Extensions to the Online Pen Testing Problem

by

Harish Prabhu Shankar

A thesis submitted in partial fulfillment
of the requirements for the degree of
Bachelor of Science
(Computer Science & Mathematics)

at the
University of Massachusetts Amherst
2026

This thesis has been approved by the following members of the Thesis Committee:
Mingda Qiao and Hedyeh Beyhaghi.

Abstract

We study a variant of the "online pen testing" problem, in which we are given n pens with an unknown amount of ink $X_1, \dots, X_n$. We want to select the pen with the maximum amount of ink. The problem arises in that we cannot access the value of X_i directly. We have the opportunity to write with the i-th pen until a certain amount of ink has been used, or the pen runs out of ink. In either case, we have diminished the amount of ink found in the pen and consequently the gained utility in selecting it. We examine the setting where there is a cost factor $c \in \mathbb{R}_{\geq 0}$ in writing with a pen i, that is if we test for T amount of ink, the remaining utility of the pen is $\max(X_i - cT, 0)$.

We consider two different pricing scenarios: one where information is discounted $(0 < c < 1)$ and where information is inflated $(c > 1)$. Furthermore, we consider two different setups: the "prophet" setting, in which each X_i is independently drawn from a distribution $\mathcal{D}_i$, and the "secretary" setting, in which each $(X_i)_{i=1}^n$ is a random permutation of some arbitrary $a_1, \dots, a_n$.

We show in a variety of settings that our algorithms are remarkably robust. In the prophet setting, given a complete description of the distributions, we can approximately maximize our obtained utility up to an $O(1)$ factor in the discounted price scenario and $e^{O(\sqrt{\ln n})}$ in the inflated price scenario. In the secretary setting, we retain the same approximations with respect to the pricing scenarios up to an arbitrary permutation given only the maxima.

Acknowledgments

I would first like to thank my advisor, Professor Mingda Qiao, for their guidance, patience, and encouragement throughout this project. Their feedback helped shape the mathematical direction of this thesis, the way to think about problems, and the way to present the ideas clearly.

I am also grateful to my committee member, Professor Hedyeh Beyhaghi, for taking the time to read my work and offer thoughtful comments. Their questions and suggestions helped improve the final version of the thesis.

Finally, I want to thank my family and friends for their constant support. Their encouragement made it easier to stay motivated through the difficult parts of this project, and I am deeply grateful for it.

Contents

Chapter 1

Introduction

1.1 Motivation

Suppose that we have a few emergency flashlights to choose from for an upcoming hike. We want to pick the flashlight that will last the longest. Before we make our decision about which one to bring, we turn each one on for some period of time to deduce whether the battery is almost dead. We face the immediate dilemma regarding how long each flashlight should be tested. If our test is just for a second, there is no way to distinguish a flashlight that has two seconds or three hours of light. At the other extreme, an overly long test could drain the battery entirely and we might not have a flashlight to choose from. How long should we test each flashlight, when the act of testing consumes exactly the resource we are trying to maximize?

This tension—information has value, and obtaining it consumes value—is not unique to our rather playful example. A firm hiring from a group of candidates may invest in a more exhaustive and demanding interview process, which improves its ability to identify the quality of candidates but requires additional resources upfront. A distinct but related scenario arises after hiring, where a probationary period serves as a continued learning phase: the firm gains more precise information about worker productivity only by employing the candidate, but at the cost of paying wages while their true quality remains uncertain [BBD85; OS11]. A venture capitalist deciding among early-stage startups can stage capital across multiple milestones, learning more about each company's trajectory before committing fully, but every staged round dilutes the eventual return [Gom95; KS03]. A clinical investigator running a Phase II dose-finding trial learns about a treatment's efficacy window only by expending patient exposure that cannot be recovered during Phase III [Ber06].

In each of these motivating examples, acquiring information consumes the resource we seek to optimize. Standard models of optimal stopping, however, decouple learning from such costs. A canonical example is the prophet inequality framework introduced by Krengel and Sucheston [KS77; KS78] and further developed by Samuel-Cahn [Sam84], in which a decision-maker observes a sequence of random variables X_i one by one, and must decide when to stop. Observations are exact and costless, though decisions are irrevocable. In the secretary problem of Lindley and Dynkin [Lin61; Dyn63], observation is free and the difficulty of the problem lies in the randomness of the order of items shown sequentially. Both of these frameworks have been extended to settings in which observation becomes costly. In one line of work, observing an option incurs a fixed fee per look [Jon90; Sam92; Kös04]; in Pandora's box, each option has its own inspection cost c_i [Wei79]. In these models, the cost of observation is paid in an external currency, separate from the realized value of the option itself.

Online pen testing, introduced by Qiao and Valiant [QV23] reformulates the problem by intertwining the cost of information acquisition and the value of the option. The only way to learn about the value of X_i is to write with pen i until it runs dry or survives past a chosen threshold θ_i. In the latter case, the decision maker may accept the pen and receive the remaining utility of $X_i - \theta_i$. In the former case, the pen has no remaining value. Qiao and Valiant show that this model admits $\Theta(\log n)$ competitive ratios throughout the prophet and secretary settings up to either a single sample per distribution in the prophet inequality setting, or no information at all besides the fact that the options presented are uniformly random in the secretary setting.

This thesis introduces a cost factor $c \in \mathbb{R}^+$ into the online pen testing problem, so that passing a test at threshold θ_i reduces the remaining utility to $X_i - c\theta_i$. We ground this discussion in the fact that the cost of acquiring information may not equal the value of the information obtained. We study two scenarios: (1) the discounted price scenario $c \in (0, 1)$, where the pen retains more ink than the cost of probing and (2) the inflated price scenario $c > 1$, where verifying the amount of ink in the pen costs more than the value it confirms. Neither is pathological: discounted prices correspond to "informationally" cheap tests (a LeetCode interview that conveys enough information to proceed with a candidate) while inflated prices correspond to higher overhead tests (an invasive diagnostic procedure to confirm a diagnosis).

1.2 Related Work

The literature most relevant to this thesis can be organized around a single structural choice: whether the cost of learning is paid externally (as a fee, discount, delay, or probing charge) or autogenously by consuming the very option whose residual value is the objective. Classical prophet and secretary problem settings are of the first variety, costless-observation benchmark. The costly observation in these problem spaces and the Pandora-box literature introduce external payments for information. Qiao and Valiant introduce the notion that information acquisition could be destructive to the option itself.

Prophet setting with costs. The prophet inequality starts with exact, costless observation: Krengel and Sucheston initiated the single-choice comparison between an online stopper and an omniscient prophet, and Samuel-Cahn gave the canonical threshold-rule form showing that the online decision maker can secure one half of the prophet benchmark in the independent nonnegative setting [KS77; KS78; Sam84]. The closest predecessors with costs are Jones's observation-fee model and the ensuing difference-prophet line of Samuel-Cahn and Kösters, where each look incurs a fixed external charge c, so the statistician optimizes realized value minus accumulated inspection cost rather than residual value of the chosen item itself [Jon90; Sam92; Kös04]. Weitzman's Pandora's box model attaches a box-specific inspection fee and deriving an optimal reservation-value policy under independent priors, but inspection still reveals the realized value exactly once the fee is paid [Wei79].

Secretary setting with costs. The secretary setting of Lindley and Dynkin similarly assumes free observation and random arrival, with the canonical sample-then-select procedure yielding a $1/e$ success threshold [Lin61; Dyn63]. Cost enters this literature in several ways. Early interview-cost variants charge for continuing to inspect candidates, so the decision maker trades off search depth against a separate interviewing penalty, while later discounted-secretary models attach a multiplicative time discount $d(t)$ to the value of the secretary eventually chosen [BG78; Bab+09]. In particular, Babaioff et al. show that discounted secretary admits an $O(1)$-competitive algorithm when the expected offline optimum is known, but in the unknown-OPT model the problem has an $\Omega(\log n/\log\log n)$ lower bound and a $O(\log n)$ upper bound; the same work also gives constant-competitive algorithms for weighted multiple-choice variants and extensions to broad matroid classes with only constant-factor or $O(\log \mathrm{rank})$ loss [Bab+09]. Qiao and Valiant show a $O(\log n)$-

competitive ratio is achievable with random order and no information, and also with arbitrary order if optimum information is given; matching $\Omega(\log n)$ lower bounds hold in easier variants; and in the easiest secretary regime—random order with full information—the optimal order is $\Theta(\log n/\log\log n)$ [QV23].

Multiple selection under constraints. On the secretary side, Kleinberg's multiple-choice secretary algorithm already showed that selecting up to k items can achieve a competitive ratio tending to 1 as k tends to infinity, specifically $1 - O(1/\sqrt{k})$, and Babaioff, Immorlica, Kempe, and Kleinberg generalized the random-order independent-set setting to the matroid secretary problem, thereby importing combinatorial feasibility constraints into online selection [Kle05; BIK07]. On the prophet side, matroid prophet inequalities show that when values are still observed exactly, the classical half-of-prophet benchmark can survive combinatorial constraints [KW12]. The external-cost literature generalized Pandora's idea to combinatorial probing: Singla's price-of-information framework reduces a wide range of combinatorial optimization problems with probing costs to corresponding full-information problems, and Gupta et al. extend this to Markovian information structures where value revelation itself evolves through costly state transitions [Sin18; Gup+19]. The closest direct predecessor for this thesis, however, is the recent combinatorial pen-testing work of Ganesh and Hartline, which transforms residual ink as consumer surplus in deferred-acceptance auctions and derives near-optimal algorithms under matroid, k-unit, knapsack, downward-closed, and online constraints [GH25]. Their bounds are explicitly logarithmic in the omniscient gap—for example $((1 + o(1))\ln n)$ for matroid constraints, $((1 + o(1))\ln(n/k))$ for $(k = o(n))$ identical-goods environments, $(2(1 + o(1))\ln n)$ for knapsack, and tight $((1 + o(1))\ln n)$-type behavior in the online i.i.d. single-item environment—precisely because the inspection cost is still paid by burning the selected objects rather than by an outside budget [GH25].

1.3 Setup

We define the online pen testing problem, identical to that from Qiao and Valiant [QV23], with the inclusion of a cost factor.

Definition 1 (Online Pen Testing)**.** An instance of the problem is defined over $X_1, \ldots, X_n \geq 0$ and some cost factor $c \in \mathbb{R}^+$. At each step $i \in [n]$, the player must go through two phases:

1. (Test) The player selects a threshold $\theta_i \in [0, \infty)$. The player then tests X_i against the chosen θ_i. If $X_i < \theta_i$, then the player observes the value of X_i and the pen has a remaining utility of $X_i' = 0$. Otherwise, the test passes ($X_i \geq \theta_i$) and the pen has a remaining utility of $X_i' = \max(X_i - c\theta_i, 0)$.

2. (Decision) If pen i passes the test, the player is given the choice to "accept" or "reject" the option irrevocably. If the player accepts the pen, no more pens from the sequence are shown to the player, and the player obtains a utility of X_i'.

Definition 2 (Prophet Setting)**.** The player is given information about the distributions $\mathcal{D}_1, \ldots, \mathcal{D}_n$ defined over $[0, +\infty)$, where the values $X_1, \ldots, X_n$ are drawn independently from their corresponding distribution. The player is given a complete description of the distributions $(\mathcal{D}_i)_{i=1}^n$.

Remark 1 (Distributional Assumptions, following [QV23])**.** We will assume for simplicity that each $\mathcal{D}_i$ is continuous; that is, its respective cumulative distribution function $F_i(x)$ is continuous. This continuity guarantees that for any $\alpha \in (0, 1]$, we may define τ_α as the smallest $(1 - \alpha)$-quantile of the distribution. Simply put, τ_α is the minimal value τ such that $\Pr_{X \sim \mathcal{D}_i}(X > \tau) = \alpha$.

Definition 3 (Secretary Setting)**.** The n values shown sequentially to the player $X_1, \ldots, X_n$ are guaranteed to be a permutation, either uniformly random or arbitrary, of some ordering of values $a_1, \ldots, a_n \geq 0$.

Remark 2. Our definition uses the weak inequality $X_i \geq \theta_i$ for a successful test, whereas Qiao and Valiant use the strict inequality $X_i > \theta_i$. In the prophet setting, this distinction is immaterial under the continuity assumption, since $\Pr(X_i = \theta_i) = 0$. We use the weak inequality because it slightly simplifies some arguments in the secretary setting, where the values are fixed rather than drawn from continuous distributions.

Definition 4 (Pricing Scenarios)**.** We will consider two cases based on the value of the cost factor $c \in \mathbb{R}^+$:

1. **Discounted Price**: $c \in (0, 1)$

2. **Inflated Price**: $c > 1$

1.4 Summary of Results

We establish the following asymptotic competitive-ratio guarantees for online pen testing, with a discounted or an inflated cost factor, under different variants of the prophet and secretary settings defined above.

The prophet setting. The first two chapters address the prophet setting, both in the IID case and in the more general independent-distribution case, under the assumption that the player is given complete descriptions of the distributions. In the discounted price scenario, we obtain an $O(1)$ competitive ratio for every fixed $c < 1$. The word "fixed" is important here: the constants in some of the discounted-price arguments depend on c, and should not always be interpreted as uniform guarantees as $c \to 1$. In the IID case, the geometric-threshold proof gives a more refined interpolation: the relevant loss factor converges to $\Theta(\log n)$ as $c \to 1$, matching the order of the original online pen testing result. In the general prophet setting, the direct discounted-price proof is instead driven by the extra gain $(1-c)\tau_{1/2}$ obtained from testing below full cost, and therefore its constant deteriorates as $c \to 1$. This does not contradict the $c = 1$ theory; rather, it shows that a different argument is needed at the boundary, which is why we separately include a simpler argument for the case $c = 1$. In the inflated price scenario, we obtain a sub-polynomial competitive ratio of $e^{O(\sqrt{\ln n})}$ for every fixed $c > 1$.

	Discounted Price	Inflated Price
IID Prophet Setting	$O(1)$	$e^{O\left(\sqrt{\ln n}\right)}$
General Prophet Setting	$O(1)$	$e^{O\left(\sqrt{\ln n}\right)}$

Table 1.1: Results for the prophet setting.

The secretary setting. The last chapter studies the secretary setting under two independent types of variation: the order in which the values arrive and the amount of information provided.

Discounted price: When the values arrive in uniformly random order, an $O(1)$ competitive algorithm is achievable even when no information about the multiset $\{a_1, \ldots, a_n\}$ is given. This immediately implies the same guarantee in the random-order settings with optimum information or full information. When the arrival order is arbitrary, an $O(1)$ competitive ratio is still achievable if the algorithm is given the maximum value $a_{[1]}$. This also implies the full-information arbitrary-order guarantee.

	Full Information	Optimum Information	No Information
Random Order	$O(1)$	$O(1)$	$O(1)$
Arbitrary Order	$O(1)$	$O(1)$	$O(n)$

Table 1.2: Results for the secretary setting in the discounted price scenario.

Remark 3. The $O(n)$ entry for arbitrary order with no information is the trivial baseline. Consider the randomized algorithm that chooses an index $j \in [n]$ uniformly at random before the sequence begins. It sets threshold 0 at time j, and rejects all other positions. Since the arrival order is arbitrary but fixed, the maximum value $a_{[1]}$ appears in some position $j^\star \in [n]$. The algorithm selects this position with probability $1/n$, and if $j = j^\star$, it receives utility $a_{[1]}$, since testing at threshold 0 incurs no cost. Therefore the expected utility is $a_{[1]}/n$. Hence this strategy is $O(n)$ competitive.

Inflated price: This requires a more careful distinction on the basis of the amount of information provided. In random arrival order, we obtain $e^{O(\sqrt{\ln n})}$ competitive ratio given full information, or optimum information, and even no information. In arbitrary arrival order, the same guarantee occurs given full information. With only optimum information, the algorithm incurs an additional logarithmic factor because the algorithm must use a sub-procedure that mimics random arrival order. This does not change the displayed asymptotic form, since

$$\log n \cdot e^{O\left(\sqrt{\ln n}\right)} = e^{O\left(\sqrt{\ln n} + \ln \log n\right)} = e^{O\left(\sqrt{\ln n}\right)}$$

this logarithmic loss is absorbed into the same asymptotic form. Thus, we still achieve an $e^{O(\sqrt{\ln n})}$ competitive ratio. As in the discounted case, arbitrary order with no information is the recorded trivial $O(n)$ baseline.

Remark 4. The $O(n)$ entry for arbitrary order with no information has the same justification as in the discounted setting. The algorithm chooses one position uniformly at random, sets threshold 0 at that position, and rejects all other positions. Since

	Full Information	Optimum Information	No Information
Random Order	$e^{O\left(\sqrt{\ln n}\right)}$	$e^{O\left(\sqrt{\ln n}\right)}$	$e^{O\left(\sqrt{\ln n}\right)}$
Arbitrary Order	$e^{O\left(\sqrt{\ln n}\right)}$	$e^{O\left(\sqrt{\ln n}\right)}$	$O(n)$

Table 1.3: Results for the secretary setting in the inflated price scenario.

the threshold is 0, the cost factor c is irrelevant: if the chosen position contains the maximum value $a_{[1]}$, the algorithm receives utility $a_{[1]}$. This happens with probability $1/n$, giving expected utility $a_{[1]}/n$.

1.5 Open Problems

The transition near $c = 1$. The boundary case $c = 1$ is conceptually important because it recovers the original online pen testing model. The results in this thesis approach this boundary in two different ways. In the IID prophet setting, the discounted and inflated geometric-threshold arguments naturally degenerate to logarithmic losses as $c \to 1$, recovering the correct order of the original model. In the general prophet discounted setting, however, the proof used for $c < 1$ does not have this behavior: it obtains the threshold term from the additional utility $(1-c)\tau_{1/2}$, and therefore gives a ratio proportional to $1/(1-c)$. As $c \to 1$, this source of gain disappears. A natural open problem is to develop a unified argument whose guarantee transitions smoothly between the discounted regime and the original $c = 1$ model.

Constant Optimization Throughout the thesis, the focus is on the asymptotic order of the competitive ratio rather than the exact constants. This leaves open the question of what the optimal dependence on the cost factor c should be.

Limited-information prophet settings. The prophet results in this thesis assume that the player is given complete descriptions of the distributions. This is a rather strong assumption. In the original online pen testing problem, Qiao and Valiant show that logarithmic competitive ratios can still be achieved when the algorithm receives only one sample from each distribution. It is natural to ask whether analogous sample-based guarantees hold in the cost-factor model. The algorithms developed here rely on quantile thresholds whose spacing depends on both n and c, and estimating the relevant tail quantiles from samples may be difficult and imprecise.

More generic cost functions After passing a test at level θ_i, the remaining utility is $X_i' = \max(X_i - c\theta_i, 0)$. In many applications, however, the cost of information may be nonlinear. For example, early testing may be cheap while later testing becomes increasingly expensive. Natural cases include convex costs, concave costs, affine costs of the form $g(\theta) = a + b\theta$, and item-dependent costs $g_i(\theta)$.

Combinatorial and multiple-selection variants. Another natural direction is to move beyond selecting a single pen. In many applications, the decision maker may want to select several options, possibly subject to a feasibility constraint. For example, one might be allowed to choose up to k pens, or more generally choose a feasible set in a matroid, knapsack, or downward-closed constraint system. The objective would be to maximize the total remaining utility of the selected options after testing. It is also closely connected to recent work on combinatorial pen testing [GH25], where residual value is interpreted as surplus under feasibility constraints. The cost-factor model introduces a new question: how do the guarantees change when testing is discounted or inflated? In the discounted regime, it is plausible that many combinatorial guarantees should improve. In the inflated regime, however, feasibility constraints may amplify the difficulty.

Chapter 2

IID Prophet Setting

This chapter develops the main ideas of the cost-factor model in the simplest distributional setting: the IID prophet setting. Throughout the chapter, the values $X_1, \ldots, X_n$ are drawn independently from a common distribution $\mathcal{D}$, and the player is given a complete description of this distribution. The IID assumption allows us to focus on the central analytic tools of the thesis without the overhead of the general prophet or secretary settings.

We begin in Section 2.1 by establishing two preliminary tools that will be used throughout the chapter. The first is a prophet upper bound, which decomposes the expected maximum of the values into a threshold term and a tail-expectation term. The second is a remaining-utility lemma, which shows how much utility can be guaranteed from testing at one quantile when a larger tail quantile is known. This will form the basis of many our algorithms in the thesis: separating the expected best value into a "threshold" part and a "tail" part, and then design algorithms that recover both parts up to a competitive factor. Thus, the chapter should be read as introducing the two recurring proof patterns of the thesis: single-threshold arguments for the tail of the distribution, and geometric-threshold arguments for recovering value from the appropriate quantile scale.

2.1 Preliminaries

We begin with the IID prophet setting, where $\mathcal{D}_1, \ldots, \mathcal{D}_n$ are all the same distribution, which for the remainder of this section we will refer to as $\mathcal{D}$. The variables $X_1, \ldots, X_n$ are independent random variables drawn from $\mathcal{D}$, and we define $X^{\max} := \max(X_1, \ldots, X_n)$.

We establish a general upper bound on the expected score of the prophet. This generalizes the structural lemmas of [QV23] to accommodate a cost factor.

Lemma 5 (Prophet Upper Bound)**.** *For any sequence of distributions $\mathcal{D}_1, \ldots, \mathcal{D}_n$, a threshold $\theta \in \mathbb{R}_{\geq 0}$ and a cost factor $c \in \mathbb{R}^+$,*

$$\mathbb{E}[X^{\max}] \leq c\theta + \sum_{i=1}^{n} \mathop{\mathbb{E}}_{X_i \sim \mathcal{D}_i}[\max(X_i - c\theta, 0)].$$

Proof. We can rewrite $X^{\max}$ as $c\theta + (X^{\max} - c\theta)$. This gives:

$$\begin{aligned}
\mathbb{E}[X^{\max}] &= c\theta + \mathbb{E}[X^{\max} - c\theta] \\
&\leq c\theta + \mathbb{E}[\max(X^{\max} - c\theta, 0)] \\
&\leq c\theta + \mathbb{E}\left[\sum_{i=1}^{n} \max(X_i - c\theta, 0)\right] = c\theta + \sum_{i=1}^{n} \mathop{\mathbb{E}}_{X_i \sim \mathcal{D}_i}[\max(X_i - c\theta, 0)].
\end{aligned}$$

□

Applying Lemma 5 to the IID variables yields the following upper bound:

$$\mathbb{E}[X^{\max}] \leq c\theta + n \cdot \mathop{\mathbb{E}}_{X \sim \mathcal{D}}[\max(X - c\theta, 0)]. \tag{2.1}$$

Definition 6 (α-Competitive Algorithm)**.** An algorithm is α-competitive if its expected score, over the randomness in the distributional information, generation of $(X_i)_{i=1}^n$, and the player itself is at least

$$\frac{1}{\alpha} \cdot \mathop{\mathbb{E}}_{X \sim \mathcal{D}}[\max_{i \in [n]} X_i].$$

We first establish that an option that passes when setting a threshold of $\theta = \tau_\alpha$ has an expected remaining utility of $\Omega\big(\tau_{\alpha/\beta} - c\tau_\alpha\big)$. We say this to reason, that conditional on passing the test at a lower threshold τ_α, there is a non-trivial chance that the value of the option lies above a higher threshold $\tau_{\alpha/\beta}$. Among the α fraction of values above τ_α, an α/β fraction lie above $\tau_{\alpha/\beta}$, so this higher event occurs with conditional probability $1/\beta$. Whenever this happens, the remaining utility is at least

$$\tau_{\alpha/\beta} - c\tau_\alpha.$$

Thus the expected remaining utility after a pass at threshold τ_α is at least this possible gain multiplied by the conditional probability $1/\beta$. The lemma below formalizes this simple tail-probability argument.

Lemma 7. *For any distribution $\mathcal{D}$, $\alpha \in (0,1]$, $c > 0$, and $\beta > 1$, the expected remaining utility from a test at τ_α satisfies*

$$\mathop{\mathbb{E}}_{X\sim\mathcal{D}}[\max(X - c\tau_\alpha, 0) \mid X > \tau_\alpha] \geq \frac{\tau_{\alpha/\beta} - c\tau_\alpha}{\beta}.$$

Proof. Rewriting the conditional expectation, we have:

$$\mathop{\mathbb{E}}_{X\sim\mathcal{D}}[\max(X - c\tau_\alpha, 0) \mid X > \tau_\alpha] = \frac{\mathbb{E}_{X\sim\mathcal{D}}[\max(X - c\tau_\alpha, 0) \cdot \mathbb{1}[X > \tau_\alpha]]}{\Pr(X > \tau_\alpha)}.$$

Since $\tau_{\alpha/\beta} \geq \tau_\alpha$ given by $\beta > 1$, we claim that for every x,

$$\max(x - c\tau_\alpha, 0) \cdot \mathbb{1}[x > \tau_\alpha] \geq \max\bigl(\tau_{\alpha/\beta} - c\tau_\alpha, 0\bigr) \cdot \mathbb{1}\bigl[x > \tau_{\alpha/\beta}\bigr].$$

To verify this, consider the possible cases:

- If $x \leq \tau_\alpha \leq \tau_{\alpha/\beta}$, then both sides are 0.
- If $\tau_\alpha < x \leq \tau_{\alpha/\beta}$, then RHS is 0, while LHS is nonnegative.
- If $\tau_\alpha \leq \tau_{\alpha/\beta} < x$, then $x - c\tau_\alpha > \tau_{\alpha/\beta} - c\tau_\alpha$, which directly implies that $\max(x - c\tau_\alpha, 0) \geq \max\bigl(\tau_{\alpha/\beta} - c\tau_\alpha, 0\bigr)$.

Taking the expectation of both sides gives:

$$\begin{aligned}\mathop{\mathbb{E}}_{X\sim\mathcal{D}}[\max(X - c\tau_\alpha, 0) \cdot \mathbb{1}[X > \tau_\alpha]] &\geq \max\bigl(\tau_{\alpha/\beta} - c\tau_\alpha, 0\bigr) \cdot \Pr\bigl(X > \tau_{\alpha/\beta}\bigr) \\ &= \frac{\alpha}{\beta} \cdot \max\bigl(\tau_{\alpha/\beta} - c\tau_\alpha, 0\bigr).\end{aligned}$$

Dividing by $\Pr(X > \tau_\alpha) = \alpha$ yields,

$$\begin{aligned}\mathop{\mathbb{E}}_{X\sim\mathcal{D}}[\max(X - c\tau_\alpha, 0) \mid X > \tau_\alpha] &\geq \frac{\max\bigl(\tau_{\alpha/\beta} - c\tau_\alpha, 0\bigr)}{\beta} \\ &\geq \frac{\tau_{\alpha/\beta} - c\tau_\alpha}{\beta}.\end{aligned}$$

$\square$

2.2 Discounted Price

To evaluate our mechanism, we benchmark against the upper bound established in Equation (2.1) evaluated at $\theta = \tau_{1/n}$:

$$\mathbb{E}[X^{\max}] \leq \tau_{1/n} + n \cdot \mathbb{E}\big[\max\big(X - \tau_{1/n}, 0\big)\big]. \tag{2.2}$$

2.2.1 Single Threshold Algorithm

Lemma 8. *Let $X_1, \ldots, X_n \sim \mathcal{D}$ i.i.d., $c \in (0, 1)$, and $\theta = \tau_{1/n}$. Then, the expected utility of the single-threshold algorithm is at least*

$$\left(1 - \frac{1}{e}\right) \cdot n \cdot \mathbb{E}[\max(X - \theta, 0)].$$

Proof. By definition, $\Pr(X > \tau_{1/n}) = 1/n$. The probability that every option fails is $(1-1/n)^n$. Therefore, the probability that at least one option passes is $1-(1-1/n)^n$. Using the standard bound $(1-1/n)^n \leq 1/e$ for all $n \geq 1$, the probability of accepting an option is strictly greater than $1 - 1/e$.

Conditioning on the fact that a pen passes the test $(X > \tau_{1/n})$, the expected score from the accepted pen is:

$$\begin{aligned}
\mathbb{E}\big[\max\big(X - c\tau_{1/n}, 0\big) \mid X > \tau_{1/n}\big] &= \frac{\mathbb{E}\big[\max\big(X - c\tau_{1/n}, 0\big) \cdot \mathbb{1}\big[X > \tau_{1/n}\big]\big]}{\Pr\big(X > \tau_{1/n}\big)} \\
&\geq \frac{\mathbb{E}\big[\max\big(X - \tau_{1/n}, 0\big) \cdot \mathbb{1}\big[X > \tau_{1/n}\big]\big]}{1/n} \\
&= n \cdot \mathbb{E}\big[\max\big(X - \tau_{1/n}, 0\big)\big].
\end{aligned}$$

Thus, the expected score is at least

$$\left(1 - \frac{1}{e}\right) \cdot n \cdot \mathbb{E}\big[\max\big(X - \tau_{1/n}, 0\big)\big]. \tag{2.3}$$

□

2.2.2 Geometric Threshold Algorithm

We define $k := \lceil \log_2 n \rceil$ and let $\beta := 2$ be a controlled parameter for the spacing of the sequence of thresholds. We construct the sequence of quantiles, $\alpha_j = \beta^{-j}$ and their corresponding thresholds $T_j = \tau_{\beta^{-j}}$ for $j = 0, 1, \ldots, k$. Set $T_0 = \tau_1 = 0$ for simplicity.

The purpose of this construction is to find a pair of consecutive quantile thresholds where the gain from moving upward in the tail is large. Since $c < 1$, testing at the lower threshold T_{j-1} and then receiving a value around the next threshold T_j leaves a positive marginal utility of roughly

$$T_j - cT_{j-1}.$$

Thus, we want to identify some scale j for which this marginal gain is large. The key point is that the largest marginal gain cannot be too small. If every difference $T_j - cT_{j-1}$ were small, then the entire sequence of thresholds $T_0, T_1, \ldots, T_k$ could not grow all the way up to T_k, which is at least $\tau_{1/n}$. The recurrence below formalizes this idea: by defining

$$B = \max_{j \in [k]} (T_j - cT_{j-1}),$$

we have $T_j \leq B + cT_{j-1}$ for every j. Iterating this inequality shows that T_k is controlled by a geometric sum in c times B. Rearranging therefore gives a lower bound on B in terms of the target quantile $\tau_{1/n}$.

With this intuition in mind, define B as we have done above. Then $T_j - cT_{j-1} \leq B$ for every j, so $T_j \leq B + cT_{j-1}$. Expanding this recurrence from $T_0 = 0$ yields:

$$\begin{aligned}
T_1 &\leq cT_0 + B = B \\
T_2 &\leq cT_1 + B \leq cB + B = (c+1)B \\
T_3 &\leq cT_2 + B \leq c(c+1)B + B = (c^2 + c + 1)B \\
&\vdots \\
T_k &\leq (c^{k-1} + c^{k-2} + \ldots + 1)B = B \cdot \frac{1 - c^k}{1 - c}.
\end{aligned}$$

Rearranging this gives a lower bound on B:

$$T_k \cdot \frac{1 - c}{1 - c^k} \leq B.$$

By the definition of k, we know $T_k = \tau_{2^{-k}} \geq \tau_{1/n}$, we can conclude

$$\tau_{1/n} \cdot \frac{1 - c}{1 - c^k} \leq B.$$

The algorithm proceeds as follows:

1. Select $j^* = \operatorname{argmax}_{j\in[k]}(T_j - cT_{j-1})$.
2. Set the threshold $\theta = T_{j^*-1}$.
3. Test each pen with θ, and accept the first pen that passes the test. Note that since $X_i \geq \theta \geq 0$ and $c \in (0,1)$, naturally $X_i - c\theta > 0$.

The probability that a pen passes the test is

$$\Pr(X > T_{j^*-1}) = \beta^{-(j^*-1)} \geq 2^{-k+1} \geq \frac{1}{n}.$$

Thus, the probability that at least one pen passes is at least

$$1 - \left(1 - \frac{1}{n}\right)^n > 1 - \frac{1}{e}.$$

Now, conditioning on the first pen that passes the test, the expected remaining utility is lower bounded using Lemma 7:

$$\begin{aligned}
\mathbb{E}\big[\max\big(X - c\tau_{\beta^{-(j^*-1)}}, 0\big) \mid X > \tau_{\beta^{-(j^*-1)}}\big] &\geq \frac{\tau_{\beta^{-j^*}} - c\tau_{\beta^{-(j^*-1)}}}{\beta} \\
&= \frac{T_{j^*} - cT_{j^*-1}}{\beta} \\
&= \frac{B}{2}.
\end{aligned}$$

Therefore, the expected score of this algorithm is at least

$$\left(1 - \frac{1}{e}\right) \cdot \frac{1-c}{2(1-c^k)} \cdot \tau_{1/n}.$$

2.2.3 Competitive Ratio

Theorem 9. *In the prophet IID setting, given a discounted cost factor $c \in (0,1)$ and a complete description of the distribution $\mathcal{D}$, there exists an algorithm that achieves a competitive ratio of $O(1)$.*

Proof. By randomizing between the Single Threshold and the Geometric Threshold Algorithm with equal probability 1/2, the overall expected score is at least:

$$\frac{1-1/e}{2}\left(\frac{1-c}{2(1-c^k)} \cdot \tau_{1/n} + n\,\mathbb{E}\big[\max\big(X - \tau_{1/n}, 0\big)\big]\right).$$

Since $1 - c^k < 1$, it follows that

$$\frac{1-c}{2} < \frac{1-c}{2(1-c^k)}.$$

Therefore, the combined mechanism achieves an expected score that is at least

$$\frac{1-1/e}{2} \cdot \left(\frac{1-c}{2} \tau_{1/n} + n\,\mathbb{E}\big[\max\big(X - \tau_{1/n}, 0\big)\big] \right).$$

When compared to the upper bound established in (2.2), the combined score is strictly off by a fixed constant factor. Hence, the algorithm is $O(1)$-competitive. □

Remark 5. The dependence on c in the geometric threshold algorithm is captured by the factor

$$\frac{1-c}{1-c^k}.$$

For any fixed $c < 1$, this is a constant independent of n, so the discounted-price result remains $O(1)$-competitive. However, this constant worsens as c approaches 1. In the limiting case,

$$\lim_{c \to 1} \frac{1-c}{1-c^k} = \frac{1}{k}.$$

Since $k = \lceil \log_2 n \rceil$, this factor becomes $\Theta(1/\log n)$. Thus, the guarantee naturally degrades from a constant-factor bound to the known $O(\log n)$ competitive ratio.

2.3 Inflated Price

Again, we benchmark our mechanism against the upper bound evaluated at $\theta = \tau_{1/n}$:

$$\mathbb{E}[X^{\max}] \le c\tau_{1/n} + n \cdot \mathbb{E}\big[\max\big(X - c\tau_{1/n}, 0\big)\big]. \tag{2.4}$$

2.3.1 Single Threshold Algorithm

Lemma 10. *Let $X_1, \ldots, X_n \sim \mathcal{D}$ i.i.d., $c > 1$, and $\theta = \tau_{1/n}$. Then the expected utility of the single-threshold algorithm is at least*

$$\left(1 - \frac{1}{e}\right) \cdot n \cdot \mathbb{E}[\max(X - c\theta, 0)].$$

Proof. As in the discounted case, the probability of accepting an option is strictly greater than $1 - 1/e$. Conditioning on the fact that a pen is the first to pass the test, the expected score from the accepted pen is:

$$\begin{aligned}\mathbb{E}\big[\max\big(X - c\tau_{1/n}, 0\big) \mid X > \tau_{1/n}\big] &= \frac{\mathbb{E}\big[\max\big(X - c\tau_{1/n}, 0\big) \cdot \mathbb{1}\big[X > \tau_{1/n}\big]\big]}{\Pr\big(X > \tau_{1/n}\big)} \\ &= n \cdot \mathbb{E}\big[\max\big(X - c\tau_{1/n}, 0\big)\big].\end{aligned}$$

Thus, the expected score is at least

$$\left(1 - \frac{1}{e}\right) \cdot n \cdot \mathbb{E}\big[\max\big(X - c\tau_{1/n}, 0\big)\big]. \tag{2.5}$$

□

2.3.2 Geometric Threshold Algorithm

We follow the same general construction as in Section 2.2.2, but use a more flexible spacing parameter to account for an unbounded c. In the discounted case, we fixed $\beta = 2$, so the thresholds moved through the tail at a fixed geometric rate. When $c > 1$, however, testing is more expensive, and the spacing between consecutive quantiles must be chosen more carefully.

Let $k \geq 1$ and $\beta > 1$ be chosen so that $\beta^k = n$. We define a sequence of tail probabilities $\alpha_j = \beta^{-j}$ for $j = 0, 1, \ldots, k$ and let $T_j = \tau_{\beta^{-j}}$ be the corresponding quantile threshold. Thus $T_0 = \tau_1 = 0$ and $T_k = \tau_{\beta^{-k}} = \tau_{1/n}$.

As in the discounted case, the goal is to find a pair of consecutive thresholds for which the marginal gain $T_j - cT_{j-1}$ is meaningfully large. This quantity represents the utility we can certify if we test at the lower threshold T_{j-1} and the realized value reaches the next threshold T_j. The difference from the discounted case is that $c > 1$, so this marginal gain may be negative for many adjacent pairs. Nevertheless, the largest marginal gain cannot be too small: if every $T_j - cT_{j-1}$ were small, then the sequence of thresholds could not grow from $T_0 = 0$ up to $T_k = \tau_{1/n}$.

Define $B = \max_{j \in [k]}(T_j - cT_{j-1})$. Since $T_j - cT_{j-1} \leq B$ for every j, we can unroll the recurrence starting with $T_0 = 0$ to yield:

$$T_k \leq B(1 + c + c^2 + \cdots + c^{k-1}) = B\frac{c^k - 1}{c - 1}.$$

Rearranging gives

$$B \geq T_k \frac{c-1}{c^k-1}.$$

The algorithm proceeds as follows:

1. Select $j^* = \text{argmax}_{j\in[k]}(T_j - cT_{j-1})$
2. Set the threshold to $\theta = T_{j^*-1}$.
3. Test each pen with threshold θ, and accept the first pen that passes. The utility obtained from the accepted pen is $\max(X_i - c\theta, 0)$.

The probability that a pen passes the test is

$$\Pr(X > T_{j^*-1}) = \beta^{-(j^*-1)} \geq \beta^{-(k-1)}.$$

Since $\beta^k = n$, we can further claim

$$\beta^{-(k-1)} = \frac{\beta}{\beta^k} = \frac{\beta}{n} > \frac{1}{n}.$$

Thus, the probability that at least one pen passes is

$$1 - \left(1 - \beta^{-(j^*-1)}\right)^n \geq 1 - \left(1 - \frac{1}{n}\right)^n > 1 - \frac{1}{e}.$$

Conditioning on the event that the first pen passes, we apply Lemma 7 with

$$\alpha = \alpha_{j^*-1} = \beta^{-(j^*-1)}.$$

Since the threshold used is $T_{j^*-1} = \tau_{\alpha_{j^*-1}} = \tau_{\beta^{-(j^*-1)}}$, the next quantile appearing in the lemma is $\tau_{\alpha/\beta} = \tau_{\beta^{-j^*}} = T_{j^*}$. Therefore,

$$\begin{aligned}
\mathbb{E}[\max(X - cT_{j^*-1}, 0) \mid X > T_{j^*-1}] &= \mathbb{E}\big[\max\big(X - c\tau_{\beta^{-(j^*-1)}}, 0\big) \mid X > \tau_{\beta^{-(j^*-1)}}\big] \\
&\geq \frac{\tau_{\beta^{-j^*}} - c\tau_{\beta^{-(j^*-1)}}}{\beta} \\
&= \frac{T_{j^*} - cT_{j^*-1}}{\beta} \\
&= \frac{B}{\beta}.
\end{aligned}$$

Combining this conditional utility bound with the probability of seeing at least one passing pen, the expected score is at least

$$\left(1 - \frac{1}{e}\right) \cdot \frac{B}{\beta}.$$

Using the lower bound we found for B, we get

$$\left(1 - \frac{1}{e}\right) \cdot \frac{B}{\beta} \geq \left(1 - \frac{1}{e}\right) \cdot \frac{1}{\beta} \cdot T_k \cdot \frac{c-1}{(c^k - 1)}.$$

Finally, since $T_k = \tau_{\beta^{-k}}$, the expected score is at least

$$\left(1 - \frac{1}{e}\right) \cdot \frac{c-1}{\beta(c^k - 1)} \cdot \tau_{\beta^{-k}}.$$

2.3.3 Competitive Ratio

Theorem 11. *In the prophet IID setting, given an inflated cost factor $c > 1$ and a complete description of the distribution $\mathcal{D}$, there exists an algorithm that achieves a competitive ratio of $O(\exp(2\sqrt{\ln n \ln c}))$.*

Proof. Since $\beta^k = n$, we have $\beta = n^{1/k}$ and $\tau_{\beta^{-k}} = \tau_{1/n}$. Randomizing between the Single Threshold Algorithm and the Geometric Threshold Algorithm with equal probability 1/2, the overall expected score is at least:

$$\frac{1 - 1/e}{2}\left(\frac{c-1}{n^{1/k}(c^k - 1)}\tau_{1/n} + n\,\mathbb{E}\big[\max\big(X - c\tau_{1/n}, 0\big)\big]\right).$$

Using $c^k - 1 < c^k$, we obtain the simpler lower bound

$$\frac{1 - 1/e}{2}\left(\frac{c-1}{n^{1/k}c^k}\tau_{1/n} + n\,\mathbb{E}\big[\max\big(X - c\tau_{1/n}, 0\big)\big]\right).$$

It remains to choose k in order to minimize the loss factor $n^{1/k}c^k$. Taking logarithms, this is equivalent to minimizing

$$f(k) = \frac{\ln n}{k} + k \ln c.$$

Treating k as a continuous parameter, the minimum occurs when

$$k = \sqrt{\ln n / \ln c}.$$

With this choice, $f(k) = 2\sqrt{\ln n \ln c}$, and hence

$$n^{1/k}c^k = \exp\Bigl(2\sqrt{\ln n \ln c}\Bigr).$$

Therefore, the expected score is at least

$$\frac{1-1/e}{2}\left(\frac{c-1}{\exp\Bigl(2\sqrt{\ln n \ln c}\Bigr)}\tau_{1/n} + n\,\mathbb{E}\bigl[\max\bigl(X - c\tau_{1/n}, 0\bigr)\bigr]\right).$$

Therefore, the algorithm is $O(\exp(2\sqrt{\ln n \ln c}))$-competitive. $\square$

Remark 6. The proof above loses a factor of

$$n^{1/k} \cdot \frac{c^k - 1}{c-1}.$$

This expression separates the two sources of loss in the geometric-threshold argument. The factor $n^{1/k}$ comes from the spacing between consecutive quantile levels, while the factor

$$\frac{c^k-1}{c-1} = 1 + c + \cdots + c^{k-1}$$

comes from unrolling the recurrence $T_j \leq B + cT_{j-1}$.

As $c \to 1$, the geometric sum degenerates into an ordinary sum:

$$\lim_{c \to 1} \frac{c^k - 1}{c-1} = k.$$

The loss becomes approximately $n^{1/k}k$. Optimizing this expression gives $k \asymp \ln n$, and the resulting loss is the known result $O(\log n)$.

Chapter 3

General Prophet Setting

3.1 Preliminaries

We now move from the IID prophet setting to the general prophet setting. Each X_i is drawn from its own continuous distribution $\mathcal{D}_i$. This added generality creates one main complication: there is no single underlying distribution whose tail quantiles describe all options simultaneously. As a result, we will use two different kinds of quantiles throughout the chapter.

For each individual distribution $\mathcal{D}_i$ and each $\alpha \in (0, 1]$, define $\tau_\alpha^{(i)}$ as the smallest $(1 - \alpha)$-quantile of $\mathcal{D}_i$; equivalently, $\tau_\alpha^{(i)}$ is the minimal τ such that

$$\Pr_{X_i \sim \mathcal{D}_i}(X_i > \tau_\alpha^{(i)}) = \alpha.$$

This notation is used when we need to reason about the tail of the underlying distribution of a particular value.

We also define an additional random variable (we will colloquially call this the the prophet benchmark random variable)

$$X^{\max} := \max_{i \in [n]} X_i.$$

Since the variables are independent and each $\mathcal{D}_i$ is continuous, $X^{\max}$ also has a continuous distribution. We therefore write τ_α for the smallest $(1 - \alpha)$-quantile of $X^{\max}$.

In particular, the value $\tau_{1/2}$ is the median of the prophet's maximum:

$$\Pr\bigl(X^{\max} \le \tau_{1/2}\bigr) = \frac{1}{2}.$$

This median will serve as the central threshold in the chapter. The reason is that, although the individual distributions may be different, the event $X^{\max} > \tau_{1/2}$ captures the combined probability that at least one option has a large value. This gives a common scale against which all distributions can be compared.

As in the IID chapter, our algorithms are analyzed by comparing their expected utility to the prophet upper bound from Lemma 5. The later sections will evaluate this upper bound at $\theta = \tau_{1/2}$. Thus, the recurring goal is to design algorithms that recover two terms: a threshold term comparable to $\tau_{1/2}$ and a tail-expectation term comparable to

$$\sum_{i=1}^{n} \mathbb{E}\bigl[\max\bigl(X_i - c\tau_{1/2}, 0\bigr)\bigr].$$

The following single-threshold algorithm recovers the second term up to a constant factor.

Single Threshold Algorithm Consider the algorithm that uses the same threshold $\tau_{1/2}$ for every option.When option i arrives, the algorithm tests it against $\tau_{1/2}$. If no earlier option has passed and option i passes, the algorithm accepts it and obtains utility of $\max\{X_i - c\tau_{1/2}, 0\}$.

Fix an option $i \in [n]$ and let E_i represent the event that the algorithm reaches option i; equivalently, none of the previous options has passed the test. Then

$$\Pr(E_i) = \prod_{j<i} \Pr\bigl(X_j \le \tau_{1/2}\bigr).$$

Each probability is at most 1 and nonnegative. That is, removing factors can only increase the product. Hence,

$$\prod_{j<i} \Pr\bigl(X_j \le \tau_{1/2}\bigr) \ge \prod_{j=1}^{n} \Pr\bigl(X_j \le \tau_{1/2}\bigr).$$

By independence,

$$\prod_{j=1}^{n} \Pr\bigl(X_j \le \tau_{1/2}\bigr) = \Pr\bigl(X^{\max} \le \tau_{1/2}\bigr) = \frac{1}{2}.$$

Therefore,

$$\Pr(E_i) \ge \frac{1}{2}.$$

E_i depends only on $X_1, \ldots, X_{i-1}$ and is independent of X_i, the expected contribution from option i is

$$\Pr(E_i) \cdot \mathbb{E}\bigl[\max\bigl(X_i - c\tau_{1/2}, 0\bigr)\bigr] \ge \frac{1}{2}\,\mathbb{E}\bigl[\max\bigl(X_i - c\tau_{1/2}, 0\bigr)\bigr].$$

Summing over all options, the expected remaining utility is at least

$$\frac{1}{2}\sum_{i=1}^{n} \mathbb{E}\bigl[\max\bigl(X_i - c\tau_{1/2}, 0\bigr)\bigr].$$

This preliminary guarantee will be used repeatedly. It shows that a fixed threshold at the median of $X^{\max}$ captures the tail-expectation part of the prophet upper bound. The remaining work in each pricing scenario is to recover the threshold term $\tau_{1/2}$.

3.2 Discounted Price

We capitalize on the fact that testing is incredible cheap in this scenario, specifically in the single-threshold algorithm found above. If an option passes the threshold $\tau_{1/2}$, then its remaining utility is

$$X_i - c\tau_{1/2} = X_i - \tau_{1/2} + (1-c)\tau_{1/2}.$$

Thus a passing option provides both the usual surplus above $\tau_{1/2}$ and an additional discounted-cost gain of $(1-c)\tau_{1/2}$.

To evaluate our algorithm, we benchmark against the upper bound established in Lemma 5 evaluated at $\theta = \tau_{1/2}$ and with $c = 1$:

$$\mathbb{E}[X^{\max}] \le \tau_{1/2} + \sum_{i=1}^{n} \mathbb{E}\bigl[\max\bigl(X_i - \tau_{1/2}, 0\bigr)\bigr]. \tag{3.1}$$

3.2.1 Algorithm

We use the single threshold algorithm from the preliminary section with the same threshold $\tau_{1/2}$. When option i arrives, the algorithm tests it against $\tau_{1/2}$. If no previous option has passed the test and option i passes, the algorithm accepts option i and obtains utility $X_i - c\tau_{1/2}$.

By the single-threshold calculation in the preliminary section, the expected utility is at least

$$\frac{1}{2}\sum_{i=1}^{n}\mathbb{E}\big[\big(X_i - c\tau_{1/2}\big)\mathbb{1}\big[X_i > \tau_{1/2}\big]\big].$$

Using our rewritten utility $X_i - c\tau_{1/2} = X_i - \tau_{1/2} + (1-c)\tau_{1/2}$ and $p_i = \Pr(X_i > \tau_{1/2})$,

$$\begin{aligned}\mathbb{E}\big[\big(X_i - c\tau_{1/2}\big)\mathbb{1}\big[X_i > \tau_{1/2}\big]\big] &= \mathbb{E}\big[\big(X_i - \tau_{1/2}\big)\mathbb{1}\big[X_i > \tau_{1/2}\big]\big] + \tau_{1/2}(1-c)\cdot p_i \\ &= \mathbb{E}\big[\max\big(X_i - \tau_{1/2}, 0\big)\big] + \tau_{1/2}(1-c)\cdot p_i.\end{aligned}$$

Summing over all options, the expected utility of the algorithm is at least

$$\frac{1}{2}\sum_{i=1}^{n}\mathbb{E}\big[\max\big(X_i - \tau_{1/2}, 0\big)\big] + \frac{(1-c)\tau_{1/2}}{2}\sum_{i=1}^{n} p_i.$$

It remains to lower bound $\sum_i p_i$. Since $\tau_{1/2}$ is the median of $X^{\max}$, we have

$$\prod_{i=1}^{n}(1-p_i) = \Pr\big(X^{\max} \le \tau_{1/2}\big) = \frac{1}{2}$$

Using the Weierstrass product inequality,

$$\prod_{i=1}^{n}(1-p_i) \ge 1 - \sum_{i=1}^{n} p_i.$$

Hence,

$$\frac{1}{2} = \prod_{i=1}^{n}(1-p_i) \ge 1 - \sum_{i=1}^{n} p_i,$$

and therefore

$$\sum_{i=1}^{n} p_i \ge \frac{1}{2}.$$

Substituting this into the expected-utility lower bound gives

$$\frac{1-c}{4}\tau_{1/2} + \frac{1}{2}\sum_{i=1}^{n} \mathbb{E}\big[\max\big(X_i - \tau_{1/2}, 0\big)\big].$$

3.2.2 Competitive Ratio

Theorem 12. *In the general prophet setting, given a discounted cost factor $c \in (0, 1)$ and complete descriptions of the distributions $(\mathcal{D}_i)_{i=1}^n$, there exists an algorithm that achieves a competitive ratio of $O(1)$.*

Proof. Since $c \in (0, 1)$, we have $\frac{1-c}{4} \leq \frac{1}{2}$. Therefore, our expected utility lower bound is at least

$$\frac{1-c}{4}\left(\tau_{1/2} + \sum_{i=1}^{n} \mathbb{E}\big[\max\big(X_i - \tau_{1/2}, 0\big)\big]\right).$$

Comparing with (3.1), the algorithm obtains at least a $\frac{1-c}{4}$ fraction of the prophet upper bound. Equivalently, its competitive ratio is at most $\frac{4}{1-c}$. Hence the algorithm is $O(1)$-competitive. □

3.3 A Simpler Argument for $c = 1$

Before we begin discussing the inflated price scenario, we revisit the original cost factor $c = 1$. This section should be viewed as a bridge between the discounted argument above and the inflated argument below. The goal is to give a cleaner proof of the same guarantee using a simpler approach structure that will aid our discussion and adapt more naturally to the inflated price setting.

Evaluating the prophet upper bound at the median $\tau_{1/2}$ of $X^{\max}$ gives

$$\mathbb{E}[X^{\max}] \leq \tau_{1/2} + \sum_{i=1}^{n} \mathbb{E}\big[\max\big(X_i - \tau_{1/2}, 0\big)\big].$$

The single-threshold algorithm from the preliminary section captures the second term up to a constant factor. Thus, the only remaining task is to recover the $\tau_{1/2}$ term, up to the desired logarithmic loss.

In the original proof, this is done by partitioning the n distributions into $k+3$ buckets according to their tail probabilities $\alpha_i = \Pr\big(X_i > \tau_{1/2}\big)$, then selecting a bucket with sufficiently large aggregate tail mass and applying thresholds tailored to the distributions in that bucket. Although effective, it is far more elaborate than necessary and the approach to partitioning does not lend itself for the inflated price setting.

Instead, we propose and showcase a simpler significant-group argument. We keep all distributions whose probability of exceeding $\tau_{1/2}$ is at least $1/4n$ and discard the rest. Since the total tail mass above $\tau_{1/2}$ is at least $1/2$, the remaining significant set still has constant total tail mass. The remaining part is to illustrate that for a distribution with $\alpha = \Pr(X > \tau)$, there is a threshold $\theta \leq \tau$ so that the option passes with probability at least α, while the expected remaining utility conditional on passing is still a logarithmic fraction of τ.

Our lemma below proves this and serves as replacement for the bucketing step in the original proof.

Lemma 13. *For any continuous distribution $\mathcal{D}$, target value $\tau > 0$, and probability $\alpha = \Pr_{X\sim\mathcal{D}}(X > \tau)$, there exists a threshold $\theta \in [0, \tau]$ such that:*

1. $\Pr_{X\sim\mathcal{D}}(X > \theta) \geq \alpha$ *(guaranteed by construction).*
2. $\mathbb{E}_{X\sim\mathcal{D}}[X - \theta \mid X > \theta] \geq \frac{\tau}{2(1-\ln\alpha)}$.

Proof. Let $F(x) = \Pr(X > x)$ and define

$$H(\theta) = \int_{\theta}^{\tau} F(x)dx.$$

The quantity $H(\theta)$ measures the amount of tail mass between the test threshold θ and the target value τ. Since,

$$\mathbb{E}[X - \theta \mid X > \theta] = \frac{\int_{\theta}^{\infty} F(x)\,dx}{F(\theta)} \geq \frac{H(\theta)}{F(\theta)},$$

it is enough to show that $H(\theta)/F(\theta)$ is large for some $\theta \in [0, \tau]$.

Define

$$C = \max_{\theta\in[0,\tau]} \frac{H(\theta)}{F(\theta)}.$$

Since $H'(\theta) = -F(\theta)$, the definition of C implies that, for every $\theta \in [0, \tau]$,

$$\frac{H(\theta)}{F(\theta)} \leq C \implies \frac{H(\theta)}{C} \leq F(\theta) = -H'(\theta).$$

Equivalently,

$$H'(\theta) \le -\frac{H(\theta)}{C}.$$

Solving this differential inequality gives

$$H(\theta) \le H(0)e^{-\theta/C}.$$

Since $F(x) \le 1$ for all x, we have $H(0) \le \tau$, and therefore

$$H(\theta) \le \tau e^{-\theta/C}.$$

Since F is non-increasing and $F(\tau) = \alpha$,

$$H(\theta) = \int_\theta^\tau F(x)\,dx \ge (\tau - \theta)F(\tau) = (\tau - \theta)\alpha.$$

Combining the upper and lower bounds, for every $\theta \in [0, \tau]$,

$$(\tau - \theta)\alpha \le \tau e^{-\theta/C}.$$

Now choose $\theta = \tau - C$. This is valid because $C \le \tau$ as $H(\theta) \le \tau F(\theta)$ for every $\theta \in [0, \tau]$, so $H(\theta)/F(\theta) \le \tau$. Substituting $\theta = \tau - C$ gives

$$C\alpha \le \tau e^{-(\tau - C)/C} = \tau e^{1-\tau/C}.$$

Dividing by e and setting $y = \tau/C$, we obtain

$$\frac{\alpha}{e} \le y e^{-y}.$$

Taking logarithms gives

$$y - \ln y \le \ln(e/\alpha).$$

Using the elementary inequality $\ln y \le y/2$ for all $y > 0$, we get

$$\frac{y}{2} \le \ln(e/\alpha).$$

Since $y = \tau/C$, this implies

$$C \ge \frac{\tau}{2\ln(e/\alpha)} = \frac{\tau}{2(1 - \ln \alpha)}.$$

Therefore, some threshold $\theta^* \in [0, \tau]$ satisfies

$$\frac{H(\theta^*)}{F(\theta^*)} \geq \frac{\tau}{2(1 - \ln \alpha)}.$$

Finally, because $\theta^* \leq \tau$ and F is non-increasing,

$$\Pr(X > \theta^*) = F(\theta^*) \geq F(\tau) = \alpha.$$

$\square$

We will target the following comparative upper bound:

$$\mathbb{E}[X^{\max}] \leq \tau_{1/2} + \sum_{i=1}^{n} \mathbb{E}\big[\max\big(X_i - \tau_{1/2}, 0\big)\big]. \tag{3.2}$$

3.3.1 Significant Group Algorithm

We now apply the threshold lemma to the collection of distributions. Let

$$\alpha_i = \Pr_{X_i \sim \mathcal{D}_i}\big(X_i > \tau_{1/2}\big).$$

The value α_i measures how much distribution $\mathcal{D}_i$ contributes to the upper tail of the prophet maximum at the benchmark scale $\tau_{1/2}$. Since $\tau_{1/2}$ is the median of $X^{\max}$,

$$\frac{1}{2} = \Pr\big(X^{\max} \leq \tau_{1/2}\big) = \prod_{i=1}^{n} \Pr\big(X_i \leq \tau_{1/2}\big) = \prod_{i=1}^{n} (1 - \alpha_i)$$

Using the Weierstrass product inequality,

$$\prod_{i=1}^{n} (1 - \alpha_i) \geq 1 - \sum_{i=1}^{n} \alpha_i.$$

Therefore,

$$\frac{1}{2} \geq 1 - \sum_{i=1}^{n} \alpha_i,$$

and finally,

$$\sum_{i=1}^{n} \alpha_i \geq \frac{1}{2}.$$

The total tail mass above $\tau_{1/2}$ is therefore at least 1/2. However, some of this mass may come from distributions whose individual tail probabilities are extremely small. We ignore these negligible distributions and keep only the significant ones:

$$S = \left\{ i \in [n] : \alpha_i \geq \frac{1}{4n} \right\}.$$

The sum of the tail probabilities of the inconsequential options is strictly bounded:

$$\sum_{i \notin S} \alpha_i < n \left(\frac{1}{4n} \right) = \frac{1}{4}.$$

Thus, the sum of probabilities in set S remains substantial:

$$\sum_{i \in S} \alpha_i = \sum_{i=1}^{n} \alpha_i - \sum_{i \notin S} \alpha_i \geq \frac{1}{2} - \frac{1}{4} = \frac{1}{4}.$$

The significant-group algorithm is now defined as follows:

1. If $i \notin S$, reject option i without testing.
2. For each $i \in S$, apply Lemma 13 to distribution $\mathcal{D}_i$ with target value $\tau = \tau_{1/2}$ and probability $\alpha = \alpha_i$. Let $\theta_i \in [0, \tau_{1/2}]$ be the threshold guaranteed by the lemma.
3. Test option i using threshold θ_i, and accept the first option in S that passes.

Let $p_i = \Pr_{X_i \sim \mathcal{D}_i}(X_i > \theta_i)$. By the first guarantee of Lemma 13, we have $p_i \geq \alpha_i$ for every $i \in S$. The probability that at least a single option from the set S passes is

$$\begin{aligned} 1 - \prod_{i \in S}(1 - p_i) &\geq 1 - \exp\left(-\sum_{i \in S} p_i \right) \\ &\geq 1 - \exp\left(-\sum_{i \in S} \alpha_i \right) \\ &\geq 1 - e^{-1/4}. \end{aligned}$$

For every $i \in S$, we have $\alpha_i \geq \frac{1}{4n}$. Lemma 13 guarantees the conditional expected reward is bounded by

$$\mathop{\mathbb{E}}_{X_i \sim \mathcal{D}_i}[X_i - \theta_i \mid X_i > \theta_i] \geq \frac{\tau_{1/2}}{2(1 - \ln \alpha_i)} = \frac{\tau_{1/2}}{2 \ln(e/\alpha_i)} \geq \frac{\tau_{1/2}}{2 \ln(4en)}.$$

Thus, whenever the algorithm accepts an option from S, its conditional expected utility is at least

$$\frac{\tau_{1/2}}{2\ln(4en)}.$$

Combining this utility lower bound with the probability of accepting some option from S, the expected utility of the significant-group algorithm is at least

$$\left(1 - e^{-1/4}\right) \cdot \frac{\tau_{1/2}}{2\ln(4en)}.$$

3.3.2 Competitive Ratio

Theorem 14 (Theorem 1 of [QV23]). *In the general prophet setting with cost factor $c = 1$, there exists an algorithm that achieves an $O(\log n)$ competitive ratio.*

Proof. We construct our final algorithm by randomizing between the mechanism described in the Single Threshold Algorithm and Section 3.3.1, choosing each with an equal probability of $1/2$. The overall expected score of the player is at least:

$$\frac{1}{4}\left(\frac{(1 - e^{-1/4})}{\ln(4ne)} \cdot \tau_{1/2} + \sum_{i=1}^{n} \mathop{\mathbb{E}}_{X_i \sim \mathcal{D}_i}\left[\max\left(X_i - \tau_{1/2}, 0\right)\right]\right).$$

Hence the combined algorithm obtains an $\Omega(1/\log n)$ fraction of the prophet benchmark in Equation 3.2, and therefore achieves an $O(\log n)$ competitive ratio. □

3.4 Inflated Price

The structure of the proof follows from the significant-group argument in Section 3.3.1, with an amendment to the threshold guarantee lemma. In this price scenario, testing at θ leaves a utility of $\max\{X - c\theta, 0\}$. Thus, it is no longer enough for X to be slightly larger than θ; the realized value must clear the larger effective cost $c\theta$.

We use a geometric ladder of quantiles, as in in the IID inflated price scenario found in Section 2.3.2. For a single distribution, we look at consecutive quantile thresholds $T_0, T_1, \ldots, T_k$. If we test at T_{j-1} and the value reaches the next threshold T_j, then the certified remaining utility is roughly $T_j - cT_{j-1}$.

The main point of the following lemma is that, among the consecutive pairs in this ladder, one pair must have a sufficiently large marginal gain. The proof is the same recurrence argument used in the IID inflated-price setting, but now applied each distribution individually.

Lemma 15. *For any continuous distribution $\mathcal{D}$, let $\tau > 0$, and define its corresponding probability $\alpha = \Pr_{X\sim\mathcal{D}}(X > \tau)$. Given a cost factor $c > 1$ and fix a spacing parameter $\beta > 1$. Let*

$$k \geq \frac{\ln(1/\alpha)}{\ln \beta}.$$

There exists a threshold $\theta \in [0, \tau]$ such that:

1. $\Pr_{X\sim\mathcal{D}}(X > \theta) \geq \alpha$.
2. $\mathbb{E}_{X\sim\mathcal{D}}[\max\{X - c\theta, 0\} \mid X > \theta] \geq \frac{\tau(c-1)}{\beta(c^k-1)}$.

Proof. For $j = 0, 1, \ldots, k$, define $\alpha_j = \beta^{-j}$ and let $T_j = \tau_{\alpha_j} = \tau_{\beta^{-j}}$ be the corresponding sequence of quantile thresholds. Thus T_j is the smallest value such that

$$\Pr_{X\sim\mathcal{D}}(X > T_j) = \beta^{-j}.$$

We also take $T_0 = \tau_1 = 0$. As a result of our definition of k, we guarantee that $\beta^{-k} \leq \alpha$. Since smaller tail probabilities correspond to larger quantiles, this implies

$$T_k = \tau_{\beta^{-k}} \geq \tau_\alpha = \tau.$$

On the other hand,

$$\beta^{-(k-1)} > \alpha,$$

so

$$T_{k-1} = \tau_{\beta^{-(k-1)}} < \tau.$$

Therefore every threshold among $T_0, \ldots, T_{k-1}$ lies in $[0, \tau]$. Define the maximum marginal gain across consecutive thresholds

$$B = \max_{j\in[k]}(T_j - cT_{j-1}, 0).$$

Necessarily $T_j - cT_{j-1} \leq B$ for every $j \in [k]$, we have

$$T_j \leq B + cT_{j-1}.$$

Unrolling this recurrence from $T_0 = 0$ gives

$$T_k \leq B(1 + c + c^2 + \cdots + c^{k-1}) = B\frac{c^k - 1}{c - 1}.$$

Rearranging,

$$B \geq T_k \frac{c-1}{c^k - 1}.$$

Since $T_k \geq \tau$, we get

$$B \geq \tau \frac{c-1}{c^k - 1}.$$

Since B is defined as the maxima over all j, there exists a $j^* \in [k]$ that attains the maxima such that

$$B = T_{j^*} - cT_{j^*-1}.$$

Set $\theta = T_{j^*-1}$. As observed above, $\theta \in [0, \tau]$. Moreover,

$$\Pr(X > \theta) = \Pr(X > T_{j^*-1}) = \beta^{-(j^*-1)} \geq \beta^{-(k-1)} > \alpha$$

This proves the first condition.

It remains to lower bound the conditional expected utility. Conditional on $X > T_{j^*-1}$, the probability that X also exceeds the next threshold T_{j^*} is

$$\Pr(X > T_{j^*} \mid X > T_{j^*-1}) = \frac{\beta^{-j^*}}{\beta^{-(j^*-1)}} = \frac{1}{\beta}.$$

By the fact that $X > T_{j^*}$, we obtain

$$\max\{X - cT_{j^*-1}, 0\} \geq T_{j^*} - cT_{j^*-1} = B$$

Therefore,

$$\mathbb{E}[\max\{X - cT_{j^*-1}, 0\} \mid X > T_{j^*-1}] \geq \frac{B}{\beta}.$$

Finally, using the lower bound obtained on B, we can conclude

$$\mathbb{E}[\max\{X - c\theta, 0\} \mid X > \theta] \geq \frac{1}{\beta} \cdot \tau \frac{c-1}{c^k - 1} = \frac{\tau(c-1)}{\beta(c^k - 1)}.$$

□

We will target the following comparative upper bound:

$$\mathbb{E}[X^{\max}] \le c\tau_{1/2} + \sum_{i=1}^{n} \mathbb{E}\big[\max\big(X_i - c\tau_{1/2}, 0\big)\big]. \tag{3.3}$$

As before, the single-threshold algorithm from the preliminary section captures the summation term up to a constant factor. The significant-group algorithm below captures the threshold term $c\tau_{1/2}$.

3.4.1 Significant Group Algorithm

As we have derived before, we claim given $\alpha_i = \Pr_{X_i \sim \mathcal{D}_i}(X_i > \tau_{1/2})$ that $\sum_{i=1}^{n} \alpha_i \ge \frac{1}{2}$. As in the $c = 1$ argument, define the significant set

$$S = \left\{ i \in [n] : \alpha_i \ge \frac{1}{4n} \right\}.$$

The sum of the pruned options is strictly smaller than

$$n \cdot \frac{1}{4n} = \frac{1}{4}.$$

Thus, the sum of the remaining options (the ones in S) must be larger than

$$\sum_{i \in S} \alpha_i \ge \frac{1}{2} - \frac{1}{4} = \frac{1}{4}.$$

Fix a spacing parameter $\beta > 1$, to be optimized later. The significant-group algorithm proceeds as follows:

1. If $i \notin S$, reject option i without testing.

2. For each $i \in S$, apply Lemma 15 to distribution $\mathcal{D}_i$ with target value $\tau = \tau_{1/2}$, tail probability $\alpha = \alpha_i$, cost factor c, and spacing parameter β. Let $\theta_i \in [0, \tau_{1/2}]$ be the threshold guaranteed by the lemma.

3. Test option i using threshold θ_i, and accept the first option in S that passes.

Let $p_i = \Pr(X_i > \theta_i)$. Given by the fist guarantee of Lemma 15, we know $p_i \geq \alpha_i$. Thus the probability that at least one option in S passes is

$$\begin{aligned} 1 - \prod_{i \in S}(1 - p_i) &\geq 1 - \exp\left(-\sum_{i \in S} p_i\right) \\ &\geq 1 - \exp\left(-\sum_{i \in S} \alpha_i\right) \\ &\geq 1 - e^{-1/4}. \end{aligned}$$

Now fix $i \in S$. Since $\alpha_i \geq 1/(4n)$, the parameter

$$k_i = \left\lceil \frac{\ln(1/\alpha_i)}{\ln \beta} \right\rceil \leq \frac{\ln(4n)}{\ln \beta} + 1.$$

By Lemma 15, conditional on option i passing its test,

$$\mathbb{E}[\max(X_i - c\theta_i, 0) \mid X_i > \theta_i] \geq \frac{\tau_{1/2}}{\beta} \cdot \frac{c-1}{c^{k_i} - 1}.$$

Using $c^{k_i} - 1 < c^{k_i}$ and the upper bound on k_i, we get

$$\begin{aligned} \mathbb{E}[\max(X_i - c\theta_i, 0) \mid X_i > \theta_i] &\geq \frac{\tau_{1/2}}{\beta} \cdot \frac{c-1}{c^{k_i}} \\ &\geq \frac{\tau_{1/2}}{\beta} \cdot \frac{c-1}{c^{\frac{\ln(4n)}{\ln \beta}+1}}. \end{aligned}$$

Rewriting the denominator using the identity $a^{\ln x/\ln b} = x^{\ln a/\ln b}$

$$\beta c^{\frac{\ln(4n)}{\ln \beta}+1} = \beta c (4n)^{\frac{\ln c}{\ln \beta}}$$

Expanding out this term,

$$\beta c (4n)^{\frac{\ln c}{\ln \beta}} = c \cdot \exp\left(\ln \beta + \frac{\ln(4n) \cdot \ln c}{\ln \beta}\right)$$

Therefore every accepted option from S has conditional expected utility at least

$$\tau_{1/2} \cdot \frac{c-1}{c} \cdot \exp\left(-\ln \beta - \frac{\ln(4n) \cdot \ln c}{\ln \beta}\right).$$

We now optimize the choice of β. Let $x = \ln \beta$. The terms inside the exponent can be expressed as

$$f(x) = x + \frac{\ln(4n)\ln c}{x}.$$

This is minimized at

$$x = \sqrt{\ln(4n)\ln c},$$

so we choose

$$\beta = \exp\Big(\sqrt{\ln(4n)\ln c}\Big).$$

Thus, conditional on accepting an option from S, the expected utility is at least

$$\tau_{1/2} \cdot \frac{c-1}{c} \cdot \exp\Big(-2\sqrt{\ln(4n)\ln c}\Big).$$

Combining this with the probability of accepting some option from S, the expected utility of the significant-group algorithm is at least

$$\big(1 - e^{-1/4}\big)\tau_{1/2} \cdot \frac{c-1}{c} \cdot \exp\Big(-2\sqrt{\ln(4n)\ln c}\Big).$$

3.4.2 Competitive Ratio

Theorem 16. *In the prophet setting, given complete descriptions of the distributions $(\mathcal{D}_i)_{i=1}^n$ and an inflated cost factor $c > 1$, there exists an algorithm that achieves a competitive ratio of $O(\exp(2\sqrt{\ln n \ln c}))$.*

Proof. We construct our final algorithm by randomizing the mechanism described in the Single Threshold Algorithm and the Significant Group Algorithm, choosing each with a probability of 1/2. The overall expected remaining utility is at least:

$$\frac{1}{2}\left(\big(1 - e^{-1/4}\big)\tau_{1/2} \cdot \frac{c-1}{c} \cdot \exp\Big(-2\sqrt{\ln(4n) \cdot \ln c}\Big) + \frac{1}{2}\sum_{i=1}^{n} \mathop{\mathbb{E}}_{X_i \sim \mathcal{D}_i}\big[\max\big(X_i - c\tau_{1/2}, 0\big)\big]\right).$$

Comparing this expected score to the upper bound established in Equation (3.3), the optimal expected score are match within a factor of $O(\exp(2\sqrt{\ln n \ln c}))$. Thus, this combined strategy achieves a competitive ratio of $O(\exp(2\sqrt{\ln n \ln c}))$. □

Chapter 4

Secretary Setting

Unlike the prophet setting, the values are no longer drawn independently from known distributions. Instead, the values shown to the player are a permutation of a fixed multiset $\{a_1, \ldots, a_n\}$. The difficulty of the setting arises in two distinct ways: the order in which the values arrive, and the amount of information the player has about the multiset before the game begins.

The chapter is organized around these two variations. We first define the information models and the competitive-ratio benchmark. We then handle the discounted price scenario. In this pricing scenario, the proofs are relatively direct. The inflated price scenario is more delicate. In our discussion we proceed from easier settings to harder ones. We begin with random order and full information, where the player can use the known values to construct a geometric ladder of thresholds. We then show how to preserve the same guarantee under arbitrary order when full information is available. We recover the same guarantee in informationally constrained environments, that is when only the maximum value is provided and even when no information is presented, given that the values are shown in uniformly random order. For the hardest setting we tackle, arbitrary ordering given optimum information, we generalize the bit sampling game lemma as seen in Qiao and Valiant's paper [QV23]. We obtain a slightly worse guarantee, when compared to everything else.

4.1 Preliminaries

In the secretary setting, the values of the multiset $\{a_1, \ldots, a_n\}$ are fixed. The sequence $X_1, \ldots, X_n$ shown to the player is a permutation of the elements of the multiset. Let $a_{[i]}$ denote the i-th largest value among $a_1, \ldots, a_n$. The player's goal is to obtain expected remaining utility comparable to $a_{[1]}$.

There are two possible assumptions on the arrival order.

1. Random order: the sequence $X_1, \dots, X_n$ is a uniformly random permutation of $a_1, \dots, a_n$.

2. Arbitrary order: the sequence $X_1, \dots, X_n$ may be any fixed permutation of $a_1, \dots, a_n$.

We also consider three information models

1. Full Information: the player is given $\{a_1, \dots, a_n\}$.

2. Optimum Information: the player is given $a_{[1]}$.

3. No information: the player is given no information.

These distinctions matter because they determine what thresholds the player can construct. With full information, the player can build thresholds from the ordered values themselves. With only optimum information, the player can construct thresholds only as functions of $a_{[1]}$. With no information, the player must first learn a scale from the sequence itself.

Definition 17 (α-Competitive Algorithm)**.** The player is α-competitive in the random order case if its expected score, over the randomness in both the permutation and the player itself, is at least

$$\frac{1}{\alpha} \cdot a_{[1]}.$$

The player is α-competitive in the arbitrary order case if for any permutation $(X_i)_{i=1}^n$ of $(a_i)_{i=1}^n$, the player's expected score, over the randomness in the player itself, is at least

$$\frac{1}{\alpha} \cdot a_{[1]}.$$

In the discounted pricing scenario, the main observation is that a passing option at threshold θ has remaining utility at least $(1-c)\theta$. In the inflated pricing scenario, we instead have to construct threshold ladders, similar to what we did in our discussion of the inflated price scenario in the prophet setting. The quantity that repeatedly appears is the marginal gain $b_j - cb_{j-1}$, where b_{j-1} is the testing threshold and b_j is the next higher target scale.

4.2 Discounted Price

In this scenario, testing is relatively cheap: if an option passes a threshold θ, then its remaining utility is at least

$$\theta - c\theta = (1-c)\theta.$$

A single well-chosen threshold leaves a constant fraction of that threshold as remaining utility.

There are two cases to prove. First, if the player knows the optimum value $a_{[1]}$, then the player can simply test every option at threshold $a_{[1]}$. This works even in arbitrary order, and therefore also proves the corresponding random-order and full-information cases. Second, if the player has no information, then the arbitrary-order setting has no nontrivial guarantee beyond the trivial $O(n)$ baseline discussed in the summary of results. The remaining case is therefore random order with no information. There, we use the first half of the sequence to estimate the scale of the maximum and the second half to select an option.

4.2.1 Arbitrary Order, Optimum Information

Theorem 18. *In the secretary setting with optimum information and a discounted cost factor $c \in (0,1)$, there exists an algorithm that achieves an $O(1)$ competitive ratio, even under an arbitrary arrival order.*

Proof. If $a_{[1]} = 0$, all options have a value of 0, so we accept the first pen and obtain the maximal value. We will assume for the remainder of the proof that $a_{[1]} > 0$.

The algorithm is simple:

1. Set a threshold of $\theta = a_{[1]}$.
2. Test each option X_i with θ.
3. Accept the first option that passes the test (i.e. $X_i \geq \theta$).

By the problem setting we know that $a_{[1]}$ must appear somewhere in the sequence of options shown to us. Since $a_{[1]} \geq \theta = a_{[1]}$, the option that is the value of $a_{[1]}$ is guaranteed to pass the test. Thus, if no earlier option passes the test, $a_{[1]}$ will inevitably pass. This guarantees that the algorithm will not exhaust the sequence and will accept some option.

Let X_j be the option that the algorithm accepts. Since it passed the test, it must satisfy $X_j \geq \theta$. Thefore, the remaining utility is:

$$\begin{aligned} X_j - c\theta &\geq \theta - c\theta \\ &= \theta(1-c) \\ &= a_{[1]}(1-c). \end{aligned}$$

Thus, the algorithm is $\frac{1}{1-c}$-competitive. Since the cost factor c is a constant in $(0,1)$, this yields an expected score that is an $\Omega(1)$ fraction of the optimum. □

Remark 7. This theorem immediately implies the same $O(1)$ guarantee for the random-order setting with optimum information, since arbitrary order is the stronger model. It also implies the full-information cases, because full information includes knowledge of $a_{[1]}$.

4.2.2 Random Order, No Information

Theorem 19. *In the secretary setting under random order with no information and a discount factor $c \in (0,1)$, there exists an algorithm that achieves an $O(1)$ competitive ratio, assuming $a_{[2]} > 0$.*

Proof. We assume $a_{[2]} > 0$; otherwise, if $a_{[1]}$ is the only non-zero value, any online algorithm without any additional information cannot set a meaningful threshold as the only value to observed is the maximal one.

We split our approach into two phases. First, the player first observes the first $m = \lfloor n/2 \rfloor$ options, allowing them to observe the exact value of each option (we achieve this by setting $\theta = +\infty$). Let $M = \max_{i \in [m]} X_i$ be the maximum value during this first phase.

For the second phase, the remaining $n-m$ options, we flip a fair coin to determine which of two strategies below the player will employ:

1. **Strategy 1:** Set threshold $\theta = 2M$. Accept the first option X_i that passes the test (i.e., $X_i \geq 2M$).
2. **Strategy 2:** Set threshold $\theta = M/2$. Accept the first option X_i that passes the test (i.e., $X_i \geq M/2$).

The two thresholds handle two different possibilities. If the maximum $a_{[1]}$ is much larger than the second-largest value $a_{[2]}$, then the high threshold $2M$ isolates the maximum. If $a_{[1]}$ is within a constant factor of $a_{[2]}$, then the lower threshold $M/2$ guarantees that any passing option still has value comparable to $a_{[1]}$.

Our analysis of these strategies is contingent on the event that $a_{[1]}$ is found in the second half of the sequence and $a_{[2]}$ is in the first half. Call this event $\mathcal{E}$. The probability of this occurring is

$$\frac{m}{n} \cdot \frac{n-m}{n-1}.$$

Since $m = \lfloor n/2 \rfloor$, necessarily $m \geq (n-1)/2$ and $n - m \geq n/2$, and thus we claim that the probability is lower bounded by

$$\frac{n-1}{2n} \cdot \frac{n}{2(n-1)} = \frac{1}{4}.$$

Conditioning on this event occurring, we examine two exhaustive cases based on the relation between $a_{[1]}$ and $a_{[2]}$.

Case 1 ($a_{[1]} > 2a_{[2]}$): If this is the case, we will rely on Strategy 1. Given that we set the threshold to $\theta = 2M = 2a_{[2]}$, the maximal option $a_{[1]}$ will pass the test. Furthermore, since $a_{[2]}$ is the second-largest value, every other option in the second half of the sequence will not pass the test besides $a_{[1]}$. The remaining utility is:

$$a_{[1]} - c\theta = a_{[1]} - 2ca_{[2]} > a_{[1]} - ca_{[1]} = a_{[1]}(1-c).$$

Accounting for the probability of $\mathcal{E}$ and the probability $1/2$ of choosing Strategy 1, the expected utility in this case is at least

$$\frac{a_{[1]}}{4}(1-c).$$

Case 2 ($a_{[1]} \leq 2a_{[2]}$): If this is the case, we will rely on Strategy 2. Given that we set the threshold to $\theta = M/2 = a_{[2]}/2$, the maximal option $a_{[1]}$ will pass the test. Unlike Case 1, there might be other options in the second half of the sequence that could also pass the test (e.g. if $a_{[2]} = a_{[3]}$). Nonetheless, any option X_j that passes the test, necessarily $X_j \geq \theta$. The remaining utility is:

$$X_j - c\theta \geq \frac{a_{[2]}}{2} - c\frac{a_{[2]}}{2} = \frac{a_{[2]}}{2}(1-c).$$

Since we are examining the case where $a_{[1]} \leq 2a_{[2]}$, this implies $a_{[2]} \geq a_{[1]}/2$. Therefore, the utility is at least

$$\frac{a_{[1]}}{4}(1-c).$$

Accounting for the probability of $\mathcal{E}$ and the probability 1/2 of choosing Strategy 2, the expected utility in this case is at least

$$\frac{a_{[1]}}{16}(1-c).$$

In both cases, the expected utility is at least

$$\frac{1-c}{32} \cdot a_{[1]}.$$

Thus, the algorithm is $\frac{32}{1-c}$-competitive. Since the cost factor is $c \in (0,1)$ is a constant, we achieve an expected score that is an $\Omega(1)$ fraction of the optimum. $\square$

4.3 Inflated Price

This is the more delicate secretary setting because passing a test at threshold θ no longer guarantees positive utility. An option that passes only tells us that $X_i \geq \theta$, but the remaining utility is $\max\{X_i - c\theta, 0\}$. Thus, to obtain positive value, the accepted option must be substantially larger than the threshold used to test it.

The recurring idea in this section is to create a ladder of thresholds $b_0, b_1, \ldots, b_k$ and to test at one level while hoping to accept an option from the next higher level. If we test at b_{j-1} and accept an option with value at least b_j, then the remaining utility is at least $b_j - cb_{j-1}$. The analysis therefore focuses on finding a scale j for which this marginal gain is large and for which the algorithm has a reasonable chance of accepting an option from the higher level.

We begin our discussion in the easiest setting: random order and full information as a warm-up. We then adapt the same threshold-ladder argument to arbitrary order with full information by explicitly sampling among the passing options. With only optimum information, the thresholds can no longer be chosen from the sorted values, so we construct them using only $a_{[1]}$. We will finally tackle the hardest settings: "arbitrary order, optimum information" and "random order, no information" by introducing additional ideas to the algorithm design.

Across these settings, the target is an

$$O\left(\exp\left(2\sqrt{\ln n \ln c}\right)\right)$$

competitive ratio, up to an additional logarithmic factor in the arbitrary-order optimum-information case (that is absorbed in our presentation of the big-O).

4.3.1 Random Order, Full Information

Theorem 20. *In the secretary setting under random order with full information and an inflated cost factor $c > 1$, there exists an algorithm that achieves a competitive ratio of $O(\exp(2\sqrt{\ln n \ln c}))$.*

Proof. If $a_{[1]} = 0$, then all values are zero and the result is trivial. Assume $a_{[1]} > 0$.

Let $k = \lceil \sqrt{\ln n / \ln c} \rceil$ and $\beta = n^{1/k}$. The parameter β controls how quickly the threshold ladder moves through the sorted values. Since $\beta^k = n$, moving down the ladder by k steps takes us from the whole set of options to the top option. For $j = 1, \dots, k$, define $i_j = \lfloor n\beta^{-j} \rfloor$ with the convention that $i_k = 1$, and set $b_j = a_{[i_j]}$. Thus, at least i_j options have value at least b_j. We also set $b_0 = 0$. This artificial choice of b_0 is only used to start the recurrence below. It does not mean that $a_{[n]} = 0$; rather, it gives the threshold ladder a clean base point.

The purpose of the ladder is to find two consecutive levels whose marginal gain is large. Define

$$B = \max_{j \in [k]} (b_j - c b_{j-1}).$$

For every $j \in [k]$, we have $b_j - cb_{j-1} \le B$ and consequently $b_j \le B + cb_{j-1}$. Unrolling this recurrence from $b_0 = 0$ gives

$$b_k \le B\left(1 + c + c^2 + \dots + c^{k-1}\right) = B\frac{c^k - 1}{c - 1}.$$

Since $b_k = a_{[1]}$, rearranging yields

$$B \ge a_{[1]}\frac{c-1}{c^k - 1}.$$

The algorithm proceeds as the following:

1. Select $j^* = \operatorname{argmax}_{j \in [k]}(b_j - cb_{j-1})$
2. Set the threshold $\theta = b_{j^*-1}$
3. Test every option X_i with θ and accept the first option that passes, given that $X_i - c\theta > 0$.

It remains to lower bound the expected utility of the accepted option. Consider the set of options with value at least b_{j^*-1}. There are at most i_{j^*-1} such options by construction, and every option with value at least b_{j^*} is contained in this set. Since the arrival order is uniformly random, the first passing option is uniformly distributed over the options that pass the threshold. Therefore, the expected utility is at least the average contribution from the top i_{j^*} options among the passing set:

$$\mathbb{E}[\max(X - c\theta, 0) \mid X \geq \theta] \geq \frac{1}{i_{j^*-1}} \cdot \sum_{X \geq b_{j^*-1}} \max(X - c\theta, 0).$$

Every option in the sum satisfies $X \geq b_{j^*}$, so each contributes at least

$$b_{j^*} - cb_{j^*-1} = B.$$

There are exactly i_{j^*} options. Therefore

$$\mathbb{E}[\max(X - c\theta, 0) \mid X \geq \theta] \geq \frac{i_{j^*}}{i_{j^*-1}} \cdot B.$$

We now lower bound the ratio of indices. Since $i_j = \lfloor n\beta^{-j} \rfloor$, we obtain

$$\begin{aligned} i_{j^*-1} &= \left\lfloor \beta \cdot n\beta^{-j^*} \right\rfloor \\ &\leq \beta \cdot \left\lfloor n\beta^{-j^*} \right\rfloor + \beta \\ &= \beta i_{j^*} + \beta. \end{aligned}$$

Hence,

$$\frac{i_{j^*}}{i_{j^*-1}} \geq \frac{i_{j^*}}{\beta i_{j^*} + \beta} = \frac{1}{\beta\left(1 + \frac{1}{i_{j^*}}\right)}.$$

Since $i_{j^*} \geq i_k = 1$, this gives

$$\frac{i_{j^*}}{i_{j^*-1}} \geq \frac{1}{2\beta}.$$

Substituting our bound for B, we conclude

$$\mathbb{E}[\max(X - c\theta, 0) \mid X \geq \theta] \geq \frac{a_{[1]}}{2\beta} \cdot \frac{c-1}{c^k - 1}.$$

It remains to estimate the size of the denominator. By the choice of k,

$$\beta = n^{1/k} \leq \exp\Big(\sqrt{\ln n \ln c}\Big).$$

Also,

$$c^k \leq c \exp\Big(\sqrt{\ln n \ln c}\Big).$$

Thus the algorithm obtains an $\Omega(\exp(-2\sqrt{\ln n \ln c}))$ fraction of $a_{[1]}$, and hence achieves the claimed competitive ratio. $\square$

4.3.2 Arbitrary Order, Full Information

Theorem 21. *In the secretary setting under arbitrary order with full information and an inflated cost factor $c > 1$, there exists an algorithm that achieves a competitive ratio of $O(\exp(2\sqrt{\ln n \ln c}))$.*

Proof. If $a_{[1]} = 0$, then all values are zero and the result is trivial. Assume $a_{[1]} > 0$.

We reuse the threshold ladder from Theorem 20. Let $k = \lceil \sqrt{\ln n / \ln c} \rceil$ and $\beta = n^{1/k}$. For $j = 1, \ldots, k$, define $i_j = \lfloor n\beta^{-j} \rfloor$, with the convention that $i_k = 1$, and set $b_j = a_{[i_j]}$. As we have done before, we set $b_0 = 0$. Define the maximum marginal gain

$$B = \max_{j \in [k]} (b_j - c b_{j-1}).$$

The same recurrence argument as in the random-order full-information case gives

$$B \geq a_{[1]} \frac{c-1}{c^k - 1}.$$

Let $j^* \in \operatorname{argmax}_{j \in [k]}(b_j - c b_{j-1})$ and set $\theta = b_{j^*-1}$.

In random order, accepting the first option above the threshold would automatically sample uniformly from the passing options. In arbitrary order, we instead sample this rank ourselves. Since the player has full information, it knows the exact number of options whose value is at least $\theta = b_{j^*-1}$. This number is i_{j^*-1} under our threshold construction.

The algorithm proceeds as follows:

1. Sample r uniformly from $\{1, \ldots, i_{j^*-1}\}$.
2. Set the threshold $\theta = b_{j^*-1}$.
3. Test each option sequentially with θ, rejecting the first $r-1$ options that pass. Accept the r-th option that passes.

Since exactly i_{j^*-1} options in the sequence have value at least b_{j^*-1}, all of them pass the test, so the r-th option that passes is guaranteed to exist regardless of the arrival order. Moreover, because r is drawn uniformly from $\{1, \ldots, i_{j^*-1}\}$ independently of the arrival order, each of the i_{j^*-1} passers is accepted with probability exactly $1/i_{j^*-1}$.

Let X denote the value of the accepted option. By the uniformity established above,

$$\mathbb{E}[\max(X - c\theta, 0)] = \frac{1}{i_{j^*-1}} \sum_{X \geq b_{j^*-1}} \max(X - c\theta, 0).$$

We lower bound this average by restricting the sum to the options with value at least b_{j^*}. There are at least i_{j^*} such options, and each contributes at least

$$b_{j^*} - cb_{j^*-1} = B.$$

Therefore,

$$\mathbb{E}[\max(X - c\theta, 0)] \geq \frac{i_{j^*}}{i_{j^*-1}} \cdot B.$$

As shown previously,

$$\frac{i_{j^*}}{i_{j^*-1}} \geq \frac{1}{2\beta}.$$

Substituting the lower bound on B, we get

$$\mathbb{E}[\max(X - c\theta, 0)] \geq \frac{a_{[1]}}{2\beta} \cdot \frac{c-1}{c^k - 1}.$$

Finally, by the choice of k and β,

$$\beta = n^{1/k} \leq \exp\left(\sqrt{\ln n \ln c}\right) \quad \text{and} \quad c^k \leq c \exp\left(\sqrt{\ln n \ln c}\right).$$

Hence

$$2\beta\frac{c^k-1}{c-1} = O\Big(\exp\Big(2\sqrt{\ln n \ln c}\Big)\Big).$$

□

4.3.3 Random Order, Optimum Information

The threshold ladder used in the full-information setting is no longer available: we cannot define thresholds such as $a_{[i_j]}$ because the player does not know the sorted values. However, knowing $a_{[1]}$ still gives us the correct global scale. We therefore build a deterministic ladder from 0 to $a_{[1]}$ whose consecutive marginal gains are all equal. Random arrival order will again ensure that the first option passing a threshold is uniformly distributed among all options that pass that threshold.

Theorem 22. *In the secretary setting under random order with optimum information and an inflated cost factor $c > 1$, there exists an algorithm that achieves an $O(\exp(2\sqrt{\ln n \ln c}))$ competitive ratio.*

Proof. If $a_{[1]} = 0$, then all values are zero and the result is trivial. Assume $a_{[1]} > 0$.

Let $k = \lceil \sqrt{\ln n / \ln c} \rceil$ and $\beta = n^{1/k}$. We construct a sequence of $k+1$ thresholds,

$$b_j = a_{[1]} \cdot \frac{c^j - 1}{c^k - 1} \qquad j \in 0, 1, \ldots, k$$

such that $b_0 = 0$ and $b_k = a_{[1]}$. For any $j \in [k]$, the marginal gain of passing b_j when testing at b_{j-1} is

$$b_j - cb_{j-1} = a_{[1]} \cdot \frac{c^j - 1 - c(c^{j-1} - 1)}{c^k - 1} = a_{[1]} \cdot \frac{c-1}{c^k - 1}.$$

Thus, unlike in the full-information case, we do not need to search for the best level of the ladder; all levels give the same certified utility if the accepted option reaches the next threshold.

For each $j = 0, 1, \ldots, k$, define

$$n_{\geq j} = \left|\{i \in [n] : a_i \geq b_j\}\right|.$$

This denotes the number of options whose value is at least b_j. Since $b_0 = 0$, we have $n_{\geq 0} = n$. Since $b_k = a_{[1]}$, we have $n_{\geq k} \geq 1$.

The algorithm is:

1. Choose j uniformly at random from $\{1, \ldots, k\}$.
2. Set the threshold to $\theta = b_{j-1}$.
3. Accept the first option that passes the test.

Fix a choice of j. Among the options that pass the threshold b_{j-1}, call an option "good" if its value is at least b_j. The fraction of passing options that are good is

$$p_j = \frac{n_{\geq j}}{n_{\geq j-1}}.$$

Thus, conditional on the choice of j, the expected utility is at least

$$p_j \cdot a_{[1]} \cdot \frac{c-1}{c^k - 1}.$$

Now average over the uniformly random choice of j. The expected utility is at least

$$\frac{1}{k} \sum_{j=1}^{k} p_j \cdot a_{[1]} \cdot \frac{c-1}{c^k - 1}.$$

It remains to lower bound the average of the p_j's. By AM-GM,

$$\begin{aligned}
\frac{1}{k} \sum_{j=1}^{k} p_j &\geq \left(\prod_{j=1}^{k} p_j \right)^{1/k} \\
&= \left(\frac{n_{\geq 1}}{n_{\geq 0}} \cdot \frac{n_{\geq 2}}{n_{\geq 1}} \cdots \frac{n_{\geq k}}{n_{\geq k-1}} \right)^{1/k} \\
&= \left(\frac{n_{\geq k}}{n_{\geq 0}} \right)^{1/k} \\
&\geq \left(\frac{1}{n} \right)^{1/k} \\
&= \frac{1}{\beta}.
\end{aligned}$$

Therefore, the algorithm's expected utility is at least

$$\frac{a_{[1]}}{\beta} \cdot \frac{c-1}{c^k - 1}.$$

Equivalently, the competitive ratio is at most

$$\beta \cdot \frac{c^k - 1}{c - 1}.$$

Finally, by the choice of k and β,

$$\beta = n^{1/k} \leq \exp\Big(\sqrt{\ln n \ln c}\Big) \quad \text{and} \quad c^k \leq c \exp\Big(\sqrt{\ln n \ln c}\Big).$$

Hence

$$\beta \frac{c^k - 1}{c - 1} = O\Big(\exp\Big(2\sqrt{\ln n \ln c}\Big)\Big).$$

□

4.3.4 Arbitrary Order, Optimum Information

The player still knows $a_{[1]}$, so the deterministic threshold ladder from the previous subsection is still available. However, the key random-order fact no longer holds: the first option passing a threshold need not be representative of all passing options. An adversarial order may place all "bad" passing options before the "good" passing options.

To overcome this, we use a sampling subroutine. For a fixed threshold b_{j-1}, consider the subsequence of options that pass this threshold. We would like to choose a roughly random element from this subsequence, but its length is unknown in advance because the player does not know the multiset. The generalized bit-sampling game abstracts this difficulty. The bit 1 represents a passing option whose value is at least the next threshold b_j, while the bit 0 represents a passing option whose value lies between b_{j-1} and b_j.

Definition 23 (Generalized Bit Sampling Game)**.** An adversary constructs a binary sequence of unknown length $m \leq n$. The player is guaranteed that the fraction of 1s in the sequence is at least ρ. The player observes the sequence one bit at a time and can irrevocably commit to the next unseen bit. The player wins if the committed bit is 1; the player loses if the committed bit is 0 or if the sequence ends before the player commits.

Lemma 24. *There exists a strategy that wins the Generalized Bit Sampling game with probability at least $\rho/(2\lceil \log n \rceil + 1)$.*

Proof. The obstacle is that the player does not know the sequence length m. The strategy guesses the scale of m and then samples uniformly from that guessed scale.

The strategy is:

1. Guess a parameter p uniformly at random from $\{0, 1, \ldots, \lceil \log n \rceil\}$.
2. Set $M = 2^p$.
3. Select an index r uniformly at random from $\{0, 1, \ldots, M\}$.
4. Commit to the r-th bit of the sequence; if the sequence ends before then, the player loses.

There exists a unique $p^* \in \{0, 1, \ldots, \lceil \log n \rceil\}$ such that $2^{p^*-1} < m \leq 2^{p^*}$. The player chooses $p = p^*$ with probability $1/(\lceil \log n \rceil + 1)$.

Condition on this event. Then we know $M = 2^{p^*} \geq m$, so every actual bit of the sequence lies among the first M possible positions. Since at least a ρ fraction of the m bits are 1s, the sequence contains at least ρm ones. Therefore, the probability that the uniformly chosen index $r \in \{1, \ldots, M\}$ points to a 1 is at least

$$\frac{\rho m}{2^{p^*}}.$$

Using $2^{p^*-1} < m$, we get

$$\frac{\rho m}{2^{p^*}} > \frac{\rho}{2}.$$

Multiplying by the probability of choosing the correct scale p^*, the unconditional winning probability is at least

$$\frac{\rho}{2(\lceil \log_2 n \rceil + 1)}.$$

□

We now use this sampling subroutine to obtain a guarantee under arbitrary order.

Theorem 25. *In the secretary setting under arbitrary order with optimum information and an inflated cost factor $c > 1$, there exists an algorithm that achieves an $O(\log n \cdot \exp(2\sqrt{\ln n \ln c}))$ competitive ratio.*

Proof. Assume $a_{[1]} > 0$, since the case $a_{[1]} = 0$ is trivial.

Let $k = \lceil \sqrt{\ln n / \ln c} \rceil$ and $\beta = n^{1/k}$. As we have done before, construct the deterministic threshold ladder

$$b_j = a_{[1]} \cdot \frac{c^j - 1}{c^k - 1} \qquad j \in 0, 1, \ldots, k$$

such that $b_0 = 0$, $b_k = a_{[1]}$, and for every $j \in [k]$,

$$b_j - cb_{j-1} = a_{[1]} \cdot \frac{c-1}{c^k - 1}.$$

The algorithm is as follows

1. Select j uniformly at random from $\{1, \ldots, k\}$.
2. Set $\theta = b_{j-1}$.
3. Play the Generalized Bit Sampling game, where p is uniformly random from $\{0, \ldots, \lceil \log n \rceil\}$ and r from $\{1, \ldots, 2^p\}$.
4. Test each option with θ. Reject the first $r-1$ options that pass, and accept the r-th option that passes.

Fix a choice of j. Let $n_{\geq j} = |\{i \in [n] : a_i \geq b_j\}|$ and define

$$p_j = \frac{n_{\geq j}}{n_{\geq j-1}}.$$

Among the options that pass the threshold b_{j-1}, a p_j fraction also have value at least b_j. We call these options good. In the corresponding bit-sampling instance, good options are encoded as 1s and the remaining passing options are encoded as 0s. Thus the fraction of 1s is exactly $\rho = p_j$. By Lemma (24) with $\rho = p_j$, the probability that the committed bit is 1 is at least

$$\frac{p_j}{2(\lceil \log n \rceil + 1)}.$$

If the accepted option is good, then its value is at least b_j, while the test threshold was b_{j-1}. Therefore its remaining utility is at least

$$b_j - cb_{j-1} = a_{[1]} \cdot \frac{c-1}{c^k - 1}.$$

Conditioned on our specific choice of j, the expected remaining utility is at least:

$$\frac{p_j}{2(\lceil \log n \rceil + 1)} \cdot a_{[1]} \cdot \frac{c-1}{c^k-1}.$$

Now average over the uniformly random choice of j. The expected utility is at least

$$\frac{1}{k}\sum_{j=1}^{k} \frac{p_j}{2(\lceil \log_2 n \rceil + 1)} \cdot a_{[1]} \cdot \frac{c-1}{c^k-1}.$$

Just as we did in the random order setting under optimum information, we can apply the AM-GM inequality to lower bound the average of p_j, to obtain

$$\frac{1}{k}\sum_{j=1}^{k} p_j \geq \frac{1}{\beta}.$$

Therefore the expected utility is at least

$$\frac{a_{[1]}}{2\beta(\lceil \log n \rceil + 1)} \cdot \frac{c-1}{c^k-1}.$$

Finally, by the choice of k and β,

$$\beta = n^{1/k} \leq \exp\Big(\sqrt{\ln n \ln c}\Big) \quad \text{and} \quad c^k \leq c \exp\Big(\sqrt{\ln n \ln c}\Big).$$

Hence

$$\beta \cdot (\lceil \log n \rceil + 1) \cdot \frac{c^k - 1}{c-1} = O\Big(\log n \cdot \exp\Big(2\sqrt{\ln n \ln c}\Big)\Big).$$

□

4.3.5 Random Order, No Information

We finally consider the random-order secretary setting with no information. This is the weakest information model in which we prove a nontrivial inflated-price guarantee. Since the player is not given $a_{[1]}$, it cannot build the deterministic threshold ladder from the previous two subsections. The main task is therefore to learn a scale from the sequence itself.

The algorithm uses the standard secretary-style idea of splitting the sequence into two halves. The first half is used only for observation: we sacrifice those options in order to estimate the scale of the largest values. The second half is used for selection. The analysis conditions on the constant-probability event that the second-largest value appears in the first half while the largest value appears in the second half. On this event, the first-half maximum is exactly $a_{[2]}$, which provides a useful estimate of the scale of $a_{[1]}$.

There are then two cases. If $a_{[1]}$ is much larger than $a_{[2]}$, a single large threshold isolates $a_{[1]}$. If $a_{[1]}$ is within a controlled factor of $a_{[2]}$, we build a threshold ladder using the first-half estimate and reuse the random-order optimum-information argument.

Theorem 26. *In the secretary setting under random order with no information and an inflated cost factor $c > 1$, there exists an algorithm that achieves a competitive ratio of $O(\exp(2\sqrt{\ln n \ln c}))$.*

Proof. If $a_{[1]} = 0$, then all values are zero and the result is trivial. Assume $a_{[1]} > 0$.

We split our approach into two phases. In the first phase, the player observes the first $m = \lfloor n/2 \rfloor$ options (by testing them with $\theta = +\infty$). Let $M = \max_{i \in [m]} X_i$ be the maximum value observed during the first phase. For the second phase, over the remaining $n - m$ options, we flip a fair coin to determine which of the following two strategies to employ:

1. Set the threshold to $\theta = 2Mc$. Accept the first option X_i that passes the test.
2. Let $H = 4Mc^2$, $k = \lceil \sqrt{\ln n / \ln c} \rceil$, and $\beta = n^{1/k}$. We construct a sequence of $k + 1$ thresholds:
$$b_j = H \cdot \frac{c^j - 1}{c^k - 1}, \qquad j = 0, 1, \ldots, k.$$
Choose j uniformly at random from $\{1, \ldots, k\}$ and set the threshold to $\theta = b_{j-1}$. Accept the first option that passes the test.

We condition our analysis on the event that $a_{[2]}$ is found in the first half and $a_{[1]}$ is found in the second half. Call this event $\mathcal{E}$. Since the arrival order is uniformly random, the probability of this event occurring is

$$\frac{m}{n} \cdot \frac{n-m}{n-1} > \frac{1}{4}.$$

Condition on $\mathcal{E}$ for the rest of the proof. Then $M = a_{[2]}$. We examine two exhaustive cases based on the relation between $a_{[1]}$ and M.

Case 1 ($a_{[1]} > 4Mc^2$): In this case, we rely on strategy 1. We test with $\theta = 2Mc$. Since $c > 1$, we have $\theta = 2cM > M = a_{[2]}$. Thus no option other than $a_{[1]}$ can pass this threshold. Also, $a_{[1]} > 4c^2M > 2cM = \theta$, so $a_{[1]}$ does pass the threshold. Therefore, under the high-threshold strategy, the algorithm accepts $a_{[1]}$.

The remaining utility is

$$a_{[1]} - c\theta = a_{[1]} - 2c^2M.$$

Using $a_{[1]} > 4c^2M$, we get

$$a_{[1]} - 2c^2M > a_{[1]} - 2c^2\left(\frac{a_{[1]}}{4c^2}\right) = \frac{a_{[1]}}{2}$$

Accounting for the probability of $\mathcal{E}$ and the probability 1/2 of choosing the high-threshold strategy, the expected utility in this case is at least

$$\frac{1}{4} \cdot \frac{1}{2} \cdot \frac{a_{[1]}}{2} = \frac{a_{[1]}}{16},$$

providing an $O(1)$ competitive ratio.

Case 2 ($a_{[1]} \leq 4Mc^2$): In this case, the first-half estimate is within a controlled factor of the true maximum. Since $H = 4Mc^2$, we know $a_{[1]} \leq H$. Moreover, since $M = a_{[2]}$, we know that $a_{[1]} \geq M$, which informs us that

$$a_{[1]} \geq M = \frac{H}{4c^2}.$$

Thus the ladder endpoint H may overestimate $a_{[1]}$, but only by a factor depending on c.

We now analyze the ladder strategy. Recall that

$$b_j = H \cdot \frac{c^j - 1}{c^k - 1}.$$

Every adjacent pair has the same marginal gain:

$$b_j - cb_{j-1} = H \cdot \frac{c-1}{c^k - 1}.$$

Let κ be the largest index such that

$$b_\kappa \leq a_{[1]}.$$

The algorithm does not need to know κ; it is used only in the analysis. We first show that κ is close to k. Define

$$\delta = \left\lceil \log_c\left(\frac{4c^3}{c-1}\right)\right\rceil.$$

This quantity depends only on c. We claim that

$$\kappa \geq k - \delta.$$

Necessarily,

$$b_{k-\delta} = H\frac{c^{k-\delta}-1}{c^k-1} < H\frac{c^{k-\delta}}{c^k-1} = \frac{H}{c^\delta(1-c^{-k})}.$$

Since $k \geq 1$,

$$1 - c^{-k} \geq 1 - \frac{1}{c} = \frac{c-1}{c}.$$

Therefore,

$$b_{k-\delta} \leq \frac{H}{c^{\delta-1}(c-1)}.$$

By the definition of δ,

$$c^\delta \geq \frac{4c^3}{c-1},$$

so

$$c^{\delta-1}(c-1) \geq 4c^2.$$

Hence

$$b_{k-\delta} \leq \frac{H}{4c^2} = M \leq a_{[1]}.$$

Thus $\kappa \geq k - \delta$.

If $k < 2\delta$, then n is bounded above by a constant depending only on c, and the trivial strategy of choosing a uniformly random position gives a constant competitive ratio depending only on c. We may therefore assume $k \geq 2\delta$. Then

$$\kappa \geq k - \delta \geq \frac{k}{2}.$$

For each $j \leq \kappa$, let $n_{\geq j}$ denote the number of options in the second half whose

value is at least b_j. Also define
$$p_j = \frac{n_{\geq j}}{n_{\geq j-1}}.$$
Conditional on choosing the ladder level j, the threshold is b_{j-1}. Among the options in the second half that pass this threshold, a p_j fraction also have value at least b_j. Because the second half is in uniformly random order, the first option passing b_{j-1} is uniformly distributed among all such passing options. Hence, with probability p_j, the accepted option has value at least b_j. When this happens, the remaining utility is at least
$$b_j - cb_{j-1} = H \cdot \frac{c-1}{c^k - 1}.$$
Therefore, conditional on using the ladder strategy, the expected utility is at least
$$\frac{1}{k}\sum_{j=1}^{\kappa} p_j \cdot H \cdot \frac{c-1}{c^k-1}.$$

We lower bound the average contribution of the p_j's. By AM-GM,
$$\frac{1}{\kappa}\sum_{j=1}^{\kappa} p_j \geq \left(\prod_{j=1}^{\kappa} p_j\right)^{1/\kappa} = \left(\frac{n_{\geq \kappa}}{n_{\geq 0}}\right)^{1/\kappa}.$$
Since $b_\kappa \leq a_{[1]}$ and $a_{[1]}$ lies in the second half under $\mathcal{E}$, we have $n_{\geq \kappa} \geq 1$. Also $n_{\geq 0} \leq n$. Therefore,
$$\frac{1}{\kappa}\sum_{j=1}^{\kappa} p_j \geq n^{-1/\kappa}.$$
It follows that
$$\frac{1}{k}\sum_{j=1}^{\kappa} p_j \geq \frac{\kappa}{k} n^{-1/\kappa}.$$
Since $\kappa \geq k/2$, this gives
$$\frac{1}{k}\sum_{j=1}^{\kappa} p_j \geq \frac{1}{2} n^{-1/\kappa}.$$
Thus, conditional on the ladder strategy, the expected utility is at least
$$\frac{1}{2} n^{-1/\kappa} \cdot H \cdot \frac{c-1}{c^k-1}.$$

Since $H \geq a_{[1]}$, this is at least

$$\frac{1}{2}n^{-1/\kappa} \cdot a_{[1]} \cdot \frac{c-1}{c^k-1}.$$

It remains to compare $n^{-1/\kappa}$ with $n^{-1/k} = 1/\beta$. Since $\kappa \geq k - \delta$,

$$n^{-1/\kappa} \geq \exp\left(-\frac{\ln n}{k-\delta}\right).$$

Rewrite this as

$$n^{-1/\kappa} \geq n^{-1/k} \exp\left(-\frac{\delta \ln n}{k(k-\delta)}\right).$$

By the definition of k,

$$\ln n \leq k^2 \ln c,$$

and since $k \geq 2\delta$, we have

$$\frac{k}{k-\delta} \leq 2.$$

Therefore,

$$\frac{\delta \ln n}{k(k-\delta)} \leq 2\delta \ln c.$$

Hence

$$n^{-1/\kappa} \geq c^{-2\delta} n^{-1/k} = \frac{c^{-2\delta}}{\beta}.$$

Substituting this into the utility bound, the ladder strategy obtains expected utility at least $\Omega(\frac{a_{[1]}}{\beta} \cdot \frac{c-1}{c^k-1})$.

Finally, accounting for the probability of $\mathcal{E}$ and the probability 1/2 of choosing the ladder strategy only changes the bound by an additional constant. Thus, in Case 2 the overall expected utility is at least

$$\Omega\left(\frac{a_{[1]}}{\beta} \cdot \frac{c-1}{c^k-1}\right).$$

As before,

$$\beta = n^{1/k} \leq \exp\left(\sqrt{\ln n \ln c}\right) \quad \text{and} \quad c^k \leq c \exp\left(\sqrt{\ln n \ln c}\right).$$

Therefore the competitive ratio in Case 2 is

$$O(\exp(2\sqrt{\ln n \ln c})).$$

Combining the two cases proves the theorem. $\square$

Chapter 5

Lower Bounds

In this chapter, we provide a lower bound for the secretary setting as well as in the prophet IID setting, given an inflated price cost factor.

5.1 Secretary Setting

We show that by constructing a hard distribution of instances even for the easiest setting: random order arrival given full information. Since this is the most accommodating setting for the player, the lower bound naturally extends to the arbitrary order and optimum as well as no information settings as well.

Theorem 27. *In the secretary setting under random order with full information and an inflated cost factor $c > 1$, no algorithm can achieve a competitive ratio better than $\Omega(\frac{1}{\sqrt{\ln n}} \cdot \exp(\sqrt{\ln n \ln c}))$.*

Proof. By Yao's Principle, it suffices to construct a fixed multiset of values and show that any deterministic strategy performs poorly when the values arrive in a uniformly random order.

We design the adversary's multiset by constructing a ladder of k distinct value levels, spaced geometrically by the cost factor c. To prevent the player from easily extracting utility, the adversary buries the valuable options at each level among a large number of "decoy" options.

Let M denote the multiplicative growth factor of the option frequencies between consecutive levels. That is, an option of some designated value v will appear with some factor of M times in the multiset; we want the frequency of the number options to decay as the options have more value. To fit k such levels within a sequence of length n, we require $M^k \leq n$, which suggests setting $M \approx n^{1/k}$.

As we will show, the player's competitive ratio on this instance will be constrained by the tradeoff between the maximum value c^k and the decoy ratio M/k. To maximize the resulting lower bound, the adversary balances these forces by setting $c^k \approx M$. Taking logarithms yields $k \ln c \approx \ln n/k$, which naturally motivates choosing $k \approx \sqrt{\ln n/\ln c}$.

Formalizing this intuition, we define the parameters k and M as follows:

$$k = \left\lfloor \sqrt{\frac{\ln n}{\ln c}} \right\rfloor \quad \text{and} \quad M = \lfloor n^{1/k} \rfloor.$$

For sufficiently large n, we have $M \geq 2c$. Specifically, since $n^{1/k} \geq \exp(\sqrt{\ln n \ln c})$, we have

$$M = \lfloor n^{1/k} \rfloor \geq 2c,$$

for all $n \geq \exp(\ln^2(2c+1)/\ln c)$. Since $c > 1$ is fixed, this holds for all sufficiently large n. Notice that by construction, $M^k \leq n$.

We now need to construct our adversarial multiset. It will contain $k+1$ distinct option values. Let $v_j = c^j$ for $j \in \{0, \ldots, k\}$. Each of these values will appear in the multiset with a frequency of

$$N_j = M^{k-j} \text{ for } j \in [k].$$

We absorb the remaining elements into the lowest value by setting $N_0 = n - \sum_{j=1}^{k} M^{k-j}$. Since,

$$\sum_{j=1}^{k} M^{k-j} = \frac{M^k - 1}{M - 1} \leq M^k \leq n,$$

the count N_0 is strictly positive.

At any step, a deterministic algorithm necessarily has to choose a threshold $\theta \geq 0$. We partition the possible choices of θ into $k+2$ levels:

1. **Level 0** $(0 \leq \theta \leq 1/c)$**:** The threshold is low enough that the minimum option $v_0 = 1$ passes the test (since $\theta \leq 1/c < 1$).

2. **Level** $j \in [k]$ **($c^{j-2} < \theta \leq c^{j-1}$):** The options that pass the test are exactly those with value $\geq c^{j-1}$, which is the subset $\{v_{j-1}, v_j, \ldots, v_k\}$.

3. **Beyond Level** k **($\theta > c^{k-1}$):** The threshold is so high that $c\theta > c^k$. Even the maximum value v_k yields a remaining utility of 0.

We now bound the expected utility the algorithm can obtain from any of these levels.

If the algorithm accepts an option while setting a threshold at Level 0, the threshold ($\theta \leq 1/c$) ensures that all options pass the test. Any acceptance at this level is thus equivalent to drawing a uniformly random option from the remaining pool. By the equivalence to random permutation, the expected value of this draw is necessarily the expected value of a random option from the initial multiset:

$$\begin{aligned}\mathbb{E}[X] &= \frac{1}{n}\left(\left(N_0 \cdot c^0\right) + \sum_{j=1}^{k} N_j c^j\right) \\ &= \frac{N_0}{n} + \sum_{j=1}^{k} \frac{M^{k-j}}{n} c^j.\end{aligned}$$

We can claim $N_0/n \leq 1$ and since $M^k \leq n$, we can also conclude $1/n \leq 1/M^k$. That is, for every $j \in [k]$:

$$\frac{M^{k-j}}{n} c^j \leq \frac{M^{k-j}}{M^k} c^j = \left(\frac{c}{M}\right)^j.$$

Using this we can bound the expected value of a random option in the multiset,

$$\begin{aligned}\mathbb{E}[X] &= \frac{N_0}{n} + \sum_{j=1}^{k} \frac{M^{k-j}}{n} c^j \\ &\leq 1 + \sum_{j=1}^{k} \left(\frac{c}{M}\right)^j.\end{aligned}$$

Predicated on $M \geq 2c$, the term of the geometric series $c/M \leq 1/2$

$$\begin{aligned}\sum_{j=1}^{k} \left(\frac{c}{M}\right)^j &\leq \sum_{j=1}^{\infty} \left(\frac{1}{2}\right)^j \\ &= \frac{1/2}{1 - 1/2} \\ &= 1.\end{aligned}$$

Then from the previous bound,

$$\mathbb{E}[X] \leq 1 + \sum_{j=1}^{k} \left(\frac{c}{M}\right)^{j} \leq 1 + 1 = 2.$$

Thus, the expected utility obtained from Level 0 is at most 2.

Suppose the algorithm accepts an option while playing at Level $j \geq 1$. The set of options that can pass the test is $S_j = \{v_{j-1}, v_j, \ldots, v_k\}$. The lowest value in this passing set is $v_{j-1} = c^{j-1}$. Since the threshold at Level j is at least $c\theta > c^{j-1}$, accepting the lowest value option v_{j-1} yields a remaining utility of 0.

Therefore, the algorithm obtains strictly positive utility at Level j only if it accepts an option from the "good" set $G_j = \{v_j, \ldots, v_k\}$. The options in $B_j = \{v_{j-1}\}$ act as "bad" passing options (decoys), yielding zero utility. The initial number of good options is

$$g_j = \sum_{l=j}^{k} M^{k-l} \leq \frac{M^{k-j+1}}{M-1}.$$

Let E_j denote the event that the algorithm accepts a good option while playing at Level j. A potential complication is that the algorithm might adaptively switch between levels based on observed outcomes, that is altering the number of "good" and "bad" options with respect to the Level j. However, as we establish in the following lemma, this adaptivity cannot increase the probability of accepting a good option beyond its initial fraction in the sequence.

Lemma 28 (Adaptivity Bound)**.** *For every $j \in [k]$ and every deterministic algorithm,*

$$\Pr(E_j) \leq \frac{2}{M}$$

Proof. Let A_t denote the number of "good" options (those of value $\geq v_j$) that have not yet arrived after the first t options, and let $T_t = n-t$ be the total number of options not yet arrived. cause the arrival order is a uniformly random permutation, the fraction $m_t = A_t/T_t$ is a martingale: removing one option chosen uniformly at random from the remaining pool leaves the expected remaining good-fraction unchanged, that is

$$\mathbb{E}\left[\frac{A_t}{T_t} \mid \mathcal{F}_{t-1}\right] = \frac{A_{t-1}}{T_{t-1}},$$

and this holds even with respect to the finest filtration $\mathcal{F}_t$ in which the exact value of every option that has already arrived is known.

Fix a time t and condition on the past $\mathcal{F}_{t-1}$. Suppose the algorithm has not yet accepted and, on the basis of the past alone (its choice of threshold is made before the t-th option is revealed), decides to test at Level j. The t-th option is drawn uniformly at random over the remaining pool, so

$$\Pr(t\text{-th option is good} \mid \mathcal{F}_{t-1}) = m_{t-1}.$$

A Level-j test only reveals the singleton information of $X \geq \theta$ and cannot distinguish a good option from the decoy v_{j-1} (both pass), so the algorithm's accept decision on a pass is the same function for both cases. Hence, the probability of accepting a good option at Level j at a time t given the filtration $\mathcal{F}_{t-1}$ is equal to

$$\Pr(\text{accept-on-pass} \mid \mathcal{F}_{t-1}) \cdot \frac{A_{t-1}}{T_{t-1}} \leq \frac{A_{t-1}}{T_{t-1}}$$

Let τ be the acceptance time. Since at most one option is ever accepted, summing the previous expression over t and using $m_t \geq 0$ together with required constraint of $\tau \wedge n$ (where n is the total time i.e. the number of options in the sequence), we obtain:

$$\Pr(E_j) = \mathbb{E}[\mathbb{1}\{\text{accept at Level } j\} \cdot m_{\tau-1}] \leq \mathbb{E}[m_{\tau\wedge n}] = m_0 = \frac{g_j}{n}.$$

Finally, because

$$g_j = \sum_{l=j}^{k} M^{k-l} = \frac{M^{k-j+1}-1}{M-1} \leq \frac{M^{k-j+1}}{M-1}$$

and $n \geq M^k$, we have

$$\frac{g_j}{n} \leq \frac{M^{1-j}}{M-1} \leq \frac{1}{M-1} \leq \frac{2}{M}$$

for $j \geq 1$. □

By Lemma 28, $\Pr(E_j) \leq \frac{2}{M}$, that is the probability of obtaining positive utility when testing at Level j.

The maximum utility any option can provide is $v_k = c^k$. By applying the union bound over all possible levels the algorithm might use to obtain positive utility, the overall expected utility of the algorithm is bounded by:

$$2 + \sum_{j=1}^{k} c^k \Pr(E_j) \leq 2 + k\frac{2c^k}{M}.$$

Thus, the competitive ratio is

$$\frac{c^k}{2 + \frac{2kc^k}{M}} = \Omega\left(\frac{M}{k}\right).$$

Substituting our parameter choices $M = \lfloor n^{1/k} \rfloor \geq \frac{1}{2}\exp(\sqrt{\ln n \ln c})$ and $k \leq \sqrt{\ln n / \ln c}$, we obtain:

$$\Omega\left(\frac{\sqrt{\ln c}}{\sqrt{\ln n}} \cdot \exp(\sqrt{\ln n \ln c})\right)$$

$\square$

There remains a factor-of-2 gap in the exponent constant between this lower bound and the $O(\exp(2\sqrt{\ln n \ln c}))$ upper bounds, leaving the exact optimal constant an open question.

5.2 Prophet IID Setting

Similarly, we will construct a hard IID instance, which is the easiest instance of the problem discussed throughout this paper. Since this is the most accommodating setting for the player, the lower bound naturally extends to the general prophet setting as well.

Theorem 29. *In the prophet IID setting with an inflated cost factor $c > 1$, no algorithm can achieve a competitive ratio better than $\Omega(\exp(\sqrt{\ln n \ln c}))$.*

Proof. Same intuition about geometric ladder determined by the cost factor c and the number of levels should be k. As before,

$$k = \left\lfloor \sqrt{\ln n / \ln c} \right\rfloor \text{ and } M = n^{1/k}.$$

For sufficiently large n, we have $M \geq 2c$. Specifically, since $n^{1/k} \geq \exp(\sqrt{\ln n \ln c})$, this holds for all sufficiently large n given a fixed $c > 1$.

Now we need to construct our adversarial distribution $\mathcal{D}$. It should contain $k+1$ unique option values. Let $v_0 = 0$ and $v_j = c^j$ for $j \in [k]$. Define the probability mass function as

$$\Pr_{X \sim \mathcal{D}}\big(X \geq c^j\big) = M^{-j},$$

for every $j \in [k]$. As a result,

$$\Pr\big(X = c^k\big) = \Pr\big(X \geq c^k\big) = M^{-k} = \frac{1}{n},$$

and similarly for any $1 \leq j < k$,

$$\Pr\big(X = c^j\big) = \Pr\big(X \geq c^j\big) - \Pr\big(X \geq c^{j+1}\big) = M^{-j} - M^{-(j+1)} \leq M^{-j}.$$

The remaining probability mass of $1 - M^{-1}$ is assigned to $v_0 = 0$.

The probability that the sequence of n independent draws contains at least one option of maximum value c^k is

$$1 - \left(1 - \frac{1}{n}\right)^n \geq 1 - \frac{1}{e}.$$

Therefore, the expected optimum is at least $c^k(1 - 1/e)$.

We now need to bound the expected utility any deterministic algorithm can obtain. Define

$$U(\theta) = \mathbb{E}[\max(X - c\theta, 0) \mid X \geq \theta].$$

to be the expected remaining utility conditional on an option passing the test at θ. We partition the possible choices of θ into 3 cases

1. $\theta > c^{k-1}$
2. $1 < \theta \leq c^{k-1}$
3. $0 \leq \theta \leq 1$

We now bound the expected utility the algorithm can obtain from any of these levels.

Case 1

Any option that passes the test must have a value $X \geq \theta > c^{k-1}$. Since the maximum value is c^k, the accepted option yields a remaining utility of at most

$$X - c\theta < c^k - c(c^{k-1}) = 0.$$

Thus, $U(\theta) = 0$.

Case 2

Let $j \in \{2, \ldots, k\}$ such that $c^{(j-2)} < \theta \leq c^{(j-1)}$. If the option's value is exactly the lower bound c^{j-1}, its remaining utility is strictly bounded by

$$c^{j-1} - c\theta < c^{j-1} - c(c^{j-2}) = 0.$$

Thus, to obtain a positive expected utility, the accepted option must have a value strictly greater than c^{j-1}, meaning $X \geq c^j$. The conditional probability of this occurring is

$$\Pr\big(X \geq c^j \mid X \geq c^{j-1}\big) = \frac{M^{-j}}{M^{-(j-1)}} = \frac{1}{M}.$$

Even then, the maximum possible utility is bounded by the maximum value, c^k. Therefore, the conditional expected utility upon accepting an option that passes the test is at most

$$\begin{aligned} U(\theta) &\leq \Pr\big(X \geq c^j \mid X \geq c^{j-1}\big) \cdot c^k + \Pr\big(X = c^{j-1} \mid X \geq c^{j-1}\big) \cdot 0 \\ &= \frac{c^k}{M}. \end{aligned}$$

Case 3

The threshold is low enough that any option yielding strictly positive utility must have $X > c\theta \geq 0$, which implies $X \geq c$. The expected utility from accepting is at most the expected value of an option conditioned on $X \geq c$:

$$U(\theta) \leq \mathbb{E}[X \mid X \geq c] = \frac{1}{\Pr(X \geq c)} \cdot \sum_{j=1}^{k} c^j \cdot \Pr\big(X = c^j\big).$$

Using the fact that $\Pr(X = c^j) \leq \Pr(X \geq c^j) = M^{-j}$ and $\Pr(X \geq c) = M^{-1}$, we obtain

$$U(\theta) \leq \frac{1}{M^{-1}} \cdot \sum_{j=1}^{k} \left(\frac{c}{M}\right)^j = c \sum_{j=1}^{k} \left(\frac{c}{M}\right)^{j-1}$$

Predicated on $M \geq 2c$, the geometric series is bounded by 2. Thus, $U(\theta) \leq 2c$.

To unify the bound across all choices of θ, our goal is to relate $2c$ to the ratio c^k/M.

By the definition of k, we have $k \leq \sqrt{\ln n/\ln c}$, which implies $k^2 \leq \ln n/\ln c$. Furthermore, since $k \geq 1$, we also have $\sqrt{\ln n/\ln c} < k+1$, which implies

$$\frac{\ln n}{\ln c} < (k+1)^2 \leq k^2 + 3k.$$

Dividing by k yields $\frac{\ln n}{k \cdot \ln c} < k+3$, which is exactly

$$\frac{\ln M}{\ln c} < k+3,$$

and is equivalent to

$$M < c^{k+3}.$$

Rearranging this gives a lower bound of $\frac{c^k}{M} > c^{-3}$. Therefore,

$$U(\theta) \leq 2c = 2c^4 \cdot c^{-3} < 2c^4 \cdot \frac{c^k}{M}.$$

So we need to consider $\Delta = \max(1, 2c^4)$. That is, for every threshold $\theta \geq 0$, we have shown $U(\theta) \leq \Delta c^k/M$.

Let V_m be the optimal expected utility from the remaining m steps. Since the options are drawn identically and independently from $\mathcal{D}$, the state of the sequence depends solely on the number of remaining steps, m, and is independent of any prior observations.

When $m = 0$, there are no more options, yielding our base case of $V_0 = 0$.

With $m > 0$ options remaining the in sequence, the algorithm is shown a independent value X drawn from the adversarial distribution $\mathcal{D}$ and must set a threshold $\theta \geq 0$. If the test fails (i.e., $X < \theta$), the option's remaining utility is zero and the algorithm is forced to reject it. So, the algorithm is left with $m-1$ remaining options and an expected future utility of V_{m-1}. On the other hand, if the passes, the algorithm must make the irrevocable choice of accepting or rejecting the option. Accepting it yields an expected remaining utility of $U(\theta)$. Rejecting it results in the same case of failing the test: an expected future utility of V_{m-1}. This can be expressed as the following recurrence relation,

$$V_m = \max_{\theta \geq 0} \Big(\Pr(X \geq \theta) \cdot \max(U(\theta), V_{m-1}) + \Pr(X < \theta) \cdot V_{m-1} \Big).$$

We proceed by induction on m to prove that $V_m \leq \Delta \frac{c^k}{M}$ for all $m \geq 0$.

For the base case $m = 0$, we trivially have $V_0 = 0 \leq \Delta \frac{c^k}{M}$.

For the inductive step, assume that the bound holds for $m-1$, that is, $V_{m-1} \leq \Delta \frac{c^k}{M}$. From our earlier analysis, we know that the conditional expected utility of accepting any passing option is uniformly bounded by $U(\theta) \leq \Delta \frac{c^k}{M}$ for all choices of $\theta \geq 0$. Consequently, the optimal decision upon a passing test is bounded by:

$$\begin{aligned} \max(U(\theta), V_{m-1}) &\leq \max\left(\Delta \frac{c^k}{M}, \Delta \frac{c^k}{M} \right) \\ &= \Delta \frac{c^k}{M}. \end{aligned}$$

Substituting this upper bound back into the recurrence gives:

$$\begin{aligned} V_m &\leq \max_{\theta \geq 0} \Big(\Pr(X \geq \theta) \cdot \Delta \frac{c^k}{M} + \Pr(X < \theta) \cdot \Delta \frac{c^k}{M} \Big) \\ &= \Delta \frac{c^k}{M} \end{aligned}$$

Therefore, the overall expected utility of any online deterministic algorithm on the full sequence of n options is bounded by $V_n \leq \Delta \frac{c^k}{M}$.

Finally, the competitive ratio is bounded by the prophet's expected utility divided by the algorithm's expected utility:

$$\frac{\Omega\left(c^k\right)}{O\left(\frac{c^k}{M}\right)} = \Omega(M)$$

Substituting our parameters defined above $M = n^{1/k}$ and $k \leq \sqrt{\ln n / \ln c}$, we obtain

$$\Omega\left(\exp\left(\sqrt{\ln n \ln c}\right)\right).$$

This proves that no algorithm can achieve a competitive ratio better than $\Omega(\exp(\sqrt{\ln n \ln c}))$.

□

Bibliography

[BIK07] Moshe Babaioff, Nicole Immorlica, and Robert Kleinberg. "Matroids, secretary problems, and online mechanisms". In: *Symposium on Discrete Algorithms (SODA'07)*. 2007, pp. 434–443.

[Bab+09] Moshe Babaioff et al. "Secretary problems: weights and discounts". In: *Proceedings of the twentieth annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM. 2009, pp. 1245–1254.

[BBD85] John M. Barron, John Bishop, and William C. Dunkelberg. "Employer search: The interviewing and hiring of new employees". In: *Review of Economics and Statistics* 67.1 (1985), pp. 43–52.

[BG78] R Bartoszyński and Z Govindarajulu. "The secretary problem with interview cost". In: *Sankhyā: The Indian Journal of Statistics, Series B* (1978), pp. 11–28.

[Ber06] Donald A. Berry. "Bayesian clinical trials". In: *Nature Reviews Drug Discovery* 5 (2006), pp. 27–36.

[Dyn63] Evgenii Borisovich Dynkin. "The optimum choice of the instant for stopping a Markov process". In: *Soviet Mathematics Doklady* 4 (1963), pp. 627–629.

[GH25] Aadityan Ganesh and Jason Hartline. "Combinatorial Pen Testing (Or Consumer Surplus of Deferred-Acceptance Auctions)". In: *Innovations in Theoretical Computer Science (ITCS)*. 2025, 52:1–52:22.

[Gom95] Paul A. Gompers. "Optimal investment, monitoring, and the staging of venture capital". In: *The Journal of Finance* 50.5 (1995), pp. 1461–1489.

[Gup+19] Anupam Gupta et al. "The markovian price of information". In: *International Conference on Integer Programming and Combinatorial Optimization*. Springer. 2019, pp. 233–246.

[Jon90] Martin Jones. "Prophet inequalities for cost of observation stopping problems". In: *Journal of Multivariate Analysis* 34.2 (1990), pp. 238–253.

[KS03] Steven N. Kaplan and Per Strömberg. "Financial contracting theory meets the real world: An empirical analysis of venture capital contracts". In: *Review of Economic Studies* 70.2 (2003), pp. 281–315.

[KW12] Robert Kleinberg and Seth Matthew Weinberg. "Matroid prophet inequalities". In: *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*. 2012, pp. 123–136.

[Kle05] Robert D Kleinberg. "A multiple-choice secretary algorithm with applications to online auctions." In: *SODA*. Vol. 5. 2005, pp. 630–631.

[Kös04] Holger Kösters. "Difference prophet inequalities for $[0, 1]$-valued i.i.d. random variables with cost for observations". In: *The Annals of Probability* 32.4 (2004), pp. 3324–3332.

[KS77] Ulrich Krengel and Louis Sucheston. "Semiamarts and finite values". In: *Bulletin of the American Mathematical Society* 83.4 (1977), pp. 745–747.

[KS78] Ulrich Krengel and Louis Sucheston. "On semiamarts, amarts, and processes with finite value". In: *Probability on Banach Spaces*. Ed. by James Kuelbs. Vol. 4. Advances in Probability and Related Topics. Marcel Dekker, 1978, pp. 197–266.

[Lin61] Dennis V. Lindley. "Dynamic programming and decision theory". In: *Journal of the Royal Statistical Society, Series C* 10.1 (1961), pp. 39–51.

[OS11] Paul Oyer and Scott Schaefer. "Personnel economics: Hiring and incentives". In: *Handbook of Labor Economics*. Vol. 4B. Elsevier, 2011, pp. 1769–1823.

[QV23] Mingda Qiao and Gregory Valiant. "Online Pen Testing". In: *Innovations in Theoretical Computer Science (ITCS)*. 2023, 91:1–91:26.

[Sam84] Ester Samuel-Cahn. "Comparison of threshold stop rules and maximum for independent nonnegative random variables". In: *The Annals of Probability* 12.4 (1984), pp. 1213–1216.

[Sam92] Ester Samuel-Cahn. "A difference prophet inequality for bounded i.i.d. variables, with cost for observations". In: *The Annals of Probability* 20.3 (1992), pp. 1222–1228.

[Sin18] Sahil Singla. “The price of information in combinatorial optimization”. In: *Proceedings of the twenty-ninth annual ACM-SIAM symposium on discrete algorithms*. SIAM. 2018, pp. 2523–2532.

[Wei79] Martin L. Weitzman. “Optimal search for the best alternative”. In: *Econometrica* 47.3 (1979), pp. 641–654.