Active Ranking with Subset-wise Preferences

Aadirupa Saha  
Indian Institute of Science, Bangalore  
aadirupa@iisc.ac.in

Aditya Gopalan  
Indian Institute of Science, Bangalore  
aditya@iisc.ac.in

Abstract

We consider the problem of probably approximately correct (PAC) ranking $n$ items by adaptively eliciting subset-wise preference feedback. At each round, the learner chooses a subset of $k$ items and observes stochastic feedback indicating preference information of the winner (most preferred) item of the chosen subset drawn according to a Plackett-Luce (PL) subset choice model unknown a priori. The objective is to identify an $\epsilon$-optimal ranking of the $n$ items with probability at least $1 - \delta$. When the feedback in each subset round is a single Plackett-Luce-sampled item, we show $(\epsilon, \delta)$-PAC algorithms with a sample complexity of $O\left(\frac{n}{\epsilon^2} \ln \frac{n}{\delta}\right)$ rounds, which we establish as being order-optimal by exhibiting a matching sample complexity lower bound of $\Omega\left(\frac{n}{\epsilon^2} \ln \frac{n}{\delta}\right)$—this shows that there is essentially no improvement possible from the pairwise comparisons setting ($k = 2$). When, however, it is possible to elicit top-$m$ ($\leq k$) ranking feedback according to the PL model from each adaptively chosen subset of size $k$, we show that an $(\epsilon, \delta)$-PAC ranking sample complexity of $O\left(\frac{n}{m^2} \ln \frac{n}{\delta}\right)$ is achievable with explicit algorithms, which represents an $m$-wise reduction in sample complexity compared to the pairwise case. This again turns out to be order-wise unimprovable across the class of symmetric ranking algorithms. Our algorithms rely on a novel pivot trick to maintain only $n$ itemwise score estimates, unlike $O(n^2)$ pairwise score estimates that has been used in prior work. We report results of numerical experiments that corroborate our findings.

1 Introduction

Ranking or sorting is a classic search problem and basic algorithmic primitive in computer science. Perhaps the simplest and most well-studied ranking problem is using (noisy) pairwise comparisons, which started from the work of Feige et al. [19], and which has recently been studied in machine learning under the rubric of ranking in ‘dueling bandits’ [9].

However, more general subset-wise preference feedback arises naturally in application domains where there is flexibility to learn by eliciting preference information from among a set of offerings, rather than by just asking for a pairwise comparison. For instance, web search and recommender systems applications typically involve users expressing preferences by clicking on one result (or a few results) from a presented set. Medical surveys, adaptive tutoring systems and multiplayer sports/games are other domains where subsets of questions, problem set assignments and tournaments, respectively, can be carefully crafted to learn users’ relative preferences by subset-wise feedback.

In this paper, we explore active, probably approximately correct (PAC) ranking of $n$ items using subset-wise, preference information. We assume that upon choosing a subset of $k \geq 2$ items, the learner receives preference feedback about the subset according to the well-known Plackett-Luce (PL) probability model [27].

The learner faces the goal of returning a near-correct ranking of all items, with respect to a tolerance parameter $\epsilon$ on the items’ PL weights, with probability at least $1 - \delta$ of correctness, after as few subset comparison rounds as possible. In this context, we make the following contributions:

1. We consider active ranking with winner information feedback, where the learner, upon playing a subset $S_t \subseteq [n]$ of exactly $k = |S_t|$ elements at each round $t$, receives as feedback a single winner sampled from the Plackett-Luce probability distribution on the elements of $S_t$. We design two $(\epsilon, \delta)$-PAC algorithms for this problem (Section 3) with sample complexity $O\left(\frac{n}{\epsilon^2} \ln \frac{n}{\delta}\right)$ rounds, for
learning a near-correct ranking on the items.

2. We show a matching lower bound of $\Omega \left( \frac{n}{m} \ln \frac{n}{\delta} \right)$ rounds on the $(\epsilon, \delta)$-PAC sample complexity of ranking with winner information feedback (Section 5), which is also of the same order as that for the dueling bandit ($k = 2$) [38]. This implies that despite the increased flexibility of playing larger sets, with just winner information feedback, one cannot hope for a faster rate of learning than in the case of pairwise comparisons.

3. In the setting where it is possible to obtain ‘top-rank’ feedback – an ordered list of $m \leq k$ items sampled from the Plackett-Luce distribution on the chosen subset – we show that natural generalizations of the winner-feedback algorithms above achieve $(\epsilon, \delta)$-PAC sample complexity of $O \left( \frac{n}{m^2} \ln \frac{n}{\delta} \right)$ rounds (Section 7), which is a significant improvement over the case of only winner information feedback. We show that this is orderwise tight by exhibiting a matching $\Omega \left( \frac{n}{m^2} \ln \frac{n}{\delta} \right)$ lower bound on the sample complexity across $(\epsilon, \delta)$-PAC algorithms.

4. We report numerical results to show the performance of the proposed algorithms on synthetic environments (Section 8).

By way of techniques, the PAC algorithms we develop leverage the property of independence of irrelevant attributes (IIA) of the Plackett-Luce model, which allows for $O(n)$ dimensional parameter estimation with tight confidence bounds, even in the face of a combinatorially large number of possible subsets of size $k$. We also devise a generic ‘pivoting’ idea in our algorithms to efficiently estimate a global ordering using only local comparisons with a pivot or probe element: split the entire pool into playable subsets all containing one common element, learn local orderings relative to this element and then merge. Here again, the IIA structure of the PL model helps to ensure consistency among preferences aggregated across disparate subsets but with a common reference pivot. Our sample complexity lower bounds are information-theoretic in nature and rely on a generic change-of-measure argument but with carefully crafted confusing instances.

Related Work. Over the years, ranking from pairwise preferences ($k = 2$) has been studied in both the batch or non-adaptive setting [20, 32, 37, 33] and the active or adaptive setting [7, 23, 4]. In particular, prior work has addressed the problem of statistical parameter estimation given preference observations from the Plackett-Luce model in the offline setting [36, 15, 26, 21]. There also have been recent developments on the PAC objective for different pairwise preference models, such as those satisfying stochastic triangle inequalities and strong stochastic transitivity [35], general utility-based preference models [36], the Plackett-Luce model [33] and the Mallows model [11]. Recent work has studied PAC-learning objectives other than identifying the single (near) best arm, e.g. recovering a few of the top arms [10, 28, 13], or the true ranking of the items [12, 18]. There is also work on the problem of Plackett-Luce parameter estimation in the subset-wise feedback setting [23, 26], but for the batch (offline) setup where the sampling is not adaptive. Recent work by Chen et al. [14] analyzes an active learning problem in the Plackett-Luce model with subset-wise feedback; however, the objective there is to recover the top-$\ell$ (unordered) items of the model, unlike full-rank recovery considered in this work. Moreover, they give instance-dependent sample complexity bounds, whereas we allow a tolerance ($\epsilon$) in defining good rankings, natural in many settings [34, 38, 11].

2 Preliminaries

Notation. We denote the set $[n] = \{1, 2, ..., n\}$. 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, the order induced by the natural global ordering $[n]$ of all the items). In this case, $S(i)$ denotes the item (member) at the $i$th position in subset $S$. $\Sigma_S = \{ \sigma | \sigma$ is a permutation over items of $S \}$. where for any permutation $\sigma \in \Sigma_S$, $\sigma(i)$ denotes the position of element $i \in S$ in the ranking $\sigma$. $1(\varphi)$ denote an indicator variable that takes the value 1 if the predicate $\varphi$ is true, and 0 otherwise. $Pr(A)$ is used to denote the probability of event $A$, in a probability space that is clear from the context. $Ber(p)$ and $Geo(p)$ respectively denote Bernoulli and geometric random variable with probability of success at each trial being $p \in [0, 1]$. Moreover, for any $n \in \mathbb{N}$, $Bin(n, p)$ and $NB(n, p)$ respectively denote Binomial and Negative Binomial distribution.

2.1 Discrete Choice Models and Plackett-Luce (PL)

A discrete choice model specifies the relative preferences of two or more discrete alternatives in a given set. A widely studied class of discrete choice models is the class of Random Utility Models (RUMs), which assume a ground-truth utility score $\theta_i \in \mathbb{R}$ for each alternative $i \in [n]$, and assign a conditional distribution $D_i(\cdot | \theta_i)$ for scoring item $i$. To model a winning alternative given any set $S \subseteq [n]$, one first draws a random utility score $X_i \sim D_i(\cdot | \theta_i)$ for each alternative in $S$, and selects an item with the highest random score.

\footnote{this is the ‘number of trials before success’ version}
One widely used RUM is the Multinomial-Logit (MNL) or Plackett-Luce model (PL), where the $D$'s are taken to be independent Gambel distributions with parameters $\theta'_i$, i.e., with probability densities $D_i(x_i|\theta'_i) = e^{-(x_i-\theta'_i)}, \theta'_i \in [n]$. Moreover assuming $\theta'_i = \theta_i$, it can be shown that the probability that an alternative $i$ emerges as the winner in the set $S \ni i$ becomes: $Pr(i|S) = \frac{\theta_i}{\sum_{j \in S} \theta_j}$.

Other families of discrete choice models can be obtained by imposing different probability distributions over the utility scores $X_i$, e.g., if $(X_1, \ldots, X_n) \sim N(\theta, \Lambda)$ are jointly normal with mean $\theta = (\theta_1, \ldots, \theta_n)$ and covariance $\Lambda \in \mathbb{R}^{n \times n}$, then the corresponding RUM-based choice model reduces to the Multinomial Probit (MNP).

**Independence of Irrelevant Alternatives** A choice model $Pr$ is said to possess the Independence of Irrelevant Attributes (IIA) property if the ratio of probabilities of choosing any two items, say $i_1$ and $i_2$ from within any choice set $S \ni i_1, i_2$ is independent of a third alternative $j$ present in $S$. Specifically, $\frac{Pr(i_1|S)}{Pr(i_2|S)} = \frac{Pr(i_1|S)}{Pr(i_2|S)}$ for any two distinct subsets $S_1, S_2 \subseteq [n]$ that contain $i_1$ and $i_2$. Plackett-Luce satisfies the IIA property.

## 3 Problem Setup

We consider the PAC version of the sequential decision-making problem of finding the ranking of $n$ items by making subset-wise comparisons. Formally, the learner is given a finite set $[n]$ of $n > 2$ arms. At each decision round $t = 1, 2, \ldots$, the learner selects a subset $S_t \subseteq [n]$ of $k$ items, and receives (stochastic) feedback about the winner (or most preferred) item of $S_t$ drawn from a Plackett-Luce (PL) model with parameters $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$, a priori unknown to the learner. The nature of the feedback is described in Section 3.1. We assume henceforth that $\theta_i \in [0, 1], \forall i \in [n]$, and also $1 = \theta_1 > \theta_2 > \ldots > \theta_n$ for ease of exposition.

**Definition 1 (ϵ-Best-Item).** For any $\epsilon \in (0, 1)$, an item $i$ is called ϵ-Best-Item if its PL score parameter $\theta_i$ is worse than the Best-Item $i^* = 1$ by no more than $\epsilon$, i.e. if $\theta_i \geq \theta_1 - \epsilon$. A 0-best item is an item with largest PL parameter, which is also a Condorcet winner in case it is unique.

**Definition 2 (ϵ-Best-Ranking).** We define a ranking $\sigma \in \Sigma_n$ to be an $\epsilon$-Best-Ranking when no pair of items in $[n]$ is misranked by $\sigma$ unless their PL scores are $\epsilon$-close to each other. Formally, $\exists i, j \in [n]$, such that $\sigma(i) > \sigma(j)$ and $\theta_i \geq \theta_j + \epsilon$. A 0-Best-Ranking will be called a Best-Ranking or optimal ranking of the PL model. With $1 = \theta_1 > \theta_2 > \ldots > \theta_n$, clearly the unique Best-Ranking is $\sigma^* = (1, 2, \ldots, n)$.

**Definition 3 (ϵ-Best-Ranking-Multiplicative).** We define a ranking $\sigma \in \Sigma_n$ of $\sigma^*$ to be $\epsilon$-Best-Ranking-Multiplicative if $\exists i, j \in [n]$, such that $\sigma(i) > \sigma(j)$, and $Pr(i|\{i, j\}) \geq \frac{1}{2} + \epsilon$.

Note: The term ‘multiplicative’ emphasizes the fact that the condition $Pr(i|\{i, j\}) \geq \frac{1}{2} + \epsilon$ equivalently imposes a multiplicative constraint $\theta_i \geq \theta_j \left(\frac{1/2 + \epsilon}{1/2 - \epsilon}\right)$ on the PL score parameters.

### 3.1 Feedback models

By feedback model, we mean the information received (from the ‘environment’) once the learner plays a subset $S \subseteq [n]$ of $k$ items. We consider the following feedback models in this work:

**Winner of the selected subset (WI):** The environment returns a single item $I \in S$, drawn independently from the probability distribution $Pr(I = i|S) = \frac{\theta_i}{\sum_{j \in S} \theta_j}$, $\forall i \in S$.

**Full ranking on the selected subset (FR):** The environment returns a full ranking $\sigma \in \Sigma_n$, drawn from the probability distribution $Pr(\sigma|S) = \prod_{i=1}^{|S|} \frac{\theta_i}{\sum_{j=1}^{|S|} \theta_j}$, $\sigma \in \Sigma_n$. This is equivalent to picking item $\sigma^{-1}(1) \in S$ according to winner (WI) feedback from $S$, then picking $\sigma^{-1}(2)$ according to WI feedback from $S \setminus \{\sigma^{-1}(1)\}$, and so on, until all elements from $S$ are exhausted, or, in other words, successively sampling $|S|$ winners from $S$ according to the PL model, without replacement. But more generally, one can define

**Top-$m$ ranking from the selected subset (TR-$m$ or TR):** The environment successively samples (without replacement) only the first $m$ items from among $S$, according to the PL model over $S$, and returns the ordered list. It follows that TR reduces to FR when $m = k = |S|$ and to WI when $m = 1$.

### 3.2 Performance Objective: $(\epsilon, \delta)$-PAC-Rank—Correctness and Sample Complexity

Consider a problem instance with Plackett-Luce (PL) model parameters $\theta \equiv (\theta_1, \ldots , \theta_n)$ and subsetsize $k \leq n$, with its Best-Ranking being $\sigma^* = (1, 2, \ldots, n)$, and $\epsilon, \delta \in (0, 1)$ are two given constants. A sequential algorithm that operates on this problem instance, with WI feedback model, is said to be $(\epsilon, \delta)$-PAC-Rank if (a) it stops and outputs a ranking $\sigma \in \Sigma_n$ after a finite number of decision rounds (subset plays) with probability $1$, and (b) the probability that its
output $\sigma$ is an $\epsilon$-Best-Ranking is at least $1 - \delta$, i.e., $Pr(\sigma$ is $\epsilon$-Best-Ranking) $\geq 1 - \delta$. Furthermore, by sample complexity of the algorithm, we mean the expected time (number of decision rounds) taken by the algorithm to stop.

In the context of our above problem objective, it is worth noting the work by [24] addressed a similar problem, except in the dueling bandit setup ($k = 2$) with the same objective as above, except with the notion of $\epsilon$-Best-Ranking-Multiplicative—we term this new objective as $(\epsilon, \delta)$-PAC-Rank-Multiplicative as referred later for comparing the results. The two objectives are however equivalent under a mild boundedness assumption as follows:

**Lemma 4.** Assume $\theta_i \in [a, b], \forall i \in [n]$, for any $a, b \in (0, 1)$. If an algorithm is $(\epsilon, \delta)$-PAC-Rank, then it is also $(\epsilon', \delta')$-PAC-Rank-Multiplicative for any $\epsilon' \leq \frac{\delta}{2}$. On the other hand, if an algorithm is $(\epsilon, \delta)$-PAC-Rank-Multiplicative, then it is also $(\epsilon', \delta')$-PAC-Rank for any $\epsilon' \leq 4\epsilon(1 + \epsilon)$.

4 Parameter Estimation with PL based preference data

We develop in this section some useful parameter estimation techniques based on adaptively sampled preference data from the PL model, which will form the basis for our PAC algorithms later on, in Section 5.1.

4.1 Estimating Pairwise Preferences via Rank-Breaking.

**Rank breaking** is a well-understood idea involving the extraction of pairwise comparisons from (partial) ranking data, and then building pairwise estimators on the obtained pairs by treating each comparison independently [26] [23], e.g., a winner $a$ sampled from among $a, b, c$ is rank-broken into the pairwise preferences $a \succ b, a \succ c$. We use this idea to devise estimators for the pairwise win probabilities $p_{ij} = P(i|\{i, j\}) = \theta_i / (\theta_i + \theta_j)$ in the active learning setting. The following result, used to design Algorithm 1 later, establishes explicit confidence intervals for pairwise win/loss probability estimates under adaptively sampled PL data.

**Lemma 5** (Pairwise win-probability estimates for the PL model). Consider a Plackett-Luce choice model with parameters $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$, and fix two 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 $i_t \in S_t, 1 \leq t \leq T$, such that for each $1 \leq t \leq T$, (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 $n_t(T) = \sum_{i=1}^T 1(i_t = i)$ and $n_{ij}(T) = \sum_{t=1}^T 1(\{i_t \in \{i, j\}\})$. Then, for any positive integer $v$, and $\eta \in (0, 1)$,

$$Pr\left(\frac{n_t(T)}{n_{ij}(T)} - \frac{\theta_i}{\theta_i + \theta_j} \geq \eta, n_{ij}(T) \geq v\right) \leq e^{-2\eta v^2},$$

$$Pr\left(\frac{n_t(T)}{n_{ij}(T)} - \frac{\theta_i}{\theta_i + \theta_j} \leq -\eta, n_{ij}(T) \geq v\right) \leq e^{-2\eta v^2}.$$

4.2 Estimating relative PL scores ($\theta_i/\theta_j$) using Renewal Cycles

We detail another method to directly estimate (relative) score parameters of the PL model, using renewal cycles and the IIA property.

**Lemma 6.** Consider a Plackett-Luce choice model with parameters $(\theta_1, \theta_2, \ldots, \theta_n), n \geq 2$, and an item $b \in [n]$. Let $i_1, i_2, \ldots$ be a sequence of iid draws from the model. Let $\tau = \min\{t \geq 1 \mid i_t = b\}$ be the first time at which $b$ appears, and for each $i \neq b$, let $w_i(\tau) = \sum_{t=1}^\tau 1(i_t = i)$ be the number of times $i \neq b$ appears until time $\tau$. Then, $\tau - 1$ and $w_i(\tau)$ are Geometric random variables with parameters $\frac{\theta_b}{\sum_{j \in [n]} \theta_j}$ and $\frac{\theta_b}{\theta_i + \theta_b}$, respectively.

With this in hand, we now show how fast the empirical mean estimates over several renewal cycles (defined by the appearance of a distinguished item) converge to the true relative scores $\theta_i/\theta_b$, a result to be employed in the design of Algorithm 3 later.

**Lemma 7** (Concentration of Geometric Random Variables via the Negative Binomial distribution.). Suppose $X_1, X_2, \ldots, X_d$ are iid $\text{Geo}(\frac{\theta_b}{\theta_i + \theta_b})$ random variables, and $Z = \sum_{i=1}^d X_i$. Then, for any $\eta > 0$,

$$Pr\left(\left|\frac{Z}{\eta} - \frac{\theta_i}{\theta_b}\right| \geq \eta\right) < 2exp\left(-\frac{2d\eta^2}{(1 + \frac{\theta_i}{\theta_b})(\eta + 1 + \frac{\theta_i}{\theta_b})}\right).$$

5 Algorithms for WI Feedback

This section describes the design of $(\epsilon, \delta)$-PAC-Rank algorithms with winner information (WI) feedback.

A key idea behind our proposed algorithms is to estimate the relative strength of each item with respect to a fixed item, termed as a pivot-item $b$. This helps to compare every item on common terms (with respect to the pivot item) even if two items are not directly compared with each other. Our first algorithm Beat-the-Pivot maintains pairwise score estimates $P_{ib}$ of the items $i \in [n]\{b\}$ with respect to the pivot element by deriving intuition from Lemma 5. The second algorithm Score-and-Rank directly estimates the relative scores $\frac{\theta_i}{\theta_b}$ for each item $i \in [n]\{b\}$, relying on Lemma 6 (Section 4.2). Once all item scores are estimated with enough confidence, the items are simply sorted with respect to their preference scores to obtain a ranking.
5.1 The Beat-the-Pivot algorithm

Algorithm 1 Beat-the-Pivot

1: Input:
2:  
3:  
4: Initialize:
5:  
6:  
7: If \( |G| < (k - 1) \), then set \( \mathcal{R} \Longleftrightarrow G \), and set \( S \setminus \mathcal{R}, S' \leftrightarrow \) Randomly sample \( k - 1 - |G| \) items from \( S \), and set \( G \Longleftrightarrow G \cup S' \).
8:  
9: for \( g = 1, 2, \ldots, G \) do
10:  
11:  
12:  
13: end for
14:  
15: Output: The ranking \( \sigma \in \Sigma_{[n]} \).

Beat-the-Pivot (Algorithm 1) first estimates an approximate Best-Item \( b \) with high probability \((1 - \delta/2)\). We do this using the subroutine Find-the-Pivot(n, k, \( \epsilon, \delta \)) (Algorithm Find-the-Pivot) that with probability at least \((1 - \delta)\) finds an \( \epsilon \)-Best-Item within a sample complexity of \( O(\frac{n}{\epsilon^2} \log \frac{2}{\delta}) \).

Once the best item \( b \) is estimated, Beat-the-Pivot divides the rest of the \( n - 1 \) items into groups of size \( k - 1 \), \( G_1, G_2, \ldots, G_C \), and appends \( b \) to each group. This way elements of every group get to compete with \( b \), which aids estimating the pairwise score compared to the pivot item \( b \), \( \hat{p}_{ib} \) owing to the HIA property of PL model and Lemma 5 (Section 4.1), sorting which we obtain the final ranking. Theorem 8 shows that Beat-the-Pivot enjoys the optimal sample complexity guarantee of \( O\left(\frac{1}{\epsilon^{2}} \log \left(\frac{2}{\delta}\right)\right) \).

Theorem 8 (Beat-the-Pivot: Correctness and Sample Complexity). Beat-the-Pivot (Algorithm 1) is \((\epsilon, \delta)\)-PAC-Rank with sample complexity \( O(\frac{n}{\epsilon^2} \log \frac{n}{\delta}) \).

5.2 The Score-and-Rank algorithm

Score-and-Rank (Algorithm 3) differs from Beat-the-Pivot in terms of the score estimate it maintains for each item. Unlike our previous algorithm, instead of maintaining pivot-preference scores \( \hat{p}_{ib} = P_{\mathcal{E}}(i > b) \), Beat-the-Pivot, aims to directly estimate the PL-score \( \theta_i \) of each item relative to score of the pivot \( \hat{b} \). In other words, the algorithm maintains the relative score estimates \( \frac{\theta_i}{\hat{b}} \) for every item \( i \in [n] \setminus \{b\} \) borrowing results from Lemma 6 and 7 and finally return the ranking sorting the items with respect to their relative pivotal-score. Score-and-Rank also runs within an optimal sample complexity of \( \left(\frac{n}{\epsilon^2} \ln \frac{n}{\delta}\right) \) as shown in Theorem 9. Pseudocode for the algorithm is detailed in Algorithm 8 in the appendix, due to space constraints.

Theorem 9 (Score-and-Rank: Correctness and Sample Complexity). Score-and-Rank (Algorithm 3) is \((\epsilon, \delta)\)-PAC-Rank with sample complexity \( O(\frac{n}{\epsilon^2} \log \frac{n}{\delta}) \).

5.3 The Find-the-Pivot subroutine (for algorithms 1 and 3)

In this section, we describe the pivot selection procedure Find-the-Pivot(n, k, \( \epsilon, \delta \)). The algorithm serves the purpose of finding an \( \epsilon \)-Best-Item with high probability \((1 - \delta)\) that is used as the pivoting element \( b \) by both Algorithm 1 and 3 (Section 5.1 and 5.2).

Find-the-Pivot is based on the simple idea of tracing the empirical best item—specifically, it maintains a running winner \( r_\ell \) at every iteration \( \ell \), making it compete with a set of \( k - 1 \) arbitrarily chosen items. After competing long enough \((\ell := O\left(\frac{n}{\epsilon^2} \ln \frac{n}{\delta}\right)\) rounds), if the empirical winner \( c_\ell \) turns out to be more than \( \frac{\epsilon}{\delta} \)-favorable than the running winner \( r_\ell \), in term of its pairwise preference score: \( \hat{p}_{c_\ell, r_\ell} > \frac{1}{2} + \frac{\epsilon}{\delta} \), then \( c_\ell \) replaces \( r_\ell \), or else \( r_\ell \) retains its place and status quo ensues. The formal description of Find-the-Pivot is in the appendix.

Lemma 10 (Find-the-Pivot: Correctness and Sample Complexity with WI). Find-the-Pivot (Algorithm 3) achieves the \((\epsilon, \delta)\)-PAC objective with sample complexity \( O\left(\frac{n}{\epsilon^2} \log \frac{n}{\delta}\right) \).

6 Lower Bound

In this section we show the minimum sample complexity required for any symmetric algorithm to be \((\epsilon, \delta)\)-PAC-Rank is at least \( \Omega\left(\frac{n}{\epsilon^2} \log \frac{n}{\delta}\right) \) (Theorem 12). Note this in fact matches the sample complexity bounds of our proposed algorithms (recall Theorem 8 and 9) showing the tightness of both our upper and lower bound guarantees. The key observation lies in noting that results are independent of \( k \), which shows the learning problem with \( k \)-subsetwise WI feedback is as hard as that of the dueling bandit setup \((k = 2)\)—the flexibility of playing a \( k \) sized subset does not help in faster information aggregation. We first define the notion of a symmetric or label-invariant algorithm.

Definition 11 (Symmetric Algorithm). A PAC algorithm \( A \) is said to be symmetric if its output is insensitive to the specific labelling of items, i.e., if for any PL model \((\theta_1, \ldots, \theta_n)\), bijection \( \phi : [n] \rightarrow [n] \),
[n] and ranking \( \sigma : [n] \rightarrow [n] \), it holds that
\[
Pr(A \text{ outputs } \sigma | (\theta_1, \ldots, \theta_n)) = Pr(A \text{ outputs } \sigma \circ \phi^{-1} | (\theta_{(1)}, \ldots, \theta_{(n)})),
\]
where \( Pr(\cdot | (\alpha_1, \ldots, \alpha_n)) \) denotes the probability distribution on the trajectory of \( A \) induced by the PL model \((\alpha_1, \ldots, \alpha_n)\).

**Theorem 12** (Lower bound on Sample Complexity with WI feedback). Given a fixed \( \epsilon \in (0, \frac{1}{3\sqrt{3}}) \), \( \delta \in [0, 1] \), and a symmetric \((\epsilon, \delta)\)-PAC-Rank algorithm \( A \) for WI feedback, there exists a PL instance \( \nu \) such that the sample complexity of \( A \) on \( \nu \) is at least \( \Omega \left( \frac{n^2}{\epsilon^2 \delta} \right) \).

**Proof. (sketch).** The argument is based on the following change-of-measure argument (Lemma 1) of [25]. (restated in Appendix D.1 as Lemma 25). To employ this result, note that in our case, each bandit instance corresponds to an instance of the problem with arm set containing all the subsets of \([n]\) of size \( k \): \( \{S = (S(1), \ldots, S(k)) \subseteq [n] | S(i) < S(j), \forall i < j \} \). The key part of our proof relies on carefully crafting a true instance, with optimal arm 1, and a family of slightly perturbed alternative instances \( \{\nu^a : a \neq 1\} \), each with optimal arm \( a \neq 1 \).

**Designing the problem instances.** We first renumber the \( n \) items as \( \{0, 1, 2, \ldots, n-1\} \). Now for any integer \( q \in [n-1] \), we define \( \nu[q] \) to be the set of problem instances where any instance \( \nu_S \in \nu[q] \) is associated to a set \( S \subseteq [n-1] \), such that \( |S| = q \), and the PL parameters \( \theta \) associated to instance \( \nu_S \) are set up as follows:
\[
\theta_0 = \theta \left( \frac{1}{2} - \epsilon^2 \right), \theta_j = \theta \left( \frac{1}{2} + \epsilon \right) \text{ for all } j \in S, \text{ and } \theta_j = \theta \left( \frac{1}{2} - \epsilon \right) \text{ for all } j \in [n-1] \setminus S, \text{ for some } \theta \in \mathbb{R}^+, \epsilon > 0.
\]
We will restrict ourselves to the class of instances of the form \( \nu[q], q \in [n-1] \).

Corresponding to each problem instance \( \nu_S \in \nu[q] \), such that \( q \in [n-2] \), consider a slightly altered problem instance \( \nu_{\tilde{S}} \) associated with a set \( \tilde{S} \subseteq [n-1] \), such that \( \tilde{S} = S \cup \{i\} \subseteq [n-1] \), where \( i \in [n-1] \setminus S \). Following the same construction as above, the PL parameters of the problem instance \( \nu_{\tilde{S}} \) are set up as:
\[
\theta_0 = \theta \left( \frac{1}{2} - \epsilon^2 \right), \theta_j = \theta \left( \frac{1}{2} + \epsilon \right) \text{ for all } j \in \tilde{S}, \text{ and } \theta_j = \theta \left( \frac{1}{2} - \epsilon \right) \text{ for all } j \in [n-1] \setminus \tilde{S}.
\]

**Remark 1.** Note that any problem instance \( \nu_S \in \nu[q], q \in [n-1] \) is thus can be uniquely defined by its underlying set \( S \in [n-1] \). For simplicity we will also use the notations \( S \in \nu[q] \) to define the problem instance.

**Remark 2.** It is easy to verify that, for any \( \theta \geq \frac{1}{1 - 2\epsilon} \), an \( \epsilon \)-Best-Ranking (Definition 1) for problem instance \( \nu_S, S \subseteq [n-1] \), say \( \sigma_S \), has to satisfy the following:
\[
\sigma_S(i) < \sigma_S(0), \forall i \in S \text{ and } \sigma_S(0) < \sigma_S(j), \forall j \in [n-1] \setminus S.
\]
Thus for any instance \( S \), the items in \( S \) should precede item 0 which itself precedes items in \([n-1] \setminus S\).

For any ranking \( \sigma \in \Sigma_n \), we denote by \( \sigma(1:i) \) the set first \( i \) items in the ranking, for any \( i \in [n] \).

We now fix any set \( S^* \subseteq [n-1] \), \( |S^*| = q = \lfloor \frac{n}{2} \rfloor \). Theorem [12] is now obtained by applying Lemma 25 on pair of instances \( (\nu_{S^*}, \nu_{\tilde{S}^*}) \), for all possible choices of \( \tilde{S} = S \cup \{i\}, i \in [n-1] \setminus S \), and for the event \( E := \{\sigma_A(1 : q + 1) = S^* \cup \{0\}\} \). However we apply a tighter upper bounds for the KL-divergence term in the right hand side of Lemma 25. It is easy to note that as \( A \) is \((\epsilon, \delta)\)-PAC-Rank, obviously \( Pr_{S^*}(\sigma_A(1 : q + 1) = S^* \cup \{0\}) > 1 - \delta \), and \( Pr_{\tilde{S}^*}(\sigma_A(1 : q + 1) = S^* \cup \{0\}) < 1 - \delta \). Further using kl\((Pr_{S^*}(E), Pr_{\tilde{S}^*}(E)) \geq kl(1 - \delta, \delta) \geq \ln \frac{1}{\epsilon} \) (due to Lemma 26), leads to a lower bound guarantee of \( \Omega \left( \frac{n}{\epsilon^2 \delta} \ln \frac{1}{\epsilon} \right) \), but that is loose by an \( \Omega(\frac{n}{\epsilon^2} \ln n) \) additive factor. Novelty of our analysis lies further utilising the symmetric property of \( A \) to prove a tighter upper bound on the KL-divergence with the following result.

**Lemma 13.** For any symmetric \((\epsilon, \delta)\)-PAC-Rank algorithm \( A \), and any problem instance \( \nu_S \in \nu[q] \) associated to the set \( S \subseteq [n-1] \), \( q \in [n-1] \), and for any item \( i \in S \), \( Pr_S(\sigma_A(1 : q) = S \setminus \{i\} \cup \{0\}) < \frac{\delta}{q} \), where \( Pr_S(\cdot) \) denotes the probability of an event under the underlying problem instance \( \nu_S \) and the internal randomness of the algorithm \( A \) (if any).

For our purpose, we use the above result for \( S = \tilde{S}^* \) which leads to the desired tighter upper bound for
\[
kl(Pr_{\nu_{S^*}}(E), Pr_{\nu_{\tilde{S}^*}}(E)) \geq kl(1 - \delta, \delta) \geq \ln \frac{1}{\epsilon},
\]
the last inequality follows due to Lemma 26 (Appendix D.2). The complete proof can be found in Appendix D.3.

**Remark 3.** Theorem 12 shows, rather surprisingly, that the PAC-ranking with winner feedback information from size-\( k \) subsets, does not become easier (in a worst-case sense) with \( k \), implying that there is no reduction in hardness of learning from the pairwise comparisons case \((k = 2)\). While one may expect sample complexity to improve as the number of items being simultaneously tested in each round \( (k) \) becomes larger, there is a counteracting effect due to the fact that it is intuitively ‘harder’ for a high-value item to win in just a single winner draw against a (large) population of \( k - 1 \) other competitors. A useful heuristic here is that the number of bits of information that a single winner draw from a size-\( k \) subset provides is \( O(\ln k) \), which is not significantly larger than when \( k > 2 \); thus, an algorithm cannot accumulate significantly more information per round compared to the pairwise case.
We also have a similar lower bound result for the $(\epsilon, \delta)$-PAC-Rank-Multiplicative objective of Szörényi et al. [24] (Section 3):

**Theorem 14.** Given a fixed $\epsilon \in \left(0, \frac{1}{4}\right]$, $\delta \in [0,1]$, and a symmetric $(\epsilon, \delta)$-PAC-Rank-Multiplicative algorithm $A$ for WI feedback model, there exists a PL instance $\nu$ such that the sample complexity of $A$ on $\nu$ is at least $\Omega\left(\frac{n}{\epsilon^2} \ln \frac{n}{4\delta}\right)$.

7 Analysis with Top Ranking (TR) feedback

We now proceed to analyze the problem with Top-$m$ Ranking (TR) feedback (Section 3.1). We first show that unlike WI feedback, the sample complexity lower bound here scales as $\Omega\left(\frac{n}{m^2} \ln \frac{n}{2}\right)$ (Theorem 15), which is a factor $m$ smaller than that in Theorem 12 for the WI feedback model. At a high level, this is because TR reveals preference information for $m$ items per feedback round, as opposed to just a single (noisy) information sample of the winning item (WI). Following this, we also present two algorithms for this setting which are shown to enjoy an exact optimal sample complexity guarantee of $O\left(\frac{n}{m^2} \ln \frac{n}{2}\right)$ (Section 7.2).

7.1 Lower Bound for Top-$m$ Ranking (TR) feedback

**Theorem 15** (Sample Complexity Lower Bound for TR). Given $\epsilon \in \left(0, \frac{1}{8}\right]$ and $\delta \in [0,1]$, and a symmetric $(\epsilon, \delta)$-PAC-Rank algorithm $A$ with top-$m$ ranking (TR) feedback $(2 \leq m \leq k)$, there exists a PL instance $\nu$ such that the expected sample complexity of $A$ on $\nu$ is at least $\Omega\left(\frac{n}{m^2} \ln \frac{n}{3}\right)$.

**Remark 4.** The sample complexity lower bound for $(\epsilon, \delta)$-PAC-Rank with top-$m$ ranking (TR) feedback model is $\frac{n}{m}$-times that of the WI model (Theorem 12). Intuitively, revealing a ranking on $m$ items in a $k$-set provides about $\ln \left(\frac{k}{m}\right)! = O(m \ln k)$ bits of information per round, which is about $m$ times as large as that of revealing a single winner, yielding an acceleration by a factor of $m$.

**Corollary 16.** Given $\epsilon \in \left(0, \frac{1}{\sqrt{8}}\right]$ and $\delta \in [0,1]$, and a symmetric $(\epsilon, \delta)$-PAC-Rank algorithm $A$ with full ranking (FR) feedback ($m = k$), there exists a PL instance $\nu$ such that the expected sample complexity of $A$ on $\nu$ is at least $\Omega\left(\frac{n}{k^2} \ln \frac{1}{4\delta}\right)$.

7.2 Algorithms for Top-$m$ Ranking (TR) feedback model

In this section we present a modification of Beat-the-Pivot (Algorithm 4) for $(\epsilon, \delta)$-PAC objective with top-$m$ ranking feedback. Algorithm 5 (Appendix E.1) shows that how a simple generalization of Beat-the-Pivot can proved to be $(\epsilon, \delta)$-PAC-Rank with optimal sample complexity guarantee (Theorem 17), using the idea of Rank-Breaking [20] on top-$m$ ranking feedback.

**Algorithm 5.** Generalizing Beat-the-Pivot for top-$m$ ranking (TR) feedback. The main trick we use in modifying Beat-the-Pivot for TR feedback is Rank Breaking, which essentially extracts pairwise comparisons from subset-wise feedback as described below:

**Rank-Breaking** [26]. Given any set $S$ of size $k$, if $\sigma \in \sum_{m}^{\sigma}$, $(S_m \subseteq S, |S_m| = m)$ denotes a possible top-$m$ ranking of $S$, the Rank Breaking subroutine considers each item in $S$ to be beaten by its preceding items in $\sigma$ in a pairwise sense. See Algorithm 4 for detailed description of the Rank-Breaking procedure.

Using Rank-Break (Algorithm 4), our modified Beat-the-Pivot algorithm now essentially maintains the empirical pivotal preferences $p_{ij}$ for each item $i \in |n| \setminus \{b\}$ by applying Rank Breaking on the TR feedback $\sigma$ of each subsetwise play. Of course in general, Rank Breaking may lead to arbitrarily inconsistent estimates of the underlying model parameters. However, owing to the IIA property of Plackett-Luce model, we get clean concentration guarantees on $p_{ij}$ using Lemma 3. This is precisely the idea used for obtaining the $\frac{m}{4}$ factor improvement in the sample complexity guarantees of Beat-the-Pivot as analysed in Theorem 8. The formal descriptions of Beat-the-Pivot generalized to the setting of TR feedback, is given in Algorithm 5.

**Theorem 17** (Beat-the-Pivot: Correctness and Sample Complexity with TR). With top-$m$ ranking (TR) feedback model, Beat-the-Pivot (Algorithm 5) is $(\epsilon, \delta)$-PAC-Rank with sample complexity $O\left(\frac{n}{m^2} \log \frac{n}{4\delta}\right)$.

**Remark 5.** Comparing Theorems 8 and 17 shows that the sample complexity of Beat-the-Pivot with TR feedback (Algorithm 5) is $m$ times smaller than its corresponding counterpart for WI feedback, owing to the additional information gain revealed from preferences of top-$m$ items instead of just $1$ (i.e. only the winner).

8 Experiments

We first present the setup of our empirical evaluations:

**Algorithms.** We simulate the results on our two proposed algorithms (1). Beat-the-Pivot and (2). Score-and-Rank. We also compare our ranking performance with the PLPAC-AMPR method, the only existing
method (to the best of our knowledge) that addresses the online PAC ranking problem, although only in the dueling bandit setup (i.e. $k = 2$).

**Ranking Performance Measure.** We use the popular pairwise Kendall’s Tau ranking loss [29] for measuring the accuracy of the estimated ranking $\sigma$ with respect to the Best-Ranking $\sigma^*$ with an additive $\epsilon$-relaxation: $d_\epsilon(\sigma^*, \sigma) = \frac{1}{2} \sum_{i< j} (g_{ij} + g_{ji})$, where each $g_{ij} = 1 ((\sigma(i) > \sigma(j)) \land (\sigma(i) > \sigma(j)))$. All reported performances are averaged across 50 runs.

**Environments.** We use four PL models: 1. `geo8` (with $n = 8$) 2. `arith10` (with $n = 10$) 3. `har20` (with $n = 20$) and 4. `arith50` (with $n = 50$). Their individual score parameters are as follows: 1. `geo8`: $\theta_i = 1$, and $\frac{\theta_i}{\theta_j} = 0.875$, $\forall i \in [7]$. 2. `arith10`: $\theta_i = 1$ and $\theta_i - \theta_{i+1} = 0.1$, $\forall i \in [9]$. 3. `har20`: $\theta = 1/(i)$, $\forall i \in [20]$. 4. `arith50`: $\theta_i = 1$ and $\theta_i - \theta_{i+1} = 0.02$, $\forall i \in [9]$.

Figure 1: Ranking performance vs. sample size (# rounds) with dueling plays ($k = 2$)

**Ranking with Pairwise Preferences ($k = 2$).** We first compare the above three algorithms with pairwise preference feedback, i.e. with $k = 2$ and $m = 1$ (WI feedback model). We set $\epsilon = 0.01$ and $\delta = 0.1$. Figure 1 clearly shows superiority of our two proposed algorithms over PLPAC-AMPR [33] as they give much higher ranking accuracy given the sample size, rightfully justifying our improved theoretical guarantees as well (Theorem 8 and 9). Note that `geo8` and `arith50` are the easiest and hardest PL model instances, respectively; the latter has the largest $n$ with gaps $\theta_i - \theta_{i+1} = 0.02$. This also reflects in our experimental results as the ranking estimation loss being the highest for `arith50` for all the algorithms, specifically PLPAC-AMPR very poorly till $10^4$ samples.

Figure 2: Ranking performance vs. subset size ($k$) with WI feedback ($m = 1$)

**Ranking with Subsetwise-Preferences ($k > 2$) with Winner feedback.** We next move to general subsetwise preference feedback ($k \geq 2$) for WI feedback model (i.e. for $m = 1$). We fix $\epsilon = 0.01$ and $\delta = 0.1$ and report the performance of Beat-the-Pivot on the datasets `har20` and `arith50`, varying $k$ over the range 4 - 40. As expected from Theorem 8 and explained in Remark 3, the ranking performance indeed does not seem to be varying with increasing subsetsize $k$ for WI feedback model for both PL models (Figure 2).

Figure 3: Ranking performance vs. feedback size ($m$) for fixed subset size ($k$)

**Ranking with Subsetwise-Preferences ($k > 2$) with Top-rank feedback.** We finally report the performance of Beat-the-Pivot (Algorithm 3) for top-$m$ ranking (TR) feedback model on two PL models: `har20` (for $k = 20$) and `arith50` (for $k = 45$), varying the range of $m$ from 2 to 40 (Figure 3). We set $\epsilon = 0.01$ and $\delta = 0.1$ as before. As expected, in this case indeed larger $m$ improves the ranking accuracy given a fixed sample size which reflects over theoretical guarantee of $\frac{1}{m}$-factor improvement of the sample complexity for TR feedback (Theorem 15 and Remark 5).

**9 Conclusion and Future Work**

We have considered the PAC version of the problem of adaptively ranking $n$ items from $k$-subset-wise comparisons, in the Plackett-Luce (PL) preference model with winner information (WI) and top ranking (TR) feedback. With just WI, the required sample complexity lower bound is $\Omega\left(\frac{n}{\epsilon^2} \ln \frac{n}{\delta}\right)$, which is surprisingly independent of the subset size $k$. We have also designed two algorithms enjoying optimal sample complexity guarantees, and based on a novel pivoting-trick. With TR feedback, a $\frac{1}{m}$-times faster learning rate is achievable, and we have given an algorithm with optimal sample complexity guarantees.

In the future, it would be of interest to analyse the problem with other choice models (e.g. multinomial probit, Mallows, nested logit, generalized extreme-value models, etc.), and perhaps to extend this theory to newer formulations such as assortment selection [5] [16], revenue maximization with item prices [35] [1], or even in contextual scenarios [17] where every individual user comes with their own model parameter.

---

3PLPAC-AMPR only works for $k = 2$ and is no longer applicable henceforth.
Acknowledgements

The authors are grateful to the anonymous reviewers for valuable feedback. This work is supported by a Qualcomm Innovation Fellowship 2018, and an Indigenous 5G Test Bed project grant from the Dept. of Telecommunications, Government of India.

References


