Order-based Prophet Inequalities: *An Overview*

Ruijie Fang, Vaibhav Mehta, Sumanth Maddirala*

Abstract

Recently different variants of *Prophet Inequalities*, a central problem in the theory of optimal stopping, has sparked an emerging interest in computational economics. The general prophet problem asks the following: given n of real-valued, nonnegative random variables $X_1, X_2, ..., X_n \sim \mathcal{D}_1, ..., \mathcal{D}_n$, arriving online, we (the gambler) wish to pick a random variable $X_{i^*} \in \{X_1, ..., X_n\}$ such that the random variable we pick is as close to the value chosen by the prophet $\mathsf{OPT} = \mathbb{E}[\max_i X_i]$, as possible. There are many variations of the same problem: for instance, when we are required to retain k random variables instead of one random variable, whether the distributions are (not) known in advance, and when the distributions are i.i.d. In this paper, we present a survey of recent developments of prophet inequalities where the distributions are known in advance, but the random variables arrive online under three different order-based settings: (1) adversarial order, in which the order of arrival of the random variables X_i is fixed by some adversary; (2) random order, in which the random variable X_i's arrive in an order chosen uniformly at random; (3) free order, where an algorithm can be used to determine the order of arrival by looking at the input parameters.

Contents

*Contact: {ruijief|vaibhavm|sumanthm}@princeton.edu. This is a final survey paper done for COS 521, Fall 2021 at Princeton University. We thank Professor Matthew Weinberg for his advice and topic suggestion.

1 Preliminaries

1.1 Introduction

The general problem of prophet inequalities asks the following: given n of real-valued, nonnegative random variables $X_1, X_2, ..., X_n \sim \mathcal{D}_1, ..., \mathcal{D}_n$, arriving online, we (the gambler) wish to pick a random variable $X_{i^*} \in \{X_1, ..., X_n\}$ such that the random variable we pick is as close to the value chosen by the prophet $\mathsf{OPT} = \mathbb{E}[\max_i X_i]$, as possible. How well the gambler performs is typically measured by a multiplicative approximation ratio α: Let τ^* denote the stopping time of the gambler and let $\mathsf{OPT} = \mathbb{E}\left[\max(X_1, ..., X_n)\right]$ denote the expected reward of the prophet. We seek to design an algorithm such that

$$\mathbb{E}\left[X_{\tau^*}\right] \geq \alpha \mathsf{OPT}.$$

In this setting, a *lower bound* generally means an algorithm that concretely gets us an expected reward within α of the optimum, whereas an *upper bound* generally means an impossibility result, that we (the gambler) cannot achieve an approximation ratio to within $< \alpha$ of the optimum.

In *order-based prophet inequalities*, the order of arrival of the n random variables can change. We present a survey on prophet inequalities under different order-based models: adversarial order, in which an adversary picks an ordering of arrivals; free order, in which the algorithm can determine some chosen order of arrival based of the random variables; and random order, in which case the random variables are permuted uniformly at random and then arrive online.

1.2 Related Works

The research along adversarial order prophet inequalities is closed and complete. The initial work on prophet inequalities, [10], proved a tight upper bound of $\alpha = \frac{1}{2}$ which can also be achieved via multiple different strategies. In the free order model, the earliest work, by T. P. Hill [7], showed that when the optimal strategy is used, the reward obtained by a predetermined offline ordering is just as good, if not better, than a dynamic ordering computed online. This motivates the subsequent work by [2], which proved that the problem of finding an optimal offline optimal ordering is NP-hard, and [11], which designed improved polynomial-time approximation algorithms for optimal ordering. In the random order model, there are two cases: (1) when the random variables are all i.i.d., in which case, the problem is well studied with a tight upper bound governed by a solution of $\beta = 0.745$ to the Kertz equation [9] and a matching algorithm (lower bound) given by [1] and [4]; (2) when the random variables are non-i.i.d.; in which case, no tight bound exists to date, and the best upper bound and lower bound are 0.732 and 0.669 correspondingly, given by [5], with most recently [11] showing that given an appropriately configured choice of ϵ, removing ϵ^{-1} random variables from the input sequence recovers the Kertz bound in the non-i.i.d. case.

The structure of this paper is as follows. In section 3, we cover the classical result of a $\frac{1}{2}$-approximation in the adversarial order model by a proof due to [12]. In section 4, we present a simplified proof of the result in [7] that the optimal reward in obtained by offline ordering is just as good as an online ordering, and give, in expository fashion, a proof by [2] that finding the optimal offline ordering is NP-hard. In section 5 and section 6 correspondingly, we give an expository account of the results given by [9], [1], [4], and [5] on the i.i.d and non-i.i.d. random order models.

2 Adversarial Order

In the adversarial order prophet model, the random variables X_i draw from independent, but not necessarily identical, distributions $\mathcal{D}_1, ..., \mathcal{D}_n$. They arrive in some fixed, adversarially determined order. In this case it can be shown that the optimal approximation bound cannot be improved beyond $\frac{1}{2}$.

2.1 1/2 Approximation

Krengel and Sucheston [10], showed that the gambler can obtain at least $\frac{1}{2}$ using an optimal stopping rule. We first present a simpler proof of $\frac{1}{2}$ approximation due Samuel-Cahn [12], that uses the following threshold policies:

1. Let $t(c)$ = smallest $i < n$ such that $X_i \geq c$, otherwise $t(c) = n$.
2. Let $s(c)$ = smallest $i < n$ such that $X_i > c$, otherwise $s(c) = n$.

We define $\mathbb{E}^+[X_{t(c)}] = \mathbb{E}[X_{t(c)}I(X_{t(c)\geq x}]$ and $\mathbb{E}^+[X_{s(c)}] = \mathbb{E}[X_{s(c)}I(X_{s(c)} > x]$. Let m be the median of the distribution of X^*, i.e.

$$\Pr(X^* < m) = q \leq \frac{1}{2}, \qquad \Pr(X^* > m) = p \leq \frac{1}{2}$$

Finally, let

$$\beta = \sum_{i=1}^{n} \mathbb{E}[X_i - m]^+$$

Now we are in a position to formally state our theorem.

Theorem 1. *Let $X_1, ..., X_n$ be independent nonnegative random variables and let $X^* =$ max$(X_1, ..., X_n)$ then*

1. *If $\beta \geq m$ then $\mathbb{E}[X_n^*] \leq 2\mathbb{E}^+[X_{s(m)}] \leq 2\mathbb{E}[X_{s(m)}]$*
2. *If $\beta \leq m$ then $\mathbb{E}[X_n^*] \leq 2\mathbb{E}^+[X_{t(m)}] \leq 2\mathbb{E}[X_{t(m)}]$*

Proof. We begin by noting that $\mathbb{E}[X_n^*] \leq m + \mathbb{E}[X_n^* - m]^+ \leq m + \beta$.
Suppose $\beta \geq m$. Since m is the median,

$$\mathbb{E}^+[X_{s(m)}] = mp + \mathbb{E}[X_{s(m)} - m]^+ = mp + \mathbb{E}\left[\sum_{i=1}^{n} \mathbb{E}[X_i - m]^+ I(s(m) = i)\right]$$

$$= mp + \mathbb{E}\left[\sum_{i=1}^{n} \mathbb{E}[X_i - m]^+ I(s(m) > i - 1)\right]$$

By independence,

$$= mp + \mathbb{E}\left[\sum_{i=1}^{n} \mathbb{E}[X_i - m]^+ \Pr(s(m) > i - 1)\right]$$

$$\geq mp + \beta(1 - p) \geq (m + \beta)/2 \geq \mathbb{E}[X_n^*]/2$$

Similarly, when $\beta \leq m$,

$$\mathbb{E}^+[X_{t(m)}] = m(1 - q) + \mathbb{E}\left[\sum_{i=1}^{n} \mathbb{E}[X_i - m]^+ \Pr(t(m) > i - 1)\right]$$

$$\geq m(1 - q) + \beta p \geq (m + \beta)/2 \geq \mathbb{E}[X_n^*]/2$$

□

We note that $1/2$ approximation cannot be improved upon in the non-Independent and identically distributed case. This is a common folklore construction that we now formalize here. Set $X_{n-1} = \mu$ and set $X_n = 1$ and 0 with probability μ and $1 - \mu$. Setting all other X_i's smaller than μ and letting $\mu \to 0$ yields the result.
This motivates us to study the i.i.d case where the constant factor can be improved.
We remark that [12] also proves any non-adaptive threshold rule cannot improve the constant factor of $1/2$ in even the i.i.d case, where random order, free order and adversarial order prophet inequalities coincide.

3 Free Order

In the free order model, the algorithm ALG decides an ordering of the arrival of random variables $X_1, ..., X_n$. As we will show, the main problem under this context is how to choose an optimal offline ordering such that algorithm yields the best reward. This is called the optimal ordering problem; its history traces back to [7], when Hill first proved that an optimal offline ordering is as good as the optimal dynamic online ordering, and [2] proved that computing a restricted case of offline ordering is NP-hard via a reduction from the subset product problem. More recent works, such as [11] aim at providing better polynomial-time approximation algorithms for optimal ordering.

3.1 Order selection in optimal stopping.

We call an ordering *offline, static* if it is decided in advance before the variables X_i arrive. We call an ordering *online, dynamic* otherwise, where in a dynamic setting, the order of arrival of subsequent variables $X_{j+1}, ..., X_n$ could change dynamically based on the current and previous arrivals $X_1, .., X_j$ at stage j. T. P. Hill showed in 1983 [7] that when all variables X_i are independent, the optimal offline ordering is just as good as the optimal online ordering, even for a random process formed by a possibly infinite sequence of arriving random variables $(X_1, X_2, ...)$. However, when the independence assumption is removed, the claim may not be true. We present an abridged version of Hill's proof of the result below, but for the case where the sequence of arriving random variables $\mathcal{X} = \{X_1, ..., X_n\}$ is finite, and present a counterexample when the random variables are dependent.

Let $\mathcal{P}(n)$ denote the set of permutations $\pi : [n] \to [n]$. Let an *online dynamic ordering* be a map $r : \mathbb{R}^\infty \to \mathcal{P}(n)$ where $r(X_1, ..., X_{i-1})[i]$ determines the index of the i-th random variable that arrives, possibly based on the value of the previous arrivals. Without loss of generality one may view an *offline static ordering* as a permutation map $\pi \in \mathcal{P}(n)$ that does not take in any context of already arrived of random variables.

For instance, when $n = 3$, we can define the following *online dynamic ordering*: $r[1] \mapsto 2$, $r(X_1)[2] \mapsto \begin{cases} 1 & X_1 > \frac{5}{3} \\ 3 & \text{Otherwise} \end{cases}$, and $r(X_1, X_2)[3] \mapsto \begin{cases} 1 & X_1 \leq \frac{5}{3} \\ 3 & \text{Otherwise} \end{cases}$. On the other hand the permutation map $\pi := \begin{pmatrix} 1 & 3 & 2 \end{pmatrix}$ is a *static offline ordering.*

The problem we study is as follows. The player fixes a possibly dynamic ordering (policy) $r \in \mathcal{R}$ at the beginning. At each time step t, the random variable $X_{r(x_1,...,x_{t-1})[t]}$ arrives. The player either accepts this random variable as reward or moves on. If the player exhausts of random variables then the game stops automatically with zero reward. Observe that offline, the optimal policy can be computed via a dynamic program via backwards induction:

Definition 2 (Optimal policy)**.** For a sequence of random variables $\mathcal{X} := X_1, ..., X_n$ we define its optimal, offline stopping value as follows:

$$\begin{aligned} V(\mathcal{X}) &= V(X_1, ..., X_n) \\ &= \mathbb{E}[\max(X_1, \mathbb{E}[\max(X_2, ..., \mathbb{E}[\max(X_n, 0)], ...,)])] \end{aligned}$$

Similarly the optimal policy for $\mathcal{X}$ w.r.t. an order $r \in \mathcal{R}(n)$ is given by the same dynamic program, where instead of using the index set $[n]$ we apply the ordering function r on the index set.

Definition 3 (Optimal reward w.r.t. chosen ordering r)**.** For a sequence of random variables $\mathcal{X} := X_1, ..., X_n$ we define its value as

$$V(r, \mathcal{X}) = \sup\{\mathbb{E}[X_{r[t]}] : t \text{ is a stopping rule}\}$$

$$V(r, \mathcal{X}) = \begin{cases} \mathbb{E}\left[\max\left(X_{r(x_1,...,x_{j-1})[j]}, V(r, \{X_1, ..., X_{j-1}\})\right)\right] & \mathcal{X} = \{X_1, ..., X_j\} \\ 0 & \mathcal{X} = \emptyset \end{cases}$$

where $\mathcal{T} = [n]$ denotes the set of all stopping locations for the sequence $X_1, X_2, ..., X_n$. Let $\mathcal{R}(n)$ denote the universe of all possible online orderings $r : \mathbb{R}^n \to \mathcal{P}(n)$. According to definition 2, we design the following dynamic program which gives us the optimal policy:

$$\begin{aligned} V(r,j) &= \mathbb{E}[\max(X_{r(X_1,...,X_{j-1})[j]}, V(r,j+1))] \\ V(r,n+1) &= 0 \end{aligned}$$

The optimal stopping policy as described by the dynamic program is written verbally as follows:

Optimal Policy (Given ordering r). At time step t, compare the value of random variable $X_{r,t} := X_{r(X_{r[1]},...,X_{r[t-1]})[t]}$ to $V(r,t+1)$. If the realized value of $X_{r,t} \geq V(r,t+1)$ then stop and accept the reward. Otherwise continue.

We would like to show that a precomputed, offline static ordering is just as good as, if not better than, a dynamic online ordering. Specifically, we prove the following theorem:

Theorem 4.

$$\sup_{r \in \mathcal{R}(n)} V(r, \mathcal{X}) = \max_{\pi \in \mathcal{P}(n)} V(\pi, \mathcal{X}).$$

Proof. Via forward induction on n and backward induction for fixed n. For the base case $n = 1$, the statement is trivially true: there is only one item which forces the same ordering on the LHS and RHS. Assume the statement holds for $1 \leq k \leq n-1$, and let $r \in \mathcal{R}(n)$ be arbitrary. Without loss of generality we may assume the following

A1. $r[1] \mapsto 1$, since otherwise we can permute $\mathcal{X}$ in the beginning such that it holds. And

A2. $V(X_2, ..., X_n) = \max_{\pi \in \mathcal{P}(n); \pi[1]=1} V(X_{\pi[2]}, ..., X_{\pi[n]})$

Since $\mathcal{P}(n)$ may be viewed as a subset of $\mathcal{R}(n)$ it suffices to show that $V(X_1, ..., X_n)$ upper bounds the LHS, i.e. for any choice of r,

$$V(r, \mathcal{X}) \leq V(X_1, ..., X_n)$$

To help our proof, we first state a lemma from [Hill], which is a natural consequence from the definition of V, X, and r.

Lemma 5 (Lemma 3.5 of [7])**.** *If* $r[1] = 1$ *and let* $X_1 \sim \mathcal{D}_1$ *then*

$$V(X_{r[2]}, ..., X_{r[n]} | X_1 = x) = V(X_{r(x)[2]}, ..., X_{r(x)[n]})$$

Let $\mathcal{D}_1$ denote the distribution of X_1, and let $A := \{x \in \mathbb{R}; x > V(X_{r[2]}, ..., X_{r[n]}; X_1 = x)\}$ and $\bar{A} = \mathbb{R} \backslash A$. By backward induction and the first lemma above, it follows that

$$\begin{aligned} V(r, \mathcal{X}) &= \max(X_{r[1]}, V(r, \{X_{r[2]}, ..., X_{[n]}\} | X_{r[1]})) \text{ by backwards induction} \\ &= \max(X_{r[1]}, V(r, \{X_{r(X_{r[1]})[2]}, ..., X_{[n]}\})) \text{ by Lemma 3.5 of Hill} \\ &\leq \max(X_1, V(X_2, ..., X_n)) \text{ by assumption A2} \\ &= V(\{X_1, ..., X_n\}) \text{ by definition of } V(\cdot) \\ &= V(\mathcal{X}) \end{aligned}$$

□

It is necessary that the theorem above requires independence as an assumption: Take the following example from [7], for instance. Let X_1, X_2, X_3 be drawn from a joint distribution and take values $(0,3,2), (0,3,4), (1,2,3), (1,4,3)$ each with uniform probability. One computes that $V(Y_\pi) = \frac{13}{4}$ for all offline orderings but for an online ordering $r \in \mathcal{R}(3)$ we can set $r[1] = 1$, $r(0, x_2, x_3)[2] = 3$, $r(1, x_2, x_3)[2] = 2$. Then $V(r, \{X_1, X_2, X_3\}) = \frac{14}{4} \leq V(\{X_1, X_2, X_3\})$.

3.2 NP-hardness of computing an optimal ordering.

Since we showed previously that a predetermined offline ordering is just as good as a dynamically determined online ordering, our goal will be to choose an optimal permutation $\pi^* \in \mathcal{P}(n)$ such that for any sequence of n independent random variable arrivals $\mathcal{X} = (X_1, ..., X_n) \sim (\mathcal{D}_1, ..., \mathcal{D}_n)$, $\pi^* = \arg\max_{\pi \in \mathcal{P}(n)} (V(\pi, \mathcal{X}))$. We show that this problem is NP-hard, by first analyzing this problem in the more restricted context where each $\mathcal{D}_i$ is a k-point distribution (as defined below). We will then show that the even case when every $\mathcal{D}_i$ is a 3-point distribution is NP-hard, via a reduction from a problem called subset product to optimal ordering for 3-point distributions; hence showing the general problem of optimal ordering is NP-hard. To get started, we first define the subset product problem, which is an analogue of subset sum but with multiplication, and show that it is NP-complete.

Problem 6 (Subset product)**.** In the subset product problem, we are given a set $A := \{a_1, ..., a_n\} \subset \mathbb{Z}$ (with $a_i > 1$ for all i), and a target B. The subset product question asks if there exists a set $T \subset A$ such that $\prod_{t \in T} t = B$.

Proof that subset product is NP-Complete. We first recal the following problem, X3C (Exact Cover by 3-Sets), which is known to be NP-Complete:

Problem 7 (X3C)**.** Given a finite set X with $|X|$ a multiple of 3, and a collection of 3-element subsets C of X, decide if C contains an exact cover C' for X such that $C' \subseteq C$ and every element in X occurs exactly once in C'.

To reduce an X3C instance to a subset product instance, we do the following:

1. Establish a bijective map f between the members of X and the first $|X|$ prime numbers. WLOG, we will henceforth denote the members of X and the subsets of X in C with the prime numbers by their mapped prime numbers.

2. For each subset c in C we multiply its mapped prime elements together to form a list of numbers that are prime products L. We feed this list into the subset product instance, and multiply every (mapped prime) element in X together to form the value B in the subset product instance.

3. Because prime numbers are used for the mapping, two subsets are equal if and only if their prime products are equal, as guaranteed by the unique factorization theorem.

The prime factors of B are exactly the elements in X, and the prime factors of the number in L correspond to the elements of the subsets in C. Hence, any solution to the subset product instance gives a solution to exact cover by 3 sets. All operations used above are polytime in $|X|$, hence this is a polytime reduction.

Definition 8 (k-point distributions)**.** A random variable X_i drawn from a k-point distribution is defined by $2k-1$ parameters $\{p_i^{(1)}, ..., p_i^{(k)}, q_i^{(1)}, ..., q_i^{(k-1)}\}$ where $X_i \in \{p_i^{(1)}, ..., p_i^{(k)}\}$ and $\Pr(X_i = p_i^{(j)}\} = \begin{cases} q_i^{(j-1)} & j \neq 1 \\ 1 - \sum_{l=1}^{k-1} q_i^{(l)} & j = 1 \end{cases}$. We call the set $\{p_i^{(1)}, ..., p_i^{(k)}\}$ the *support* of the k-point distribution.

We will start by stating some properties of the optimal ordering for 3-point distributions, as proved in [2]. First, for a 3-point distribution, let its support be $\{0, m_i, 1\}$ where $m_i \in (0,1)$. Let the 3-point distribution be defined by $q_i = \Pr(X_i^{(3)} = 1)$ and $p_i = \Pr(X_i^{(3)} = m_i)$ and $\Pr(X_i^{(3)} = 0) = 1 - (p_i + q_i)$, for $0 < q_i, p_i < 1$ and $q_i + p_i < 1$. Hence, every element in the support can be realized with some positive probability, and no element is realized, or never realized, almost surely.

Observe at step i in our optimal policy given an ordering π, either $V(\pi, i+1) > m_{\pi[i]}$ meaning that $X_{\pi[i]}$ attains value 1, and is accepted; or else $X_{\pi(i)}$ fails to attain 1. We may hence partition the input sequence $\mathcal{X} := (X_1, ..., X_n)$ into two sets: $S^\pi := \{X_{\pi[i]}; V_{\pi(i+1)} > m_{\pi(i)}\}$ and $T^\pi = \mathcal{X} \backslash S^\pi$.

Proposition 9 (Claim 3.1 of [2])**.** *In an arbitrary optimal ordering $\pi \in \mathcal{P}(n)$, the following holds:*

1. *Elements in S^π appear before elements in T^π.*
2. *Elements in S^π are ordered arbitrarily in π.*
3. *Elements in T^π are ordered in nonincreasing order of E_i where $E_i := \mathbb{E}[X_i | X > 0] = \frac{m_i p_i + q_i}{p_i + q_i}$.*

We can now define a closely related problem called optimal partitioning. We call a sequence $(S, T) = \mathcal{X} = (X_1, ..., X_n)$ of n random variables and ordered partition if (1) T is nonempty, (2) variables in S are ordered arbitrarily and (3) variables in T are arranged in nonincreasing order according to E_i. In the optimal partitioning problem, we seek to find an ordered partition (S, T) such that $V(S, T)$ is maximized. As a corollary of Claim 3.1 of [2], in the case of 3-point distributions, optimal partitioning is equivalent to optimal ordering.

We now prove the following theorem:

Theorem 10. *Optimal ordering is NP-hard in the case where $X_i \sim \mathcal{D}_i$ where $\mathcal{D}_i$ is a 3-point distribution with support $\{0, m_i, 1\}$ for some $m_i \in (0,1)$, and corresponding probabilities $\{1 - p_i - q_i, p_i, q_i\}$ obeying $0 < p_i, q_i < 1$ and $p_i + q_i < 1$ for all $i \in [n]$.*

Proof. We assume that an instance of subset product is given as a set $A := \{a_1, ..., a_n\} \subset \mathbb{Z}$ (with $a_i > 1$ for all i), and a target B. Recall the subset product question asks if there exists a set $T \subset A$ such that $\prod_{t \in T} t = B$. We consider the following valuation of random variables: Let $X_i = 0$ w.p. $\frac{1}{a_i^2}$, $X_i = m_i = \frac{B^2 - a_i}{B^2 + 1}$ w.p. $\frac{a_i - 1}{a_i^2}$, and $X_i = 1$ w.p. $\frac{a_i - 1}{a_i}$. In addition

$$E_i = \mathbb{E}[X_i | X_i > 0] = \frac{\frac{B^2 - a_i}{B^2+1} \cdot \frac{a_i - 1}{a_i^2} + \frac{a_i - 1}{a_i}}{1 - \frac{1}{a_i^2}}$$
$$= \frac{B^2}{B^2 + 1}$$

which means $E_i = E_j$ for all i, j since E_i does not depend on i hence the order within S^π and T^π are irrelevant. Let i_S and i_T denote an ordered partition of the index set $\{1, 2, ..., n\}$ that correspond to the ordered partition (S^π, T^π). Then the expected reward $V(S^\pi, T^\pi)$ is written according to definition as

$$
\begin{aligned}
V(S^\pi, T^\pi) &= 1 - \prod_{i \in i_S}(1 - q_i) + \frac{B^2}{B^2+1}\left(\prod_{i \in i_S} 1 - q_i\right)\left(1 - \prod_{j \in i_T}(1 - p_j - q_j)\right) \\
&= 1 - \prod_{i \in i_S}[1 - (1 - \frac{1}{a_i})] + \frac{B^2}{B^2+1}\left(\prod_{i \in i_S}[1 - (1 - \frac{1}{a_i})]\right)\left(1 - \prod_{j \in i_T}(1 - (1 - \frac{1}{a_i}) - (\frac{1}{a_i^2} - \frac{1}{a_i}))\right) \\
&= 1 - \prod_{i \in i_S}\frac{1}{a_i} + \frac{B^2}{B^2+1}\left(\prod_{i \in i_S}\frac{1}{a_i}\right)\left(1 - \prod_{j \in i_T}\frac{1}{a_i^2}\right)
\end{aligned}
$$

If we let $\gamma = \prod_{i=1}^n a_i$, $\gamma_S = \prod_{i \in i_S} a_i$ and $\gamma_T = \prod_{i \in i_T} a_i$ then $\frac{1}{\gamma_S} = \frac{\gamma_T}{\gamma}$ and

$$
V(S^\pi, T^\pi) = 1 - \frac{\gamma_T}{\gamma} + \frac{B^2}{B^2+1} \cdot \frac{\gamma_T}{\gamma}\left(1 - \frac{1}{\gamma_T^2}\right)
$$

Hence $V(S^\pi, T^\pi)$ may be written as a function of γ_T. We call this function $f(\gamma_T) = V(S^\pi, T^\pi)$. We claim that f is strictly concave in γ_T and achieves its maximum when $\gamma_T = B$; this can be shown using elementary calculus. Hence, the optimal partition (S^π, T^π) has $\gamma_T = B$ if and only if the given instance of subset product is feasible, i.e. a subset of S sums to B. Otherwise if the subset product instance is infeasible, the function $f(\gamma_T)$ will always attain some value strictly less than its maximum. $\square$

4 Random Order: i.i.d. Case

In the i.i.d. prophet inequality, the n random variables $(X_1, X_2, ..., X_n)$ are all sampled from the same distribution $\mathcal{D}$ independently identically at random.

Early approaches to the i.i.d prophet inequality are mostly mathematical, non-algorithmic approaches aimed at proving impossibility results instead of giving algorithms that lower bound the best achievable gambler-to-prophet ratio. The pioneering work by Hill and Kertz [8] showed that the gambler can obtain at least $1 - \frac{1}{e}$ of the prophet. They also constructed a sequence of samples α_n, for the which no algorithm can obtain an approximation ratio better than $\beta = \frac{1}{1.342} \approx 0.745$, where β is governed by the Kertz Equation, given as follows.

$$
\int_0^1 [(\beta^{-1} - 1)] - y(\log y - 1)]^{-1}\, dy = 1 \tag{1}
$$

The proof for this upper bound utilizes conjugate duality theory, probabilistic convexity arguments, and functional equation analysis, and is hence omitted. We refer the readers to [8] and [9] for details. Both problems of beating the $1 - \frac{1}{e}$ approximation and trying to achieve a β-approximation remained open for 3 decades until 2017 when Abolhassani et al.

[1] and Correa et al. [4] beat the $1 - \frac{1}{e}$ approximation. [1] give a 0.738 approximation when n is larger than a chosen large constant n_0. Simultaneously and independently, [4] show a 0.745 approximation for this problem, thereby completely closing the gap between upper and lower bounds for the i.i.d. case. We first present the algorithm due to Abolhassani et. al, and then discuss the work of Correa et. al.

4.1 A 0.738 Approximation

In order to achieve a 0.738 Approximation, we consider Algorithm 1 from [1]. We will focus on providing some intuition as to why this algorithm works. For a given sequence of thresholds,

Algorithm 1

Input: n i.i.d items with distribution function $\mathcal{D}$

1. Set a to 1.306 (root of $\cos(a) - \sin(a)/(a-1)$)
2. Set $\theta_i = \mathcal{D}^{-1}(\cos(ai/n)/\cos(a(i-1)/n))$
3. Pick the first item i for which $X_i \geq \theta_i$

let $q_0, q_1 ... q_n$ denote the probability of the algorithm not choosing any of the first items. More specifically, let $q_i = \Pr[\theta > i]$ for every $0 \leq i \leq n$. Knowing the thresholds $\theta_1, ...\theta_n$one can find this sequence by starting from $q_0 = 1$ and computing the rest using $q_i = q_{i-1}\mathcal{D}(\theta_i)$ Inversely, one can simply find the thresholds from $q_0, q_1 ... q_n$ using $\theta_i = \mathcal{D}^{-1}(q-i/q_{i-1})$ Hence, the design of the algorithm focuses on finding the sequence $q_1, .., q_n$. To this end, we need to find a continuous function $h : [0, 1] \to [0, 1]$ with $h(0)$= 1 such that by setting $q_i = h(i/n)$ we can achieve our desired set of thresholds. In order for Algorithm 1 to work, h needs to be $\alpha-$strong.

Definition 11. A function h is $\alpha-$strong is it has the following properties:

1. $h(1) \leq 1 - \alpha$
2. $\int_0^1 h(r)dr \geq \alpha$
3. $\forall\ 0 \leq s \leq 1 : 1 - h(s) - h'(s)/h(s) \int_0^1 h(r)dr \geq \alpha(1 - \exp(h'(s)/h(s))$

Theorem 12. *If h is an α-strong function, then for every $\epsilon > 0$ there exists an n_ϵ such that for every $n \geq n_\epsilon$ the threshold algorithm that acts based on h is at least $(1-\epsilon)\alpha$-approximation on n iid items*

Proof. We now give a sketch of a proof presented by Abolhassani et al. We can write the expected reward acquired by the optimal stopping time algorithm OPT (i.e. the prophet) as

$$\mathbb{E}[\mathsf{OPT}] = \int_0^\infty \Pr[\max X_i \geq x]dx \tag{2}$$

We can, similarly, write the expected reward of our approximation algorithm ALG as

$$\mathbb{E}[\mathsf{ALG}] = \int_0^\infty \Pr[\max X_{\theta_i} \geq x]dx \tag{3}$$

The proof proceeds by showing that, for every ϵ there exists some n_ϵ such that for every $n \geq n_\epsilon$, the second integrand is at $(1-\epsilon)\alpha$ times the first integrand and this proves the theorem. We get the 0.738 approximation by showing $\cos(as)$ is $\approx 0.7388-$ strong. $\square$

4.2 Achieving Optimality

Now, we turn to the work of [4] which provides a 0.745 approximation for this problem. This matches the upper bound given by [9], closing the gap between the lower and upper bounds. It is interesting to note that [4] was originally considering this problem in the context of posted price mechanisms. In particular, they give an adaptive posted price mechanism that achieves 0.745 of the Myerson Optimal Auction, which simultaneously yields an algorithm for random-order prophet inequalities. The intuition behind the algorithm is that in an auction, as fewer customers are left, the price should decrease. The second key insight is to draw the threshold from a well-chosen distribution that mimics the optimal scheme. Below, we only consider their algorithm modified for the i.i.d prophet inequality setting.

Algorithm 2

Input: n i.i.d items with distribution function $\mathcal{D}$

1. Partition the interval $[0,1]$ into intervals $A_i = [\epsilon_{i-1}, \epsilon_i]$, s.t $0 < \epsilon_0 < \epsilon_1 ... < \epsilon_n < 1$
2. Draw q_i from interval A_i according to the probability density, $f_i(q) = \frac{(n-1)(1-q_i)^{n-2}}{\alpha_i}$, where α_i is a normalization parameter computed by integration of the numerator.
3. Set threshold $\theta_i = \mathcal{D}^{-1}(1-q_i)$ for the r.v at step i

---

Theorem 13. *Given n non-negative i.i.d. random variables, $X_1, ..., X_n$, with $X_i \sim X$ for all i, there exist thresholds, $\theta_1, .., \theta_n$, such that for a sequence σ, drawn uniformly at random, the expected value of the first variable that exceeds its threshold according to that sequence, $X_{\sigma(i)} \geq \theta_i$, is at least a $1/\beta$ fraction of the expected value of* $\max\{X_1, ..., X_n\}$.

Proof. We give a brief sketch of the proof of this theorem, omitting many details and computations. When $\mathcal{D}$ is continuous and strictly increasing, we express the expected value of the maximum as follows:

$$\mathbb{E}(\max X_i) = n \int_0^1 t\mathcal{D}^{n-1} d(\mathcal{D}(t) = n \int_0^q (n-1)(1-q)^{n-2} R(q) q dq$$

where t is the solution to a differential equation similar to Kertz equation and $R(q) = \mathbb{E}(X | X > \mathcal{D}^{-1}(1-q))$ is the expected value of a random variable given that the probability that that variable attains a larger value is at most q. We once again omit the details about the differential equation. Since $R(q)$ corresponds to the expected value we get when setting the threshold $\mathcal{D}^{-1}(1-q)$, one can prove the expected revenue of this strategy is equal to

$$\sum_{i=1}^{n} \rho_i \int_{\epsilon_{i-1}}^{\epsilon_i} (n-1)(1-q)^{n-2} R(q) q dq$$

where $\rho_i = \frac{1}{\alpha_1}$ and $\rho_{i+1} = \frac{\rho_i}{\alpha_i}\int_0^q (n-1)(1-q)^{n-1}dq$. Setting ϵ_i such that $\rho_i = ...\rho_n$, gives that

$$\sum_{i=1}^{n} \rho_i \int_{\epsilon_{i-1}}^{\epsilon_i} (n-1)(1-q)^{n-2}R(q)qdq \geq \frac{1}{\beta}\mathbb{E}(\max X_i)$$

where $\beta \approx 1.342$ $\square$

Both [1] and [4] present a strategy that uses adaptive thresholds. Both papers, define ideal properties for threshold functions to obtain a good ratio and work towards finding such function. It is interesting that in this problem, the lower bound is given by an algorithmic construction, and the upper bound is given by a theoretical construction, as usually in such types of problems the reverse is true. Since [4] were motivated by a different question, they were able to build on a different line of work and apply their result to the i.i.d prophet inequality setting. Resolving the i.i.d prophet inequalities is a huge success for algorithm design.

5 Random Order: Non i.i.d. Case

In the non i.i.d. case, the random variables $X_1, ..., X_n$ draw from n distributions $\mathcal{D}_1, ..., \mathcal{D}_n$ that are independent but not necessarily distinct; their arrival ordering is then permuted uniformly at random. [5] recently proved that the optimal approximation is bounded above by 0.732 and bounded below by 0.669. In this section we outline an algorithm which guarantees the gambler a factor of roughly $1 - \frac{1}{e} + \frac{1}{27}$ of the optimal reward. At the end, we provide an example which shows that no algorithm achieves the gambler better than a factor of $\sqrt{3} - 1$ of the optimal reward. The work by Correa is an expansion of other recent works by [6] and [3].

5.1 Time Threshold Algorithm

Recall the statement of the prophet secretary problem. We aim to find the largest possible constant c, where $\max\{\mathbb{E}(X_{\sigma(T)}|T \in T_n\} \geq c\mathbb{E}(\max X_i)$, where T_n is the set of stopping times. In the paper developed by [6], which inspired [5], a single threshold strategy is implemented. They choose their threshold τ, such that $\Pr(max\{\mathbb{E}(X_i)\} \leq \tau) = \frac{1}{e}$, which then allows for an algorithmic performance of $1 - \frac{1}{e}$.
For this paper, we consider multiple threshold strategies, denoted as blind strategies, where the thresholds are determined by a non-increasing continuous function. Specifically, we take our non-increasing function α:[0, 1] $\rightarrow$ [0, 1], and we select u_i from [0, 1] uniformly and independently, and then select thresholds τ_i where $\Pr(\max_{i\in[n]}\{\mathbb{E}(X_i)\} \leq \tau_i) = \alpha(min_{i\in[n]}\{u_i\})$. In implementing this algorithm, the gambler stops at the first time they encounter a value which exceeds a threshold. The formalization of this algorithm is listed below. For clarity's sake, please note that X_{σ_i} is the i^{th} randomly chosen variable.

Note that the above algorithm gives an expected payoff of $X_{\sigma(T)}$, where T is the stopping time of the function. Also, from here forward please consider $\mathcal{D}_i$ to be the distribution corresponding to the variable X_i.

Algorithm 3

Time Threshold Algorithm (TTA)

1. For each time instance $i = 1, 2, ...n$ check if X_{σ_i} is greater than the selected threshold τ_i.
2. If X_{σ_i} is greater than the selected threshold τ_i, then we select the variable X_{σ_i}. Otherwise, we continue the algorithm from step 1.

5.2 Existence of a Suitable Threshold Function

First, we aim to show that there exists a non-increasing function α:[0, 1] $\rightarrow$ [0, 1] that meets our desired properties, as stated in the theorem statement below.

Theorem 14. *There exists a non-increasing function α: $[0, 1] \rightarrow [0, 1]$ such that*

$$\mathbb{E}(X_T) \geq 0.665\mathbb{E}(\max_{i\in[n]}\{X_i\})$$

where T is the stopping time of the blind strategy α.

In order to show the existence of this function, we must first expand our consideration of strategies to deterministic blind strategies, which are essentially the limits of blind strategies when n tends to infinity. We formally define non-deterministic blind strategies as follows.

Definition 15 (Deterministic blind strategy)**.** For a sequence of random variables X_i, thresholds τ_i, and a non-increasing function $\alpha : [0, 1] \rightarrow [0, 1]$, a deterministic blind strategy is one that obeys the following condition:

$$\forall j \in [n] : \Pr(\max_{i\in[n]}\{X_i\}) = \alpha(\frac{j}{n})$$

We can convert a deterministic blind strategy into a blind strategy as follows. Consider the sequence $\mathcal{D}_1$, ..., $\mathcal{D}_n$ of distributions corresponding to the n variables X_i. We then add to this distribution an additional m deterministic variables, X_{n+i} for $i \leq m$, and set them equal to 0 with probability 1. In other words, the variables X_{n+i} for $i \leq m$ have distributions $\mathcal{D}_{n+i} = 0$, with probability 1. Letting T_m denote the stopping time here, it is easy to see that $\lim_{m\rightarrow\infty} \mathbb{E}(X_{\sigma(T_m)}) = \mathbb{E}(X_{\sigma(T)})$. It can be seen that the blind strategy of choosing the random variables $u_1, ...u_n$ uniformly from $\{1, ..., \frac{1}{m+n}\}$ instead of from $[0, 1]$ is analogous to the deterministic blind strategy proposed above, when m $\rightarrow \infty$.

We next show some properties of deterministic blind strategies from [6]. They follow from standard computations.

Lemma 7. Given independent random variables $\mathcal{X} = (X_1, ..., X_i, ..., X_n)$ and non-increasing thresholds τ_i with stopping time T given by the TTA, we then have, for all j $\in$ [n+1] and t $\in$ [τ_j, τ_{j-1}):

$$\Pr(X_{\sigma(T)} > t) = \Pr(T \leq j - 1) + \sum_{i\in[n]} \Pr(X_i > t)(\sum_{k>j-1}^{n} \frac{\Pr(T \geq k|\sigma_k = i)}{n})$$

This proof follows from the fact that $\Pr(X_{\sigma(T)} > t) = \Pr(T \leq j-1) + \Pr(X_{\sigma(T)} > t, T \geq j)$ because thresholds are non-increasing. From standard computation using independence, we have $\Pr(X_{\sigma(T)} > t, T \geq j) = \sum_{i \in [n]} \Pr(X_i > t)(\sum_{k>j-1}^{n} \Pr(\sigma_k = i, T \geq k)) = \sum_{i \in [n]} \Pr(X_i > t)(\sum_{k>j-1}^{n} \frac{\Pr(T \geq k | \sigma_k = i)}{n})$.

Lower bounding the last term by using $\Pr(T \geq k | \sigma_k = i) \geq \Pr(T > k)$. we have, $\Pr(X_{\sigma(T)} > t) \geq \sum_{i \in [n]} \Pr(X_i > t)(\sum_{k>j-1}^{n} \frac{\Pr(T>k)}{n})$. Taking the j $\in$ [n+1], which minimizes this, and noting that $1 - \alpha(\frac{j}{n}) = \Pr(\max_{i \in [n]}\{X_i\} > \tau_k) \geq \Pr(\max_{i \in [n]}\{X_i\} > t)$, the result of Lemma 8, listed below, holds.

Lemma 8. Under the conditions of Lemma 7, for $j \in [n+1]$, $t > 0$, and deterministic blind stopping time T where α is non-increasing with $\alpha(\frac{n+1}{n}) = 0$:

$$\Pr(X_{\sigma(T)} > t) \geq \min_{j \in [n+1]} \{\frac{\Pr(T \leq j-1)}{1 - \alpha(\frac{j}{n})} + \frac{1}{n} \sum_{k=j}^{n} \Pr(T > k)\} \Pr(\max_{i \in [n]}\{X_i\})$$

We then use the result of Lemma 8, to find a bound on $\Pr(T \leq k)$. Let α_i denote $\alpha(\frac{i}{n})$ for the remainder of the proof. We omit the proof of Lemma 9 here, however the majority of the argument comes down to utilizing the symmetric properties induced by the random ordering σ to bound $\Pr(T \geq k)$

Lemma 9. Choose and fix $\alpha_1, ..., \alpha_n \in [0, 1]$. For the distributions $\mathcal{D}_1, ..., \mathcal{D}_n$ of the variables X_i, choose thresholds τ_i where $\Pr(\max_{i \in [n]}\{X_i\} \leq \tau_i) = \alpha_i$. We then have for all $k \in [n]$:

$$\frac{1}{n} \sum_{j \in [k]} 1 - \alpha_j \leq \Pr(T \leq k) \leq 1 - (\prod_{j=1}^{k} \alpha_j)^{\frac{1}{n}}$$

Using Lemmas 8 and 9, and selecting the values α_i from Lemma 9 to fit a specific non-increasing function α achieves the desired result of Lemma 10, listed below.

Lemma 10. Let α:[0, 1] $\to$ [0, 1] be a continuous non-increasing function and stopping time T for a deterministic blind stopping time. For X_i with distributions F_i, for any $t > 0$ we have:

$$\Pr(X_{\sigma(T)} > t) \geq \min_{j \in [n+1]} \{f_j(\alpha)\} \Pr(\max_{i \in [n]}\{X_i\})$$

where for all $j \in [n+1]$:

$$f_j(\alpha) = \sum_{k=1}^{j-1} \frac{1 - \alpha(\frac{k}{n})}{n(1 - \alpha(\frac{j}{n})} + \frac{1}{n} \sum_{k=j}^{n} (\prod_{l=1}^{k} \alpha(\frac{l}{n}))^{\frac{1}{n}}$$

Using a Riemann-sum analysis to approximate the expressions for $f_j(\alpha)$, and numerically minimizing these sums yields a minimal value of 0.665 as $n \to \infty$. Recalling that as $n \to \infty$, the deterministic blind strategies approximate the blind strategy, it is evident that the lower bound is achieved. This finally proves Theorem 5.

In order to improve the lower bound to 0.669, as opposed to 0.665 above, [5] proved the sharper inequality listed below and used this, inspired by a paper by Esfandiari et al., in lieu of $\Pr(T \geq k|\sigma_k = i) \geq \Pr(T > k)$ in bounding the result of Lemma 7.

Lemma 11. Given independent random variables X_i and non-increasing thresholds τ_i with stopping time T, we then have, for all i, j $\in [n]$:

$$\Pr(T \geq k|\sigma_k = i) \geq \frac{\Pr(T > k)}{1 - \frac{k}{n} + \frac{1}{n}\sum_{l\in[k]} \Pr(X_i \leq \tau_l)}$$

In addition, as an alternative to $\sum_{i\in[n]} \Pr(X_i > t) \geq \Pr(\max_{i\in[n]}\{X_i\} > t)$ in the proof of Theorem 5, to instead achieve the bound of 0.669, the following lemma is also used.

Lemma 12. Given independent random variables X_i and non-increasing thresholds τ_i, we then have, for all $t < \tau_1$, $k \leq \frac{n}{2}$:

$$\sum_{i\in[n]} \frac{\Pr(X_i > t)}{1 - \frac{1}{n}\sum_{l\in[k]} \Pr(X_i \leq \tau_l)} \geq \frac{\Pr(\max_{i\in[n]}\{X_i\} > t)}{1 - \frac{k}{n}\sum_{l\in[k]} \Pr(\max_{i\in[n]}\{X_i\} \leq \tau_1)}$$

Using these following two optimizations, leads the proof in a similar direction. The final steps, however, require solving an integral formulated as a Mayer optimal control problem, and so we omit the details of this proof here.

5.3 Example Showing the Upper Bound

In this section, we provide a simple example which shows that has an upper bound of $\sqrt{3}-1$ performance ratio for any strategy. We choose a $\in$ [0, 1] and we consider n+1 randomly distributed variables where

$$X_i \sim \begin{cases} \frac{1}{n} & p = \frac{1}{n^2} \\ 0 & p = 1 - \frac{1}{n^2} \end{cases}$$

for i < n, and X_{n+1} is identically a. This gives an expectation of $\mathbb{E}(X_{\sigma(T)}|\sigma(i) = n+1)$ of $n[1-(1-\frac{1}{n^2})^n]$, where a is not considered (at times n or earlier), and $\mathbb{E}(X_{\sigma(T)}|\sigma(i) = n+1)$ $= n[1-(1-\frac{1}{n^2})^{i-1}] + (1-\frac{1}{n^2})^{i-1}a$, when a is considered (at times n+1 or later). Summing over all possibilities of i, where $\sigma(i) = n+1$, we have $\mathbb{E}(X_{\sigma(T)}) = \frac{j-1}{n+1}n[1-(1-\frac{1}{n^2})^n] + \frac{1}{n+1}\sum_{i=j}^{n+1} n[1-(1-\frac{1}{n^2})^{i-1}] + (1-\frac{1}{n^2})^{i-1}a$. We get algebraically, that $\mathbb{E}(X_{\sigma(T)}) \leq 1 + \frac{a^2}{2} + O(\frac{1}{n})$.

Meanwhile, $\mathbb{E}(\max_{i\in[n]}\{X_i\}) = n[1-(1-\frac{1}{n^2})^n] + (1-\frac{1}{n^2})^n a$, which tends to $1+a$ as n tends to infinity. We then choose $a = \sqrt{3}-1$, and get an asymptotic upper bound as desired:

$$\lim_{n\to\infty} \sup \frac{\mathbb{E}(X_{\sigma(T)})}{\mathbb{E}(\max_{i\in[n]}\{X_i\})} \leq \frac{1 + a^2/2}{1+a} \approx \sqrt{3} - 1$$

References

[1] Abolhassani, M., Ehsani, S., Esfandiari, H., Hajiaghayi, M., Kleinberg, R. D., and Lucier, B. Beating 1-1/e for ordered prophets. In *Proceedings of 49th Annual ACM SIGACT Symposium on Theory of Computing (STOC)* (Montreal, QC, Canada, June 19-23 2017), Association for Computing Machinary, pp. 61–71.

[2] Agrawal, S., Sethuraman, J., and Zhang, X. On optimal ordering in the optimal stopping problem. *Proceedings of the 21st ACM Conference on Economics and Computation* (2020), 187–188.

[3] Azar, Y., Chiplunkar, A., and Kaplan, H. Prophet secretary: Surpassing the 1-1/e barrier. In *Proceedings of the 2018 ACM Conference on Economics and Computation* (Ithaca, NY, USA, June 18-22 2018), Association for Computing Machinary.

[4] Correa, J. R., Foncea, P., Hoeksma, R., Oosterwijk, T., and Vredeveld, T. Posted price mechanisms for a random stream of customers. In *Proceedings of the 2017 ACM Conference on Economics and Computation (EC)* (Cambridge, MA, USA, June 26-30 2017), Association for Computing Machinary, pp. 26–30.

[5] Correa, J. R., Saona, R., and Ziliotto, B. Prophet secretary through blind strategies. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)* (San Diego, California, USA, January 6-9 2019), Association for Computing Machinary, pp. 1946–1961.

[6] Ehsani, S., Hajiaghayi, M., Kesselheim, T., and Singla, S. Prophet secretary for combinatorial auctions and matroids. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)* (New Orleans, LA, USA, January 7-10 2018), Association for Computing Machinary, pp. 700–714.

[7] Hill, T. P. Prophet inequalities and order selection in optimal stopping problems. *Proceedings of American Mathematics Society 88*, 1 (1983), 131–137.

[8] Hill, T. P., and Kertz, R. P. Comparisons of stop rule and supremum expectations of iid random variables. *Annals of Probability 10*, 2 (1982), 336–345.

[9] Kertz, R. P. Stop rule and supremum expectations of iid random variables: a complete comparison by conjugate duality. *Journal of Multivariate Analysis 19*, 1 (1986), 88–112.

[10] Krengel, U., and Sucheston, L. On semiarts, amarts, and processes with finite value. *Adv. in Probab. 4* (1978), 197–266.

[11] Liu, A., Leme, R. P., Pal, M., Schneider, J., and Sivan, B. Variable decomposition for prophet inequalities and optimal ordering. *Proceedings of the 22nd ACM Conference on Economics and Computation (EC 2021)* (2021).

[12] Samuel-Cahn, E. Comparison of threshold stop rules and maximum for independent nonnegative random variables. *Annals of Probability 12*, 4 (1984), 1213–1216.