\Big(1 - \frac{\delta_\ell}{2k}\Big)$, it holds that $q_{i_g} > (1-\eta)\mathbf{E}[q_{i_g}] \geq (1-\eta)\frac{tm}{2\Theta_{\mathcal{G}_g}}$. $\square$

In particular, fixing $\eta = \frac{1}{2}$ in Lemma 17, we get that with probability at least $\big(1 - \frac{\delta_\ell}{2}\big)$, $q_{i_g} > (1 - \frac{1}{2})\mathbf{E}[w_{i_g}] > \frac{mt}{4\Theta_{\mathcal{G}_g}}$. Note that for any round $\tau \in [t]$, whenever an item $i \in \mathcal{G}_g$ appears in the top-m set $\mathcal{G}_{gm}^\tau$, then the rank breaking update ensures that every element in the top-m set gets compared with rest of the $k-1$ elements of $\mathcal{G}_g$. Based on this observation, we now prove that for any set $\mathcal{G}_g$, its best item i_g is retained as the winner c_g with probability at least $\big(1 - \frac{\delta_\ell}{2}\big)$. More formally, we make the following observation.

Lemma 18. *Consider any particular set $\mathcal{G}_g$ at any iteration $\ell \in \lfloor \frac{n}{k} \rfloor$. If $i_g = \arg\max_{i \in \mathcal{G}_g} \theta_i$ and $\theta_{i_g} > \frac{1}{2}$, then the following events occur with probability at least $\Big(1 - \delta_\ell\Big)$: (1) $\hat{p}_{i_g j} + \frac{\epsilon_\ell}{2} \geq \frac{1}{2}$ for all ϵ_ℓ-optimal items in $\mathcal{G}_g$, i.e., $\forall j \in \mathcal{G}_g$ such that $p_{i_g j} \in \big(\frac{1}{2}, \frac{1}{2} + \epsilon_\ell\big]$, and (2) $\hat{p}_{i_g j} - \frac{\epsilon_\ell}{2} \geq \frac{1}{2}$ for all non ϵ_ℓ-optimal items in $\mathcal{G}_g$, i.e., $j \in \mathcal{G}_g$ such that $p_{i_g j} > \frac{1}{2} + \epsilon_\ell$.*

Proof. With top-m ranking feedback, the crucial observation lies in the fact that at any round $\tau \in [t]$, whenever an item $i \in \mathcal{G}_g$ appears in the top-m set $\mathcal{G}_g^m$, then the rank breaking update ensures that every element in the top-m set gets compared with each of the rest of the $k-1$ elements of $\mathcal{G}_g$: it gets defeated by every element preceding it in $\sigma \in \Sigma_{\mathcal{G}_g^m}$, and defeats all other items in the top-m set $\mathcal{G}_g^m$. Therefore, defining $n_{ij} = w_{ij} + w_{ji}$ to be the number of times item i and j are compared after rank-breaking, $i, j \in \mathcal{G}_g$. Clearly $n_{ij} = n_{ji}$, and $0 \leq n_{ij} \leq tk$. Moreover, from Lemma 17 with $\eta = \frac{1}{2}$, we have that $n_{i_g j} \geq \frac{mt}{4\Theta_{\mathcal{G}_g}}$. Given the above arguments in place let us analyze the probability of a 'bad event', i.e.:

Case 1. j is ϵ_ℓ-optimal with respect to i_g, i.e. $p_{i_g j} \in \big(\frac{1}{2}, \frac{1}{2} + \epsilon_\ell\big]$. Then we have

$$
\begin{aligned}
Pr\left(\left\{\hat{p}_{i_g j} + \frac{\epsilon_\ell}{2} < \frac{1}{2}\right\} \cap \left\{n_{i_g j} \geq \frac{mt}{4\Theta_{\mathcal{G}_g}}\right\}\right) &= Pr\left(\left\{\hat{p}_{i_g j} < \frac{1}{2} - \frac{\epsilon_\ell}{2}\right\} \cap \left\{n_{i_g j} \geq \frac{mt}{4\Theta_{\mathcal{G}_g}}\right\}\right) \\
&\leq Pr\left(\left\{\hat{p}_{i_g j} - p_{i_g j} < -\frac{\epsilon_\ell}{2}\right\} \cap \left\{n_{i_g j} \geq \frac{mt}{4\Theta_{\mathcal{G}_g}}\right\}\right)
\end{aligned}
$$

$$\leq \exp\Big(-2\frac{mt}{4\Theta_{\mathcal{G}_g}}(\epsilon_\ell/2)^2\Big)\Big) = \frac{\delta_\ell}{2k},$$

where the first inequality follows as $p_{i_g j} > \frac{1}{2}$, and the second inequality follows from Lemma 22 with $\eta = \frac{\epsilon_\ell}{2}$ and $v = \frac{mt}{4\Theta_{\mathcal{G}_g}}$.

Case 2. j is non ϵ_ℓ-optimal with respect to i_g, i.e. $p_{i_g j} > \frac{1}{2} + \epsilon_\ell$. Similar to before, we have

$$\begin{aligned}
Pr\bigg(\Big\{\hat{p}_{i_g j} - \frac{\epsilon_\ell}{2} < \frac{1}{2}\Big\} \cap \Big\{n_{i_g j} \geq \frac{mt}{4\Theta_{\mathcal{G}_g}}\Big\}\bigg) &= Pr\bigg(\Big\{\hat{p}_{i_g j} < \frac{1}{2} + \frac{\epsilon_\ell}{2}\Big\} \cap \Big\{n_{i_g j} \geq \frac{mt}{4\Theta_{\mathcal{G}_g}}\Big\}\bigg) \\
&\leq Pr\bigg(\Big\{\hat{p}_{i_g j} - p_{i_g j} < -\frac{\epsilon_\ell}{2}\Big\} \cap \Big\{n_{i_g j} \geq \frac{mt}{4\Theta_{\mathcal{G}_g}}\Big\}\bigg) \\
&\leq \exp\Big(-2\frac{mt}{4\Theta_{\mathcal{G}_g}}(\epsilon_\ell/2)^2\Big)\Big) = \frac{\delta_\ell}{2k},
\end{aligned}$$

where the third last inequality follows since in this case $p_{i_g j} > \frac{1}{2} + \epsilon_\ell$, and the last inequality follows from Lemma 22 with $\eta = \frac{\epsilon_\ell}{2}$ and $v = \frac{mt}{2k}$.

Let us define the event $\mathcal{E} := \Big\{\exists j \in \mathcal{G}_g$ such that $\hat{p}_{i_g j} + \frac{\epsilon_\ell}{2} < \frac{1}{2}, p_{i_g j} \in \big(\frac{1}{2}, \frac{1}{2} + \epsilon_\ell\big]$ or $\hat{p}_{i_g j} - \frac{\epsilon_\ell}{2} < \frac{1}{2}, p_{i_g j} > \frac{1}{2} + \epsilon_\ell\Big\}$. Then by combining Case 1 and 2, we get

$$\begin{aligned}
Pr\Big(\mathcal{E}\Big) &= Pr\bigg(\mathcal{E} \cap \Big\{n_{i_g j} \geq \frac{mt}{4\Theta_{\mathcal{G}_g}}\Big\}\bigg) + Pr\bigg(\mathcal{E} \cap \Big\{n_{i_g j} < \frac{mt}{4\Theta_{\mathcal{G}_g}}\Big\}\bigg) \\
&\leq \sum_{j \in \mathcal{G}_g \text{ s.t. } p_{i_g j} \in \left(\frac{1}{2}, \frac{1}{2} + \epsilon_\ell\right]} Pr\bigg(\Big\{\hat{p}_{i_g j} + \frac{\epsilon_\ell}{2} < \frac{1}{2}\Big\} \cap \Big\{n_{i_g j} \geq \frac{mt}{4\Theta_{\mathcal{G}_g}}\Big\}\bigg) \\
&+ \sum_{j \in \mathcal{G}_g \text{ s.t. } p_{i_g j} > \frac{1}{2} + \epsilon_\ell} Pr\bigg(\Big\{\hat{p}_{i_g j} - \frac{\epsilon_\ell}{2} < \frac{1}{2}\Big\} \cap \Big\{n_{i_g j} \geq \frac{mt}{4\Theta_{\mathcal{G}_g}}\Big\}\bigg) + Pr\bigg(\Big\{n_{i_g j} < \frac{mt}{4\Theta_{\mathcal{G}_g}}\Big\}\bigg) \\
&\leq \frac{(k-1)\delta_\ell}{2k} + \frac{\delta_\ell}{2k} \leq \delta_\ell
\end{aligned}$$

where the last inequality follows from the above two case analyses and Lemma 17. $\square$

Given Lemma 18 in place, let us now analyze with what probability the algorithm can select a non ϵ_ℓ-optimal item $j \in \mathcal{G}_g$ as c_g at any iteration $\ell \in \lceil \frac{n}{k} \rceil$. For any set $\mathcal{G}_g$ (or set S for the last iteration $\ell = \lceil \frac{n}{k} \rceil$), we define the set of non-ϵ_ℓ-optimal elements $\mathcal{O}_g = \{j \in \mathcal{G}_g \mid p_{i_g j} > \frac{1}{2} + \epsilon_\ell\}$, and recall the event $\mathcal{E} := \Big\{\exists j \in \mathcal{G}_g$ such that $\hat{p}_{i_g j} + \frac{\epsilon_\ell}{2} < \frac{1}{2}, p_{i_g j} \in \big(\frac{1}{2}, \frac{1}{2} + \epsilon_\ell\big]$ or $\hat{p}_{i_g j} - \frac{\epsilon_\ell}{2} < \frac{1}{2}, p_{i_g j} > \frac{1}{2} + \epsilon_\ell\Big\}$. We then have

$$\begin{aligned}
Pr(c_g \in \mathcal{O}_g) &\leq Pr\bigg(\Big\{\exists j \in \mathcal{G}_g, \hat{p}_{i_g j} + \frac{\epsilon_\ell}{2} < \frac{1}{2}\Big\} \cup \Big\{\exists j \in \mathcal{O}_g, \hat{p}_{j i_g} + \frac{\epsilon_\ell}{2} \geq \frac{1}{2}\Big\}\bigg) \\
&\leq Pr\bigg(\mathcal{E} \cup \Big\{\exists j \in \mathcal{O}_g, \hat{p}_{j i_g} + \frac{\epsilon_\ell}{2} \geq \frac{1}{2}\Big\}\bigg) \\
&= Pr\Big(\mathcal{E}\Big) + Pr\bigg(\Big\{\exists j \in \mathcal{O}_g, \hat{p}_{j i_g} + \frac{\epsilon_\ell}{2} \geq \frac{1}{2}\Big\} \cap \mathcal{E}^c\bigg)
\end{aligned}$$

$$= Pr\Big(\mathcal{E}\Big) + Pr\Bigg(\bigg\{\exists j \in \mathcal{O}_g, \hat{p}_{ji_g} + \frac{\epsilon_\ell}{2} \geq \frac{1}{2}\bigg\} \cap \mathcal{E}^c\Bigg) \leq \delta_\ell + 0 = \delta_\ell, \tag{1}$$

where the last inequality follows from Lemma 18 and the fact that $\hat{p}_{i_g j} - \frac{\epsilon_\ell}{2} \geq \frac{1}{2} \implies \hat{p}_{ji_g} + \frac{\epsilon_\ell}{2} < \frac{1}{2}$. The proof now follows by combining all the above parts together.

At each iteration ℓ, let us define $g_\ell \in [G]$ to be the index of the set that contains the *best item* of the currently surviving set S, i.e., the index g_ℓ such that $\arg\max_{i \in S} \theta_i \in \mathcal{G}_{g_\ell}$. Then from (1), with probability at least $(1-\delta_\ell)$, $\tilde{p}_{c_{g_\ell} i_{g_\ell}} > -\epsilon_\ell$. Now for each iteration ℓ, recursively applying (1) and Lemma 16 to $\mathcal{G}_{g_\ell}$, we get that $\tilde{p}_{r_* 1} > -\Big(\frac{\epsilon}{8} + \frac{\epsilon}{8}\Big(\frac{3}{4}\Big) + \cdots + \frac{\epsilon}{8}\big(\frac{3}{4}\big)^{\lfloor \frac{n}{k} \rfloor}\Big) + \frac{\epsilon}{2} \geq -\frac{\epsilon}{8}\Big(\sum_{i=0}^{\infty}\big(\frac{3}{4}\big)^i\Big) + \frac{\epsilon}{2} = \epsilon$. (Note that for this analysis to go through, it is in fact sufficient to consider only the set of iterations $\{\ell \geq \ell_0 \mid \ell_0 = \min\{l \mid 1 \notin \mathcal{R}_l,\, l \geq 1\}\}$, because prior to considering item 1, it does not matter even if the algorithm makes a mistake in any of the iterations $\ell < \ell_0$). Thus assuming that the algorithm does not fail in any of the iterations ℓ, we have that $p_{r_* 1} > \frac{1}{2} - \epsilon$.

Finally, since at each iteration ℓ, the algorithm fails with probability at most δ_ℓ, the total failure probability of the algorithm is at most $\Big(\frac{\delta}{4} + \frac{\delta}{8} + \cdots + \frac{\delta}{2^{\lceil \frac{n}{k} \rceil}}\Big) + \frac{\delta}{2} \leq \delta$. This concludes the correctness of the algorithm showing that it indeed returns an ϵ-best element r_* such that $p_{r_* 1} \geq \frac{1}{2} - \epsilon$ with probability at least $1-\delta$. □

A.4. Proof of Thm. 3

Theorem 3 (*PAC-Wrapper*$(0,\delta)$-PAC sample complexity bound with Winner feedback)**.** *With probability at least* $(1-\delta)$*,* $\mathcal{A}$ *as PAC-Wrapper (Algorithm 1) returns the Best-Item with sample complexity* $N_{\mathcal{A}}(0,\delta) = O\bigg(\frac{\Theta_{[k]}}{k}\sum_{i=2}^{n}\max\Big(1, \frac{1}{\Delta_i^2}\Big)\ln\frac{k}{\delta}\Big(\ln\frac{1}{\Delta_i}\Big)\bigg)$*, where* $\Theta_{[k]} := \max_{S \subseteq [n] || S| = k}\sum_{i \in S}\theta_i$*.*

Proof. The proof is based on the following four main observations:

1. The Best-Item a^* (i.e. item 1 in our case) is likely to beat the (ϵ_s, δ_s)-PAC item by sufficiently high margin, for any sub-phase $s = 1, 2 \ldots$, and hence is never discarded (see Lem. 19).
2. With high probability the set of suboptimal items get discarded at a fixed rate once played for sufficiently long duration (see Lem. 20).
3. The number of occurrences of any sub-optimal item i before it gets discarded is proportional to $O\Big(\frac{1}{\Delta_i^2}\Big)$ which yields the desired sample complexity of the algorithm (see Lem.21).
4. Lastly we show (using Thm. 6 and Lem. 14)) that the additional sample complexity incurred due to invoking the subroutine (ϵ,δ)-*PAC Best-Item* and *Score-Estimate* at every sub-phase s is orderwise same as the sample complexity incurred by *PAC-Wrapper* in the rest of the sub-phase, due to which (ϵ,δ)-*PAC Best-Item* so they do not actually contribute to the overall sample complexity of the algorithm modulo some constant factors.

While analysing any particular batch $\mathcal{B}_b$ of a given phase s, we will denote by $S = \mathcal{B}_b \setminus \{b_s\}$ and by $\Theta_S = \sum_{i \in S}\theta_i$. We will first prove the correctness of the algorithm, i.e. with high probability $(1-\delta)$, *PAC-Wrapper* indeed returns the Best-Item , i.e. item 1 in our case. We prove this using the following two lemmas: Lem. 19 and Lem. 20 respectively.

Lemma 19. *With high probability of at least* $(1 - \frac{\delta}{20})$*, item* 1 *is never eliminated, i.e.* $1 \in \mathcal{A}_s$ *for all sub-phase* s*. More formally, at the end of any sub-phase* $s = 1, 2, \ldots$*,* $\hat{p}_{1b_s} > \frac{1}{2} - \epsilon_s$*.*

Proof. Firstly note that at any sub-phase s, each batch $b \in B$ within that phase is played for $t_s = \frac{2\Theta_{[k]}}{\epsilon_s^2}\ln\frac{k}{\delta_s}$ rounds. Now consider the batch $\mathcal{B} \ni 1$ at any phase s. Clearly $b_s \in \mathcal{B}$ too. Again since b_s is returned by Alg. 5, by Thm. 6 we know that with probability at least $(1-\delta_s)$, $p_{b_s 1} > \frac{1}{2} - \epsilon_s \implies \theta_{b_s} > \theta_1 - 4\epsilon$. This further implies $\theta_{b_s} \geq \theta_1 - \frac{1}{2} = \frac{1}{2}$ (since we assume $\theta_1 = 1$, and at any s, $\epsilon_s < \frac{1}{8}$). Moreover by Lem. 14, we have $\hat{\Theta}_S \geq \frac{\theta_{b_s} + \Theta_S}{\theta_{b_s}} > \frac{\Theta_S + 1}{2}$ (recall we denote $S = \mathcal{B}_b \setminus \{b_s\}$)

Now let us define w_i as number of times item $i \in \mathcal{B}$ was returned as the winner in t_s rounds and i_τ be the winner retuned by the environment upon playing $\mathcal{B}$ for the τ^{th} round, where $\tau \in [t_s]$. Then clearly $Pr(\{i_\tau = 1\}) = \frac{\theta_1}{\sum_{j\in\mathcal{B}}\theta_j} = \frac{1}{\theta_{b_s}+\Theta_S} \geq \frac{1}{1+\Theta_S}, \forall \tau \in [t_s]$, as $1 := \arg\max_{i\in\mathcal{B}}\theta_i$. Hence $\mathbf{E}[w_1] = \sum_{\tau=1}^{t_s}\mathbf{E}[\mathbf{1}(i_\tau = 1)] = \frac{t_s}{\theta_{b_s}+\Theta_S} \geq \frac{t_s}{(1+\Theta_S)}$. Now assuming b_s to be indeed an (ϵ_s,δ_s)-PAC Best-Item and the bound of Lem. 14 to hold good as well, applying the multiplicative form of the Chernoff-Hoeffding bound on the random variable w_1, we get that for any $\eta \in (\sqrt{2}\epsilon_s, 1]$,

$$
\begin{aligned}
Pr\Big(w_1 \leq (1-\eta)\mathbf{E}[w_1]\Big) &\leq \exp\bigg(-\frac{\mathbf{E}[w_1]\eta^2}{2}\bigg) \leq \exp\bigg(-\frac{t_s\eta^2}{2(1+\Theta_S)}\bigg)\\
&= \exp\bigg(-\frac{2\hat{\Theta}_S\eta^2}{2\epsilon_s^2(1+\Theta_S)}\ln\frac{k}{\delta_s}\bigg)\\
&\overset{(a)}{\leq} \exp\bigg(-\frac{2(\Theta_S+1)\eta^2}{4\epsilon_s^2(1+\Theta_S)}\ln\frac{k}{\delta_s}\bigg)\\
&\leq \exp\bigg(-\frac{\eta^2}{2\epsilon_s^2}\ln\bigg(\frac{k}{\delta_s}\bigg)\bigg) \leq \exp\bigg(-\ln\bigg(\frac{k}{\delta_s}\bigg)\bigg) = \frac{\delta_s}{k},
\end{aligned}
$$

where (a) holds since we proved $\hat{\Theta}_S \geq \frac{\theta_{b_s}+\Theta_S}{\theta_{b_s}} > \frac{\Theta_S+1}{2}$, and the last inequality holds as $\eta > \epsilon_s\sqrt{2}$.

In particular, note that $\epsilon_s \leq \frac{1}{8}$ for any sub-phase s, due to which we can safely choose $\eta = \frac{1}{2}$ for any s, which gives that with probability at least $\big(1-\frac{\delta_s}{k}\big)$, $w_1 > (1-\frac{1}{2})\mathbf{E}[w_1] > \frac{t_s}{2(\theta_{b_s}+\Theta_S)}$, for any subphase s.

Thus above implies that with probability atleast $(1-\frac{\delta_s}{k})$, after t_s rounds we have $w_{1b_s} \geq \frac{t_s}{2(\theta_{b_s}+\Theta_S)} \implies w_{1b_s} + w_{b_s1} \geq \frac{t_s}{2(\theta_{b_s}+\Theta_S)}$. Let us denote $n_{1b_s} = w_{1b_s} + w_{b_s1}$. Then the probability of the event:

$$
\begin{aligned}
Pr\bigg(\hat{p}_{1b_s} < \frac{1}{2} - \epsilon_s, n_{1b_s} \geq \frac{t_s}{2(\theta_{b_s}+\Theta_S)}\bigg) &= Pr\bigg(\hat{p}_{1b_s} - \frac{1}{2} < -\epsilon_s, n_{1b_s} \geq \frac{t_s}{2(\theta_{b_s}+\Theta_S)}\bigg)\\
&\leq Pr\bigg(\hat{p}_{1b_s} - p_{1b_s} < -\epsilon_s, n_{1b_s} \geq \frac{t_s}{2(\theta_{b_s}+\Theta_S)}\bigg) \quad (\text{as } \mathbf{p}_{1b_s} > \frac{1}{2})\\
&\overset{(a)}{\leq} \exp\Big(-2\frac{t_s}{2(\theta_{b_s}+\Theta_S)}\big(\epsilon_s\big)^2\Big)\\
&\leq \exp\Big(-\frac{2(\theta_{b_s}+\Theta_S)}{\epsilon_s^2\theta_{b_s}(\theta_{b_s}+\Theta_S)}\big(\epsilon_s\big)^2\Big) \leq \frac{\delta_s}{k},
\end{aligned}
$$

where the last inequality (a) follows from Lem. 22 for $\eta = \epsilon_s$ and $v = \frac{t_s}{2k}$.

Thus under the two assumptions that (1). b_s is indeed an (ϵ_s,δ_s)-PAC Best-Item and (2). the bound of Lem. 14 holds good, combining the above two claims, at any sub-phase s, we have

$$
\begin{aligned}
&Pr\bigg(\hat{p}_{1b_s} < \frac{1}{2} - \epsilon_s\bigg)\\
&= Pr\bigg(\hat{p}_{1b_s} < \frac{1}{2} - \epsilon_s, n_{1b_s} \geq \frac{t_s}{2(\theta_{b_s}+\Theta_S)}\bigg) + Pr\bigg(\hat{p}_{1b_s} < \frac{1}{2} - \epsilon_s, n_{1b_s} < \frac{t_s}{2(\theta_{b_s}+\Theta_S)}\bigg)\\
&\leq \frac{\delta_s}{k} + Pr\bigg(n_{1b_s} < \frac{t_s}{2(\theta_{b_s}+\Theta_S)}\bigg) \leq \frac{2\delta_s}{k} \leq \delta_s \quad (\text{since } k \geq 2)
\end{aligned}
$$

Moreover from Thm. 6 and Lem. 14 we know that the above two assumptions hold with probability at least $(1-2\delta_s)$. Then taking union bound over all sub-phases $s = 1, 2, \ldots$, the probability that item 1 gets eliminated at any round:

$$Pr\bigg(\exists s = 1,2,\ldots \text{ s.t. } \hat p_{1b_s} < \frac{1}{2} - \epsilon_s\bigg) = \sum_{s=1}^{\infty} 3\delta_s = \sum_{s=1}^{\infty}\frac{3\delta}{120s^3} \le \frac{\delta}{40}\sum_{s=1}^{\infty}\frac{1}{s^2} \le \frac{\delta}{40}\frac{\pi^2}{6} \le \frac{\delta}{20},$$

where the first inequality holds since $k \ge 2$. □

We next introduce few notations before proceeding to the next claim of Lem. 20.

Notations. Recall that we defined $\Delta_i = \theta_1 - \theta_i$ (Sec. 2). We further denote $\Delta_{\min} = \min_{i\in[n]\setminus\{1\}}\Delta_i$. We define the set of arms $[n]_r := \{i \in [n] : \frac{1}{2^r} \le \Delta_i < \frac{1}{2^{r-1}}\}$, and denote the set of surviving arms in $[n]_r$ at s^{th} sub-phase by $\mathcal{A}_{r,s}$, i.e. $\mathcal{A}_{r,s} = [n]_r \cap \mathcal{A}_s$, for all $s = 1, 2, \ldots$.

Lemma 20. *Assuming that the best arm 1 is not eliminated at any sub-phase $s = 1, 2, \ldots$, then with probability at least $(1 - \frac{19\delta}{20})$, for any sub-phase $s \ge r$, $|\mathcal{A}_{r,s}| \le \frac{2}{19}|\mathcal{A}_{r,s-1}|$, for any $r = 0, 1, 2, \ldots \log_2(\Delta_{\min})$.*

Proof. Consider any sub-phase s, and let us start by noting some properties of b_s. Note that by Lem. 6, with high probability $(1-\delta_s)$, $p_{b_s1} > \frac{1}{2} - \epsilon_s$. Then this further implies

$$\begin{aligned} p_{b_s1} > \frac{1}{2} - \epsilon_s &\implies \frac{(\theta_{b_s} - \theta_1)}{2(\theta_{b_s} + \theta_1)} > -\epsilon_s \\ &\implies (\theta_{b_s} - \theta_1) > -2\epsilon_s(\theta_{b_s} + \theta_1) > -4\epsilon_s \;\; (\text{as } \theta_i \in (0,1)\, \forall i \in [n]\setminus\{1\}) \end{aligned}$$

So we have with probability atleast $(1-\delta_s)$, $\theta_{b_s} > \theta_1 - 4\epsilon_s = \theta_1 - \frac{1}{2^s}$.

Now consider any fixed $r = 0, 1, 2, \ldots \log_2(\Delta_{\min})$. Clearly by definition, for any item $i \in [n]_r$, $\Delta_i = \theta_1 - \theta_i > \frac{1}{2^r}$.

Then combining the above two claims, we have for any sub-phase $s \ge r$, $\theta_{b_s} > \theta_1 - \frac{1}{2^s} \ge \theta_i + \frac{1}{2^r} - \frac{1}{2^s} > 0 \implies p_{b_si} > \frac{1}{2}$, at any $s \ge r$.

Moreover note that for any $s \ge 1$, $\epsilon_s > \frac{1}{8}$, so that implies $\theta_{b_s} > \theta_1 - 4\epsilon_s > \frac{1}{2}$.

Recall that at any sub-phase s, each batch within that phase is played for $t_s = \frac{2\Theta_{[k]}}{\epsilon_s^2}\ln\frac{k}{\delta_s}$ many rounds. Now consider any batch such that $\mathcal{B} \ni i$ for any $i \in [n]_r$. Of course $b_s \in \mathcal{B}$ as well, and note that we have shown $p_{b_si} > \frac{1}{2}$ with high probability $(1-\delta_s)$.

Same as Lem. 19, let us again define w_i as number of times item $i \in \mathcal{B}$ was returned as the winner in t_s rounds, and i_τ be the winner retuned by the environment upon playing $\mathcal{B}$ for the τ^{th} rounds, where $\tau \in [t_s]$. Then given $\theta_{b_s} > \frac{1}{2}$ (as derived earlier), clearly $Pr(\{i_\tau = b_s\}) = \frac{\theta_{b_s}}{\theta_{b_s} + \Theta_S}$, $\forall \tau \in [t_s]$. Hence $\mathbf{E}[w_{b_s}] = \sum_{\tau=1}^{t_s}\mathbf{E}[\mathbf{1}(i_\tau = b_s)] = \frac{\theta_{b_s}t_s}{(\theta_{b_s}+\Theta_S)}$. Now applying multiplicative Chernoff-Hoeffdings bound on the random variable w_{b_s}, we get that for any $\eta \in (\sqrt{2}\epsilon_s, 1]$,

$$\begin{aligned} &Pr\Big(w_{b_s} \le (1-\eta)\mathbf{E}[w_{b_s}] \mid \theta_{b_s} > \frac{1}{2}, \hat\Theta_S > \frac{(\theta_{b_s}+\Theta_S)}{\theta_{b_s}}\Big) \\ &\le \exp\bigg(-\frac{\mathbf{E}[w_{b_s}]\eta^2}{2}\bigg) \le \exp\bigg(-\frac{\theta_{b_s}t_s\eta^2}{2(\theta_{b_s}+\Theta_S)}\bigg) \\ &\le \exp\bigg(-\frac{\eta^2}{2\epsilon_s^2}\ln\bigg(\frac{k}{\delta_s}\bigg)\bigg) \le \exp\bigg(-\ln\bigg(\frac{k}{\delta_s}\bigg)\bigg) = \frac{\delta_s}{k}, \end{aligned}$$

where the last inequality holds as $\eta > \sqrt{2}\epsilon_s$. So as a whole, for any $\eta \in (\sqrt{2}\epsilon_s, 1]$,

$$
\begin{aligned}
&Pr\Big(w_{b_s} \le (1-\eta)\mathbf{E}[w_{b_s}]\Big)\\
&\le Pr\Big(w_{b_s} \le (1-\eta)\mathbf{E}[w_{b_s}] \mid \theta_{b_s} > \frac{1}{2}, \hat\Theta_S > \frac{(\theta_{b_s}+\Theta_S)}{\theta_{b_s}}\Big) Pr\Big(\theta_{b_s} > \frac{1}{2}\Big) + Pr\Big(\theta_{b_s} < \frac{1}{2}, \hat\Theta_S > \frac{(\theta_{b_s}+\Theta_S)}{\theta_{b_s}}\Big)\\
&\le \frac{\delta_s}{k} + 2\delta_s
\end{aligned}
$$

In particular, note that $\epsilon_s < \frac{1}{8}$ for any sub-phase s, due to which we can safely choose $\eta = \frac{1}{2}$ for any s, which gives that with probability at least $\big(1-\frac{\delta_s}{k}\big)$, $w_{b_s} > (1-\frac{1}{2})\mathbf{E}[w_{b_s}] > \frac{t_s\theta_{b_s}}{2(\Theta_S+\theta_{b_s})}$, for any subphase s.

Thus above implies that with probability at least $(1-\frac{\delta_s}{k} - 2\delta_s)$, after t_s rounds we have $w_{b_s i} \ge \frac{t_s\theta_{b_s}}{2(\Theta_S+\theta_{b_s})} \implies w_{ib_s} + w_{b_s i} \ge \frac{t_s\theta_{b_s}}{2(\Theta_S+\theta_{b_s})}$. Let us denote $n_{ib_s} = w_{ib_s} + w_{b_s i}$. Then the probability that item i is not eliminated at any sub-phase $s \ge r$ is:

$$
\begin{aligned}
Pr\Bigg(\hat p_{ib_s} > \frac{1}{2} - \epsilon_s, n_{ib_s} \ge \frac{t_s\theta_{b_s}}{2(\Theta_S+\theta_{b_s})}\Bigg) &= Pr\Bigg(\hat p_{ib_s} - \frac{1}{2} > -\epsilon_s, n_{ib_s} \ge \frac{t_s\theta_{b_s}}{2(\Theta_S+\theta_{b_s})}\Bigg)\\
&\le Pr\Bigg(\hat p_{ib_s} - p_{ib_s} > -\epsilon_s, n_{1b_s} \ge \frac{t_s\theta_{b_s}}{2(\Theta_S+\theta_{b_s})}\Bigg) \quad (\text{as } \mathbf{p}_{ib_s} < \frac{1}{2})\\
&\overset{(a)}{\le} \exp\Big(-2\frac{t_s}{2(\Theta_S+\theta_{b_s})}(\epsilon_s)^2\Big)\\
&\le \exp\Big(-2\frac{(\Theta_S+\theta_{b_s})}{2\theta_{b_s}(\Theta_S+\theta_{b_s})}(\epsilon_s)^2 \ln\frac{\delta_s}{k}\Big) \le \frac{\delta_s}{k},
\end{aligned}
$$

where (a) follows from Lem. 22 for $\eta = \epsilon_s$ and $v = \frac{t_s}{2k}$.

Now combining the above two claims, at any sub-phase s, we have:

$$
\begin{aligned}
Pr\Bigg(\hat p_{ib_s} > \frac{1}{2} - \epsilon_s\Bigg) &= Pr\Bigg(\hat p_{ib_s} > \frac{1}{2} - \epsilon_s, n_{ib_s} \ge \frac{t_s}{4k}\Bigg) + Pr\Bigg(\hat p_{ib_s} > \frac{1}{2} - \epsilon_s, n_{ib_s} < \frac{t_s}{4k}\Bigg)\\
&\le \frac{\delta_s}{k} + 2\delta_s + \frac{\delta_s}{k} \le 3\delta_s \quad (\text{since } k \ge 2)
\end{aligned}
$$

This consequently implies that for any sup-phase $s \ge r$, $\mathbf{E}[|\mathcal{A}_{r,s}|] \le 3\delta_s \mathbf{E}[|\mathcal{A}_{r,s-1}|]$. Then applying Markov's Inequality we get:

$$
Pr\Bigg(|\mathcal{A}_{r,s}| \le \frac{2}{19}|\mathcal{A}_{r,s-1}|\Bigg) \le \frac{3\delta_s|\mathcal{A}_{r,s-1}|}{\frac{2}{19}|\mathcal{A}_{r,s-1}|} = \frac{57\delta_s}{2}
$$

Finally applying union bound over all sub-phases $s = 1, 2, \ldots$, and all $r = 1, 2, \ldots s$, we get:

$$
\sum_{s=1}^{\infty}\sum_{r=1}^{s}\frac{57\delta_s}{2} = \sum_{s=1}^{\infty} s\frac{57\delta}{240s^3} = \frac{57\delta}{240}\sum_{s=1}^{\infty}\frac{1}{s^2} \le \frac{57\delta}{240}\frac{\pi^2}{6} \le \frac{57\delta}{120} \le \frac{19\delta}{20}.
$$

$\square$

Thus combining Lem. 19 and 20, we get that the total failure probability of *PAC-Wrapper* is at most $\frac{\delta}{20} + \frac{19\delta}{20} = \delta$.

The remaining thing is to prove the sample complexity bound which crucially relies on the following claim. At any sub-phase s, we call the item b_s as the pivot element of phase s.

Lemma 21. *Assume both Lem. 19 and Lem. 20 holds good and the algorithm does not do a mistake. Consider any item $i \in [n]_r$, for any $r = 1, 2, \ldots \log_2(\Delta_{\min})$. Then the total number of times item i gets played as a non-pivot item (i.e. appear in at most one of the k-subsets per sub-phase s) during the entire run of Alg. 1 is $O\Big((2^r)^2\Theta_{[k]} \ln \frac{rk}{\delta}\Big)$.*

Proof. Let us denote the sample complexity of item i (as a non-pivot element) from phase x to y as $\mathcal{N}^{(i)}_{x,y}$, for any $1 \le x < y < \infty$. Additionally, recalling from Lem. 14 that $\hat{\Theta}_S \le 7\Theta_{[k]}$, we now prove the claim with the following two case analyses:

(Case 1) Sample complexity till sub-phase $s = r-1$: Note that in the worst case item i can get picked at every sub-phase $s = 1, 2, \ldots r-1$, and at every s it is played for t_s round. Additionally, recalling from Lem. 14 that $\hat{\Theta}_S \le 7\Theta_{[k]}$, the total number of plays of item $i \in [n]_r$ (as a non-pivot item), till sub-phase $r-1$ becomes:

$$\mathcal{N}^{(i)}_{1,r-1} \le \sum_{s=1}^{r-1} t_s = \sum_{s=1}^{r-1} \frac{14\Theta_{[k]}}{\epsilon_s^2} \ln \frac{k}{\delta_s} = \frac{14\Theta_{[k]}}{4^{-2}} \sum_{s=1}^{r-1} (2^s)^2 \ln \frac{120k^3}{\delta} = O\Big((2^r)^2\Theta_{[k]} \ln \frac{rk}{\delta}\Big)$$

(Case 2) Sample complexity from sub-phase $s \ge r$ onwards: Assuming Lem. 20 holds good, note that if we define a random variable I_s for any sub-phase $s \ge r$ such that $I_s = \mathbf{1}(i \in \mathcal{A}_s)$, then clearly $\mathbf{E}[I_s] \le \frac{2}{19}\mathbf{E}[I_{s-1}]$ (as follows from the analysis of Lem. 19). Then the total expected sample complexity of item $i \in [n]_r$ for round $r, r+1, \ldots \infty$ becomes:

$$\begin{aligned}
\mathcal{N}^{(i)}_{r,\infty} &\le 224\Theta_{[k]} \sum_{s=r}^{\infty} \left(\frac{2}{19}\right)^{s-r+1} 4^s \ln \frac{k}{\delta_s} = 224\Theta_{[k]}(2^r)^2 \sum_{s=0}^{\infty} \left(\frac{2}{19}\right)^{s+1} (2^s)^2 \ln \frac{120k(s+r)^3}{\delta} \\
&= \frac{448}{19}\Theta_{[k]}(2^r)^2 \sum_{s=0}^{\infty} \left(\frac{8}{19}\right)^{s} \ln \frac{120k(s+r)^3}{\delta} \\
&\le \frac{448}{19}\Theta_{[k]}(2^r)^2 \left[\ln \frac{120kr}{\delta} \sum_{s=0}^{\infty} \left(\frac{8}{19}\right)^{s} + \sum_{s=0}^{\infty} \left(\frac{8}{19}\right)^{s} \ln(120ks) \right] = O\Big((2^r)^2\Theta_{[k]} \ln \frac{rk}{\delta}\Big)
\end{aligned}$$

Combining the two cases above we get $\mathcal{N}^{(i)}_{1,\infty} = O\Big((2^r)^2\Theta_{[k]} \ln \frac{rk}{\delta}\Big)$ as well, which concludes the proof. □

Following similar notations as $\mathcal{N}^{(i)}_{x,y}$, we now denote the number of times any k-subset $S \subseteq [n]$ played by the algorithm in sub-phase x to y as $\mathcal{N}^{(S)}_{x,y}$. Then using Lem. 21, the total sample complexity of the algorithm *PAC-Wrapper* (lets call it algorithm $\mathcal{A}$) can be written as:

$$\begin{aligned}
\mathcal{N}_{\mathcal{A}}(0,\delta) &= \sum_{S \subset [n] \mid |S| = k} \sum_{s=1}^{\infty} \mathbf{1}(S \in \{\mathcal{B}_1, \ldots, \mathcal{B}_{B_s}\}) t_s = \sum_{i \in [n]} \sum_{s=1}^{\infty} \frac{\mathbf{1}(i \in \mathcal{A}_s \setminus \{b_s\})}{k-1} t_s \\
&= \sum_{s=1}^{\infty} \sum_{i \in [n]} \frac{\mathbf{1}(i \in \mathcal{A}_s \setminus \{b_s\})}{k-1} t_s = \sum_{s=1}^{\infty} \sum_{r=1}^{\log_2(\Delta_{\min})} \sum_{i \in [n]_r} \frac{\mathbf{1}(i \in \mathcal{A}_s \setminus \{b_s\})}{k-1} t_s \\
&= \frac{1}{k-1} \sum_{r=1}^{\log_2(\Delta_{\min})} \sum_{i \in [n]_r} \mathcal{N}^{(i)}_{1,\infty} \quad (\text{ Lem. } 21) \\
&= \frac{1}{k-1} \sum_{r=1}^{\log_2(\Delta_{\min})} \sum_{i \in [n]_r} O\Big((2^r)^2\Theta_{[k]} \ln \frac{rk}{\delta}\Big)
\end{aligned}$$

$$= \frac{\Theta_{[k]}}{k-1} \sum_{r=1}^{\log_2(\Delta_{\min})} |[n]_r| O\Big((2^r)^2 \ln \frac{rk}{\delta}\Big) = O\Big(\frac{\Theta_{[k]}}{k} \sum_{i=2}^{n} \frac{1}{\Delta_i^2} \ln\big(\frac{k}{\delta} \ln \frac{1}{\Delta_i}\big)\Big), \tag{2}$$

where the last inequality follows since $2^r < \frac{2}{\Delta_i}$ by definition for all $i \in [n]_r$. Finally the last thing to account for is the additional sample complexity incurred due to calling the subroutine (ϵ, δ)-*PAC Best-Item* and *Score-Estimate* at every sub-phase s, which is combinedly known to be of $O\bigg(\frac{|\mathcal{A}_s|\Theta_[k]}{k}\Big(1, \frac{1}{(2^s)^2}\Big) \ln \frac{k}{\delta}\bigg)$ at any sub-phase s (from Thm. 6 and Cor. 15). And using a similar summation as shown above over all $s = 1, 2, \ldots \infty$, combined with Lem. 20 and using the fact that $2^r < \frac{2}{\Delta_i}$, one can show that the total sample complexity incurred due to the above subroutines is at most $\frac{\Theta_{[k]}}{k} \sum_{i=2}^{n} \max\big(1, \frac{1}{\Delta_i^2}\big) \log \frac{k}{\delta}$). Considering the above sample complexity added with that derived in Eqn. 2 finally gives the desired $O\bigg(\frac{\Theta_{[k]}}{k} \sum_{i=2}^{n} \max\Big(1, \frac{1}{\Delta_i^2}\Big) \ln \frac{k}{\delta}\Big(\ln \frac{1}{\Delta_i}\Big)\bigg)$ sample complexity bound of Alg. 1.

□

Lemma 22 (Deviations of pairwise win-probability estimates for the PL model (Saha & Gopalan, 2019))**.** *Consider a Plackett-Luce choice model with parameters $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_n)$, and fix two distinct items $i, j \in [n]$. Let $S_1, \ldots, S_T$ be a sequence of (possibly random) subsets of $[n]$ of size at least 2, where T is a positive integer, and $i_1, \ldots, i_T$ a sequence of random items with each $i_t \in S_t$, $1 \le t \le T$, such that for each $1 \le t \le T$, (a) S_t depends only on $S_1, \ldots, S_{t-1}$, and (b) i_t is distributed as the Plackett-Luce winner of the subset S_t, given $S_1, i_1, \ldots, S_{t-1}, i_{t-1}$ and S_t, and (c) $\forall t : \{i, j\} \subseteq S_t$ with probability 1. Let $n_i(T) = \sum_{t=1}^{T} \mathbf{1}(i_t = i)$ and $n_{ij}(T) = \sum_{t=1}^{T} \mathbf{1}(\{i_t \in \{i, j\}\})$. Then, for any positive integer v, and $\eta \in (0, 1)$,*

$$Pr\left(\frac{n_i(T)}{n_{ij}(T)} - \frac{\theta_i}{\theta_i + \theta_j} \ge \eta,\ n_{ij}(T) \ge v\right) \vee Pr\left(\frac{n_i(T)}{n_{ij}(T)} - \frac{\theta_i}{\theta_i + \theta_j} \le -\eta,\ n_{ij}(T) \ge v\right) \le e^{-2v\eta^2}.$$

A.5. Modified version of *PAC-Wrapper* (Alg. 1) for general (ϵ, δ)-PAC guarantee (for any $\epsilon \in [0, 1]$)

Algorithm 6 Modified *PAC-Wrapper* (for a general (ϵ, δ)-PAC guarantee)

1: **input:** Set of items: $[n]$, Subset size: $n \geq k > 1$, Confidence term $\delta > 0$
2: **init:** $\mathcal{A}_0 \leftarrow [n]$, $s \leftarrow 1$
3: **while** $|\mathcal{A}_{s-1}| > 1$ **do**
4:   Set $\epsilon_s = \frac{1}{2^{s+2}}$, $\delta_s = \frac{\delta}{120s^3}$, and $\mathcal{R}_s \leftarrow \emptyset$.
5:   $b_s \leftarrow (\epsilon, \delta)$-*PAC Best-Item*$(\mathcal{A}_{s-1}, k, 1, \epsilon_s/4, \delta_s)$
6:   **If** $(\epsilon_s \leq \epsilon)$ **then** $\mathcal{A}' \leftarrow \{b_s\}$, and exit the **while** loop (go to Line 35)
7:   $\mathcal{B}_1, \ldots \mathcal{B}_{B_s} \leftarrow$ *Partition*$(\mathcal{A}_{s-1} \setminus \{b_s\}, k-1)$
8:   **if** $|\mathcal{B}_{B_s}| < k-1$, **then** $\mathcal{R}_s \leftarrow \mathcal{B}_{B_s}$ and $B_s = B_s - 1$
9:   **for** $b = 1, 2 \ldots B_s$ **do**
10:     $\hat{\Theta}_S \leftarrow$ *Score-Estimate*$(b_s, \mathcal{B}_b, \delta_s)$. Set $\hat{\Theta}_S \leftarrow \max(2\hat{\Theta}_S + 1, 2)$.
11:     Set $\mathcal{B}_b \leftarrow \mathcal{B}_b \cup \{b_s\}$
12:     Play $\mathcal{B}_b$ for $t_s := \frac{2\hat{\Theta}_S}{\epsilon_s^2} \ln \frac{k}{\delta_s}$ rounds
13:     Receive the winner feedback: $\sigma_1, \sigma_2, \ldots \sigma_{t_s} \in \boldsymbol{\Sigma}^1_{\mathcal{B}_b}$ after each respective t_s rounds.
14:     Update pairwise empirical win-count w_{ij} using *Rank-Breaking* on $\sigma_1 \ldots \sigma_{t_s}$, $\forall i, j \in \mathcal{B}_b$
15:     $\hat{p}_{ij} := \frac{w_{ij}}{w_{ij} + w_{ji}}$ for all $i, j \in \mathcal{B}_b$
16:     **If** $\exists i \in \mathcal{B}_b$ s.t. $\hat{p}_{ib_s} > \frac{1}{2} - \epsilon_s$, **then** $\mathcal{A}_s \leftarrow \mathcal{A}_s \cup \{i\}$
17:   **end for**
18:   $\mathcal{A}_s \leftarrow \mathcal{A}_s \cup \mathcal{R}_s$, $s \leftarrow s + 1$
19:   **if** $1 < |\mathcal{A}_{s-1}| \leq k$ **then**
20:     Append $\mathcal{A}_{s-1}$ with any $(k - |\mathcal{A}_{s-1}|)$ elements from $[n] \setminus \mathcal{A}_{s-1}$
21:     Pairwise empirical win-count $w_{ij} \leftarrow 0$, $\forall i, j \in \mathcal{A}_{s-1}$; $\mathcal{A} \leftarrow \mathcal{A}_{s-1}$; $\mathcal{A}' \leftarrow \mathcal{A}_{s-1}$
22:     **while** $|\mathcal{A}'| > 1$ **do**
23:       Set $\epsilon_s = \frac{1}{2^{s+2}}$, and $\delta_s = \frac{\delta}{1200s^3}$
24:       $b_s \leftarrow (\epsilon, \delta)$-*PAC Best-Item*$(\mathcal{A}, k, 1, \epsilon_s/4, \delta_s)$
25:       **If** $(\epsilon_s \leq \epsilon)$ **then** $\mathcal{A}' \leftarrow \{b_s\}$, and exit the **while** loops (go to Line 35)
26:       $\hat{\Theta}_S \leftarrow$ *Score-Estimate*$(b_s, \mathcal{A} \setminus \{b_s\}, \delta_s)$. Set $\hat{\Theta}_S \leftarrow \max(2\hat{\Theta}_S + 1, 2)$.
27:       Play $\mathcal{B}$ for $t_s := \frac{2\hat{\Theta}_S}{\epsilon_s^2} \ln \frac{k}{\delta_s}$ rounds, and receive the corresponding winner feedback: $\sigma_1, \ldots \sigma_{t_s} \in \boldsymbol{\Sigma}^1_{\mathcal{A}}$ per round.
28:       Update pairwise empirical win-count w_{ij} using *Rank-Breaking* on $\sigma_1 \ldots \sigma_{t_s}$, $\forall i, j \in \mathcal{A}'$
29:       Update $\hat{p}_{ij} := \frac{w_{ij}}{w_{ij} + w_{ji}}$ for all $i, j \in \mathcal{A}'$
30:       **If** $\exists i \in \mathcal{A}'$ with $\hat{p}_{ib_s} < \frac{1}{2} - \epsilon_s$, **then** $\mathcal{A}' \leftarrow \mathcal{A}' \setminus \{i\}$
31:       $s \leftarrow s + 1$
32:     **end while**
33:   **end if**
34: **end while**
35: **output:** The item remaining in $\mathcal{A}'$

A.6. Proof of Thm. 4

Theorem 4 (*PAC-Wrapper* (ϵ, δ)-PAC sample complexity bound with Winner feedback)**.** *For any $\epsilon \in [0, 1]$, with probability at least $(1 - \delta)$, $\mathcal{A}$ as PAC-Wrapper (Algorithm 1) returns the ϵ-Best-Item (see Defn. 2) with sample complexity* $N_{\mathcal{A}}(\epsilon, \delta) = O\bigg(\frac{\Theta_{[k]}}{k} \sum_{i=2}^{n} \max\Big(1, \frac{1}{\max(\Delta_i, \epsilon)^2}\Big) \ln \frac{k}{\delta}\Big(\ln \frac{1}{\max(\Delta_i, \epsilon)}\Big)\bigg)$.

Proof. Let us denote by s_0 to be the sub-phase at which ϵ_s falls below ϵ for the first time, i.e. $s_0 := \arg\min_{s=1,2,\ldots} \mathbf{1}(\epsilon_s \leq \epsilon)$. We first proof the (ϵ, δ)-PAC correctness of the algorithm:

(Proof of Correctness): Note from Lem. 19 that the probability the Best-Item 1 gets eliminated till sub-phase $s = 1, 2, \ldots (s_0 - 1)$ is upper bounded by $\sum_{s=1}^{s_0-1} \frac{2\delta_s}{k} \leq \sum_{s=1}^{s_0-1} \delta_s$, since $k \geq 2$.

So with probability at least $(1 - \sum_{s=1}^{s_0-1} \delta_s)$, item 1 survives till the beginning of sub-phase s_0. And by Thm. 4, we know that with probability at least $(1-\delta_s)$, $p_{b_s 1} > \frac{1}{2} - \epsilon_s/4 \implies \theta_{b_s} > \theta_1 - \epsilon_s$, which ensures ϵ_s optimality of the item b_s (see Defn. 2). So at $s = s_0$, we have $Pr(\theta_{b_{s_0}} > \theta_1 - \epsilon_{s_0}) > (1-\delta_{s_0})$ which ensures the (ϵ,δ) correctness of the algorithm as at s_0, $\epsilon_{s_0} \le \epsilon$.

Moreover the over all probability of the algorithm failing to return an ϵ-optimal item is $\sum_{s=1}^{s_0-1}\delta_s + \delta_{s_0} \le \sum_{s=1}^{\infty}\delta_s = \frac{\delta}{120}\sum_{s=1}^{\infty}\frac{1}{s^2} \le \frac{\delta}{120}\frac{\pi^2}{6} \le \frac{\delta}{20}$.

For the rest of the analysis we will assume that the claim of Lem. 20 holds good for all $s = 1, 2\ldots s_0$, which we know to satisfy with probability at least $(1-\frac{19\delta}{20})$.

(Proof for Sample-complexity): We now proceed to prove the sample complexity of the algorithm. Let us call b_s to be the pivot item of any phase s, and denote the sample complexity of item i (as a non-pivot element) from phase x to y as $\mathcal{N}_{x,y}^{(i)}$, for any $1 \le x < y < \infty$. Additionally, recalling from Lem. 14 that $\hat{\Theta}_S \le 7\Theta_{[k]}$, we now prove the claim with the following two case analyses:

(Case 1) For suboptimal item $i \in [n]\setminus\{1\}$ such that $\Delta_i > \epsilon$: Recall from Lem. 21 that the sample complexity of item i (as a non-pivot) is $\mathcal{N}_{1,\infty}^{(i)} = O\Big((2^r)^2\Theta_{[k]}\ln\frac{rk}{\delta}\Big)$, where $i \in [n]_r$. Hence we further get $\mathcal{N}_{1,\infty}^{(i)} = O\Big(\frac{\Theta_{[k]}}{\Delta_i^2}\ln\Big(\frac{k}{\delta}\ln\frac{1}{\Delta_i}\Big)\Big)$ as since $i \in [n]_r$, so by definition $\Delta_i < \frac{2}{2^r}$.

(Case 2) For items i such that $\Delta_i \le \epsilon$: Recall due to Thm. 6 the orderwise sample complexity of playing the sets $\mathcal{B}_1,\ldots,\mathcal{B}_{B_s}$ is same as that incurred due to calling the subroutine (ϵ,δ)-*PAC Best-Item* at sub-phase s, for all $s = 1, 2, \ldots$. Now in the worst case, all items i with $\Delta_i < \epsilon$ might survive till phase s_0. Thus the maximum sample complexity of any such item i (as a non-pivot) till sub-phase s_0 can be upper bounded as:

$$
\begin{aligned}
\mathcal{N}_{1,\infty}^{(i)} = \mathcal{N}_{1,s_0}^{(i)} &\le \sum_{s=1}^{s_0} t_s = \sum_{s=1}^{r-1}\frac{14\Theta_{[k]}}{\epsilon_s^2}\ln\frac{k}{\delta_s} = \frac{14\Theta_{[k]}}{4^{-2}}\sum_{s=1}^{s_0}(2^s)^2\ln\frac{120ks^3}{\delta} \\
&= O\Big((2^{s_0+1})^2\Theta_{[k]}\ln\frac{(s_0+1)k}{\delta}\Big) = O\bigg(\frac{\Theta_{[k]}}{\epsilon_{s_0}^2}\ln\Big(\frac{k}{\delta}\ln\frac{1}{\epsilon_{s_0}}\Big)\bigg) = O\bigg(\frac{\Theta_{[k]}}{\epsilon^2}\ln\Big(\frac{k}{\delta}\ln\frac{1}{\epsilon}\Big)\bigg),
\end{aligned}
$$

where the last equality follows as $\epsilon < 2\epsilon_{s_0} = \epsilon_{s_0-1}$, by definition of s_0.

Now denoting the number of times any k-subset $S \subseteq [n]$ played by the algorithm in sub-phase x to y as $\mathcal{N}_{x,y}^{(S)}$, and using the claims from above two cases, the total sample complexity of the algorithm (lets call it algorithm $\mathcal{A}$) becomes:

$$
\begin{aligned}
\mathcal{N}_{\mathcal{A}}(0,\delta) &= \sum_{S\subset[n]||S|=k}\sum_{s=1}^{\infty}\mathbf{1}(S\in\{\mathcal{B}_1,\ldots,\mathcal{B}_{B_s}\})t_s = \sum_{i\in[n]}\sum_{s=1}^{\infty}\frac{\mathbf{1}(i\in\mathcal{A}_s\setminus\{b_s\})}{k-1}t_s \\
&= \sum_{s=1}^{\infty}\sum_{i\in[n]}\frac{\mathbf{1}(i\in\mathcal{A}_s\setminus\{b_s\})}{k-1}t_s \\
&= \sum_{s=1}^{\infty}\sum_{r=1}^{\log_2(\Delta_{\min})}\sum_{i\in[n]_r}\frac{\mathbf{1}(i\in\mathcal{A}_s\setminus\{b_s\})}{k-1}t_s = \frac{1}{k-1}\sum_{r=1}^{\log_2(\Delta_{\min})}\sum_{i\in[n]_r}\mathcal{N}_{1,\infty}^{(i)} \\
&= \frac{1}{k-1}\sum_{r=1}^{\log_2(\Delta_{\min})}\bigg(\sum_{\{i\in[n]_r|\Delta_i>\epsilon\}}\mathcal{N}_{1,\infty}^{(i)} + \sum_{\{i\in[n]_r|\Delta_i\le\epsilon\}}\mathcal{N}_{1,\infty}^{(i)}\bigg) \\
&= \frac{1}{k-1}\sum_{r=1}^{\log_2(\Delta_{\min})}\bigg(\sum_{\{i\in[n]_r|\Delta_i>\epsilon\}}O\Big(\frac{\Theta_{[k]}}{\Delta_i^2}\ln\Big(\frac{k}{\delta}\frac{1}{\Delta_i}\Big)\Big) + \sum_{\{i\in[n]_r|\Delta_i\le\epsilon\}}O\Big(\frac{\Theta_{[k]}}{\epsilon^2}\ln\Big(\frac{k}{\delta}\ln\frac{1}{\epsilon}\Big)\Big)\bigg) \\
&= O\bigg(\frac{\Theta_{[k]}}{k}\sum_{i=2}^{n}\frac{1}{\max(\Delta_i,\epsilon)^2}\ln\frac{k}{\delta}\Big(\ln\frac{1}{\max(\Delta_i,\epsilon)}\Big)\bigg),
\end{aligned}
$$

where note that the second last inequality is follows from Case 1 and 2 derived above. Finally, as shown in the proof of

Algorithm 7 ***PAC-Wrapper (for Top-m Ranking feedback)***

```
input: Set of items: [n], Subset size: n ≥ k > 1, Ranking feedback size: m ∈ [k − 1], Confidence term δ > 0
init: A_0 ← [n], s ← 1
while |A_{s−1}| ≥ k do
    Set ε_s = 1/2^{s+2}, δ_s = δ/(120s^3), and R_s ← ∅.
    b_s ← (ε, δ)-PAC Best-Item(A_{s−1}, k, m, ε_s, δ_s)
    B_1, ... B_{B_s} ← Partition(A_{s−1} \ {b_s}, k − 1)
    if |B_{B_s}| < k − 1, then R_s ← B_{B_s} and B_s = B_s − 1
    for b = 1, 2 ... B_s do
        Θ̂_S ← Score-Estimate(b_s, B_b, δ_s). Set Θ̂_S ← max(2Θ̂_S + 1, 2).
        Set B_b ← B_b ∪ {b_s}
        Play B_b for t_s := (2Θ̂_S/(m ε_s^2)) ln(k/δ_s) rounds
        Receive the winner feedback: σ_1, σ_2, ... σ_{t_s} ∈ Σ^m_{B_b} after each respective t_s rounds.
        Update pairwise empirical win-count w_ij using Rank-Breaking on σ_1 ... σ_{t_s}, ∀i, j ∈ B_b
        p̂_ij := w_ij/(w_ij + w_ji) for all i, j ∈ B_b
        If ∃i ∈ B_b with p̂_{ib_s} > 1/2 − ε_s, then A_s ← A_s ∪ {i}
    end for
    A_s ← A_s ∪ R_s, s ← s + 1
end while
A ← A_{s−1}; B ← A_{s−1} ∪ {(k − |A_{s−1}|) elements from [n] \ A_{s−1}}
Pairwise empirical win-count w_ij ← 0, ∀i, j ∈ A
while |A| > 1 do
    Set ε_s = 1/2^{s+2}, and δ_s = δ/(120s^3)
    b_s ← (ε, δ)-PAC Best-Item(B, k, m, ε_s, δ_s)
    Θ̂_S ← Score-Estimate(b_s, A \ {b_s}, δ_s). Set Θ̂_S ← max(2Θ̂_S + 1, 2).
    Play B for t_s := (2Θ̂_S/(m ε_s^2)) ln(k/δ_s) rounds, and receive the corresponding winner feedback: σ_1, σ_2, ... σ_{t_s} ∈ Σ^m_B per round.
    Update pairwise empirical win-count w_ij using Rank-Breaking on σ_1 ... σ_{t_s}, ∀i, j ∈ A
    Update p̂_ij := w_ij/(w_ij + w_ji) for all i, j ∈ A
    If ∃i ∈ A with p̂_{ib_s} < 1/2 − ε_s, then A ← A \ {i}
    s ← s + 1
end while
output: The item remaining in A
```

Thm. 3, further taking into consideration the additional sample complexity incurred at each sub-phase s due to invoking the (ϵ,δ)-*PAC Best-Item* and *Score-Estimate* subroutine can shown to be at most $\frac{\Theta_{[k]}}{k}\sum_{i=2}^{n}\max\big(1, \frac{1}{\max(\epsilon^2,\Delta_i^2)}\big)\log\frac{k}{\delta}\big)$, combining which with the above sample complexity gives the desired sample complexity bound of Alg. 6.

□

A.7. Modified version of *PAC-Wrapper* (Alg. 1) for Top-m Ranking feedback

The pseudo code is provided in Alg. 7.

A.8. Proof of Thm. 5

Theorem 5 (*PAC-Wrapper*: Sample Complexity for $(0,\delta)$-PAC Guarantee for Top-m Ranking feedback)**.** *With probability at least $(1-\delta)$, PAC-Wrapper (Algorithm 1) returns the Best-Item with sample complexity* $N_{\mathcal{A}}(0,\delta) = O\left(\frac{\Theta_{[k]}}{k}\sum_{i=2}^{n}\max\left(1,\frac{1}{m\Delta_i^2}\right)\ln\frac{k}{\delta}\left(\ln\frac{1}{\Delta_i}\right)\right)$.

Proof. As argued, the main idea behind the $\frac{1}{m}$ factor improvement in the sample complexity w.r.t Winner feedback (as proved in Thm. 3), lies behind using *Rank-Breaking* updates (see Alg. 4) to the general Top-m Ranking feedback. This actually gives rise to $O(m)$ times additional number of pairwise preferences in comparison to Winner feedback which is why in this case it turns out to be sufficient to sample any batch $\mathcal{B}_b, \forall b \in [B_s]$ for only $O\big(\frac{1}{m}\big)$ times compared to the earlier case—precisely the reason behind $\frac{1}{m}$-factor improved sample complexity of *PAC-Wrapper* for Top-m Ranking feedback. The rest of the proof argument is mostly similar to that of Thm. 3. We provide the detailed analysis below for the sake of completeness.

We start by proving the correctness of the algorithm, i.e. with high probability $(1-\delta)$, *PAC-Wrapper* indeed returns the Best-Item, i.e. item 1 in our case. Towards this we first prove the following two lemmas: Lem. 23 and Lem. 24, same as what was derived for Thm. 3 as well—However it is important to note that its is due to the Top-m Ranking feedback feedback the exact same guarantees holds in this case as well, even with a m-times lesser observed samples.

Lemma 23. *With high probability of at least $(1-\frac{\delta}{20})$, item 1 is never eliminated, i.e. $1 \in \mathcal{A}_s$ for all sub-phase s. More formally, at the end of any sub-phase $s = 1, 2, \ldots$, $\hat p_{1b_s} > \frac{1}{2} - \epsilon_s$.*

Proof. Firstly note that at any sub-phase s, each batch $b \in B$ within that phase is played for $t_s = \frac{2\Theta_{[k]}}{m\epsilon_s^2}\ln\frac{k}{\delta_s}$ rounds. Now consider the batch $\mathcal{B} \ni 1$ at any phase s. Clearly $b_s \in \mathcal{B}$ too. Again since b_s is returned by Alg. 5, by Thm. 6 we know that with probability at least $(1-\delta_s)$, $p_{b_s 1} > \frac{1}{2} - \epsilon_s \implies \theta_{b_s} > \theta_1 - 4\epsilon$. This further implies $\theta_{b_s} \ge \theta_1 - \frac{1}{2} = \frac{1}{2}$ (since we assume $\theta_1 = 1$, and at any s, $\epsilon_s < \frac{1}{8}$). Moreover by Lem. 14, we have $\hat\Theta_S \ge \frac{\theta_{b_s}+\Theta_S}{\theta_{b_s}} > \frac{\Theta_S+1}{2}$ (recall we denote $S = \mathcal{B}_b \setminus \{b_s\}$)

Now let us define w_i as number of times item $i \in \mathcal{B}$ was returned as the winner in t_s rounds and i_τ be the winner retuned by the environment upon playing $\mathcal{B}$ for the τ^{th} round, where $\tau \in [t_s]$. Then clearly $Pr(\{1 \in \sigma_\tau\}) = \sum_{j=1}^m Pr\big(\sigma_\tau(j) = 1\big) \ge \frac{m\theta_1}{\sum_{j\in\mathcal{B}}\theta_j} = \frac{m}{\theta_{b_s}+\Theta_S} \ge \frac{m}{1+\Theta_S}, \forall \tau \in [t_s]$, as $1 := \arg\max_{i\in\mathcal{B}}\theta_i$. Hence $\mathbf{E}[w_1] = \sum_{\tau=1}^{t_s}\mathbf{E}[\mathbf{1}(i_\tau = 1)] = \frac{mt_s}{\theta_{b_s}+\Theta_S} \ge \frac{mt_s}{(1+\Theta_S)}$. Now assuming b_s to be indeed an (ϵ_s,δ_s)-PAC Best-Item and the bound of Lem. 14 to hold good as well, applying multiplicative Chernoff-Hoeffdings bound on the random variable w_1, we get that for any $\eta \in (\sqrt{2}\epsilon_s, 1]$,

$$
\begin{aligned}
Pr\Big(w_1 \le (1-\eta)\mathbf{E}[w_1]\Big) &\le \exp\bigg(-\frac{\mathbf{E}[w_1]\eta^2}{2}\bigg) \le \exp\bigg(-\frac{mt_s\eta^2}{2(1+\Theta_S)}\bigg)\\
&= \exp\bigg(-\frac{2\hat\Theta_S\eta^2}{2m\epsilon_s^2(1+\Theta_S)}\ln\frac{k}{\delta_s}\bigg)\\
&\overset{(a)}{\le} \exp\bigg(-\frac{2(\Theta_S+1)\eta^2}{4\epsilon_s^2(1+\Theta_S)}\ln\frac{k}{\delta_s}\bigg)\\
&\le \exp\bigg(-\frac{\eta^2}{2\epsilon_s^2}\ln\bigg(\frac{k}{\delta_s}\bigg)\bigg) \le \exp\bigg(-\ln\bigg(\frac{k}{\delta_s}\bigg)\bigg) = \frac{\delta_s}{k},
\end{aligned}
$$

where (a) holds since we proved $\hat\Theta_S \ge \frac{\theta_{b_s}+\Theta_S}{\theta_{b_s}} > \frac{\Theta_S+1}{2}$, and the last inequality holds as $\eta > \epsilon_s\sqrt{2}$.

In particular, note that $\epsilon_s \le \frac{1}{8}$ for any sub-phase s, due to which we can safely choose $\eta = \frac{1}{2}$ for any s, which gives that with probability at least $\big(1-\frac{\delta_s}{k}\big)$, $w_1 > (1-\frac{1}{2})\mathbf{E}[w_1] > \frac{mt_s}{2(\theta_{b_s}+\Theta_S)}$, for any subphase s.

Thus above implies that with probability atleast $(1-\frac{\delta_s}{k})$, after t_s rounds we have $w_{1b_s} \ge \frac{mt_s}{2(\theta_{b_s}+\Theta_S)} \implies w_{1b_s} + w_{b_s 1} \ge \frac{mt_s}{2(\theta_{b_s}+\Theta_S)}$. Let us denote $n_{1b_s} = w_{1b_s} + w_{b_s 1}$. Then the probability of the event:

$$
\begin{aligned}
Pr\Bigg(\hat p_{1b_s} < \frac{1}{2} - \epsilon_s, n_{1b_s} \ge \frac{mt_s}{2(\theta_{b_s}+\Theta_S)}\Bigg) &= Pr\Bigg(\hat p_{1b_s} - \frac{1}{2} < -\epsilon_s, n_{1b_s} \ge \frac{mt_s}{2(\theta_{b_s}+\Theta_S)}\Bigg)\\
&\le Pr\Bigg(\hat p_{1b_s} - p_{1b_s} < -\epsilon_s, n_{1b_s} \ge \frac{mt_s}{2(\theta_{b_s}+\Theta_S)}\Bigg) \quad (\text{as } \mathbf{p}_{1b_s} > \frac{1}{2})\\
&\overset{(a)}{\le} \exp\Big(-2\frac{mt_s}{2(\theta_{b_s}+\Theta_S)}\big(\epsilon_s\big)^2\Big)
\end{aligned}
$$

$$\le \exp\Big(-\frac{2m(\theta_{b_s}+\Theta_S)}{m\epsilon_s^2\theta_{b_s}(\theta_{b_s}+\Theta_S)}\big(\epsilon_s\big)^2\Big) \le \frac{\delta_s}{k},$$

where the last inequality (a) follows from Lem. 22 for $\eta = \epsilon_s$ and $v = \frac{t_s}{2k}$.

Thus under the two assumptions that (1). b_s is indeed an (ϵ_s, δ_s)-PAC Best-Item and (2). the bound of Lem. 14 holds good, combining the above two claims, at any sub-phase s, we have

$$\begin{aligned}
&Pr\Bigg(\hat p_{1b_s} < \frac{1}{2}-\epsilon_s\Bigg)\\
&= Pr\Bigg(\hat p_{1b_s} < \frac{1}{2}-\epsilon_s, n_{1b_s} \ge \frac{mt_s}{2(\theta_{b_s}+\Theta_S)}\Bigg) + Pr\Bigg(\hat p_{1b_s} < \frac{1}{2}-\epsilon_s, n_{1b_s} < \frac{mt_s}{2(\theta_{b_s}+\Theta_S)}\Bigg)\\
&\le \frac{\delta_s}{k} + Pr\Bigg(n_{1b_s} < \frac{mt_s}{2(\theta_{b_s}+\Theta_S)}\Bigg) \le \frac{2\delta_s}{k} \le \delta_s \quad (\text{since } k \ge 2)
\end{aligned}$$

Moreover from Thm. 6 and Lem. 14 we know that the above two assumptions hold with probability at least $(1-2\delta_s)$. Then taking union bound over all sub-phases $s = 1, 2, \ldots$, the probability that item 1 gets eliminated at any round:

$$Pr\Bigg(\exists s = 1, 2, \ldots \text{ s.t. } \hat p_{1b_s} < \frac{1}{2}-\epsilon_s\Bigg) = \sum_{s=1}^{\infty} 3\delta_s = \sum_{s=1}^{\infty}\frac{3\delta}{120s^3} \le \frac{\delta}{40}\sum_{s=1}^{\infty}\frac{1}{s^2} \le \frac{\delta}{40}\frac{\pi^2}{6} \le \frac{\delta}{20},$$

where the first inequality holds since $k \ge 2$. □

Recall the notations introduced in the proof of Thm. 3: $\Delta_i = \theta_1 - \theta_i$ (Sec. 2), $\Delta_{\min} = \min_{i\in[n]\setminus\{1\}}\Delta_i$. Further $[n]_r := \{i \in [n] : \frac{1}{2^r} \le \Delta_i < \frac{1}{2^{r-1}}\}$, and $\mathcal{A}_{r,s}$, i.e. $\mathcal{A}_{r,s} = [n]_r \cap \mathcal{A}_s$, for all $s = 1, 2, \ldots$. Then in this case again we claim:

Lemma 24. *Assuming that the best arm 1 is not eliminated at any sub-phase $s = 1, 2, \ldots$, then with probability at least $(1-\frac{19\delta}{20})$, for any sub-phase $s \ge r$, $|\mathcal{A}_{r,s}| \le \frac{2}{19}|\mathcal{A}_{r,s-1}|$, for any $r = 0, 1, 2, \ldots \log_2(\Delta_{\min})$.*

Proof. Consider any sub-phase s, and let us start by noting some properties of b_s. Note that by Lem. 6, with high probability $(1-\delta_s)$, $p_{b_s1} > \frac{1}{2}-\epsilon_s$. Then this further implies

$$\begin{aligned}
p_{b_s1} > \frac{1}{2}-\epsilon_s \implies & \frac{(\theta_{b_s}-\theta_1)}{2(\theta_{b_s}+\theta_1)} > -\epsilon_s\\
\implies & (\theta_{b_s}-\theta_1) > -2\epsilon_s(\theta_{b_s}+\theta_1) > -4\epsilon_s \ \ (\text{as } \theta_i \in (0,1)\, \forall i \in [n]\setminus\{1\})
\end{aligned}$$

So we have with probability atleast $(1-\delta_s)$, $\theta_{b_s} > \theta_1 - 4\epsilon_s = \theta_1 - \frac{1}{2^s}$.

Now consider any fixed $r = 0, 1, 2, \ldots \log_2(\Delta_{\min})$. Clearly by definition, for any item $i \in [n]_r$, $\Delta_i = \theta_1 - \theta_i > \frac{1}{2^r}$. Then combining the above two claims, we have for any sub-phase $s \ge r$, $\theta_{b_s} > \theta_1 - \frac{1}{2^s} \ge \theta_i + \frac{1}{2^r} - \frac{1}{2^s} > 0 \implies p_{b_si} > \frac{1}{2}$, at any $s \ge r$. Moreover note that for any $s \ge 1$, $\epsilon_s > \frac{1}{8}$, so that implies $\theta_{b_s} > \theta_1 - 4\epsilon_s > \frac{1}{2}$.

Recall that at any sub-phase s, each batch within that phase is played for $t_s = \frac{2\Theta_{[k]}}{m\epsilon_s^2}\ln\frac{k}{\delta_s}$ many rounds. Now consider any batch such that $\mathcal{B} \ni i$ for any $i \in [n]_r$. Of course $b_s \in \mathcal{B}$ as well, and note that we have shown $p_{b_si} > \frac{1}{2}$ with high probability $(1-\delta_s)$.

Same as Lem. 19, let us again define w_i as number of times item $i \in \mathcal{B}$ was returned as the winner in t_s rounds, and i_τ be the winner retuned by the environment upon playing $\mathcal{B}$ for the τ^{th} rounds, where $\tau \in [t_s]$. Then given $\theta_{b_s} > \frac{1}{2}$ (as derived earlier), clearly $Pr(\{b_s \in \sigma_\tau\}) = \sum_{j=1}^{m} Pr\Big(\sigma_\tau(j) = b_s\Big) \ge \sum_{j=0}^{m-1} \frac{\theta_{b_s}}{(\theta_{b_s}+\Theta_S)} = \frac{m\theta_{b_s}}{\theta_{b_s}+\Theta_S}, \forall \tau \in [t_s]$. Hence $\mathbf{E}[w_{b_s}] = \sum_{\tau=1}^{t_s} \mathbf{E}[\mathbf{1}(i_\tau = b_s)] = \frac{m\theta_{b_s} t_s}{(\theta_{b_s}+\Theta_S)}$. Now applying multiplicative Chernoff-Hoeffdings bound on the random variable w_{b_s}, we get that for any $\eta \in (\sqrt{2}\epsilon_s, 1]$,

$$
\begin{aligned}
&Pr\Big(w_{b_s} \le (1-\eta)\mathbf{E}[w_{b_s}] \mid \theta_{b_s} > \frac{1}{2}, \hat\Theta_S > \frac{(\theta_{b_s}+\Theta_S)}{\theta_{b_s}}\Big) \\
&\le \exp\bigg(-\frac{\mathbf{E}[w_{b_s}]\eta^2}{2}\bigg) \le \exp\bigg(-\frac{m\theta_{b_s} t_s \eta^2}{2(\theta_{b_s}+\Theta_S)}\bigg) \\
&\le \exp\bigg(-\frac{\eta^2}{2\epsilon_s^2}\ln\bigg(\frac{k}{\delta_s}\bigg)\bigg) \le \exp\bigg(-\ln\bigg(\frac{k}{\delta_s}\bigg)\bigg) = \frac{\delta_s}{k},
\end{aligned}
$$

where the last inequality holds as $\eta > \sqrt{2}\epsilon_s$. So as a whole, for any $\eta \in (\sqrt{2}\epsilon_s, 1]$,

$$
\begin{aligned}
&Pr\Big(w_{b_s} \le (1-\eta)\mathbf{E}[w_{b_s}]\Big) \\
&\le Pr\Big(w_{b_s} \le (1-\eta)\mathbf{E}[w_{b_s}] \mid \theta_{b_s} > \frac{1}{2}, \hat\Theta_S > \frac{(\theta_{b_s}+\Theta_S)}{\theta_{b_s}}\Big) Pr\Big(\theta_{b_s} > \frac{1}{2}\Big) + Pr\Big(\theta_{b_s} < \frac{1}{2}, \hat\Theta_S > \frac{(\theta_{b_s}+\Theta_S)}{\theta_{b_s}}\Big) \\
&\le \frac{\delta_s}{k} + 2\delta_s
\end{aligned}
$$

In particular, note that $\epsilon_s < \frac{1}{8}$ for any sub-phase s, due to which we can safely choose $\eta = \frac{1}{2}$ for any s, which gives that with probability at least $\big(1 - \frac{\delta_s}{k}\big)$, $w_{b_s} > (1-\frac{1}{2})\mathbf{E}[w_{b_s}] > \frac{mt_s\theta_{b_s}}{2(\Theta_S+\theta_{b_s})}$, for any subphase s.

Thus above implies that with probability at least $(1 - \frac{\delta_s}{k} - 2\delta_s)$, after t_s rounds we have $w_{b_s i} \ge \frac{mt_s\theta_{b_s}}{2(\Theta_S+\theta_{b_s})} \implies w_{ib_s} + w_{b_s i} \ge \frac{mt_s\theta_{b_s}}{2(\Theta_S+\theta_{b_s})}$. Let us denote $n_{ib_s} = w_{ib_s} + w_{b_s i}$. Then the probability that item i is not eliminated at any sub-phase $s \ge r$ is:

$$
\begin{aligned}
Pr\bigg(\hat p_{ib_s} > \frac{1}{2} - \epsilon_s, n_{ib_s} \ge \frac{mt_s\theta_{b_s}}{2(\Theta_S+\theta_{b_s})}\bigg) &= Pr\bigg(\hat p_{ib_s} - \frac{1}{2} > -\epsilon_s, n_{ib_s} \ge \frac{mt_s\theta_{b_s}}{2(\Theta_S+\theta_{b_s})}\bigg) \\
&\le Pr\bigg(\hat p_{ib_s} - p_{ib_s} > -\epsilon_s, n_{1b_s} \ge \frac{mt_s\theta_{b_s}}{2(\Theta_S+\theta_{b_s})}\bigg) \quad (\text{as } \mathbf{p}_{ib_s} < \frac{1}{2}) \\
&\overset{(a)}{\le} \exp\bigg(-2\frac{mt_s}{2(\Theta_S+\theta_{b_s})}\big(\epsilon_s\big)^2\bigg) \\
&\le \exp\bigg(-2\frac{m(\Theta_S+\theta_{b_s})}{2m\theta_{b_s}(\Theta_S+\theta_{b_s})}\big(\epsilon_s\big)^2 \ln\frac{\delta_s}{k}\bigg) \le \frac{\delta_s}{k},
\end{aligned}
$$

where (a) follows from Lem. 22 for $\eta = \epsilon_s$ and $v = \frac{t_s}{2k}$.

Now combining the above two claims, at any sub-phase s, we have:

$$
\begin{aligned}
Pr\bigg(\hat p_{ib_s} > \frac{1}{2} - \epsilon_s\bigg) &= Pr\bigg(\hat p_{ib_s} > \frac{1}{2} - \epsilon_s, n_{ib_s} \ge \frac{t_s}{4k}\bigg) + Pr\bigg(\hat p_{ib_s} > \frac{1}{2} - \epsilon_s, n_{ib_s} < \frac{t_s}{4k}\bigg) \\
&\le \frac{\delta_s}{k} + 2\delta_s + \frac{\delta_s}{k} \le 3\delta_s \quad (\text{since } k \ge 2)
\end{aligned}
$$

This consequently implies that for any sup-phase $s \ge r$, $\mathbf{E}[|\mathcal{A}_{r,s}|] \le 3\delta_s \mathbf{E}[|\mathcal{A}_{r,s-1}|]$. Then applying Markov's Inequality we get:

$$Pr\bigg(|\mathcal{A}_{r,s}| \le \frac{2}{19}|\mathcal{A}_{r,s-1}|\bigg) \le \frac{3\delta_s|\mathcal{A}_{r,s-1}|}{\frac{2}{19}|\mathcal{A}_{r,s-1}|} = \frac{57\delta_s}{2}$$

Finally applying union bound over all sub-phases $s = 1, 2, \ldots$, and all $r = 1, 2, \ldots s$, we get:

$$\sum_{s=1}^{\infty}\sum_{r=1}^{s}\frac{57\delta_s}{2} = \sum_{s=1}^{\infty} s\frac{57\delta}{240s^3} = \frac{57\delta}{240}\sum_{s=1}^{\infty}\frac{1}{s^2} \le \frac{57\delta}{240}\frac{\pi^2}{6} \le \frac{57\delta}{120} \le \frac{19\delta}{20}.$$

□

Thus combining Lem. 23 and 24, we get that the total failure probability of *PAC-Wrapper* is at most $\frac{\delta}{20} + \frac{19\delta}{20} = \delta$.

The remaining thing is to prove the sample complexity bound which crucially follows from a similar claim as proved in Lem. 21. As before, at any sub-phase s, we call the item b_s as the pivot element of phase s, then

Lemma 25. *Assume both Lem. 23 and Lem. 24 holds good and the algorithm does not do a mistake. Consider any item $i \in [n]_r$, for any $r = 1, 2, \ldots \log_2(\Delta_{\min})$. Then the total number of times item i gets played as a non-pivot item (i.e. appear in at most one of the k-subsets per sub-phase s) during the entire run of Alg. 7 is $O\Big(\frac{(2^r)^2\Theta_{[k]}}{m}\ln\frac{rk}{\delta}\Big)$.*

Proof. Let us denote the sample complexity of item i (as a non-pivot element) from phase x to y as $\mathcal{N}_{x,y}^{(i)}$, for any $1 \le x < y < \infty$. Additionally, recalling from Lem. 14 that $\hat{\Theta}_S \le 7\Theta_{[k]}$, we now prove the claim with the following two case analyses:

(Case 1) Sample complexity till sub-phase $s = r-1$: Note that in the worst case item i can get picked at every sub-phase $s = 1, 2, \ldots r-1$, and at every s it is played for t_s rounds. Thus the total number of plays of item $i \in [n]_r$ (as a non-pivot item), till sub-phase $r-1$ becomes:

$$\mathcal{N}_{1,r-1}^{(i)} \le \sum_{s=1}^{r-1} t_s = \sum_{s=1}^{r-1}\frac{14\Theta_{[k]}}{m\epsilon_s^2}\ln\frac{k}{\delta_s} = \frac{14\Theta_{[k]}}{m4^{-2}}\sum_{s=1}^{r-1}(2^s)^2\ln\frac{120ks^3}{\delta} = O\Big(\frac{(2^r)^2\Theta_{[k]}}{m}\ln\frac{rk}{\delta}\Big)$$

(Case 2) Sample complexity from sub-phase $s \ge r$ onwards: Assuming Lem. 24 holds good, note that if we define a random variable I_s for any sub-phase $s \ge r$ such that $I_s = \mathbf{1}(i \in \mathcal{A}_s)$, then clearly $\mathbf{E}[I_s] \le \frac{2}{19}\mathbf{E}[I_{s-1}]$ (as follows from the analysis of Lem. 23). Then the total expected sample complexity of item $i \in [n]_r$ for round $r, r+1, \ldots \infty$ becomes:

$$\begin{aligned}
\mathcal{N}_{r,\infty}^{(i)} &\le \frac{224\Theta_{[k]}}{m}\sum_{s=r}^{\infty}\bigg(\frac{2}{19}\bigg)^{s-r+1}4^s\ln\frac{k}{\delta_s} = \frac{224\Theta_{[k]}(2^r)^2}{m}\sum_{s=0}^{\infty}\bigg(\frac{2}{19}\bigg)^{s+1}(2^s)^2\ln\frac{120k(s+r)^3}{\delta} \\
&= \frac{448}{19m}\Theta_{[k]}(2^r)^2\sum_{s=0}^{\infty}\bigg(\frac{8}{19}\bigg)^{s}\ln\frac{120k(s+r)^3}{\delta} \\
&\le \frac{448}{19m}\Theta_{[k]}(2^r)^2\Bigg[\ln\frac{120kr}{\delta}\sum_{s=0}^{\infty}\bigg(\frac{8}{19}\bigg)^{s} + \sum_{s=0}^{\infty}\bigg(\frac{8}{19}\bigg)^{s}\ln(120ks)\Bigg] = O\Big(\frac{(2^r)^2\Theta_{[k]}}{m}\ln\frac{rk}{\delta}\Big)
\end{aligned}$$

Combining the two cases above we get $\mathcal{N}_{1,\infty}^{(i)} = O\Big(\frac{(2^r)^2\Theta_{[k]}}{m}\ln\frac{rk}{\delta}\Big)$ as well, which concludes the proof. □

Following similar notations as $\mathcal{N}^{(i)}_{x,y}$, we now denote the number of times any k-subset $S \subseteq [n]$ played by the algorithm in sub-phase x to y as $\mathcal{N}^{(S)}_{x,y}$. Then using Lem. 25, the total sample complexity of the algorithm *PAC-Wrapper* (lets call it algorithm $\mathcal{A}$) can be written as:

$$
\begin{aligned}
\mathcal{N}_{\mathcal{A}}(0,\delta) &= \sum_{S \subset [n] || S| = k} \sum_{s=1}^{\infty} \mathbf{1}(S \in \{\mathcal{B}_1, \ldots, \mathcal{B}_{B_s}\}) t_s = \sum_{i \in [n]} \sum_{s=1}^{\infty} \frac{\mathbf{1}(i \in \mathcal{A}_s \setminus \{b_s\})}{k-1} t_s \\
&= \sum_{s=1}^{\infty} \sum_{i \in [n]} \frac{\mathbf{1}(i \in \mathcal{A}_s \setminus \{b_s\})}{k-1} t_s = \sum_{s=1}^{\infty} \sum_{r=1}^{\log_2(\Delta_{\min})} \sum_{i \in [n]_r} \frac{\mathbf{1}(i \in \mathcal{A}_s \setminus \{b_s\})}{k-1} t_s \\
&= \frac{1}{k-1} \sum_{r=1}^{\log_2(\Delta_{\min})} \sum_{i \in [n]_r} \mathcal{N}^{(i)}_{1,\infty} \;\; (\text{ Lem. } 21) \\
&= \frac{1}{k-1} \sum_{r=1}^{\log_2(\Delta_{\min})} \sum_{i \in [n]_r} O\Big(\frac{(2^r)^2 \Theta_{[k]}}{m} \ln \frac{rk}{\delta} \Big) = \frac{\Theta_{[k]}}{k-1} \sum_{r=1}^{\log_2(\Delta_{\min})} |[n]_r| O\Big(\frac{(2^r)^2}{m} \ln \frac{rk}{\delta} \Big) \\
&= O\Big(\frac{\Theta_{[k]}}{k} \sum_{i=2}^{n} \frac{1}{m\Delta_i^2} \ln \big(\frac{k}{\delta} \ln \frac{1}{\Delta_i} \big) \Big),
\end{aligned} \tag{3}
$$

where the last inequality follows since $2^r < \frac{2}{\Delta_i}$ by definition for all $i \in [n]_r$. Finally, same as derived in the proof of Thm. 3, the last thing to account for is the additional sample complexity incurred due to calling the subroutine (ϵ,δ)-*PAC Best-Item* and *Score-Estimate* at every sub-phase s, which is combinedly known to be of $O\bigg(\frac{|\mathcal{A}_s|\Theta_[k]}{k} \Big(1, \frac{1}{(m2^s)^2}\Big) \ln \frac{k}{\delta} \bigg)$ at any sub-phase s (from Thm. 6 and Cor. 15). And using a similar summation as shown above over all $s = 1, 2, \ldots \infty$, combined with Lem. 24 and using the fact that $2^r < \frac{2}{\Delta_i}$, one can show that the total sample complexity incurred due to the above subroutines is at most $\frac{\Theta_{[k]}}{k} \sum_{i=2}^{n} \max\big(1, \frac{1}{m\Delta_i^2}\big) \log \frac{k}{\delta}\big)$. Considering the above sample complexity added with the one derived in Eqn. 3 finally gives the desired $O\bigg(\frac{\Theta_{[k]}}{k} \sum_{i=2}^{n} \max\Big(1, \frac{1}{m\Delta_i^2}\Big) \ln \frac{k}{\delta} \Big(\ln \frac{1}{\Delta_i} \Big) \bigg)$ sample complexity bound of Alg. 7.

$\square$

B. Appendix for Sec. 4

B.1. Proof of Thm. 7

Theorem 7 (Sample complexity lower bound: $(0,\delta)$-PAC or *Probably-Correct-Best-Item* with Winner feedback)**.** *Given $\delta \in [0,1]$, suppose $\mathcal{A}$ is an online learning algorithm for Winner feedback which, when run on any Plackett-Luce instance, terminates in finite time almost surely, returning an item I satisfying $Pr(\theta_I = \max_i \theta_i) > 1-\delta$. Then, on any Plackett-Luce instance $\theta_1 > \max_{i \geq 2} \theta_i$, the expected number of rounds it takes to terminate is $\Omega\bigg(\max\Big(\sum_{i=2}^{n} \frac{\theta_i \theta_1}{\Delta_i^2} \ln \frac{1}{\delta}, \frac{n}{k} \ln \frac{1}{\delta} \Big) \bigg)$.*

Proof. The argument is based on a change-of-measure argument (Lemma 1) of (Kaufmann et al., 2016), restated below for convenience:

Consider a multi-armed bandit (MAB) problem with n arms or actions $\mathcal{A} = [n]$. At round t, let A_t and Z_t denote the arm played and the observation (reward) received, respectively. Let $\mathcal{F}_t = \sigma(A_1, Z_1, \ldots, A_t, Z_t)$ be the sigma algebra generated by the trajectory of a sequential bandit algorithm upto round t.

Lemma 26 (Lemma 1, (Kaufmann et al., 2016))**.** *Let ν and ν' be two bandit models (assignments of reward distributions to arms), such that ν_i (resp. ν'_i) is the reward distribution of any arm $i \in \mathcal{A}$ under bandit model ν (resp. ν'), and such that for all such arms i, ν_i and ν'_i are mutually absolutely continuous. Then for any almost-surely finite stopping time τ with respect* *to* $(\mathcal{F}_t)_t$,

$$\sum_{i=1}^{n} \mathbf{E}_\nu[N_i(\tau)] KL(\nu_i, \nu_i') \geq \sup_{\mathcal{E} \in \mathcal{F}_\tau} kl(Pr_\nu(\mathcal{E}), Pr_{\nu'}(\mathcal{E})),$$

where $kl(x,y) := x\log(\frac{x}{y}) + (1-x)\log(\frac{1-x}{1-y})$ *is the binary relative entropy,* $N_i(\tau)$ *denotes the number of times arm* i *is played in* τ *rounds, and* $Pr_\nu(\mathcal{E})$ *and* $Pr_{\nu'}(\mathcal{E})$ *denote the probability of any event* $\mathcal{E} \in \mathcal{F}_\tau$ *under bandit models* ν *and* ν'*, respectively.*

The heart of the lower bound analysis stands on the ground on constructing PL$(n, \boldsymbol{\theta})$ instances, and slightly modified versions of it such that no $(0,\delta)$-PAC algorithm can correctly identify the Best-Item of both the instances without examining enough (precisely $\Omega\Big(\sum_{i=2}^{n} \frac{\theta_i\theta_1}{\Delta_i^2} \ln\big(\frac{1}{\delta}\big)\Big)$) many subsetwise samples per instance. We describe the our constructed problem instances below:

Consider an PL$(n, \boldsymbol{\theta})$ instance with the arm (item) set A containing all subsets of size k of $[n]$ defined as $A = \{S \subseteq [n] \mid |S| = [k]\}$. Let PL$(n, \boldsymbol{\theta}^1)$ be the true distribution associated to the bandit arms $[n]$, given by the score parameters $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_n)$, such that $\theta_1 > \theta_i$, $\forall i \in [n] \setminus \{1\}$. Thus we have

$$\textbf{True Instance: } \text{PL}(n, \boldsymbol{\theta}^1) : \theta_1^1 > \theta_2^1 \geq \ldots \geq \theta_n^1.$$

Clearly, the Best-Item of PL$(n, \boldsymbol{\theta}^1)$ is $a^* = 1$. Now for every suboptimal item $a \in [n] \setminus \{1\}$, consider the altered problem instance PL$(n, \boldsymbol{\theta}^a)$ such that:

$$\textbf{Instance a: } \text{PL}(n, \boldsymbol{\theta}^a) : \theta_a^a = \theta_1^1 + \epsilon;\ \theta_i^a = \theta_i^1,\ \ \forall i \in [n] \setminus \{a\}$$

for some $\epsilon > 0$. Clearly, the Best-Item of PL$(n, \boldsymbol{\theta}^a)$ is $a^* = a$. Note that, for problem instance PL$(n, \boldsymbol{\theta}^a)$ $a \in [n]$, the probability distribution associated to arm $S \in A$ is given by:

$$p_S^a \sim Categorical(p_1, p_2, \ldots, p_k), \text{ where } p_i = Pr(i|S) = \frac{\theta_i^a}{\sum_{j \in S} \theta_j^a},\ \ \forall i \in [k],\, \forall S \in A,\, \forall a \in [n],$$

recall the definition of $Pr(i|S)$ is as defined in Sec. 2. Now applying Lem. 26 we get:

$$\sum_{\{S \in A \mid a \in S\}} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(\tau_{\mathcal{A}})] KL(p_S^1, p_S^a) \geq kl(Pr_{\boldsymbol{\theta}^1}(\mathcal{E}), Pr_{\boldsymbol{\theta}^a}(\mathcal{E})), \tag{4}$$

where $\tau_{\mathcal{A}} := N_{\mathcal{A}}(0,\delta)$ denotes the sample complexity (number of rounds of subsetwise game played before stopping) of Alg. $\mathcal{A}$ and for any subset $S \in A$, $N_S(\tau)$ denotes the number of times S was played by $\mathcal{A}$ in $N_{\mathcal{A}}(0,\delta)$ rounds. The above result holds from the straightforward observation that for any arm $S \in \mathcal{A}$ with $a \notin S$, p_S^1 is same as p_S^a, hence $KL(p_S^1, p_S^a) = 0, \forall S \in A,\, a \notin S$.

For the notational convenience we will henceforth denote $S^a = \{S \in \mathcal{A} \mid a \in S\}$. Now let us analyse the right hand side of (4), for any set $S \in S^a$. We further denote $\Delta_a' = \Delta_a + \epsilon = (\theta_1 - \theta_a) + \epsilon$, and $\theta_S^a = \sum_{i \in S} \theta_i^a$ for any $a \in [n]$. Now using the following upper bound on $KL(\mathbf{p}, \mathbf{q}) \leq \sum_{x \in \mathcal{X}} \frac{p^2(x)}{q(x)} - 1$, $\mathbf{p}$ and $\mathbf{q}$ be two probability mass functions on the discrete random variable $\mathcal{X}$ (Popescu et al., 2016), we get:

$$\begin{aligned}
KL(p_S^1, p_S^a) &\leq \sum_{i \in S \setminus \{a\}} \left(\frac{\theta_i^1}{\theta_S^1}\right)^2 \left(\frac{\theta_S^a}{\theta_i^a}\right) + \left(\frac{\theta_a^1}{\theta_S^1}\right)^2 \left(\frac{\theta_S^a}{\theta_a^a}\right) - 1 \\
&= \sum_{i \in S \setminus \{a\}} \left(\frac{\theta_i^1}{\theta_S^1}\right)^2 \left(\frac{\theta_S^1 + \Delta_a'}{\theta_i^1}\right) + \left(\frac{\theta_a^1}{\theta_S^1}\right)^2 \left(\frac{\theta_S^1 + \Delta_a'}{\theta_a^1 + \Delta_a'}\right) - 1 \\
&= \left(\frac{\theta_S^1 + \Delta_a'}{(\theta_S^1)^2}\right) \left(\sum_{i \in S \setminus \{a\}} \theta_i^1 + \frac{(\theta_a^1)^2}{\theta_a^1 + \Delta_a'}\right) - 1
\end{aligned}$$

$$= \bigg(\frac{\theta_S^1+\Delta_a'}{(\theta_S^1)^2}\bigg)\bigg(\frac{\theta_a^1\theta_S^1+\Delta_a'(\theta_S^1-\theta_a^1)}{\theta_a^1+\Delta_a'}\bigg)-1 \quad \bigg[\text{replacing } \sum_{i\in S\setminus\{a\}}\theta_i^1=(\theta_S^1-\theta_a^1)\bigg]$$

$$= \frac{\Delta_a'^2(\theta_S^1-\theta_a^1)}{(\theta_S^1)^2(\theta_a^1+\Delta_a')} \le \frac{\Delta_a'^2}{\theta_S^1(\theta_a^1+\Delta_a')} = \frac{\Delta_a'^2}{\theta_S^1(\theta_1^1+\epsilon)} \tag{5}$$

Now, consider $\mathcal{E}_0 \in \mathcal{F}_\tau$ be an event such that the algorithm $\mathcal{A}$ returns the element $i=1$, and let us analyse the left hand side of (4) for $\mathcal{E}=\mathcal{E}_0$. Clearly, A being an $(0,\delta)$-PAC algorithm, we have $Pr_{\boldsymbol\theta^1}(\mathcal{E}_0)>1-\delta$, and $Pr_{\boldsymbol\theta^a}(\mathcal{E}_0)<\delta$, for any suboptimal arm $a\in[n]\setminus\{1\}$. Then we have:

$$kl(Pr_{\boldsymbol\theta^1}(\mathcal{E}_0), Pr_{\boldsymbol\theta^a}(\mathcal{E}_0)) \ge kl(1-\delta,\delta) \ge \ln\frac{1}{2.4\delta} \tag{6}$$

where the last inequality follows from (Kaufmann et al., 2016)(see Eqn. (3)). Now combining (4) and (6), for each problem instance PL$(n,\boldsymbol\theta^a)$, $a\in[n]\setminus\{1\}$, we get,

$$\sum_{S\in S^a}\mathbf{E}_{\boldsymbol\theta^1}[N_S(\tau_A)]KL(p_S^1,p_S^a) \ge \ln\frac{1}{2.4\delta}$$

Moreover, using (5), we further get:

$$\ln\frac{1}{2.4\delta} \le \sum_{S\in S^a}\mathbf{E}_{\boldsymbol\theta^1}[N_S(\tau_A)]KL(p_S^1,p_S^a) \le \sum_{S\in S^a}\mathbf{E}_{\boldsymbol\theta^1}[N_S(\tau_A)]\frac{\Delta_a'^2}{\theta_S^1(\theta_1^1+\epsilon)} \tag{7}$$

Clearly, the total sample complexity of $\mathcal{A}$: $\tau_A=\sum_{S\in A}N_S(\tau_{\mathcal{A}})$, then note that the problem of finding the sample complexity lower bound problem actually reduces down to

$$\textbf{Primal LP (P):} \qquad \min_{S\in A}\sum_{S\in A}\mathbf{E}_{\boldsymbol\theta^1}[N_S(\tau_{\mathcal{A}})]$$

$$\text{such that,}\quad \ln\frac{1}{2.4\delta} \le \sum_{S\in S^a}\mathbf{E}_{\boldsymbol\theta^1}[N_S(\tau_A)]\frac{\Delta_a'^2}{\theta_S^1(\theta_1^1+\epsilon)},\ \forall a\in[n]\setminus\{1\},$$

which can equivalently be written as a linear programming (LP) of the following form:

$$\textbf{Dual LP (D):} \qquad \min_y \mathbf{b}^\top\mathbf{y}$$

$$\text{such that, } \mathbf{K}^\top\mathbf{y}\ge\mathbf{z},\ \text{and } \mathbf{y}\ge 0,$$

where $\mathbf{y}\in\mathbb{R}^M$, $M=|A|=\binom{n}{k}$, with $y(S)=\mathbf{E}_{\boldsymbol\theta^1}[N_S(\tau_A)],\ \forall S\in A$, $\mathbf{z}\in\mathbb{R}^{n-1}$ with $z(i)=\ln\frac{1}{2.4\delta}\ \forall i\in[n-1]$, $\mathbf{K}\in\mathbb{R}^{M\times(n-1)}$ such that $K(S,a)=\begin{cases}\frac{\Delta_a'^2}{\theta_S^1(\theta_1^1+\epsilon)}, & \text{if } S\in S^a\\ 0, & \text{otherwise}\end{cases}$, and $\mathbf{b}\in\mathbb{R}^{M\times 1}$ such that $b(i)=1\ \forall i\in[M]$.

Then taking the dual of the above LP (see Chapter 5, (Boyd & Vandenberghe, 2004)) we get:

$$\max_{\mathbf{x}}\mathbf{z}^\top\mathbf{x},\quad \text{such that, } \mathbf{K}\mathbf{x}\le\mathbf{b},\ \text{and } \mathbf{x}\ge 0,$$

where clearly $\mathbf{x} \in \mathbb{R}^{n-1}$ is the dual optimization variable.

Now we know that by *strong duality* if $\mathbf{y}^*$ and $\mathbf{x}^*$ respectively denotes the optimal solution of **(P)** and **(D)**, then $\mathbf{b}^\top \mathbf{y}^* = \mathbf{z}^\top \mathbf{x}^*$. Thus at any feasible solution $\mathbf{x}'$ of **(D)**, $\mathbf{z}^\top \mathbf{x}' \le \mathbf{z}^\top \mathbf{x}^* = \mathbf{b}^\top \mathbf{y}^*$.

Claim. $x'_i = \frac{\theta^1_{i+1}(\theta^1_1+\epsilon)}{{\Delta'_a}^2}$ for all $i \in [n-1]$ is a feasible solution of **(D)**.

Proof. Clearly, $x'_i \ge 0 \; \forall i \in [n-1]$ which ensures that the second set of constraints of **(D)** hold good. Expanding the first set of constraints $\mathbf{K}\mathbf{x}' \le \mathbf{b}$ we get M constraints, one for each $S \in A$ such that

$$\begin{aligned}\sum_{i=1}^{n-1} K(S,i)x'_i &= \sum_{i=1}^{n-1} \mathbf{1}(S \in S^{i+1}) K(S,i) \frac{\theta^1_{i+1}(\theta^1_1+\epsilon)}{{\Delta'_a}^2} \\ &= \sum_{i=2}^{n} \mathbf{1}(i \in S) \frac{\Delta'^2_a}{\theta^1_S(\theta^1_1+\epsilon)} \frac{\theta^1_i(\theta^1_1+\epsilon)}{{\Delta'_a}^2} \begin{cases} = 1 \text{ if } 1 \notin S \\ \le 1 \text{ otherwise} \end{cases} .\end{aligned}$$

The claim now follows recalling that $b(i) = 1 \; \forall i \in [M]$. $\square$

Thus we get $\ln\left(\frac{1}{\delta}\right) \sum_{i=2}^{n} \frac{\theta_i \theta_1}{{\Delta'_i}^2} = \mathbf{z}^\top \mathbf{x}' \le \mathbf{z}^\top \mathbf{x}^* = \mathbf{b}^\top \mathbf{y}^* = \sum_{S \in A} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(\tau_{\mathcal{A}})]$. Moreover since $\epsilon > 0$ is a construction dependent parameter, taking $\epsilon \to 0$ the expected sample complexity of $\mathcal{A}$ under PL$(n, \boldsymbol{\theta}^1)$ becomes:

$$\mathbf{E}_{\boldsymbol{\theta}^1}\Big[N_{\mathcal{A}}(0,\delta)\Big] = \sum_{S \in A} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(\tau_{\mathcal{A}})] \ge \sum_{i=2}^{n} \frac{\theta_i \theta_1}{\Delta_i^2} \ln \frac{1}{\delta}$$

Now taking $\epsilon \to 0$, the above construction shows that for any general problem instance, precisely PL$(n, \boldsymbol{\theta}^1)$, it requires a sample complexity of $\Omega\left(\sum_{a=2}^{n} \frac{\theta_1 \theta_a}{\Delta_a^2} \ln \frac{1}{\delta} \right)$ on expectation, to find the Best-Item (i.e. to achieve $(0,\delta)$-PAC objective). Finally to get the additional $\frac{n}{k} \log \frac{1}{\delta}$ term we appeal to the lower bound argument provided in (Chen et al., 2018) (see their Thm. $B.9$) for the $\left(0, \frac{1}{8}\right)$-PAC best-arm identification problem. For such 'low confidence' regimes, i.e., when $\delta = \Omega(1) \nrightarrow 0$, these explicitly shows a simple $\frac{n}{k} \log \frac{1}{\delta}$ term (independent of the instance) lower bound, which slightly improves the bound of Thm. 7 for instances when $\theta_i \to 0$ (or $\Delta_i \to 1$) for all suboptimal item $i \in [n] \setminus \{1\}$—note that a term like $\frac{n}{k} \log \frac{1}{\delta}$ is also intuitive, as for any Plackett-Luce instance, the learner needs to query at the least $\Omega\left(\frac{n}{k} \ln \frac{1}{\delta} \right)$ many samples to make sure it covers the entire set of n items. $\square$

B.2. Proof of Thm. 8

Theorem 8 (Sample complexity Lower Bound: $(0,\delta)$-*Probably-Correct-Best-Item* with Top-m Ranking feedback)**.** *Suppose $\mathcal{A}$ is an online learning algorithm for Top-m Ranking feedback which, given $\delta \in [0,1]$ and run on any Plackett-Luce instance, terminates in finite time almost surely, returning an item I satisfying $Pr(\theta_I = \max_i \theta_i) > 1-\delta$. Then, on any Plackett-Luce instance $\theta_1 > \max_{i \ge 2} \theta_i$, the expected number of rounds it takes to terminate is* $\Omega\left(\max\left(\frac{1}{m} \sum_{i=2}^{n} \frac{\theta_i \theta_1}{\Delta_i^2} \ln\left(\frac{1}{\delta}\right), \frac{n}{k} \ln \frac{1}{\delta} \right) \right)$.

Proof. The proof proceeds almost same as the proof of Thm. 7, the only difference lies in the analysis of the KL-divergence terms with Top-m Ranking feedback.

Consider the exact same set of PL instances, PL$(n, \boldsymbol{\theta}^a)$ we constructed for Thm. 7. It is now interesting to note that how Top-m Ranking feedback affects the KL-divergence analysis, precisely the KL-divergence shoots up by a factor of m which in fact triggers an $\frac{1}{m}$ reduction in regret learning rate. Note that for Top-m Ranking feedback for any problem instance PL$(n, \boldsymbol{\theta}^a)$, $a \in [n]$, each k-set $S \subseteq [n]$ is associated to $\binom{k}{m}(m!)$ number of possible outcomes, each representing one possible ranking of set of m items of S, say S_m. Also the probability of any permutation $\boldsymbol{\sigma} \in \boldsymbol{\Sigma}_S^m$ is given by $p_S^a(\boldsymbol{\sigma}) = Pr_{\boldsymbol{\theta}^a}(\boldsymbol{\sigma}|S)$, where $Pr_{\boldsymbol{\theta}^a}(\boldsymbol{\sigma}|S)$ is as defined for Top-m Ranking feedback (in Sec. 2). More formally, for problem **Instance-a**, we have that:

$$p_S^a(\boldsymbol{\sigma}) = \prod_{i=1}^{m} \frac{\theta^a_{\sigma(i)}}{\sum_{j=i}^{m} \theta^a_{\sigma(j)} + \sum_{j \in S \setminus \sigma(1:m)} \theta^a_{\sigma(j)}}, \quad \forall a \in [n],$$

The important thing now to note is that for any such top-m ranking of $\boldsymbol{\sigma} \in \boldsymbol{\Sigma}_S^m$, $KL(p_S^1(\boldsymbol{\sigma}), p_S^a(\boldsymbol{\sigma})) = 0$ for any set $S \not\ni a$. Hence while comparing the KL-divergence of instances $\boldsymbol{\theta}^1$ vs $\boldsymbol{\theta}^a$, we need to focus only on sets containing a. Applying *Chain-Rule* of KL-divergence, we now get

$$\begin{aligned} KL(p_S^1, p_S^a) = KL(p_S^1(\sigma_1), &p_S^a(\sigma_1)) + KL(p_S^1(\sigma_2 \mid \sigma_1), p_S^a(\sigma_2 \mid \sigma_1)) + \cdots \\ &+ KL(p_S^1(\sigma_m \mid \sigma(1:m-1)), p_S^a(\sigma_m \mid \sigma(1:m-1))), \end{aligned} \tag{8}$$

where we abbreviate $\sigma(i)$ as σ_i and $KL(P(Y \mid X), Q(Y \mid X)) := \sum_x Pr\Big(X = x\Big)\big[KL(P(Y \mid X = x), Q(Y \mid X = x))\big]$ denotes the conditional KL-divergence. Moreover it is easy to note that for any $\sigma \in \Sigma_S^m$ such that $\sigma(i) = a$, we have $KL(p_S^1(\sigma_{i+1} \mid \sigma(1:i)), p_S^a(\sigma_{i+1} \mid \sigma(1:i))) := 0$, for all $i \in [m]$.

Now as derived in (5) in the proof of Thm. 7, we have

$$KL(p_S^1(\sigma_1), p_S^a(\sigma_1)) \le \frac{\Delta_a'^2}{\theta_S^1(\theta_1^1 + \epsilon)}.$$

To bound the remaining terms of (8), note that for all $i \in [m-1]$

$$\begin{aligned} &KL(p_S^1(\sigma_{i+1} \mid \sigma(1:i)), p_S^a(\sigma_{i+1} \mid \sigma(1:i))) \\ &\quad = \sum_{\sigma' \in \Sigma_S^i} Pr(\sigma') KL(p_S^1(\sigma_{i+1} \mid \sigma(1:i)) = \sigma', p_S^a(\sigma_{i+1} \mid \sigma(1:i)) = \sigma') \\ &\quad = \sum_{\sigma' \in \Sigma_S^i \mid a \notin \sigma'} \left[\prod_{j=1}^{i} \left(\frac{\theta^1_{\sigma'_j}}{\theta_S^1 - \sum_{j'=1}^{j-1} \theta_{\sigma'_{j'}}} \right) \right] \frac{\Delta_a'^2}{(\theta_S^1 - \sum_{l=1}^{i} \theta^1_{\sigma_l})(\theta_1^1 + \epsilon)} = \frac{\Delta_a'^2}{\theta_S^1(\theta_1^1 + \epsilon)} \end{aligned}$$

Thus applying above in (8) we get:

$$\begin{aligned} KL(p_S^1, p_S^a) &= KL(p_S^1(\sigma_1) + \cdots + KL(p_S^1(\sigma_m \mid \sigma(1:m-1)), p_S^a(\sigma_m \mid \sigma(1:m-1))) \\ &\le \frac{m\Delta_a'^2}{\theta_S^1(\theta_1^1 + \epsilon)}. \end{aligned} \tag{9}$$

Eqn. (9) gives the main result to derive Thm. 8 as it shows an m-factor blow up in the KL-divergence terms owning to Top-m Ranking feedback. The rest of the proof follows exactly the same argument used in 7. We add the steps below for convenience.

Same as before, consider $\mathcal{E}_0 \in \mathcal{F}_\tau$ be an event such that the algorithm $\mathcal{A}$ returns the element $i = 1$, and combining (4) and (6), for each problem instance PL$(n, \boldsymbol{\theta}^a)$, $a \in [n] \setminus \{1\}$, we get,

$$\sum_{S \in S^a} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(\tau_A)] KL(p_S^1, p_S^a) \ge \ln \frac{1}{2.4\delta}$$

Now using (9), we further get:

$$\ln \frac{1}{2.4\delta} \le \sum_{S \in S^a} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(\tau_A)] KL(p_S^1, p_S^a) \le \sum_{S \in S^a} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(\tau_A)] \frac{m\Delta_a'^2}{\theta_S^1(\theta_1^1+\epsilon)} \tag{10}$$

Again consider the primal problem towards finding the sample complexity lower bound:

$$\textbf{Primal LP (P):} \qquad \min_{S \in A} \sum_{S \in A} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(\tau_{\mathcal{A}})]$$
$$\text{such that,} \quad \ln \frac{1}{2.4\delta} \le \sum_{S \in S^a} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(\tau_A)] \frac{m\Delta_a'^2}{\theta_S^1(\theta_1^1+\epsilon)}, \ \forall a \in [n] \setminus \{1\},$$

which can equivalently be written as a linear programming (LP) of the following form:

$$\textbf{Dual LP (D):} \qquad \min_{y} \mathbf{b}^\top \mathbf{y}$$
$$\text{such that, } \mathbf{K}^\top \mathbf{y} \ge \mathbf{z}, \text{ and } \mathbf{y} \ge 0,$$

where $\mathbf{y} \in \mathbb{R}^M$, $M = |A| = \binom{n}{k}$, with $y(S) = \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(\tau_A)]$, $\forall S \in A$, $\mathbf{z} \in \mathbb{R}^{n-1}$ with $z(i) = \ln \frac{1}{2.4\delta}\ \forall i \in [n-1]$, $\mathbf{K} \in \mathbb{R}^{M \times (n-1)}$ such that $K(S,a) = \begin{cases} \frac{m\Delta_a'^2}{\theta_S^1(\theta_1^1+\epsilon)}, \text{ if } S \in S^a \\ 0, \text{ otherwise} \end{cases}$, and $\mathbf{b} \in \mathbb{R}^{M \times 1}$ such that $b(i) = 1\ \forall i \in [M]$.

The dual of the above LP boils down to:

$$\max_{\mathbf{x}} \mathbf{z}^\top \mathbf{x}$$
$$\text{such that, } \mathbf{K}\mathbf{x} \le \mathbf{b}, \text{ and } \mathbf{x} \ge 0,$$

where clearly $\mathbf{x} \in \mathbb{R}^{n-1}$ is the dual optimization variable.

Claim. $x_i' = \frac{\theta_{i+1}^1(\theta_1^1+\epsilon)}{m{\Delta_a'}^2}$ for all $i \in [n-1]$ is a feasible solution of **(D)**.

Proof. Clearly, $x_i' \ge 0\ \forall i \in [n-1]$ which ensures that the second set of constraints of **(D)** hold good. Expanding the first set of constraints $\mathbf{K}\mathbf{x}' \le \mathbf{b}$ we get M constraints, one for each $S \in A$ such that

$$\begin{aligned}\sum_{i=1}^{n-1} K(S,i)x_i' &= \sum_{i=1}^{n-1} \mathbf{1}(S \in S^{i+1}) K(S,i) \frac{\theta_{i+1}^1(\theta_1^1+\epsilon)}{m{\Delta_a'}^2} \\ &= \sum_{i=2}^{n} \mathbf{1}(i \in S) \frac{m\Delta_a'^2}{\theta_S^1(\theta_1^1+\epsilon)} \frac{\theta_i^1(\theta_1^1+\epsilon)}{m{\Delta_a'}^2} \begin{cases} = 1 \text{ if } 1 \notin S \\ \le 1 \text{ otherwise} \end{cases}.\end{aligned}$$

The claim now follows recalling that $b(i) = 1\ \forall i \in [M]$. □

Thus we get $\ln\left(\frac{1}{\delta}\right) \sum_{i=2}^{n} \frac{\theta_i\theta_1}{m{\Delta_i'}^2} = \mathbf{z}^\top\mathbf{x}' \le \mathbf{z}^\top\mathbf{x}^* = \mathbf{b}^\top\mathbf{y}^* = \sum_{S \in A} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(\tau_{\mathcal{A}})]$. Moreover since $\epsilon > 0$ is a construction dependent parameter, taking $\epsilon \to 0$ the expected sample complexity of $\mathcal{A}$ under PL$(n, \boldsymbol{\theta}^1)$ becomes:

$$\mathbf{E}_{\boldsymbol{\theta}^1}\Big[N_{\mathcal{A}}(0,\delta)\Big] = \sum_{S \in A} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(\tau_{\mathcal{A}})] \ge \sum_{i=2}^{n} \frac{\theta_i\theta_1}{m\Delta_i^2} \ln \frac{1}{\delta}$$

Now taking $\epsilon \to 0$, the above construction shows that for any general problem instance, precisely PL$(n, \boldsymbol{\theta}^1)$, it requires a sample complexity of $\Omega\bigg(\sum_{a=2}^{n} \frac{\theta_1\theta_a}{m\Delta_a^2} \ln\frac{1}{\delta} \bigg)$ on expectation, to find the Best-Item (i.e. to achieve $(0,\delta)$-PAC objective) with Top-m Ranking feedback. Finally, to prove the additional instance independent $\Omega\Big(\frac{n}{k}\log\frac{1}{\delta}\Big)$ term, we can use a similar argument provided in the Thm. 7, which ensures that no matter what the underlying Plackett-Luce instance is, the learner needs to query at the least $\Omega\Big(\frac{n}{k}\ln\frac{1}{\delta}\Big)$ queries to cover the entire set of n items–note that this term is independent of m. $\square$

C. Appendix for Sec. 5

C.1. Proof of Thm. 11

Theorem 11 (Confidence lower bound in fixed sample complexity Q for Top-m Ranking feedback)**.** *Let $\mathcal{A}$ be a Budget-Consistent and Order-Oblivious algorithm for identifying the Best-Item under Top-m Ranking feedback. For any Plackett-Luce instance $\boldsymbol{\theta}$ and sample size (budget) Q, its probability of error in identifying the best arm in $\boldsymbol{\theta}$ satisfies* $Pr_{\boldsymbol{\theta}}\left(I \neq \arg\max_{i\in[n]}\theta_i\right) = \Omega\left(\exp\left(-2mQ\tilde{\Delta}\right)\right)$, *where the complexity parameter* $\tilde{\Delta} := \left(\sum_{a=2}^{n}\frac{(\theta_a)^2}{\Delta_a^2}\right)^{-1}$.

Proof. Similar to our lower bounds proofs for *Probably-Correct-Best-Item* setting (see Thm. 7, 8), we again use a *change-of-measure argument* to prove the instance-dependent lower bounds for the *Fixed-Sample-Complexity* setting.

We start by constructing the problem instances as follows: Consider a general the true underlying PL$(n,\boldsymbol{\theta})$ problem instance PL$(n,\boldsymbol{\theta}^1)$: $\theta_1^1 > \theta_2^1 \geq \ldots \geq \theta_n^1$, and corresponding to each suboptimal item $a \in [n]\setminus\{1\}$, let us define an alternative problem instance PL$(n,\boldsymbol{\theta}^a)$: $\theta_a^a = \theta_1^1;\ \theta_1^a = \theta_a^1;\ \theta_i^a = \theta_i^1,\ \ \forall i \in [n]\setminus\{a,1\}$, for some $\epsilon > 0$.

Then using a similar derivation shown for Eqn. (9), for above construction of problem instances in this case we can can show that:

$$KL(p_S^1, p_S^a) \leq \frac{m\Delta_a^2}{\theta_S^1}\left(\frac{\theta_1^1\mathbf{1}(1\in S) + \theta_a^1\mathbf{1}(a\in S)}{\theta_1^1\theta_a^1}\right) \tag{11}$$

where recall that we denote $\Delta_a = \theta_1^1 - \theta_a^1$, for any sub-optimal arm $a \in [n]\setminus\{a\}$. Clearly for any subset $S \subset [n]$ such that $\{1,a\}\cap S = \emptyset$ must lead to $KL(p_S^1,p_S^a) = 0$ which is also follows from (11).

Same as the proof of Thm. 8, now applying Lem. 26 for any event $\mathcal{E} \in \mathcal{F}_\tau$ we get:

$$\sum_{\{S\subseteq[n], |S|=k \mid a\in S\}} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(Q)]KL(p_S^1,p_S^a) \geq kl(Pr_{\boldsymbol{\theta}^1}(\mathcal{E}), Pr_{\boldsymbol{\theta}^a}(\mathcal{E})),$$

where for any k-subset S, $N_S(Q)$ denotes the total number of times S was played (i.e. queried upon for the Top-m Ranking feedback) by $\mathcal{A}$ in Q samples. Now, consider $\mathcal{E}_0 \in \mathcal{F}_\tau$ be an event such that the algorithm $\mathcal{A}$ indeed outputs the Best-Item 1 upon termination, and let us analyse the left hand side of (4) for $\mathcal{E} = \mathcal{E}_0$. Now $\mathcal{A}$ being *Budget-Consistent* algorithm (see Defn. 9), we have $Pr_{\boldsymbol{\theta}^1}(\mathcal{E}_0) > 1 - \exp(-f(\boldsymbol{\theta})Q)$. Moreover, since $\mathcal{A}$ is *Order-Oblivious* as well, we also have $Pr_{\boldsymbol{\theta}^a}(\mathcal{E}_0) < \exp(-f(\boldsymbol{\theta})Q)$, for any suboptimal arm $a \in [n]\setminus\{1\}$. Combining above two claims and denoting $\delta = \exp(-f(\boldsymbol{\theta})Q)$, we get:

$$kl(Pr_{\boldsymbol{\theta}^1}(\mathcal{E}_0), Pr_{\boldsymbol{\theta}^a}(\mathcal{E}_0)) \geq kl(1-\delta,\delta) \geq \ln\frac{1}{2.4\delta}$$

where the last inequality follows from (Kaufmann et al., 2016) (see Eqn. (3)). Then combining the above two claims with (11), for any problem instance PL$(n,\boldsymbol{\theta}^a)$, $a\in[n]\setminus\{1\}$, we get,

$$\ln\frac{1}{2.4\delta} \leq \sum_{S\in\mathcal{S}} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(Q)]KL(p_S^1,p_S^a) \leq \sum_{S\in\mathcal{S}} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(Q)]\frac{m\Delta_a^2}{\theta_S^1}\left(\frac{\theta_1^1\mathbf{1}(1\in S)+\theta_a^1\mathbf{1}(a\in S)}{\theta_1^1\theta_a^1}\right), \tag{12}$$

where we denote the set of all possible k-subsets of $[n]$ by $\mathcal{S} = \{S \subseteq [n] \mid |S| = k\}$.

Now coming back to our actual problem objective, recall that our goal is to understand the best possible lower bound on the quantity δ—since the left hand side above is a decreasing function of δ, at best any algorithm can aim to minimize δ as much as possible without violating the right hand side constraints for any $a \in [n] \setminus \{1\}$. In other words any algorithm can at best aim to achieve a error confidence δ such that $\ln\left(\frac{1}{2.4\delta}\right)$ is upper bounded by:

Max-Min Optimization (P):

$$\max_{\{N_S(Q)\}_{S\in\mathcal{S}}} \min_{a=2}^{n} \sum_{S\in\mathcal{S}} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(Q)] \frac{m\Delta_a^2}{\theta_S^1}\left(\frac{\theta_1^1\mathbf{1}(1\in S) + \theta_a^1\mathbf{1}(a\in S)}{\theta_1^1\theta_a^1}\right)$$
$$\text{such that, } \sum_{S\in\mathcal{S}} \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(Q)] = Q, \text{ and } \mathbf{E}_{\boldsymbol{\theta}^1}[N_S(Q)] \ge 0, \; \forall S \in \mathcal{S}$$

Clearly the optimization variables in **(P)** are $\{\mathbf{E}_{\boldsymbol{\theta}^1}[N_S(Q)]\}_{S\in\mathcal{S}}$. We denote the simplex on $\mathcal{S}$ by $\Pi_{\mathcal{S}} = \{\boldsymbol{\pi} \in [0,1]^{\binom{n}{k}} \mid \sum_i \pi(i) = 1\}$. In general, we denote any d-dimensional simplex by Π_d, for any $d \in \mathbb{N}$. Then it is easy to follow that the above optimization problem **(P)** can be equivalently written in terms of optimization variables $x_S := \frac{\mathbf{E}_{\boldsymbol{\theta}^1}[N_S(Q)]}{Q}$ as:

Equivalent Max-Min Optimization (P'):

$$Q\left[\max_{\{x_S\}_{S\in\mathcal{S}}} \min_{\boldsymbol{\lambda}\in\Pi_{n-1}} \sum_{a=2}^{n} \lambda(a)\left(\sum_{S\in\mathcal{S}} x_S \frac{m\Delta_a^2}{\theta_S^1}\left(\frac{\theta_1^1\mathbf{1}(1\in S) + \theta_a^1\mathbf{1}(a\in S)}{\theta_1^1\theta_a^1}\right)\right)\right]$$
$$\text{such that, } \sum_{S\in\mathcal{S}} x_S = 1, \text{ and } x_s \ge 0, \; \forall S \in \mathcal{S}$$

We denote by opt(**P**) and opt(**P'**) the optimal values of problem **P** and **P'** respectively. Note that opt(**P**) = opt(**P'**). Also note that $(x_S)_{S\in\mathcal{S}} \in \Pi_{\mathcal{S}}$. Then, opt(**P'**) can be further rewritten as:

$$\begin{aligned}
\frac{\text{opt}(\mathbf{P'})}{Q} &= \max_{\{x_S\}_{S\in\mathcal{S}}} \min_{\boldsymbol{\lambda}\in\Pi_{n-1}} \sum_{a=2}^{n} \lambda(a)\left(\sum_{S\in\mathcal{S}} x_S \frac{m\Delta_a^2}{\theta_S^1}\left(\frac{\theta_1^1\mathbf{1}(1\in S) + \theta_a^1\mathbf{1}(a\in S)}{\theta_1^1\theta_a^1}\right)\right)\\
&= \min_{\boldsymbol{\lambda}\in\Pi_{n-1}} \max_{\{x_S\}_{S\in\mathcal{S}}} \sum_{S\in\mathcal{S}} \sum_{a=2}^{n} \lambda(a) x_S \frac{m\Delta_a^2}{\theta_S^1}\left(\frac{\theta_1^1\mathbf{1}(1\in S) + \theta_a^1\mathbf{1}(a\in S)}{\theta_1^1\theta_a^1}\right)\\
&= \min_{\boldsymbol{\lambda}\in\Pi_{n-1}} \max_{S\in\mathcal{S}} \sum_{a=2}^{n} \lambda(a) \frac{m\Delta_a^2}{\theta_S^1}\left(\frac{\theta_1^1\mathbf{1}(1\in S) + \theta_a^1\mathbf{1}(a\in S)}{\theta_1^1\theta_a^1}\right)
\end{aligned}$$

where the second equality follows from Von Neumann's well-known Minmax Theorem (Freund & Schapire, 1996). Now further setting $\lambda'(a) = \dfrac{(\theta_a^1)^2/\Delta_a^2}{\sum_{i=2}^n (\theta_i^1)^2/\Delta_i^2}$, for all $a \in [n] \setminus \{1\}$ (note that $\boldsymbol{\lambda}' \in \Pi_{n-1}$), using $\boldsymbol{\lambda} = \boldsymbol{\lambda}'$ in opt(**P'**), it can further be upper bounded as:

$$\begin{aligned}
\frac{\text{opt}(\mathbf{P'})}{Q} &\le \max_{S\in\mathcal{S}} \sum_{a=2}^{n} \lambda'(a) \frac{m\Delta_a^2}{\theta_S^1}\left(\frac{\theta_1^1\mathbf{1}(1\in S) + \theta_a^1\mathbf{1}(a\in S)}{\theta_1^1\theta_a^1}\right)\\
&= \max_{S\in\mathcal{S}} \sum_{a=2}^{n} \frac{(\theta_a^1)^2}{\sum_{i=2}^n (\theta_i^1)^2/\Delta_i^2} \frac{m}{\theta_S^1}\left(\frac{\theta_1^1\mathbf{1}(1\in S) + \theta_a^1\mathbf{1}(a\in S)}{\theta_1^1\theta_a^1}\right)
\end{aligned}$$

$$
= \begin{cases} = \max\limits_{S \in \mathcal{S}} \sum_{a=2}^{n} \dfrac{(\theta_a^1)^2}{\sum_{i=2}^{n}(\theta_i^1)^2/\Delta_i^2} \dfrac{m}{\theta_S^1}\Bigg(\dfrac{1}{\theta_1^1}\Bigg) \le m\Bigg(\sum_{a=2}^{n}\frac{(\theta_a^1)^2}{\Delta_a^2}\Bigg)^{-1}, \text{ if } 1 \notin S,\, a \in S \\ \le \max\limits_{S \in \mathcal{S}} \sum_{a=2}^{n} \dfrac{(\theta_a^1)^2}{\sum_{i=2}^{n}(\theta_i^1)^2/\Delta_i^2} \dfrac{m}{\theta_S^1}\Bigg(\dfrac{1}{\theta_a^1}\Bigg) = m\Bigg(\sum_{a=2}^{n}\frac{(\theta_a^1)^2}{\Delta_a^2}\Bigg)^{-1}, \text{ if } a \notin S,\, 1 \in S \\ \le \max\limits_{S \in \mathcal{S}} \sum_{a=2}^{n} \dfrac{(\theta_a^1)^2}{\sum_{i=2}^{n}(\theta_i^1)^2/\Delta_i^2} \dfrac{m}{\theta_S^1}\Bigg(\dfrac{\theta_a^1+\theta_1^1}{\theta_1^1\theta_a^1}\Bigg) \le 2m\Bigg(\sum_{a=2}^{n}\frac{(\theta_a^1)^2}{\Delta_a^2}\Bigg)^{-1}, \text{ if both } 1, a \in S \\ = 0 \text{ otherwise} \end{cases}
$$

$$
\le 2m\Big(\sum_{a=2}^{n}\frac{(\theta_a^1)^2}{\Delta_a^2}\Big)^{-1}
$$

Then combining above upper bound to Eqn. 12, we finally get:

$$
\frac{\ln\frac{1}{2.4\delta}}{Q} \le 2m\Big(\sum_{a=2}^{n}\frac{(\theta_a^1)^2}{\Delta_a^2}\Big)^{-1} \implies \frac{1}{2.4\delta} \le \exp\Bigg(2mQ\Big(\sum_{a=2}^{n}\frac{(\theta_a^1)^2}{\Delta_a^2}\Big)^{-1}\Bigg)
$$

$$
\implies \delta \ge \frac{\exp\Big(-2mQ\Big(\sum_{a=2}^{n}\frac{(\theta_a^1)^2}{\Delta_a^2}\Big)^{-1}\Big)}{2.4},
$$

which proves the claim. Thus we show for any general problem instance, precisely $\text{PL}(n, \boldsymbol{\theta}^1)$, such that any $(0,\delta)$-PAC algorithm incurs an error on at least $\Omega\Bigg(\exp\Big(-2mQ\Big(\sum_{a=2}^{n}\frac{(\theta_a^1)^2}{\Delta_a^2}\Big)^{-1}\Big)\Bigg)$ towards identifying the Best-Item with Top-m Ranking feedback. $\square$

C.2. Pseudo-code for *Uniform-Allocation*

Algorithm 8 ***Uniform-Allocation***

input: Set of items: $[n]$, Subset size: $k \le n$, Ranking feedback size: $m \in [k-1]$, Sample complexity Q
init: $\mathcal{A} \leftarrow [n]$, $s \leftarrow 1$
while $|\mathcal{A}| \ge k$ **do**
  $\mathcal{B}_1, \mathcal{B}_2, \ldots \mathcal{B}_B \leftarrow$ *Partition*$(\mathcal{A}, k)$
  if $|\mathcal{B}_B| < k$, **then** $B \leftarrow B-1$, $\mathcal{R} \leftarrow \mathcal{B}_B$
  for $b \in [B]$ **do**
    Play the set $\mathcal{B}_b$ for $Q' := \frac{kQ}{2n + k\log_2 k}$ times
    For all $i, j \in \mathcal{B}_b$, update $\hat{p}_{ij}$ with *Rank-Breaking*
    Compute $w_i := \sum_{j \in \mathcal{B}_b} \mathbf{1}(\hat{p}_{ij} > \frac{1}{2})$
    Define $\bar{w} \leftarrow \text{Median}(\{w_i\}_{i \in \mathcal{B}_b})$, $\forall i \in \mathcal{B}_b$
    $\mathcal{A} \leftarrow \{i \in \mathcal{B}_b \mid w_i \ge \bar{w}\}$
  end for
  $\mathcal{A} \leftarrow \mathcal{A} \cup \mathcal{R}$;
end while
$\mathcal{B} \leftarrow \mathcal{A} \cup \{k - |\mathcal{A}|$ random elements from $[n] \setminus \mathcal{A}\}$
while $|\mathcal{A}| > 1$ **do**
  Play the set $\mathcal{B}$ for $Q' := \frac{kQ}{2n + k\log_2 k}$ times
  For all $i, j \in \mathcal{A}$, update $\hat{p}_{ij}$ with *Rank-Breaking*
  Compute $z_i := \sum_{j \in \mathcal{A}} \mathbf{1}(\hat{p}_{ij} > \frac{1}{2})$, $\forall i \in \mathcal{A}$
  Define $\bar{z} \leftarrow \text{Median}(\{z_i\}_{i \in \mathcal{A}})$
  $\mathcal{A} \leftarrow \{i \in \mathcal{A} \mid z_i \ge \bar{z}\}$
end while
output: The remaining item in $\mathcal{A}$

C.3. Proof of Thm. 12

Theorem 12 (*Uniform-Allocation*: Confidence bound for Best-Item identification with fixed sample complexity Q)**.** *Given a budget of Q rounds, Uniform-Allocation returns the Best-Item of $PL(n, \boldsymbol{\theta})$ with probability at least $1 - O\bigg(\log_2 n \exp\bigg(-\frac{mQ\Delta_{\min}^2}{16(2n+k\log_2 k)}\bigg)\bigg)$, where $\Delta_{\min} = \min_{i=2}^n \Delta_i$.*

Proof. Firstly, we establish that the sample complexity of *Uniform-Allocation* is always within the stipulated constraint Q.

Correctness of stipulated sample complexity (Q). To show this note that inside any round, for any batch $\mathcal{B}_b$, $b \in [B]$ $\frac{k}{2}$ items of $\mathcal{B}_b$, by definition of $\bar{w}$. Thus at each consecutive round, the number of surviving elements gets halved, which implies that the total number of rounds can be at most $\log_2 n$. Hence size of the set of surviving items $|\mathcal{A}|$ at round i is approximately $\frac{n}{2^{\ell-1}}$, for any round $\ell = 1, 2, \ldots \log_2 n$. Also number of sets formed at round i is $\Big\lfloor \frac{|\mathcal{A}|}{k} \Big\rfloor < \frac{|\mathcal{A}|}{k}$. Then total number of sets formed by the algorithm during its entire run can be at most:

$$\frac{n}{k}\big(1 + \frac{1}{2} + \frac{1}{2^2} + \cdots + \frac{1}{2^{\left\lceil \log_2 \frac{n}{k} \right\rceil}}\big) + \log_2 k < \frac{n}{k}\big(\sum_{i=0}^{\infty}\frac{1}{2^i}\big) + \log_2 k = \frac{(2n + k\log_2 k)}{k}$$

where the extra $\log_2 k$ term is due to the final $\log_2 k$ rounds for which $|\mathcal{A}| < k$. Now since our strategy is to allocate uniform budget across all sets, the assumign sample complexity per set is $Q' = \frac{Q}{\frac{(2n+k\log_2 k)}{k}} = \frac{Qk}{2n+k\log_2 k}$. Hence our algorithm is always within the budget constraint Q. The only part left is to now prove the confidence bound of Thm. 12, as analysed below:

Bounding the Best-Item identification confidence. We first analyse the any particular batch $\mathcal{B} \in \{\mathcal{B}_b\}_{b\in[B]}$, for any particular round $\ell = 1, 2, \ldots \log_2 n$, such that $1 \in \mathcal{B}$. Let us analyze the probability of item 1 getting eliminated from batch $\mathcal{B}$ at the end of round ℓ.

First recall the number of times $\mathcal{B}$ is sampled is $Q' = \frac{kQ}{2n+k\log_2 k}$. Now let us define w_i as the number of times item $i \in \mathcal{B}$ was returned in the top-m winner (i.e. i appeared in the Top-m Ranking feedback $\sigma \in \Sigma_S^m$) in Q' plays, and σ_τ be the top-m ranking retuned by the environment upon playing the batch $\mathcal{B}$ for the τ^{th} round, $\forall \tau \in [Q']$. Then given $\theta_1 = \arg\max_{i\in[n]} \theta_i$, clearly $Pr(\{1 \in \sigma_\tau\}) = \sum_{j=1}^m Pr\big(\sigma_\tau(j) = 1\big) = \sum_{j=0}^{m-1} \frac{1}{2(k-j)} \geq \frac{m}{k}$, since $Pr(\{1|S\}) = \frac{\theta_1}{\sum_{j\in S}\theta_j} \geq \frac{1}{|S|}$ for any $S \subseteq \mathcal{B}$. Thus $\mathbf{E}[w_1] = \sum_{\tau=1}^{Q'} \mathbf{E}[\mathbf{1}(1 \in \sigma_\tau)] \geq \frac{mQ'}{k}$. Now applying multiplicative Chernoff-Hoeffdings bound on the random variable w_1, we get that for any $\eta \in (0, 1]$,

$$Pr\Big(w_1 \leq (1-\eta)\mathbf{E}[w_1]\Big) \leq \exp\bigg(-\frac{\mathbf{E}[w_1]\eta^2}{2}\bigg) \leq \exp\bigg(-\frac{mQ'\eta^2}{2k}\bigg)$$

In particular, setting $\eta = \frac{1}{2}$ we get with probability at least $\Bigg(1 - \exp\bigg(-\frac{mQ'}{8k}\bigg)\Bigg)$, $w_1 > (1-\frac{1}{2})\mathbf{E}[w_1] > \frac{mQ'}{2k}$, for any such batch $\mathcal{B}$, at any round ℓ. This further implies that with probability at least $1 - \exp\bigg(-\frac{mQ'}{8k}\bigg)$, after Q' plays, we have $w_{1i} + w_{i1} \geq \frac{mQ'}{2k}$, for any item $i \in \mathcal{B} \setminus \{1\}$, as due to *Rank-Breaking* update whenever an item appears in Top-m Ranking feedback σ_τ, it ends up getting pairwise compared with the rest of the $k-1$ items in $\mathcal{B}$ after τ^{th} play. Let us denote $n_{1i} = w_{1i} + w_{i1}$. Then the probability that any suboptimal item $i \in \mathcal{B} \setminus \{1\}$ beats 1 after Q' plays is:

$$\begin{aligned}
Pr\bigg(\hat{p}_{1i} > \frac{1}{2}, n_{1i} \geq \frac{mQ'}{2k}\bigg) &= Pr\bigg(\hat{p}_{1i} - p_{1i} > \frac{1}{2} - \mathbf{p}_{1i}, n_{1i} \geq \frac{mQ'}{2k}\bigg) \\
&= Pr\bigg(\hat{p}_{1i} - p_{1i} > \mathbf{p}_{1i} - \frac{1}{2}, n_{1i} \geq \frac{mQ'}{2k}\bigg)
\end{aligned}$$

$$\leq Pr\bigg(\hat{p}_{1i} - p_{1i} > \frac{\Delta_i}{4}, n_{1i} \geq \frac{mQ'}{2k}\bigg) \;\; \bigg[\text{as, } \mathbf{p}_{1i} - \frac{1}{2} = \frac{(\theta_1 - \theta_i)}{2(\theta_1 + \theta_i)} > \frac{(\theta_1 - \theta_i)}{4}\bigg]$$
$$\leq \exp\Big(-2\frac{mQ'}{2k}\Big(\frac{\Delta_i}{4}\Big)^2\Big) = \exp\Big(-\frac{mQ'\Delta_i^2}{16k}\Big),$$

where the last inequality follows from Lem. 22 for $\eta = \frac{\Delta_i}{4}$, and $v = \frac{mQ'}{2k}$. So combining the above two claims, we get that the total probability of

$$\begin{aligned}
Pr\bigg(\hat{p}_{1i} > \frac{1}{2}\bigg) &= Pr\bigg(\hat{p}_{1i} > \frac{1}{2}, n_{1i} \geq \frac{mQ'}{2k}\bigg) + Pr\bigg(\hat{p}_{1i} > \frac{1}{2}, n_{1i} < \frac{mQ'}{2k}\bigg)\\
&\leq \exp\Big(-\frac{mQ'\Delta_i^2}{16k}\Big) + Pr\bigg(w_{1i} < \frac{mQ'}{2k}\bigg) \leq \exp\Big(-\frac{mQ'\Delta_i^2}{16k}\Big) + \exp\Big(-\frac{mQ'}{8k}\Big)\\
&\leq 2\exp\Big(-\frac{mQ'\Delta_i^2}{16k}\Big).
\end{aligned} \tag{13}$$

Now let us try to analyze that for a fixed round ℓ, how many such suboptimal item $i \in \mathcal{B} \setminus \{1\}$ can beat the Best-Item 1. Towards this we define a random variable $V := \sum_{i \in \mathcal{B}\setminus\{1\}} \mathbf{1}(\hat{p}_{i1} > \frac{1}{2})$. Now from (13) we get that:

$$\mathbf{E}[V] = \sum_{i \in \mathcal{B}\setminus\{1\}} Pr(\hat{p}_{i1} > \frac{1}{2}) \leq 2(k-1)\exp\Big(-\frac{mQ'\Delta_i^2}{16k}\Big)$$

Then applying Markov's inequality we have:

$$Pr\Big(V \geq \frac{k}{2}\Big) \leq \frac{\mathbf{E}[V]}{\frac{k}{2}} \leq \frac{4(k-1)}{k}\exp\Big(-\frac{mQ'\Delta_i^2}{16k}\Big) \leq \frac{4(k-1)}{k}\exp\Big(-\frac{mQ'\Delta_{\min}^2}{16k}\Big)$$

It is important to note that in case if $V < \frac{k}{2}, \implies z_1 > \frac{k}{2}$ and hence $z_1 > \bar{z}$, as $\bar{z} \leq \frac{k}{2}$.

Therefore with probability at least $\Big(1 - \frac{4(k-1)}{k}\exp\Big(-\frac{mQ'\Delta_{\min}^2}{16k}\Big)\Big)$, item 1 is not eliminated in round ℓ. Then the total probability of item 1 getting eliminated in the entire run of *Uniform-Allocation* can be upper bounded as:

$$\begin{aligned}
&Pr\Big(\exists \ell = 1, 2, \ldots log_2 n \text{ s.t. item 1 is eliminated at round } \ell\Big)\\
&\qquad\qquad \leq \sum_{\ell=1}^{\log_2 n} Pr\Big(\text{ Item 1 is eliminated at round } \ell\Big)\\
&\qquad\qquad \leq 4\log_2 n\frac{(k-1)}{k}\exp\Big(-\frac{mQ'\Delta_{\min}^2}{16k}\Big)\\
&\qquad\qquad = 4\log_2 n\frac{(k-1)}{k}\exp\Big(-\frac{mQ\Delta_{\min}^2}{16(2n + k\log_2 k)}\Big),
\end{aligned}$$

where the last equality follows recalling that we set $Q' = \frac{kQ}{2n + k\log_2 k}$, which concludes the first claim. The second claim simply follows from the first as the total error probability is upper bounded by δ, this further implies

$$\begin{aligned}
&\delta \leq 4\log_2 n\frac{(k-1)}{k}\exp\Big(-\frac{mQ\Delta_{\min}^2}{16(2n + k\log_2 k)}\Big)\\
&\implies Q \geq \frac{16(2n + k\log_2 k)}{m\Delta_{\min}^2}\ln\bigg(\frac{4(k-1)\log_2 n}{k\delta}\bigg)\bigg) = O\bigg(\frac{16(2n + k\log_2 k)}{m\Delta_{\min}^2}\ln\bigg(\frac{\log_2 n}{\delta}\bigg)\bigg)\bigg)
\end{aligned}$$

which proves the second claim. $\square$

D. Appendix for Sec. 6

Environments. 1. *g1*, 2. *g4*, 3. *arith*, 4. *geo*, 5. *b1* all with $n = 16$, and three larger models 5. *g4-big*, 6. *arith-big*, and 7. *geo-big* each with $n = 50$ items. Their individual score parameters are as follows: **1. g1:** $\theta_1 = 0.8, \theta_i = 0.2, \forall i \in [16] \setminus \{1\}$ **2. g4:** $\theta_1 = 1, \theta_i = 0.7, \forall i \in \{2, \ldots 6\}, \theta_i = 0.5, \forall i \in \{7, \ldots 11\}$, and $\theta_i = 0.01$ otherwise. **3. arith:** $\theta_1 = 1$ and $\theta_i - \theta_{i+1} = 0.06, \ \forall i \in [15]$. **4. geo:** $\theta_1 = 1$, and $\frac{\theta_{i+1}}{\theta_i} = 0.8, \ \forall i \in [15]$. **5. b1:** $\theta_1 = 0.8, \theta_i = 0.6, \forall i \in [16] \setminus \{1\}$ **6. g4b:** $\theta_1 = 1, \theta_i = 0.7, \forall i \in \{2, \ldots 18\}, \theta_i = 0.5, \forall i \in \{19, \ldots 45\}$, and $\theta_i = 0.01$ otherwise. **7. arithb:** $\theta_1 = 1$ and $\theta_i - \theta_{i+1} = 0.2, \ \forall i \in [49]$. **8. geob:** $\theta_1 = 1$, and $\frac{\theta_{i+1}}{\theta_i} = 0.9, \ \forall i \in [49]$.