Oracle-Augmented Prophet Inequalities

Sariel Har-Peled
University of Illinois Urbana-Champaign, USA

Elfarouk Harb
University of Illinois Urbana-Champaign, USA

Vasilis Livanos
University of Chile, Chile

Abstract

In the classical prophet inequality setting, a gambler is given n random variables $X_1, \ldots, X_n$ taken from known distributions, observes their realizations in this (potentially adversarial) order, and select one of them, immediately after it is being observed, so that its value is as high as possible. The classical *prophet inequality* shows a strategy that guarantees a value at least half of that an omniscience prophet that picks the maximum, and this ratio is optimal.

Here, we generalize the prophet inequality, allowing the gambler some additional information about the future that is otherwise privy only to the prophet. Specifically, at any point in the process, the gambler is allowed to query an oracle $\mathcal{O}$. The oracle $\mathcal{O}$ responds with a single bit answer: YES if the current realization is the largest of the remaining realizations, and NO otherwise. We show that the oracle model with m oracle calls is equivalent to the Top-1-of-$(m+1)$ model when the objective is maximizing the probability of selecting the maximum. This equivalence fails to hold when the objective is maximizing the competitive ratio, but we still show that any algorithm for the oracle model implies an equivalent competitive ratio for the Top-1-of-$(m+1)$ model.

We completely resolve the oracle model for any m, giving a tight lower and upper bound on the best possible competitive ratio. As a consequence, we provide new results as well as improvements on known results for the Top-1-of-m model.

2012 ACM Subject Classification Theory of computation → Algorithmic mechanism design

Keywords and phrases prophet inequalities, predictions, top-1-of-k model

Digital Object Identifier 10.4230/LIPIcs.ICALP.2024.23

1 Introduction

The field of optimal stopping theory concerns optimization settings where one makes decisions in a sequential manner, given imperfect information about the future, in a bid to maximize a reward or minimize a cost. A canonical setting in this area is the *prophet inequality* [24, 25]. In the classical prophet inequality setting, a gambler is presented with rewards $X_1, \ldots, X_n$, one after the other, drawn independently from known distributions. Upon seeing a reward X_i, the gambler must immediately make an irrevocable decision to either accept X_i, in which case the process ends, or to reject X_i and continue, losing the option to select X_i in the future. The goal of the gambler is to maximize the selected reward comparing against a *prophet* who knows all realizations in advance and selects the maximum realized reward. Throughout, we assume, without loss of generality, that $X_1, \ldots, X_n$ are continuous random variables.

The performance of the gambler can be measured in terms of several objectives. A common metric used in the literature is the *competitive ratio*, which is also known as the *Ratio of Expectations (RoE)* - see Definition 1.1. Another common distinction is between the case in which the given distributions are different and the case in which they are identical. For the former, Krengel, Sucheston and Garling [24, 25] showed an optimal strategy that is $1/2$-competitive. In this setting, the optimal competitive ratio can be achieved by simple, single-threshold algorithms [30, 23]. For I.I.D. random variables, Hill and Kertz [21] initially gave a $(1 - 1/e)$-competitive algorithm. This was improved to ≈ 0.738 [1] and later ≈ 0.745 [10], which is tight, due to a matching upper bound [21, 22].

Another relevant metric, introduced by Gilbert and Mosteller [19] for I.I.D. random variables, is that of maximizing the *Probability of selecting the Maximum realization (PbM)* - see Definition 1.2. For this objective and I.I.D. random variables, Gilbert and Mosteller [19] gave an algorithm that achieves a probability of ≈ 0.58, which is the best possible. Later, Esfandiari, Hajiaghayi, Lucier and Mitzenmacher [15] studied the same objective for general random variables, obtaining a tight probability equal to $1/e$ when the random variables arrive in adversarial order and 0.517 when the random variables arrive in random order. The latter case was recently improved to the tight ≈ 0.58 by Nuti [29], showing that the I.I.D. setting is not easier than the non-I.I.D. setting with random order. In this paper, we introduce a new model as a means to study variations of both the I.I.D. and the general settings, for both the *RoE* and *PbM* objectives.

A setting that is very related to ours is the TOP-1-OF-m model, formally introduced by Assaf and Samuel-Cahn [5] for I.I.D. random variables, although it had been studied initially by Gilbert and Mosteller [19]. In this setting, the algorithm is allowed to select $m \geq 1$ values, but the value it gets judged by is the maximum selected value. Gilbert and Mosteller [19] gave numerical approximations of the PbM objective for $2 \leq m \leq 10$, using a simple, single-threshold algorithm. Later, Assaf and Samuel-Cahn [5] studied the *RoE* objective for general distributions and gave a very elegant and simple $(1 - 1/m+1)$-competitive algorithm. The same authors, along with Larry Goldstein, later improved this in [4], bounding the competitive ratio of the optimal algorithm by a recursive differential equation. They gave numerical approximations for $2 \leq m \leq 5$, but studying the asymptotic nature of the constants for large m turned out to be difficult. Ezra, Feldman, and Nehama [17] later revisited the problem and gave a new algorithm for large m that is $1 - \mathrm{O}\left(e^{-m/6}\right)$-competitive for the same problem. This improves the error term from [4] from linear in m to exponential in m. Harb [20] recently improved this into a $1 - e^{-mW_0\left(\frac{\sqrt[m]{m!}}{m}\right)}$-competitive algorithm, where W_0 is the Lambert-W function[1], and improved the lower bound for $m = 2$ separately. However, the asymptotic nature of this function is difficult to analyze.

Model.

We introduce a new model that generalizes the standard prophet inequality setting, and analyze it as a means to obtain new results and improvements in the TOP-1-OF-m model. Our model allows the algorithm some information about the future that is otherwise privy only to the prophet. Specifically, at any point in the process, upon seeing a reward X_i, the algorithm is allowed to query an oracle $\mathcal{O}$. The oracle $\mathcal{O}$ responds with a single bit answer: YES if the current realization is the largest of the remaining realizations, i.e $X_i \geq \max_{j=i+1}^{n} X_j$ and NO otherwise. In other words, the oracle $\mathcal{O}$ informs the algorithm on whether the latter should select X_i, or reject it because there is a higher reward coming in the future. Clearly, with no queries available, one recovers the classical prophet inequality setting, whereas with $n-1$ queries, the strategy of using a query on every X_i, for $i = 1, \ldots, n-1$, leads to the algorithm selecting the highest realization always. Thus, this model interpolates nicely between the two extremes of full information and no information about the future.

[1] The Lambert-W function is $W_0(x)$ defined as the solution y to the equation $ye^y = x$.

We use Z to denote $\max\{X_1, \ldots, X_n\}$. Before we present our results, we define the different settings and objectives we consider in this paper.

▶ **Definition 1.1.** *The competitive ratio or* Ratio of Expectations *is denoted by RoE. An algorithm ALG is* α-competitive*, for* $\alpha \in [0,1]$*, if* $\mathbb{E}[ALG] \geq \alpha \cdot \mathbb{E}[\max_i X_i]$*, and* α *is called the* competitive ratio.

▶ **Definition 1.2.** *The* Probability of selecting the Maximum *realization is denoted by PbM. An algorithm ALG achieves a PbM of* α *if it returns a value* v *such that* $\Pr[v = Z] \geq \alpha$.

▶ **Definition 1.3.** *We use the term I.I.D to refer to the setting where* $X_1, \ldots, X_n$ *are independent and identically distributed random variables. We use* NON-I.I.D. *to refer to the more general setting where* $X_1, \ldots, X_n$ *are independent, but not necessarily identical.*

▶ **Definition 1.4.** *We use* PROPH_m *to refer to the* TOP-1-OF-m *model, in which the algorithm is allowed to choose up to* m *values, and its payoff is the maximum of the chosen values. We use* $\mathcal{O}_m$ *refers to our oracle model where the algorithm has access to* m *oracle calls, and can only select one value.*

Note that it makes sense to compare the model PROPH_{m+1} to $\mathcal{O}_m$ since in the former, the algorithm can choose $m+1$ values, where as the later can ask the oracle m times, and then choose an item. To help distinguish between the different settings, we denote each model as $\mathcal{M}(x, y, z)$, where

- x is either PROPH_m or $\mathcal{O}_m$ with $m \in \mathbb{N}$,
- y is either I.I.D or NON-I.I.D. and
- z is either *PbM* or *RoE*.

Motivation.

Our oracle model is loosely motivated by the idea of enhancing algorithms via the use of machine-learned *predictions*, in order to go beyond worst-case analysis [18, 2, 6, 3, 26]. This idea of using learning to improve the performance of algorithms has received significant attention recently, for example in designing auctions to maximize revenue [8, 27] or in matching problems [13, 31]. For more information on this line of work see the survey of Mitzenmacher and Vassilvitskii [28]. In real-world applications such as posted-pricing mechanisms for auctions, machine-learning models can capture behavioral patterns of buyers and accurately predict their future actions. This allows them to provide highly accurate predictions on future realizations in repeated prophet inequality settings, which makes studying prediction-enhanced models of prophet inequalities significantly important. While our model is less realistic since the predictions are always assumed to be correct, our analysis provides a theoretical upper bound on the performance of prediction-enhanced algorithms and our algorithms can serve as inspiration for settings where the predictions can also be inaccurate.

1.1 Our Contributions

In this paper, we study the oracle model for independent random variables following identical or general distributions with the PbM and RoE objectives and make the following contributions:

- We establish an *equivalence* between the oracle model and the TOP-1-OF-m model for the PbM objective.

- We show that this equivalence fails to hold for the RoE objective and that the best-possible competitive ratios in the two settings are quite separated. However, we show that guarantees for RoE in the oracle model translate to guarantees in the TOP-1-OF-m model, thus further motivating our study of the oracle model.
- We resolve the oracle model $\mathcal{M}(\mathcal{O}_m, \text{NON-I.I.D.}, RoE)$ by presenting an optimal single-threshold algorithm. Our algorithm achieves a competitive ratio of $1 - e^{-\xi_m}$ for general m, where ξ_m is the unique *positive* solution[2] to the equation $1 - e^{-\xi_m} = \frac{\Gamma(m+1,\xi_m)}{m!}$[3]. Furthermore, we prove that this lower bound is optimal by showing a construction that yields an equal upper bound. Since we showed that guarantees for $\mathcal{M}(\mathcal{O}_m, \text{NON-I.I.D.}, RoE)$ also hold for $\mathcal{M}(\text{PROPH}_{m+1}, \text{NON-I.I.D.}, RoE)$, our algorithm improves upon the latter setting's current best-known bounds of [20], even though the guarantees are obtained in the weaker oracle model.
- We give a single-threshold algorithm for the oracle model and the PbM objective $\mathcal{M}(\mathcal{O}_m, \text{I.I.D}, PbM)$ that achieves a $1 - O\left(m^{-m/5}\right)$ probability of selecting the maximum, as well as providing an upper bound that is asymptotically (almost) tight. To the best of our knowledge, this is the first result for the PbM objective and general m in the well studied TOP-1-OF-m model. Our algorithm achieves a probability of ≈ 0.797 even with $m = 1$ calls to the oracle, a significant improvement on the ≈ 0.58 achieved without oracle calls [19].

As discussed earlier, the main motivation behind our oracle model comes from our first two results which relate it to the TOP-1-OF-m model.

▶ **Theorem 1.5.** *The $\mathcal{M}(\mathcal{O}_m, y, PbM)$ model is equivalent to the $\mathcal{M}(\text{PROPH}_{m+1}, y, PbM)$ model, where $y = \text{I.I.D}$ or NON-I.I.D. In other words, for every prophet inequality instance, the probability achieved by the best-possible algorithm in the $\mathcal{M}(\mathcal{O}_m, y, PbM)$ model is the same as the one achieved by the best-possible algorithm in the $\mathcal{M}(\text{PROPH}_{m+1}, y, PbM)$ model.*

Theorem 1.5 is perhaps not that surprising due to the apparent similarity of the two models. However, thinking about the TOP-1-OF-m setting from the viewpoint of oracle calls allows for a different perspective that we exploit in our analysis. Perhaps more surprisingly, our oracle model and the TOP-1-OF-m model stop being equivalent when one considers the RoE objective; as we show in our second result, the oracle model is strictly weaker.

▶ **Theorem 1.6.** *There exists a prophet inequality instance and an algorithm $\mathcal{A}$ for an $\mathcal{M}(\text{PROPH}_{m+1}, \text{NON-I.I.D.}, RoE)$ instance for which no algorithm of $\mathcal{M}(\mathcal{O}_m, \text{NON-I.I.D.}, RoE)$ can achieve the same competitive ratio as that of $\mathcal{A}$.*

However, for every instance of $\mathcal{M}(\mathcal{O}_m, y, RoE)$ where $y = \text{I.I.D}$ or NON-I.I.D., there exists an algorithm $\mathcal{A}$ for the same instance of $\mathcal{M}(\text{PROPH}_{m+1}, y, RoE)$ that achieves a competitive ratio that is at least as good as that of the optimal algorithm for $\mathcal{M}(\mathcal{O}_m, y, RoE)$.

After establishing the relationship between our oracle model and the TOP-1-OF-m model, we turn our attention to upper and lower bounds for the oracle model. First, for the NON-I.I.D. setting and the RoE objective, we present an extremely simple single-threshold algorithm achieving a competitive ratio that approaches 1 exponentially in m. Even though our algorithm is for the oracle model, for which weaker guarantees are expected due to Theorem 1.6, it improves upon the best-known guarantee for the TOP-1-OF-m setting, due

---

[2] In Section 3, we prove that there is indeed a unique positive solution.
[3] $\Gamma(n, x) = \int_x^\infty t^{n-1} e^{-t}\, dt$ denotes the upper incomplete gamma function

to [20]. Our algorithm relies on two techniques; sharding and Poissonization, introduced by [20] for the analysis of threshold-based algorithms for prophet inequalities. We also appeal to stochastic dominance for the analysis. As an added benefit, the algorithm's analysis is very simple to understand.

▶ **Theorem 1.7.** *For every $m \geq 1$, let ξ_m denote the unique positive solution to $1 - e^{-\xi_m} = \frac{\Gamma(m+1,\xi_m)}{m!}$, where $\Gamma(n,x) = \int_x^\infty t^{n-1}e^{-t}\,dt$ denotes the upper incomplete gamma function. For every instance of the oracle model $\mathcal{M}(\mathcal{O}_m, \text{NON-I.I.D.}, RoE)$, there exists an algorithm that achieves a competitive ratio at least $1 - e^{-\xi_m}$. As $m \to +\infty$, this behaves as $1 - e^{-m/e + o(m)}$.*

In addition, we provide a construction for every m that gives a matching upper bound to the competitive ratio, thus completely resolving the problem for the case of general distributions and the RoE objective. The construction we have is perhaps of independent interest in the design of counterexamples for other settings, as it combines and generalizes standard counterexamples of prophet inequalities.

▶ **Theorem 1.8.** *For every $m \geq 1$, let ξ_m denote the unique positive solution to $1 - e^{-\xi_m} = \frac{\Gamma(m+1,\xi_m)}{m!}$, where $\Gamma(k,x) = \int_x^\infty t^{k-1}e^{-t}\,dt$ denotes the upper incomplete gamma function. For every $\delta > 0$, there exists an instance of $\mathcal{M}(\mathcal{O}_m, \text{NON-I.I.D.}, z)$, where $z =$ RoE or PbM, in which no algorithm can achieve a $\left(1 - e^{-\xi_m} + \delta\right)$-competitive ratio or select the maximum realization with probability $\left(1 - e^{-\xi_m} + \delta\right)$.*

We call the sequence $\{\xi_m\}_{m\geq 1}$ the *exponent sequence*, and analyze its properties and asymptotic behaviour to obtain our tight results. The idea behind why this is the right answer for the oracle model is the following: Intuitively, an algorithm for the oracle model performs poorly when, every time it expends an oracle call and gets a YES answer, the next value it sees that is above the queried value is only slightly larger, and thus the oracle call was expended without any real gain. The idea behind the worst-case for this setting is to have what is essentially a Poisson random variable with rate ξ_m, providing the algorithm with several non-zero values, each roughly the same. By carefully selecting ξ_m in order to equate the probability of having no non-zero values and the probability of having more than m non-zero values, we are forcing the algorithm to expend a query for every non-zero realization, thus rendering the oracle calls as useless as possible. This is the intuition behind the definition of ξ_m.

Next, we turn our attention to the I.I.D setting with m oracles calls and the PbM objective. We present a simple, single-threshold algorithm that selects the maximum realization with probability that approaches 1 in a super-exponential fashion. As a warm-up, we first present the analysis for $m = 1$ before generalizing it to all m.

▶ **Theorem 1.9.** *For sufficiently large m, n, and an instance of $\mathcal{M}(\mathcal{O}_m, \text{I.I.D}, PbM)$, there exists an algorithm that selects the maximum realization with probability at least $1 - O\left(m^{-m/5}\right)$.*

We also present an upper bound on the probability of success that is asymptotically tight, up to small multiplicative constants in the exponent. Because of Theorem 1.5, both upper and lower bounds on the probability of success carry over in the TOP-1-OF-m setting as well.

▶ **Theorem 1.10.** *There exists an instance of $\mathcal{M}(\mathcal{O}_m, \text{I.I.D}, PbM)$ for which no algorithm can select the maximum realization with probability greater than $1 - O\left(m^{-m}\right)$.*

Below is a table summarizing our results for the oracle model in the different settings.

Model	Lower Bound		Upper Bound	
	Previous Best	Current Best	Previous Best	Current Best
RoE, General Setting	$1 - \mathrm{O}\left(e^{-m/6}\right)$ [17]	$1 - e^{-m/e + \mathrm{o}(m)}$	–	$1 - e^{-m/e + \mathrm{o}(m)}$
PbM, I.I.D. Setting	≈ 0.58 [19]	≈ 0.797 $(m = 1)$ $1 - \mathrm{O}\left(m^{-m/5}\right)$	–	$1 - \mathrm{O}\left(m^{-m}\right)$

1.2 Additional Related Work

We have already mentioned the related work on algorithms with predictions, as well as the works of Gilbert and Mosteller [19], Esfandiari, Hajiaghayi, Lucier and Mitzenmacher [15] and Nuti [29] for the PbM objective. Related work includes the study of order-aware algorithms by Ezra, Feldman, Gravin and Tang [16], algorithms with fairness guarantees by Correa, Cristi, Dütting and Norouzi-Fard [9] and algorithms with a-priori information of some of the values by Correa, Cristi, Epstein and Soto [11]. In addition to these, Esfandiari, Hajiaghayi, Lucier and Mitzenmacher [15] study a related but distinct variant to ours. They relax the objective to allow the return of one out of the top k realizations, and show exponential upper and lower bounds. Their model, however, is incomparable to ours.

Organization

In Section 2 we relate our model to Top-1-of-m model of Assaf and Samuel-Cahn [5] and prove the reductions. In Section 3 we present our tight algorithm for the Non-I.I.D. setting. Section 4 contains our algorithms and upper bounds for the I.I.D setting. Due to space constraints, we present some background on concentration inequalities that we use for our results in Appendix A and several missing proofs in Appendix B.

2 Reductions

To motivate our oracle model, we start by establishing an equivalence between $\mathcal{M}(\mathcal{O}_m, y, PbM)$ and $\mathcal{M}(\text{Proph}_{m+1}, y, PbM)$, for both the $y =$ I.I.D and $y =$ Non-I.I.D. case (Theorem 1.5). We also show that, perhaps surprisingly, this equivalence does not hold for the RoE objective; guarantees for $\mathcal{M}(\mathcal{O}_m, y, RoE)$ translate to guarantees for $\mathcal{M}(\text{Proph}_{m+1}, y, RoE)$ (Theorem 1.6), but not the converse. Later, we will use this result to improve the best-known guarantees on $\mathcal{M}(\text{Proph}_{m+1}, y, RoE)$.

2.1 The PbM objective

▶ **Theorem 1.5.** *The $\mathcal{M}(\mathcal{O}_m, y, PbM)$ model is equivalent to the $\mathcal{M}(\text{Proph}_{m+1}, y, PbM)$ model, where $y =$ I.I.D or Non-I.I.D. In other words, for every prophet inequality instance, the probability achieved by the best-possible algorithm in the $\mathcal{M}(\mathcal{O}_m, y, PbM)$ model is the same as the one achieved by the best-possible algorithm in the $\mathcal{M}(\text{Proph}_{m+1}, y, PbM)$ model.*

Theorem 1.5 follows from Lemma 2.1 and Lemma 2.2.

▶ **Lemma 2.1.** *Fix an instance of $\mathcal{M}(\text{Proph}_{m+1}, y, PbM)$ where $y =$ I.I.D or Non-I.I.D. and let α denote the probability of selecting the maximum that an algorithm $\mathcal{A}$ for $\mathcal{M}(\mathcal{O}_m, y, PbM)$ achieves on this instance. Then, there exists an algorithm $\mathcal{B}$ for $\mathcal{M}(\text{Proph}_{m+1}, y, PbM)$ on this instance, with black-box access to $\mathcal{A}$ such that the probability that $\mathcal{B}$ selects the maximum realization is at least α.*

Proof. *Proof in Appendix B.1.* ◀

Next, we show that $\mathcal{M}(\mathcal{O}_m, y, PbM)$ can be reduced to $\mathcal{M}(\text{PROPH}_{m+1}, y, PbM)$.

▶ **Lemma 2.2.** *Fix an instance of $\mathcal{M}(\mathcal{O}_m, y, PbM)$ where $y =$ I.I.D or NON-I.I.D. and let α denote the probability of selecting the maximum that an algorithm $\mathcal{B}$ for $\mathcal{M}(\text{PROPH}_{m+1}, y, PbM)$ achieves on this instance. Then, there exists an algorithm $\mathcal{A}$ for $\mathcal{M}(\mathcal{O}_m, y, PbM)$ on this instance, with black-box access to $\mathcal{B}$ such that the probability that $\mathcal{A}$ selects the maximum realization is at least α.*

Proof. *Proof in Appendix B.2.* ◀

2.2 The RoE Objective

Given the apparent similarity of the two models, one may wonder whether the equivalence continues to hold even for the RoE objective. As we show in this section, this is not the case, but studying the oracle model for the RoE objective is still useful.

▶ **Theorem 1.6.** *There exists a prophet inequality instance and an algorithm $\mathcal{A}$ for an $\mathcal{M}(\text{PROPH}_{m+1}, \text{NON-I.I.D.}, RoE)$ instance for which no algorithm of $\mathcal{M}(\mathcal{O}_m, \text{NON-I.I.D.}, RoE)$ can achieve the same competitive ratio as that of $\mathcal{A}$.*

However, for every instance of $\mathcal{M}(\mathcal{O}_m, y, RoE)$ where $y =$ I.I.D or NON-I.I.D. , there exists an algorithm $\mathcal{A}$ for the same instance of $\mathcal{M}(\text{PROPH}_{m+1}, y, RoE)$ that achieves a competitive ratio that is at least as good as that of the optimal algorithm for $\mathcal{M}(\mathcal{O}_m, y, RoE)$.

We first present an example that shows the first part of the theorem.

▶ **Example 2.3.** For a fixed $\varepsilon > 0$ and $m = 1$, consider the following instance:

$$X_1 = 1 \quad \text{w.p. } 1, \quad X_2 = \begin{cases} 1+\varepsilon & \text{w.p. } \frac{1}{2} - \varepsilon \\ 0 & \text{w.p. } \frac{1}{2} + \varepsilon \end{cases}, \quad X_3 = \begin{cases} \frac{1}{\varepsilon} & \text{w.p. } \varepsilon \\ 0 & \text{w.p. } 1-\varepsilon \end{cases}.$$

First, observe that

$$\mathbb{E}[\max\{X_1, X_2, X_3\}] = \frac{1}{\varepsilon} \cdot \varepsilon + (1+\varepsilon)(1-\varepsilon)\left(\frac{1}{2} - \varepsilon\right) + 1 \cdot (1-\varepsilon)\left(\frac{1}{2} + \varepsilon\right).$$

Notice that, for small ε, an algorithm $\mathcal{B}$ that is optimal for the $Proph_2$ model in this instance is to select X_1, ignore X_2 and then select X_3 if it is non-zero. This yields

$$\mathbb{E}[\mathcal{B}] = 1 \cdot (1-\varepsilon) + \frac{1}{\varepsilon} \cdot \varepsilon.$$

However, the optimal $\mathcal{A}$ will query $\mathcal{O}$ at X_1. With probability $(1-\varepsilon)(1/2+\varepsilon)$, it will stop and select X_1, getting a value of 1. Otherwise, it will continue, with no oracle calls left. It will ignore X_2 and select X_3. Thus,

$$\mathbb{E}[\mathcal{A}] = 1 \cdot \left(\frac{1}{2} + \varepsilon\right)(1-\varepsilon) + \frac{1}{\varepsilon} \cdot \varepsilon.$$

The competitive ratios of $\mathcal{A}$ and $\mathcal{B}$ respectively are

$$RoE_{\mathcal{A}} = \frac{\left(\frac{1}{2}+\varepsilon\right)(1-\varepsilon) + \frac{1}{\varepsilon}\cdot\varepsilon}{\frac{1}{\varepsilon}\cdot\varepsilon + (1+\varepsilon)(1-\varepsilon)\left(\frac{1}{2}-\varepsilon\right) + 1\cdot(1-\varepsilon)\left(\frac{1}{2}+\varepsilon\right)}$$

and

$$RoE_{\mathcal{B}} = \frac{(1-\varepsilon) + \frac{1}{\varepsilon}\cdot\varepsilon}{\frac{1}{\varepsilon}\cdot\varepsilon + (1+\varepsilon)(1-\varepsilon)\left(\frac{1}{2}-\varepsilon\right) + 1\cdot(1-\varepsilon)\left(\frac{1}{2}+\varepsilon\right)},$$

and thus, as $\varepsilon \to 0$, we get

$$RoE_{\mathcal{A}} \to \frac{3/2}{2} = \frac{3}{4}, \qquad \text{and} \qquad RoE_{\mathcal{B}} \to \frac{2}{2} = 1.$$

The above example, appropriately generalized for $m > 1$ by having random variables

$$X_1 = 1 \quad \text{w.p. } 1, \quad X_i = \begin{cases} 1 + (i-1)\varepsilon & \text{w.p. } \frac{1}{2} - \varepsilon \\ 0 & \text{w.p. } \frac{1}{2} + \varepsilon \end{cases}, \quad \text{for } i = 2, \dots, m+1, \text{ and}$$

$$X_{m+2} = \begin{cases} \frac{1}{\varepsilon} & \text{w.p. } \varepsilon \\ 0 & \text{w.p. } 1 - \varepsilon \end{cases},$$

yields the following corollary.

▶ **Corollary 2.4.** *For every $m \geq 1$, there exists an instance such that*

$$\frac{\mathcal{M}(\mathcal{O}_m, \text{NON-I.I.D.}\ , RoE)}{\mathcal{M}(Proph_{m+1}, \text{NON-I.I.D.}\ , RoE)} \leq 1 - \frac{1}{2^{m+1}}.$$

The analysis of this example for general m follows immediately from the $m = 1$ case. We do not present it here as, even though this example is very simple, this gap is not the tightest one possible. For a tighter gap between the competitive ratio of the two models, see the example in the proof of Theorem 1.8.

Next, we present the proof of the second part of Theorem 1.6, showing that an algorithm for $\mathcal{M}(\text{PROPH}_{m+1}, y, RoE)$ that has access to an algorithm for $\mathcal{M}(\mathcal{O}_m, y, RoE)$ can always do at least as well. The theorem follows from Lemma 2.5, whose proof is essentially the same as the proof of Lemma 2.1.

▶ **Lemma 2.5.** *Fix an instance of $\mathcal{M}(\text{PROPH}_{m+1}, y, PbM)$ where $y =$ I.I.D or NON-I.I.D. , and let α denote the competitive ratio that an algorithm $\mathcal{A}$ for $\mathcal{M}(\mathcal{O}_m, y, PbM)$ achieves on this instance. Then, there exists an algorithm $\mathcal{B}$ for $\mathcal{M}(\text{PROPH}_{m+1}, y, PbM)$ on this instance, with black-box access to $\mathcal{A}$, that achieves competitive ratio at least α.*

Proof. Again, the idea is that $\mathcal{B}$ can simulate $\mathcal{A}$'s behaviour by selecting each realization that $\mathcal{A}$ decides to query. Initially, $\mathcal{B}$ starts with an empty set S of selected values. Whenever $\mathcal{B}$ is presented with a realization X_i, it feeds it to $\mathcal{A}$. If $\mathcal{A}$ decides to select X_i or expend a query for X_i, regardless of the outcome of the query, $\mathcal{B}$ always selects X_i into S, otherwise $\mathcal{B}$ decides not to select X_i. By induction, S contains exactly all the realizations that were queried by $\mathcal{A}$ as well as at most one more realization that might have been selected by $\mathcal{A}$ if it run out of queries. Therefore, $|S| \leq m + 1$.

Now, notice that, for every possible sequence of realizations, whatever value $\mathcal{A}$ has selected is also in S. Therefore, if $V_{\mathcal{A}}$ is the value selected by $\mathcal{A}$ and $V_{\mathcal{B}}$ is the value selected by $\mathcal{B}$, we have $\mathbb{E}[V_{\mathcal{B}}] \geq \mathbb{E}[V_{\mathcal{A}}]$, and thus the competitive ratio of $\mathcal{B}$ is at least α. ◀

3 The Non-I.I.D. Setting

After describing how the oracle model is related to the TOP-1-OF-m model, we continue by providing algorithms for the oracle model. Using the reduction of Theorem 1.6, any guarantees we provide for the oracle model with the RoE objective can be directly translated to guarantees for the Top-1-of-m model, improving upon the previous work on this model [5, 4, 17, 20]. We provide a simple, single-threshold algorithm that completely resolves our oracle model for the RoE objective and along the way improves upon the current best-known competitive ratio in the Top-1-of-m model.

3.1 The Exponent Sequence

Before we describe our algorithm, we introduce a sequence that is crucial in the analysis of our algorithm as well as the matching upper bound.

▶ **Definition 3.1.** *For every $m \geq 1$, let ξ_m denote the unique positive solution to the following equation:*

$$1 - e^{-\xi_m} = \frac{\Gamma(m+1, \xi_m)}{m!},$$

where $\Gamma(n,x) = \int_x^\infty t^{n-1} e^{-t}\, dt$ denotes the upper incomplete gamma function. We call $\{\xi_m\}_{m \geq 1}$ the exponent sequence.

The exponent sequence is important since, as we show later, the optimal competitive ratio of $\mathcal{M}(\mathcal{O}_m, \text{Non-I.I.D.}, RoE)$ is exactly $1 - e^{-\xi_m}$. It is also easy to see that it is an increasing sequence in m. Next, we present some useful lemmas about the exponent sequence and its asymptotic behaviour.

▶ **Lemma 3.2.** *We have*

$$\lim_{m \to \infty} \frac{\xi_m}{m} = \frac{1}{e}.$$

Proof. *Proof in Appendix B.3.* ◀

▶ **Lemma 3.3.** *For all $m \geq 1, (m!)^{1/m} \leq \xi_m \leq ((m+1)!)^{1/m+1}$.*

Proof. *Proof in Appendix B.4.* ◀

▶ **Lemma 3.4.** *Let $k, m \geq 0$ be integers. Define*

$$f(k,m) = \sum_{j=1}^{k} \frac{\xi_m^j}{j!} - \sum_{j=m+1}^{m+k} \frac{\xi_m^j}{j!}$$

Then $f(k,m) \geq 0$ for all k, m.

Proof. *Proof in Appendix B.5.* ◀

3.2 Sharding, Poissonization, and Stochastic Dominance

For the lower bound, we will use the Poissonization and sharding approach introduced by Harb in [20] that is useful in tackling lower bounds for prophet inequalities. In this technique, given random variables $X_1, \ldots, X_n$ with cdfs $F_1, \ldots, F_n$, instead of sampling X_i from F_i, we instead imagine sampling K *shards* $Y_1, \ldots, Y_K$ independently from the same distribution with cdf $F_i^{1/K}$ and returning $\max\{Y_1, \ldots, Y_K\}$. In this way, the distribution of $\max\{Y_1, \ldots, Y_K\}$ is F_i. Harb [20] calls this *sharding* the random variable X_i.

We will shard all the random variables $\{X_i\}_{1\le i\le n}$ into $\{Y_{i,j}\}_{\substack{1\le i\le n\\1\le j\le K}}$, and let τ be a threshold such that $\sum_{i=1}^{n}\sum_{j=1}^{K}\Pr[Y_{i,j}\ge\tau]=c$ for some constant c to be determined. We can rewrite this into the following.

$$\sum_{i=1}^{n} K\left(1-\Pr[X_i\le\tau]^{1/K}\right)=c \tag{1}$$

Taking the limit of Eq. (1) as $K\to+\infty$, we get $\sum_{i=1}^{n}-\log\Pr[X_i\le\tau]=c$, or equivalently $\Pr[Z\le\tau]=e^{-c}$ where $Z=\max\{X_1,...,X_n\}$. Hence, we retrieve maximum-based thresholds. Moreover, because $\Pr[Y_{i,j}\ge\tau]\le\frac{1}{K}\to 0$, then we can use a Poissonization argument to count the number of shards $Y_{i,j}\ge\tau$ using a Poisson distribution with rate c. For more details, see [20].

Next we discuss stochastic dominance. Given a threshold-based algorithm that uses a single threshold τ, how do we lower bound its competitive ratio? One standard idea is to use stochastic dominance, that we briefly present here. Recall that, for $Z=\max\{X_1,\ldots,X_n\}$,

$$\mathbb{E}[ALG]=\int_0^\infty \Pr[ALG\ge x]dx,\qquad \mathbb{E}[Z]=\int_0^\infty \Pr[Z\ge x]dx$$

If we can guarantee that there is $c_1\in[0,1]$ such that for all $\nu\in[0,\tau]$, $\Pr[ALG\ge\nu]\ge c_1\Pr[Z\ge\nu]$, and that there is a $c_2\in[0,1]$ such that for all $\nu\in[\tau,+\infty]$, $\Pr[ALG\ge\nu]\ge c_2\Pr[Z\ge\nu]$, then we get the chain of inequalities

$$\begin{aligned}\mathbb{E}[ALG]&=\int_0^\tau \Pr[ALG\ge x]dx+\int_\tau^\infty \Pr[ALG\ge x]dx && (2)\\ &\ge c_1\int_0^\tau \Pr[Z\ge x]dx+c_2\int_\tau^\infty \Pr[Z\ge x]dx && (3)\\ &\ge \min(c_1,c_2)\,\mathbb{E}[Z] && (4)\end{aligned}$$

And hence $c=\min(c_1,c_2)$ would be a lower bound on the competitive ratio of ALG. This argument is used in several results on prophet inequalities and is often referred to as *majorizing* ALG with Z.

3.3 An Optimal Single-Threshold Algorithm

We are now ready to describe a single-threshold algorithm that achieves the optimal competitive ratio in our oracle model.

▶ **Theorem 1.7.** *For every $m\ge 1$, let ξ_m denote the unique positive solution to $1-e^{-\xi_m}=\frac{\Gamma(m+1,\xi_m)}{m!}$, where $\Gamma(n,x)=\int_x^\infty t^{n-1}e^{-t}\,dt$ denotes the upper incomplete gamma function. For every instance of the oracle model $\mathcal{M}(\mathcal{O}_m,$ NON-I.I.D. $, RoE)$, there exists an algorithm that achieves a competitive ratio at least $1-e^{-\xi_m}$. As $m\to+\infty$, this behaves as $1-e^{-m/e+o(m)}$.*

Proof. Let $Z=\max\{X_1,\ldots,X_n\}$. The algorithm ALG follows the sharding and Poissonization paradigm, setting τ to be the $e^{-\xi_m}$ quantile of the maximum, i.e. $\Pr[Z\le\tau]=e^{-\xi_m}$. It then greedily expends its oracle calls for every value above τ. For every oracle call, if the answer is YES, it stops and selects the current realization. Otherwise, it updates the querying threshold to be the current realization and continues. Finally, when it run out of oracle calls, it accepts the first value above the current threshold.

The analysis proceeds by stochastic dominance. Let $\beta\in[0,\tau]$. The probability the algorithm selects a value above β is equal to the probability it selects some value. Thus,

$$\Pr[ALG \geq \beta] \geq \Pr[Z \geq \tau] = 1 - e^{-\xi_m} \geq \left(1 - e^{-\xi_m}\right) \Pr[Z \geq \beta] \tag{5}$$

Next, consider $\beta \in [\tau, +\infty)$. Let $\Pr[Z \leq \beta] = e^{-q} > e^{-\xi_m}$, implying $\Pr[Z \geq \beta] = 1 - e^{-q}$. We lower bound $\Pr[ALG \geq \beta]$ via sharding each $\{X_i\}$. Consider how many shards have values in the range $[\tau, \beta]$. By Poissonization, the number of shards is a Poisson random variable with rate $\xi_m - q$. Consider the event of there being at most m shards with values in the range $[\tau, \beta]$, and there being at least one shard in $[\beta, +\infty)$; the algorithm must get a value at least β in that case. Hence,

$$\frac{\Pr[ALG \geq \beta]}{\Pr[Z \geq \beta]} \geq \frac{(1 - e^{-q}) \sum_{i=0}^{m} e^{-(\xi_m - q)} \frac{(\xi_m - q)^i}{i!}}{1 - e^{-q}} = \frac{\Gamma(m+1, \xi_m - q)}{m!} \tag{6}$$

Note that Eq. (6) is minimized for $q \to 0$, and hence

$$\Pr[ALG \geq \beta] \geq \frac{\Gamma(m+1, \xi_m)}{m!} \cdot \Pr[Z \geq \beta]. \tag{7}$$

Combining Eq. (5) and Eq. (7), the competitive ratio is at least $\min\left\{1 - e^{-\xi_m}, \frac{\Gamma(m+1,\xi_m)}{m!}\right\} = 1 - e^{-\xi_m}$ by the definition of ξ_m. ◀

3.4 A Tight Upper Bound

Next, we give an instance such that no algorithm for $\mathcal{M}(\mathcal{O}_m, \text{NON-I.I.D.}\ , RoE)$ can achieve a competitive ratio greater than $1 - e^{-\xi_m}$. The same counterexample works also for the PbM objective, as can easily be seen from its proof, giving us the same upper bound for the $\mathcal{M}(\mathcal{O}_m, \text{NON-I.I.D.}\ , PbM)$ setting. Our construction once again makes use of the exponential sequence – see Definition 3.1.

▶ **Theorem 1.8.** *For every $m \geq 1$, let ξ_m denote the unique positive solution to $1 - e^{-\xi_m} = \frac{\Gamma(m+1,\xi_m)}{m!}$, where $\Gamma(k, x) = \int_x^\infty t^{k-1} e^{-t}\, dt$ denotes the upper incomplete gamma function. For every $\delta > 0$, there exists an instance of $\mathcal{M}(\mathcal{O}_m, \text{NON-I.I.D.}\ , z)$, where $z = RoE$ or PbM, in which no algorithm can achieve a $\left(1 - e^{-\xi_m} + \delta\right)$-competitive ratio or select the maximum realization with probability $\left(1 - e^{-\xi_m} + \delta\right)$.*

Proof. Consider the following instance with $n + 2$ random variables for n large enough. Let $\varepsilon > 0$ be a small enough constant and

$$X_1 = 1 \quad \text{w.p. } 1, X_i = \begin{cases} 1 + \varepsilon \cdot (i - 1) & \text{w.p. } \xi_m/n \\ 0 & \text{w.p. } 1 - \xi_m/n \end{cases} \quad \text{for } i = 2, \dots, n+1, \text{ and}$$

$$X_{n+2} = \begin{cases} \frac{1}{\varepsilon} & \text{w.p. } \varepsilon \\ 0 & \text{w.p. } 1 - \varepsilon \end{cases}.$$

First notice that the prophet obtains a value of

$$\mathbb{E}\left[\max\left\{X_1, \dots, X_{n+2}\right\}\right] = \frac{1}{\varepsilon} \cdot \varepsilon + (1 - \varepsilon)\, \mathbb{E}\left[\max\left\{X_1, \dots, X_{n+1}\right\}\right],$$

where, for $\varepsilon \to 0$, the maximum of $X_1, \dots, X_{n+1}$ is always 1. Therefore, as $\varepsilon \to 0$, the prophet's expected value is 2.

Also notice that, for large n, $X_2, \ldots, X_{n+1}$ resemble a Poisson random variable with rate ξ_m (as can be seen by Lemma A.2). Therefore,

$$\Pr\left[\text{exactly } k \text{ out of } X_2, \ldots, X_{n+1} \neq 0\right] \to e^{-\xi_m} \frac{(\xi_m)^k}{k!}, \quad \text{as } n \to \infty.$$

Before we proceed, let

$$Q(k+1, x) = \frac{\Gamma(k+1, x)}{k!} = e^{-x} \sum_{j=0}^{k} \frac{x^j}{j!}$$

denote the regularized incomplete gamma function; $1 - Q(k+1, x)$ is also known as the tail of the Poisson distribution with rate x. Next, consider an algorithm for this setting. There are two cases: either the algorithm decides to query $\mathcal{O}$ at $X_1 = 1$ or not. The only differences are that (i) the former algorithm has $m-1$ queries available for $X_2, \ldots, X_{n+2}$, while the latter has m and (ii) the former algorithm gets an expected value of 2 when $X_2, \ldots, X_{n+1}$ are all 0 ($1/\varepsilon \cdot \varepsilon + (1-\varepsilon) \cdot 1$ as $\varepsilon \to 0$), whereas the latter gets an expected value of 1.

Given this, we analyze the performance of an algorithm $\mathcal{A}$ that has q queries available for $X_2, \ldots, X_{n+1}$, as if X_1 was not there. Then, we set q to be $m-1$ or m and compare the performance of both algorithms. $\mathcal{A}$ observes $X_2, \ldots, X_{n+1}$ and can decide to skip some number, say k, of the non-zero random variables among $X_2, \ldots, X_{n+1}$ that it sees, without expending an oracle call (k could be 0). What is the competitive ratio of $\mathcal{A}$? The only way for the algorithm to get a competitive ratio that does not approach 1 as $\varepsilon \to 0$ is if it expends an oracle call at a random variable X_i, and $X_{i+1}, \ldots, X_{n+1} = 0$. Then, its competitive ratio is $\varepsilon \cdot 1/\varepsilon + (1-\varepsilon) \cdot 1$, as $\varepsilon \to 0$; in all other cases it is 1. Thus, if we think of $X_2, \ldots, X_{n+1}$ as a Poisson random variable P, the algorithm gets q tries to "guess" the correct number of the realization of P, i.e. the number of non-zeroes of $X_2, \ldots, X_{n+1}$, and expend an oracle call at the last one.

Consider another algorithm $\mathcal{B}$ that expends a query for the i-th non-zero value of $X_2, \ldots, X_{n+1}$ if and only if the i-th term of the Poisson tail, i.e. $e^{-\xi_m} \frac{(\xi_m)^i}{i!}$ is in the highest q terms of the Poisson tail, and if it runs out of oracle calls, it accepts the first non-zero realization it sees afterwards. Every algorithm that decides to skip some non-zero values of $X_2, \ldots, X_{n+1}$ without expending an oracle call does at most as good as $\mathcal{B}$, since $\mathcal{B}$ expends the oracle calls at the terms of the Poisson tail that have the highest probability of being the correct number of non-zeroes of $X_2, \ldots, X_{n+1}$. Furthermore, since the pdf of the Poisson distribution is unimodal, we know that the highest q terms of the Poisson tail occur successively. Specifically, let $\ell + 1$ to $\ell + q$ denote the highest q terms of the Poisson tail. This implies that $\mathcal{A}$'s performance is maximized if the k terms it decides to skip are all in the beginning.

Next, we analyze the performance of $\mathcal{A}$, given that it skips the first k non-zero realizations of $X_2, \ldots, X_{n+1}$ and starts expending the q oracle calls it has on the $k+1$ non-zero realization of $X_2, \ldots, X_{n+1}$. $\mathcal{A}$ gets an expected value of 2 as $\varepsilon \to 0$ when the number of non-zero values of $X_2, \ldots, X_{n+1}$ is between $k+1$ and $k+q$, and an expected value of 1 otherwise, either because it saw too many non-zeros in $X_2, \ldots, X_{n+1}$ or too few. If $q = m$, then the expected value of such an algorithm is

$$\begin{aligned}
&1 \cdot Q(k+1, \xi_m) + 2 \cdot (Q(k+m+1, \xi_m) - Q(k+1, \xi_m)) + 1 \cdot (1 - Q(k+m+1, \xi_m)) \\
=&1 + Q(k+m+1, \xi_m) - Q(k+1, \xi_m),
\end{aligned}$$

whereas if $q = m - 1$, i.e. $\mathcal{A}$ expended an oracle call at X_1, then its expected value is 2 also when $X_2, \ldots, X_{n+1} = 0$, and thus the total expected value of such an algorithm is

$$\begin{aligned}
&2 \cdot Q(1, \xi_m) + 1 \cdot (Q(k+1, \xi_m) - Q(1, \xi_m)) + 2 \cdot (Q(k+m, \xi_m) - Q(k+1, \xi_m)) \\
&\qquad + 1 \cdot (1 - Q(k+m, \xi_m)) \\
&= 1 + Q(1, \xi_m) - Q(k+1, \xi_m) + Q(k+m, \xi_m).
\end{aligned}$$

Thus, the expected value of every algorithm ALG is upper bounded by

$$\begin{aligned}
\mathbb{E}[ALG] &\leq 1 - Q(k+1, \xi_m) + \max\{Q(k+m+1, \xi_m), Q(1, \xi_m) + Q(k+m, \xi_m)\} \\
&= 1 + Q(k+m, \xi_m) - Q(k+1, \xi_m) + e^{-\xi_m} \max\left\{ \frac{\xi_m^{k+m}}{(k+m!)}, 1 \right\}. \qquad (8)
\end{aligned}$$

By Lemma 3.3, for $k = 0$, we have $\frac{\xi_m^m}{m!} \geq 1$, whereas for $k > 0$, we have $\frac{\xi_m^{k+m}}{(k+m)!} \leq 1$.

Assume that $k = 0$. Then,

$$\mathbb{E}[ALG] \leq 1 + Q(m+1, \xi_m) - Q(1, \xi_m) = 1 + Q(m+1, \xi_m) - e^{-\xi_m}.$$

Recall, however, that $Q(m+1, \xi_m) = 1 - e^{-\xi_m}$ by the definition of the exponential sequence, and thus

$$\mathbb{E}[ALG] \leq 2\left(1 - e^{-\xi_m}\right).$$

Next, assume that $k > 0$. Then,

$$\begin{aligned}
\mathbb{E}[ALG] &\leq 1 + Q(k+m, \xi_m) - Q(k+1, \xi_m) + e^{-\xi_m} \\
&= 1 + Q(m+1, \xi_m) + e^{-\xi_m} \sum_{j=m+1}^{k+m-1} \frac{\xi_m^j}{j!} - e^{-\xi_m} \sum_{j=1}^{k} \frac{\xi_m^j}{j!} \\
&= 2 - e^{-\xi_m} - e^{-\xi_m} \frac{\xi_m^{k+m}}{(k+m)!} - e^{-\xi_m} \left(\sum_{j=1}^{k} \frac{\xi_m^j}{j!} - \sum_{j=m+1}^{m+k} \frac{\xi_m^j}{j!} \right) \\
&\leq 2 - e^{-\xi_m} \left(1 + \frac{\xi_m^{k+m}}{(k+m)!} \right) \\
&\leq 2 - 2e^{-\xi_m} \\
&= 2\left(1 - e^{-\xi_m}\right)
\end{aligned}$$

where the second equality follows from $Q(m+1, \xi_m) = 1 - e^{-\xi_m}$, the second inequality follows by Lemma 3.4 and the third inequality follows from Lemma 3.3. Therefore, the competitive ratio of every algorithm is

$$RoE \leq \frac{2\left(1 - e^{-\xi_m}\right)}{2} = 1 - e^{-\xi_m}.$$

◀

4 The I.I.D. Setting

Motivated by the early work of [19] for the TOP-1-OF-m model, in this section we study the I.I.D. setting and the PbM objective. As a warm-up, we take a look at the I.I.D setting with the PbM objective and the case of $m = 1$, providing a simple single-threshold algorithm.

4.1 A Single-Threshold Algorithm for $m = 1$

Our single-threshold algorithm $\mathcal{A}_p$ for $\mathcal{M}(\mathcal{O}_1, \text{I.I.D }, PbM)$ selects a threshold τ equal to the p-th quantile of the given distribution $\mathcal{D}$, for some $p \in [0,1]$. In other words, τ is set such that $p = \Pr[X_i \geq \tau]$. The first time the algorithm observes a realization above τ, it queries the oracle to see whether the realization should be selected or not. If it continues, it simply accepts the first value encountered above the observed realization on which it queried $\mathcal{O}$.

▶ **Lemma 4.1.** *There exists $p \in [0,1]$ such that $\mathcal{A}_p$ selects the maximum realization with probability at least 0.797 in the $\mathcal{M}(\mathcal{O}_1, I.I.D\ , PbM)$ model for large n.*

Proof. Let Y be the total number of realizations above τ, and $i_1 < i_2 < \cdots < i_Y$ be the indices of the random variables above τ, i.e. $X_{i_t} > \tau$, for $t = 1, \ldots, Y$. Furthermore, let r_t be the rank of X_{i_t} in $\mathcal{X} = \{X_{i_1}, \ldots, X_{i_Y}\}$, i.e. the number k such that X_{i_t} is the k-th largest number in $\mathcal{X}$, and Z be the maximum realization of $X_1, \ldots, X_n$.

X_{i_1} is the first realization we observe above τ. Notice that if $r_1 = 1$ or $r_1 = 2$ then the algorithm always selects the maximum realization Z. In other words, given that $Y = 1$ or $Y = 2$, the algorithm selects Z with probability 1. Consider the case $Y > 2$. Again, if $r_1 \leq 2$, the algorithm selects Z with probability 1. Otherwise, if $r_1 > 2$, the algorithm returns Z if and only if for all realizations above τ that appear after X_{i_1} and are also larger than X_{i_1}, the first to encounter is Z. In other words, for the algorithm to succeed in this case, it must be that among the $r_1 - 1$ values of rank smaller than r_1, the first one in the arrival order is the element of rank 1. Since the random variables are I.I.D , the probability of this event is exactly $1/r_1-1$.

Let j be the first index such that $X_{i_j} > X_{i_1}$, and $\alpha(Y) = \Pr\{\mathcal{A} \text{ selects } Z \mid Y\}$. Conditioned on $Y \geq 3$, the probability that the algorithm selects Z is

$$\begin{aligned}
\alpha(Y \mid Y \geq 3) &= \Pr[r_1 = 1] + \Pr[r_1 = 2] + \sum_{t=3}^{Y} \Pr[r_1 = t] \Pr\{r_j = 1 \mid r_1 = t\} \\
&= \frac{2}{Y} + \sum_{t=3}^{Y} \frac{\Pr[r_z = 1 \mid r_1 = t]}{Y} \\
&= \frac{1}{Y}\left(2 + \sum_{t=3}^{Y} \Pr[r_z = 1 \mid r_1 = t]\right) \\
&= \frac{1}{Y}\left(2 + \sum_{t=3}^{Y} \frac{1}{t-1}\right) \\
&= \frac{1}{Y}\left(1 + \sum_{t=1}^{Y-1} \frac{1}{t}\right) \\
&= \frac{1}{Y}(1 + H_{Y-1}),
\end{aligned}$$

where H_n denotes the n-th harmonic number. Recall also that $\alpha(Y \mid Y = 1) = \alpha(Y \mid Y = 2) = 1$.

Next, we estimate $\Pr[Y = i]$, by approximating Y with a Poisson distribution via Lemma A.2 (Le Cam's theorem). Let $\delta_i = \left|\binom{n}{i} p^i (1-p)^{n-i} - e^{-np}\frac{(np)^i}{i!}\right|$. The idea is to set p such that $np = q$, where $q \geq 1$ is a fixed constant. We know that $\Pr[Y = i] = \binom{n}{i} p^i (1-p)^{n-i}$, and thus, by Lemma A.2, we have

$$\sum_{i=0}^{\infty} \delta_i = \sum_{i=0}^{\infty} \left|\Pr[Y = i] - e^{-np}\frac{(np)^i}{i!}\right| = \sum_{i=0}^{\infty} \left|\Pr[Y = i] - e^{-q}\frac{(q)^i}{i!}\right| \leq \frac{2qp}{\max\{1, q\}} \leq 2p = \frac{2q}{n}.$$

Overall, the probability that $\mathcal{A}$ selects Z is

$$
\begin{aligned}
\alpha(Y) &= \sum_{i=0}^{n} \Pr[Y = i] \cdot \alpha\left(Y \mid Y = i\right) \\
&= \Pr[Y = 1] + \sum_{i=2}^{n} \Pr[Y = i] \cdot \alpha\left(Y \mid Y = i\right) \\
&\geq np(1-p)^{(n-1)} + \sum_{i=2}^{n} \left(e^{-q} \frac{q^i}{i!} - \delta_i \right) \cdot \alpha\left(Y \mid Y = i\right),
\end{aligned}
$$

where the last inequality follows by the definition of δ_i. Thus,

$$
\begin{aligned}
\alpha(Y) &= q(1-q/n)^{(n-1)} + \sum_{i=2}^{n} e^{-q} \frac{q^i}{i!} \cdot \alpha\left(Y \mid Y = i\right) - \sum_{i=2}^{n} \delta_i \cdot \alpha\left(Y \mid Y = i\right) \\
&\geq q(1-q/n)^{(n-1)} + \sum_{i=2}^{n} e^{-q} \frac{q^i}{i!} \frac{1 + H_{i-1}}{i} - \sum_{i=2}^{n} \delta_i \\
&\geq q(1-q/n)^{(n-1)} + e^{-q} \sum_{i=2}^{n} \frac{q^i \left(1 + H_{i-1}\right)}{i! \cdot i} - \frac{2q}{n}. \qquad (9)
\end{aligned}
$$

It is easy to see that simply setting $q = 2$, which corresponds to $p = 2/n$ and τ being the $2/n$-th quantile of $\mathcal{D}$, yields $\alpha(Y) > 0.5801$ for all $n \geq 20$. Thus, our simple single-threshold algorithm, augmented with a single oracle call, beats, even for small n, the optimal algorithm for the I.I.D prophet inequality which uses different thresholds per distribution and achieves a probability of success approximately 0.5801 [19].

Since the worst-case probability of ≈ 0.5801 by [19] is achieved for $n \to \infty$, one might be interested in the asymptotic behaviour of the probability of our algorithm, $\alpha(Y)$, for large n. It is not too difficult to see after some calculations that, as $n \to \infty$, Eq. (9) is maximized for $q \approx 2.435$, yielding $\alpha(Y) \approx 0.798$.

◀

4.2 A Single-Threshold Algorithm for General m

As we saw in the previous section, even for a simple, single-threshold algorithm, the analysis of the winning probability gets tedious very quickly. In this section, we generalize our single-threshold algorithm to the case of general m, and use the fact that the maximum of a uniformly random permutation of n values changes $O(\log n)$ times with high probability to obtain a guarantee on the winning probability that is super-exponential with respect to m.

As before, our algorithm selects a threshold τ such that $p = \Pr\left[X \geq \tau\right]$ and every time the algorithm observes a realization above τ, it uses an oracle query and asks $\mathcal{O}$ if the realization should be selected or not. If not, then it updates the threshold to the new higher value. If the algorithm runs out of oracle calls, then it selects the first element above the current threshold τ that is encounters, if any. In other words, the algorithm uses the oracle calls greedily for all realizations above τ.

▶ **Theorem 1.9.** *For sufficiently large m, n, and an instance of $\mathcal{M}(\mathcal{O}_m, I.I.D\ , PbM)$, there exists an algorithm that selects the maximum realization with probability at least $1 - O\left(m^{-m/5}\right)$.*

Proof. *Proof in Appendix B.6.* ◀

4.3 An (Almost) Tight Upper Bound

Now that we have presented a simple, single-threshold algorithm for the $\mathcal{M}(\mathcal{O}_m, \text{I.I.D}\ , PbM)$ setting, a reasonable question to ask is how far it is from being optimal. As we show in this section, the algorithm is asymptotically almost optimal.

▶ **Theorem 1.10.** *There exists an instance of $\mathcal{M}(\mathcal{O}_m, I.I.D\ , PbM)$ for which no algorithm can select the maximum realization with probability greater than $1 - \mathrm{O}\left(m^{-m}\right)$.*

Proof. *Proof in Appendix B.7.* ◀

References

1 Melika Abolhassani, Soheil Ehsani, Hossein Esfandiari, MohammadTaghi Hajiaghayi, Robert D. Kleinberg, and Brendan Lucier. Beating 1-1/e for ordered prophets. In Hamed Hatami, Pierre McKenzie, and Valerie King, editors, *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 61–71. ACM, 2017. doi:10.1145/3055399.3055479.

2 Keerti Anand, Rong Ge, Amit Kumar, and Debmalya Panigrahi. Online algorithms with multiple predictions. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 582–598. PMLR, July 2022. URL: https://proceedings.mlr.press/v162/anand22a.html.

3 Antonios Antoniadis, Christian Coester, Marek Elias, Adam Polak, and Bertrand Simon. Learningaugmented dynamic power management with multiple states via new ski rental bounds. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *NIPS_2021*, volume 34, pages 16714–16726. Curran Associates, Inc., 2021. URL: https://proceedings.neurips.cc/paper/2021/file/8b8388180314a337c9aa3c5aa8e2f37a-Paper.pdf.

4 David Assaf, Larry Goldstein, and Ester Samuel-Cahn. Ratio prophet inequalities when the mortal has several choices. *The Annals of Applied Probability*, 12(3):972–984, 2002.

5 David Assaf and Ester Samuel-Cahn. Simple ratio prophet inequalities for a mortal with multiple choices. *Journal of Applied Probability*, 37(4):1084–1091, 2000. URL: http://www.jstor.org/stable/3215496.

6 Siddhartha Banerjee, Vincent Cohen-Addad, Anupam Gupta, and Zhouzi Li. Graph searching with predictions, 2022. URL: https://arxiv.org/abs/2212.14220, doi:10.48550/ARXIV.2212.14220.

7 Lucien Le Cam. An approximation theorem for the poisson binomial distribution. *Pacific Journal of Mathematics*, 10:1181–1197, 1960.

8 Richard Cole and Tim Roughgarden. The sample complexity of revenue maximization. In *Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing*, STOC '14, pages 243–252, New York, NY, USA, 2014. Association for Computing Machinery. doi:10.1145/2591796.2591867.

9 Jose Correa, Andres Cristi, Paul Duetting, and Ashkan Norouzi-Fard. Fairness and bias in online selection. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 2112–2121. PMLR, 7 2021. URL: https://proceedings.mlr.press/v139/correa21a.html.

10 José R. Correa, Patricio Foncea, Ruben Hoeksma, Tim Oosterwijk, and Tjark Vredeveld. Posted price mechanisms and optimal threshold strategies for random arrivals. *Math. Oper. Res.*, 46(4):1452–1478, 2021. doi:10.1287/moor.2020.1105.

11 José R. Correa, Andrés Cristi, Boris Epstein, and José A. Soto. *The Two-Sided Game of Googol and Sample-Based Prophet Inequalities*, pages 2066–2081. URL: https://epubs.siam.org/doi/abs/10.1137/1.9781611975994.127, arXiv:https://epubs.siam.org/doi/pdf/10.1137/1.9781611975994.127, doi:10.1137/1.9781611975994.127.

12 Frank Den Hollander. Probability theory: The coupling method.

13 Nikhil R. Devanur and Thomas P. Hayes. The adwords problem: Online keyword matching with budgeted bidders under random permutations. In *Proceedings of the 10th ACM Conference on Electronic Commerce*, EC '09, pages 71–78, New York, NY, USA, 2009. Association for Computing Machinery. doi:10.1145/1566374.1566384.

14 Devdatt P. Dubhashi and Alessandro Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press, 2009. URL: http://www.cambridge.org/gb/knowledge/isbn/item2327542/.

15 Hossein Esfandiari, MohammadTaghi Hajiaghayi, Vahid Liaghat, and Morteza Monemizadeh. Prophet secretary. *SIAM Journal on Discrete Mathematics*, 31(3):1685–1701, 2017. arXiv:https://doi.org/10.1137/15M1029394, doi:10.1137/15M1029394.

16 Tomer Ezra, Michal Feldman, Nick Gravin, and Zhihao Gavin Tang. "who is next in line?" on the significance of knowing the arrival order in bayesian online settings. In Nikhil Bansal and Viswanath Nagarajan, editors, *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023, Florence, Italy, January 22-25, 2023*, pages 3759–3776. SIAM, 2023. doi:10.1137/1.9781611977554.ch145.

17 Tomer Ezra, Michal Feldman, and Ilan Nehama. Prophets and secretaries with overbooking. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, EC '18, page 319–320, New York, NY, USA, 2018. Association for Computing Machinery. doi:10.1145/3219166.3219211.

18 Khashayar Gatmiry, Thomas Kesselheim, Sahil Singla, and Yifan Wang. Bandit algorithms for prophet inequality and pandora's box. *CoRR*, abs/2211.08586, 2022. arXiv:2211.08586, doi:10.48550/arXiv.2211.08586.

19 John P. Gilbert and Frederick Mosteller. Recognizing the maximum of a sequence. *Journal of the American Statistical Association*, 61(313):35–73, 1966. URL: http://www.jstor.org/stable/2283044.

20 Elfarouk Harb. Fishing for better constants: The prophet secretary via poissonization, 2023. arXiv:2307.00971.

21 T. P. Hill and Robert P. Kertz. Comparisons of stop rule and supremum expectations of i.i.d. random variables. *Ann. Probab.*, 10(2):336–345, 05 1982. doi:10.1214/aop/1176993861.

22 Robert P Kertz. Stop rule and supremum expectations of i.i.d. random variables: A complete comparison by conjugate duality. *Journal of Multivariate Analysis*, 19(1):88 – 112, 1986. URL: http://www.sciencedirect.com/science/article/pii/0047259X86900953, doi:10.1016/0047-259X(86)90095-3.

23 Robert Kleinberg and S. Matthew Weinberg. Matroid prophet inequalities and applications to multi-dimensional mechanism design. *Games Econ. Behav.*, 113:97–115, 2019. doi:10.1016/j.geb.2014.11.002.

24 Ulrich Krengel and Louis Sucheston. Semiamarts and finite values. *Bull. Amer. Math. Soc.*, 83(4):745–747, 07 1977. URL: https://projecteuclid.org:443/euclid.bams/1183538915.

25 Ulrich Krengel and Louis Sucheston. On semiamarts, amarts, and processes with finite value. *Probability on Banach spaces*, 4:197–266, 1978.

26 Thodoris Lykouris and Sergei Vassilvitskii. Competitive caching with machine learned advice. *J. ACM*, 68(4), 7 2021. doi:10.1145/3447579.

27 Andrés Muñoz Medina and Sergei Vassilvitskii. Revenue optimization with approximate bid predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pages 1856–1864, Red Hook, NY, USA, 2017. Curran Associates Inc.

28 Michael Mitzenmacher and Sergei Vassilvitskii. Algorithms with predictions. *Commun. ACM*, 65(7):33–35, 6 2022. doi:10.1145/3528087.

29 Pranav Nuti. The secretary problem with distributions. In *Integer Programming and Combinatorial Optimization: 23rd International Conference, IPCO 2022, Eindhoven, The Netherlands,*

June 27–29, 2022, Proceedings, page 429–439, Berlin, Heidelberg, 2022. Springer-Verlag. doi:10.1007/978-3-031-06901-7_32.

30 Ester Samuel-Cahn. Comparison of threshold stop rules and maximum for independent nonnegative random variables. *The Annals of Probability*, 12(4):1213–1216, 1984. URL: http://www.jstor.org/stable/2243359.

31 Erik Vee, Sergei Vassilvitskii, and Jayavel Shanmugasundaram. Optimal online assignment with forecasts. In *Proceedings of the 11th ACM Conference on Electronic Commerce*, EC '10, pages 109–118, New York, NY, USA, 2010. Association for Computing Machinery. doi:10.1145/1807342.1807360.

A Background on Concentration Inequalities

We briefly present two lemmas that will be useful in the analysis of our algorithms; the standard Chernoff bound for binary random variables and Le Cam's theorem.

▶ **Lemma A.1** ([14]). *Let $Y_1, \ldots, Y_n$ be independent indicator random variables with $p_i = \Pr Y_i = 1$ and $Y = \sum_i Y_i$. Let $\mu = \mathbb{E}[Y] = \sum_i p_i$. Then,*

1. *For $\delta \geq 0$,*
$$\Pr\left[Y \geq (1+\delta)\mu\right] \leq \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\mu}.$$
2. *For $\delta \geq 0$,*
$$\Pr\left[Y \leq (1-\delta)\mu\right] \leq \left(\frac{e^{-\delta}}{(1-\delta)^{(1-\delta)}}\right)^{\mu}.$$
3. *For $\delta \in (0,1]$,*
$$\Pr\left[Y \geq (1+\delta)\mu\right] \leq e^{-\mu\delta^2/3}.$$
4. *For $\delta \in (0,1]$*
$$\Pr\left[Y \leq (1-\delta)\mu\right] \leq e^{-\mu\delta^2/2}.$$
5. *For $\delta > e^2$,*
$$\Pr\left[Y \geq (1+\delta)\mu\right] < e^{-\frac{\mu\delta\log\delta}{2}}.$$

Le Cam's theorem is useful in bounding the approximation error of a binomial distribution by a Poisson distribution. We will use a slightly tighter version [12].

▶ **Lemma A.2** ([7, 12]). *For every $n \in \mathbb{N}, p \in (0,1)$, we have*

$$\sum_{i=0}^{\infty}\left|\binom{n}{i}p^i(1-p)^{n-i} - e^{-np}\frac{(np)^i}{i!}\right| \leq \frac{2np^2}{\max\{1, np\}}.$$

B Proofs

B.1 Proof of Lemma 2.1

Proof. Again, the idea is that $\mathcal{B}$ can simulate $\mathcal{A}$'s behaviour by selecting each realization that $\mathcal{A}$ decides to query. Initially, $\mathcal{B}$ starts with an empty set S of selected values. Whenever

$\mathcal{B}$ is presented with a realization X_i, it feeds it to $\mathcal{A}$. If $\mathcal{A}$ decides to select X_i or expend a query for X_i, regardless of the outcome of the query, $\mathcal{B}$ always selects X_i into S, otherwise $\mathcal{B}$ decides not to select X_i. By induction, S contains exactly all the realizations that were queried by $\mathcal{A}$ as well as at most one more realization that might have been selected by $\mathcal{A}$ if it run out of queries. Therefore, $|S| \leq m+1$.

Now, notice that $\mathcal{A}$ succeeds if and only if it selects the maximum, and it only selects a realization X_i if (i) it chose to expend a query on X_i, or (ii) when it observed X_i it run out of queries. In both cases, by the description of $\mathcal{B}$, we know that $X_i \in S$, and thus the probability that $\mathcal{B}$ succeeds is at least α. ◀

B.2 Proof of Lemma 2.2

Proof. The idea is that $\mathcal{A}$ can simulate $\mathcal{B}$'s behaviour using the oracle queries instead of storing the values like $\mathcal{B}$ does. Initially, $\mathcal{B}$ starts with an empty set S of selected values. Whenever $\mathcal{A}$ is presented with a realization X_i, it feeds it to $\mathcal{B}$. If $\mathcal{B}$ selects X_i into S, $\mathcal{A}$ chooses to expend a query and ask $\mathcal{O}$ whether $X_i \geq \max_{j=i+1}^{n} X_j$. Consider the first i where this happens. We distinguish between the two possible answers:

- If $\mathcal{O}$ answers YES, then we know that all future realizations are smaller than X_i. However, we also know that since the objective is PbM , any optimal algorithm for $\textsc{Proph}_{m+1}$ will only select a value X_i if it is larger than any previously observed value (otherwise it "wastes" a spot in S for a value that is definitely not the maximum). Therefore, if $\mathcal{B}$ selects X_i, we know that $X_i \geq \max_{j<i} X_j$. In this case, both $\mathcal{B}$ and $\mathcal{A}$ succeed in selecting the maximum realization.
- If $\mathcal{O}$ answers NO, then we know that there exists a future realization that is greater than X_i. In this case, the instance for $\mathcal{B}$ reduces to $\mathcal{M}(\textsc{Proph}_m, y, PbM)$ on $X_{i+1}, \ldots, X_n$, whereas the instance for $\mathcal{A}$ reduces to $\mathcal{M}(\mathcal{O}_{m-1}, y, PbM)$. Since we know that $\mathcal{M}(\textsc{Proph}_1, y, PbM) = \mathcal{M}(\mathcal{O}_0, y, PbM)$ by definition, we have that by induction, the probability that $\mathcal{A}$ succeeds is at least α.

◀

B.3 Proof of Lemma 3.2

Proof. The fraction $\frac{\Gamma(m+1,x)}{m!}$ is also sometimes called the regularized gamma function. Taking the series expansion of the regularized gamma function as $m \to \infty$, we obtain

$$\frac{\Gamma(m+1,x)}{m!} = 1 - \left(\frac{e \cdot x}{m}\right)^m \cdot \frac{e^{-x}}{\sqrt{2\pi m}}.$$

Thus, for large m, ξ_m tends to

$$1 - e^{-\xi_m} = 1 - \left(\frac{e \cdot \xi_m}{m}\right)^m \cdot \frac{e^{-\xi_m}}{\sqrt{2\pi m}} \iff$$

$$\left(\frac{e \cdot \xi_m}{m}\right)^m = \sqrt{2\pi m} \iff$$

$$\xi_m = (2\pi m)^{1/2m} \cdot \frac{m}{e}.$$

Notice, however, that $\lim_{m\to\infty} (2\pi m)^{1/2m} = 1$, and thus,

$$\lim_{m\to\infty} \frac{\xi_m}{m} = \frac{1}{e}.$$

◀

B.4 Proof of Lemma 3.3

Proof. Define

$$h(x) = \sum_{i=0}^{m} e^{-x}\frac{x^i}{i!} - 1 + e^{-x} = \frac{\Gamma(m+1,x)}{m!} - 1 + e^{-x} \tag{10}$$

Clearly, ξ_m is the unique positive root of h by definition. Note that h is continuous, $h(0) = 1 > 0$, and h is strictly decreasing because $h'(x) = -e^{-x} - \frac{e^{-x}x^m}{m!} < 0$. Hence, it is sufficient to show that $h\left(((m+1)!)^{1/m+1}\right) < 0$ and conclude by the intermediate value theorem that $\xi_m \leq ((m+1)!)^{1/m+1}$. Rewriting $h(x)$,

$$h(x) = \sum_{i=0}^{m} e^{-x}\frac{x^i}{i!} - 1 + e^{-x} = e^{-x} - \sum_{i=m+1}^{\infty} e^{-x}\frac{x^i}{i!} = e^{-x}\left(1 - \sum_{i=m+1}^{\infty}\frac{x^i}{i!}\right) \tag{11}$$

In particular, we want to show that for $x \geq ((m+1)!)^{1/m+1}$, we have $\sum_{i=m+1}^{\infty}\frac{x^i}{i!} > 1$. Note that $R_m(x) = \sum_{i=m+1}^{\infty}\frac{x^i}{i!}$ is precisely the tail of the Taylor expansion of e^x, $R_m(x)$, and thus, by the Taylor remainder theorem, there exists a $\zeta \in (0,x)$ such that

$$R_m(x) = \frac{e^{\zeta}x^{m+1}}{(m+1)!} \geq \frac{x^{m+1}}{(m+1)!}$$

Hence for $x \geq ((m+1)!)^{1/m+1}$, we have $h(x) < 0$.

To show the lower bound, it suffices to prove

$$g(m) = \sum_{i=m+1}^{\infty}\frac{(m!)^{i/m}}{i!} < 1$$

Let $a_i = \frac{(m!)^{i/m}}{i!}$. First, note that that $m! \leq ({(m+1)}/{2})^m$ by the inequality of arithmetic and geometric means. Thus,

$$\frac{a_{i+1}}{a_i} = \frac{\frac{(m!)^{i+1/m}}{(i+1)!}}{\frac{(m!)^{i/m}}{i!}} = \frac{(m!)^{1/m}}{i+1} \leq \frac{m+1}{2(i+1)} < \frac{1}{2},$$

for $i \geq m+1$, which implies that $a_i < \frac{a_{m+1}}{2^{m+1-i}}$. But,

$$a_{m+1} = \frac{(m!)^{m+1/m}}{(m+1)!} = \frac{(m!)^{1/m}}{m+1} \leq \frac{1}{2}$$

Hence,

$$g(m) = \sum_{i=m+1}^{\infty} a_i < a_{m+1}\sum_{i=m+1}^{\infty}\frac{1}{2^{m+1-i}} \leq \frac{1}{2}\sum_{i=0}^{\infty} 2^{-i} = 1.$$

◀

B.5 Proof of Lemma 3.4

Proof. First, clearly $f(0,m)=0$. Next, we will show that $f(k+1,m)\geq f(k,m)$ which will imply the claim. We have

$$f(k+1,m)-f(k,m)=\sum_{j=1}^{k}\frac{\xi_m^j}{j!}+\frac{\xi_m^{k+1}}{(k+1)!}-\sum_{j=m+1}^{m+k}\frac{\xi_m^j}{j!}-\frac{\xi_m^{m+k+1}}{(m+k+1)!}-f(k,m) \tag{12}$$

$$=\frac{\xi_m^{k+1}}{(k+1)!}-\frac{\xi_m^{m+k+1}}{(m+k+1)!}\geq 0 \iff \xi_m\leq\left(\frac{(m+k+1)!}{(k+1)!}\right)^{1/m} \tag{13}$$

Note that $g(k)=\left(\frac{(m+k+1)!}{(k+1)!}\right)^{1/m}$ is strictly increasing on k. One way to see this for example is to observe that

$$\frac{g(k+1)}{g(k)}=\left(\frac{(m+k+2)!(k+1)!}{(m+k+1)!(k+2)!}\right)^{1/m}=\left(\frac{m+k+2}{k+2}\right)^{1/m}>1$$

Hence, to guarantee Eq. (13) for all k, it is sufficient to prove $\xi_m\leq((m+1)!)^{1/m}$. This follows by Lemma 3.3 since $((m+1)!)^{1/m}>((m+1)!)^{1/m+1}$. ◀

B.6 Proof of Theorem 1.9

Proof. Let $L=e^{\sqrt{m}}$. The idea is to set τ so that $p=\Pr[X\geq\tau]=L/n$. As before, let Y be the number of realizations above τ. By Lemma A.1, we have

$$\Pr[|Y-L|\geq\delta L]\leq 2e^{-\delta^2 L/3}.$$

Setting $\delta=1$ yields that $1\leq Y\leq 2L$ with probability at least $1-2e^{-L/3}=1-2e^{-e^{\sqrt{m}}/3}\geq 1-m^{-m/4}$ for all m.

Next, let $X_1',\ldots,X_Y'$ be the subsequence of all realizations larger than τ, according to their arrival order, and let $Z_i=1$ if $X_i'>\max_{j=1}^{i-1}X_j'$, in other words if X_i' is larger than all previous realizations, and $Z_i=0$ otherwise. Observe that $\Pr[Z_i=1]=1/i$, and that the random variables $Z_1,\ldots,Z_n$ are independent. Furthermore, let $M=\sum_i Z_i$ be the number of times that the maximum realization changes in the sequence $X_1',\ldots,X_Y'$. Observe that if $M\leq m+1$, then m oracle queries are sufficient for the algorithm to always select the maximum realization. Therefore, our goal is to bound the probability that this event happens.

Conditioned on $1\leq Y\leq 2L$, we have

$$\mathbb{E}[M]=\sum_{i=1}^{2L}\frac{1}{i}\leq\log(2L)+1\leq\sqrt{m}+2.$$

For $\delta=m+2/\mathbb{E}[M]-1$, we have

$$\Pr[M\geq m+2]=\Pr[M\geq(1+\delta)\,\mathbb{E}[M]].$$

Notice that for $m\geq 98$, we have $\delta\geq e^2$, and thus, by Lemma A.1, we obtain

$$\Pr[M\geq m+2]\leq e^{-\mathbb{E}[M]\delta\log\delta/2}\leq e^{-\frac{(m-\sqrt{m})(\log(m-\sqrt{m})-\log(m+2)/2)}{2}}\leq m^{-m/5}.$$

If we instead use the tight Chernoff bound in Lemma A.1, we can show that $\Pr[M\geq m+2]\leq m^{-m/4+\varepsilon}$ for all m and $\varepsilon>0$.

Putting everything together, for our algorithm to succeed, it suffices to have $1\leq Y\leq 2L$ and $M\leq m+1$, both of which happen together with probability at least $1-O\left(m^{-m/5}\right)$. ◀

B.7 Proof of Theorem 1.10

Proof. To construct an instance in which no algorithm can achieve a high probability, fix m and consider n random variables $X_1, \ldots, X_n$ drawn I.I.D from the uniform distribution on $[0, 1]$, where n is a sufficiently large number. We first divide $[0, 1]$ into $k = n/m \log m$ intervals $B_1, \ldots, B_k$ of length $m \log m/n$ each, with $B_i = \left((i-1) \cdot m \log m/n, i \cdot m \log m/n\right]$. For each $i = 1, \ldots, n$, let Y_i denote the random variable that is equal to 1 if $X_i \in B_k$ and 0 otherwise, where B_k is the last interval. Also, let $Y = \sum_{i=1}^{n} Y_i$. Since the X_i's follow the uniform distribution, we have $\Pr[Y_i = 1] = \frac{m \log m}{n}$ for all i, and $\mathbb{E}[Y] = m \log m$.

Next, consider an algorithm $\mathcal{A}$ for $\mathcal{M}(\mathcal{O}_m, \text{I.I.D }, PbM)$ on this instance, and assume that $Y \geq 1$, i.e. there exists at least one realization that falls in the last interval. Consider the moment that $\mathcal{A}$ observes a realization $X_i \in B_k$ that is larger than all previous realizations (including previous realizations in B_k). There are two cases:

- If $\mathcal{A}$ decides not to expend a query to $\mathcal{O}$ for this realization and skip it, there is a chance it fails to select the highest realization. This definitely happens if no other realization in the future is in B_k, which occurs with probability

$$(1 - X_i)^{n-i} \geq \left(1 - \frac{m \log m}{n}\right)^{n-i} \geq \left(1 - \frac{m \log m}{n}\right)^{n} \geq e^{-m \log m - 1} = \Omega\left(m^{-m/1-\varepsilon}\right)$$

  for sufficiently large n, for any $\varepsilon > 0$.
- If $\mathcal{A}$ decides to expend a query to $\mathcal{O}$ for this realization, there is a chance it fails to select the highest realization by running out of queries, deciding to select the next realization in B_k that is higher than all previous ones, and missing out on a higher realization in the future. For this to happen, it must be that $Y \geq m + 2$. Let $\delta = 1 - (1+1/m)/\log m$. By Lemma A.1, this happens with probability

$$\begin{aligned}\Pr[Y > m + 1] &= 1 - \Pr[Y \leq m + 1] \\ &= 1 - \Pr[Y \leq (1 - \delta)\, \mathbb{E}[Y]] \\ &\geq 1 - e^{-\frac{m \log m (\log m - 1 - 1/m)^2}{2 \log m^2}} \\ &\geq 1 - m^{-m/4}.\end{aligned}$$

  Given that $Y \geq m+2$, the probability that the first $m+2$ realizations arrive in increasing order is $1/(m+2)!$. Therefore, $\mathcal{A}$ misses out on the maximum realization in this case with probability at least (for $m \geq 6$)

$$\frac{1 - m^{-m/4}}{(m+2)!} \geq m^{-m}.$$

Therefore, $\mathcal{A}$ must miss the maximum realization with probability at least $\Omega\left(m^{-m}\right)$. ◀