A Survey of Preference-based Online Learning with Bandit Algorithms*

Róbert Busa-Fekete and Eyke Hüllermeier

Department of Computer Science
University of Paderborn, Germany
{busarobi,eyke}@upb.de

Abstract. In machine learning, the notion of *multi-armed bandits* refers to a class of online learning problems, in which an agent is supposed to simultaneously explore and exploit a given set of choice alternatives in the course of a sequential decision process. In the standard setting, the agent learns from stochastic feedback in the form of real-valued rewards. In many applications, however, numerical reward signals are not readily available—instead, only weaker information is provided, in particular relative preferences in the form of qualitative comparisons between pairs of alternatives. This observation has motivated the study of variants of the multi-armed bandit problem, in which more general representations are used both for the type of feedback to learn from and the target of prediction. The aim of this paper is to provide a survey of the state-of-the-art in this field, that we refer to as *preference-based multi-armed bandits*. To this end, we provide an overview of problems that have been considered in the literature as well as methods for tackling them. Our systematization is mainly based on the assumptions made by these methods about the data-generating process and, related to this, the properties of the preference-based feedback.

Keywords: Multi-armed bandits, online learning, preference learning, ranking, top-k selection, exploration/exploitation, cumulative regret, sample complexity, PAC learning.

1 Introduction

Multi-armed bandit (MAB) algorithms have received considerable attention and have been studied quite intensely in machine learning in the recent past. The great interest in this topic is hardly surprising, given that the MAB setting is not only theoretically challenging but also practically useful, as can be seen from their use in a wide range of applications. For example, MAB algorithms turned out to offer effective solutions for problems in medical treatment design [35, 34], online advertisement [16], and recommendation systems [33], just to mention a few.

* To appear in: Proc. ALT 2014, 25th International Conference on Algorithmic Learning Theory, Bled, Slovenia.

The multi-armed bandit problem, or bandit problem for short, is one of the simplest instances of the sequential decision making problem, in which a *learner* (also called decision maker or agent) needs to select *options* from a given set of alternatives repeatedly in an online manner—referring to the metaphor of the eponymous gambling machine in casinos, these options are also associated with "arms" that can be "pulled". More specifically, the agent selects one option at a time and observes a numerical (and typically noisy) *reward* signal providing information on the quality of that option. The goal of the learner is to optimize an evaluation criterion such as the *error rate* (the expected percentage of playing a suboptimal arm) or the *cumulative regret* (the expected difference between the sum of the rewards actually obtained and the sum of rewards that could have been obtained by playing the best arm in each round). To achieve the desired goal, the online learner has to cope with the famous exploration/exploitation dilemma [5, 14, 35]: It has to find a reasonable compromise between playing the arms that produced high rewards in the past (exploitation) and trying other, possibly even better arms the (expected) reward of which is not precisely known so far (exploration).

The assumption of a numerical reward signal is a potential limitation of the MAB setting. In fact, there are many practical applications in which it is hard or even impossible to quantify the quality of an option on a numerical scale. More generally, the lack of precise feedback or exact supervision has been observed in other branches of machine learning, too, and has led to the emergence of fields such as *weakly supervised learning* and *preference learning* [25]. In the latter, feedback is typically represented in a purely qualitative way, namely in terms of pairwise comparisons or rankings. Feedback of this kind can be useful in online learning, too, as has been shown in online information retrieval [28, 42]. As another example, think of crowd-sourcing services like the Amazon Mechanical Turk, where simple questions such as pairwise comparisons between decision alternatives are asked to a group of annotators. The task is to approximate an underlying target ranking on the basis of these pairwise comparisons, which are possibly noisy and partially inconsistent [17]. Another application worth mentioning is the ranking of XBox gamers based on their pairwise online duels; the ranking system of XBox is called TrueSkill$^{\text{TM}}$[26].

Extending the multi-armed bandit setting to the case of preference-based feedback, i.e., the case in which the online learner is allowed to compare arms in a qualitative way, is therefore a promising idea. And indeed, extensions of that kind have received increasing attention in the recent years. The aim of this paper is to provide a survey of the state-of-the-art in the field of preference-based multi-armed bandits (PB-MAB). After recalling the basic setting of the problem in Section 2, we provide an overview of methods that have been proposed to tackle PB-MAB problems in Sections 3 and 4. Our main criterion for systematization is the assumptions made by these methods about the data-generating process or, more specifically, the properties of the pairwise comparisons between arms. Our survey is focused on the *stochastic* MAB setup, in which feedback is generated according to an underlying (unknown but stationary) probabilistic process; we do not cover the case of an *adversarial* data-generating processes, although this setting has recently received a lot of attention, too [1, 15, 14].

2 The Preference-based Bandit Problem

The stochastic MAB problem with pairwise comparisons as actions has been studied under the notion of "dueling bandits" in several papers [45, 44]. However, since this term is associated with specific modeling assumptions, we shall use the more general term "preference-based bandits" throughout this paper.

Consider a fixed set of arms (options) $\mathcal{A} = \{a_1, \ldots, a_K\}$. As actions, the learning algorithm (or simply the learner or agent) can perform a comparison between any pair of arms a_i and a_j, i.e., the action space can be identified with the set of index pairs (i, j) such that $1 \leq i \leq j \leq K$. We assume the feedback observable by the learner to be generated by an underlying (unknown) probabilistic process characterized by a *preference relation*

$$\mathbf{Q} = [q_{i,j}]_{1 \leq i,j \leq K} \in [0,1]^{K \times K} \enspace .$$

More specifically, for each pair of actions (a_i, a_j), this relation specifies the probability

$$\mathbf{P}\left(a_i \succ a_j\right) = q_{i,j} \tag{1}$$

of observing a preference for a_i in a direct comparison with a_j. Thus, each $q_{i,j}$ specifies a Bernoulli distribution. These distributions are assumed to be stationary and independent, both across actions and iterations. Thus, whenever the learner takes action (i, j), the outcome is distributed according to (1), regardless of the outcomes in previous iterations.

The relation $\mathbf{Q}$ is reciprocal in the sense that $q_{i,j} = 1 - q_{j,i}$ for all $i, j \in [K] = \{1, \ldots, K\}$. We note that, instead of only observing strict preferences, one may also allow a comparison to result in a *tie* or an *indifference*. In that case, the outcome is a trinomial instead of a binomial event. Since this generalization makes the problem technically more complicated, though without changing it conceptually, we shall not consider it further. In [12, 11], indifference was handled by giving "half a point" to both arms, which, in expectation, is equivalent to deciding the winner by flipping a coin. Thus, the problem is essentially reduced to the case of binomial outcomes.

We say arm a_i beats arm a_j if $q_{i,j} > 1/2$, i.e., if the probability of winning in a pairwise comparison is larger for a_i than it is for a_j. Clearly, the closer $q_{i,j}$ is to $1/2$, the harder it becomes to distinguish the arms a_i and a_j based on a finite sample set from $\mathbf{P}\left(a_i \succ a_j\right)$. In the worst case, when $q_{i,j} = 1/2$, one cannot decide which arm is better based on a finite number of pairwise comparisons. Therefore,

$$\Delta_{i,j} = q_{i,j} - \frac{1}{2}$$

appears to be a reasonable quantity to characterize the hardness of a PB-MAB task (whatever goal the learner wants to achieve). Note that $\Delta_{i,j}$ can also be negative (unlike the value-based setting, in which the quantity used for characterizing the complexity of a multi-armed bandit task is always positive and depends on the gap between the means of the best arm and the suboptimal arms).

2.1 Pairwise probability estimation

The decision making process iterates in discrete steps, either through a finite time horizon $\mathbb{T} = [T]$ or an infinite horizon $\mathbb{T} = \mathbb{N}$. As mentioned above, the learner is allowed to compare two actions in each iteration $t \in \mathbb{T}$. Thus, in each iteration t, it selects an index pair $1 \leq i(t) \leq j(t) \leq K$ and observes

$$\begin{cases} a_{i(t)} \succ a_{j(t)} & \text{with probability } q_{i(t),j(t)} \\ a_{j(t)} \succ a_{i(t)} & \text{with probability } q_{j(t),i(t)} \end{cases}$$

The pairwise probabilities $q_{i,j}$ can be estimated on the basis of finite sample sets. Consider the set of time steps among the first t iterations, in which the learner decides to compare arms a_i and a_j, and denote the size of this set by $n_{i,j}^t$. Moreover, denoting by $w_{i,j}^t$ and $w_{j,i}^t$ the frequency of "wins" of a_i and a_j, respectively, the proportion of wins of a_i against a_j up to iteration t is then given by

$$\widehat{q}_{i,j}^{\,t} = \frac{w_{i,j}^t}{n_{i,j}^t} = \frac{w_{i,j}^t}{w_{i,j}^t + w_{j,i}^t} \enspace .$$

Since our samples are assumed to be independent and identically distributed (i.i.d.), $\widehat{q}_{i,j}^{\,t}$ is a plausible estimate of the pairwise probability (1). Yet, this estimate might be biased, since $n_{i,j}^t$ depends on the choice of the learner, which in turn depends on the data; therefore, $n_{i,j}^t$ itself is a random quantity. A high probability confidence interval for $q_{i,j}$ can be obtained based on the Hoeffding bound [27], which is commonly used in the bandit literature. Although the specific computation of the confidence intervals may differ from case to case, they are generally of the form $[\widehat{q}_{i,j}^{\,t} \pm c_{i,j}^t]$. Accordingly, if $\widehat{q}_{i,j}^{\,t} - c_{i,j}^t > 1/2$, arm a_i beats arm a_j with high probability; analogously, a_i is beaten by arm a_j with high probability, if $\widehat{q}_{j,i}^{\,t} + c_{j,i}^t < 1/2$.

2.2 Evaluation criteria

The goal of the online learner is usually stated as minimizing some kind of cumulative regret. Alternatively, in the "pure exploration" scenario, the goal is to identify the best arm (or the best k arms, or a ranking of all arms) both quickly and reliably. As an important difference between these two types of targets, note that the regret of a comparison of arms depends on the concrete arms being chosen, whereas the sample complexity penalizes each comparison equally.

It is also worth mentioning that the notion of optimality of an arm is far less obvious in the preference-based setting than it is in the value-based (numerical) setting. In the latter, the optimal arm is simply the one with the highest expected reward—more generally, the expected reward induces a natural total order on the set of actions $\mathcal{A}$. In the preference-based case, the connection between the pairwise preferences $\mathbf{Q}$ and the order induced by this relation on $\mathcal{A}$ is less trivial; in particular, the latter may contain preferential cycles. We shall postpone a more detailed discussion of these issues to subsequent sections, and for the time being simply assume the existence of an arm a_{i^*} that is considered optimal.

2.3 Cumulative regret

In a preference-based setting, defining a reasonable regret is not as straightforward as in the value-based setting, where the sub-optimality of an action can be expressed easily on a numerical scale. In particular, since the learner selects two arms to be compared in an iteration, the sub-optimality of both of these arms should be taken into account. A commonly used definition of regret is the following [46, 43, 47, 45]: Suppose the learner selects arms $a_{i(t)}$ and $a_{j(t)}$ in time step t. Then, the *cumulative regret* incurred by the learner A up to time T is

$$R_A^T = \sum_{t=1}^{T} r^t = \sum_{t=1}^{T} \frac{\Delta_{i^*,i(t)} + \Delta_{i^*,j(t)}}{2} \quad . \tag{2}$$

This regret takes into account the optimality of both arms, meaning that the learner has to select two nearly optimal arms to incur small regret. Note that this regret is zero if the optimal arm a_{i^*} is compared to itself, i.e., if the learner effectively abstains from gathering further information and instead fully commits to the arm a_{i^*}.

2.4 Regret bounds

In a theoretical analysis of a MAB algorithm, one is typically interested in providing a bound on the (cumulative) regret produced by that algorithm. We are going to distinguish two types of regret bound. The first one is the *expected regret bound*, which is of the form

$$\mathbf{E}\left[R^T\right] \leq B(\mathbf{Q}, K, T) \ , \tag{3}$$

where $\mathbf{E}[\cdot]$ is the expected value operator, R^T is the regret accumulated till time step T, and $B(\cdot)$ is a positive real-valued function with the following arguments: the pairwise probabilities $\mathbf{Q}$, the number of arms K, and the iteration number T. This function may additionally depend on parameters of the learner, however, we neglect this dependence here. The expectation is taken with respect to the stochastic nature of the data-generating process and the (possible) internal randomization of the online learner. The regret bound (3) is technically akin to the expected regret bound of value-based multi-armed bandit algorithms like the one that is calculated for UCB [5], although the parameters used for characterizing the complexity of the learning task are different.

The bound in (3) does not inform about how the regret achieved by the learner is concentrated around its expectation. Therefore, we consider a second type of regret bound, namely one that holds with high probability. This bound can be written in the form

$$\mathbf{P}\Big(R^T < B(\mathbf{Q}, K, T, \delta) \Big) \geq 1 - \delta \ .$$

For simplicity, we also say that the regret achieved by the online learner is $\mathcal{O}(B(\mathbf{Q}, K, T, \delta))$ with high probability.

2.5 Sample complexity

The sample complexity analysis is considered in a "pure exploration" setup where the learner, in each iteration, must either select a pair of arms to be compared or terminate and return its recommendation. The *sample complexity of the learner* is then the number of pairwise comparisons it queries prior to termination, and the corresponding bound is denoted $B(\mathbf{Q}, K, \delta)$. Here, $1 - \delta$ specifies a lower bound on the probability that the learner terminates and returns the correct solution.[1] Note that only the number of the pairwise comparisons is taken into account, which means that pairwise comparisons are equally penalized, independently of the suboptimality of the arms chosen.

The recommendation of the learner depends on the task to be solved. In the simplest case, it consists of the best arm. However, as will be discussed in Section 4, more complex predictions are conceivable, such as a complete ranking of all arms.

The above sample complexity bound is valid most of the time (more than $1 - \delta$ of the runs). However, in case an error occurs and the correct recommendation is not found by the algorithm, the bound does not guarantee anything. Therefore, it cannot be directly linked to the expected sample complexity. In order to define the expected sample complexity, the learning algorithm needs to terminate in a finite number of steps with probability 1. Under this condition, running a learning algorithm on the same bandit instance results in a finite sample complexity, which is a random number distributed according to an unknown law $\mathbf{P} : \mathbb{N} \rightarrow [0, 1]$. The distribution $\mathbf{P}$ has finite support, since the algorithm terminates in a finite number of steps in every case. By definition, the *expected sample complexity* of the learning algorithm is the finite mean of the distribution $\mathbf{P}$. Moreover, the *worst case sample complexity* is the upper bound of the support of $\mathbf{P}$.

2.6 PAC algorithms

In many applications, one might be interested in gaining efficiency at the cost of optimality: The algorithm is allowed to return a solution that is only approximately optimal, though it is supposed to do so more quickly. For standard bandit

[1] Here, we consider the pure exploration setup with fixed confidence. Alternatively, one can fix the horizon and control the error of the recommendation [4, 8, 9]

problems, for example, this could mean returning an arm the expected reward of which deviates by at most some ϵ from the expected reward of the optimal arm.

In the preference-based setup, approximation errors are less straightforward to define. Nevertheless, the sample complexity can also be analyzed in a PAC-framework as originally introduced by Even-Dar *et al.* [20] for value-based MABs. A preference-based MAB algorithm is called (ϵ, δ)-PAC preference-based MAB algorithm with a *sample complexity* $B(\mathbf{Q}, K, \epsilon, \delta)$, if it terminates and returns an ϵ-optimal arm with probability at least $1 - \delta$, and the number of comparisons taken by the algorithm is at most $B(\mathbf{Q}, K, \epsilon, \delta)$. If the problem is to select a single arm, ϵ-optimality could mean, for example, that $\Delta_{i^*,j} < \epsilon$, although other notions of approximation can be used as well.

2.7 Explore-then-exploit algorithms

Most PB-MAB algorithms for optimizing regret are based on the idea of decoupling the exploration and exploitation phases: First, the algorithm tries to identify the best arm with high probability, and then fully commits to the arm found to be best for the rest of the time (i.e., repeatedly compares this arm to itself). Algorithms implementing this principle are called "explore-then-exploit" algorithms.

Such algorithms need to know the time horizon T in advance, since being aware of the horizon, the learning algorithm is able to control the regret incurred in case it fails to identify the best arm. More specifically, assume a so-called exploratory algorithm A to be given, which is able to identify the best arm a_{i^*} with probability at least $1 - \delta$. By setting δ to $1/T$, algorithm A guarantees that $\mathbf{P}(\,\widehat{i^*} = i^*\,) > 1 - 1/T$ if it terminates before iteration step T, where $\widehat{i^*}$ is the arm index returned by A. Thus, if A terminates and commits a mistake, i.e., $\widehat{i^*} \neq i^*$, then the expected regret incurred in the exploitation phase is $1/T \cdot \mathcal{O}(T) = \mathcal{O}(1)$, since the per-round regret is upper-bounded by one and the exploitation phase consists of at most T steps. Consequently, the expected regret of an explore-then-exploit algorithm is

$$\mathbf{E}[R^T] \leq (1 - 1/T)\, \mathbf{E}[R_A^T] + (1/T)\, \mathcal{O}(T) = \mathcal{O}\left(\mathbf{E}[R_A^T] + 1\right) \quad .$$

Note that the inequality is trivially valid if A does not terminate before T.

The same argument as given above for the case of expected regret also holds for high probability regret bounds in the explore-then-exploit framework. In summary, the performance of an explore-then-exploit algorithm is bounded by the performance of the exploration algorithm. More importantly, since the per round regret is at most one, the sample complexity of the exploration algorithm readily upper-bounds the expected regret; this fact was pointed out in [46, 44]. Therefore, like in the case of value-based MABs, explore-then-exploit algorithms somehow blur the distinction between the "pure exploration" and regret optimization setting.

However, in a recent study [47], a novel preference-based MAB algorithm is proposed that optimizes the cumulative regret without decoupling the exploration from the exploitation phase (for more details see Section 3.1). Without decoupling, there is no need to know the horizon in advance, which allows one to provide a *horizonless* regret bound that holds for any time step T.

The regret defined in (2) reflects the average quality of the decision made by the learner. Obviously, one can define a more strict or less strict regret by taking the maximum or minimum, respectively, instead of the average. Formally, the strong and weak regret in time step t are defined, respectively, as

$$\begin{aligned} r^t_{\max} &= \max\left\{\Delta_{i^*,i(t)}, \Delta_{i^*,j(t)}\right\} \quad , \\ r^t_{\min} &= \min\left\{\Delta_{i^*,i(t)}, \Delta_{i^*,j(t)}\right\} \quad . \end{aligned}$$

From a theoretical point of view, when the number of pairwise comparisons is bounded by a known horizon, these regret definitions do not lead to a fundamentally different problem. Roughly speaking, this is because most of the methods designed for optimizing regret seek to identify the best arm with high probability in the exploration phase, based on as few sample as possible.

3 Learning from Consistent Pairwise Comparisons

As explained in Section 2.1, learning in the preference-based MAB setting essentially means estimating the pairwise preference matrix $\mathbf{Q}$, i.e., the pairwise probabilities $q_{i,j}$. The target of the agent's prediction, however, is not the relation $\mathbf{Q}$ itself, but the best arm or, more generally, a ranking $\succ$ of all arms $\mathcal{A}$. Consequently, the least assumption to be made is a connection between $\mathbf{Q}$ and $\succ$, so that information about the former is indicative of the latter. Or, stated differently, the pairwise probabilities $q_{i,j}$ should be sufficiently consistent, so as to allow the learner to approximate and eventually identify the target (at least in the limit when the sample size grows to infinity). For example, if the target is a ranking $\succ$ on $\mathcal{A}$, then the $q_{i,j}$ should be somehow consistent with that ranking, e.g., in the sense that $a_i \succ a_j$ implies $q_{i,j} > 1/2$.

While this is only an example of a consistency property that might be required, different consistency or regularity assumptions on the pairwise probabilities $\mathbf{Q}$ have been proposed in the literature—needless to say, these assumptions have a major impact on how PB-MAB problems are tackled algorithmically. In this section and the next one, we provide an overview of approaches to such problems, categorized according to these assumptions (see Figure 1).

3.1 Axiomatic approaches

The seminal work of Yue *et al.* [44] relies on three regularity properties on the set of arms and their pairwise probabilities:

- *Total order over arms*: there exists a total order $\succ$ on $\mathcal{A}$, such that $a_i \succ a_j$ implies $\Delta_{i,j} > 0$.

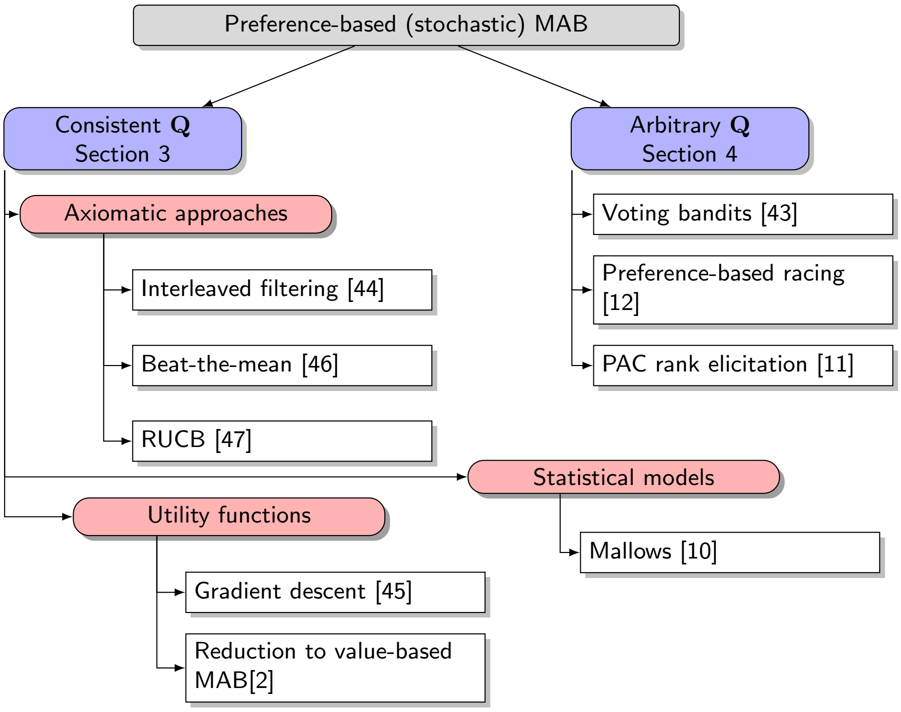
Preference-based (stochastic) MAB
Consistent **Q** Section 3
Arbitrary **Q** Section 4
Axiomatic approaches
Interleaved filtering [44]
Beat-the-mean [46]
RUCB [47]
Voting bandits [43]
Preference-based racing [12]
PAC rank elicitation [11]
Statistical models
Mallows [10]
Utility functions
Gradient descent [45]
Reduction to value-based MAB[2]

Fig. 1. A taxonomy of (stochastic) PB-MAB algorithms.

- *Strong stochastic transitivity*: for any triplet of arms such that $a_i \succ a_j \succ a_k$, the pairwise probabilities satisfy $\Delta_{i,k} \geq \max\left(\Delta_{i,j}, \Delta_{j,k}\right)$.
- *Stochastic triangle inequality*: for any triplet of arms such that $a_i \succ a_j \succ a_k$, the pairwise probabilities satisfy $\Delta_{i,k} \leq \Delta_{i,j} + \Delta_{j,k}$.

The first assumption of a total order with arms separated by positive margins ensures the existence of a unique best arm, which in this case coincides with the Condorcet winner.[2] The second and third assumptions induce a strong structure on the pairwise preferences, which allows one to devise efficient algorithms.

Interleaved filtering. Yue *et al.* [44] propose an explore-then-exploit algorithm. The exploration step consists of a simple sequential elimination strategy, called Interleaved Filtering (IF), which identifies the best arm with probability at least $1 - \delta$. The IF algorithm successively selects an arm which is compared to other arms in a one-versus-all manner. More specifically, the currently selected arm a_i is compared to the rest of the active (not yet eliminated)

[2] In voting and choice theory, an option is a Condorcet winner if it beats all other options in a pairwise comparison. In our context, this means an arm a_i is considered a Condorcet winner if $\Delta_{i,j} > 1/2$ for all $j \in [K]$.

arms. If an arm a_j beats a_i, that is, $\widehat{q}_{i,j} + c_{i,j} < 1/2$, then a_i is eliminated, and a_j is compared to the rest of the (active) arms, again in a one-versus-all manner. In addition, a simple pruning technique can be applied: if $\widehat{q}_{i,j} - c_{i,j} > 1/2$ for an arm a_j at any time, then a_j can be eliminated, as it cannot be the best arm anymore (with high probability). After the exploration step, the exploitation step simply takes the best arm $a_{\widehat{i}^*}$ found by IF and repeatedly compares $a_{\widehat{i}^*}$ to itself.

The authors analyze the expected regret achieved by IF. Assuming the horizon T to be finite and known in advance, they show that IF incurs an expected regret

$$\mathbf{E}\left[R^T_{\mathrm{IF}}\right] = \mathcal{O}\left(\frac{K}{\min_{j \neq i^*} \mathit{\Delta}_{i^*,j}} \log T\right) \quad .$$

Beat the mean. In a subsequent work, Yue and Joachims [46] relax the strong stochastic transitivity property and only require a so-called *relaxed stochastic transitivity* for the pairwise probabilities: There is a $\gamma \geq 1$ such that, for any triplet of arms such that $a_{i^*} \succ a_i \succ a_j$ with respect to the total order $\succ$,

$$\gamma \, \mathit{\Delta}_{i^*,j} \geq \max\left\{\mathit{\Delta}_{i^*,i}, \mathit{\Delta}_{i,j}\right\} \quad .$$

Obviously, the strong stochastic transitivity is recovered for $\gamma = 1$, albeit it is still restricted to triplets involving the best arm a_{i^*}. The stochastic triangle inequality is relaxed in a similar way, and again, it is required to hold only relative to the best arm.

With these relaxed properties, Yue and Joachims [46] propose a preference-based online learning algorithm called Beat-The-Mean (BTM), which is an elimination strategy resembling IF. However, while IF compares a single arm to the rest of the (active) arms in a one-versus-all manner, BTM selects an arm with the fewest comparisons so far and pairs it with a randomly chosen arm from the set of active arms (using the uniform distribution). Based on the outcomes of the pairwise comparisons, a score b_i is assigned to each active arm a_i, which is an empirical estimate of the probability that a_i is winning in a pairwise comparison (not taking into account which arm it was compared to). The idea is that comparing an arm a_i to the "mean" arm, which beats half of the arms, is equivalent to comparing a_i to an arm randomly selected from the active set. One can deduce a confidence interval for the b_i scores, which allows for deciding whether the scores for two arms are significantly different. An arm is then eliminated as soon as there is another arm with a significantly higher score.

In the regret analysis of BTM, a high probability bound is provided for a finite time horizon. More precisely, the regret accumulated by BTM is

$$\mathcal{O}\left(\frac{\gamma^7 K}{\min_{j \neq i^*} \mathit{\Delta}_{i^*,j}} \log T\right)$$

with high probability. This result is stronger than the one proven for IF, in which only the expected regret is upper bounded. Moreover, this high probability regret bound matches with the expected regret bound in the case $\gamma = 1$

(strong stochastic transitivity). The authors also analyze the BTM algorithm in a PAC setting, and find that BTM is an (ϵ, δ)-PAC preference-based learner (by setting its input parameters appropriately) with a sample complexity of $\mathcal{O}(\frac{\gamma^6 K}{\epsilon^2} \log \frac{KN}{\delta})$ if N is large enough, that is, N is the smallest positive integer for which $N = \left\lceil \frac{36\gamma^6}{\epsilon^2} \log \frac{K^3 N}{\delta} \right\rceil$. One may simplify this bound by noting that $N < N' = \left\lceil \frac{864\gamma^6}{\epsilon^2} \log \frac{K}{\delta} \right\rceil$. Therefore, the sample complexity is

$$\mathcal{O}\left(\frac{\gamma^6 K}{\epsilon^2} \log \frac{K\gamma \log(K/\delta)}{\delta\epsilon} \right) .$$

Preference-based UCB. In a very recent work by Zoghi *et al.* [47], the well-known UCB [5] algorithm is adapted from the value-based to the preference-based MAP setup. One of the main advantages of the proposed algorithm, called RUCB (for Relative UCB), is that only the existence of a Condorcet winner is required. Consequently, it is more broadly applicable. The RUCB algorithm is based on the "optimism in the face of uncertainty" principle, which means that the arms to be compared next are selected based on the optimistic estimates of the pairwise probabilities, that is, based on the upper boundaries $\widehat{q}_{i,j} + c_{i,j}$ of the confidence intervals. In an iteration step, RUCB selects the set of potential Condorcet winners for which all $\widehat{q}_{i,j} + c_{i,j}$ values are above $1/2$, and then selects an arm a_i from this set uniformly at random. Finally, a_i is compared to the arm a_j, $j = \operatorname{argmax}_{\ell \neq i} \widehat{q}_{i,\ell} + c_{i,\ell}$, that may lead to the smallest regret, taking into account the optimistic estimates.

In the analysis of the RUCB algorithm, horizonless regret bounds are provided, both for the expected regret and high probability bound. Thus, unlike the bounds for IF and BTM, these bounds are valid for each time step. Both the expected regret bound and high probability bound of RUCB are $\mathcal{O}(K \log T)$. However, while the regret bounds of IF and BTM only depend on $\min_{j \neq i^*} \mathit{\Delta}_{i^*,j}$, the constants are now of different nature, despite being still calculated based on the $\mathit{\Delta}_{i,j}$ values. Therefore, the regret bounds for RUCB are not directly comparable with those given for IF and BTM.

3.2 Regularity through latent utility functions

The representation of preferences in terms of utility functions has a long history in decision theory [22]. The idea is that the absolute preference for each choice alternative can be reflected by a real-valued utility degree. Obviously, such degrees immediately impose a total order on the set of alternatives. Typically, however, the utility degrees are assumed to be latent and not directly observable.

In [45], a preference-based stochastic MAB setting is introduced in which the pairwise probabilities are directly derived from the (latent) utilities of the arms. More specifically, the authors assume a space $\mathcal{S}$ of arms, which is not necessarily finite.[3] The probability of an arm $a \in \mathcal{S}$ beating arm $a' \in \mathcal{S}$ is given by

$$\mathbf{P}(a \succ a') = \frac{1}{2} + \delta(a, a')$$

where $\delta : \mathcal{S} \times \mathcal{S} \rightarrow [-1/2, 1/2]$. Obviously, the closer the value of the function δ is to 0, the harder it becomes to compare the corresponding pair of arms. The authors furthermore assume the pairwise δ-values to be connected to an underlying (differentiable and strictly concave) utility function $u : \mathcal{S} \rightarrow \mathcal{R}$:

$$\frac{1}{2} + \delta(a, a') = \sigma\big(u(a) - u(a')\big) \enspace ,$$

where $\sigma : \mathbb{R} \rightarrow [0, 1]$ is called *link function*, as it establishes a connection between the pairwise probabilities and utilities. This function is assumed to satisfy the following conditions: $\lim_{x\rightarrow\infty} \sigma(x) = 1$ and $\lim_{x\rightarrow-\infty} \sigma(x) = 0$, $\sigma(x) = 1 - \sigma(x)$, $\sigma(0) = 1/2$. An example of such a function is the logistic function, which was used in [45].

The problem of finding the optimal arm can be viewed as a noisy optimization task [21]. The underlying search space is $\mathcal{S}$, and the function values cannot be observed directly; instead, only noisy pairwise comparisons of function values (utilities) are available. In this framework, it is hard to have a reasonable estimate for the gradient, therefore the authors opted for applying an online convex optimization method [23], which does not require the gradient to be calculated explicitly.

In the theoretical analysis of the proposed method, the regret definition is similar to the one in (2), and can be written as

$$R^T = \sum_{t=1}^{T} \delta(a_*, a_t) + \delta(a_*, a'_t) \enspace .$$

Here, however, the reference arm a_* is the best one known only in hindsight. In other words, a_* is the best arm among those evaluated during the search process.

Under a strong convexity assumption on ϵ, an expected regret bound for the proposed algorithm is computed as follows. Assuming the search space $\mathcal{S}$ to be given by the d-dimensional ball of radius R, the expected regret is

$$\mathbf{E}[R^T] \leq 2T^{3/4}\sqrt{10RdL} \enspace .$$

Ailon *et al.* [47] propose various methodologies to reduce the utility-based PB-MAB problem to the standard value-based MAB problem. In their setup, the utility of an arm is assumed to be in $[0, 1]$. Formally, $u : \mathcal{S} \rightarrow [0, 1]$, and the link function is a linear function $\sigma_{lin}(x) = \frac{1}{2}x$. Therefore, the probability of an arm $a \in \mathcal{S}$ beating another arm $a' \in \mathcal{S}$ is

$$\mathbf{P}(a \succ a') = \frac{1 + u(a) - u(a')}{2} \enspace ,$$

[3] This space corresponds to our set of arms $\mathcal{A}$. However, as we assume $\mathcal{A}$ to be finite, we use another notation here.

which is again in $[0,1]$. The regret considered is the one defined in (2), where the reference arm a_{i^*} is the globally best arm with maximal utility.

In [47], two reduction techniques are proposed for a finite and an infinite set of arms. In both techniques, value-based MAB algorithms such as UCB [5] are used as a black box for driving the search in the space of arms. For a finite number of arms, value-based bandit instances are assigned to each arm, and these bandit algorithms are run in parallel. More specifically, assume that an arm $i(t)$ is selected in iteration t (to be explained in more detail shortly). Then, the bandit instance that belongs to arm $i(t)$ suggests another arm $j(t)$. These two arms are then compared in iteration t, and the reward, which is 0 or 1, is assigned to the bandit algorithm that belongs to $i(t)$. In iteration $t+1$, the arm $j(t)$ suggested by the bandit algorithm is compared, that is, $i(t+1) = j(t)$. What is nice about this reduction technique is that, under some mild conditions on the performance of the bandit algorithm, the preference-based expected regret defined in (2) is asymptotically identical to the one achieved by the value-based algorithm for the standard value-based MAB task.

For infinitely many arms, the reduction technique can be viewed as a two player game. A run is divided into epochs: the ℓ-th epoch starts in round $t = 2^\ell$ and ends in round $t = 2^{\ell+1} - 1$, and in each epoch the players start a new game. During the ℓth epoch, the second player plays adaptively according to a strategy provided by the value-based bandit instance, which is able to handle infinitely many arms, such as the ConfidenceBall algorithm by Dani *et al.* [19]. The first player obeys some stochastic strategy, which is based on the strategy of the second player from the previous epoch. That is, the first player always draws a random arm from the multi-set of arms that contains the arms selected by the second player in the previous epoch. This reduction technique incurs an extra $\log T$ factor to the expected regret of the value-based bandit instance.

3.3 Regularity through statistical models

Since the most general task in the realm of preference-based bandits is to elicit a ranking of the complete set of arms based on noisy (probabilistic) feedback, it is quite natural to establish a connection to statistical models of rank data [37]. This idea was recently put forward by Busa-Fekete *et al.* [10], who assume the underlying data-generating process to be given in the form of a probability distribution $\mathbf{P} : \mathbb{S}_K \to [0,1]$. Here, $\mathbb{S}_K$ is the set of all permutations of $[K]$ (the symmetric group of order K) or, via a natural bijection, the set of all rankings (total orders) of the K arms.

The probabilities for pairwise comparisons are then obtained as marginals of $\mathbf{P}$. More specifically, with $\mathbf{P}(\mathbf{r})$ the probability of observing the ranking $\mathbf{r}$, the probability $q_{i,j}$ that a_i is preferred to a_j is obtained by summing over all rankings $\mathbf{r}$ in which a_i precedes a_j:

$$q_{i,j} = \mathbf{P}(a_i \succ a_j) = \sum_{\mathbf{r} \in \mathcal{L}(r_j > r_i)} \mathbf{P}(\mathbf{r}) \tag{4}$$

where $\mathcal{L}(r_j > r_i) = \{\mathbf{r} \in \mathbb{S}_K \,|\, r_j > r_i\}$ denotes the subset of permutations for which the rank r_j of a_j is higher than the rank r_i of a_i (smaller ranks indicate higher preference).

In this setting, the learning problem essentially comes down to making inference about $\mathbf{P}$ based on samples in the form of pairwise comparisons. Concretely, three different goals of the learner are considered, depending on whether the application calls for the prediction of a single arm, a full ranking of all arms, or the entire probability distribution:

- The **MPI** problem consists of finding the most preferred item i^*, namely the item whose probability of being top-ranked is maximal:
$$i^* = \operatorname*{argmax}_{1\le i\le K} \mathbf{E}_{\mathbf{r}\sim\mathbf{P}}\, \mathbb{I}\{r_i = 1\} = \operatorname*{argmax}_{1\le i\le K} \sum_{\mathbf{r}\in\mathcal{L}(r_i=1)} \mathbf{P}(\mathbf{r}) \enspace ,$$
where $\mathbb{I}\{\cdot\}$ denotes the indicator function.
- The **MPR** problem consists of finding the most probable ranking $\mathbf{r}^*$:
$$\mathbf{r}^* = \operatorname*{argmax}_{\mathbf{r}\in\mathbb{S}_K} \mathbf{P}(\mathbf{r})$$
- The **KLD** problem calls for producing a good estimate $\widehat{\mathbf{P}}$ of the distribution $\mathbf{P}$, that is, an estimate with small KL divergence:
$$\mathrm{KL}\left(\mathbf{P}, \widehat{\mathbf{P}}\right) = \sum_{\mathbf{r}\in\mathbb{S}_K} \mathbf{P}(\mathbf{r}) \log \frac{\mathbf{P}(\mathbf{r})}{\widehat{\mathbf{P}}(\mathbf{r})} < \epsilon$$

All three goals are meant to be achieved with probability at least $1-\delta$.

Busa-Fekete *et al.* [10] assume the underlying probability distribution $\mathbf{P}$ to be a Mallows model [36], one of the most well-known and widely used statistical models of rank data [37]. The Mallows model or, more specifically, Mallows ϕ-distribution is a parameterized, distance-based probability distribution that belongs to the family of exponential distributions:

$$\mathbf{P}(\mathbf{r}\,|\,\theta, \widetilde{\mathbf{r}}) = \frac{1}{Z(\phi)}\, \phi^{d(\mathbf{r},\widetilde{\mathbf{r}})} \tag{5}$$

where ϕ and $\widetilde{\mathbf{r}}$ are the parameters of the model: $\widetilde{\mathbf{r}} = (\tilde{r}_1, \ldots, \tilde{r}_K) \in \mathbb{S}_K$ is the location parameter (center ranking) and $\phi \in (0,1]$ the spread parameter. Moreover, $d(\cdot,\cdot)$ is the Kendall distance on rankings, that is, the number of discordant pairs:

$$d(\mathbf{r}, \widetilde{\mathbf{r}}) = \sum_{1\le i<j\le K} \mathbb{I}\{\, (r_i - r_j)(\tilde{r}_i - \tilde{r}_j) < 0 \,\} \enspace .$$

The normalization factor in (5) can be written as

$$Z(\phi) = \sum_{\mathbf{r}\in\mathbb{S}_K} \mathbf{P}(\mathbf{r}\,|\,\theta, \widetilde{\mathbf{r}}) = \prod_{i=1}^{K-1} \sum_{j=0}^{i} \phi^j$$

and thus only depends on the spread [24]. Note that, since $d(\mathbf{r}, \widetilde{\mathbf{r}}) = 0$ is equivalent to $\mathbf{r} = \widetilde{\mathbf{r}}$, the center ranking $\widetilde{\mathbf{r}}$ is the mode of $\mathbf{P}(\cdot \,|\, \theta, \widetilde{\mathbf{r}})$, that is, the most probable ranking according to the Mallows model.

In the case of Mallows, it is easy to see that $\widetilde{r}_i < \widetilde{r}_j$ implies $q_{i,j} > 1/2$ for any pair of items a_i and a_j. That is, the center ranking defines a total order on the set of arms: If an arm a_i precedes another arm a_j in the (center) ranking, then a_i beats a_j in a pairwise comparison.[4] Moreover, as shown by Mallows [36], the pairwise probabilities can be calculated analytically as functions of the model parameters ϕ and $\widetilde{\mathbf{r}}$ as follows: Assume the Mallows model with parameters ϕ and $\widetilde{\mathbf{r}}$. Then, for any pair of items i and j such that $\widetilde{r}_i < \widetilde{r}_j$, the pairwise probability is given by $q_{i,j} = g(\widetilde{r}_i, \widetilde{r}_j, \phi)$, where

$$g(i, j, \phi) = h(j - i + 1, \phi) - h(j - i, \phi)$$

with $h(k, \phi) = k/(1 - \phi^k)$. Based on this result, one can show that the "margin"

$$\min_{i \neq j} |1/2 - q_{i,j}|$$

around 1/2 is relatively wide; more specifically, there is no $q_{i,j} \in (\frac{\phi}{1+\phi}, \frac{1}{1+\phi})$. Moreover, the result also implies that $q_{i,j} - q_{i,k} = O(\ell \phi^\ell)$ for arms a_i, a_j, a_k satisfying $\widetilde{r}_i = \widetilde{r}_j - \ell = \widetilde{r}_k - \ell - 1$ with $1 < \ell$, and $q_{i,k} - q_{i,j} = O(\ell \phi^\ell)$ for arms a_i, a_j, a_k satisfying $\widetilde{r}_i = \widetilde{r}_j + \ell = \widetilde{r}_k + \ell + 1$ with $1 < \ell$. Therefore, deciding whether an arm a_j has higher or lower rank than a_i (with respect to $\widetilde{\mathbf{r}}$) is easier than selecting the preferred option from two candidates a_j and a_k for which $j, k \neq i$.

Based on these observations, one can devise an efficient algorithm for identifying the most preferred arm when the underlying distribution is Mallows. The algorithm proposed in [10] for the **MPI** problem, called MallowsMPI, is similar to the one used for finding the largest element in an array. However, since a stochastic environment is assumed in which the outcomes of pairwise comparisons are random variables, a single comparison of two arms a_i and a_j is not enough; instead, they are compared until

$$1/2 \notin \left[\, \widehat{q}_{i,j} - c_{i,j}, \widehat{q}_{i,j} + c_{i,j} \,\right] \ . \tag{6}$$

This simple strategy finds the most preferred arm with probability at least $1 - \delta$ for a sample complexity that is of the form $\mathcal{O}\left(\frac{K}{\rho^2} \log \frac{K}{\delta\rho}\right)$, where $\rho = \frac{1-\phi}{1+\phi}$.

For the **MPR** problem, a sampling strategy called MallowsMerge is proposed, which is based on the merge sort algorithm for selecting the arms to be compared. However, as in the case of **MPI**, two arms a_i and a_j are not only compared once but until condition (6) holds. The MallowsMerge algorithm

[4] Recall that this property is an axiomatic assumption underlying the IF and BTM algorithms. Interestingly, the stochastic triangle inequality, which is also assumed by Yue *et al.* [44], is not satisfied for Mallows ϕ-model [36].

finds the most probable ranking, which coincides with the center ranking of the Mallows model, with a sample complexity of

$$\mathcal{O}\left(\frac{K\log_2 K}{\rho^2}\log\frac{K\log_2 K}{\delta\rho}\right) ,$$

where $\rho = \frac{1-\phi}{1+\phi}$. The leading factor of the sample complexity of MallowsMerge differs from the one of MallowsMPI by a logarithmic factor. This was to be expected, and simply reflects the difference in the worst case complexity for finding the largest element in an array and sorting an array using the merge sort strategy.

The **KLD** problem turns out to be very hard for the case of Mallows, and even for small K, the sample complexity required for a good approximation of the underlying Mallows model is extremely high with respect to ϵ. In [10], the existence of a polynomial algorithm for this problem (under the assumption of the Mallows model) was left as an open question.

4 Learning from Inconsistent Pairwise Comparisons

The methods presented in the previous section essentially proceed from a given target, for example a ranking $\succ$ of all arms, which is considered as a "ground truth". The preference feedback in the form of (stochastic) pairwise comparisons provide information about this target and, consequently, should obey certain consistency or regularity properties. This is perhaps most explicitly expressed in Section 3.3, in which the $q_{i,j}$ are derived as marginals of a probability distribution on the set of all rankings, which can be seen as modeling a noisy observation of the ground truth given in the form of the center ranking.

Another way to look at the problem is to start from the pairwise preferences $\mathbf{Q}$ themselves, that is to say, to consider the pairwise probabilities $q_{i,j}$ as the ground truth. In tournaments in sports, for example, the $q_{i,j}$ may express the probabilities of one team a_i beating another one a_j. In this case, there is no underlying ground truth ranking from which these probabilities are derived. Instead, it is just the other way around: A ranking is derived from the pairwise comparisons. Moreover, there is no reason for why the $q_{i,j}$ should be consistent in a specific sense. In particular, preferential cyclic and violations of transitivity are commonly observed in many applications.

This is exactly the challenge faced by *ranking procedures*, which have been studied quite intensely in operations research and decision theory [40, 18]. A ranking procedure $\mathcal{R}$ turns $\mathbf{Q}$ into a complete preorder relation $\succ^{\mathcal{R}}$ of the alternatives under consideration. Thus, another way to pose the preference-based MAB problem is to instantiate $\succ$ with $\succ^{\mathcal{R}}$ as the target for prediction—the connection between $\mathbf{Q}$ and $\succ$ is then established by the ranking procedure $\mathcal{R}$, which of course needs to be given as part of the problem specification.

Formally, a ranking procedure $\mathcal{R}$ is a map $[0,1]^{K\times K} \rightarrow \mathcal{C}_K$, where $\mathcal{C}_K$ denotes the set of complete preorders on the set of alternatives. We denote the complete preorder produced by the ranking procedure $\mathcal{R}$ on the basis of $\mathbf{Q}$ by $\succ^{\mathcal{R}}_{\mathbf{Q}}$, or simply by $\succ^{\mathcal{R}}$ if $\mathbf{Q}$ is clear from the context. In [12], three instantiations of the ranking procedure $\mathcal{R}$ are considered:

- Copeland's ranking (CO) is defined as follows [40]: $a_i \succ^{\mathrm{CO}} a_j$ if and only if $d_i > d_j$, where $d_i = \#\{k \in [K] \,|\, 1/2 < q_{i,k}\}$. The interpretation of this relation is very simple: An option a_i is preferred to a_j whenever a_i "beats" more options than a_j does.
- The sum of expectations (SE) (or Borda) ranking is a "soft" version of CO: $a_i \succ^{\mathrm{SE}} a_j$ if and only if
$$q_i = \frac{1}{K-1}\sum_{k \neq i} q_{i,k} > \frac{1}{K-1}\sum_{k \neq j} q_{j,k} = q_j \quad . \tag{7}$$
- The idea of the random walk (RW) ranking is to handle the matrix $\mathbf{Q}$ as a transition matrix of a Markov chain and order the options based on its stationary distribution. More precisely, RW first transforms $\mathbf{Q}$ into the stochastic matrix $\mathbf{S} = [s_{i,j}]_{K \times K}$ where $s_{i,j} = q_{i,j} / \sum_{\ell=1}^{K} q_{i,\ell}$. Then, it determines the stationary distribution $(v_1, \ldots, v_K)$ for this matrix (i.e., the eigenvector corresponding to the largest eigenvalue 1). Finally, the options are sorted according to these probabilities: $a_i \succ^{\mathrm{RW}} a_j$ iff $v_i > v_j$. The RW ranking is directly motivated by the PageRank algorithm [7], which has been well studied in social choice theory [3, 6] and rank aggregation [41], and which is widely used in many application fields [7, 32].

Top-k selection. The learning problem considered in [12] is to find, for some $k < K$, the top-k arms with respect to the above ranking procedures with high probability. To this end, three different learning algorithms are proposed in the finite horizon case, with the horizon given in advance. In principle, these learning problems are very similar to the value-based racing task [38, 39], where the goal is to select the k arms with the highest means. However, in the preference-based case, the ranking over the arms is determined by the ranking procedure instead of the means. Accordingly, the algorithms proposed in [12] consist of a successive selection and rejection strategy. The sample complexity bounds of all algorithms are of the form $\mathcal{O}(K^2 \log T)$. Thus, they are not as tight in the number of arms as those considered in Section 3. This is mainly due to the lack of any assumptions on the structure of $\mathbf{Q}$. Since there are no regularities, and hence no redundancies in $\mathbf{Q}$ that could be exploited, a sufficiently good estimation of the entire relation is needed to guarantee a good approximation of the target ranking in the worst case.

PAC rank elicitation. In a subsequent work by Busa-Fekete *et al.* [11], an extended version of the top-k selection problem is considered. In the PAC *rank elicitation problem*, the goal is to find a ranking that is "close" to the ranking produced by the ranking procedure with high probability. To make this problem feasible, more practical ranking procedures are considered. In fact, the problem of ranking procedures like Copeland is that a minimal change of a value $q_{i,j} \approx \frac{1}{2}$ may strongly influence the induced order relation $\succ^{\text{CO}}$. Consequently, the number of samples needed to assure (with high probability) a certain approximation quality may become arbitrarily large. A similar problem arises for $\succ^{\text{SE}}$ as a target order if some of the individual scores q_i are very close or equal to each other.

As a practical (yet meaningful) solution to this problem, the relations $\succ^{\text{CO}}$ and $\succ^{\text{SE}}$ are made a bit more "partial" by imposing stronger requirements on the order. To this end, let $d_i^* = \#\{k \,|\, 1/2 + \epsilon < q_{i,k}, i \neq k\}$ denote the number of options that are beaten by a_i with a margin $\epsilon > 0$, and let

$$s_i^* = \#\{k : |1/2 - q_{i,k}| \leq \epsilon,\, i \neq k\} \ \ .$$

Then, the ϵ-insensitive Copeland relation is defined as follows: $a_i \succ^{\text{CO}_\epsilon} a_j$ if and only if $d_i^* + s_i^* > d_j^*$. Likewise, in the case of $\succ^{\text{SE}}$, small differences of the q_i are neglected the ϵ-insensitive sum of expectations relation is defined as follows: $a_i \succ^{\text{SE}_\epsilon} a_j$ if and only if $q_i + \epsilon > q_j$.

These ϵ-insensitive extensions are interval (and hence partial) orders, that is, they are obtained by characterizing each option a_i by the interval $[d_i^*, d_i^* + s_i^*]$ and sorting intervals according to $[a, b] \succ [a', b']$ iff $b > a'$. It is readily shown that $\succ^{\text{CO}_\epsilon} \subseteq \succ^{\text{CO}_{\epsilon'}} \subseteq \succ^{\text{CO}}$ for $\epsilon > \epsilon'$, with equality $\succ^{\text{CO}_0} \equiv \succ^{\text{CO}}$ if $q_{i,j} \neq 1/2$ for all $i \neq j \in [K]$ (and similarly for SE). The parameter ϵ controls the strictness of the order relations, and thereby the difficulty of the rank elicitation task.

As mentioned above, the task in PAC rank elicitation is to approximate $\succ^{\mathcal{R}}$ without knowing the $q_{i,j}$. Instead, relevant information can only be obtained through sampling pairwise comparisons from the underlying distribution. Thus, the options can be compared in a pairwise manner, and a single sample essentially informs about a pairwise preference between two options a_i and a_j. The goal is to devise a *sampling strategy* that keeps the size of the sample (the sample complexity) as small as possible while producing an estimation $\succ$ that is "good" in a PAC sense: $\succ$ is supposed to be sufficiently "close" to $\succ^{\mathcal{R}}$ with high probability. Actually, the algorithms in [11] even produce a total order as a prediction, i.e., $\succ$ is a ranking that can be represented by a permutation τ of order K, where τ_i denotes the rank of option a_i in the order.

To formalize the notion of "closeness", appropriate distance measures are applied that compare a (predicted) permutation τ with a (target) order $\succ$. In [11], the following two measures are used: The *number of discordant pairs* (NDP), which is closely connected to Kendall's rank correlation [31], and can be expressed as follows:

$$d_{\mathcal{K}}(\tau, \succ) = \sum_{i=1}^{K} \sum_{j \neq i} \mathbb{I}\{\tau_j < \tau_i\} \mathbb{I}\{a_i \succ a_j\}.$$

The *maximum rank difference* (MRD) is defined as the maximum difference between the rank of an object a_i according to τ and $\succ$, respectively. More specifically, since $\succ$ is a partial but not necessarily total order, τ is compared to the set $\mathcal{L}^{\succ}$ of its linear extensions:[5]

$$d_{\mathcal{M}}(\tau, \succ) = \min_{\tau' \in \mathcal{L}^{\succ}} \max_{1 \le i \le K} |\tau_i - \tau_i'| .$$

In [11], the authors propose four different methods for the two ϵ-sensitive ranking procedures, along with the two distance measures described above. Each algorithm calculates a surrogate ranking based on the empirical estimate of the preference matrix whose distance can be upper-bounded again based on some statistics of the empirical estimates of preference. The sampling is carried out in a greedy manner in every case, in the sense that those arms are compared which are supposed to result in a maximum decrease of the upper bound calculated for the surrogate ranking.

An expected sample complexity bound is calculated for the ϵ-sensitive Copeland ranking procedure along with the MRD distance in a similar way like in [29, 30]. The bound is of the form $\mathcal{O}\left(R_1 \log\left(\frac{R_1}{\delta}\right)\right)$, where R_1 is a task dependent constant. More specifically, R_1 depends on the $\Delta_{i,j}$ values, and on the robustness of the ranking procedure to small changes in the preference matrix (i.e., on how much the ranking produced by the ranking procedure might be changed in terms of the MRD distance if the preference matrix is slightly altered). Interestingly, an expected sample complexity can also be calculated for the ϵ-insensitive sum of expectations ranking procedure along with the MRD distance with a similar flavor like for the ϵ-sensitive Copeland ranking procedure. The analysis of the NDP distance is more difficult, since small changes in the preference matrix may strongly change the ranking in terms of the NDP distance. The sample complexity analysis for this distance has therefore been left as an open question.

Urvoy *et al.* [43] consider a setup similar to the one in [11]. Again, a ranking procedure is assumed that produces a ranking over the arms, and the goal of the learner is to find a maximal element according to this ranking (instead of the top-k). Note that a ranking procedure only defines a complete preorder, which means there can be more than one "best" element. The authors propose an algorithm called SAVAGE as a general solution to this problem, which can be adapted to various ranking procedure. Concretely, the Copeland and the sum of expectations (or Borda counts) procedure are used in their study. Moreover, they also devise a method to find the Condorcet winner, assuming it exists—a problem that is akin to the axiomatic approaches described in Subsection 3.1.

The sample complexity of the implementations in [43] are of order K^2 in general. Just like in [11], this is the price to pay for a "model-free" learning procedure that does not make any assumptions on the structure of the preference matrix. The analysis of the authors is more general, because they also investigate the infinite horizon case, where a time limit is not given in advance.

5 Summary and Perspectives

This paper provides a survey of the state-of-the-art in preference-based online learning with bandit algorithms, an emerging research field that we referred to

[5] $\tau \in \mathcal{L}^{\succ}$ iff $\forall i, j \in [K] : (a_i \succ a_j) \Rightarrow (\tau_i < \tau_j)$

as preference-based multi-armed bandits (PB-MAB). In contrast to standard MAB problems, where bandit information is understood as (stochastic) real-valued rewards produced by the arms the learner decided to explore (or exploit), feedback is assumed to be of a more indirect and qualitative nature in the PB-MAB setting. In particular, the work so far has focused on preference information in the form of comparisons between pairs of arms. We have given an overview of instances of the PB-MAP problem that have been studied in the literature, algorithms for tackling them and criteria for evaluating such algorithms.

Needless to say, the field is still in its beginning and far from being mature. The contributions so far are highly interesting, and some of them have already been used in concrete applications, such as preference-based reinforcement learning [13]. Yet, they are still somewhat fragmentary, and a complete and coherent theoretical framework is still to be developed. With this survey, we hope to contribute to the popularization, development and shaping of the field.

We conclude the paper with a short (and certainly non-exhaustive) list of open problems that we consider particularly interesting for future work:

- As we have seen, the difficulty of PB-MAB learning strongly depends on the assumptions on properties of the preference relation $\mathbf{Q}$: The more restrictive these assumptions are, the easier the learning task becomes. An interesting question in this regard concerns the “weakest” assumptions one could make while still guaranteeing the existence of an algorithm that scales linearly in the number of arms.
- A similar question can be asked for the regret. The RUCB algorithm achieves a high probability regret bound of order $K \log T$ by merely assuming the existence of a Condorcet winner. Yet, this assumption is arguable and certainly not always valid.
- For most of the settings discussed in the paper, such as those based on statistical models like Mallows, a lower bound on the sample complexity is not known. Thus, it is difficult to say whether an algorithm is optimal or not. There are a few exceptions, however. For dueling bandits, it is known that, for any algorithm A, there is a bandit problem such that the regret of A is $\Omega(K \log T)$. Obviously, this is also a lower bound for all settings starting from weaker assumptions than dueling bandits, including RUCB.
- Another important problem concerns the development of (statistical) tests for verifying the assumptions made by the different approaches in a real application. In the case of the statistical approach based on the Mallows distribution, for example, the problem would be to decide, based on data in the form of pairwise comparisons, whether the underlying distribution could indeed be Mallows. Similarly, one could ask for methods to test the validity of strong stochastic transitivity and stochastic triangle inequality as required by methods such as IF and BTM.
- Last but not least, it would be important to test the algorithms in real applications—crowd-sourcing platforms appear to provide an interesting testbed in this regard.

Acknowledgments. The authors are grateful for financial support by the German Research Foundation (DFG).

References

1. N. Ailon, K. Hatano, and E. Takimoto. Bandit online optimization over the permutahedron. *CoRR*, abs/1312.1530, 2014.
2. N. Ailon, Z. Karnin, and T. Joachims. Reducing dueling bandits to cardinal bandits. In *Proceedings of the International Conference on Machine Learning (ICML), JMLR W&CP*, volume 32(1), pages 856–864, 2014.
3. A. Altman and M. Tennenholtz. Axiomatic foundations for ranking systems. *Journal of Artificial Intelligence Research*, 31(1):473–495, 2008.
4. J.Y. Audibert, S. Bubeck, and R. Munos. Best arm identification in multi-armed bandits. In *Proceedings of the Twenty-third Conference on Learning Theory (COLT)*, pages 41–53, 2010.
5. P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47:235–256, 2002.
6. F. Brandt and F. Fischer. PageRank as a weak tournament solution. In *Proceedings of the 3rd International Conference on Internet and Network Economics*, pages 300–305, 2007.
7. S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks*, 30(1-7):107–117, 1998.
8. S. Bubeck, R. Munos, and G. Stoltz. Pure exploration in finitely-armed and continuous-armed bandits. *Theoretical Computer Science*, 412:1832–1852, 2011.
9. S. Bubeck, T. Wang, and N. Viswanathan. Multiple identifications in multi-armed bandits. In *Proceedings of the International Conference on Machine Learning (ICML), JMLR W&CP*, volume 28 (1), pages 258–265, 2013.
10. R. Busa-Fekete, E. Hüllermeier, and B. Szörényi. Preference-based rank elicitation using statistical models: The case of Mallows. In *Proceedings of the International Conference on Machine Learning (ICML), JMLR W&CP*, volume 32 (2), pages 1071–1079, 2014.
11. R. Busa-Fekete, B. Szörényi, and E. Hüllermeier. PAC rank elicitation through adaptive sampling of stochastic pairwise preferences. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence (AAAI-14)*, 2014.
12. R. Busa-Fekete, B. Szörényi, P. Weng, W. Cheng, and E. Hüllermeier. Top-k selection based on adaptive sampling of noisy preferences. In *Proceedings of the International Conference on Machine Learning (ICML), JMLR W&CP*, volume 28 (3), pages 1094–1102, 2013.
13. R. Busa-Fekete, B. Szörényi, P. Weng, W. Cheng, and E. Hüllermeier. Preference-based reinforcement learning: Evolutionary direct policy search using a preference-based racing algorithm. *Machine Learning*, page accepted, 2014.
14. N. Cesa-Bianchi and G. Lugosi. *Prediction, Learning, and Games.* Cambridge University Press, NY, USA, 2006.
15. N. Cesa-Bianchi and G. Lugosi. Combinatorial bandits. In *Proceedings of the Twenty-second Conference on Learning Theory (COLT)*, pages 237–246, 2009.
16. D. Chakrabarti, R. Kumar, F. Radlinski, and E. Upfal. Mortal Multi-Armed Bandits. In *Neural Information Processing Systems, (NIPS)*, pages 273–280. MIT Press, 2008.

17. X. Chen, P. N Bennett, K. Collins-Thompson, and E. Horvitz. Pairwise ranking aggregation in a crowdsourced setting. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 193–202, 2013.
18. Y. Chevaleyre, U. Endriss, J. Lang, and N. Maudet. A short introduction to computational social choice. In *SOFSEM 2007: Theory and Practice of Computer Science*, pages 51–69. Springer Berlin Heidelberg, 2007.
19. V. Dani, T. P. Hayes, and S. M. Kakade. Stochastic linear optimization under bandit feedback. In *Proceedings of the Twenty-first Conference on Learning Theory (COLT)*, pages 355–366, 2008.
20. E. Even-Dar, S. Mannor, and Y. Mansour. PAC bounds for multi-armed bandit and markov decision processes. In *Proceedings of the 15th Conference on Learning Theory (COLT)*, pages 255–270, 2002.
21. S. Finck, H. Beyer, and A. Melkozerov. Noisy optimization: a theoretical strategy comparison of ES, EGS, SPSA & IF on the noisy sphere. In *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation (GECCO)*, pages 813–820. ACM, 2011.
22. P.C. Fishburn. *Utility theory for decision making.* New York: John Wiley and Sons, 1970.
23. A. Flaxman, A. T. Kalai, and B. H. McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. In *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 385–394, 2005.
24. M. A. Fligner and J. S. Verducci. Distance based ranking models. *Journal of the Royal Statistical Society. Series B (Methodological)*, 48(3):359–369, 1986.
25. J. Fürnkranz and E. Hüllermeier, editors. *Preference Learning.* Springer-Verlag, 2011.
26. S. Guo, S. Sanner, T. Graepel, and W. Buntine. Score-based bayesian skill learning. In *Proceedings of the European Conference on Machine Learning (ECML)*, pages 1–16, September 2012.
27. W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58:13–30, 1963.
28. K. Hofmann. *Fast and Reliably Online Learning to Rank for Information Retrieval.* PhD thesis, Dutch Research School for Information and Knowledge Systems, Off Page, Amsterdam, 2013.
29. S. Kalyanakrishnan. *Learning Methods for Sequential Decision Making with Imperfect Representations.* PhD thesis, University of Texas at Austin, 2011.
30. S. Kalyanakrishnan, A. Tewari, P. Auer, and P. Stone. Pac subset selection in stochastic multi-armed bandits. In *Proceedings of the Twenty-ninth International Conference on Machine Learning (ICML 2012)*, pages 655–662, 2012.
31. M.G. Kendall. *Rank correlation methods.* Charles Griffin, London, 1955.
32. A. Kocsor, R. Busa-Fekete, and S. Pongor. Protein classification based on propagation on unrooted binary trees. *Protein and Peptide Letters*, 15(5):428–34, 2008.
33. P. Kohli, M. Salek, and G. Stoddard. A fast bandit algorithm for recommendation to users with heterogenous tastes. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence (AAAI-13)*, 2013.
34. V. Kuleshov and D. Precup. Algorithms for multi-armed bandit problems. *CoRR*, abs/1402.6028, 2014.
35. T.L. Lai and H. Robbins. Asymptotically efficient allocation rules. *Advances in Applied Mathematics*, 6(1):4–22, 1985.
36. C. Mallows. Non-null ranking models. *Biometrika*, 44(1):114–130, 1957.

37. J. I. Marden. *Analyzing and Modeling Rank Data.* Chapman & Hall, 1995.
38. O. Maron and A.W. Moore. Hoeffding races: accelerating model selection search for classification and function approximation. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 59–66, 1994.
39. O. Maron and A.W. Moore. The racing algorithm: Model selection for lazy learners. *Artificial Intelligence Review*, 5(1):193–225, 1997.
40. H. Moulin. *Axioms of cooperative decision making.* Cambridge University Press, 1988.
41. S. Negahban, S. Oh, and D. Shah. Iterative ranking from pairwise comparisons. In *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, pages 2483–2491, 2012.
42. F. Radlinski, M. Kurup, and T. Joachims. How does clickthrough data reflect retrieval quality? In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM)*, pages 43–52, 2008.
43. T. Urvoy, F. Clerot, R. Féraud, and S. Naamane. Generic exploration and k-armed voting bandits. In *Proceedings of the 30th International Conference on Machine Learning (ICML), JMLR W&CP*, volume 28, pages 91–99, 2013.
44. Y. Yue, J. Broder, R. Kleinberg, and T. Joachims. The K-armed dueling bandits problem. *Journal of Computer and System Sciences*, 78(5):1538–1556, 2012.
45. Y. Yue and T. Joachims. Interactively optimizing information retrieval systems as a dueling bandits problem. In *Proceedings of the 26th International Conference on Machine Learning (ICML)*, pages 1201–1208, 2009.
46. Y. Yue and T. Joachims. Beat the mean bandit. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 241–248, 2011.
47. M. Zoghi, S. Whiteson, R. Munos, and M. de Rijke. Relative upper confidence bound for the k-armed dueling bandit problem. In *Proceedings of the International Conference on Machine Learning (ICML), JMLR W&CP*, volume 32 (1), pages 10–18, 2014.