PAC Maxing in Dueling Bandits under Stochastic Transitivity Models

Shivam Sharma

Department of Computer Science
Paderborn University, Germany
sshivam@mail.uni-paderborn.de

Abstract. Maximum selection (maxing) is a crucial and classical issue in Computer Science. It is one of the three fundamental problems in Probably Approximately Correct (PAC) preference learning through random pairwise comparisons. Maxing has been studied under different models and assumptions. This report presents a literature survey of the maxing issue under noisy probabilistic pairwise comparisons and looks upon the theoretical results given by Falahatgar et al. In this report, we will see why we need PAC maxing and some Notations used in PAC maxing along with some model constraints, which ensures the existence of an ϵ-maximum element in a set of elements of size n, where ϵ is the accuracy parameter.

Keywords: PAC maxing · Dueling bandits · Stochastic transitivity · Sample complexity · ϵ-maximum

1 Introduction

1.1 Background and Motivation

Maximum selection, or maxing, is a problem of identifying the best element from a set of different elements. This problem is a classical problem in computer science and has been revisited in machine learning. The identification of the best element is done through comparisons between the elements of the set. The best element is the one that wins all the comparisons against all the elements in the set.

In several real-time applications, pairwise comparisons produce random outcomes. In a chess tournament among players, the maxing problem is to find the best chess player based on the tournament's multiple chess matches. However, the outcome is rarely deterministic because, in each match, the result is not always the same. Either of the players can win the match, or there can be a tie or a stalemate. In other words, the outcome may differ from time to time for every match. Similarly, on Netflix, different individual content preferences can vary by a large margin, so the overall best content is different for each individual. Another example of maxing can be shown in the popular crowd-sourcing website GIFGIF [1]. The website uses maxing to assign emotions to the GIF image. The user is presented with two images. The platform asks to select one of them, which corresponds better to a given emotion. The maximum element in this example is the top GIF, which presents a given emotion better than all the other candidate GIFs in a set of GIF images.

The random pairwise comparison model is referred to as the dueling bandits model, which is a variation of the standard multi-armed bandit (MAB) problem [2]. In the MAB problem, the learner needs to select the elements or choices from a given set of alternative elements repeatedly in an online manner [3]. The "elements" or "choices" can be referred to as *arms* that can be "pulled," which represent the terminology of a slot machine in a casino [3]. The user selects one arm at a time and the learner observes a reward that determines the arm's quality [3]. In the dueling bandits problem [2], instead of pulling a single-arm, we choose a pair of arms (i, j) to duel and observe the winner feedback, i.e., which arm has won in the duel.

In many applications, efficiency is given preference over optimality, i.e., the algorithm is allowed to return an approximately optimal solution quickly instead of giving an accurate solution, which takes a long time to calculate [3]. This setting (where the priority is given to efficiency over optimality) is called Probably Approximately Correct (PAC) setting [4]. Even-Dar et al. [4] revisited the MAB problem in PAC setting and showed that, for given n arms, instead of pulling the arms $\mathcal{O}(\frac{n}{\epsilon^2} \log \frac{n}{\delta})$ times, it is adequate to pull the arms $\mathcal{O}(\frac{n}{\epsilon^2} \log \frac{1}{\delta})$ times under the PAC model to find an ϵ-optimal arm with probability of at least $1 - \delta$. Thus giving a general upper limit to the "sample complexity of the learner" [3]. The term "sample complexity of the learner" [3] is defined as "the amount of pairwise comparisons a learner queries prior to termination to return a nearly optimal target" in paper [3]. We will use the same definition for the expression "sample complexity" which is with respect to the number of arms in the set in this report. These notations are explained in Section 1.2.

1.2 Problem Description and Notations

Note: All the Notations in this section is due to [5,6,7], which will be used in this report to describe the problem statement.

Let $N = \{1, 2, \ldots, n\}$ be a fixed set of n arms, where n is a natural number. When two arms i and j are compared, the winner of the duel, say arm i, is returned with unknown pairwise probability $p_{i,j} \in [0, 1]$. The probability $p_{i,j}$ indicates the probability of observing a preference for i in comparison with j. These probabilistic pairwise comparisons are regarded as noisy comparisons. The learning algorithm can compare any pair of arms i and j such that $1 \leq i \leq j \leq n$. The pairwise probability $p_{j,i} = 1 - p_{i,j}$ (no ties) and $p_{i,i} = \frac{1}{2}$ holds true for all $i, j \in N$. We say that i beats j if $p_{i,j} > \frac{1}{2}$. This defines an order $\succ$ over arms i and j such that if i beats j, $i \succ j$ is implied which means i is preferable over j. Suppose $p_{i,j}$ is close to $\frac{1}{2}$, eg. $p_{i,j} = 0.49$, in that case, it becomes difficult to distinguish which arm wins and may take many arbitrary comparisons to find which arm is preferred. In this situation, we may not find an exact winner.

Instead, we can find an output that is a close approximation of the intended outcome.

For each $i, j \in N$, let $\tilde{p}_{i,j} = p_{i,j} - \frac{1}{2}$ be the calibrated pairwise probability [3] by which i is preferable over j. Note that $\tilde{p}_{i,j} = -\tilde{p}_{j,i}$ which means that the calibrated pairwise probability can be negative since there are no ties. For an accuracy parameter ϵ, an arm i is called ϵ-preferable to j if $\tilde{p}_{i,j} \geq -\epsilon$. Therefore, if arm $i \in N$ is ϵ-preferable to all other arms, then it is called ϵ-maximum in N for $\epsilon \in (0, \frac{1}{2})$. Given $\epsilon > 0$ and a confidence parameter $0 < \delta \leq \frac{1}{2}$, an (ϵ, δ)-PAC maxing algorithm must output an ϵ-maximum arm with probability $\geq 1 - \delta$. Arm i is called a maximal if it is preferable to every other arm in the set N i.e, $\tilde{p}_{i,j} > 0$ $\forall j \in N \setminus \{i\}$. However, there are cases where a maximal is not guaranteed to exist. An example of such a case is shown below.

Example 1 Let $N = \{1, 2, 3\}$ and $p_{1,2} = 0.7$, $p_{2,3} = 0.6$, $p_{3,1} = 0.9$. Since, $p_{j,i} = 1 - p_{i,j}$ (no ties) and $p_{i,i} = 0.5$, this gives us the following "probability preference matrix" [3] $P = [p_{i,j}]$, whose $(i,j)^{th}$ entry is $p_{i,j}$:

$$P = \begin{bmatrix} 0.5 & 0.7 & 0.1 \\ 0.3 & 0.5 & 0.6 \\ 0.9 & 0.4 & 0.5 \end{bmatrix}$$

This gives us the corresponding "calibrated probability preference matrix" [3]:

$$\tilde{P} = \begin{bmatrix} 0 & 0.2 & -0.4 \\ -0.2 & 0 & 0.1 \\ 0.4 & -0.1 & 0 \end{bmatrix}$$

where $\tilde{P} = [\tilde{p}_{i,j}]$, whose $(i,j)^{th}$ entry is the calibrated pairwise probability $\tilde{p}_{i,j} = p_{i,j} - \frac{1}{2}$. We can see from the first row of matrix $\tilde{P}$ that arm 1 is preferable over arm 2 ($\tilde{p}_{1,2} > 0$) but it is not preferred over arm 3 because $\tilde{p}_{1,3} < 0$. Hence, there is no maximal arm, preferable over every other arm in the set N. Therefore, we need additional constraints on $p_{i,j}$ to ensure the existence of a maximal.

We define some model restrictions on stochastic transitivity notions in Section 1.3. Aside from stochastic transitivity, another restriction called *Stochastic triangle inequality* (STI) is also used in some models. The STI is defined as: For a given total order over arms $\succ$ in N, the inequality $\tilde{p}_{i,k} \leq \tilde{p}_{i,j} + \tilde{p}_{j,k}$ holds for any triplet of arms $i, j, k \in N$ such that $i \succ j \succ k$ [5,7].

1.3 Stochastic Transitivity

Note: The definitions used in this section are defined in papers [5,6,7].
For any calibrated probability preference matrix $\tilde{P}$ on a set of arms N, the following transitivities are defined:

- *Strong stochastic transitivity* (SST): For all pairwise distinct $i, j, k \in N$ such that $\tilde{p}_{i,j} \geq 0$ and $\tilde{p}_{j,k} \geq 0$, the inequality $\tilde{p}_{i,k} \geq \max\{\tilde{p}_{i,j}, \tilde{p}_{j,k}\}$ holds.

- *γ-relaxed stochastic transitivity* (γ-RST): The inequality $\gamma{\cdot}\tilde{p}_{i,k} \geq \max\{\tilde{p}_{i,j}, \tilde{p}_{j,k}\}$ holds for $\gamma \geq 1$ and all pairwise distinct $i, j, k \in N$ such that $\tilde{p}_{i,j} \geq 0$ and $\tilde{p}_{j,k} \geq 0$. Note that, SST is a special case of γ-RST if $\gamma = 1$ [8].

- *Moderate stochastic transitivity* (MST): The inequality $\tilde{p}_{i,k} \geq \min\{\tilde{p}_{i,j}, \tilde{p}_{j,k}\}$ holds for all pairwise distinct $i, j, k \in N$ such that $\tilde{p}_{i,j} \geq 0$ and $\tilde{p}_{j,k} \geq 0$.

- *Weak stochastic transitivity* (WST): If $\tilde{p}_{i,j} \geq 0$ and $\tilde{p}_{j,k} \geq 0$, this implies that $\tilde{p}_{i,k} \geq 0$ for any triplet of arms $i, j, k \in N$. This means that the models which have total order in N satisfies WST [7]. However, a reverse implication is not true [3, Section 3.1].

The model becomes more restrictive when moving from WST to MST to SST [7, Section 1.2], i.e., the set of "consistent calibrated probability preference matrix" gets smaller as we move from WST to SST. To explain this, we consider the following example:

Example 2 Let $N = \{1, 2, 3\}$ and $\tilde{p}_{1,2} = 0.3$, $\tilde{p}_{2,3} = 0.1$, $\tilde{p}_{1,3} = x$. $\tilde{p}_{j,i} = -\tilde{p}_{i,j}$ (no ties) and $\tilde{p}_{i,i} = 0$ holds true for all $i, j \in N$. Then we have a calibrated probability preference matrix $\tilde{P} = [\tilde{p}_{i,j}]$:

$$\tilde{P} = \begin{bmatrix} 0 & 0.3 & x \\ -0.3 & 0 & 0.1 \\ -x & -0.1 & 0 \end{bmatrix}$$

where each $(i, j)^{th}$ entry is the calibrated pairwise probability.

- To satisfy WST, x should be in the range $(0, 0.5]$.

- To satisfy MST, i.e, $x \geq \min\{0.3, 0.1\}$, x should be in the range $[0.1, 0.5]$. This restricts the value for x in a small range.

- To satisfy γ-RST, i.e., for any $\gamma \geq 1$, $\gamma \cdot x \geq \max\{0.3, 0.1\}$, x should be in the range $(0, \frac{0.3}{\gamma}]$ for $\gamma > 1$ and in the range $[0.3, 0.5]$ for $\gamma = 1$. The range for x changes with respect to γ.

- Now to satisfy SST, i.e, $x \geq \max\{0.3, 0.1\}$, x should be in the range $[0.3, 0.5]$. This restricts the value for x in a more restricted range.

- However, to satisfy STI, the upper bound of the range changes from 0.5 to 0.4 since for STI, $x \leq 0.3 + 0.1$. Therefore, the value of x should lie in the range $(0, 0.4]$.

We can observe from the definitions that SST implies MST which further implicates WST. Although the reverse is not true. An example of these implications is shown below:

Example 3 Let us consider $\tilde{P}$ from Example 2 and choose $x = 0.5$, then the following holds true:

- SST is satisfied since $\tilde{p}_{1,2} \geq 0$, $\tilde{p}_{2,3} \geq 0$ and $\tilde{p}_{1,3} = 0.5 \geq \max\{0.1, 0.3\}$
- MST is also being satisfied since $\tilde{p}_{1,2} \geq 0$, $\tilde{p}_{2,3} \geq 0$ and $\tilde{p}_{1,3} \geq \min\{0.1, 0.3\}$.
- And finally WST is also satisfied.

Note that in Example 3, γ-RST won't be satisfied for $\gamma > 1$. However, since SST is a special case of γ-RST for $\gamma = 1$ [8], we can say that SST also implies γ-RST. Furthermore, γ-RST implies WST by definition. However, this does not mean that an exact implication between γ-RST and MST transitivity is fixed [3, Section 3.1]. Although, through definition, we can see that MST is weaker than γ-RST.

We can also observe that STI is also not satisfied in Example 3. This means that SST does not implies STI. Same can be said about the implication over STI by MST and WST and vice versa.

1.4 Outline

Section 2 looks into the work done in PAC maxing paradigm and provides an understanding of different optimal algorithms proposed by Falahatgar et al. [5,6,7] for different stochastic models. Section 3 looks into the SEQ-ELIMINATE algorithm given in [6] for PAC maxing in SST models without STI constraint. Section 4 looks into the PAC maxing in WST models with and without STI constraint given in [7]. Section 5 looks into the OPT-MAX algorithm given in [7] for PAC maxing in MST models with and without STI constraint.

2 PAC Maxing

Several researchers [4,8,9] have explored the maxing problem and came up with different algorithms to find a maximal from a set of arms with probability $\geq 1 - \delta$. But when the comparison probabilities approached half, these algorithms required arbitrarily many comparisons to find the maximal [5]. To achieve finite complexity, these researchers adopted the PAC paradigm. Nevertheless, even after adopting the PAC setting, many questions were raised regarding the PAC maximum selection in different stochastic scenarios:

1. Does a linearly complex PAC maxing algorithm, i.e., an algorithm which requires $\mathcal{O}(n)$ number of comparisons to find a maximum arm, exists for models with only SST and without STI constraints? [6]

2. Is PAC maxing for the most general models, i.e. the models under WST assumption, possible linearly with respect to the number of comparisons? [7]

3. If a linear PAC maxing is not possible for models under WST assumption, then is there any model which is more general than SST and less general than WST, for which PAC maxing of linear complexity is possible? [7]

From the paper [7, Table 1], Table 1 shows the upper/lower bounds on the sample complexity to solve PAC maxing with probability $\geq 1-\delta$ in different stochastic transitivity models.

Model	Maxing
SST, γ-RST with STI	SST: $\varTheta(\frac{n}{\epsilon^2}\log\frac{1}{\delta})$ γ-RST: $\varTheta(\frac{n\gamma^2}{\epsilon^2}\log\frac{1}{\delta})$ [5]
SST	$\varTheta(\frac{n}{\epsilon^2}\log\frac{1}{\delta})$ [6]
MST with and without STI	$\varTheta(\frac{n}{\epsilon^2}\log\frac{1}{\delta})^*$ [7]
WST with and without STI	$\varOmega(n^2)$ $\mathcal{O}(\frac{n^2}{\epsilon^2}\log\frac{n}{\delta})$ [7]

Table 1.

*: for $\delta \geq \min(\frac{1}{n}, e^{-n^{1/4}})$

As observed from Table 1, the upper bounds for models under MST +/- STI constraints and models under SST +/- STI constraints are the same, where "+" means "with" and "-" means "without". This means that there is **some** algorithm, which solves maxing under MST assumptions and maxing under SST assumptions optimally with the same upper bound on the sample complexity.

Note that although WST is the most general transitivity [7], however in Section 4, we will see that any algorithm under WST assumption requires a number of comparisons, which is quadratic in the number of arms, to solve maxing [7]. As MST is weaker than γ-RST and SST by definition, the upper bound on maxing under MST implies the upper bounds for maxing under SST and γ-RST, respectively. Thus, in this context, the upper bound on maxing under MST is the **strongest** bound among SST, γ-RST and MST.

Therefore, this report's primary focus will be the maxing under WST and maxing under MST assumption. We will give a summary on maxing for SST models without STI to answer Question 1.

3 For SST models without STI

For finding an ϵ-maximum arm from a set N of size n but with SST property only, Falahatgar et al. [6] presented the SEQ-ELIMINATE algorithm for the maxing problem in the (ϵ, δ)-PAC setting. This algorithm uses $\mathcal{O}(\frac{n}{\epsilon^2}\log\frac{n}{\delta})$ comparisons with probability $\geq 1-\delta$ to find an ϵ-maximum arm among the given alternatives for $\delta \leq \frac{1}{n}$ [6]. However, for $\delta > \frac{1}{n}$, the authors modified the algorithm to obtain an ϵ-maximum arm with probability $\geq 1-\delta$ using $\mathcal{O}(\frac{n}{\epsilon^2}\log\frac{1}{\delta})$ comparisons. This shows that, for PAC maxing with only SST assumption, a PAC maxing algorithm with $\mathcal{O}(n)$ sample complexity **exists**, thus answering Question 1.

Since, SEQ-ELIMINATE runs linearly in terms of sample complexity under a less restrictive setting (as described by it's theoretical guarantee), this means that its performance will not be affected by adding more constraints such as STI since PAC maxing becomes easier under more constraints [6]. However, SEQ-ELIMINATE will fail if SST is not satisfied [6].

Furthermore, its theoretical guarantee of using linear number of comparisons to find an ϵ-maximum arm with probability $\geq 1 - \delta$ is proved using Hoeffding's inequality and union bound. Hoeffding's inequality is used to prove that an expected feedback is returned for a duel between two arms with probability $\geq 1 - \delta$. The union bound guarantees that the feedback is valid for all the duels in the set of arms.

4 For WST models with and without STI

In a more restrictive environment, like SST +/- STI, it is easier to find an ϵ-maximum arm with probability $\geq 1 - \delta$. Algorithms such as KNOCKOUT [5] and SEQ-ELIMINATE [6] solve maxing linearly under a much restrictive environment. Thus, we focus on more general transitive models for which maxing is difficult, but possible, to solve linearly. By definition, WST is the most general transitive model. This property led to the search for a linear complexity maxing algorithm for models under WST assumptions.

To find the lower bound for PAC maxing in WST models, Falahatgar et al. [7] reduced the problem of finding a $\frac{1}{4}$-maximum for $\delta \leq \frac{1}{8}$ to find the left most piece of a linear jigsaw puzzle. They assumed a model called M_μ [7] with set $N = \{a_1, a_2, \ldots, a_n\}$ containing n arms such that

- $\tilde{p}_{a_i,a_{i+1}} = \frac{1}{2}, \forall i < n$
- $\tilde{p}_{a_i,a_j} = \mu, \forall j > i + 1$ where $0 < \mu < \frac{1}{n^{10}}$

The arm a_1 is the only $\frac{1}{4}$-maximum arm in this model [7]. Since $\tilde{p}_{a_i,a_j} > 0$, this means that $a_i \succ a_j$ if $i < j$ and therefore, this model satisfies WST [7].

In this model, for $\delta \leq \frac{1}{8}$, any algorithm requires $\Omega(n^2)$ many comparisons to find a $\frac{1}{4}$-maximum [7]. The proof of this statement is quite long; therefore, we provide a proof sketch which is from the proof in [7, Apendix A].

Proof Sketch: Since $\tilde{p}_{a_i,a_{i+1}} = \frac{1}{2}$, $\forall i < n$, a_i is always preferred to a_{i+1}, but for the output of comparisons of every non-consecutive pair of arms, it is almost a fair coin flip. This scenario poses a threat of arbitrary multiple comparisons to find the winner between non-consecutive arms. It can be simplified by giving extra information on whether the two arms that are being compared are adjacent or not. Thus modifying the problem, which makes it similar to a linear jigsaw puzzle problem. For defining the modified model, namely M_0, M_μ is reduced to a new one where $\mu = 0$, which now represents the model for a linear jigsaw puzzle. In this new model, the arms are called "pieces". If two pieces are compared, the comparison's feedback tells which piece is on the left if the two pieces are adjacent. The goal of the algorithm is to find the left most piece. The two models are compared to find a similarity between them. It turns out that if an algorithm uses $\leq n^2/20$ number of comparisons to find $\frac{1}{4}$-maximum arm under M_μ with probability $\geq \frac{7}{8}$, then $\frac{1}{4}$-maximum is obtained with probability $\geq \frac{3}{4}$ by applying the same algorithm over the M_0 model. After establishing a similarity, a lower bound on the sample complexity is determined for any algorithm which $\frac{1}{4}$-maximum for M_0. Since the models are similar, the lower bound of M_0 implies the lower bound on M_μ.

Thus giving the following theorem for a lower bound in WST assumption models.

Theorem 1. *There exists a model that satisfies WST for which any algorithm requires $\Omega(n^2)$ comparisons to find a $\frac{1}{4}$-maximum with probability $\geq \frac{7}{8}$ [7]*

Theorem 1 proves that with probability $\geq 1 - \delta$, no algorithm uses $\Omega(n)$ comparisons for finding the left most piece in the jigsaw puzzle [7]. Thus, answering Question 2 that there is **no** linearly complex PAC maxing algorithm possible for models with only WST property. The inclusion of STI property does not impact the quadratic complexity for maxing in WST models [7].

After defining a lower bound on quadratic sample complexity, Falahatgar et al. [7] proposed a trivial algorithm called BRUTE-FORCE algorithm to provide an upper bound on the sample complexity for PAC maxing for WST models. BRUTE-FORCE guarantees to estimate all pairwise probabilities to ϵ precision by using $\mathcal{O}(\frac{n^2}{\epsilon^2} \log \frac{n}{\delta})$ comparisons with probability $\geq 1 - \delta$ [7]. This algorithm's correctness is based on Hoeffding's inequality and union bound [7]. Hoeffding's inequality bounds the approximations of pairwise comparison for a single pair (i, j) to ϵ and using union bound, BRUTE-FORCE approximates **all** pairwise probabilities to ϵ with probability $1 - \delta$ [7].

5 For MST models with and without STI

Note: All results stated in this section are due to [7].

After seeing that maxing under the WST assumption requires a quadratic sample complexity, we move our focus on PAC maxing for MST models, which is less general than WST but more general than SST. Falahatgar et al. [7] proposed their algorithm OPT-MAX for the maxing problem in MST models with and without STI property. OPT-MAX uses $\mathcal{O}(\frac{n}{\epsilon^2} \log \frac{1}{\delta})$ comparisons and outputs an ϵ-maximal arm with probability $\geq 1 - \delta$, for $\delta \geq \min(\frac{1}{n}, e^{-n^{1/4}})$ [7]. This answers Question 3 that, there **is** a model, called MST, which is more general than SST and less general than WST for which a linear complexity PAC maxing is possible [7]. This makes MST the "most general" model **known** for which maxing is possible linearly[7, Section 1.1]. Similar to WST models, STI property's inclusion does not impact complexity for maxing in MST models [7].

Before showing the correctness of OPT-MAX, let us look into some sub-optimal algorithm which builds OPT-MAX. Falahatgar et al. [7] introduced a COMPARE algorithm to find pairwise comparison feedback between a pair of arms (i, j). Based on this feedback, the SOFT-SEQ-ELIM algorithm, which is the primary building block of OPT-MAX, decides which arm is the winner. COMPARE guarantees that it returns an expected relationship with probability $\geq 1 - \delta$. This theoretical guarantee is based on Hoeffding's inequality and union bound. The Hoeffding's inequality guarantees that COMPARE gives expected results for a pair of arms (i, j) at a time step t with probability $\geq 1 - \delta$ and union bound expands this guarantee over the total number of times arms i and j are compared [7].

The two sub-optimal algorithms, namely SOFT-SEQ-ELIM and NEAR-OPT-MAX, act as a backbone of OPT-MAX. SOFT-SEQ-ELIM is the primary building block of OPT-MAX to locate an ϵ-maximum arm [7]. It uses quadratic number of comparisons to find an $\epsilon-$maximum with probability $\geq 1-\delta$ [7]. This algorithm's efficiency is highly dependent on the selection of the starting anchor arm r with which all the other competing arms in the set N are compared to [7]. Suppose a good anchor arm is selected initially. In that case, the number of changes to the anchor arm decreases, hence, lowering the sample complexity with respect to the number of comparisons in each round. An arm a is an (ϵ, m)-good anchor arm if $|e| \leq m$ for every $e \in N$ such that $\tilde{p}_{e,a} > \epsilon$ [7]. In other words, there are at most m arms to which arm a is not ϵ-preferable to in the set N [7]. SOFT-SEQ-ELIM guarantees to find an ϵ_u-maximum of set N under linear number of comparisons with probability $\geq 1 - \delta$ if the starting anchor arm r is (ϵ_l, m)-good anchor arm, where ϵ_l and $\epsilon_u > \epsilon_l$ are lower and upper bias, respectively [7]. The correctness of this algorithm is based on the exploitation of MST property: "Under MST, if $\epsilon > 0$, $\tilde{p}_{i,j} \leq \epsilon, \tilde{p}_{k,j} > \epsilon$ then $\tilde{p}_{i,k} \leq \epsilon$" [7, Lemma 16].

In the worst case, SOFT-SEQ-ELIM has a quadratic sample complexity[7]. To avoid this scenario, the NEAR-OPT-MAX algorithm is used. The algorithm first finds a good anchor arm and then uses SOFT-SEQ-ELIM guaranteeing $\mathcal{O}(\frac{N}{\epsilon^2}(\log \frac{N}{\delta})^2)$ comparisons for finding an ϵ-maximum arm with probability $\geq 1 - \delta$.

For different ranges of confidence δ, Falahatgar et al. [7] proposes three distinct maxing algorithms namely OPT-MAX-HIGH, OPT-MAX-MEDIUM and OPT-MAX-LOW.

- OPT-MAX-LOW is used when $\min(e^{-n^{1/4}}, \frac{1}{n}) \leq \delta \leq \frac{1}{n^{1/3}}$,
- OPT-MAX-MEDIUM is used when $\frac{1}{n^{1/3}} \leq \delta \leq \frac{1}{\log n}$.
- OPT-MAX-HIGH is used when $\frac{1}{\log n} \leq \delta$.

The reason behind this segregation is if SOFT-SEQ-ELIM is used with a strong anchor, a linear sample complexity is assured for low levels of confidence $\delta \leq \frac{1}{n^{1/3}}$ [7]. However, this is not true for higher levels of δ. The authors solved this problem by using a smaller set from set N. The new set includes all the good anchor arms and then by using SOFT-SEQ-ELIM with a good anchor arm, guarantees a linear sample complexity to find an ϵ-maximum arm with probability $\geq 1 - \delta$ [7]. These three algorithms guarantee to use $\mathcal{O}(\frac{n}{\epsilon^2} \log \frac{1}{\delta})$ comparisons and outputs an ϵ-maximal arm with probability $\geq 1 - \delta$ [7]. The correctness of these three algorithms is based on union bound, which binds the number of comparisons used in all recursive calls in each algorithm [7]. These three algorithms are finally merged into OPT-MAX.

6 Conclusion

Revisiting all the Sections, we come to know that the models get more restrictive as we move from more general (WST) to less general (SST) transitivity [7]. Models under MST assumption are the most general **known** models for which linearly complex PAC maxing algorithms are possible to find an ϵ-maximum arm with probability $\geq 1 - \delta$ [7]. The upper bound on the maxing under MST is the **strongest** bound among SST, γ-RST and MST [7]. The correctness of most of the algorithms to find an ϵ-maximum arm from a set N with probability $\geq 1 - \delta$ is based on Hoeffding's inequality and union bound [5,6,7].

References

1. `http://gifgif.media.mit.edu/`.
2. Yisong Yue, Josef Broder, Robert Kleinberg, and Thorsten Joachims. The k-armed dueling bandits problem. *Journal of Computer and System Sciences*, 78(5):1538–1556, 2012.
3. Viktor Bengs, Robert Busa-Fekete, Eyke Hüllermeier, and Adil El Mesaoudi-Paul. Preference-based online learning with dueling bandits: A survey. Under review.
4. Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Pac bounds for multi-armed bandit and markov decision processes. In *International Conference on Computational Learning Theory*, pages 255–270. Springer, 2002.
5. Moein Falahatgar, Alon Orlitsky, Venkatadheeraj Pichapati, and Ananda Theertha Suresh. Maximum selection and ranking under noisy comparisons. *arXiv preprint arXiv:1705.05366*, 2017.
6. Moein Falahatgar, Yi Hao, Alon Orlitsky, Venkatadheeraj Pichapati, and Vaishakh Ravindrakumar. Maxing and ranking with few assumptions. *Advances in Neural Information Processing Systems*, 30:7060–7070, 2017.
7. Moein Falahatgar, Ayush Jain, Alon Orlitsky, Venkatadheeraj Pichapati, and Vaishakh Ravindrakumar. The limits of maxing, ranking, and preference learning. In *International Conference on Machine Learning*, pages 1427–1436, 2018.
8. Yisong Yue and Thorsten Joachims. Beat the mean bandit. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 241–248, 2011.
9. Yuan Zhou, Xi Chen, and Jian Li. Optimal pac multiple arm identification with applications to crowdsourcing. In *International Conference on Machine Learning*, pages 217–225, 2014.