Prophet Inequalities: Theory and Methods

José Correa, Paul Dütting, Michal Feldman, Thomas Kesselheim,
Brendan Lucier

November 19, 2025

First Draft, with Chapters 1–4.
Published on: https://prophetinequalities.github.io/

Contents (Tentative)

PART ONE

SINGLE CHOICE

1
Introduction

In many real-world scenarios, individuals and organizations must make immediate and irrevocable decisions before knowing which alternatives might be possible in the future. These *online* decision problems often involve accepting or rejecting opportunities as they arise, such as accepting a job offer or deciding on a purchase offer for a car or an Airbnb apartment. A common characteristic of these scenarios is that the rewards or outcomes are uncertain and only revealed over time.

A fundamental question arises: how should we measure the performance of a decision-maker under these conditions? This problem dates back to the 1970s and is rooted in the theory of stopping problems. A natural measure for evaluating performance is the expected reward, calculated over all anticipated randomness in the decision problem. This expected reward is then measured relative to a benchmark. An optimistic yet common benchmark is the expected best possible outcome in hindsight; that is, the highest expected reward that the decision-maker could have achieved had they known all possible options and rewards in advance.

At first glance, this benchmark may seem unattainably strong, akin to having the foresight of a "prophet." Quite remarkably, it is actually possible to approximate this benchmark surprisingly well. Specifically, it is always possible to achieve at least half of the expected value of the optimal solution in hindsight. This foundational result has come to be known as the "prophet inequality." The prophet inequality forms the basis of much of the modern study of decision-making under uncertainty, providing a framework for analyzing and improving online algorithms.

1.1 Formal Model

Fix an integer $n \geq 1$. Consider n non-negative independent random variables $V_1, \ldots, V_n$, distributed according to $F_1, \ldots, F_n$, respectively. The random variables are realized sequentially so that at time i, we observe the realization v_i of $V_i \sim F_i$. We write $\boldsymbol{V} = (V_1, \ldots, V_n)$ and $\boldsymbol{F} = F_1 \times \ldots \times F_n$, so that $\boldsymbol{V} \sim \boldsymbol{F}$.

An online algorithm ("gambler"), knowing $\boldsymbol{F}$ from the outset, gets to observe the realizations v_i one-by-one. At each time i, it can decide to either accept v_i, in which case the game ends with reward (or value) v_i, or continue to the next step, in which case v_i is lost forever. We denote by $\textsc{Alg}(\boldsymbol{V})$ the value (random variable) a given algorithm $\textsc{Alg}$ obtains. The goal of the online algorithm is to maximize the expected reward it achieves, namely $\mathbf{E}[\textsc{Alg}(\boldsymbol{V})]$. We denote the latter quantity as $\textsc{Alg}(\boldsymbol{F})$. We write $\textsc{Opt}^{\text{ON}}(\boldsymbol{F})$ for the expected reward achieved by the optimal online algorithm under input $\boldsymbol{F}$; that is, the online algorithm that maximizes the expected reward.

It will be often convenient, when clear from context, to drop the dependency on $\boldsymbol{F}$ and simply use the terms $\textsc{Alg}$ and $\textsc{Opt}^{\text{ON}}$ to denote the previously defined quantities.

We measure the performance of an online algorithm against the "prophet" benchmark, defined as follows.[1] A "prophet" can foresee all the realizations v_i in advance and can simply pick the maximum value. Therefore, her expected reward is the expected maximum value. This prophet is equivalent to the optimal offline algorithm, which can select the value after all values have arrived. In other words, the optimal offline algorithm or "prophet" sees the entire sequence of values $v_1, \ldots, v_n$ at once and chooses $\max_{i \in [n]} v_i$. We denote the (random) reward of the prophet on a given instance as $\textsc{Opt}(\boldsymbol{V})$, so that its expected reward is $\textsc{Opt}(\boldsymbol{F}) = \mathbf{E}[\textsc{Opt}(\boldsymbol{V})] = \mathbf{E}[\max_{i \in [n]} V_i]$.

More precisely, we seek to establish a *prophet inequality*, which states that there exists an algorithm $\textsc{Alg}$ and an $\alpha \in [0, 1]$ such that for all n and all $\boldsymbol{F}$,

$$\textsc{Alg}(\boldsymbol{F}) \geq \alpha \cdot \textsc{Opt}(\boldsymbol{F}).$$

In other words, a prophet inequality asserts that an online algorithm ("gambler") can achieve an expected value of at least an α-fraction of the expected value of the optimal offline algorithm ("prophet").

We say that an online algorithm is a *threshold* algorithm if there exist real-valued thresholds $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_n)$, such that the online algorithm accepts value v_i if $v_i > \tau_i$ and rejects value v_i if $v_i < \tau_i$. When $v_i = \tau_i$ the threshold algorithm's behavior is unrestricted and potentially randomized, specified by a probability $p_i \in [0, 1]$ of accepting v_i. We sometimes refer to the probabilities

[1] This is also referred to as the "gambler" vs "prophet" ratio

p_i as a tie-breaking rule, as they specify the algorithm's behavior when value v_i ties the threshold τ_i. A threshold algorithm for which $p_i \in \{0, 1\}$ for all $i \in [n]$ is a *deterministic threshold algorithm*. If $p_i = 1$ for all i we say the threshold algorithm is *inclusive*, and if $p_i = 0$ for all i we say the threshold algorithm is *exclusive*.

Finally, a threshold algorithm is a *single threshold algorithm* if there exists a single threshold $\tau \geq 0$ and probability $p \in [0, 1]$ such that $\tau_i = \tau$ and $p_i = p$ for all $i \in [n]$. Similar to general threshold algorithms, a single threshold algorithm is *deterministic* if $p \in \{0, 1\}$, *inclusive* if $p = 1$, and *exclusive* if $p = 0$.

1.2 Optimal Policy

Our goal is to understand the expected value $\textsc{Opt}^{\text{ON}}(F_1, \ldots, F_n)$ achieved by the gambler's optimal online policy. But what is the optimal policy, exactly? Actually, it turns out that the optimal strategy of the gambler is straightforward to describe using backward induction.

Theorem 1.1 *The gambler's optimal payoff* $\textsc{Opt}^{\text{ON}}(F_1, \ldots, F_n)$ *is achieved by a threshold policy. Moreover, the appropriate thresholds can be computed by backward induction.*

Proof For each $1 \leq i \leq n$, write $\textsc{Opt}^{\text{ON}}_{\geq i}(F_1, \ldots, F_n)$ for the expected value of the optimal algorithm (not necessarily a threshold algorithm) that always declines the first $i-1$ rewards. Note then that $\textsc{Opt}^{\text{ON}}(F_1, \ldots, F_n) = \textsc{Opt}^{\text{ON}}_{\geq 1}(F_1, \ldots, F_n)$.

We will compute $\textsc{Opt}^{\text{ON}}_{\geq i}(F_1, \ldots, F_n)$ for $i \in [n]$ by backwards induction. Consider $\textsc{Opt}^{\text{ON}}_{\geq n}(F_1, \ldots, F_n)$. This is the maximum expected value attainable by a gambler standing in the final round without having accepted any rewards up to that point. As the gambler has only the final reward available to accept, it will always be optimal to accept it. This can be viewed as setting an acceptance threshold of $\tau_n = 0$. We therefore have that $\textsc{Opt}^{\text{ON}}_{\geq n}(F_1, \ldots, F_n) = \mathbf{E}[V_n]$.

Now fix any $i < n$ and consider $\textsc{Opt}^{\text{ON}}_{\geq i}(F_1, \ldots, F_n)$. After declining the first $i - 1$ prizes, the gambler observes the realization v_i of V_i and faces the choice of whether or not to accept reward v_i. If the gambler accepts, the obtained reward is precisely v_i. If the gambler rejects, the expected reward obtained will be $\textsc{Opt}^{\text{ON}}_{\geq i+1}(F_1, \ldots, F_n)$. It is, therefore, strictly optimal for the gambler to accept if $v_i > \textsc{Opt}^{\text{ON}}_{\geq i+1}(F_1, \ldots, F_n)$, and strictly optimal to reject if $v_i < \textsc{Opt}^{\text{ON}}_{\geq i+1}(F_1, \ldots, F_n)$. When the latter quantities are equal, the gambler is indifferent between accepting or rejecting.

The optimal policy, therefore, sets a threshold of $\tau_i = \textsc{Opt}^{\text{ON}}_{\geq i+1}(F_1, \ldots, F_n)$, and accepts the first reward $v_i \geq \tau_i$. The expected value from this policy starting at round i is then,

$$\textsc{Opt}^{\textsc{on}}_{\geq i}(F_1, \ldots, F_n) = \mathbf{E}_{V_i \sim F_i}\left[\max\{V_i, \tau_i\}\right] = \tau_i + \mathbf{E}_{V_i \sim F_i}\left[|V_i - \tau_i|_+\right].$$

Here, $|x|_+ = \max\{x, 0\}$, denotes the positive part of x. We emphasize that $\tau_i = \textsc{Opt}^{\textsc{on}}_{\geq i+1}(F_1, \ldots, F_n)$ is a constant in this expression since it is independent of V_i.

We have thus determined $\textsc{Opt}^{\textsc{on}}_{\geq i}(F_1, \ldots, F_n)$ for all i, and moreover each value $\textsc{Opt}^{\textsc{on}}_{\geq i}(F_1, \ldots, F_n)$ is obtained by a policy that rejects the first $i - 1$ rewards and then employs thresholds $\tau_i, \tau_{i+1}, \ldots, \tau_n$ for the remaining rewards. Since $\textsc{Opt}^{\textsc{on}}_{\geq 1}(F_1, \ldots, F_n)$ is precisely the gambler's optimal payoff $\textsc{Opt}^{\textsc{on}}(F_1, \ldots, F_n)$, we conclude that the threshold algorithm with thresholds defined by:

$$\begin{aligned} \tau_n &= \mathbf{E}\left[V_n\right] \\ \tau_{i-1} &= \tau_i + \mathbf{E}_{V_i \sim F_i}\left[|V_i - \tau_i|_+\right], \quad i = n, \ldots, 2, \end{aligned}$$

is optimal. □

Theorem 1.1 gives a full description of the gambler's optimal policy for a fixed sequence of reward distributions. But there is still much to do! In particular, while we know *how* to achieve value $\textsc{Opt}^{\textsc{on}}(\boldsymbol{F})$ (and how to calculate it), our primary goal is to understand the comparison between $\textsc{Alg}(\boldsymbol{F})$, obtained by different, hopefully simple algorithms, and the offline benchmark $\textsc{Opt}(\boldsymbol{F})$. In particular we are interested in studying not only how $\textsc{Opt}^{\textsc{on}}(\boldsymbol{F})$ performs when compared to $\textsc{Opt}(\boldsymbol{F})$, but also how single threshold algorithms perform. This is the core contribution of the prophet inequality, which we describe next.

1.3 The Classic Prophet Inequality

In this section, we state the basic prophet inequality. This result states that, when faced with a sequence of independent random variables $V_1, \ldots, V_n$, the gambler can obtain, as reward, half of what the prophet can get. Interestingly, this is best possible as shown by the following example.

Example 1.1 Consider $V_1 = 1$ a deterministic random variable and let V_2 be a *long shot*, namely a random variable taking the value $1/\varepsilon$ with probability ε and value 0 with probability $1 - \varepsilon$. In this example, upon seeing V_1, the gambler is indifferent between stopping at V_1 and getting a reward of 1, or continuing to V_2 to get an expected reward of $\varepsilon \cdot 1/\varepsilon = 1$. Therefore, in this instance, $\textsc{Opt}^{\textsc{on}}(\boldsymbol{F}) = 1$. On the other hand, the prophet can see whether V_2 takes the value $1/\varepsilon$ in advance and therefore gets $\text{Opt}(F_1, F_2) = \varepsilon \cdot 1/\varepsilon + (1-\varepsilon) \cdot 1 = 2-\varepsilon$. Thus, as $\varepsilon \to 0$, we have that $\text{Opt}^{\text{ON}}(\boldsymbol{F})/\text{Opt}(\boldsymbol{F}) \to 1/2$.

Theorem 1.2 (The Prophet Inequality) *Fix an integer $n \geq 1$. Consider a sequence of n non-negative independent random variables $\boldsymbol{V} = (V_1, \ldots, V_n)$, such that $V_i \sim F_i$, so that $\boldsymbol{V}$ is distributed according to $\boldsymbol{F} = F_1 \times \ldots \times F_n$. Then,*

$$\text{Opt}^{\text{ON}}(\boldsymbol{F}) \geq \frac{1}{2} \cdot \text{Opt}(\boldsymbol{F}).$$

Furthermore, the previous inequality can be achieved by a single threshold algorithm.

The classic prophet inequality (Theorem 1.2) answers both of our main questions above. First, it shows that, even in worst-case instances, the gambler can do almost as well as the prophet. Second, it shows that it can do so using a particularly simple algorithm that precomputes a single threshold and accepts the first reward that exceeds this threshold.

We will review several different proofs of Theorem 1.2 in Chapter 2. In particular, we shall see that there are different algorithms and choices of thresholds that yield the optimal guarantee of $1/2$. The fact that the optimal guarantee can be achieved with a *single* threshold is rather remarkable. Indeed, any such rule will not be instance-wise optimal (e.g., setting any non-zero threshold for the final round is clearly sub-optimal). In addition, we shall see that there is in fact a *deterministic* single threshold algorithm that achieves the optimal guarantee of $1/2$. Again this is rather remarkable, as the optimal single threshold algorithm for a given instance typically requires randomization (see Example 1.2). Finally, we shall see that it in order to set a threshold that achieves the optimal guarantee of $1/2$ it is sufficient for the gambler to know aggregate statistics of the distribution of the maximum reward.[2]

Example 1.2 Consider the case of two identically distributed random variables, V_1 and , V_2. Each is equally likely to take value 1 or 2. The optimal online algorithm for this problem instance is very simple: Accept V_1 if it equals 2; otherwise reject it and take V_2. This algorithm obtains an expected value of $2 \cdot 3/4 + 1 \cdot 1/4 = 1.75$, which turns out to be equal to the prophet's reward for the instance.

What about the optimal single threshold algorithm? If the single threshold algorithm is deterministic, then there are only two possibilities: either the algorithm only accepts rewards of value 2, or it accepts any reward of value 1

[2] In fact, as we shall see in Chapter 5, the optimal guarantee of $1/2$ can be obtained even if the gambler has access only to a single sample from each distribution.

or 2. In the former case (e.g., by setting the algorithm's threshold τ to be 2 and inclusively accepting in case of a tie), the expected reward is $2 \cdot 3/4 = 1.5$. In the latter case (e.g., by setting the threshold τ to 1 and inclusively accepting in case of a tie) then the algorithm always accepts V_1 and again receives 1.5 in expectation.

We can do better, however, if we allow the single threshold to use randomized tie-breaking. Suppose we set the threshold to be 1 and choose $p_1 = p_2 = 1/2$. We then have four cases, each happening with probability $1/4$:

- $V_1 = 2, V_2 = 2$. Here we obtain 2 since we take V_1
- $V_1 = 2, V_2 = 1$. Here we obtain 2 since we take V_1
- $V_1 = 1, V_2 = 2$. Here we take V_1 with probability $1/2$ and V_2 with probability $1/2$, for a total expected value of $3/2$.
- $V_1 = 1, V_2 = 1$. Here we take V_1 with probability $1/2$ and V_2 with probability $1/2 \cdot 1/2 = 1/4$, for a total expected value of $3/4$.

Thus overall the single threshold algorithm (with the uniform tie-breaking probability of $1/2$) obtains: $2 \cdot 1/4 + 2 \cdot 1/4 + 3/2 \cdot 1/4 + 3/4 \cdot 1/4 = 25/16$.

1.4 Applications

The prophet inequality shows that it is possible to obtain, in expectation, half of what would be achievable if all rewards were revealed in advance. Moreover, this guarantee can be obtained with a simple threshold rule that simply accepts the first reward higher than a pre-computed target quantity that depends only on the distribution of the maximum reward. This may at first seem surprising and unintuitive. But before we discuss different ways to prove this result in Chapter 2, let's explore some of its implications.

First, the fact that the prophet inequality can be achieved with a threshold rule that depends only on the distribution of the maximum reward has immediate practical implications. Since the threshold depends only on the distribution of the maximum reward, it is independent of the order in which the rewards arrive. This means that the threshold policy can be implemented even if the order of arrival is not known in advance. Moreover, once we have determined the threshold, the choice of whether or not to accept a reward or not does not depend on the reward's distribution, only its realized value. This means that we don't even need to know the order of the rewards as they arrive, or even the identity of any given box; it's enough to just only observe the rewards themselves. So a policy that achieves the prophet inequality guarantee can be implemented in any setting where we have advance knowledge of the distribution over the maximum reward, and then we have to process rewards in any order (even a worst-case order).

Using a threshold rule for decision-making is also significant for another reason: thresholds have natural interpretations in many practical contexts. Imagine, for example, that we would like to give something away; a used bicycle, perhaps. We would prefer that the bicycle goes to someone who would make good use of it. Potential recipients arrive over time, each with an unknown value for getting the bicycle. The prophet inequality suggests that we can approximate the best allocation in hindsight by setting a threshold "value" in advance, and then giving the bicycle to whichever potential recipient first exceeds that threshold. At first glance this feels like a difficult policy to implement for many reasons, not least of which is that a recipient's "value" can be difficult to determine! But if we interpret value as willingness to pay, then this threshold policy is exactly equivalent to posting a take-it-or-leave-it price on the bicycle and selling to the first person willing to accept the offer. Indeed, this connection between threshold rules and posted prices has been a common refrain in recent uses of prophet inequalities in economic settings, and one that we will explore further in future chapters.

The example above is an instance of a welfare-maximization problem in resource allocation. Online decision problems in the spirit of the prophet inequality arise in other contexts as well. For instance, a firm looking to fill a critical position may need to accept or reject candidates before all possible candidates have been interviewed. In this case, the prophet inequality suggests that setting an appropriate threshold on candidate quality will achieve a constant approximation to the expected value of the best candidate in hindsight. Prophet inequalities are also useful for online revenue maximization, where connections between expected revenue and a quantity known as virtual surplus means that one can approximate online the best possible revenue that could be raised by selling a good even if all potential buyers were together in one room and competing against one another.

1.5 Generalizations

The prophet inequality applies to a simple and clean model. While the baseline setting already has numerous applications, there are many contexts to which it *almost* applies, or for which we can make additional structural assumptions that might improve what is possible. As such, it is natural to consider many possible extensions. Some are related to applications of markets that are more complex than simple accept/reject decisions on rewards; others to different assumptions about what information is available to the decision maker and when; and yet others relate to weaker, perhaps more realistic, benchmarks. For each of these we can explore the power of online policies and study different algorithmic approaches to the corresponding decision problems. Here we summarize some of these extensions and special cases, many of which we will explore in the later chapters of this book.

Identical distributions. The baseline prophet inequality applies to arbitrary reward distributions. But what about the case where the distributions are identical, meaning that the reward is drawn from the same distribution each round? Can one obtain an improved prophet inequality in this case? It turns out that if one insists on a threshold rule, then $1/2$ is still the best possible approximation factor. However, more generally, it is possible to improve upon this factor if we allow more general online policies. We explore this setting in Chapter 3.

Arrival order. The standard prophet inequality assumes that rewards arrive in an arbitrary but known order. As we've already discussed, it is possible to achieve the prophet inequality bound of 2 even if the arrival order is unknown. But are improved approximations possible for other arrival orders? What if rewards come in a uniformly random order, or if the designer can choose to observe the rewards in whichever order they prefer? It turns out that even for a random order, a single-threshold policy can achieve better than $1/2$ of the expected offline optimum. This variant of the prophet inequality is the subject of Chapter 4.

Accepting multiple rewards. Motivated in part by resource allocation problems, there has been an explosion of interest in prophet inequalities for richer classes of decision problems. For example, what if the decision-maker is allowed to choose multiple rewards? For instance, what if the decision-maker can select at most k rewards? More generally, there may be some downward-closed feasibility constraint determining which subsets of rewards can be simultaneously selected. We discuss the problem of selecting up to k rewards, and general downward-closed feasibility constraints in Chapter 6. Afterwards, in Chapter 7, we discuss multi-choice problems on matroids, which is a family of downward-closed feasibility constraints that generalizes the problem of selecting up to k rewards.

Combinatorial decisions. We can likewise consider extensions of the prophet inequality that allow more complex decisions in each round. For example, perhaps rather than choosing whether or not to accept a given reward, the decision-maker must choose one of k possible recipients for each reward, with different recipients having different values. Or perhaps the decision-maker has access to a pool of resources up-front, and must choose which subset of those resources (if any) to allocate to potential recipients who themselves arrive online. These matching and combinatorial allocation settings are the subject of Chapter 8 and Chapter 9, respectively.

Balanced prices and contention resolution. In Chapter 11 and Chapter 12, we review two of the leading techniques for showing prophet inequalities in combinatorial settings. The first is the so-called "balanced prices" approach, and the second is the (online) contention resolution approach. We will see how these abstract and unify the majority of the proofs that we discussed in earlier chapters. The discussion of these approaches presents more advanced and technical material, which is not essential for the following chapters.

Imperfect knowledge of the reward distributions. We explore the case where the decision maker has limited information about the distributions. One common way to quantify this is to assume that this partial knowledge takes the form of *samples*: the decision-maker can observe one or more draws from the reward distributions, and can use that information when making their online decisions. As it turns out, even having only a single sample from each reward distribution is enough to recover the factor $1/2$ from the prophet inequality. We discuss this and other settings with data sample access in Chapter 5 and Chapter 10.

Comparison with other benchmarks The prophet inequality compares the expected reward of an online policy with the expected best reward in hindsight. It is natural to ask about other benchmarks, as well. How does a given policy, or class of policies, compare with the best possible online policy? What if we compare with the best reward in hindsight, but under some additional restrictions—say, observing fewer rewards in an i.i.d. setting, or being able to accept fewer rewards in a scenario where one can accept multiple times. We discuss these and other alternative benchmarks in Chapter 13.

Take-Aways

- A prophet inequality measures the performance of an online, stochastic reward-selection policy by comparing its expected performance against the expected value of the best choices in hindsight.
- The original prophet inequality of Krengel, Sucheston, and Garling

from 1977/78 establishes a tight 2-approximation for irrevocably choosing a single reward.
- An important feature of the classic prophet inequality is that it can be implemented with a simple threshold policy, enabling practical applications like pricing.
- Since 2007 the area has seen an enormous amount of interest from computer science, leading to new theory and applications. A main goal of this book, in addition, to introducing the basic machinery and results, is to guide the reader through these new developments.

Chapter Notes

The two foundational studies on prophet inequalities are (Krengel and Sucheston, 1977b) and (Krengel and Sucheston, 1978). The titles of these studies, "Semiarts and Finite Values" and "On Semiamarts, Amarts, and Processes with Finite Values," point to the more general nature of the stochastic processes considered. The main result for which these garnered fame, is the tight prophet inequality of $1/2$ discussed in this chapter. Notably, Krengel and Sucheston credit D. J. H. (Ben) Garling for showing the tight $1/2$ factor, writing that they themselves only showed that the factor must be between $1/4$ and $1/2$. For this reason the classic prophet inequality is also sometimes credited to Krengel, Sucheston, and Garling.

The original studies by Krengel and Sucheston did not use the gambler-vs-prophet analogy. According to Harten et al. (1997), the term "prophet" was first used in Krengel and Sucheston (1977a). The work by Krengel and Sucheston sparked a new field in probability theory, with many exciting contributions, by authors such as Theodore P. Hill, Douglas P. Kennedy, Robert P. Kertz, and Ester Samuel-Cahn. Among this important first wave of studies, is the very elegant work by Samuel-Cahn (1984), who provided a much simpler proof of the optimal $1/2$ factor established by Krengel and Sucheston, and showed that this factor can indeed be achieved with a simple single-threshold policy.

A second wave of work, this time mostly in computer science rather than mathematics, was triggered by Hajiaghayi et al. (2007), who used the standard prophet inequality (and generalizations thereof) as a tool for online mechanism design and pricing problems. Motivated in part by applications in economic problems, but also realizing that prophet inequalities offer a new, less pessimistic paradigm for the analysis of online algorithms, the past 15–20 years witnessed an explosive growth of work on prophet inequalities in computer science. This has led to numerous breakthroughs ranging from generalizations of the classic prophet inequality to more complex settings such as online matching or online combinatorial auctions, to data-driven prophet inequalities that relax the assumption that the distributions from which the rewards are sampled are fully known.

There are excellent mathematical text books on the broader topic of optimal stopping (Chow et al., 1971; Shiryaev, 2007; Peskir and Shiryaev, 2006). There is also a dedicated survey on prophet inequalities (Hill and Kertz, 1992), and even a book (Harten et al., 1997). However, all of these predate (and do not cover) the recent developments in computer science on which we focus in this book. The closest in this regard are the surveys by Lucier (2017) and Correa et al. (2019)—but these are much less comprehensive.

References

Chow, Yuan Shih, Robbins, Herbert, and Siegmund, David. 1971. *Great Expectations: The Theory of Optimal Stopping*. Houghton Mifflin, Boston.

Correa, Jose, Foncea, Patricio, Hoeksma, Ruben, Oosterwijk, Tim, and Vredeveld, Tjark. 2019. Recent developments in prophet inequalities. *SIGecom Exch.*, **17**(1), 61–70.

Hajiaghayi, Mohammad Taghi, Kleinberg, Robert D., and Sandholm, Tuomas. 2007. Automated Online Mechanism Design and Prophet Inequalities. Pages 58–65 of: *AAAI 2007*. AAAI Press.

Harten, Friedrich, Meyerthole, Andreas, and Schmitz, Norbert. 1997. *Prophetentheorie*. Teubner Skripten zur Mathematischen Stochastik. Vieweg+Teubner Verlag, Wiesbaden. In German.

Hill, Theodore P., and Kertz, Robert P. 1992. A Survey of Prophet Inequalities in Optimal Stopping Theory. *Contemp. Math.*, **125**(1), 191–207.

Krengel, Ulrich, and Sucheston, Louis. 1977a. Abstract #742-60-3: How to Bet Against a Prophet. *Notices of the Am. Math. Soc*, **24**(1). Available at `https://www.ams.org/journals/notices/197701/197701FullIssue.pdf`.

Krengel, Ulrich, and Sucheston, Louis. 1977b. Semiamarts and finite values. *Bull. Amer. Math. Soc.*, **83**(4), 745–747.

Krengel, Ulrich, and Sucheston, Louis. 1978. On semiamarts, amarts, and processes with finite value. Pages 197–266 of: Kuelbs, J. (ed), *Prob. on Banach Spaces*. M. Dekker, New York.

Lucier, Brendan. 2017. An economic view of prophet inequalities. *SIGecom Exch.*, **16**(1), 24–47.

Peskir, Goran, and Shiryaev, Albert. 2006. *Optimal stopping and free-boundary problems*. Lectures in Mathematics (ETH Zürich). Birkhäuser Basel.

Samuel-Cahn, Ester. 1984. Comparison of Threshold Stop Rules and Maximum for Independent Nonnegative Random Variables. *Ann. Probab.*, **12**(4), 1213–1216.

Shiryaev, Albert N. 2007. *Optimal Stopping Problems*. Springer Berlin, Heidelberg. Original Russian edition published by Nauka, Moscow, 1976, 2nd ed.

2
Four Proofs of the Prophet Inequality

In this chapter, we present four proofs of the basic prophet inequality theorem, namely Theorem 1.2. Recall that this result asserts that a gambler, when faced with a sequence of random rewards that are revealed online, can guarantee an expected reward that is at least half of the expected reward that a prophet with complete foresight could achieve by simply picking the maximum reward.

Theorem 1.2 (The Prophet Inequality) *Fix an integer $n \geq 1$. Consider a sequence of n non-negative independent random variables $\boldsymbol{V} = (V_1, \ldots, V_n)$, such that $V_i \sim F_i$, so that $\boldsymbol{V}$ is distributed according to $\boldsymbol{F} = F_1 \times \ldots \times F_n$. Then,*

$$\textsc{Opt}^{\textsc{on}}(\boldsymbol{F}) \geq \frac{1}{2} \cdot \textsc{Opt}(\boldsymbol{F}).$$

Furthermore, the previous inequality can be achieved by a single threshold algorithm.

In Section 2.1 we give the classic, arguably simplest proof. It is constructive: it identifies an explicit threshold that attains the algorithm's guarantee, and in fact shows that more than one threshold works. Two natural choices are the median and the mean of the distribution of the maximum reward. As discussed at the end of the chapter, the mean-based rule underlies the *balanced prices* approach, which extends to broader settings.

The second proof, presented in Section 2.2, follows a similar approach but replaces the direct analysis of the gambler's expected reward with a stochastic-dominance viewpoint, bounding the probability that the gambler's reward exceeds a given level.

The third and fourth proofs are much more different, and indeed, they do not establish the second part of Theorem 1.2 in that, while they do establish a prophet inequality of ratio $1/2$, they fail to show that this can be achieved with a single threshold. The third proof, presented in Section 2.4 is based on the important idea of *contention resolution* and furthermore compares the rewards of the gambler to the *ex-ante* relaxation, a certain optimization problem that upper bounds the reward of the prophet. Finally, the proof presented in Section 2.5 is one of the earliest proofs of the result and it is based on a completely different idea: It inductively shows that worse and worse ratios are obtained by replacing the first and last two random rewards by a deterministic reward and a *long-shot*. In the end, we are left with an instance like that in Example 1.1 and can easily conclude.

Our rationale for going through different proof techniques is twofold: First, different proofs uncover different features of the classic prophet inequality. Second, often the discovery of a new proof technique enabled progress in more complex settings.

2.1 Proof 1: Median Rule/Balanced Prices

We start by considering a single threshold τ and the inclusive single-threshold algorithm, $\textsc{Alg}$, that accepts the first reward greater than or equal to τ. Define $q = \mathbf{Pr}[\max_{i\in[n]} V_i \geq \tau]$, the probability that some random variable exceeds the threshold (including the case of a tie). We further define $|x|_+ = \max\{x, 0\}$.

Clearly, if some box exceeds the threshold, $\textsc{Alg}$ gets at least the threshold τ. We refer to this reward as the baseline. On top of this, the reward $\textsc{Alg}$ gets may exceed the threshold. For instance, if $\textsc{Alg}$ stops with the i-th random variable, having value v_i, then $\textsc{Alg}$ gets a bonus $\text{bonus}_i = v_i - \tau = |v_i - \tau|_+$. With this we may write the expected reward of $\textsc{Alg}$ as the baseline plus the sum of the bonuses.

$$\begin{aligned}
\textsc{Alg}(\boldsymbol{F}) &= \mathbf{E}\left[\textsc{Alg}(\boldsymbol{V})\right] \\
&= \mathbf{E}\left[\sum_{i=1}^{n} V_i \cdot \mathbb{1}(V_1 < \tau) \cdots \mathbb{1}(V_{i-1} < \tau) \cdot \mathbb{1}(V_i \geq \tau)\right] \\
&= \mathbf{E}\left[\sum_{i=1}^{n} |V_i - \tau|_+ \cdot \mathbb{1}(V_1 < \tau) \cdots \mathbb{1}(V_{i-1} < \tau)\right] \\
&\quad + \tau \cdot \mathbf{E}\left[\sum_{i=1}^{n} \mathbb{1}(V_1 < \tau) \cdots \mathbb{1}(V_{i-1} < \tau) \cdot \mathbb{1}(V_i \geq \tau)\right] \\
&= \sum_{i=1}^{n} \mathbf{E}\left[|V_i - \tau|_+ \cdot \mathbb{1}(V_1 < \tau) \cdots \mathbb{1}(V_{i-1} < \tau)\right] + \tau q.
\end{aligned}$$

Here, the first term corresponds to the sum of the bonuses while the second corresponds to the baseline. For the latter, we observe that the sum of the indicators equals q since $\sum_{i=1}^{n} \mathbb{1}(V_1 < \tau) \cdots \mathbb{1}(V_{i-1} < \tau) \cdot \mathbb{1}(V_i \geq \tau)$ equals 1 if and only if there is a value V_i that exceeds or matches the threshold. Continuing from the last expression we conclude that:

$$\begin{aligned}
\textsc{Alg}(\boldsymbol{F}) =& \sum_{i=1}^{n} \mathbf{E}\left[|V_i - \tau|_+\right] \mathbf{Pr}\left[V_1 < \tau, \ldots, V_{i-1} < \tau\right] + \tau q. \\
\geq& \sum_{i=1}^{n} \mathbf{E}\left[|V_i - \tau|_+\right] \mathbf{Pr}\left[V_1 < \tau, \ldots, V_n < \tau\right] + \tau q. \\
=& (1-q) \sum_{i=1}^{n} \mathbf{E}\left[|V_i - \tau|_+\right] + \tau q. \\
\geq& (1-q)\, \mathbf{E}\left[\sum_{i=1}^{n} |V_i - \tau|_+\right] + \tau q. \\
\geq& (1-q)\, \mathbf{E}\left[\max_{i \in [n]} |V_i - \tau|_+\right] + \tau q. \\
\geq& (1-q)(\textsc{Opt}(\boldsymbol{F}) - \tau) + \tau q.
\end{aligned} \tag{2.1}$$

The first equality follows from the assumption of independence. The second step follows since we include more events in the probability. The third step holds the definition of q. The next follows by linearity of expectation. The fifth step follows by lower bounding the sum of nonnegative terms by the maximum. And the final step follows by removing the $|\cdot|_+$, and the definition of $\textsc{Opt}(\boldsymbol{F})$.

To conclude the proof, we need to specify the choice of τ and its corresponding $q = \mathbf{Pr}[\max_{i \in [n]} V_i \geq \tau]$. Interestingly, there are several possible choices. For instance, if $\tau = 1/2 \cdot \textsc{Opt}(\boldsymbol{F})$ (i.e., the mean of the distribution of $\max_{i \in [n]} V_i$), then regardless of the value of q we obtain

$$\textsc{Alg}(\boldsymbol{F}) \geq \frac{1}{2} \cdot \textsc{Opt}(\boldsymbol{F}).$$

Alternatively, if there exists some τ such that $q = 1/2$ (e.g., τ is the median of the distribution of $\max_{i \in [n]} V_i$ and is not an atom of that distribution), this yields

$$\textsc{Alg}(\boldsymbol{F}) \geq (1-q)(\textsc{Opt}(\boldsymbol{F}) - \tau) + \tau q = \frac{1}{2} \cdot \textsc{Opt}(\boldsymbol{F}).$$

Moreover, whenever τ lies strictly between the mean and the median of $\max_{i \in [n]} V_i$, the result holds. To see this, note that any scenario where τ lies strictly between the mean and median falls into one of two cases:

1. $\tau \geq \frac{1}{2} \cdot \textsc{Opt}(\boldsymbol{F})$ and $q \geq \frac{1}{2}$ (the mean is above the median), or

2. $\tau \le \frac{1}{2} \cdot \textsc{Opt}(\boldsymbol{F})$ and $q \le \frac{1}{2}$ (the mean is below the median).

In the former case, since $2q - 1 \ge 0$ we have that:

$$\textsc{Alg}(\boldsymbol{F}) \ge (1-q)\textsc{Opt}(\boldsymbol{F}) + (2q-1)\tau \ge (1-q+2q-1)\textsc{Opt}(\boldsymbol{F}) \ge \frac{1}{2} \cdot \textsc{Opt}(\boldsymbol{F}).$$

Similarly, in the latter case, since $2q - 1 \le 0$, we have that

$$\textsc{Alg}(\boldsymbol{F}) \ge (1-q)\textsc{Opt}(\boldsymbol{F}) + (2q-1)\tau \ge (1-q)\textsc{Opt}(\boldsymbol{F}) \ge \frac{1}{2} \cdot \textsc{Opt}(\boldsymbol{F}).$$

We can make two important conclusions from this proof. First, only very limited information is needed to set the threshold τ (such as the mean or median of the distribution of the maximum reward). Second, with the mean-based rule, there is always an inclusive (and hence deterministic) single threshold algorithm that achieves the optimal guarantee of $1/2$.

2.2 Proof 2: Approximate Stochastic Dominance

We now turn to the second proof of Theorem 1.2. While similar to the previous proof, this one essentially establishes that, for all value $x \ge 0$, the probability that the gambler gets a reward of x or more, is at least half that of the prophet getting x or more. This introduces a new technique that we call *approximate stochastic dominance*, and that will be useful in future developments.

Again, pick a threshold τ and consider the inclusive single-threshold algorithm $\textsc{Alg}$ that accepts the first reward of value τ or more. Let r be the index of the first sampled value above the threshold. Thus, V_r denotes the reward of $\textsc{Alg}$ so that $\textsc{Alg}(\boldsymbol{F}) = \mathbf{E}[V_r]$. For convenience let also $\hat{V} := \max_{1 \le i \le n} V_i$ be the expected maximum reward, and thus $\textsc{Opt}(\boldsymbol{F}) = \mathbf{E}[\hat{V}]$. We show that $\mathbf{E}[V_r] \ge 1/2 \cdot \mathbf{E}[\hat{V}]$.

As in the previous proof, we let q be the probability that some value meets or exceeds τ, i.e., $q = \mathbf{Pr}[\hat{V} \ge \tau]$. Note that for any nonnegative value x we have that:

$$\mathbf{Pr}\,[V_r > x] \ge \begin{cases} q & x < \tau\,, \\ (1-q)\,\mathbf{Pr}[\hat{V} > x] & x \ge \tau\,. \end{cases}$$

Indeed, the first case is trivial since we only stop when seeing a value of at least τ and the probability that this value exists equals q. On the other hand, for $x > \tau$, by conditioning on the stopping time and using the union bound, we get

$$\mathbf{Pr}\,[V_r > x] = \sum_{i=1}^{n} \mathbf{Pr}\,[V_i > x] \prod_{j<i} \mathbf{Pr}\left[V_j < \tau\right]$$

$$\geq (1-q)\sum_{i=1}^{n} \mathbf{Pr}\left[V_i > x\right] \geq (1-q)\,\mathbf{Pr}\left[\hat{V} > x\right].$$

The latter inequality allows us to lower bound our reward as follows:

$$\begin{aligned}\mathbf{E}\left[V_r\right] = \int_0^\infty \mathbf{Pr}\left[X_r > x\right] dx &\geq \int_0^\tau q dx + (1-q)\int_\tau^\infty \mathbf{Pr}\left[\hat{V} > x\right] dx \\ &\geq q\tau + (1-q)\left(\mathbf{E}\left[\hat{V}\right] - \tau\right) = q\tau + (1-q)\left(\textsc{Opt}(\boldsymbol{F}) - \tau\right).\end{aligned}$$

The last inequality follows since $\mathbf{E}[\hat{V}] = \int_0^\infty \mathbf{Pr}[\hat{V} > x]dx \leq \tau + \int_\tau^\infty \mathbf{Pr}[\hat{V} > x]dx$. With this, we have the same inequality as (2.1) in the previous proof, so we conclude similarly.

2.3 Intermission: Handling Ties

So far we have focused attention on deterministic single threshold algorithms that are inclusive, meaning that they accept a value v_i whenever $v_i \geq \tau$. In fact, the mean-based approach in Section 2.1 shows that there is always an inclusive single threshold policy that achieves the worst-case optimal guarantee of $1/2$. However, other rules, such as the median-based rule discussed in the same section, require more care. For instance, there may not be a threshold τ such that $q = \mathbf{Pr}[\max_{i\in[n]} V_i \geq \tau] = 1/2$.

However, as we shall show next, it is always possible to convert a (without loss generality, deterministic) threshold-based algorithm that achieves a certain approximation to the prophet for continuous distributions, into a randomized threshold-based algorithm that achieves the same guarantee for general distributions.

Lemma 2.1 *Fix integer $n \geq 1$ and constant $\gamma \in [0,1]$. Suppose $\textsc{Alg}$ is a threshold algorithm such that, for any* continuous *distribution $\boldsymbol{F} = F_1\times\ldots\times F_n$,*

$$\textsc{Alg}(\boldsymbol{F}) \geq \gamma \cdot \textsc{Opt}(\boldsymbol{F}).$$

Then $\textsc{Alg}$ can be extended to a threshold algorithm $\textsc{Alg}'$ such that, for any (possibly discontinuous) distribution $\boldsymbol{F}'$, $\textsc{Alg}'(\boldsymbol{F}') \geq \gamma\textsc{Opt}(\boldsymbol{F}')$. Moreover, if $\textsc{Alg}$ is a single threshold algorithm, then $\textsc{Alg}'$ is as well.

Proof Fix any (possibly discontinuous) distribution $\boldsymbol{F} = F_1 \times \ldots \times F_n$. For each $\epsilon > 0$, we define a perturbed distribution $\boldsymbol{F}^{(\epsilon)} = F_1^{(\epsilon)} \times \ldots \times F_n^{(\epsilon)}$ as follows: to draw a value from $F_i^{(\epsilon)}$, first draw $V_i \sim F_i$ then add to V_i a value drawn independently and uniformly from $[0, \epsilon]$. Note then that, for all i and all $V_i \geq 0$, $F_i(V_i - \epsilon) \leq F_i^{(\epsilon)}(V_i) \leq F_i(V_i)$.

For any $\epsilon > 0$, $\boldsymbol{F}^{(\epsilon)}$ is a continuous distribution. Therefore, by assumption, Alg proceeds as a threshold algorithm on input $\boldsymbol{F}^{(\epsilon)}$, say with thresholds $\boldsymbol{\tau}^{(\epsilon)} = (\tau_1^{(\epsilon)}, \ldots, \tau_n^{(\epsilon)})$. For each $i \in [n]$, let $p_i^{(\epsilon)} = 1 - F_i^{(\epsilon)}(\tau_i^{(\epsilon)})$ denote the probability that element i is chosen under these thresholds, conditional on no earlier element having been chosen. We write $\mathbf{p}^{(\epsilon)} = (p_1^{(\epsilon)}, \ldots, p_n^{(\epsilon)}) \in [0,1]^n$ for the vector of such probabilities.

As Opt($\boldsymbol{F}$) is finite, we can assume without loss that the thresholds $\tau_i^{(\epsilon)}$ are uniformly bounded.[1] There therefore exists an infinite decreasing sequence $\epsilon_1 > \epsilon_2 > \epsilon_3 > \ldots$, converging to 0, such that, as $j \to \infty$, threshold $\tau_i^{(\epsilon_j)}$ converges to some τ_i and $p_i^{(\epsilon_j)}$ converges to some p_i for all i. Write $\mathbf{p} = (p_1, \ldots, p_n)$ and $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_n)$.

We are now ready to define the behavior of algorithm Alg′ on distribution $\boldsymbol{F}$. We let Alg′ be the threshold algorithm with thresholds $\boldsymbol{\tau}$, and with tie-breaking set so that, on input $\boldsymbol{F}$, each element i is chosen with probability p_i conditional on Alg′ not having chosen any prior element.

We claim first that this Alg′ is well-defined as a threshold algorithm. In particular, this requires that the claimed tie-breaking rule exists, which is true if and only if, for each i,

$$\Pr[V_i > \tau_i] \leq p_i \leq \Pr[V_i \geq \tau_i]. \tag{2.2}$$

To see why (2.2) holds, suppose first for contradiction that $\Pr[V_i > \tau_i] > p_i$. Then there exists some $\tau_i' > \tau_i$ and $p_i' > p_i$ such that $F_i(\tau_i') = 1 - p_i'$. Since $\tau_i' > \tau_i$, convergence of $\tau_i^{(\epsilon_j)}$ to τ_i implies that there exists some $\epsilon_0 < |\tau_i - \tau_i'|/2$ such that for all $\epsilon_j < \epsilon_0$, $\tau^{(\epsilon_j)} < \tau_i' - \epsilon_0$. But now, for all such j, we have

$$\begin{aligned}
p_i^{(\epsilon_j)} &= 1 - F_i^{(\epsilon_j)}(\tau_i^{(\epsilon_j)}) && \text{(definition of } p_i^{(\epsilon_j)}\text{)} \\
&\geq 1 - F_i(\tau_i^{(\epsilon_j)}) && \text{(as } F_i^{(\epsilon_j)}(x) \leq F_i(x) \text{ for all } x\text{)} \\
&\geq 1 - F_i(\tau_i') && \text{(since } \tau_i' > \tau_i^{(\epsilon_j)} + \epsilon_0\text{)} \\
&= p_i',
\end{aligned}$$

which contradicts the assumption that $p^{(\epsilon_j)}$ converges to p_i.

For the other half of (2.2), suppose for contradiction that $\Pr[V_i \leq \tau_i] < p_i$. Then (as above) there exists some $\tau_i' < \tau_i$ and $p_i' < p_i$ such that $F_i(\tau_i') = 1 - p_i'$. Further, there exists some $\epsilon_0 < |\tau_i - \tau_i'|/2$ such that for all $\epsilon_j < \epsilon_0$, $\tau^{(\epsilon_j)} > \tau_i' + \epsilon_0$.

[1] Replacing each $\tau_i^{(\epsilon)}$ with $\min\{\tau_i^{(\epsilon)}, \text{Opt}(\boldsymbol{F}) + \epsilon\}$ yields uniform boundedness and retains the property that $\text{Alg}(\boldsymbol{F}^{(\epsilon)}) \geq \gamma \cdot \text{Opt}(\boldsymbol{F}^{(\epsilon)})$ for any $\gamma \in [0,1]$.

But now, for all such j, we have

$$\begin{aligned}
p_i^{(\epsilon_j)} &= 1 - F_i^{(\epsilon_j)}(\tau_i^{(\epsilon_j)}) && \text{(definition of } p_i^{(\epsilon_j)}\text{)} \\
&\le 1 - F_i(\tau_i^{(\epsilon_j)} - \epsilon_j) && \text{(as } F_i^{(\epsilon_j)}(x) \ge F_i(x - \epsilon_j) \text{ for all } x\text{)} \\
&\le 1 - F_i(\tau_i^{(\epsilon_j)} - \epsilon_0) && \text{(since } \epsilon_0 > \epsilon_j\text{)} \\
&\le 1 - F_i(\tau_i') && \text{(since } \tau_i' < \tau_i^{(\epsilon_j)} - \epsilon_0\text{)} \\
&= p_i',
\end{aligned}$$

which contradicts the assumption that $p^{(\epsilon_j)}$ converges to p_i. Taken together, these two cases establish (2.2), so algorithm $\textsc{Alg}'$ is well-defined.

We are now ready to bound the expected reward obtained by $\textsc{Alg}'$ on distribution $\boldsymbol{F}$. For each i, write Q_i for the expected value of V_i conditioned on V_i being selected by $\textsc{Alg}'$ (accounting for both the threshold and tie-breaking rules). Noting that $\textsc{Alg}'$ considers element i with probability $\prod_{k<i}(1 - p_k)$, linearity of expectation implies

$$\textsc{Alg}'(\boldsymbol{F}) = \sum_i Q_i \prod_{k<i}(1 - p_k).$$

Recall that $p_i^{(\epsilon_j)}$ converges to p_i as $j \to \infty$. Furthermore, the expected value of the top p_i quantile of $F^{(\epsilon_j)}$ is at least Q_i and is within ϵ_j of Q_i, by definition of $F^{(\epsilon_j)}$; that is,

$$Q_i \le \mathbf{E}\left[V^{(\epsilon_j)} \mid F^{(\epsilon_j)}(V^{(\epsilon_j)}) > 1 - p_i\right] \le Q_i + \epsilon_j.$$

So, in particular, the expectation in the inequality above converges to Q_i as $j \to \infty$. We conclude that for any $\delta > 0$ there exists some $\epsilon_0 > 0$ such that, for all $\epsilon_j < \epsilon_0$,

$$\begin{aligned}
\textsc{Alg}'(\boldsymbol{F}) &\ge \sum_i \mathbf{E}\left[V^{(\epsilon_j)} \mid F^{(\epsilon_j)}(V^{(\epsilon_j)}) > 1 - p_i\right] \prod_{k<i}(1 - p_k) - \delta \\
&\ge \sum_i \mathbf{E}\left[V^{(\epsilon_j)} \mid F^{(\epsilon_j)}(V^{(\epsilon_j)}) > 1 - p_i^{(\epsilon_j)}\right] \prod_{k<i}(1 - p_k^{(\epsilon_j)}) - 2\delta \\
&= \textsc{Alg}(\boldsymbol{F}^{(\epsilon_j)}) - 2\delta.
\end{aligned}$$

But we now recall that $\textsc{Alg}(\boldsymbol{F}^{(\epsilon_j)}) \ge \gamma \textsc{Opt}(\boldsymbol{F}^{(\epsilon_j)})$ by assumption, and note that $\textsc{Opt}(\boldsymbol{F}^{(\epsilon_j)}) \ge \textsc{Opt}(\boldsymbol{F})$ since $\boldsymbol{F}^{(\epsilon_j)}$ first-order stochastically dominates $\boldsymbol{F}$. We therefore conclude that

$$\textsc{Alg}'(\boldsymbol{F}) \ge \textsc{Alg}(\boldsymbol{F}^{(\epsilon_j)}) - 2\delta \ge \gamma \textsc{Opt}(\boldsymbol{F}^{(\epsilon_j)}) - 2\delta \ge \gamma \textsc{Opt}(\boldsymbol{F}) - 2\delta$$

for all $\delta > 0$, and hence $\textsc{Alg}'(\boldsymbol{F}) \ge \gamma \textsc{Opt}(\boldsymbol{F})$ as claimed. □

2.4 Proof 3: Relax and Round

In this section, we give a third approach to the prophet inequality problem by using the so-called *ex-ante relaxation*. For this, we consider continuous and strictly increasing distributions $F_1, \ldots, F_n$ (so that we can take inverses).

The ex-ante relaxation corresponds to an optimization problem whose optimal value upper bounds the reward of the prophet. The first observation required to be able to write it down is the following.

Proposition 2.1 *If $X \sim F$ is a real random variable and A is any event of probability $a \in [0, 1]$, then*

$$\mathbf{E}\left[X|A\right] \leq \mathbf{E}\left[X|X > F^{-1}(1-a)\right].$$

Proof To see this we prove a stronger statement. The conditional random variable on the right, $X|X > F^{-1}(1-a)$, actually stochastically dominates the random variable $X|A$. Indeed,

$$\begin{aligned}
\mathbf{Pr}\left[X > x|A\right] &= \frac{\mathbf{Pr}[X > x \cap H]}{a} \\
&\leq \frac{\min\{\mathbf{Pr}[X > x], \mathbf{Pr}[H)]\}}{a} \\
&= \frac{\min\{\mathbf{Pr}[X > x], \mathbf{Pr}[X > F^{-1}(1-a)]\}}{a} \\
&= \frac{\mathbf{Pr}[X > \max\{x, F^{-1}(1-a)\}]}{a} \\
&= \mathbf{Pr}\left[X > x|X > F^{-1}(1-a)\right],
\end{aligned}$$

as claimed. □

Now, let q_i be the probability that $i = \arg\max_j V_j$. Note that $\sum_{i=1}^n q_i = 1$. Note also that $q_i = \mathbf{Pr}[V_i > \max_{j \neq i} V_j]$ by definition (recall that we are assuming continuous distributions), and therefore

$$\mathbf{E}\left[V_i|V_i > \max_{j \neq i} V_j\right] \leq \mathbf{E}\left[V_i|V_i > F^{-1}(1-q_i)\right].$$

From this it follows that the prophet's reward, $\textsc{Opt}(\boldsymbol{F}) = \mathbf{E}[\max_i V_i] = \sum_{i=1}^n \mathbf{E}[V_i|V_i > \max_{j \neq i} V_j] q_i$ can be upper bounded by the following optimization problem:

$$\begin{aligned}
\text{[Ex-Ante]} \quad \text{maximize} \; & \sum_{i=1}^n \mathbf{E}\left[V_i|V_i > F^{-1}(1-x_i)\right] x_i \qquad (2.3) \\
\text{subject to} \; & \sum_{i=1}^n x_i = 1
\end{aligned}$$

$$x_i \geq 0$$

This optimization problem is called the *ex-ante relaxation*, because we replace the ex-post constraint of taking only one reward with an ex-ante constraint on the expected number of rewards taken.

We now derive a lower bound between the performance of the gambler and the ex-ante relaxation, which immediately implies a prophet inequality.

Let x^* be the optimal solution to [Ex-Ante], and let Ex-Ante(x^*) denote the objective value attained by the optimal solution. Consider the deterministic thresholds algorithm, $\textsc{Alg}$, obtained by setting $\tau_i = F^{-1}(1 - x_i^*/2))$; namely, the gambler stops the first time i in which $V_i \geq \tau_i$.

Note that for every i, we have $\mathbf{Pr}[V_i \geq \tau_i] = x_i^*/2$. Therefore, by union bound, we also have

$$\mathbf{Pr}\left[\exists j \neq i : V_j \geq \tau_j\right] \leq \sum_{j=1, j\neq i}^{n} \mathbf{Pr}\left[V_j \geq \tau_j\right] \leq \sum_{j=1, j\neq i}^{n} x_i^*/2 \leq \frac{1}{2}.$$

With this bound we can lower bound $\textsc{Alg}(\boldsymbol{F})$ as follows.

$$\begin{aligned}
\textsc{Alg}(\boldsymbol{F}) &= \sum_{i=1}^{n} \mathbf{E}\left[V_i \mathbb{1}(V_i \geq \tau_i)\mathbb{1}(\forall j < i : V_j < \tau_j)\right] \\
&\geq \sum_{i=1}^{n} \mathbf{E}\left[V_i \mathbb{1}(V_i \geq \tau_i)\mathbb{1}(\forall j \neq i : V_j < \tau_j)\right] \\
&= \sum_{i=1}^{n} \mathbf{E}\left[V_i | V_i \geq \tau_i\right] \frac{x_i^*}{2} \mathbf{Pr}\left[\forall j \neq i : V_j \geq \tau_j\right] \\
&= \sum_{i=1}^{n} \mathbf{E}\left[V_i | V_i \geq \tau_i\right] \frac{x_i^*}{2} (1 - \mathbf{Pr}\left[\exists j \neq i : V_j \geq \tau_j\right]) \\
&\geq \frac{1}{4} \sum_{i=1}^{n} \mathbf{E}\left[V_i | V_i \geq \tau_i\right] x_i^* \\
&\geq \frac{1}{4} \sum_{i=1}^{n} \mathbf{E}\left[V_i | V_i \geq F_i^{-1}(1 - x_i^*)\right] x_i^* \\
&\geq \frac{1}{4} \cdot \text{Ex-Ante}(x^*) \geq \frac{1}{4} \cdot \textsc{Opt}(\boldsymbol{F}).
\end{aligned}$$

While the previous proof provides a bound against an upper bound on $\textsc{Opt}(\boldsymbol{F})$, namely the ex-ante relaxation, the actual constant that we proved is only $1/4$ rather than the stronger $1/2$ we proved earlier. In what follows, we show that the stronger bound can also be achieved when compared to the ex-ante relaxation, but a more careful definition of the thresholds is needed.

For this, observe that the previous argument works just fine so long as we can find thresholds τ_i satisfying $\mathbf{Pr}[V_j < \tau_j \forall j < i, V_i \geq \tau_i] = x_i^*/2$ for all $i = 1, \dots, n$. Indeed, if this is the case

$$\begin{aligned}
\textsc{Alg}(\boldsymbol{F}) &= \sum_{i=1}^{n} \mathbf{E}\left[V_i \mathbb{1}(V_i \geq \tau_i)\mathbb{1}(\forall j < i : V_j < \tau_j)\right] \\
&= \sum_{i=1}^{n} \mathbf{E}\left[V_i | V_i \geq \tau_i\right] \mathbf{Pr}\left[\forall j < i : V_j < \tau_j, V_i \geq \tau_i\right] \\
&= \sum_{i=1}^{n} \mathbf{E}\left[V_i | V_i \geq \tau_i\right] \frac{x_i^*}{2} \\
&\geq \frac{1}{2} \sum_{i=1}^{n} \mathbf{E}\left[V_i | V_i \geq \tau_i\right] x_i^* \\
&\geq \frac{1}{2} \sum_{i=1}^{n} \mathbf{E}\left[V_i | V_i \geq F_i^{-1}(1 - x_i^*/2)\right] x_i^* \\
&\geq \frac{1}{2} \sum_{i=1}^{n} \mathbf{E}\left[V_i | V_i \geq F_i^{-1}(1 - x_i^*)\right] x_i^* \\
&\geq \frac{1}{2} \cdot \text{Ex-Ante}(x^*) \geq \frac{1}{2} \cdot \textsc{Opt}(\boldsymbol{F}).
\end{aligned}$$

All that remains to argue is that there exist thresholds $\tau_1, \dots, \tau_n$ that satisfy

$$\mathbf{Pr}\left[V_1 < \tau_1, \dots, V_{i-1} < \tau_{i-1}, V_i \geq \tau_i\right] = \frac{1}{2} x_i^*, \text{ for all } i = 1, \dots, n.$$

To this end, consider $\tilde{x}_i = \mathbf{Pr}[V_i \geq \tau_i]$ and define $\tau_1, \dots, \tau_n$ in this order recursively by setting

$$\tau_i = F_i^{-1}\left(1 - \frac{x_i^*}{2\prod_{j<i}(1 - \tilde{x}_j)}\right).$$

By this definition and applying F_i to the previous equation, we obtain the equations

$$\tilde{x}_i = \frac{x_i^*}{2\prod_{j<i}(1 - \tilde{x}_j)} \qquad \text{for all } i = 1, \dots, n.$$

Note that for the τ_i's to be well defined, all we need is that the solution to the previous equations $\tilde{x}$, satisfies $\tilde{x}_i \in [0, 1]$ for all $i = 1, \dots, n$. To see this, observe that for all i it holds that

$$\prod_{j<i}(1 - \tilde{x}_j)\tilde{x}_i = \frac{x_i^*}{2},$$

and therefore $\prod_{j<i+1}(1-\tilde{x}_j) = \prod_{j<i}(1-\tilde{x}_j) - x_i^*/2$. By repeatedly applying this identity to $\prod_{j<i}(1-\tilde{x}_j)$, we obtain

$$\prod_{j<i}(1-\tilde{x}_j) = 1 - \sum_{j<i} \frac{x_j^*}{2}.$$

Because of the constraint $\sum_j x_j^* \le 1$, we obtain $1 \ge \prod_{j<i}(1-\tilde{x}_j) \ge \frac{1}{2}$, so

$$\tilde{x}_i = \frac{x_i^*}{2\prod_{j<i}(1-\tilde{x}_j)} \in \left[\frac{x_i^*}{2}, x_i^*\right] \subseteq [0,1],$$

as needed.

2.5 Proof 4: Induction

We finally present an early proof of Theorem 1.2 based on a completely different idea. Consider an instance $\boldsymbol{V} = (V_1, \ldots, V_n)$. The proof first establishes that the ratio $\textsc{Opt}^{\textsc{on}}(\boldsymbol{V})/\textsc{Opt}(\boldsymbol{V})$ decreases if we replace the random random variable variable V_1 with a deterministic random variable λ. Secondly, we establish that the ratio decreases again if we replace the last two random variables V_{n-1}, V_n by a single random variable that is a *long shot*, i.e., it is very large with very small probability and zero otherwise. This last step allows us to use induction to reduce the problem of lower bounding $\textsc{Opt}^{\textsc{on}}(\boldsymbol{V})/\textsc{Opt}(\boldsymbol{V})$ to a problem with just two random variables.[2] To finish the proof, we observe that

$$\frac{V(V_1,V_2)}{\mathbf{E}[\max\{V_1,V_2\}]} \ge \frac{\max\{\mathbf{E}[V_1],\mathbf{E}[V_2]\}}{\mathbf{E}[V_1]+\mathbf{E}[V_2]} \ge \frac{1}{2}.$$

Let us now establish the two steps just described. For the first part define $\lambda = V(V_2, \ldots, V_n)$, the expected value the gambler gets on the last $n-1$ random variables. Note that for any set of 3 values a, b, c we have that

$$\max\{a,b\} \le \max\{c,b\} + |a-c|_+.$$

This can be easily seen by conditioning on which of a, b, c is largest. We can thus apply the inequality to $a = V_1$, $b = \max_{2\le i\le n} V_i$, and $c = \lambda$, and take expectation to obtain:

$$\mathbf{E}\left[\max_{i\in[n]} V_i\right] \le \mathbf{E}\left[\max\{\lambda, \max_{2\le i\le n} V_i\}\right] + \mathbf{E}\left[|V_1-\lambda|_+\right].$$

[2] Moreover, it tells us that the first value is deterministic and the second a long shot, as we already know what the worst case looks like.

On the other hand, by the backward induction of Section 1.2 we have that

$$\begin{aligned}\textsc{Opt}^{\text{ON}}(V_1, V_2, \ldots, V_n) &= \textsc{Opt}^{\text{ON}}(V_2, \ldots, V_n) + \mathbf{E}\left[|V_1 - \lambda|_+\right] \\ &= \textsc{Opt}^{\text{ON}}(\lambda, V_2, \ldots, V_n) + \mathbf{E}\left[|V_1 - \lambda|_+\right].\end{aligned}$$

Therefore,

$$\begin{aligned}\frac{\textsc{Opt}^{\text{ON}}(V_1, V_2, \ldots, V_n)}{\mathbf{E}[\max_{i\in[n]} V_i]} &\geq \frac{\textsc{Opt}^{\text{ON}}(\lambda, V_2, \ldots, V_n) + \mathbf{E}[|V_1 - \lambda|_+]}{\mathbf{E}[\max\{\lambda, \max_{2\leq i\leq n} V_i\}] + \mathbf{E}[|V_1 - \lambda|_+]} \\ &\geq \frac{\textsc{Opt}^{\text{ON}}(\lambda, V_2, \ldots, V_n)}{\mathbf{E}[\max\{\lambda, \max_{2\leq i\leq n} V_i\}]}.\end{aligned}$$

We conclude that instance $(\lambda, V_2, \ldots, V_n)$ is harder for the prophet than instance $(V_1, \ldots, V_n)$.

The second step of the proof consists of modifying the instance once more to make the gambler's life yet harder. As described above, this time, we will replace the last two random variables V_{n-1} and V_n by a single random variable L that we term a long shot. So let L be the random variable taking the value $\textsc{Opt}^{\text{ON}}(V_{n-1}, V_n)/p$ with probability p and the value 0 otherwise.

Clearly, $\textsc{Opt}^{\text{ON}}(\lambda, V_2, \ldots, V_{n-2}, L) = \textsc{Opt}^{\text{ON}}(\lambda, V_2, \ldots, V_n)$. On the other hand, $\mathbf{E}[\max(\lambda, V_2, \ldots, V_{n-2}, L)] \to \mathbf{E}[\max(\lambda, V_2, \ldots, V_{n-2})] + \mathbf{E}[L]$ as $p \to 0$. Recalling that $\mathbf{E}[L] = V(V_{n-1}, V_n)$, we obtain that

$$\begin{aligned}\mathbf{E}\left[L\right] &= \mathbf{E}\left[V_n\right] + \mathbf{E}\left[|V_{n-1} - \mathbf{E}(V_n)|_+\right] \\ &\geq \mathbf{E}\left[V_n\right] + \mathbf{E}\left[|V_{n-1} - \lambda|_+\right] \\ &\geq \mathbf{E}\left[V_n\right] + \mathbf{E}\left[|V_{n-1} - \max(\lambda, V_2, \ldots, V_{n-2}|_+\right].\end{aligned}$$

Therefore,

$$\begin{aligned}\mathbf{E}\left[\max(\lambda, V_2, \ldots, V_{n-2}, L)\right] &\xrightarrow[p\to 0]{} \mathbf{E}\left[\max(\lambda, V_2, \ldots, V_{n-2})\right] + \mathbf{E}\left[L\right] \\ &\geq \mathbf{E}\left[\max(\lambda, V_2, \ldots, V_{n-1})\right] + \mathbf{E}\left[V_n\right] \\ &\geq \mathbf{E}\left[\max(\lambda, V_2, \ldots, V_n)\right].\end{aligned}$$

We thus conclude that for p small enough

$$\frac{\textsc{Opt}^{\text{ON}}(\lambda, V_2, \ldots, V_n)}{\mathbf{E}[\max\{\lambda, \max_{2\leq i\leq n} V_i\}]} \geq \frac{\textsc{Opt}^{\text{ON}}(\lambda, V_2, \ldots, V_{n-2}, L)}{\mathbf{E}[\max(\lambda, V_2, \ldots, V_{n-2}, L)]},$$

as needed.

2.6 Discussion/Outlook

As mentioned at the outset of this chapter, the main purpose of going different proofs for the same result is that these proof techniques often can be generalized to more complex settings. Indeed, we will be returning to these basic proof patterns throughout this book.

As an example, consider the first proof covered in this chapter. The key quantities, *baseline* and *bonus*, that arise in this proof can be economically, in terms of *revenue* and *utility*. To see this, consider a posted-price mechanism for selling a single good with fixed price τ. Suppose that the random variables $V_1, \ldots, V_n$ correspond to the values of buyers arriving one-by-one, and that buyer i buys the good if it's still available and if it's value v_i is at least the price τ. The revenue of this mechanism is τ if at least one buyer's value v_i exceeds τ, while the utility of the buyer i that purchases the good is $v_i - \tau$. Since the mean-based rule equalizes the two contributions, the revenue and utility part, it is also referred to as the *balanced prices* approach. It turns out that this idea is much more general, and leads to relatively simple proofs in more general settings.

Similarly, the third proof in this chapter, based on relax-and-round may be very natural to a someone with an algorithms background. Indeed, suitable generalizations and formalizations of this approach, known as *online contention resolution* schemes, have driven a lot of progress in prophet inequalities over the last two decades.

Take-Aways

- There are many different techniques that have been developed for proving the classic prophet inequality.
- Often the discovery of new proof techniques for the basic setting has enabled progress in more complex settings (e.g., combinatorial domains).
- Handling discontinuous distributions is a common subtlety in prophet inequality proofs, typically handled by careful randomization of decisions.

Chapter Notes

The first proof presented in this chapter is due to Samuel-Cahn (1984). Indeed, she noted that a simple single threshold strategy was enough to achieve the optimal competitive ratio. Samuel-Cahn's proof coincides with our first proof, with the threshold τ chosen to be the median of the distribution of $\max_{i \in [n]} V_i$ (i.e., setting $q = 1/2$). To the best of our knowledge, the first proof that choosing $\tau = 1/2 \cdot \textsc{Opt}(\boldsymbol{F})$ also gives a $1/2$ prophet inequality is due to (Wittmann, 1995, Theorem 8). This was also independently observed in Kleinberg and Weinberg (2019).

The balanced prices idea is very powerful, and we will use it extensively throughout the book. The key articles that developed this proof approach into a general framework include (Feldman et al., 2015; Kleinberg and Weinberg, 2019; Dütting et al., 2020). We will return to these studies and the general approach in Chapters 7, 9, and 11.

The second proof based on approximate stochastic dominance is due to Kleinberg and Weinberg (2019). This proof technique is also very general and has been adopted to numerous settings. It plays a particular important role in the proof of the optimal prophet inequality for the case of identical distributions (see Chapter 3) and a variant of the classic prophet inequality problem known as the prophet secretary problem (see Chapter 4).

The third proof, which takes a relax-and-round approach, is known as online contention resolution. It is developed in (Chekuri et al., 2014; Feldman et al., 2021; Lee and Singla, 2018). The ex-ante relaxation as a useful benchmark was introduced in (Chawla et al., 2007) and further popularized in (Yan, 2011; Chawla et al., 2010; Alaei, 2014). Similar to the balanced prices approach, the online contention resolution approach is very general and powerful and we return to it throughout this book (see Chapters 6 to 8 and 12).

The fourth proof based on induction and successive simplification of the instance is due to Hill and Kertz (1981). While we are not aware of other studies that follow exactly this proof approach, the general proof strategy of identifying hard instances has been successfully applied in many studies. See, for example, Liu et al. (2021) and Brustle et al. (2024).

There are, of course, additional proof techniques that we did not cover in this chapter; we will see a particularly elegant approach based on samples in Chapter 5.

References

Alaei, Saeed. 2014. Bayesian Combinatorial Auctions: Expanding Single Buyer Mechanisms to Many Buyers. *SIAM J. Comput.*, **43**(2), 930–972.

Brustle, Johannes, Correa, José, Dütting, Paul, and Verdugo, Victor. 2024. The Competition Complexity of Dynamic Pricing. *Math. Oper. Res.*, **49**(3), 1986–2008. An extended abstract appeared in EC 2022.

Chawla, Shuchi, Hartline, Jason D., and Kleinberg, Robert D. 2007. Algorithmic pricing via virtual valuations. Pages 243–251 of: *EC 2007*. ACM.

Chawla, Shuchi, Hartline, Jason D., Malec, David L., and Sivan, Balasubramanian.

2010. Multi-parameter mechanism design and sequential posted pricing. Pages 311–320 of: *STOC 2010*. ACM.

Chekuri, Chandra, Vondrák, Jan, and Zenklusen, Rico. 2014. Submodular Function Maximization via the Multilinear Relaxation and Contention Resolution Schemes. *SIAM J. Comput.*, **43**(6), 1831–1879.

Dütting, Paul, Feldman, Michal, Kesselheim, Thomas, and Lucier, Brendan. 2020. Prophet Inequalities Made Easy: Stochastic Optimization by Pricing Nonstochastic Inputs. *SIAM J. Comput.*, **49**(3), 540–582. An extended abstract appeared in FOCS 2017.

Feldman, Michal, Gravin, Nick, and Lucier, Brendan. 2015. Combinatorial Auctions via Posted Prices. Pages 123–135 of: *SODA 2015*. SIAM.

Feldman, Moran, Svensson, Ola, and Zenklusen, Rico. 2021. Online Contention Resolution Schemes with Applications to Bayesian Selection Problems. *SIAM J. Comput.*, **50**(2), 255–300. An extended abstract appeared with title "Online Contention Resolution Schemes"' in SODA 2016.

Hill, Theodore P., and Kertz, Robert P. 1981. Ratio Comparisons of Supremum and Stop Rule Expectations. *Z. Wahrscheinlichkeitstheor. Verw. Geb.*, **56**, 283–285.

Kleinberg, Robert, and Weinberg, S. Matthew. 2019. Matroid prophet inequalities and applications to multi-dimensional mechanism design. *Games Econ. Behav.*, **113**, 97–115. An extended abstract with title "Matroid Prophet Inequalities" appeared in STOC 2012.

Lee, Euiwoong, and Singla, Sahil. 2018. Optimal Online Contention Resolution Schemes via Ex-Ante Prophet Inequalities. Pages 57:1–57:14 of: *ESA 2018*. LIPIcs, vol. 112. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Liu, Allen, Leme, Renato Paes, Pál, Martin, Schneider, Jon, and Sivan, Balasubramanian. 2021. Variable Decomposition for Prophet Inequalities and Optimal Ordering. Page 692 of: *EC 2021*. ACM.

Samuel-Cahn, Ester. 1984. Comparison of Threshold Stop Rules and Maximum for Independent Nonnegative Random Variables. *Ann. Probab.*, **12**(4), 1213–1216.

Wittmann, Rainer. 1995. Prophet inequalities for dependent random variables. *Stoch. Rep.*, **52**(3–4), 283–293.

Yan, Qiqi. 2011. Mechanism Design via Correlation Gap. Pages 710–719 of: *SODA 2011*. SIAM.

3
Identical Distributions

A natural question arising from Theorem 1.2 is whether the constant $1/2$ can be improved when we restrict the distributions to be identical. In this chapter, we tackle this problem and show a substantially improved prophet inequality in the i.i.d. case. This is not simply a matter of tightening the arguments from the previous chapter to the i.i.d. setting: new ideas and methods are needed to obtain this improvement.

Indeed, we start by showing that the factor $1/2$ cannot be improved using deterministic single-threshold algorithms, but if we allow randomization when there is a tie, that is, if we consider general single-threshold algorithms, then the gambler can obtain a reward which is within a factor of $1 - 1/e \approx 0.63$ of that of the prophet. Then, by considering general threshold algorithms, we further improve this result and obtain the best possible prophet inequality for i.i.d. valuations whose approximation factor is ≈ 0.745.

Throughout this chapter, we consider n (non-negative) random variables $V_1, \ldots, V_n$, that are independent and *identically* distributed according to F. As usual, the random variables are realized sequentially so that at time i, we observe the realization v_i of $V_i \sim F$. We write $\boldsymbol{V} = (V_1, \ldots, V_n)$ and $\boldsymbol{F} = F^n$, so that $\boldsymbol{V} \sim \boldsymbol{F}$.

Approximate Stochastic Dominance. The following notion of approximate stochastic dominance will be a main tool throughout this chapter. Consider two non-negative random variables $X \sim F$ and $Y \sim G$. We say that X *α-stochastically dominates* Y if and only if, for all $z \geq 0$,[1]

$$\mathbf{Pr}\left[X \geq z\right] \geq \alpha \cdot \mathbf{Pr}\left[Y \geq z\right].$$

Lemma 3.1 *If X α-stochastically dominates Y then* $\mathbf{E}[X] \geq \alpha\, \mathbf{E}[Y]$.

[1] Note that for $\alpha = 1$ this coincides with the standard definition of first order stochastic dominance.

Proof The result follows immediately by integration. Indeed,

$$\mathbf{E}\left[X\right] = \int_0^\infty (1 - F(t))dt \geq \alpha \cdot \int_0^\infty (1 - G(t))dt = \alpha\,\mathbf{E}\left[Y\right]$$

as claimed. □

3.1 Necessity of Randomization

We start by examining deterministic single-threshold algorithms, showing that no such algorithm can improve upon the factor $1/2$ even in the i.i.d. setting. To this end, consider the instance $\boldsymbol{V} = (V_1, \ldots, V_n)$ in which each V_i is distributed as follows:

$$\mathbf{Pr}\left[V_i = 1\right] = 1 - \frac{1}{n^2} \qquad \text{and} \qquad \mathbf{Pr}\left[V_i = n\right] = \frac{1}{n^2}.$$

Note first that the expected reward of the prophet $\textsc{Opt}(\boldsymbol{F})$ is given by

$$\begin{aligned}
\textsc{Opt}(\boldsymbol{F}) &= \mathbf{E}\left[\max_{i \in [n]} V_i\right] \\
&= \left(1 - \frac{1}{n^2}\right)^n + \left(1 - \left(1 - \frac{1}{n^2}\right)^n\right) n \\
&= 1 + \left(1 - \left(1 - \frac{1}{n^2}\right)^n\right)(n-1) \\
&= 1 + \left(1 - \left[\left(1 - \frac{1}{n^2}\right)^{n^2}\right]^{1/n}\right)(n-1) \\
&\geq 1 + \left(1 - e^{-1/n}\right)(n-1) \\
&\geq 1 + \frac{n-1}{n+1} = 2 - \frac{2}{n+1}.
\end{aligned}$$

Here, for the first inequality, we use that the sequence $(1 - 1/k)^k$ monotonically converges to $1/e$ from below. The second inequality holds because $e^{1/n} \geq 1 + 1/n$ and thus $e^{-1/n} \leq n/(n+1)$.

Let us now consider a deterministic single-threshold algorithm given by a threshold τ. Since each V_i is supported on $\{1, n\}$, we can assume without loss of generality that the deterministic single-threshold algorithm is inclusive (since any exclusive single-threshold algorithm for this problem instance can be equivalently represented as an inclusive algorithm with slightly lower threshold). We will show that no such τ can guarantee the gambler an expected reward better than $1 + 1/n$.

Indeed, setting $\tau \leq 1$ guarantees that V_1 is accepted, so we get value $\mathbf{E}[V_1] = 1 - 1/n^2 + n/n^2 \leq 1 + 1/n$. Alternatively, if we set $n \geq \tau > 1$, we never accept a value of 1 and obtain reward n if and only if it appears in the sequence. This happens with probability

$$1 - \left(1 - \frac{1}{n^2}\right)^n = 1 - \left[\left(1 - \frac{1}{n^2}\right)^{n^2}\right]^{1/n} \leq 1 - \left[\left(1 - \frac{1}{n}\right)^n\right]^{1/n} \leq \frac{1}{n}.$$

Thus, the reward we obtain is at most $n \cdot 1/n = 1$. Finally, if $\tau > n$, we never accept and we obtain a value of 0.

So in all cases, our expected reward, $\textsc{Alg}(\boldsymbol{F})$, is no more than $1 + 1/n$ while $\textsc{Opt}(\boldsymbol{F}) \geq 2 - 2/(n+1)$. Therefore

$$\frac{\textsc{Alg}(\boldsymbol{F})}{\textsc{Opt}(\boldsymbol{F})} \leq \frac{1 + 1/n}{2 - 2/(n+1)} \longrightarrow 1/2.$$

The previous instance gives us little hope for the possibility of improving the factor $1/2$ in the i.i.d. setting using deterministic single-threshold algorithms. However, quite surprisingly, a small algorithmic twist does the job and allows us to surpass this barrier. The additional feature we will use is that whenever there is a tie between the chosen threshold and the value, we will allow the stopping decision to be randomized.

To illustrate this, let us go back to the example, but now consider the following *randomized* single-threshold algorithm: it uses threshold $\tau = 1$, and whenever the observed value is precisely equal to 1 the algorithm accepts it with probability $1/n$. In this case, upon observing V_i, we stop with probability $1/n^2 + (1 - 1/n^2) \cdot 1/n = 1/n + 1/n^2 - 1/n^3$. Therefore, the overall probability of stopping, and thus obtaining some reward, is $1 - (1 - 1/n - 1/n^2 + 1/n^3)^n \geq 1 - (1 - 1/n)^n$. Furthermore, conditional on stopping, we obtain reward n with probability

$$\frac{1/n^2}{1/n + 1/n^2 - 1/n^3} = \frac{1/n}{1 + 1/n - 1/n^2} = \frac{n}{n^2 + n - 1} \geq \frac{1}{n+1},$$

and value 1 otherwise. Overall, the expected reward conditional on stopping is at least $2n/(n+1)$. We conclude that our total expected reward is at least

$$(1 - (1 - 1/n)^n) \cdot \frac{2n}{n+1} \longrightarrow 2(1 - 1/e) \approx 1.26.$$

Interestingly, this is substantially better than what we could obtain with a deterministic single-threshold algorithm. In the next section we prove that, in the general i.i.d. setting, there always exists a randomized single-threshold algorithm that guarantees a fraction $1 - 1/e$ of the prophet's reward.

3.2 A $(1 - 1/e)$-Approximation

In this section, we show that a *randomized* single-threshold algorithm achieves a $(1-1/e)$-approximation to the prophet, and that this is best possible. We start by showing that such an algorithm exists.

Theorem 3.1 *Let $\boldsymbol{V} = (V_1, \dots, V_n)$ be independent, identically distributed, non-negative random variables, distributed according to F. Then there exists a (randomized) single-threshold algorithm such that*

$$\textsc{Alg}(\boldsymbol{F}) \geq \left(1 - \frac{1}{e}\right) \textsc{Opt}(\boldsymbol{F}) \ .$$

In light of Lemma 2.1, it suffices to show the following lemma, which establishes the existence of a fixed threshold algorithm for the case where F is a *continuous* distribution.

Lemma 3.2 *Let $\boldsymbol{V} = (V_1, \dots, V_n)$ be independent, identically distributed, non-negative random variables, distributed according to continuous distribution F. Let τ be a threshold such that $F(\tau) = 1 - 1/n$ and consider the algorithm, $\textsc{Alg}$, that accepts the first reward of value τ or more. Then $\textsc{Alg}(\boldsymbol{V})$ $(1 - 1/e)$-stochastically dominates $\textsc{Opt}(\boldsymbol{V})$, and hence*

$$\textsc{Alg}(\boldsymbol{F}) \geq \left(1 - \frac{1}{e}\right) \textsc{Opt}(\boldsymbol{F}) \ .$$

Proof Let $\textsc{sel}$ be the index of the first i for which $V_i \geq \tau$. So that $V_{\textsc{sel}} = \textsc{Alg}(\boldsymbol{V})$, and then $\textsc{Alg}(\boldsymbol{F}) = \mathbf{E}[V_{\textsc{sel}}]$.

Because $1 - (1 - 1/n)^n \geq 1 - 1/e$ for all n, it is sufficient to show that $\boldsymbol{V}_{\textsc{sel}}$ α-stochastically dominates $\textsc{Opt}(\boldsymbol{V})$, for $\alpha = 1 - (1 - 1/n)^n$. This is showing that for, all $z \geq 0$,

$$\mathbf{Pr}\left[V_{\textsc{sel}} \geq z\right] \geq \left(1 - \left(1 - \frac{1}{n}\right)^n\right) \mathbf{Pr}\left[\max_i V_i \geq z\right].$$

We distinguish two cases. If $z \leq \tau$, then $V_{\textsc{sel}}$ automatically has value τ or more if any V_i exceeds the threshold. Therefore,

$$\mathbf{Pr}\left[V_{\textsc{sel}} \geq z\right] = \mathbf{Pr}\left[\exists i : V_i \geq \tau\right] = 1 - \left(1 - \frac{1}{n}\right)^n \ ,$$

and the inequality is direct.

Now we consider $z > \tau$. We can decompose the event $V_{\textsc{sel}} \geq z$ into disjoint events $V_1, \dots, V_{i-1} < \tau, V_i \geq z$. Because the V_i are independent and identically distributed, this is

$$\begin{aligned}
\mathbf{Pr}\left[V_{\text{SEL}} \geq z\right] &= \sum_{i=1}^{n} \mathbf{Pr}\left[V_1, \dots, V_{i-1} < \tau, V_i \geq z\right] \\
&= \sum_{i=1}^{n} \mathbf{Pr}\left[V_1 < \tau\right] \dots \mathbf{Pr}\left[V_{i-1} < \tau\right] \mathbf{Pr}\left[V_i \geq z\right] \\
&= \sum_{i=1}^{n} \left(1 - \frac{1}{n}\right)^{i-1} \mathbf{Pr}\left[V_1 \geq z\right] \\
&= \frac{1 - \left(1 - \frac{1}{n}\right)^{n}}{1 - \left(1 - \frac{1}{n}\right)} \mathbf{Pr}\left[V_1 \geq z\right] \\
&= \left(1 - \left(1 - \frac{1}{n}\right)^{n}\right) n \, \mathbf{Pr}\left[V_1 \geq z\right] \\
&= \left(1 - \left(1 - \frac{1}{n}\right)^{n}\right) \sum_{i=1}^{n} \mathbf{Pr}\left[V_i \geq z\right] \\
&\geq \left(1 - \left(1 - \frac{1}{n}\right)^{n}\right) \mathbf{Pr}\left[\max_{i=1,\dots,n} V_i \geq z\right] \quad .
\end{aligned}$$

Here the last inequality follows from what is known as the *union bound*. For this note that we can write the event that the maximum V_i is at least z as a union of disjoint events of the form $V_1 < z, \dots, V_{i-1} < z, V_i \geq z$ and thus;

$$\begin{aligned}
\mathbf{Pr}\left[\max_i V_i \geq z\right] &= \mathbf{Pr}\left[\bigcup_{i=1}^{n} V_1 < z, \dots, V_{i-1} < z, V_i \geq z\right] \\
&= \sum_{i=1}^{n} \mathbf{Pr}\left[V_1 < z, \dots, V_{i-1} < z, V_i \geq z\right] \\
&\leq \sum_{i=1}^{n} \mathbf{Pr}\left[V_i \geq z\right] \quad .
\end{aligned}$$

In combination, these two cases give the desired inequality as claimed. □

An alternative approach to obtain Theorem 3.1 is to argue in *quantile space*. So we take $V_1, \dots, V_n$ non-negative i.i.d. random variables, distributed according to F and refer to V as a random variable with the same common distribution. Let $F^{-1}(q) = \inf\{x \geq 0 \mid F(x) \geq q\}$ be the generalized inverse of F (or quantile function) and let $\tau(q) = F^{-1}(1-q)$.

Let $R(q) = \int_0^q F^{-1}(1-\theta) d\theta$, which equals the expected reward from a random variable that is accepted with probability q. Indeed, $R(q) = \mathbf{Pr}[V \geq \tau(q)]\, \mathbf{E}[V|V \geq \tau(q)] = q\tau(q) + \int_{\tau(q)}^{\infty} 1 - F(t)dt = \int_0^q F^{-1}(1-\theta)d\theta$

Now, we use integration by parts, then the change of variables $F(t) = 1 - q$, and then integration by parts again.

$$\begin{aligned}
\textsc{Opt}(\boldsymbol{F}) = \mathbf{E}(\max\{V_1, \ldots, V_n\}) &= \int_0^{\infty} 1 - F^n(t)dt \\
&= \int_0^{\infty} nF^{n-1}(t)F'(t)dt \\
&= n\int_0^1 (1-q)^{n-1}F^{-1}(1-q)dq \\
&= n\int_0^1 (n-1)(1-q)^{n-2}R(q)dq\,.
\end{aligned}$$

Noting that $(n-1)(1-q)^{n-2}$ is the PDF of a beta distribution with parameters 1 and $n-1$, $B(1, n-1)$, we note that $\textsc{Opt}(\boldsymbol{F}) = n\,\mathbf{E}_{q\sim B(1,n-1)}[R(q)]$. Observing that $R(q)$ is concave since it is the integral of an decreasing function we can apply Jensen's inequality to conclude that:

$$\textsc{Opt}(\boldsymbol{F}) \leq nR(\mathbf{E}_{q\sim B(1,n-1)}[q]) = nR(1/n).$$

Now take quantile $q = 1/n$ and consider the natural algorithm, $\textsc{Alg}$, that stops the first time it sees a value $V_i \geq \tau(q)$. Clearly,

$$\begin{aligned}
\mathbf{E}\left[\textsc{Alg}\right] &= \sum_{i=1}^{n}(1-q)^{i-1}R(q) = R(1/n)\sum_{i=0}^{n-1}(1-q)^i \\
&= R(1/n)\frac{1-(1-1/n)^n}{1/n} \geq (1-1/e)nR(1/n).
\end{aligned}$$

Optimality of $(1 - 1/e)$**.** We next show that no randomized single-threshold algorithm can achieve a better than $(1 - 1/e)$-approximation.

Theorem 3.2 *For every n, there exists a distribution F such that for every randomized single-threshold algorithm* $\textsc{Alg}_{\boldsymbol{\tau}}$ *facing the sequence of random variables* $V_1, \ldots, V_n$ *sampled independently from F, it holds that*

$$\textsc{Alg}(\boldsymbol{F}) \leq \left(1 - \frac{1}{e} + O\left(\frac{1}{n}\right)\right) \cdot \mathbf{E}[\max_{i\in[n]}\{V_i\}].$$

Proof Consider the following problem instance. Each V_i is equal to $n/(e-1)$ with probability $1/n^2$, and $(e-2)/(e-1)$ otherwise. Note that

$$\mathbf{E}\left[\max_{i\in[n]}\{V_i\}\right] = \left(1 - \frac{1}{n^2}\right)^n \cdot \frac{e-2}{e-1} + \left(1 - \left(1 - \frac{1}{n^2}\right)^n\right) \cdot \frac{n}{e-1}$$

$$= 1 - O\left(\frac{1}{n}\right).$$

Next we analyze the performance of an arbitrary randomized single-threshold algorithm with threshold τ, which accepts a value of exactly τ with probability p. Without loss of generality, we may assume that $\tau \in \{(e-2)/(e-1), n/(e-1)\}$ and that $p = 1$ if $\tau = n/(e-1)$.

In the case where $\tau = n/(e-1)$ and $p = 1$, the algorithm only accepts the high reward and

$$\textsc{Alg}(\boldsymbol{F}) = \left(1 - \left(1 - \frac{1}{n^2}\right)^n\right) \cdot \frac{n}{e-1} \leq \frac{1}{e-1} + O\left(\frac{1}{n}\right) \approx 0.58 + O\left(\frac{1}{n}\right).$$

For the remaining case where $\tau = (e-2)/(e-1)$, let $q = (1 - 1/n^2) \cdot (1 - p) \leq 1 - 1/n^2$ denote the probability of skipping any given random variable conditional on reaching it. We then have

$$\begin{aligned}
\textsc{Alg}(\boldsymbol{F}) &= \sum_{i=1}^{n} q^i \cdot \left(\left(1 - \frac{1}{n^2} - q\right) \cdot \frac{e-2}{e-1} + \frac{1}{n^2} \cdot \frac{n}{e-1}\right) \\
&= \frac{1 - q^n}{1 - q} \left(\left(1 - \frac{1}{n^2} - q\right) \cdot \frac{e-2}{e-1} + \frac{1}{n^2} \cdot \frac{n}{e-1}\right) \\
&\leq \frac{1 - q^n}{e - 1} \cdot \left((e-2) + \frac{1}{(1-q) \cdot n}\right), \qquad (3.1)
\end{aligned}$$

where in the final inequality we simply used $1 - 1/n^2 - q \leq 1 - q$ and rearranged.

To analyze the upper bound in Equation (3.1), let us substitute $y = (1-q) \cdot n$. Note that then $q^n = (1 - y/n)^n$. We can approximate $1 - (1 - y/n)^n$ with $1 - e^{-y} + O(1/n)$. We thus obtain

$$\textsc{Alg}(\boldsymbol{F}) \leq \frac{1 - e^{-y}}{e - 1} \cdot \left((e-2) + \frac{1}{y}\right) + O\left(\frac{1}{n}\right).$$

Maximizing $(1 - e^{-y})/(e-1) \cdot ((e-2) + 1/y)$ over y yields $1 - 1/e$ at $y = 1$. We conclude that $\textsc{Alg}(\boldsymbol{F}) \leq 1 - 1/e + O(1/n)$ as claimed. □

3.3 A 0.745-Approximation

Interestingly, for the i.i.d. prophet inequality, single-threshold algorithms, discussed in the last section, are suboptimal. Indeed, a more general algorithm that sets multiple thresholds can improve upon the factor $1 - 1/e$ and indeed provides the best possible approximation.

Theorem 3.3 *Let $\boldsymbol{V} = (V_1, \ldots, V_n)$ be independent, identically distributed, non-negative random variables, distributed according to F. Then there exists a threshold algorithm such that*

$$\textsc{Alg}(\boldsymbol{F}) \geq 0.745 \cdot \textsc{Opt}(\boldsymbol{F}) \ .$$

To show Theorem 3.3 we consider a *continuous* distribution F, and a sequence of thresholds $\tau_1, \ldots, \tau_n$. Recall that, by Lemma 2.1, any guarantee that we show this way translates to general F through appropriate randomization. Let q_i be the probability that we do not stop within the first i steps. We have $q_0 = 1$ and $q_i = F(\tau_i) \cdot q_{i-1}$. Note that since F is continuous, instead of defining $\tau_1, \ldots, \tau_n$, we can also define $q_1, \ldots, q_n$ arbitrarily as long as it is a non-increasing sequence.

Let V_{SEL} denote the (random) value that a threshold algorithm based on the chosen thresholds selects. Motivated by Lemma 3.1, our goal will be to find an appropriate sequence of thresholds $\tau_1, \ldots, \tau_n$, or equivalently, probabilities $q_1 \geq q_2 \geq \ldots \geq q_n$ such that V_{SEL} then α-stochastically dominates $\textsc{Opt}(\boldsymbol{V})$, for α as large as possible. That is, we will need to show that for all $z \geq 0$, we have

$$\mathbf{Pr}\left[V_{\text{SEL}} \geq z\right] \geq \alpha \cdot \mathbf{Pr}\left[\max_i V_i \geq z\right] . \tag{3.2}$$

The event $V_{\text{SEL}} \geq z$ can be decomposed into n disjoint events, namely that we stop on V_i and $V_i \geq z$ for $i = 1, \ldots, n$. For each of these events, we may not have stopped on $V_1, \ldots, V_{i-1}$ and furthermore we need $V_i \geq \tau_i$ to stop and $V_i \geq z$. The probability of this to happen is $q_{i-1}(1 - F(\max\{z, \tau_i\})$. Therefore, we can write

$$\mathbf{Pr}\left[V_{\text{SEL}} \geq z\right] = \sum_{i=1}^{n} q_{i-1}(1 - F(\max\{z, \tau_i\}).$$

The key challenge is to relate the right-hand side of this equation to $\mathbf{Pr}\left[\max_i V_i \geq z\right]$, ideally in a way that enables optimization over thresholds. To achieve this, we choose a non-increasing function $h\colon [0,1] \to [0,1]$ and set $q_i = h(i/n)$. We require that $h(0) = 1$ and $h(t) > 0$ for all $t \in [0,1]$, that h is differentiable, and that the implied function $r : [0,1] \to \mathbb{R}$ defined via $r(t) = -h'(t)/h(t)$ is non-decreasing.[2]

We will show that

$$\mathbf{Pr}\left[V_{\text{SEL}} \geq z\right] \geq \int_{t=0}^{1} h(t) \min\{r(t), -n \ln F(z)\} dt. \tag{3.3}$$

The intuition behind this lower bound is that this is exactly the limit we would

[2] One particular choice of h to keep in mind is $h(t) = (1 - 1/n)^{nt}$, which exactly corresponds to $F(\tau_i) = 1 - 1/n$ for all i as in Section 3.2. The limit as n grows large is $h(t) = e^{-t}$.

obtain under $k \longrightarrow \infty$ when replacing our n random draws from the distribution F by $n \cdot k$ draws from $F^{1/k}$—a modification that would leave the distribution of the maximum intact.

To show Equation (3.3), we establish the following lemma. The lemma implies the claimed inequality by taking the sum over all i.

Lemma 3.3 *For every i and any $x \geq 0$, it holds that*

$$q_{i-1}(1 - F(\max\{x, \tau_i\}) \geq \int_{t=\frac{i-1}{n}}^{\frac{i}{n}} h(t) \min\{r(t), -n \ln F(x)\} dt.$$

Proof We distinguish two cases. In the first case, $\tau_i \geq x$. In this case, by the definition of r,

$$\begin{aligned}
\int_{t=\frac{i-1}{n}}^{\frac{i}{n}} h(t) \min\{r(t), -n \ln F(x)\} &= \int_{t=\frac{i-1}{n}}^{\frac{i}{n}} \min\{-h'(t), -h(t) n \ln F(x)\} dt \\
&\leq \int_{t=\frac{i-1}{n}}^{\frac{i}{n}} -h'(t) dt \\
&= h\left(\frac{i-1}{n}\right) - h\left(\frac{i}{n}\right) \\
&= q_{i-1} - q_i \\
&= q_{i-1} - q_{i-1} F(\tau_i) \\
&= q_{i-1}(1 - F(\max\{x, \tau_i\})),
\end{aligned}$$

where the inequality holds because $\min\{-h'(t), -h(t) n \ln F(x)\} \leq -h'(t)$.

In the second case, $\tau_i < x$. In this case, first note that, by the definition of r, for $t \geq (i-1)/n$, we have

$$\int_{s=\frac{i-1}{n}}^{t} r(s) ds = \int_{s=\frac{i-1}{n}}^{t} -\frac{h'(s)}{h(s)} ds = -\ln(h(t)) + \ln\left(h\left(\frac{i-1}{n}\right)\right).$$

Exponentiating both sides gives:

$$\begin{aligned}
h(t) &= h\left(\frac{i-1}{n}\right) \exp\left(-\int_{s=\frac{i-1}{n}}^{t} r(s) ds\right) \\
&\leq h\left(\frac{i-1}{n}\right) \exp\left(-\int_{s=\frac{i-1}{n}}^{t} \min\{r(s), -n \ln F(x)\} ds\right),
\end{aligned} \tag{3.4}$$

where the inequality holds because $\exp(-z)$ is a decreasing function in z.

As a consequence, with $U(t) = \int_{s=(i-1)/n}^{t} \min\{r(s), -n \ln F(x)\}$ and thus

$U'(t) = \min\{r(t), -n \ln F(x)\}$, we obtain

$$\begin{aligned}
&\int_{t=\frac{i-1}{n}}^{\frac{i}{n}} h(t) \min\{r(t), -n \ln F(x)\} dt \\
&\qquad\qquad \leq h\left(\frac{i-1}{n}\right) \int_{t=\frac{i-1}{n}}^{\frac{i}{n}} \exp\left(-U(t)\right) U'(t) dt \\
&\qquad\qquad = h\left(\frac{i-1}{n}\right) \cdot \Big[-\exp(-u) \Big]_{u=0}^{U(\frac{i}{n})} \\
&\qquad\qquad = h\left(\frac{i-1}{n}\right) \left(1 - \exp\left(-\int_{t=\frac{i-1}{n}}^{\frac{i}{n}} \min\{r(t), -n \ln F(x)\} dt\right)\right) \\
&\qquad\qquad \leq h\left(\frac{i-1}{n}\right) \left(1 - \exp\left(-\int_{t=\frac{i-1}{n}}^{\frac{i}{n}} -n \ln F(x) dt\right)\right) \\
&\qquad\qquad = h\left(\frac{i-1}{n}\right) (1 - F(x)) \\
&\qquad\qquad = q_{i-1} \left(1 - F(\max\{x, \tau_i\})\right),
\end{aligned}$$

where the first inequality holds by Equation (3.4) and the definition of $U(t)$ and $U'(t)$, the first equality follows from integration by substitution with $u = U(t)$ so that $du = U'(t)dt$, and the second inequality holds because $1 - \exp(-z)$ is increasing in z. □

Note that by monotonicity of r, there is a value $s \in [0, 1]$ such that $r(t) \leq -n \ln F(z)$ for $t \leq s$ and $r(t) > -n \ln F(z)$ for $t > s$. Defining $g(z) = -n \ln F(z)$, this simplifies our bound for the probability of stopping on a value of at least z to

$$\mathbf{Pr}\left[V_{\text{SEL}} \geq z\right] \geq 1 - h(s) + \int_{t=s}^{1} h(t) g(z) dt.$$

This way also,

$$\mathbf{Pr}\left[\max_i V_i > z\right] = 1 - (F(z))^n = 1 - \exp(-g(z)).$$

So, in order to show the stochastic dominance relation in Equation (3.2), it suffices to show that, for all $z \geq 0$,

$$1 - h(s) + g(z) \int_{t=s}^{1} h(t) dt \geq \alpha(1 - \exp(-g(z))).$$

Equivalently, substituting $y(s) = 1/\alpha \cdot \int_{t=s}^{1} h(t) dt$, because $y'(s) = -1/\alpha \cdot h(s)$,

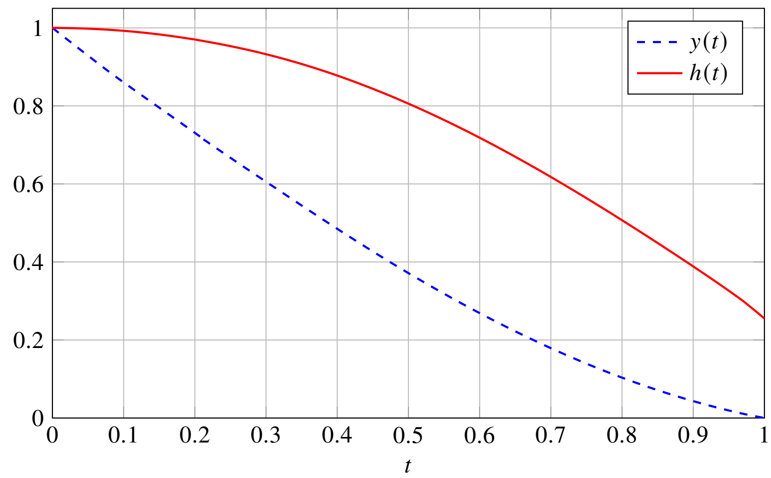

Figure 3.1 Plot of the functions y and h.

it suffices to show that, for all $z \geq 0$,

$$1 + \alpha y'(s) + g(z)\alpha y(s) - \alpha\,(1 - \exp(-g(z))) \geq 0. \tag{3.5}$$

So, it remains to find a function y such that this inequality is fulfilled for all z. To get an intuition regarding the optimal choice of such a function, consider the left-hand side of the inequality as a function of g. That is $\xi(g) = 1+\alpha y'(s)+g\alpha y(s)-\alpha\,(1-\exp(-g))$. Its derivative $\xi'(g) = \alpha y(s)-\alpha\exp(-g)$ has a unique zero at $g^\star = -\ln(y(s))$. Moreover, its second derivative $\xi''(g) = \alpha\exp(-g)$ is always positive. So the smallest value ξ attains is $\xi(g^\star) = 1 + \alpha y'(s) - \alpha y(s)\ln y(s) - \alpha + \alpha y(s)$. We would like this expression to be equal to zero for all values of s. This gives rise to the differential equation $y'(t) = y(t)\ln y(t) - y(t) + 1 - 1/\alpha$ for all t.

Since we require $h(0) = 1$, we must have $y'(0) = -1/\alpha \cdot h(0) = -1/\alpha$ and therefore, from the definition of the differential equation, $y(0) = 1$. It can be shown that, with this initial condition, the differential equation has a unique solution. This solution does not have a closed form expression, but we can define the function y (see Figure 3.1) via its inverse

$$y^{-1}(\tilde{y}) = -\int_{y=\tilde{y}}^{1} \frac{1}{y\ln y - y + 1 - \frac{1}{\alpha}}\,dy.$$

Note that $y^{-1}(1) = 0$ and therefore $y(0) = 1$ as needed. Additionally, we would like to have $y(1) = 0$ (from the definition of y via the integral of h), or equivalently, $y^{-1}(0) = 1$. This requires choosing $\alpha \approx 0.745$.

In order to show the claimed approximation guarantee, rather than solving the differential equation, we can also start from the definition of the inverse and choice of $\alpha \approx 0.745$, and conclude that this satisfies all conditions. The function y^{-1} defined this way is a strictly decreasing, differentiable function. So its inverse, the function y, is well-defined and fulfills these properties as well. We have $y(0) = 1$ and $y(1) = 0$ and so $y(t) \in [0, 1]$. Moreover, by the inverse function theorem, the derivative of y fulfills

$$y'(t) = \frac{1}{(y^{-1})'(y(t))} = y(t) \ln y(t) - y(t) + 1 - \frac{1}{\alpha}.$$

We claim that setting $h(t) = -\alpha y'(t)$ (see Figure 3.1) is a feasible choice for our prophet-inequality problem. To this end, we first verify that $h(0) = 1$ and additionally observe that $h(1) = 1 - \alpha > 0$. Both of these follow from $h(s) = -\alpha y'(s)$ and the definition of the differential equation, combined with $y(0) = 1$ and $y(1) = 0$, respectively. Additionally, since y satisfies the differential equation, we have

$$y''(t) = y'(t) \ln y(t) + y(t) \frac{y'(t)}{y(t)} - y'(t) = y'(t) \ln y(t).$$

This implies that h is differentiable, with derivative $h'(t) = -\alpha y''(t) = -\alpha y'(t) \ln y(t)$. Moreover, since y is decreasing and $y(t) \in [0, 1]$ for all $t \in [0, 1]$, we have $h'(t) = -\alpha y'(t) \ln y(t) \leq 0$ for all $t \in [0, 1]$, so h is a non-increasing function. Combining this with the fact that $h(0) = 1$ and $h(1) = 1 - \alpha > 0$ shows that $0 < h(t) \leq 1$ for all $0 \leq t \leq 1$.

Finally, $r(t) = -h'(t)/h(t) = -y''(t)/y'(t) = -\ln y(t)$ is an increasing function because y is decreasing.

So it only remains to show that Equation (3.5) is fulfilled. This is done in the following lemma.

Lemma 3.4 *The function y and the value α defined above fulfill*

$$1 + \alpha y'(s) + g\alpha y(s) - \alpha\,(1 - \exp(-g)) \geq 0$$

for all $g \geq 0$ and all $0 \leq s \leq 1$.

Proof Fix any value of s with $0 \leq s \leq 1$. Consider the function $\xi \colon \mathbb{R}_+ \to \mathbb{R}$ defined by $\xi(g) = 1 + \alpha y'(s) + g\alpha y(s) - \alpha\,(1 - \exp(-g))$. We would like to show that $\xi(g) \geq 0$ for all $g \in \mathbb{R}_+$.

Note that ξ is twice differentiable. Its derivative is $\xi'(g) = \alpha y(s) - \alpha \exp(-g)$, and its second derivative is $\xi''(g) = \alpha \exp(-g) > 0$. So, ξ is first decreasing up to $g^\star = -\ln y(s)$ and then increasing. For this reason, we only need to show

$\xi(g^\star) \geq 0$. By definition,

$$\xi(g^\star) = 1 + \alpha y'(s) - \alpha y(s) \ln y(s) - \alpha + \alpha y(s).$$

As y fulfills $y'(t) = y(t) \ln y(t) - y(t) + 1 - 1/\alpha$ for all t, we have $\xi(g^\star) = 0$. □

The analysis so far only shows that it is possible to obtain a prophet inequality with $\alpha \approx 0.745$. However, it can also be shown that the above analysis becomes tight as $n \to \infty$, showing that it is impossible to achieve a better than $\alpha \approx 0.745$ approximation via stochastic dominance. In fact, a careful recursive construction of hard instances shows that this factor is best possible in general.

Take-Aways

- For prophet inequalities with i.i.d. random variables, randomization is beneficial and required to beat the $1/2$ bound.
- The best (randomized) single-threshold algorithm achieves a $(1 - 1/e)$ approximation.
- The best multi-threshold algorithm achieves a guarantee of approximately 0.745, and this is best possible.
- Both these guarantees are shown through approximate stochastic dominance.
- For the 0.745 bound, tractability is achieved by lower-bounding with the limit behavior as $n \to \infty$.

Chapter Notes

The observation that randomization is necessary in the i.i.d. case to beat the $1/2$ bound is due to Samuel-Cahn (1984). Hill and Kertz (1982) provide a recursive characterization of the best possible α_n such that, if T_n is the set of stopping times of i.i.d. random variables $V_1, \ldots, V_n$, then

$$\sup\{\mathbf{E}\left[X_t\right] \;:\; t \in T_n\} \geq \alpha_n \cdot \mathbf{E}\left[\max\{V_1, \ldots, V_n\}\right].$$

They further showed that for all $n > 1$, it holds that $0.\overline{90} > \alpha_n > 0.625$; and conjectured that the sequence of α_n is monotonically increasing, with limit $e/(e+1) \approx 0.731$. Shortly afterwards, Samuel-Cahn (1984) reported that Kertz (1986) showed that $\alpha_n \to \alpha^\star \approx 0.745$ as $n \to \infty$ and conjectured that this is the best possible bound. More recent work by Saint-Mont (2002) provides a simpler proof of this result.

However, until recently, the best known positive result was that, for all $n > 1$, it holds that $\alpha_n \geq 1 - 1/e \approx 0.632$ (Kertz, 1986, Lemma 3.4). The first to show that this bound of $1 - 1/e$ can be attained with a single threshold were Ehsani et al. (2024). The bound of $1 - 1/e$ was broken only recently by Abolhassani et al. (2017), who showed that it is possible to achieve a $e/(e + 1) \approx 0.731$ approximation. In independent and parallel work, Correa et al. (2021) showed how to match the $\alpha^\star \approx 0.745$ bound, providing a tight resolution of the i.i.d. problem. More recent work has established this result through alternative techniques. For instance, Liu et al. (2021) use a variable decomposition technique, while Brustle et al. (2024) present a primal-dual argument based on linear programming.

The same tight bound of $\alpha^\star \approx 0.745$ established in Correa et al. (2021) was previously established by Allaart (2007) for a model, with Poisson arrivals over a fixed time horizon (and hence random n). Our proof for the i.i.d. case is similar to a proof for the Poisson arrival model presented in Singla (2018). Our proof bridges between the two models and results, by formally establishing that the large-n-limit implies a bound for all n (via Lemma 3.3), and exploiting that for large n the two models converge to each other.

References

Abolhassani, Melika, Ehsani, Soheil, Esfandiari, Hossein, Hajiaghayi, Mohammad-Taghi, Kleinberg, Robert D., and Lucier, Brendan. 2017. Beating 1-1/e for ordered prophets. Pages 61–71 of: *STOC*. ACM.

Allaart, Pieter C. 2007. Prophet inequalities for i.i.d. random variables with random arrival times. *Sequential Anal.*, **26**, 403–413.

Brustle, Johannes, Correa, José, Dütting, Paul, and Verdugo, Victor. 2024. The Competition Complexity of Dynamic Pricing. *Math. Oper. Res.*, **49**(3), 1986–2008. An extended abstract appeared in EC 2022.

Correa, José, Foncea, Patricio, Hoeksma, Ruben, Oosterwijk, Tim, and Vredeveld, Tjark. 2021. Posted Price Mechanisms and Optimal Threshold Strategies for Random Arrivals. *Math. Oper. Res.*, **46**(4), 1452–1478. An extended abstract appeared in EC 2017.

Ehsani, Soheil, Hajiaghayi, MohammadTaghi, Kesselheim, Thomas, and Singla, Sahil. 2024. Prophet Secretary for Combinatorial Auctions and Matroids. *SIAM J. Comput.*, **53**(6), 1641–1662. An extended abstract appeared in SODA 2018.

Hill, Theodore P., and Kertz, Robert P. 1982. Comparisons of stop rule and supremum expectations of i.i.d. random variables. *Ann. Probab.*, **10**(2), 336–345.

Kertz, Robert P. 1986. Stop rule and supremum expectations of i.i.d. random variables: A complete comparison by conjugate duality. *J. Multivariate Anal.*, **19**, 88–112.

Liu, Allen, Leme, Renato Paes, Pál, Martin, Schneider, Jon, and Sivan, Balasubramanian. 2021. Variable Decomposition for Prophet Inequalities and Optimal Ordering. Page 692 of: *EC 2021*. ACM.

Saint-Mont, Uwe. 2002. A simple derivation of a complicated prophet region. *J. Multivariate Anal.*, **80**(1), 67–72.

Samuel-Cahn, Ester. 1984. Comparison of Threshold Stop Rules and Maximum for Independent Nonnegative Random Variables. *Ann. Probab.*, **12**(4), 1213–1216.

Singla, Sahil. 2018. *Combinatorial Optimization Under Uncertainty: Probing and Stopping-Time Algorithms*. Doctoral Dissertation, School of Computer Science, Carnegie Mellon University.

4
Random and Free Order

In the previous chapter we explored ways to improve the competitive ratio of $1/2$ from Theorem 1.2 by restricting attention to identical distributions. Another natural direction is to instead reconsider the order in which the rewards are offered. Of course, arrival order is irrelevant when the reward distributions are identical. When the distributions are not identical, however, the arrival order can have a dramatic effect on the decision-maker's optimal policy and the expected value it obtains. Reconsider, for instance, Example 1.1. In that example, a deterministic reward of 1 is offered first, followed by a long-shot risky reward that is either 0 or $1/\epsilon$. This example is set up to make an online decision-maker's life difficult, since one must choose to take or forego the deterministic reward before the risky outcome is revealed. If the rewards were offered in the opposite order, however, then it is easy to achieve the best-in-hindsight reward: the decision-maker can simply accept the long shot whenever the risk pays off, and retains the option to take the safe deterministic reward if it does not. This motivates a line of work considering variations of the prophet inequality where rewards need not arrive in a worst-case order.

In this chapter we study the prophet inequality problem under two variations of how the arrival order is determined. First, we consider a version of the problem where rewards are assumed to arrive in a uniformly random order. As this is reminiscent of the famous secretary problem, this variant is often referred to as the Prophet Secretary problem. Next, we consider a version where the decision-maker can choose the arrival order, based on the rewards' distributions. This second variation is known as the Free-Order Prophet problem. Recall that in the original fixed-order problem, the optimal competitive ratio of $1/2$ is achievable with a simple threshold policy that accepts the first reward greater than half the expected maximum reward. Crucially, this choice of threshold is independent of the arrival order! Moreover, given this choice of threshold, the resulting $1/2$-approximation holds for any arrival order. This means in particular that a $1/2$-approximation is also achievable, using the same threshold, for both the Prophet Secretary and Free-Order Prophet problems. The main question we explore in this chapter is: can we do better?

4.1 Random Order

In the random-order model, there are n random variables V_i, where $V_i \sim F_i$ independently. The distributions F_i are known to the algorithm, and the random variables arrive in uniform random order. We use $\sigma_t \in [n]$ to denote the index of the t-th random variable to arrive.

Since the arrival order is stochastic, one question is whether (and when) the randomly-chosen order is revealed to the algorithm. One option is that the order is revealed up front, before the first reward arrives, and the algorithm can use this information to plan its selection decisions. A second, less-informative option is that the order is revealed online, with the index of each reward provided only as it arrives. Yet a third option is that the order is never explicitly revealed: rather, the decision-maker observes only the reward realizations but not their indices; the order can only be inferred from the rewards.

Since an algorithm can always disregard information and simulate being in a less-informative environment, it must be that for any problem instance the obtainable approximation factor is (weakly) best when the arrival order is revealed up front, and (weakly) worst when the arrival order is never revealed. Moreover, the distinction can matter: the following example shows that these approximation factors may be strictly different on some problem instances.

Example 4.1 There are three rewards, each drawn from a Bernoulli distribution. Reward V_1 is equal to $5/4$ with probability $1/5$; reward V_2 is equal to $1/2$ with probability 1; and reward V_3 is $1/\epsilon$ with probability ϵ. Each reward is otherwise 0.

To see that observing the arrival order in advance is strictly better than observing the arrival order online, consider the difference between arrival orders $[123]$ and $[132]$. In the former case, if reward 1 is not taken, then the expected continuation payoff is equal to 1 (obtained by always rejecting V_2 and accepting V_3). But in the latter case, the continuation payoff after rejecting V_1 is $3/2 - O(\epsilon)$, obtained by accepting the first non-zero reward. So if V_1 is positive, it is strictly optimal to accept it if the order is $[123]$, but strictly optimal to reject it if the order is $[132]$. The algorithm therefore obtains strictly higher expected reward if it knows in advance whether the order is $[123]$ or $[132]$ when choosing whether to accept V_1.

To see that observing the arrival order online is strictly better than not observing the arrival order at all, consider the difference between arrival orders $[123]$ and $[321]$. Suppose that the algorithm observes that the first realized reward is 0 (which it should always reject without loss of generality), and then the second realized reward is $1/2$. Note that if reward indices are revealed online then the algorithm can correctly infer the arrival order when choosing whether to accept this second reward. If indices are not revealed, however, then it is ambiguous to the algorithm whether the arrival order is $[123]$ or $[321]$. If the arrival order is $[321]$, then it is strictly optimal to accept the reward of $1/2$, since the expected value of V_1 is $1/4$. On the other hand, if the arrival order is $[123]$, then it is strictly optimal to reject the reward of $1/2$ and instead obtain the expected value of V_3, which is 1. The algorithm is therefore strictly better off if it knows whether the first reward of 0 was V_1 or V_3 before it must choose whether to accept V_2.

In all of the results in this chapter, all positive results (i.e., algorithms) apply in the least-informative setting where no information about the arrival order is revealed, and all negative results apply in the most-informative setting where the order is revealed to the algorithm up-front. Thus all upper and lower bounds on the worst-case competitive ratio proven in this chapter apply to all three options for information revelation.

4.1.1 Single-Threshold Algorithms

We first show that we can achieve a $(1 - 1/e)$-fraction of $\textsc{Opt}(\boldsymbol{F})$ with a single-threshold algorithm, where $\boldsymbol{F} = F_1 \times \ldots \times F_n$. Note that this generalizes Theorem 3.1. For ease of presentation, we will assume that the distributions F_i are continuous. This means, in particular, that we need not worry about the choice of tie-breaking behavior when V_i is equal to our chosen threshold.

Theorem 4.1 (Single threshold) *Consider the Prophet Secretary problem and the single-threshold algorithm* $\textsc{Alg}$ *with threshold* τ *so that* $\mathbf{Pr}[\max_i V_i \le \tau] = 1/e$. *Then,*

$$\textsc{Alg}(\boldsymbol{F}) \ge \left(1 - \frac{1}{e}\right) \cdot \textsc{Opt}(\boldsymbol{F}).$$

To show Theorem 4.1, we will establish a stochastic dominance relation, showing that for τ such that $\mathbf{Pr}[\max_i V_i \le \tau] = 1/e$ it holds that $\mathbf{Pr}[V_{\sigma_T} > t] \ge (1 - 1/e) \cdot \mathbf{Pr}[\max_i V_i \ge z]$. This is cast in the following lemma, which establishes a stochastic dominance relation between V_{σ_T} and $\max_i V_i$ for general τ. See Figure 4.1 for a visualization.

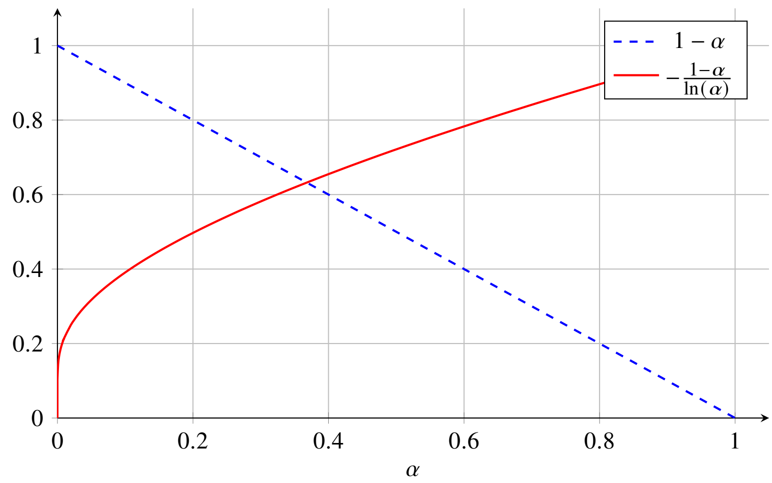

Figure 4.1 Plot of $1-\alpha$ and $-(1-\alpha)/\ln(\alpha)$.

Lemma 4.1 (Stochastic dominance) *Consider the single-threshold algorithm with threshold τ so that $\mathbf{Pr}[\max_i V_i \le \tau] = \alpha$. Then with T the stopping time of that algorithm, it holds that*

$$\mathbf{Pr}\left[V_{\sigma_T} > t\right] \ge \min\left\{1-\alpha, -\frac{1-\alpha}{\ln(\alpha)}\right\} \cdot \mathbf{Pr}\left[\max_i V_i > t\right].$$

To prove Lemma 4.1, we first show the following lemma, which relates the probability that $V_{\sigma_T} > t$ to the stopping time T of the single-threshold algorithm with threshold τ.

Lemma 4.2 (Connection to stopping time) *Consider the single-threshold algorithm with threshold τ and stopping time T. For all $t \ge 0$, it holds that*

$$\mathbf{Pr}\left[V_{\sigma_T} > t\right] \ge \gamma_1(t) \cdot \mathbf{Pr}\left[\max_i V_i > t\right],$$

where

$$\gamma_1(t) := \begin{cases} \mathbf{Pr}[T < \infty] & \text{for } t \le \tau, \text{ and} \\ \frac{1}{n}\sum_{k=1}^{n} \mathbf{Pr}[T > k] & \text{for } \tau < t. \end{cases}$$

To prove Lemma 4.2, we will use the following lemma.

Lemma 4.3 *Consider a threshold algorithm that uses the sequence of non-increasing thresholds $\tau_1 \ge \ldots \ge \tau_n$. Then, for the stopping time T of that* *algorithm, it holds that for all* $i, k \in [n]$*,*

$$\mathbf{Pr}\left[T \geq k \mid \sigma_k = i\right] \geq \mathbf{Pr}\left[T > k\right].$$

Proof We will show that the lemma holds pointwise, for any realization of the random variables. So fix some realizations of $V_1, \ldots, V_n$.

Fix $i, k \in [n]$. By conditioning on the arrival time of V_i (or more precisley, its realized value v_i), we have

$$\mathbf{Pr}\left[T > k\right] = \frac{1}{n} \sum_{\ell \in [n]} \mathbf{Pr}\left[T > k \mid \sigma_\ell = i\right].$$

Thus, in order to show the claim, it suffices to show that for any fixed ℓ, it holds that $\mathbf{Pr}[T > k \mid \sigma_\ell = i] \leq \mathbf{Pr}[T \geq k \mid \sigma_k = i]$.

To this end, consider some permutation ρ of $[n] \setminus \{i\}$. Construct σ^ℓ by inserting i in position ℓ, and σ^k by inserting i in position k. Let T^ℓ and T^k be the stopping times of the algorithm on σ^ℓ and σ^k, respectively.

First consider the case where $\ell > k$. We claim that in this case the event $\{T^\ell > k\}$ implies the event $\{T^k \geq k\}$. This is because, up to time $k-1$, both events deal with the same realizations of the same variables placed in the same order, implying that the executions of the algorithm on the two permutations σ^ℓ and σ^k are identical up to this point.

Next consider the case where $\ell \leq k$. We claim that also in this case, the event $\{T^\ell > k\}$ implies the event $\{T^k \geq k\}$. This is because the same realizations of the $k-1$ variables that are not V_i that are placed in the first k positions by σ^ℓ are placed in the first $k-1$ positions by σ^k. As a result these realizations are compared with thresholds in σ^k that can only be larger than those in σ^ℓ. Thus, if the algorithm does not stop on these realizations in σ^ℓ then it also does not stop on these realizations in σ^k.

The proof is completed by taking expectation over permutations ρ, and noting that the implied distributions over σ^ℓ and σ^k are the conditional distributions in question. □

We are now ready to prove Lemma 4.2.

Proof of Lemma 4.2 First consider the case where $t \leq \tau$. In this case, $V_{\sigma_T} > t$ whenever the algorithm stops, so

$$\mathbf{Pr}\left[V_{\sigma_T} > t\right] = \mathbf{Pr}\left[T < \infty\right] \geq \mathbf{Pr}\left[T < \infty\right] \cdot \mathbf{Pr}\left[\max_i V_i > t\right],$$

as claimed.

Next consider the case where $t \geq \tau$. In this case,

$$\mathbf{Pr}\left[V_{\sigma_T} > t\right]$$

$$
\begin{aligned}
&= \sum_{i \in [n]} \mathbf{Pr}\left[V_i > t, \sigma_T = i\right] && \text{(partition on } \sigma_T\text{)} \\
&= \sum_{i \in [n]} \sum_{k \in [n]} \mathbf{Pr}\left[V_i > t, \sigma_k = i, T = k\right] && \text{(partition on } T\text{)} \\
&= \sum_{i \in [n]} \sum_{k \in [n]} \mathbf{Pr}\left[V_i > t, \sigma_k = i, T \geq k\right] && (t \geq \tau) \\
&= \sum_{i \in [n]} \sum_{k \in [n]} \mathbf{Pr}\left[V_i > t, T \geq k \mid \sigma_k = i\right] \cdot \frac{1}{n} && \text{(conditioning)} \\
&= \sum_{i \in [n]} \sum_{k \in [n]} \mathbf{Pr}\left[V_i > t \mid \sigma_k = i\right] \cdot \mathbf{Pr}\left[T \geq k \mid \sigma_k = i\right] \cdot \frac{1}{n} && \text{(independence)} \\
&= \left[\frac{1}{n} \sum_{k \in [n]} \mathbf{Pr}\left[T \geq k \mid \sigma_k = i\right]\right] \cdot \sum_{i \in [n]} \mathbf{Pr}\left[V_i > t \mid \sigma_k = i\right] && \text{(reorder)} \\
&= \left[\frac{1}{n} \sum_{k \in [n]} \mathbf{Pr}\left[T \geq k \mid \sigma_k = i\right]\right] \cdot \sum_{i \in [n]} \mathbf{Pr}\left[V_i > t\right] && \text{(independence)} \\
&\geq \left[\frac{1}{n} \sum_{k \in [n]} \mathbf{Pr}\left[T > k\right]\right] \cdot \sum_{i \in [n]} \mathbf{Pr}\left[V_i > t\right] && \text{(Lemma 4.3)} \\
&\geq \left[\frac{1}{n} \sum_{k \in [n]} \mathbf{Pr}\left[T > k\right]\right] \cdot \mathbf{Pr}\left[\max_i V_i > t\right], && \text{(union bound)}
\end{aligned}
$$

which completes the proof. □

It remains to show that Lemma 4.2 implies Lemma 4.1. To this end, we next show a lower bound on $\mathbf{Pr}[T > k]$.

Lemma 4.4 (Lower bound stopping time) *Consider a threshold algorithm that uses the sequence of non-increasing thresholds $\tau_1 \geq \ldots \geq \tau_n$. For $i \in [n]$, define*

$$\alpha_i := \mathbf{Pr}\left[\max_i V_i \leq \tau_i\right].$$

Then, for all $k \in [n]$,

$$\mathbf{Pr}\left[T > k\right] \geq \left(\prod_{j \in [k]} \alpha_j\right)^{\frac{1}{n}}.$$

Proof To prove the lower bound, we rewrite $\mathbf{Pr}[T > k]$ using the exponential function, and exploit the convexity of that function.

More formally, we have

$$\begin{aligned}
\mathbf{Pr}\left[T > k\right] &= \mathbf{E}_\sigma\left[\mathbf{Pr}\left[V_{\sigma_1} \le \tau_1, \ldots, V_{\sigma_k} \le \tau_k\right]\right] && \text{(def. } T\text{)}\\
&= \mathbf{E}_\sigma\left[\prod_{j\in[k]} \mathbf{Pr}\left[V_{\sigma_j} \le \tau_j\right]\right] && \text{(independence)}\\
&= \mathbf{E}_\sigma\left[\exp\left(\sum_{j\in[k]} \ln\left(\mathbf{Pr}\left[V_{\sigma_j} \le \tau_j\right]\right)\right)\right] && (\exp(\ln(x)) = x)\\
&\ge \exp\left(\mathbf{E}_\sigma\left[\sum_{j\in[k]} \ln\left(\mathbf{Pr}\left[V_{\sigma_j} \le \tau_j\right]\right)\right]\right) && \text{(convexity)}\\
&= \exp\left(\sum_{j\in[k]} \frac{1}{n} \sum_{i\in[n]} \ln\left(\mathbf{Pr}\left[V_i \le \tau_j\right]\right)\right) && \text{(random permutation)}\\
&= \left(\prod_{j\in[k]} \prod_{i\in[n]} \mathbf{Pr}\left[V_i \le \tau_j\right]\right)^{\frac{1}{n}}\\
&= \left(\prod_{j\in[k]} \mathbf{Pr}\left[V_1 \le \tau_j, \ldots, V_n \le \tau_j\right]\right)^{\frac{1}{n}} && \text{(independence)}\\
&= \left(\prod_{j\in[k]} \alpha_j\right)^{\frac{1}{n}}, && \text{(def. } \alpha_j\text{)}
\end{aligned}$$

which establishes the claimed bound. □

We are now ready to prove Lemma 4.1.

Proof of Lemma 4.1 For $t \le \tau$, we have

$$\begin{aligned}
\mathbf{Pr}\left[V_{\sigma_T} > t\right] &\ge \mathbf{Pr}\left[T < \infty\right] \cdot \mathbf{Pr}\left[\max_i V_i > t\right] && \text{(Lemma 4.4)}\\
&= (1-\alpha) \cdot \mathbf{Pr}\left[\max_i V_i > t\right]. && \text{(def. } \alpha\text{)}
\end{aligned}$$

For $t > \tau$, we have

$$\begin{aligned}
\mathbf{Pr}\left[V_{\sigma_T} > t\right] &\ge \left[\frac{1}{n}\sum_{k=1}^{n} \mathbf{Pr}\left[T > k\right]\right] \cdot \mathbf{Pr}\left[\max_i V_i > t\right] && \text{(Lemma 4.2)}\\
&\ge \left[\frac{1}{n}\sum_{k=1}^{n} \alpha^{\frac{k}{n}}\right] \cdot \mathbf{Pr}\left[\max_i V_i > t\right] && \text{(Lemma 4.4)}
\end{aligned}$$

$$= \frac{(1-\alpha)\alpha^{\frac{1}{n}}}{n \cdot (1-\alpha^{\frac{1}{n}})} \cdot \mathbf{Pr}\left[\max_i V_i > t\right]$$
$$\geq -\frac{(1-\alpha)}{\ln(\alpha)} \cdot \mathbf{Pr}\left[\max_i V_i > t\right] \qquad (n \to \infty),$$

which completes the proof. □

We conclude with the proof of Theorem 4.1.

Proof of Theorem 4.1 Consider the single-threshold algorithm Alg with threshold τ such that $\mathbf{Pr}[\max_i V_i \leq \tau] = 1/e$. Then, by Lemma 4.1,

$$\begin{aligned} \textsc{Alg}(\boldsymbol{F}) = \int_0^\infty \mathbf{Pr}\left[V_{\sigma_T} > t\right]\, dt &\geq \int_0^\infty \left(1 - \frac{1}{e}\right) \cdot \mathbf{Pr}\left[\max_i V_i > t\right]\, dt \\ &= \left(1 - \frac{1}{e}\right) \cdot \mathbf{E}\left[\max_i V_i\right] \\ &= \left(1 - \frac{1}{e}\right) \cdot \textsc{Opt}(\boldsymbol{F}), \end{aligned}$$

as claimed. □

4.1.2 Two Thresholds

We have already seen in Theorem 3.2 that with a single threshold we cannot hope to go beyond a $(1-1/e)$-approximation, even in the special case of i.i.d. random variables. However, similar to the i.i.d. case, it is possible to improve the approximation guarantee by using more than one threshold. However, existing analyses are rather intricate, and unlike in the i.i.d. case there remains a gap between the best known upper and lower bounds.

Below we provide a sample of the spirit of these improved approximation guarantees, by showing how to beat the $(1 - 1/e)$-guarantee of Theorem 4.1, with an algorithm that uses two thresholds $\tau_1 \geq \tau_2$, with τ_1 the threshold for the first $\lceil n/2 \rceil$ random variables, and τ_2 the threshold for the remaining $\lfloor n/2 \rfloor$ random variables. For ease of presentation, in addition to assuming that the F_i are continuous, in the following we assume that n is an even number.

Theorem 4.2 (Two thresholds) *Consider the Prophet Secretary problem and the two-thresholds algorithm* Alg *that uses thresholds $\tau_1 \geq \tau_2$ such that* $\mathbf{Pr}[\max_i V_i \leq \tau_1] = 0.3849 > 1/e$ *and* $\mathbf{Pr}[\max_i V_i \leq \tau_2] = 0.3406 < 1/e$. *Then,*

$$\textsc{Alg}(\boldsymbol{F}) \geq \left(1 - \frac{1}{e} + \frac{1}{200}\right) \cdot \textsc{Opt}(\boldsymbol{F}).$$

The key insight that drives this improvement is a lemma that refines the stochastic dominance relation established in Lemma 4.1.

Lemma 4.5 *(Stochastic dominance) Let $0 < \beta < \alpha < 1$. Consider the two-thresholds algorithm that uses thresholds $\tau_1 \geq \tau_2$ such that $\mathbf{Pr}[\max_i V_i \leq \tau_1] = \alpha$ and $\mathbf{Pr}[\max_i V_i \leq \tau_2] = \beta$. Then with T the stopping time of that algorithm, it holds that*

$$\mathbf{Pr}\left[V_{\sigma_T} > t\right] \geq \min\left\{1 - \frac{\alpha+\beta}{2}, \right.$$
$$\frac{1-\alpha}{2(1-\beta)} - \sqrt{\alpha}\frac{1-\sqrt{\beta}}{\ln\beta},$$
$$\left. -\frac{1-\sqrt{\alpha}}{\ln\alpha} - \sqrt{\alpha}\frac{1-\sqrt{\beta}}{\ln\beta}\right\} \cdot \mathbf{Pr}\left[\max_i V_i > t\right].$$

Similar to our single-threshold proof, we prove Lemma 4.5 by relating the probability that $V_{\sigma_T} > t$ to the stopping time T of the algorithm.

Lemma 4.6 (Connection to stopping time) *Consider the single-threshold algorithm with threshold τ and stopping time T. For all $t \geq 0$, it holds that*

$$\mathbf{Pr}\left[V_{\sigma_T} > t\right] \geq \gamma_2(t) \cdot \mathbf{Pr}\left[\max_i V_i > t\right],$$

where

$$\gamma_2(t) := \begin{cases} \mathbf{Pr}[T < \infty] & \text{for } t \leq \tau_2 \\ \frac{\mathbf{Pr}[T \leq n/2]}{1-\beta} + \frac{1}{n}\sum_{k>n/2}^{n} \mathbf{Pr}[T > k] & \text{for } \tau_2 < t \leq \tau_1\text{, and} \\ \frac{1}{n}\sum_{k=1}^{n} \mathbf{Pr}[T > k] & \text{for } \tau_1 < t. \end{cases}$$

Note that the first and the last case in the definition of $\gamma_2(t)$ coincide with the first and second case in the definition of $\gamma_1(t)$ in Lemma 4.2.

Proof of Lemma 4.6 The first case, where $t \leq \tau_2$, and the third case, where $\tau_1 < t$, are analogous to the proof of Lemma 4.2. So, let us focus on the case where $\tau_2 < t \leq \tau_1$. In this case,

$$\begin{aligned}
&\mathbf{Pr}\left[V_{\sigma_T} > t\right] \\
&\quad = \mathbf{Pr}\left[V_{\sigma_T} > t, T \leq n/2\right] + \mathbf{Pr}\left[V_{\sigma_T} > t, T > n/2\right] && \text{(partition on } T) \\
&\quad = \mathbf{Pr}\left[T \leq n/2\right] + \mathbf{Pr}\left[V_{\sigma_T} > t, T > n/2\right] && (t \leq \tau_1) \\
&\quad \geq \frac{\mathbf{Pr}[T \leq n/2]}{1-\beta}\, \mathbf{Pr}\left[\max_i V_i > t\right] + \mathbf{Pr}\left[V_{\sigma_T} > t, T > n/2\right]. && (t > \tau_2)
\end{aligned}$$

An argument analogous to the case $t > \tau$ in the proof of Lemma 4.2 shows that for $t \geq \tau_2$,

$$\mathbf{Pr}\left[V_{\sigma_T} > t, T > n/2\right] \geq \frac{1}{n} \sum_{k>n/2}^{n} \mathbf{Pr}\left[T > k\right].$$

By combining the two bounds we obtain the required bound for the case where $\tau_2 < t \leq \tau_1$. □

As a final ingredient, before we prove how Lemma 4.6 implies Lemma 4.5, we show an upper bound on $\mathbf{Pr}[T > k]$.

Lemma 4.7 (Upper bound stopping time) *Consider a threshold algorithm that uses the sequence of non-increasing thresholds $\tau_1 \geq \ldots \geq \tau_n$. For $i \in [n]$, define*

$$\alpha_i = \mathbf{Pr}\left[\max_i V_i \leq \tau_i\right].$$

Then, for all $k \in [n]$,

$$\mathbf{Pr}\left[T > k\right] \leq 1 - \frac{k}{n} + \frac{1}{n} \sum_{j \in [k]} \alpha_j.$$

Proof Fix the distributions of $V_1, \ldots, V_n$ and the thresholds $\tau_1, \ldots, \tau_n$. We construct a coupling between the stopping times on the instance given by $V_1, \ldots, V_n$ and the instance given by the variables $\max_i V_i, 0, \ldots, 0$ as follows. First, draw the values of $V_1, \ldots, V_n$ and a uniform random permutation. Then, define the stopping time T as usual in the instance $V_1, \ldots, V_n$. Focus on the maximum realization of the values. Define a second stopping $\overline{T}$ time as follows. If the maximum realization surpassed the threshold assigned to its arrival time, then assign it this time. Otherwise, it is infinity. Note that the second stopping time behaves as the usual stopping time in the instance given by $\max_i V_i, 0, \ldots, 0$.

By construction,

$$T \leq \overline{T}.$$

In particular, for all $k \in [n]$,

$$\begin{aligned}
\mathbf{Pr}\left[T > k\right] &\leq \mathbf{Pr}\left[\overline{T} > k\right] \\
&= \frac{n-k}{n} + \frac{1}{n} \sum_{j \in [k]} \mathbf{Pr}\left[\max_i V_i \leq \tau_j\right] \\
&= 1 - \frac{k}{n} + \frac{1}{n} \sum_{j \in [k]} \alpha_j,
\end{aligned}$$

where the first equality holds because $\bar{T} > k$ if the maximum does not land in any of the first k positions, or if it lands in one of the first k positions and does not exceed its threshold. $\square$

We are now ready to prove Lemma 4.5.

Proof of Lemma 4.5 We prove the lemma by providing bounds on the terms of $\gamma_2(t)$ as in Lemma 4.5. By Lemma 4.7, we have that

$$\mathbf{Pr}\,[T<\infty]=1-\mathbf{Pr}\,[T>n]\ \geq 1-\frac{\alpha}{2}-\frac{\beta}{2},$$

and

$$\mathbf{Pr}\,[T\leq n/2]=1-\mathbf{Pr}\,[T>n/2]\geq 1-\frac{1}{2}-\frac{\alpha}{2}=\frac{1-\alpha}{2}.$$

On the other hand, by Lemma 4.4, we have

$$\begin{aligned}\frac{1}{n}\sum_{k>n/2}^{n}\mathbf{Pr}\,[T>k]&\geq\sqrt{\alpha}\frac{1}{n}\sum_{k>n/2}^{n}\beta^{\frac{k-n/2}{n}}=\sqrt{\alpha}\frac{1}{n}\sum_{k=1}^{n/2}\beta^{\frac{k}{n}}\\&=\sqrt{\alpha}\frac{\beta^{\frac{1}{n}}}{n}\left(\frac{1-\sqrt{\beta}}{1-\beta^{\frac{1}{n}}}\right)\xrightarrow[n\to\infty]{}-\sqrt{\alpha}\frac{1-\sqrt{\beta}}{\ln\beta},\end{aligned}$$

and

$$\begin{aligned}\frac{1}{n}\sum_{k=1}^{n}\mathbf{Pr}\,[T>k]&=\frac{1}{n}\sum_{k=1}^{n/2}\mathbf{Pr}\,[T>k]+\frac{1}{n}\sum_{k>n/2}^{n}\mathbf{Pr}\,[T>k]\\&\geq\frac{1}{n}\sum_{k=1}^{n/2}\alpha^{\frac{k}{n}}+\sqrt{\alpha}\frac{\beta^{\frac{1}{n}}}{n}\left(\frac{1-\sqrt{\beta}}{1-\beta^{\frac{1}{n}}}\right)\\&=\frac{\alpha^{\frac{1}{n}}}{n}\left(\frac{1-\sqrt{\alpha}}{1-\alpha^{\frac{1}{n}}}\right)+\sqrt{\alpha}\frac{\beta^{\frac{1}{n}}}{n}\left(\frac{1-\sqrt{\beta}}{1-\beta^{\frac{1}{n}}}\right)\\&\xrightarrow[n\to\infty]{}-\frac{1-\sqrt{\alpha}}{\ln\alpha}-\sqrt{\alpha}\frac{1-\sqrt{\beta}}{\ln\beta}.\end{aligned}$$

By substituting the corresponding terms in the definition of $\gamma_2(t)$, we obtain the claimed bound. $\square$

We conclude with the proof of Theorem 4.2.

Proof of Theorem 4.2 The proof follows by the same argument as the one used to establish Theorem 4.1, except that now we use Lemma 4.5 and choose $\alpha=0.3849$ and $\beta=0.3406$. $\square$

4.1.3 Upper bound of $\sqrt{3}-1$

We conclude with an impossibility result showing that no algorithm for the Prophet Secretary problem can achieve a better than $\sqrt{3}-1 \approx 0.732$ approximation. Note that this in particular presents a formal separation from the i.i.d. case studied in Section 3.3.

Theorem 4.3 *For any $\varepsilon > 0$ there exist an instance $\boldsymbol{F} = F_1 \times \cdots \times F_n$ of the Prophet Secretary problem, such that for any algorithm* Alg *it holds that*

$$\textsc{Alg}(\boldsymbol{F}) \leq (\sqrt{3} - 1 + \varepsilon) \cdot \textsc{Opt}(\boldsymbol{F})$$

Note that just like in previous hardness results, the following proof uses a discrete distribution but allows the algorithm to randomize.

Proof Take $\gamma \in [0, 1]$ and consider the following instance with $n+1$ random variables $V_1, \ldots, V_{n+1}$ distributed as follows:

$$V_1, \ldots, V_n \sim \begin{cases} n & \text{w.p. } \frac{1}{n^2} \\ 0 & \text{w.p. } 1 - \frac{1}{n^2} \end{cases} \quad \text{and} \quad V_{n+1} \equiv \gamma.$$

Consider any algorithm Alg, and let T be the implied stopping time. Clearly, any reasonable algorithm Alg would always accept a reward of n, and never accept a reward of 0. Therefore, the algorithm's only decision is whether to accept a reward of γ or not.

Solving the dynamic programming that defines the optimal online algorithm, there will be an index $j_n^* \in [n+1]$ such that the algorithm accepts reward γ if and only if it appears in position $j_n^\star$ or later.

Next we derive formulas for the expected value achieved by the algorithm, conditioned on the arrival time of the $(n+1)$st random variable. For $i = 1, \ldots, j_n^* - 1$ we have

$$\mathbf{E}\left[V_{\sigma_T} \mid \sigma_i = n+1\right] = \left(1 - \left(1 - \frac{1}{n^2}\right)^n\right) \cdot n,$$

and for $i = j_n^*, \ldots, n+1$ we have

$$\mathbf{E}\left[V_{\sigma_T} \mid \sigma_i = n+1\right] = \left(1 - \left(1 - \frac{1}{n^2}\right)^i\right) \cdot n + \left(1 - \frac{1}{n^2}\right)^i \cdot \gamma.$$

Combining the two formulas for $\mathbf{E}\left[V_{\sigma_T} \mid \sigma_i = n+1\right]$, we obtain the following upper bound

$$\mathbf{E}\left[V_{\sigma_T}\right] = \frac{j_n^* - 1}{n+1} \cdot \left(1 - \left(1 - \frac{1}{n^2}\right)^n\right) \cdot n$$

$$+\frac{1}{n+1}\sum_{i=j_n^*}^{n+1}\left[\left(1-\left(1-\frac{1}{n^2}\right)^i\right)\cdot n+\left(1-\frac{1}{n^2}\right)^i\cdot\gamma\right].$$

Consider $\lambda := \limsup_{n\to\infty}\frac{j_n^*}{n+1} \in [0,1]$. Note that as n grows to infinity $(1-1/n^2)^i \approx e^{-i/n^2} \approx 1-i/n^2$ so that

$$\mathbf{E}\left[V_{\sigma_T}\right]=\frac{j_n^*-1}{n+1}+\frac{1}{n}\sum_{i=j_n^*}^{n+1}\left(\frac{i}{n}+\gamma\right)+o\left(\frac{1}{n}\right).$$

We thus have:

$$\limsup_{n\to\infty}\mathbf{E}\left[V_{\sigma_T}\right]=\lambda+\int_{x=\lambda}^{1}(x+\gamma)\,dx=\lambda+\frac{1-\lambda^2}{2}+\gamma(1-\lambda)\le 1+\frac{\gamma^2}{2},$$

where the last inequality comes from maximizing over $\lambda \in [0,1]$.

On the other hand,

$$\textsc{Opt}(\boldsymbol{F})=\left(1-\left(1-\frac{1}{n^2}\right)^n\right)\cdot n+\left(1-\frac{1}{n^2}\right)^n\cdot\gamma\xrightarrow[n\to\infty]{}1+\gamma.$$

Choosing $\gamma=\sqrt{3}-1$, we thus obtain

$$\limsup_{n\to\infty}\frac{\mathbf{E}[V_{\sigma_T}]}{\textsc{Opt}(\boldsymbol{F})}\le\frac{1+\frac{\gamma^2}{2}}{1+\gamma}=\sqrt{3}-1,$$

as claimed. □

4.2 Free Order

In the free order problem, the designer is free to select the order in which the random variables are inspected. Recall that, for any permutation σ of $[n]$, if the random variables arrive in order $V_{\sigma(1)},\ldots,V_{\sigma(n)}$ then the optimal online policy can be computed by backward induction. Let V_σ denote the expected reward of the optimal online policy for arrival order σ. Then the free order problem is to choose σ maximizing V_σ.

We note that restricting to a static, non-adaptive ordering σ is without loss of generality. More generally, an adaptive ordering policy selects which variable to check in each round based on the realizations observed in previous rounds. However, adaptive ordering does not provide any benefit over non-adaptive ordering.

Proposition 4.1 *For any adaptive ordering policy, say with expected reward V^*, there exists a static ordering σ such that $V_\sigma \ge V^*$.*

Proof We prove this result by induction on n. If $n = 1$ then there is only a single ordering so the result is trivial. Suppose $n > 1$. Our adaptive policy selects some variable V_i to reveal first, possibly at random. Given realization v_i, the policy either accepts v_i or obtains the expected continuation value of the policy. As the latter is the value of an adaptive ordering policy on the remaining variables, we conclude by induction there is some static ordering $\sigma^{(i)}$ over $[n] \setminus \{i\}$ that yields weakly higher expected continuation value, which we will denote by $W(\sigma^{(i)})$.

As the adaptive policy makes some (possibly randomized) choice of index i, we conclude that V^* is at most $\max_i \mathbf{E}[\max\{v_i, W(\sigma^{(i)})\}]$. But this value is achievable by the static ordering σ that first selects

$$i \in \arg\max \mathbf{E}[\max\{v_i, W(\sigma^{(i)})\}]$$

then proceeds according to ordering $\sigma^{(i)}$. So $V_\sigma = \max_i \mathbf{E}[\max\{v_i, W(\sigma^{(i)})\}] \geq V^*$ as claimed. □

Another natural question is whether the power to select the order is helpful in worst-case problem instances. One simple observation is that when the reward distributions are identically distributed, the arrival order does not influence the expected reward from the optimal policy. It may be tempting to conclude that iid problem instances are worst-case instances for a free-order decision-maker, but this is not necessarily the case for fixed n. Consider the following example:

Example 4.2 In this example there are two variables, V_1 and V_2. The value of V_1 is 0 or 2 with equal probability. The value of V_2 is $1/\epsilon$ with probability ϵ and otherwise value 1. This problem instance does not feature identical value distributions, but retains an important order-obliviousness property of i.i.d. problem instances: the optimal policy obtains expected reward 2 regardless of the order in which the variables are processed. This means that, as in i.i.d. problem instances, the power to select the order does not convey a benefit. Moreover, the expected maximum approaches $5/2$ as ϵ grows small, meaning that the ratio between the expected maximum and the expected value of the best policy approaches $4/5$. This is worse than the worst i.i.d. problem instance for $n = 2$.

The example above applies to the case $n = 2$. For larger n, the worst-case competitive ratio for i.i.d. instances becomes more stringent. It is an open question whether there is separation between worst-case competitive ratio, over all possible n, for the i.i.d. problem and the free order problem.

So what *is* the best static order? For some classes of problems, selecting the best order is easy. Let's start with a simple example: Bernoulli instances, where each random variable V_i is supported on two values, one of which being 0. For Bernoulli problem instances, it turns out that the best order is trivial: it is always best to reveal the random variables in descending order by the positive value in their support.

Proposition 4.2 *Consider a Bernoulli instance of the free order problem where each variable V_i is supported on $\{0, w_i\}$ for some $w_i \geq 0$. Then it is optimal the observe the variables in decreasing order of w_i.*

Proof The offline optimal solution selects whichever variable V_i has largest w_i from among all those with positive realization. But this solution can be implemented online if variables are inspected in decreasing order of w_i, by choosing the first observed positive value. □

Unfortunately, this simple analysis for Bernoulli random variables does not extend. It turns out that for even slightly more complicated distributions, it can be computationally difficult to determine the best order.

Theorem 4.4 *The free-order problem is NP-hard. This is true even when each random variable V_i has positive support on exactly three values: 0, 1, and a positive value $m_i \in (0, 1)$.*

We prove this result in two steps. First, we characterize the structure of optimal orderings for a special case of problem instances. It will turn out that choosing the best ordering reduces to choosing a subset of the random variables that will be accepted only if their value is 1. Second, we show that the subset sum problem, which is well-known to be NP-hard, reduces to this simplified free-order problem, which implies that the free-order problem is NP-hard as well.

We begin by describing a restricted set of problem instances. First, as alluded to in the theorem statement, we assume that each V_i is supported on three values: 0, 1, and some value $m_i \in (0, 1)$ (which can vary for different random variables). Second, we assume that each random variable has the same expected value, conditional on being positive: there is some constant $C \geq 0$ such that $\mathbf{E}[V_i | V_i > 0] = C$ for all i.

We now claim that for any such problem instance, there is an optimal policy for the free-order problem with the following structure: the variables are partitioned into two sets $S \subseteq [n]$ and $T = [n] \setminus S$, all variables in S are inspected before all variables in T, a variable in S is accepted if and only if its value is 1, and a variable in T is accepted if and only if its value is positive. Call such a policy *well-ordered*.

Lemma 4.8 *For any problem satisfying the assumptions above, every optimal free-order policy is well-ordered.*

Proof Pick some optimal policy, say with ordering σ. We can assume without loss of generality that this policy does not accept any variable with realized reward 0. Suppose for contradiction that it is not well-ordered. Write $S \subseteq [n]$ for the set of indices j such that V_j is accepted (conditional on being observed) only if its realized value is 1, and T for those indices j such that V_j is accepted if its realized value is either m_j or 1. Then S and T form a disjoint partition of $[n]$. Since the policy is not well-ordered, the variables in S don't all appear before the variables in T according to σ. In particular, there exists some i such that $j := \sigma(i) \in T$ and $k := \sigma(i+1) \in S$.

Consider now a different policy that uses the same acceptance threshold for each random variable, but changes the ordering from σ to σ' as follows: $\sigma'(i) = k$, $\sigma'(i+1) = j$, and on all other indices σ and σ' agree. In other words, σ' reverses the order of j and k, observing k first followed by j. We then observe that this new policy and the original policy obtain the same reward in the following cases:

- if round i isn't reached, since then we never observe V_j or V_k;
- if $V_j = V_k$, since then the order of observation doesn't matter; or
- if $V_j < m_j$ or $V_k < 1$, since then at most one of the two variables would be accepted and hence the order of observation has no effect.

We conclude that the reward obtained by the two policies can differ only if round i is reached, $V_j = m_j$, and $V_k = 1$. In such an instance, the original policy accepts V_j for a reward of m_j, whereas the modified policy accepts V_k for a reward of 1. Thus, pointwise for every realization, the new policy has weakly higher reward, and strictly higher for some realizations. The new policy therefore has strictly higher expected reward, contradicting optimality. □

Note that the well-ordered property does not constrain the order in which variables in S are observed, nor the order of the variables in T. We next claim that the ordering of the variables in T does not influence the expected reward. This claim makes use of the assumption that all variables have the same expected value conditional on being positive.

Lemma 4.9 *Any two well-ordered policies with the same partition S and T have the same expected reward.*

Proof Some notation: say that variable i has value 0 with probability p_i, value m_i with probability q_i, and value 1 with probability r_i. Fix some well-ordered policy, with corresponding partition S and T. This policy selects a variable from S, receiving reward 1, as long as not all elements of S have $V_i < 1$. In other words, the contribution to the expected reward from the variables in S is precisely $(1 - \prod_{i \in S}(1 - r_i))$, the probability that at least one of those variables has value 1. With probability $\prod_{i \in S}(1 - r_i)$, the policy rejects all variables in S and proceeds to the variables in T.

We next claim that the policy that considers only the elements of T, and accepts the first with non-zero value, generates expected reward $W(1-\prod_{i \in T} p_i)$. Indeed, $(1 - \prod_{i \in T} p_i)$ is the probability that not all variables in T are 0. But, conditioning on this event, for any realization of the first variable with non-zero value, the expected reward from that variable is precisely W (the assumed expected value of each variable conditioned on being non-zero).

We conclude that the expected reward of the policy is precisely

$$\left(1 - \prod_{i \in S}(1 - r_i)\right) + W\left(\prod_{i \in S}(1 - r_i)\right)\left(1 - \prod_{i \in T} p_i\right),$$

which is independent of the ordering of variables within S and within T. □

We have now argued that finding an optimal policy reduces to the problem of choosing an appropriate the partition of $[n]$ into S and T. We can now establish NP-hardness by showing that if we can build the optimal partition, then we can solve the well-known subset sum problem.

Definition 4.1 (Subset Sum) The subset sum problem takes as input a collection of n positive numbers $a_1, \ldots, a_n$ and a target value $B > 0$. The problem is to determine whether there is a subset $S \subseteq [n]$ such that $\sum_{i \in S} a_i = B$, or that no such subset exists.

The subset sum problem is known to be NP-hard. It will actually be more convenient for us to use the closely related subset product problem, where each a_i is an integer greater than 1 and at most B, and we are to determine whether there is a subset $S \subseteq [n]$ such that $\prod_{i \in S} a_i = B$. This problem is NP-hard as well, as it reduces to the subset sum problem by taking logarithms.

Proof of Theorem 4.4 Given an instance of the subset product problem, specified by values $a_1, \ldots, a_n$ and B, we construct an instance of the free-order problem as follows. Variable V_i has value 0 with probability $1/a_i^2$, value $m_i := (B^2 - a_i)/(B^2 + 1)$ with probability $(a_i - 1)/(a_i^2)$, and value 1 with probability $(a_i - 1)/a_i$. Note that these probabilities add to 1, and that $m_i \in (0, 1)$ (where the latter uses the fact that each a_i is greater than 1 and at most B). Furthermore, we note that for each V_i we have

$$\mathbf{E}[V_i | V_i > 0] = \frac{B^2}{B^2 + 1}$$

which is independent of i.

We conclude that this problem instance satisfies our conditions, and hence the value of any well-ordered policy with partition S and T is independent of the ordering of variables within S and T.

Indeed, from our calculation above, the value of a well-ordered policy with partition S and T is precisely

$$\left(1 - \prod_{i \in S} \left(1 - \frac{a_i - 1}{a_i}\right)\right) + \left(\frac{B^2}{B^2 + 1}\right) \left(\prod_{i \in S} \left(1 - \frac{a_i - 1}{a_i}\right)\right) \left(1 - \prod_{i \in T} \frac{1}{a_i^2}\right)$$

which is equal to

$$1 - \prod_{i \in S} \frac{1}{a_i} + \left(\frac{B^2}{B^2 + 1}\right) \left(\prod_{i \in S} \frac{1}{a_i}\right) \left(1 - \prod_{i \in T} \frac{1}{a_i^2}\right).$$

Letting $z = \prod_{i \in [n]} a_i$ and $z_T = \prod_{i \in T} a_i$, we can rewrite the policy value as

$$f(z_T) := 1 - \frac{z_T}{z} + \frac{z_T}{z} \left(1 - \frac{1}{z_T^2}\right) \frac{B^2}{B^2 + 1}.$$

Taking derivatives, we have

$$f'(z_T) = -\frac{1}{z} + \left(\frac{B^2}{B^2 + 1}\right) \left(\frac{1}{z} + \frac{1}{z_T^2 z}\right)$$

and

$$f''(z_T) = \frac{-2B^2}{z_T^3 z (B^2 + 1)} < 0$$

and hence f is strictly concave and thus achieves its unique maximum where $f'(z_T) = 0$, which occurs when $z_T = B$.

We conclude that if our instance of subset product is a "yes" instance, then the corresponding free-order problem achieves its maximum via a partition S and T in which $z_T = B$. On the other hand, if the subset product is a "no" instance then there exists no T for which $z_T = B$, and hence the optimal partition will have $z_T \neq B$. An oracle for the free-order problem therefore solves the subset product problem, so the NP-hardness of subset product therefore implies NP-hardness of the free-order problem. □

Take-Aways

- The classic prophet inequality takes arrival order as arbitrary and fixed. We can also consider cases where rewards arrive in uniformly random order—the Prophet Secretary problem— or where the gambler can choose the order—the Free Order Prophet problem.
- For the Prophet Secretary problem, a single-threshold policy can obtain a $1 - 1/e$ approximation, matching the tight bound for single-threshold policies even in the i.i.d. setting. But more complex policies can do better than $1 - 1/e$.
- For the Free-Order Prophet problem, choosing the best order is NP-hard but there is an FPTAS.
- For both Prophet Secretary and the Free-Order Prophet problem, the worst-case competitive ratio is not known. For Prophet Secretary the competitive ratio is strictly worse than the bound for i.i.d. valuations; it is an open question whether the same is true for the Free-Order Prophet problem.

Chapter Notes

In this chapter, we explored variants of the prophet inequality problem, in which we weaken the adversary by either assuming that the input comes in random oder or by allowing the algorithm to choose the order in which it inspects the random variables. In both cases, this results in an improved worst-case approximation guarantee. A related paper, which shows that being able to choose the arrival order is beneficial, is Arsenis et al. (2021). They show that for any fixed order of the random variables, the better of "forward" and "backward" yields an approximation ratio equal to the reciprocal of the golden ratio, namely $\psi^{-1} = 2/(1 + \sqrt{5}) \approx 0.618$.

The Prophet Secretary problem was introduced in (Esfandiari et al., 2017), who showed that it is possible to attain a $(1 - 1/e)$-approximation with one threshold per random variable. The result that this approximation guarantee can also be attained with a single threshold algorithm is due to Ehsani et al. (2024) (who also showed that this is best possible, even in the i.i.d. case). The subsequent work of Correa et al. (2021) showed how to obtain a 0.669-approximation through an approach called *blind strategies*, which sets thresholds corresponding to the same sequence of quantiles, irrespective of the distributions. In addition to the impossibility of $\sqrt{3} - 1 \approx 0.732$ that applies to any algorithm, they also showed an improved impossibility of 0.675 for blind strategies. The proofs that we presented for a single threshold and two thresholds are also blind strategies. We are grateful to Raimundo Saona for suggesting the unified argument that allows to derive the two bounds.

Improved bounds of 0.6724 and 0.688 were recently given by Harb (2025) and Chen et al. (2025). Importantly, the latter breaks the impossibility that applies to blind strategies, which shows that blind strategies are provably suboptimal. However, there still remains a gap between the best known algorithm, and the best know impossibility. The best known impossibilities currently stand at 0.7254 due to Bubna and Chiplunkar (2023) and 0.7235 due to Giambartolomei et al. (2024). It remains an open problem to close the remaining gap, and thus identify which algorithmic approach yields a worst-case optimal bound.

A classic paper that studies the Free-Order Prophet problem is Hill (1983), who among other things shows that there is no gain from adaptively choosing the order. The NP-hardness result discussed in this chapter is by Agrawal et al. (2020). The results of Chakraborty et al. (2010) together with the reduction of Correa et al. (2019) imply a Polynomial-Time Approximation Scheme (PTAS). Subsequent work by Liu et al. (2021) implies the existence of a PTAS with improved running time.

An algorithm for the Free-Order Prophet problem with a competitive ratio of 0.7251 was given by Peng and Tang (2022). Bubna and Chiplunkar (2023) further improve this to 0.7258, formally separating the Free-Order Prophet inequality problem from Prophet Secretary problem. Whether there is a similar separation between the free-order problem and the i.i.d. case remains an open problem.

References

Agrawal, Shipra, Sethuraman, Jay, and Zhang, Xingyu. 2020. On Optimal Ordering in the Optimal Stopping Problem. Pages 187—-188 of: *EC 2020*.

Arsenis, Makis, Drosis, Odysseas, and Kleinberg, Robert. 2021. Constrained-Order Prophet Inequalities. Pages 2034–2046 of: *SODA 2021*. ACM-SIAM.

Bubna, Archit, and Chiplunkar, Ashish. 2023. Prophet Inequality: Order selection beats random order. Pages 302—336 of: *EC 2023*.

Chakraborty, Tanmoy, Even-Dar, Eyal, Guha, Sudipto, Mansour, Yishay, and Muthukrishnan, S. 2010. Approximation schemes for sequential posted pricing in multi-unit auctions. Pages 158—169 of: *WINE 2010*.

Chen, Ziyun, Huang, Zhiyi, Li, Dongchen, and Tang, Zhihao Gavin. 2025. *Prophet Secretary and Matching: the Significance of the Largest Item*. Pages 1371–1401.

Correa, José, Saona, Raimundo, and Ziliotto, Bruno. 2021. Prophet secretary through blind strategies. *Math. Program*, **190**, 483—521. An extended abstract appeared in SODA 2019.

Correa, José, Foncea, Patricio, Pizarro, Dana, and Verdugo, Victor. 2019. From pricing to prophets, and back! *Operations Research Letters*, **47**(1), 25–29.

Ehsani, Soheil, Hajiaghayi, MohammadTaghi, Kesselheim, Thomas, and Singla, Sahil. 2024. Prophet Secretary for Combinatorial Auctions and Matroids. *SIAM J. Comput.*, **53**(6), 1641–1662. An extended abstract appeared in SODA 2018.

Esfandiari, Hossein, Hajiaghayi, MohammadTaghi, Liaghat, Vahid, and Monemizadeh, Morteza. 2017. Prophet Secretary. *SIAM J. Discrete Math.*, **31**(3), 1685–1701. An extended abstract appeared in ESA 2015.

Giambartolomei, Giordano, Mallmann-Trenn, Frederik, and Saona, Raimundo. 2024. *Prophet Inequalities: Separating Random Order from Order Selection*.

Harb, Elfarouk. 2025. New Prophet Inequalities via Poissonization and Sharding. Pages 1222–1269 of: *SODA 2025*.

Hill, Theodore P. 1983. Prophet Inequalities and Order Selection in Optimal Stopping Problems. *Proceedings of the American Mathematical Society*, **88**(1), 131–137.

Liu, Allen, Leme, Renato Paes, Pál, Martin, Schneider, Jon, and Sivan, Balasubramanian. 2021. Variable Decomposition for Prophet Inequalities and Optimal Ordering. Page 692 of: *EC 2021*. ACM.

Peng, Bo, and Tang, Zhihao Gavin. 2022. Order Selection Prophet Inequality: From Threshold Optimization to Arrival Time Design. Pages 171–178 of: *FOCS 2022*.