NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS

SCHOOL OF SCIENCES
DEPARTMENT OF INFORMATICS AND TELECOMMUNICATIONS

BSc THESIS

Prophet Inequality Problem: A Comprehensive Analysis of Research and Findings

Eleni M. Liarou

Supervisor: Christos Tzamos, Associate Professor

ATHENS

JANUARY 2024

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Το Πρόβλημα της Ανισότητας του Προφήτη: Μία Επισκόπηση της Έρευνας και των Αποτελεσμάτων

Ελένη Μ. Λιάρου

Επιβλέπων: **Χρήστος Τζάμος,** Αναπληρωτής καθηγητής

ΑΘΗΝΑ

ΙΑΝΟΥΑΡΙΟΣ 2024

BSc THESIS

Prophet Inequality Problem: A Comprehensive Analysis of Research and Findings

Eleni M. Liarou
S.N.: 1115201900100

SUPERVISOR: **Christos Tzamos,** Associate Professor

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Το Πρόβλημα της Ανισότητας του Προφήτη: Μία Επισκόπηση της Έρευνας και των Αποτελεσμάτων

Ελένη Μ. Λιάρου
Α.Μ.: 1115201900100

ΕΠΙΒΛΕΠΩΝ: **Χρήστος Τζάμος,** Αναπληρωτής καθηγητής

ABSTRACT

This thesis explores the Prophet Inequality Problem within the domains of Optimal Stopping Theory and Mechanism Design, aiming for a comprehensive analysis of its fundamental concepts and relevance in contemporary decision-making. Introduced by Krengel and Sucheston in the 1970s[30], the problem revolves around a gambler encountering a sequence of items drawn independently from known distributions. Upon the arrival of each item, its value is realized and the gambler either accepts it and the game ends, or irrevocably rejects it and continues to the next item. The goal is to maximize the value of the selected item and compete against the expected maximum value of all items. In this thesis, we first provide an overview of core concepts of optimal stopping theory, mechanism design, price mechanisms and various statistical tools and metrics necessary for algorithm assessment. Additionally, we present research findings related to both profit maximization and cost minimization and highlight the challenges of the latter. The exploration extends to various variations of the problem, such as matching, multiple choices, and matroid prophet inequalities. Notably, emphasis is placed on understanding the Prophet Secretary problem and scenarios involving variables drawn from unknown distributions.

SUBJECT AREA: Data Structures and Algorithms

KEYWORDS: Online algorithms, Prophet inequality, Mechanism design, Optimal stopping theory, Game theory

ΠΕΡΙΛΗΨΗ

Αυτή η πτυχιακή διερευνά το Πρόβλημα της Ανισότητας του Προφήτη στα πεδία της Θεωρίας Βέλτιστης Διακοπής και του Σχεδιασμού Μηχανισμών, στοχεύοντας σε μια ολοκληρωμένη ανάλυση των θεμελιωδών εννοιών και της επιρροής του στη σύγχρονη διαδικασία λήψης αποφάσεων. Το πρόβλημα παρουσιάστηκε από τους Krengel και Sucheston στη δεκαετία του 1970[30], και περιστρέφεται γύρω από έναν παίκτη που συναντά μια ακολουθία αντικειμένων με αξίες που προκύπτουν ανεξάρτητα από γνωστές κατανομές. Με την άφιξη του κάθε στοιχείου, η αξία του αποκαλύπτεται και ο παίκτης είτε το αποδέχεται και το παιχνίδι τελειώνει, είτε το απορρίπτει αμετάκλητα και συνεχίζει στο επόμενο αντικείμενο. Στόχος είναι η μεγιστοποίηση της αξίας του επιλεγμένου στοιχείου και ο ανταγωνισμός με την αναμενόμενη μέγιστη τιμή όλων των αντικειμένων. Στην παρούσα πτυχιακή παρέχουμε αρχικά μια επισκόπηση των βασικών εννοιών της βέλτιστης θεωρίας διακοπής, του σχεδιασμού μηχανισμών, των μηχανισμών τιμών και των διαφόρων στατιστικών εργαλείων και μετρικών που είναι απαραίτητα για την αξιολόγηση αλγορίθμων. Στη συνέχεια, παρουσιάζουμε τα ευρήματα που σχετίζονται με την μεγιστοποίηση του κέρδους, και την ελαχιστοποίηση του κόστους, επισημαίνοντας τις δυσκολίες αυτής. Η ανάλυση επεκτείνεται σε διάφορες παραλλαγές του προβλήματος, όπως ταίριασμα, πολλαπλές επιλογές και μητροειδής ανισότητα του προφήτη. Ειδικότερα, δόθηκε έμφαση στην κατανόηση του προβλήματος του προφήτη γραμματέα και της περίπτωσης όπου οι αξίες των αντικειμένων προέρχονται από άγνωστες κατανομές.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Δομές Δεδομένων και Αλγόριθμοι

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Online αλγόριθμοι, Ανισότητα του προφήτη, Σχεδιασμός μηχανισμού, Βέλτιστη θεωρία διακοπής, Θεωρία Παιγνίων

Στους γονείς μου

ACKNOWLEDGEMENTS

On an academic level, I would like to thank Associate Professor Christos Tzamos, my supervisor, for his unwavering patience, support, and invaluable guidance throughout the entire journey of completing this thesis.

On a more personal note, I want to express my heartfelt appreciation to my friends and family for their steadfast support from the very beginning of my academic endeavors.

CONTENTS

LIST OF FIGURES

LIST OF ALGORITHMS

PREFACE

This thesis was conducted as part of my undergraduate studies at the Department of Informatics and Telecommunications at the National and Kapodistrian University of Athens during the academic year 2023-2024.

1. INTRODUCTION

1.1 Motivation

The motivation for this thesis arises from recognizing the fundamental role prophet inequalities play in our daily decision-making. Even though at first glance they may appear like intricate mathematical problems with no immediate practical relevance, they underpin virtually every decision we encounter. Let's illustrate this with an example. Imagine you have finally gotten your bachelor's degree and you start preparing for graduate studies. You embark on a series of interviews to secure the best PhD position in terms of benefits, salary, and research prospects. After each interview, the benefits of the position are revealed, compelling you to make an immediate, irrevocable decision—accept the current offer or continue searching for a potentially better option. Ideally, you'd prefer knowing about offers from top-choice advisors before committing to a second-choice advisor, yet the sequence of these decisions might not align with your preferences. The prophet inequality problem arises when faced with an offer that's good but not the best encountered. This dilemma emerges: accept the offer immediately or persist in the hope of finding a superior one later. This scenario reflects the challenge of pinpointing the optimal stopping point in your search. Waiting too long risks missing the best opportunity, while accepting too soon risks missing a potentially better offer. This highlights the trade-off between immediate benefits and future, more favorable outcomes—a core aspect of the prophet inequality problem, not confined solely to PhD hunting but extending to various situations involving sequential decisions with limited information.

1.2 The purpose of this thesis

The following thesis studies the Prophet Inequality problem, first introduced by Krengel and Sucheston in the 1970s[30]. Although originating in Optimal Stopping Theory, these inequalities have become central in market design studies, particularly in relation to posted price mechanisms prevalent in online sales and mechanism design. The rapid growth of interest in the Prophet Inequality problem can be attributed to its relevance in optimizing decision-making processes across diverse fields like economics, computer science, and operations research. Furthermore , online auctions play a major role in modern markets. In online markets, information about customers and goods is revealed over time. Irrevocable decisions are made at certain discrete times, such as when a customer arrives to the market. One of the fundamental and basic tools to model this scenario is the prophet inequality and its variants. In today's era of Big Data, as companies and organizations are extremely reliant on data-driven decision-making, the Prophet Inequality problem offers a framework for making sequential decisions with uncertain information, making it particularly relevant in modern data-centric contexts.

The rest of the thesis is structured as follows:

- In Chapter 2, we lay the theoretical groundwork, that is necessary in order to understand research papers on the prophet inequality problem and its variations.
- Chapter 3 presents the findings from research papers that focus on profit maximization.

- In Chapter 4, we delve into the intricacies of the cost minimization problem, highlighting its distinct challenges compared to the profit maximization one. We also display the limited progress that has been made in recent years on this specific variation.
- In Chapter 5, we showcase several variations of the problem, with a particular focus on the Prophet Secretary and the setting where variables are drawn from unknown distributions.
- The last Chapter 6 contains the conclusion of this thesis and mentions open problems for future work.

2. PRELIMINARIES

2.1 Optimal Stopping Theory

2.1.1 What does optimal stopping theory entail?

The theory of optimal stopping addresses the problem of choosing a time to take a given action based on sequentially observed random variables to maximize an expected reward or minimize an expected cost. Problems of this nature can be found in the area of statistics, where the action may be to test a hypothesis, and in the area of operation research, where the action may be to buy a machine, hire a new employee, etc. It provides valuable insight into decision-making under uncertainty and has been used to develop strategies for hiring, investment, and other real-world situations, where sequential decision-making is involved.

2.1.2 Historical Overview

The theory of optimal stopping can be traced back to the early 20th century when mathematicians and statisticians inspired by statistical and decision theory began examining problems related to sequential decision-making. The concept of optimal stopping gained more recognition in the 1940s with the publication of Sequential Analysis by Wald in 1945 [37] and of Statistical Decision Functions [38] in 1950, which deal with the sequential analysis of statistical observations. The Bayesian perspective of this problem was explored in 1948 by Arrow, Blackwell and Girshick [5]. Snell extended the concept of sequential analysis to purely stopping problems without statistical structure in 1952. In the 1960's, the papers of Chow and Robbins gave impetus to a new interest and rapid growth of the subject. The book, Great Expectations: The Theory of Optimal Stopping by Chow, Robbins and Siegmund in 1971 [36], summarizes this development. In the 21st century, the optimal stopping theory has found applications in machine learning, artificial intelligence and data science, particularly in designing algorithms for online decision-making.

2.1.3 Definition of the Problem

Definition 2.1.1 [21] Stopping rule problems are defined by 2 objects:

1. a sequence of random Variables $X_1, X_2, \dots, X_n$, whose joined distribution is assumed to be known,
2. a sequence of real-valued reward functions, $y_0, y_1(x_1), y_2(x_1, x_2), \dots$.

Given these 2 objects, the associated stopping rule problem can be defind as follows. You may observe the sequence $X_1, X_2, \dots$ for as long as you wish. For each $n = 1, 2, \dots$, after observing $X_1 = x_1, X_2 = x_2, \dots, X_n = x_n$ you may stop and receive the known reward $y_n(x_1, \dots, x_n)$ or you may continue and observe X_{n+1}. If you decide to not take any observations you get the constant reward y_0. If you never stop you receive the reward $y_\infty(x_1, x_2, \dots)$.

Your goal is to choose a time to stop to maximize the expected profit. You are allowed to use randomized decisions. That is, given that you reach stage n having observed $X_1 =$

$x_1, X_2 = x_2, \ldots, X_n = x_n$, you are to choose a probability of stopping that may depend on these observations. We denote this probability by $\phi_n(x_1, x_2, \ldots, x_n)$. A randomized stopping rule consists of the sequence of these functions, $\phi = (\phi_0, \phi_1(x_1), \phi_2(x_1, x_2), \ldots)$, where for all n and $x_1, x_2, \ldots,$ $0 \leq \phi_n(x_1, x_2, \ldots, x_n) \leq 1$. The stopping rule is said to be non-randomized if each $\phi_n(x_1, x_2, \ldots, x_n)$ is either 0 or 1.

Therefore, ϕ_0 represents the probability that you take no observation at all. Given that you take the first observation and given that you observe $X_1 = x_1$, $\phi_1(x_1)$ represents the probability you stop after the first observation, and so on. The stopping rule ϕ and the sequence of observations $X = (X_1, X_2, \ldots)$ determined the random time N at which stopping occures, $0 \leq N \leq \infty$, where if stopping never occurs then $N = \infty$. The probability mass function of N given $X = x = (x_1, x_2, \ldots)$ is denoted by $\psi = (\psi_0, \psi_1, \ldots, \psi_\infty)$ where $\psi_n(x_1, \ldots, x_n) = P(N = n|X = x) for\ n = 0, 1, 2, \ldots,$ $\psi_\infty(x_1, x_2, \ldots) = P(N = \infty|X = x)$.

This may be related to the stopping rule ϕ as follows:

$\psi_0 = \phi_0$

$\psi_1(x_1) = (1 - \phi_0)\phi_1(x_1)$

$\vdots$

$\psi_n(x_1, x_2, \ldots, x_n) = [\prod_1^{n-1}(1 - \phi_j(x_1, \ldots, x_j))]\phi_n(x_1, \ldots, x_n)$

$\vdots$

$\psi_\infty(x_1, x_2, \ldots) = 1 - \sum_0^\infty \psi_j(x_1, \ldots, x_j)$.

$\psi_\infty(x_1, x_2, \ldots)$ represents the probability of never stopping given all the observations.

Your problem then is to choose a stopping rule ϕ to maximize the expected return, $V(\phi)$, defined as

$V(\phi) = E_{yN}(X_1, \ldots, X_n) = E\sum_{j=0}^{=\infty} \psi_j(X_1, \ldots, X_j)\psi_j(X_1, \ldots, X_j)$, where the "$= \infty$" above the summation sign indicates that the summation is over values of j from 0 to ∞. In terms of the random stopping time N, the stopping rule ϕ may be expressed as

$\phi_n(X_1, X_2, \ldots, X_n) = P(N = n|N \geq n, X = x) for\ n = 0, 1, \ldots.$

Definition 2.1.1 [2] Optimal ordering for optimal stopping problem. For any given ordering σ of n random variables $X_1, \ldots, X_n$, let V_σ be the expected value at the optimal stopping time. We define the problem of optimal ordering for optimal stopping as the problem of choosing an ordering σ that maximizes V_σ i.e., the problem of finding $\sigma^* = argmax_\sigma V_\sigma$.

2.2 Game Theory vs Mechanism Design

2.2.1 What is Mechanism Design?

Mechanism design is a branch of microeconomics that explores how businesses and institutions can achieve desirable social or economic outcomes given the constraints of individuals' self-interest and incomplete information. Mechanism design takes private information and incentives into account to enhance economists' comprehension of market mechanisms and shows how the right incentives can induce participants to reveal their private information and create an optimal outcome. For example, in bargaining between a buyer and a seller, the seller would like to act as if the item is very costly thus raising the price, and the buyer would like to pretend to have a low value for the object to keep the price down. One question is whether one can design a mechanism through which the bargaining occurs (in this application, a bargaining protocol) to induce efficient trade of the good - so that successful trade occurs whenever the buyer's valuation exceeds that of the seller. Another question is whether there exists such a mechanism so that the buyer and seller voluntarily participate in the mechanism.

The Stanley Reiter Diagram in Figure 2.1 below illustrates a game of mechanism design. It consists of agents who interact to produce outcomes according to a social choice function f(Θ). We define mechanism (M, g) consisting of a set of messages M_i for each agent and an outcome function g: $\prod_i M_i \to X$, that assigns an outcome for each profile of messages received from agents. To accommodate various rules, agents can submit programs to act as their proxies. Messages represent strategies we're making available to the agent, which are then translated into outcomes by g.

In the mechanism, agents select the message $m_i(\theta_i)$ they will send based on their individual types (θ), resulting in an overall outcome $g(m_1(\theta_1), \ldots, m_n(\theta_n))$. The mechanism (M, g) implements a social choice function f if, for all profiles of types θ, the outcome we get under the mechanism equals the outcome we want: $g(m_1(\theta_1), \ldots, m_n(\theta_n)) = f(\theta_1, \ldots, \theta_n)$, for all profiles $(\theta_1, \ldots, \theta_n) \in \Theta = \prod_i \Theta_i$.

In simpler words we want the strategies (determined by whatever behavioral theory we have for each agent) to compose with the outcome function (which we are free to choose, up to design constraints) to match up with our goal, as shown in the following diagram:

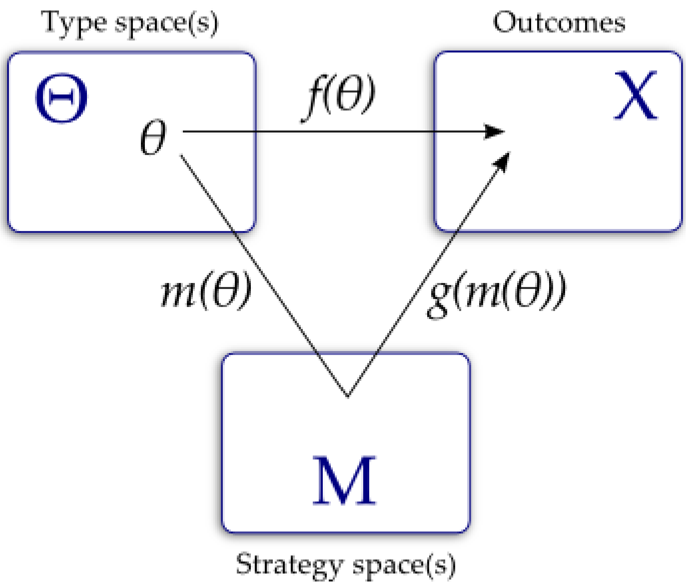

Figure 2.1: Stanley Reiter Diagram

A social choice function f is implementable if some mechanism exists that implements it. Whether a social choice function is implementable depends on our behavioral theory. If we think agents choose strategies in Nash equilibrium with each other, we'll have more flexiblity in finding a mechanism than if agents need the stronger incentive of a dominant strategy, since more Nash equilibria exist than dominant-strategy equilibria. Rather than assuming agents choose strategically based on their preferences, perhaps we think agents are naively honest. In this case, we can trivially implement a social choice rule by having each agent tell us their full type and simply choosing the corresponding goal by picking $M_i = \Theta_i$ and $g = f$. Here the interesting question is instead which mechanism can implement f with the minimal amount of communication, either by minimizing the number of dimensions or bits in each message.

2.2.2 What is Game Theory?

Game theory is a mathematical and interdisciplinary framework used to study strategic interactions and decision-making among rational individuals or entities, often referred to as ”players” or ”agents.” It provides a formal structure for analyzing situations where the outcome of an individual’s decision depends on the decisions made by others. Moreover, game theory helps researchers and decision-makers understand and predict behavior in strategic situations and provides insights into optimal decision-making strategies.

2.2.3 How are Mechanism Design and Game Theory related?

Game theory and mechanism design both deal with interactions among strategic agents. On the one hand, mechanism design theory studies a scenario by beginning with an outcome and understanding how entities work together to achieve the particular outcome, and on the other hand, game theory looks at how entities can potentially influence multiple outcomes. Because it starts at the end of the game, then goes backwards, Mechanism Design is also called reverse game theory. Game Theory is relevant for mechanism design because it gives us the skills to analyze a game and start thinking in terms of players, choices, incentives, and final outcomes.

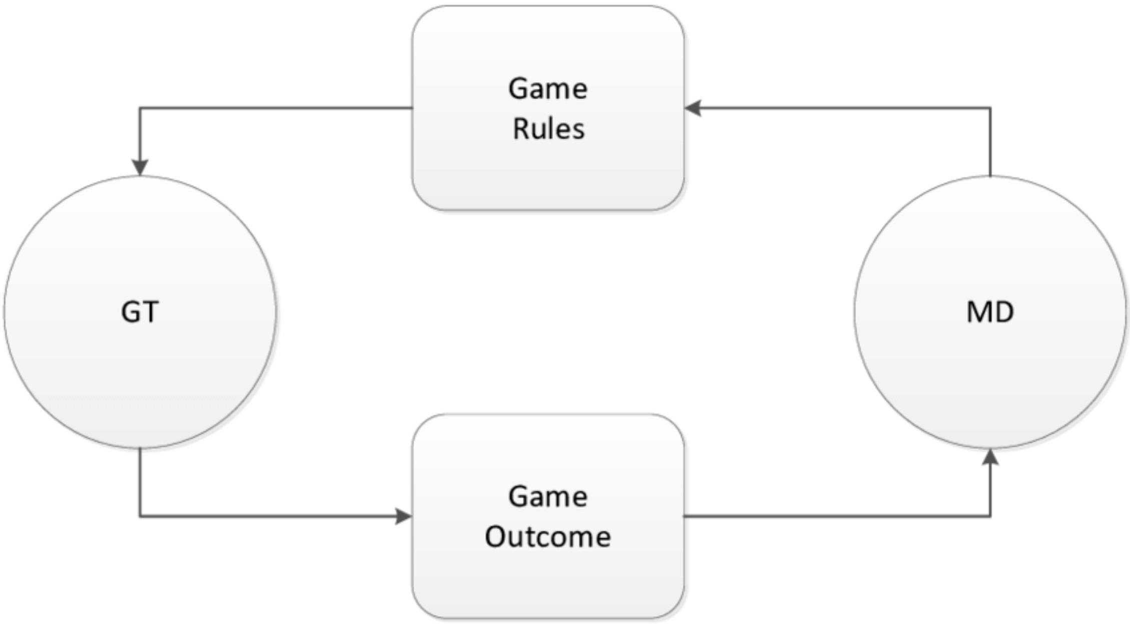

Figure 2.2: Game Theory(GT) vs Mechanism Design(MD)

2.3 Price Mechanisms

2.3.1 What is a Price Mechanism?

The price mechanism is the cornerstone of modern economic systems, shaping the way goods and services are bought and sold, resources are allocated, and businesses make decisions. At its core, the price mechanism is a dynamic force that brings together buyers and sellers in the marketplace, determining not only the cost of products but also influencing the choices and behaviors of individuals and businesses alike. This intricate system of pricing, rooted in the principles of supply and demand, serves as a powerful regulator of economic activity, promoting efficiency, transparency, and equitable resource allocation.

2.3.2 Real-world example of price mechanisms

An example of a price mechanism uses announced bid and ask prices. Generally speaking, when two parties wish to engage in trade, the purchaser will announce a price he is willing to pay (the bid price) and the seller will announce a price he is willing to accept (the ask price). The primary advantage of such a method is that conditions are laid out in advance, and transactions can proceed with no further permission or authorization from any participant. When any bid and ask pair are compatible, a transaction occurs.

2.3.3 How price mechanism works

- **Supply and Demand:** The process begins with the presence of both supply and demand for a particular product or service.
- **Market equilibrium:** The price mechanism aims to find an equilibrium price, where the quantity demanded equals the quantity supplied. This is the price at which the two curves intersect.
- **Price adjustment:** If the current market price is not at equilibrium, it creates either excess demand or excess supply. In excess demand consumers want more of the product than is available at the current price. This exerts upward pressure on prices. In Excess Supply: producers are offering more of the product than consumers are willing to buy at the current price. This puts downward pressure on prices.
- **Equilibrium Reached:** Through this process of price adjustments, the market eventually reaches an equilibrium where the quantity demanded equals the quantity supplied. At this point, the market price is set, and transactions occur at that price.
- **Market Signals:** Prices also serve as signals in the market. When prices rise, it can indicate increased demand or scarcity, encouraging producers to supply more. Conversely, falling prices may suggest oversupply, prompting producers to reduce production.
- **Resource Allocation:** The price mechanism efficiently allocates resources by guiding producers toward goods and services that are in high demand (higher prices) and away from those with lower demand (lower prices).
- **Flexibility:** The price mechanism is flexible and responds to changes in market conditions, such as shifts in consumer preferences, production costs, or external shocks.

2.3.4 Posted Price Mechanisms

The posted price mechanism, on the other hand, involves setting fixed, predetermined prices for goods or services. In this mechanism, sellers determine the prices in advance and display them publicly for potential buyers to see. Buyers can choose to purchase the

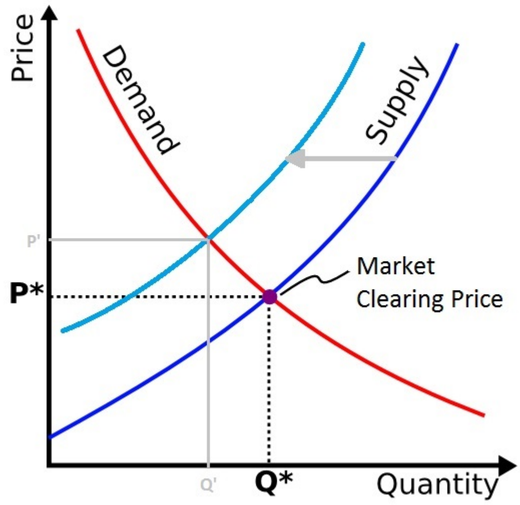

Figure 2.3: Demand-Supply Curve

product at the posted price or negotiate for a different price, but the initial price is typically set by the seller. It's commonly used in various retail settings, where price tags display the cost of items.

While both mechanisms involve the determination of prices, the key difference lies in how prices are set:

- In the price mechanism, prices are determined dynamically through the interaction of supply and demand, and they can change over time.
- In the posted price mechanism, prices are fixed and determined by the seller, with less flexibility for immediate price adjustments based on market dynamics.

2.3.4.1 How posted price mechanisms work

In posted-price mechanism, the auctioneer announces (i.e., posts) the price π at which they are willing to sell the good, after which any bidder who indicates that they are willing to pay the posted price is uniformly eligible to win the good. The winner is then charged the posted price π, and all others pay nothing.

2.3.4.2 Posted Price Mechanisms and Prophet Inequalities

Algorithms for problems like the Prophet Inequality and its variations correspond to posted price mechanisms for approximately maximizing social welfare. A parallel line of work has been to design posted price mechanisms under similar settings for approximate revenue maximization, taking as benchmark the revenue obtained by Myerson's mechanism [33]. The connection between (revenue maximizing) PPMs and prophet inequalities was first studied by Chawla et al.[11] and Hajiaghayi et al. [23].

In the prophet inequality problem there is a gambler, who is faced with a sequence of random variables and has to pick a stopping time so that the expected value he gets is as close as possible to the expectation of the maximum of all random variables, interpreted as what a prophet, who knows the realizations in advance, could get. They implicitly show that any prophet type inequality can be turned into a PPM with the same approximation guarantee. This is obtained by noting that a PPM for revenue maximization can be seen as a (threshold) stopping rule for the gambler, but on the virtual values space, and later identify these virtual thresholds with prices. Correa et al. [13] managed to fill a gap in this area of research by proving the converse of the above result, that any PPM can be turned into a prophet type inequality with the same approximation guarantee.

Bounds obtained for these variants of the prophet inequality problem can be directly used as bounds on the total welfare achieved through PPMs relative to offline welfare maximizing auctions.

2.4 Competitive Analysis

2.4.1 Offline Optimization

Offline optimization refers to the process of solving optimization problems, when all the data is available in advance and decisions can be made non-sequentially so that a certain objective can be optimized. In offline optimization we have complete knowledge of the problem, including parameters, constraints, and objectives prior to making any choices. There can be no better performance than the one computed by the best Offline Algorithm. However this type of framework is very restrictive in cases where input data become available over time. It relies on precomputed decisions made without awareness of the evolving circumstances, thus giving rise to the need for online optimization.

2.4.2 Online Optimization

Online optimization addresses problems having no or incomplete knowledge of the future. The research on online optimization can be distinguished into two caregories: online problems where multiple decision are made sequentialy based on piece by piece input and those were a decision is made only once. An online algorithm must satisfy an unpredictable sequence of requests, completing each request without being able to see the future. As it does not know the whole input, an online algorithm is forced to make decisions that may later turn out not to be optimal, and the study of online algorithms has focused on the quality of decision-making that is possible in this setting.

2.4.3 Randomized Algorithms

Randomized algorithms are a class of algorithms that use randomness or probability to make decisions at each step. Unlike deterministic algorithms that output the same result for a given input, randomized algorithms due to their random nature lead to different outcomes on different runs, each with a certain probability. This kind of algorithm is typically used to reduce either the running time, or time complexity; or the memory used, or space complexity, in a standard algorithm. Also it could help speed up a brute force process by randomly sampling the input in order to obtain a solution that may not be totally optimal, but will be good enough for the specified purposes.

2.4.4 How we assess the performance of an online algorithm

The method used to measure the efficiency of an online algorithm is called competitive analysis. We assume that the offline algorithm serves as a benchmark because it ensures access to all input in advance, making optimal decisions and setting the performance standard for the online algorithm. The primary metric in competitive analysis is the competitive ratio. It quantifies how well the online algorithm performs compared to the offline algorithm. The competitive ratio is typically expressed as a worst-case ratio of the online algorithm's cost (e.g., solution quality, resource usage) to the offline algorithm's cost. By worst-case we refer to worst possible input sequence, specifically chosen to challenge the online algorithm. In some cases, competitive analysis extends to randomized online algorithms. Here, the analysis considers the expected competitive ratio over a random sequence of inputs, providing a probabilistic assessment of algorithm performance. An algorithm will be considered competitive if its competitive ratio is bounded by a constant. In competitive analysis, one imagines an "adversary" which deliberately chooses difficult data, to maximize the ratio of the cost of the algorithm being studied and some optimal algorithm. When considering a randomized algorithm, one must further distinguish between an oblivious adversary, which has no knowledge of the random choices made by the algorithm pitted against it, and an adaptive adversary which has full knowledge of the algorithm's internal state at any point during its execution.

Definition 2.3.1 [32] An online algorithm A is c-competitive if there exists a constant c such that for every input sequence I, the cost of A on I is at most c times the cost of an optimal offline algorithm (OPT) on I. $ALG(I) \leq C * OPT(I)$

c-competitive ratio =$\max_{i \in I} \frac{cost(A)\, with\, input\, i}{cost(OPT)\, with\, input\, i}$

A high competitive ratio, usually greater than 1, indicates that the online algorithm's performance is significantly worse than that of the optimal offline algorithm in the worst-case scenario. Thus we aim to design online algorithms that minimize constant c.

Definition 2.3.2 [22] An upper bound on any algorithm's highest possible competitive ratio for a given instance will be called hardness of the instance. Saying that a prophet inequality model is c-hard or has a hardness of c, means that there is a c-hard instance for that problem, but it does not necessarily mean that c is the lowest hardness possible amongst all instances for that model.

2.5 Adaptive and Non-Adaptive Algorithms

Adaptive and Non-Adaptive Algorithms are two types of online algorithms, that differ on the way they approach decision-making based on the available information. Adaptive algorithms adjust their decision based on the information they have observed or received so far. They are designed to respond to changing conditions or new data as they become available. The flexibility of adaptive algorithms is what makes them ideal for dynamic or uncertain enviroments where real-time updates are essential.

On the other hand, non-adaptive algorithms make decisions based on a plan that is predetermined before any information becomes available. They do not take into consideration any new data or adjust their decisions based on new observations. Non-adaptive algorithms are predictable and deterministic, as they produce the same results for the same input.

2.6 Statistical Tools

The distribution D from where the variables will be drawn is characterized by its:

- Cumulative Distribution Function (CDF) $F : [0, \infty) \to [0, 1]$, $F(x) = PR_{X\sim D}[X \leq x]$
- Probability Density Function (PDF) $f : [0, \infty) \to [0, 1]$

Definition 2.6.1 [31] Hazard rate. For a distribution D with cumulative distribution function F and probability density function f, the hazard rate of D is defined as: $h(x) \overset{\delta}{=} \frac{f(x)}{1-F(x)}$, for all x in the support of D. Sometimes it can be found as failure rate and is a necessary tool for several fields of mathematics and economics.

Definition 2.6.2 [31] Cumulative Hazard rate. It symbolizes the integral H of h: $H(x) \overset{\delta}{=} \int_0^x h(u)du$.

Definition 2.6.3 [31] Poiseux Series. We say that a function $f : \mathbb{R} \to \mathbb{R}$ has a Puiseux series expansion if there exist integers n > 0 and $i_0 \in \mathbb{Z}$ as well as coefficients $a_1, a_2, \ldots$ where $a_1 \neq 0$, such that $f(x) = \sum_{i=i_0}^{\infty} a_i x^{\frac{i}{n}}$. Poiseux series are essentially a generalization of Taylor series because they they allow for fractional exponents in the indeterminate, as long as they have a bounded denominator. The radius of convergence of a Puiseux series around 0 is the largest number r geq 0 such that the series converges if x is substituted for a non-zero real number t $\leq$ r. A Puiseux series is convergent at a point x if x $\leq$ r.

Definition 2.6.4 [31] Entire Distribution. A continuous distribution D with support in $[0, \infty)$ and cumulative hazard rate H is called Entire if $E_{X\sim D}[X] < \infty$ and H has a Puiseux series around 0, $H(x) = \sum_{i=1}^{\infty} a^i x^{d_i}$, the Puiseux series is not identically zero and is convergent for every point in the support of D. Some example of such distributions are uniform, exponential, Gaussian, Weibull, Rayleigh, arcsine, beta and gamma distributions.

Definition 2.6.5 [31] Monotone Hazard Rate Distribution. A distribution D is called a Monotone Hazard Rate (MHR) distribution if and only if the hazard rate function h of D is monotonically increasing.

A hardness for a model is said to be tight (or optimal) when it is matched by the competitive ratio of an algorithm solving it. Similarly, the competitive ratio of an algorithm solving a model is tight (or optimal) when it is matched by a hardness known for that model. Often, when determining a hardness or a competitive ratio, numerical computations are involved. Thus tightness can be used in a broad sense. For two models A and B, we say that A beats B if the hardness of B is strictly less than the competitive ratio of an algorithm solving A. When A beats B or B beats A, we say that A and B are separated.

Definition 2.6.6 [2] Two-point Distributions. A random variable X_i with a two-point distribution is defined by three parameters a_i, b_i, p_i, and takes value

$$X_i = \begin{cases} a_i & \text{w.p.} \quad 1 - p_i \\ b_i & \text{w.p.} \quad p_i \end{cases}$$

Here, $a_i \leq b_i$ are referred to as the left and the right end-point.

Definition 2.6.7 [2] Three-point Distributions. A random variable Xi with a three-point distribution is defined by five parameters a_i, m_i, b_i, p_i, q_i, and takes value

$$X_i = \begin{cases} a_i & \text{w.p.} \quad 1 - p_i - q_i \\ m_i & \text{w.p.} \quad p_i \\ b_i & \text{w.p.} \quad q_i \end{cases}$$

Here, $a_i \leq m_i \leq b_i$, are referred to as the left end-point, the middle point, and the right end-point of the support, respectively.

Definition 2.6.8 [1] m-frequent multiset of distributions. A multiset of independent distributions $\{F_1, \ldots, F_n\}$ is m-frequent if for each distribution F_i in this multiset there are at least m copies of this distribution in the multiset. Generally, a set of n items is called m-frequent if for every item i with distribution function F_i there are at least $m-1$ other items in the set with the same distribution function as F_i.

Definition 2.6.9 [1] m-partitioned items. A sequence of items with distribution functions $F_1, \ldots, F_n$ is m-partitioned if $n = mk$ and the sequence of functions $F_{ik+1}, \ldots, F_{ik+k}$ is a permutation of $F_1, \ldots, F_k$ for every $0 \leq i < m$.

Definition 2.6.10 NP-hardness. A decision problem H is NP-hard when for every problem L in NP, there is a polynomial-time many-one reduction from L to H.

Definition 2.6.11 Nash-equilibrium. The Nash equilibrium is the most common way to define the solution of a non-cooperative game involving two or more players. In a Nash equilibrium, each player is assumed to know the equilibrium strategies of the other players, and no one has anything to gain by changing only one's own strategy.

Definition 2.6.12 [2] Subset Product Problem. Given integers $a_1, \ldots, a_n$ with each $a_i > 1$ and a positive integer B, is there a subset $T \subseteq N$ such that $\sum_{i \in T} ai = B$?

3. PROFIT MAXIMIZATION

3.1 Definition of the Problem

In this chapter, we will delve into the Prophet Inequality problem, with the objective of maximizing our profit. This problem was initially introduced and studied by Krengel and Sucheston [30] in the 1970s. In this scenario, a gambler encounters a finite sequence of non-negative independent random variables $X_1, \ldots, X_n$, each with known distributions D_i. These variables represent the prizes that are drawn iteratively. Upon observing a prize, the gambler faces a critical decision: either accept the prize and conclude the game or irrevocably reject the prize and proceed to the next one.

The ultimate goal is to develop a strategy that maximizes the expected value of the accepted prize. The challenge lies in competing against a prophet who possesses complete knowledge of the realized prizes in advance and is therefore always capable of selecting the best prize. The order in which the prizes are presented can be adversarial, random, or selectively determined. The gambler, who cannot predict the future, aims to maximize the expected value of their rewards while competing against the prophet's expectations, often referred to as the offline maximum. To put it simply, our objective is to maximize the ratio of the gambler's expected value to that of the prophet. The expected profit of the prophet is $E[max_i X_i]$.

3.2 Classic Result by Krengel and Sucheston

The classic result for the Prophet Inequality, on which all further research is based on, is the following: the gambler can always obtain at least half of the expected reward that a prophet can make who knows the realizations of the prizes beforehand, for any arrival order of the random variables.
That is, $sup\{E[X_t] : t\ stopping\ rule\} \geq \frac{1}{2}E[max_i X_i]$. Furthemore, Krengel and Sucheston showed that this bound is the best possible. Samuel-Chan [19] in 1984 indicated that the bound of $\frac{1}{2}$ can be obtained by a simple threshold rule, which stops as soon as a prize is above a fixed threshold T, such that $Pr(max_i X_i > T) = \frac{1}{2}$.

This bound is tight for predetermined orders.

The classical PI has subsequently been relaxed giving more power to the gambler, which leads to larger competitive ratios.

3.3 Order Selection vs Random Order

Naturally, an algorithm consists of two parts: selecting the order and setting the thresholds. Each step is easy to optimize on its own. Specifically, when the arrival order is fixed, the optimal thresholds can be calculated through backward induction; when the thresholds are fixed for each item, we can calculate the expected value of each item conditioning on that its value exceeds the threshold, and then set the arrival order to be a descending order of the calculated values.

When the order in which the random variables $X_1, \ldots, X_n$ are probed is fixed, the optimal stopping strategy can be found by solving a simple dynamic program. Under this strategy, at every step i, the player would compare the realized value of the current random variable X_i to the expected reward (under the optimal strategy for the remaining subproblem) from the remaining variables $X_{i+1}, \ldots, X_n$, and stop if the former is greater than the latter. The celebrated prophet inequalities compare the expected reward of the optimal stopping strategy to $E[max(X_1, X_2, \ldots, X_n)]$, where the latter can be interpreted as the expected reward of a prophet who can foresee the values of all random variables in advance and therefore always stops at the random variable with maximum value. Unlike this case, the problem of finding an optimal ordering for optimal stopping cannot be easily solved in polynomial time by dynamic programming.

The assumption, that the gambler is given an extra power for selecting the arrival order of each item,is natural in the application of sequential posted pricing mechanisms, as the mechanism designer plays the role of the gambler.

One difficulty in such a study is that the nature of this problem changes significantly depending on the type of distributions considered. For example, when distributions are Bernoulli or exponential, the optimal ordering can be found analytically [24], but, the problem remains nontrivial for uniform distributions and distributions with small support.

The model, in which the gambler selects the arrival order first, and then observes the values, is known as Order Selection or Free Order. Whereas, if the gambler chooses the arrival order (uniformly) at random, we obtain the Random Order model.

3.3.1 First appearance of Order Selection setting

T.P. Hill was the first to concern himself with how the order in which variables are presented to the gambler can affect the outcome of the strategy. Hill in 1983 [25] proved that in optimal stopping problems with independent random variables where the gambler is free to choose the order of observation of these variables, the player may do just as well with a prespecified fixed ordering as he can with order selections which depend sequentially on past outcomes. He concluded that the decision maker can choose the order in which to observe the random variables and the time at which to stop the permuted sequence.

To sum up, the aim is now to design a permutation $\pi : [n] \rightarrow [n]$ and a stopping rule τ adapted to the sequence $X_{\pi(1)}, ..., X_{\pi(n)}$, such that $E[X_\tau]$ is maximized.

Later, in 1985 Hill and Hordijk [24] provided simple ordering rules for some families of random variables. This includes the case when every random variable X_i is uniformly distributed between 0 and some positive number α_i, and some very specific cases of two-point distributions.

3.3.2 Free Order Prophet Inequalities

Definition 3.3.2.1 [6] Free order prophet inequality. For the optimal stopping problem with order selection, bounds on the ratio between the gambler's and prophet's expected values are known as free order prophet inequalities.

The optimal factor in the free-order prophet inequality is known to be between 0.669 and 0.745, and closing the gap between these two bounds is a major open question.

The gap between the gambler-to-prophet ratios attainable with or without order selection formalizes, and quantifies, the advantage that a decision maker gains by being able to control the order in which decisions are made under uncertainty. But how much control over the ordering is needed to gain this advantage?

3.3.3 Computational hardness of order selection

3.3.3.1 3-point Distributions

Agrawal et al.[2] showed in 2020 that the optimal ordering problem is NP-hard even under a special case of 3-point distributions where the highest and lowest points of the support are the same for all the distributions. More specifically, by making a reduction from the subset product problem, they proved that the problem of finding an optimal ordering is NP-hard even when each random variable X_i is restricted to be a 3-point distribution with support on $\{0, m_i, 1\}$ for some $m_i \in (0, 1)$, and $E[X_i|X_i > 0] = E[X_j|X_j > 0]$ for all i, j.

However, they provide an FPTAS for the optimal ordering problem for the case when each random variable X_i,$i = 1, \ldots, n$ has a three-point distribution with support on $\{a_i, m_i, 1\}$ for some $a_i, m_i \in [0, 1]$. The FPTAS they propose is as follows: Given a set of n random variables $X_1, \ldots, X_n$, where each random variable X_i, $i = 1, \ldots, n$ has three-point distribution with support on $\{a_i, m_i, 1\}$ for some $m_i \in [0, 1]$ and $a_i < m_i$. Then, there exists an algorithm that runs in time $O(\frac{n^5}{\epsilon^2})$to find an ordering σ such that $ALG = V_\sigma \geq (1-\epsilon)OPT$. Here, $OPT := V_{\sigma^*}$ denotes the optimal expected reward at stopping time under an optimal ordering σ^*. To establish the aforementioned result, they employ an FPTAS for the special case in which both the left and right endpoints are identical for all i. Then they subsequently expand this approach to encompass instances where only the right endpoints are identical.

The algorithm for same left and right end points is:

Algorithm 1 FPTAS for finding the optimal ordering through optimal partitioning [2]

Input: Ordered sequence of variables $X_1, \ldots, X_n$ such that $E_1 \leq \cdots \leq E_n$, parameters MAX, ϵ.
Initialize: $L^0 = \{(\phi, \phi)\}, L^1 = \cdots = L^n = \phi$
for $k = 1$ to n **do**
  for all $(S, T) \in L^{k-1}$ **do**
    Add two partitions $(\{X_k, S\}, T)$ and $(S, \{X_k, T\})$ to L_k.
  end for
  Call Algorithm 2 to reduce the number of partitions in L^k by setting $Lk \leftarrow TRIM(L^k, \epsilon, MAX)$.
end for
return L^n

Algorithm 2 $TRIM(L^k, \epsilon, MAX)$ [2]

Initialize: $\rho := (1 - \frac{\epsilon}{2n}), max := max_{(S,T)\in L} V(T)$, and $J := max\{j : \rho^j max \geq \frac{\epsilon}{2n} MAX\}$
Divide the partitions in L into $J + 1$ buckets as
$B_j := \{(S, T) : {}^j max < V(T) \leq {}^{j-1} max\}, for j = 1, \ldots, J$
$B_0 := \{(S, T) : T = \phi\}$
Set$(S_j, T_j) := argmax_{(S,T)\in B_j} V(S)$,for $j = 0, 1, \ldots, J$.
return $L := \{(S^j, T^j)\}_{j=0}^{J}$.

3.3.3.2 2-point Distributions

Whereas, in the case of 2-point distributions they managed to design an effective polynomial time algorithm for finding an optimal ordering. Furthermore, they showed that when provided with any set of variables characterized by 2-point distributions, the prophet inequality holds with a significantly enhanced factor of 1.25 under the optimal ordering, compared to the factor of 2 for arbitrary orderings. It's important to note that the prophet inequality is tight for 2-point distributions. This highlights the significance of the ability to choose an ordering.

The strategy they proposed for the 2-point distributions is as follows:
Given n variables with two-point distributions, define n orderings as follows: for each $i = 1, \ldots, n$, define σ_i as any ordering obtained by setting the last variable as X_i, and ordering the remaining variables in weakly descending order of their right endpoints. Then, at least one of these n orderings is optimal. This algorithm has complexity $O(n^2)$.

Hill and Hordijk gave examples to support their idea that simple rules of thumb-ordering based on mean or variance; stochastic ordering, assuming the variables are all stochastically ordered—do not work. The NP-hardness result of Agrawal et al. further suports this point as they suggest that such heuristic rules are unlikely to be optimal.

3.3.4 Beating the $1 - \frac{1}{e}$ bound

The first free order prophet inequality was proven by Yan [39], who showed that the gambler-to-prophet ratio is always at least $1 - \frac{1}{e} = 0.632$. This bound was later shown to be attainable even if the gambler is constrained to observe the values in uniformly-random order [20] and to use a threshold stopping rule [14]. More about how these bounds are obtained can be found in section 5.4. Furthermore, $1 - \frac{1}{e}$ is asymptotically the best possible ratio attainable by threshold stopping rules (Kleinberg and Kleinberg, 2018), even if the distributions of $X_1, \ldots, X_n$ are identical. However, general stopping rules can do strictly better: the optimal factor in the free-order prophet inequality is known to be between 0.669 and 0.745, and closing the gap between these two bounds is a major open question.

3.3.5 I.I.D. Variables

The case of I.I.D Variables was initially studied by Hill and Kertz [26] in 1985. They proved the theoretical bound of $1 - \frac{1}{e}$ on the approximation factor by designing complicated recursive functions. Through a computer program that run the algorithm for an input of $n = 1000$ distributions, they showed that it can achieve a 0.745-approximation. Finally, they speculated that the best approximation factor for arbitrarily large n is $\frac{1}{1+\frac{1}{e}} \approx 0.731$.

Three decades later Abolhasani et al.[1] presented a threshold-based algorithm for the prophet inequality with n i.i.d. distributions, which can obtain a 0.738-approximation for large enough n, beating the bound of $\frac{1}{1+\frac{1}{e}}$ conjectured by Hill and Kertz. The algorithm is as follows:

Algorithm 3 0.738 approximation [1]

Input: n iid items with distribution function F.
Set a to $1.306(rootofcos(a){-}sin(a)/a{-}1)$.
Set θ_i = $F^{-1}(cos(ai/n)/cos(a(i{-}1)/n))$.
Pick the first item i for which $X_i \geq \theta_i$.

For simplicity purposes they have made the following assumptions: $X_1, \ldots, X_n$ are iid random variables with common distribution function F, F is continuous and strictly increasing on a subinterval of $R^{\geq 0}$, τ denotes the stopping time of this algorithm, where τ is n + 1 when the algorithm selects no item and X_{n+1} is a zero random variable. Just a brief reminder from the previous chapter, the approximation factor of an algorithm based on $\theta_1, \ldots, \theta_n$ is defined as $E[X_\tau]/E[maxX_i]$. This factor captures the ratio between what a player achieves in expectation by acting based on these thresholds and what a prophet achieves in expectation by knowing all X_i's in advance and taking the maximum of them.

In 2017, Correa et al.[12] proved that 0.745 is a tight value. The strategy they proposed is a variant of the following algorithm:

Algorithm 4 0.745 approximation [12]

Input: Customers $i \in I$ with valuation i.i.d. according to F.
Partition the interval $[0, 1$]into intervals $A_i = [a_{i-1}, a_i]$, s.t. $a_0 = 0, a_n = 1$.
Sample q_i from A_i with an appropriately chosen distribution.
When the i−th buyer comes, offer price p_i =max$\{F^{-1}(1{-}q_i), v^*\}$,where v^* is the reservation price of the optimal auction.

This algorithm can be derandomized using standard techniques. The strategy they designed gives a sequence of thresholds $\tau_1, \ldots, \tau_n$ such that, if we take the first of n i.i.d. random variables whose value is above the threshold, we obtain a value of at least a 0.745 fraction of the expectation of the maximum of the random variables. This result can be seen as a follow up on a result by Hill and Kertz[26] on the prophet inequality for i.i.d. random variables. The lower bound of Correa et al.[12] is in fact known to be tight due to an impossibility result of Hill and Kertz[26] and Kertz[28] that implies a matching upper bound.

In general, the problem of the prophet inequality with random ordered i.i.d. variables is equivalent to the Prophet Secretary that will be discussed in section 5.4, any results that hold for the latter problem also hold for the former.

3.3.6 Non-I.I.D. Variables

Abolhasani et al. [1] extened their results to cover the case of different distributions, but they assume that we have several copies of each distribution. This can be reinterpreted as a large market assumption. They managed to show that by allowing the algorithm to pick the order of the distributions, there exists a 0.738-approximation algorithm for any prophet inequality instance on a set of m-frequent distributions, for large enough m. This can also be achieved even in the random order setting. However, this result cannot be extened to the worst case order setting.

They showed for the best order and random order of a large market instance that one can find a sequence of thresholds which in expectation performs as good as the algorithm for iid items. Roughly speaking, they designed algorithms that are α-approximation for large enough m-frequent instances, where $\alpha \approx 0.7388$.

To prove the result for best and random order, they provide an algorithm for a specific class of large market instances, namely partitioned sequences. The algorithm is the following:

Algorithm 5 partitioned sequences for large market instances [1]

Input: An m-partitioned sequence of items with distribution functions $F_1, \ldots, F_n$.
Let k=n/m.
Let $F(x) = \prod_{i=1}^{k} F_i(x)$.
Let $\theta_1, \ldots, \theta_m$ be the thresholds by Algorithm 3 for m iid items with distribution function F.
Pick the first item i if $X_i \geq \theta_{\lceil \frac{i}{k} \rceil}$.

3.3.7 Constrained-Order Prophet Inequalities

Definition 3.3.7.1 [6] A non-empty set of permutations $\Pi \subseteq S_n$ is said to satisfy a constrained -order prophet inequality with factor α if for every n-tuple of distributions $X_1, \ldots, X_n$ supported on the non-negative reals, there is a permutation $\pi \in \Pi$ and a stopping rule τ adapted to $X_{\pi(1)}, X_{\pi(2)}, ..., X_{\pi(n)}$, such that $E[X_\tau] \geq \alpha E[max_{1 \leq i \leq n} X_i]$.

Definition 3.3.7.2 [6] Prophet ratio of Π. The prophet ratio of Π, PR(Π), is the supremum of all α such that Π satisfies a constrained-order prophet inequality with factor α.
Constrained-order threshold prophet inequalities and the threshold prophet ratio TPR(Π) are defined similarly, but allowing the gambler to optimize only over threshold stopping rules rather than all stopping rules.

Up to this point it has been proven that if the gambler utilizes a threshold based stopping rule then the gambler-to-prophet ratio is at least $1-\frac{1}{e} \approx 0.632$, whereas if he has to examine the values in a predetermined order the ratio is bounded by $\frac{1}{2}$. Arsenis et al.[6] were one of the first that studied a setting that lies between these two extremes. This instance of the Prophet Inequality problem allow us to gain deeper insight into how and why optimizing the order of decisions leads to better outcomes for optimal stopping rules.

They examine the scenario where the gambler has a set of predetermined permutations of the set indexing the random variables, and he is free to choose the order of observation to be any one of these predefined permutations.

Arsenis et al. [6] concluded that even when only 2 orderings are available to choose from, the ratio improves to the inverse of the golden ratio $\phi^-1 = \frac{1}{2}(\sqrt{5}-1) = 0.618$. The bound of $\phi^-1 + o(1)$ cannot be broken with less than $O(logn)$ allowed permutations. Lastly, the ratio reaches $1-\frac{1}{e}-\epsilon$ for a suitably chosen set of $O(poly(\epsilon^{-1}) \cdot logn)$ permutations and does not exceed $1-\frac{1}{e}$ even when the full set of $n!$ permutations is allowed.

In the extreme cases where Π has only one element or Π is the entire permutation group S_n, one recovers the definitions of prophet inequality and free-order prophet inequality, respectively.

Even though it would be ideal to achieve a similarly deep understanding of the minimum set of permutations required to attain a specific prophet ratio (rather than threshold prophet ratio), at the moment it still remains an open problem. For the time being, the extent of their findings lies in the establishment of lower bounds on the prophet ratio PR(Π) for the sets of permutations they investigate. This is a direct consequence of the elementary observation that for any given set Π, $PR(\Pi) \geq TPR(\Pi)$.

3.3.8 Arrival Time Decision Problem

In 2022 Peng and Tang [35], designed a strategy that achieves the optimal 0.745 competitive ratio for the i.i.d. model. Despite the NP-hardness result of Agrawal et al.[2], they introduce a novel algorithm design framework that translates the discrete order selection problem into a continuous arrival time design problem. This perspective allows them to concentrate on arrival time design without the need to address threshold optimization afterward. By exploiting the power of order selection they designed a 0.725-competitive algorithm.

Before we proceed to the algorithm, we need to introduce some assumptions that were made. We remind that the time horizon is $[0, 1]$ and each item i arrives at time $t_i \sim Uniform[0, 1]$. However, they start by rescaling the time horizon and fixing the time-dependent thresholds. More specifically, at any time t, the threshold $\tau(t)$ is chosen such that the maximum value of all items exceeds it with a probability of precisely t. Subsequently, they design an arrival time distribution F_i for each item i, and items arrive at random times according to F_i. The algorithm draws the arrival time t_i of each item independently from carefully constructed arrival time distributions F_i, and subjects them to a common time-dependent threshold function. The items are made to arrive in ascending order of their arrival times and the algorithm accepts the first item that satisfies $v_i > \tau(t_i)$, where $\tau(t)$ satisfies $P[max_{v_i} > \tau(t)] = t$. This formulation is essentially without loss of generality, as we can choose deterministic distributions. In this context, optimization focuses solely on arrival times. Notably, when the distributions F_i are identical, their algorithm can be applied in the prophet secretary setting. Their algorithm achieves a competitive ratio of 0.725, which is enhanced to 0.745 in the i.i.d. setting.

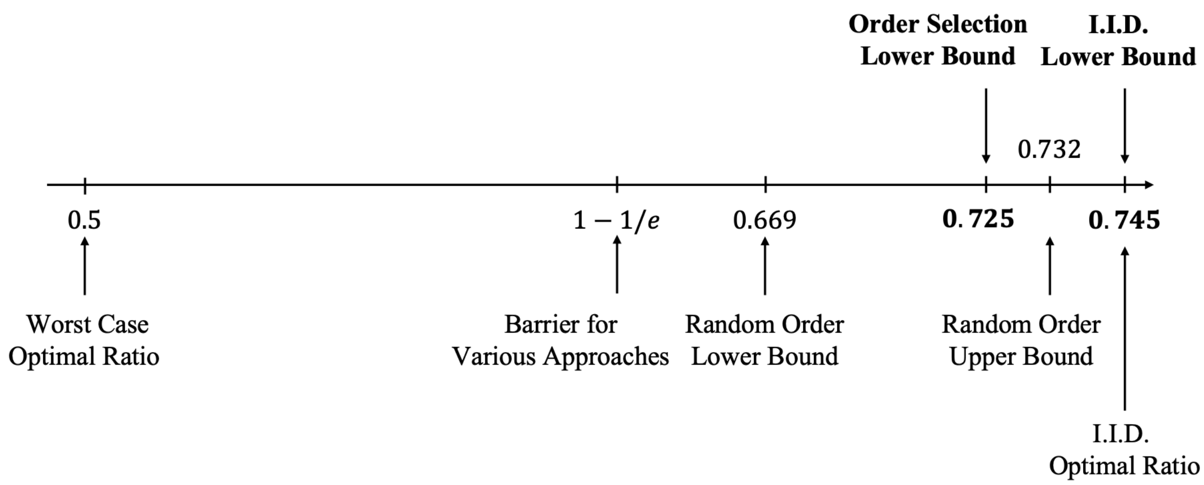

Figure 3.1: Summary of results [35]

Peng and Tang's [35] analysis bridges the i.i.d. setting and the order selection setting, and suggests that their novel arrival time design perspective to be the right framework.

3.3.9 Order selection beats random order

In 2023, Bubna and Chiplunkar [10] identified an instance for which Peng and Tang's algorithm failed to achieve the claimed ratio of 0.7251. In response, they proposed an improved strategy. Drawing inspiration from their algorithm, they presented a more general algorithm design framework capable of attaining a 0.7258-competitive ratio. Bubna and Chiplunkar [10] further refined Correa et al.'s [14] 0.732-hardness result (which is discussed in more detail in section 5.4), demonstrating a heightened hardness of 0.7254 for general algorithms in the random order setting. This holds true even when the gambler has prior knowledge of the arrival order. Consequently, they successfully establish a distinction between the order selection and random order settings.

Essentialy Bubna et al.[10] extend Peng et al.'s design by proposing a more general framework to design algorithms that use independent arrival times. This framework allows the algorithm to use a different time-dependent threshold function for each item.

Firstly, Bubna et al.[10] identify a set of distributions for which Peng and Tang's algorithm can perform no better than their claimed competitive ratio. This instance is composed of N IID variables whose maximum is distributed uniformly over $[0, 1]$, and another variable which is uniformly distributed over the interval $[\alpha, \alpha + \frac{1}{N}]$, where $\alpha \simeq$ 0.2109. Using N calculations, it can be verified that as N approaches ∞, the maximum competitive ratio that Peng and Tang's algorithm can achieve for the above instance approaches 0.7251. The framework utilized by Bubna et al. allows them to choose a set of identity-dependent threshold functions $\tau_i(t)$ for the items, and they show that they can find a set of thresholds for which the algorithm generated by the framework guarantees 0.7258.

Bubna et al.'s algorithm differs from Peng et al.'s at the step which defines which items are accepted. Peng et al. use common threshold function for all items, whereas Bubna et al. have a separate threshold function for each item. Their contribution in the order selection problem is not the numerical improvement they prove but rather it is the demonstration of the fact that Peng and Tang's ratio can be beaten by relaxing the constraints of their algorithm and using a more general approach.

3.3.10 Separating random order from order selection

Giambartolomei et al.[22] further improved the hardness of Random Order, to the extent of separating it from the Order Selection setting. In order to establish a hardness for Random Order it is enough to show a uniform upper bound on the competitive ratio of all stopping rules, that is an upper bound, holding for some given instance and uniformly for all stopping rules τ, on $\frac{EV_{\pi_\tau}}{Emax_i V_i}$.This is done by upper-bounding the expected reward EV_{π_T} of an optimal stopping rule T, which is a stopping rule (existing by backward induction) maximising the expected reward. Moreover, they managed to prove a 0.7235-hardness for random order and thus order selection is separated from random order. More information about the techniques they used can be found in section 5.4.6.

4. COST MINIMIZATION

Up to this point, our focus has been solely on studying the Prophet Inequality problem with the aim of maximizing our profit. However, in this chapter, we will delve into the Prophet Inequality problem with the objective of minimizing our costs. It's worth noting that the problem's complexity undergoes a significant transformation when the goal shifts from maximizing rewards to minimizing expenses. These cost-related prophet inequalities have practical applications in procurement auctions, particularly when a single buyer seeks to acquire items from multiple sellers.

The cost minimization variant was firsly observed by Esfandiari et al. [20] in 2015 and most recently by Livanos and Mehta [31]. In the minimization problem, we have to choose one element from the input and aim to minimize the expected value of the chosen element.

4.1 Definition of the Problem

Let's consider a real-life example where the cost minimization problem can be applied. Imagine you are house hunting, trying to decide when to buy a house in the seller's market. When a house arrives with its price listed, you have to decide irrevocably the same day whether to buy it or not. Given that you may have only distributional knowledge of future house prices, the goal is to devise a buying strategy so that the price paid is minimized.

A more formal definition is as follows: We are given as input n distributions $D_1, \dots, D_n$ supported on $[0, \infty)$, and we sequentially observe the realizations of n random costs $X_1 \sim D_1, \dots, X_n \sim D_n$. We must "stop" at some point and take the last cost seen. In particular, at any point after observing an X_i, we can irrevocably choose to select or discard it. If we select X_i, then the process ends and we receive a cost equal to X_i. Otherwise X_i gets discarded forever and the process continues. Now insert an all-knowning prophet, who knows all the realizations of the X_i's in advance and always selects the minimum realized cost and therefore his expected cost is Offline-OPT = $E[min_i X_i]$. Our goal is to design a stopping strategy that minimizes the expected cost, and is comparable to Offline-OPT.

At this point, we would just like to remind what a-competitive means in the context of cost minimization: For an $\alpha \geq 1$, we say that algorithm ALG achieves an α-factor cost prophet inequality, or is α-competitive/approximate, if $E[ALG] \leq \alpha \cdot E[miniX_i]$

4.2 Difference from profit maximization

At first glance, we would expect the two problems to be equivalent. However, due to the upwards-closed constraints nature of the cost minimization, qualitatively different guarantees are required. On the other hand, an algorithm for the profit maximization utilizes downwards-closed constraints and thus if all X_i's are negative, the optimal solution is trivial and the algorithm will not select any X_i and obtain a value of 0. Something similar is not possible for the cost variant.

4.3 Not identically distributed variables

4.3.1 Random or adversarial order

Esfandiari et al. [20] observed that for the non-I.I.D. case where the arrival order of the variables is random or adversarial, no algorithm can achieve any bounded approximation.Most recently, Livanos and Mehta [31] showcased a simplified version of their counter-example and proved that with adversarial or random order arrival, no algorithm is α-factor competitive for any bounded α, even when restricted to n = 2 and distributions with support size at most two.

4.4 I.I.D. variables

The negative results of the above setting serve as a motivation to study the I.I.D. case.

Some useful notation for the algorithms discussed below:
Given an algorithm A, we will symbolise as $G_A(i)$ its expected cost when it has observed i I.I.D random variables drawn from D. So the overall expected cost of A is $E[A] = G_A(n)$. The expected cost of the prophet who has all the information in advance will be denoted as β_n and the competitive ratio of A as $R_A(n) = \frac{G_A(n)}{\beta_n}$. Lastly, Livanos and Mehta proved that $\beta_n = \int_0^\infty e^{-nH(u)du}$

4.4.1 Infeasibility of Online Algorithms with One Exchange

Esfandiari et al. [20] proved that even for the simple case of identical and independent distributions, there is no $\frac{1.11^n}{6}$-competitive online algorithm for the minimization variant of the prophet inequality problem. Since the input items come from identical distributions, independently, randomly reordering them does not change the distribution of the items in the input. Then, they construct an algorithm, which has the power to change its decision once, it is called online algorithm with one exchange. They proved that for any large number C there is no C-competitive online algorithm with one exchange for the minimization variant of the prophet inequality.

4.4.2 Single threshold algorithms

Livanos and Mehta initiated their analysis by questioning whether single threshold algorithms can prove as effective in achieving constant-factor approximations in the cost setting as they are in the profit-oriented one. The intuition behind this idea was that if n is very large, one could set a single threshold close to $E[min_i X_i]$ and with good probability there will be at least one realization below the threshold. Unfortunately, this intuition turned out to be wrong as it did not yeald the desired results.

The single threshold algorithm that utilizes the threshold T is as follows:

Algorithm 6 Single threshold algorithm [31]

```
set T = Θ((logn/n)^k)
for i = 1 to n do
    let z_i be the realization of X_i
    if z_1, z_2, ..., z_{i-1} were not selected and z_i ≤ T then
        select z_i and exit
    end if
end for
```

If the algorithmnd reaches X_n then the gambler is forced to pick the realization of X_n, no matter how high its cost is. Through this strategy they obtain a O (polylog n)-factor cost prophet inequality. In this instance the power in the poly-logarithmic factor inversely depends on the smallest degree of the Puiseux series of H. The thershold they chose is: $T = \Theta((\frac{logn}{n})^k)$, where the value of k depends on the given distribution.

4.4.2.1 Upper Bound

Let D be an entire distribution on $[0, \infty)$ for which the cumulative hazard rate H has Puiseux series H(x) = $\sum_{i=1}^{\infty} a_i x^{d_i}$, where $d_1 < d_2 < \ldots$. They proved that there exists a single threshold T=T(n, d_1, a_1) such that if used for the algorithm described above we achieve a $O((logn)^{\frac{1}{d_1}})$-competitive ratio compared to β_n for a large enough n.

4.4.2.2 Lower Bound

Considering a distribution D for which $H(x) = x^d$ for $d \geq 0$, there is no $o((logn)^{\frac{1}{d}})$-competitive single-threshold cost prophet inequality for the single-item setting and I.I.D. random variables drawn from D. Therefore the upper bound is asymptotically tight.

Their results imply that given $X_1, \ldots, X_n$ drawn independently from a non-negative Entire distribution D, there exists a single-threshold algorithm that is O(polylogn)-competitive, for large enough n. Moreover, this factor is tight, i.e. there exist distributions for which no single-threshold algorithm is o(polylogn)-competitive.

4.4.3 Multiple threshold algorithms

Their goal was to design algorithms for the CPI setting, which can achieve the smallest possible α in $E[ALG] \leq \alpha \cdot E[miniX_i]$. They managed to achieve a (distribution dependent) constant factor CPI for the class of Entire distributions, and a 2-factor CPI for Entire MHR distributions. Following the steps of the profit setting, they focus on threshold algorithms, where the algorithm in advance chooses n thresholds ($\tau_1, \tau_2, \ldots, \tau_n$) only taking into account the distribution D. These algorithms are also known as oblivious algorithms because they do not depend on the realizations of the random variables. Also the process that these algorithms follow is memoryless as the decision at each steps solely depends on the realization of X_i and the number of the remaining variables.

4.4.3.1 How optimal thresholds algorithms work

Optimal threshold algorithms have a very natural interpretation: the algorithm should select the next random variable X_i if and only if its value is smaller than the value it expects to receive by ignoring X_i and continuing the process. The optimal threshold for the next random variable when we have k realizations left to examine is exactly the expected cost incurred by an optimal algorithm when its input is k − 1 I.I.D. random variables. After analyzing the performance of this strategy, they show it can obtain constant-factor competitive ratio for the class of Entire distributions.

The algorithm they proposed, which achieves the best possible competitive ratio for CPI is as follows:

Algorithm 7 Multi-threshold algorithm [31]

```
set τ_n ← ∞ and τ_{n−1} ← E_{X∼D}[X]
for i = n − 2 to 1 do
    τ_i = F(τ_{i+1})E[X|X ≤ τ_{i+1}] + (1 − F(τ_{i+1})τ_{i+1})
end for
for i = 1 to n do
    let z_i be the realization of X_i
    if z_1, z_2, . . . , z_{i−1} were not selected and z_i ≤ τ_i then
        select z_i
    end if
end for
```

4.4.3.2 Competitive ratio and Upper and Lower Bounds

Livanos and Mehta proved for the constant factor that: For the I.I.D. setting under any given non-negative Entire distribution D, for large enough n, there exists a $\lambda(d)$-factor cost prophet inequality, where $\lambda(d) = \frac{(1+\frac{1}{d})^{\frac{1}{d}}}{\Gamma(1+\frac{1}{d})}$, d is the smallest degree of the Poiseux series of H and Γ is the Gamma function. Furthermore, this factor is tight for the distribution with $H(x) = x^d$. This result it is not good nor bad. As we can obtain a constant-factor competitive ratio for every fixed distribution, but the constant may be too large.

Upper Bound

Let D be an Entire distribution on $[0, \infty)$ with cumulative hazard rate H, which has a Puiseux series H(x) = $\sum_{i=1}^{\infty} a_i x^{d_i}$, where $d_1 < d_2 < \ldots$, let $\lambda(d_1) = \frac{(1+\frac{1}{d_1})^{\frac{1}{d_1}}}{\Gamma(1+\frac{1}{d_1})}$. Then the above algorithm achieves a $\lambda(d_1)$-competitive ratio with respect to β_n, for large n. As for the expected cost of the algorithm they showed that it is equal to $G(n) = \int_0^{G(n-1)} e^{H(u)} du$. Also $\beta_n = \frac{\Gamma(1+\frac{1}{d_1})}{(a_1 n)^{\frac{1}{d_1}}} + o(\frac{1}{n^{\frac{1}{d_1}}})$. As a result, the competitve ratio R(n) is bounded by $R(n) \leq \frac{(1+\frac{1}{d_1})^{\frac{1}{d_1}}}{\Gamma(1+\frac{1}{d_1})}$.

Lower Bound

They managed to show that considering the distribution D for which $H(x) = x^d$ for $d \geq 0$, for any $\epsilon > 0$, there is no $\frac{(1+\frac{1}{d})^{\frac{1}{d}}}{\Gamma(1+\frac{1}{d})}$–-competitive cost prophet inequality for the single-item setting and I.I.D.random variables drawn from D. In simpler words there exist Entire distributions for which the upper bounds given by λ are tight.

4.4.3.3 MHR Distributions

Even though the competitive ratio that is achieved by the above algorithm is distribution-dependent, they proved that for MHR distributions a uniform factor of 2 can be obtained. This factor is tight.

4.5 CPI and mechanism design

Finally, it's worth mentioning that, similar to how classical prophet inequalities have extensive applications in creating simple yet nearly optimal posted-price mechanisms for selling items, the above-mentioned algorithms and findings for the cost prophet inequality can also be applied to the design and analysis of posted-price-style mechanisms for procuring items.

Let's consider a procurement auction, also known as a reverse auction, where the auctioneer (the buyer) aims to purchase a single item from among n different sellers. In this scenario, each seller's valuation for selling the item to the auctioneer, or in simpler terms, their cost or price, follows an I.I.D. distribution D. If the sellers present themselves in an online fashion with take-it-or-leave-it offers, such as in a seller's housing market, then the standard approach of reducing a posted-price mechanism to a prophet inequality, applies directly to the cost-based setting when the goal is to minimize the overall social cost.

To minimize the procurement price paid by the buyer (auctioneer), one can simply use the virtual costs defined as $\phi(c) = c + \frac{F(c)}{f(c)}$. This approach aligns with Myerson's optimal auction [33], which can be applied in any single-parameter environment. However, it's important to note that this approach is valid only when the distribution D is regular (a category that includes various distributions, including MHR distributions, among others). For non-regular D distributions, a slight modification of the social cost function is necessary, similar to what is done in the traditional profit-oriented setting, followed by a similar process.

5. OTHER VARIATIONS

5.1 Multiple Choices Prophet Inequality

The first generalization of the classic prophet inequality problem [30] is the multiple-choices prophet inequality problem([7], [27]), in which either the gambler, the prophet, or both are given the power to choose more than one element.

Assaf et al.[7] studied the problem of designing k stopping rules $\tau_1 \leq \cdots \leq \tau_k$ to optimize the quantity $E(max_i\{x_{\tau_i}\})$. They proved that there exists a sequence of k stopping rules such that the expected maximum of the k choices is within a factor $\frac{k+1}{k}$ of the prophet's payoff. However, in the auction setting, the natural objective is to maximize the expected sum of the k choices rather than their expected maximum. Only one more paper by Kennedy [27] considers this objective. He compares the sum of the k values chosen by the gambler with a single value chosen by the prophet.

Currently, the best algorithm for the most natural case, in which both the gambler and the prophet have k > 1 choices is due to Alaei [4], who gave an algorithm with competitive ratio $1 - O(k - \frac{1}{2})$, which is known to be asymptotically optimal.

5.2 Matroid Prophet Inequality

Kleinberg and Weinberg [29] considered the matroid prophet inequality problem, where the feasible subsets of random variables are independent sets of a given matroid.

In the matroid prophet inequality, we are given a matroid whose elements have random weights sampled independently from (not necessarily identical) probability distributions on $\mathbb{R}^+$. We then run an online algorithm with knowledge of the matroid structure and of the distribution of each element's weight. The online algorithm must then choose irrevocably an independent subset of the matroid by observing the sampled value of each element (in a fixed, prespecified order). The online algorithm's payoff is defined to be the sum of the weights of the selected elements. Kleinberg and Weinberg [29] show that for every matroid, there is an online algorithm whose expected payoff is at least half of the expected weight of the maximum-weight basis($\frac{1}{2}$-competitive algorithm).

It is interesting to note that the original prophet inequality introduced by Krengel and Sucheston [30] corresponds to the special case of rank-one matroids.

5.3 Matching Prophet Inequality

The matching prophet inequality is due to Alaei, Hajiaghayi, and Liaghat [3]. They study the problem of online prophet-inequality matching in bipartite graphs.

We assume there is a static set of bidders and an online stream of items. The interest of bidders in items is represented by a weighted bipartite graph. Each bidder has a capacity, i.e., an upper bound on the number of items that can be allocated to him. The weight of a matching is the total weight of edges matched to the bidders. Upon the arrival of an item, the online algorithm should either allocate it to a bidder or discard it. The objective is to maximize the weight of the resulting matching. They generalize the $\frac{1}{2}$-competitive ratio of Krengel and Sucheston [30] by designing an algorithm with an approximation ratio of $1 - \frac{1}{\sqrt{k+3}}$, where k is the minimum capacity.

It is worth to mention that the classical prophet inequality is a special case of this model where we have only one bidder with capacity one, i.e., k = 1 for which they get the same $\frac{1}{2}$-competitive ratio.

5.4 Prophet Secretary

5.4.1 Definition of the Secretary Problem

Until now, we have exclusively discussed the Prophet Inequality problem. Let's now introduce the Secretary Problem and explore how these two problems can be integrated.

The basic concept of the Secretary Problem is: Imagine that you manage a company, and you want to hire a secretary from a pool of n applicants. You aim to hire only the best and brightest. Unfortunately, you cannot tell how good a secretary is until you interview him, and you must make an irrevocable decision whether or not to make an offer at the time of the interview. The problem is to design a strategy which maximizes the probability of hiring the most qualified secretary. For this problem, Dynkin [17] presents a simple but elegant algorithm that succeeds with probability at least $\frac{1}{e}$; indeed, the success probability converges from above to $\frac{1}{e}$ as n grows large, and $\frac{1}{e}$ is the best possible bound (up to lower order terms) we can achieve for this problem. This bound can be achieved by rejecting the first t-1 applicants and accepting the first applicant whose qualifications, exceed that of the first t-1. t is defined as: $\sum_{j=t+1}^{n} \frac{1}{j-1} \leq 1 < \sum_{j=t}^{n} \frac{1}{j-1}$.

5.4.2 Definition of the Prophet Secretary Problem

Prophet Secretary is a natural combination of the Prophet Inequality and the Secretary Problem. Let's describe a real-world example of this variant. Consider a seller that has an item to sell on the market to a set of arriving customers. The seller knows the types of customers that may be interested in the item and he has a price distribution for each type: the price offered by a customer of a type is anticipated to be drawn from the corresponding distribution. However, the customers arrive in a random order. Upon the arrival of a customer, the seller makes an irrevocable decision whether to sell the item at the offered price. We address the question of finding a strategy for selling the item at a high price.

A more formal definition of the problem: We are given a set $\{D_1, D_2, \ldots, D_n\}$ of distributions, which are not necessarily identical. A number X_i is drawn from its distribution D_i, and then after applying a random permutation $\pi_1, \pi_2, \ldots, \pi_n$ the numbers are presented to us in an online fashion, for example in step k, we are presented with X_{π_k} and π_k. We can choose only one number and we must do so irrevocably when the number is revealed. Our goal is to maximize the expectation of the chosen value, in comparison with the expectation of the optimum offline solution that knows the drawn values from the start(OPT).

The line of work studying prophet secretary can be viewed as a two-step approach of first selecting the order, and then designing the thresholds. More accurately, the algorithm selects the uniform distribution over all permutations and then focuses on designing the thresholds.

If $X_1, X_2, \ldots, X_n$ are identical then the Prophet Secretary is equivalent to the Prophet

Inequality problem.

As we have mentioned in Section 3.2 in the classic prophet inequality problem, a tight competitve ratio of $\frac{1}{2}$ can be achieved by choosing the same threshold $\frac{OPT}{2}$ in every step. However, in the classic secretary problem, where distributions are not known the optimal strategy is to let $\tau_1 = \cdots = \tau_{\frac{n}{e}} = \infty$ and $\tau_{\frac{n}{e}+1} = \cdots = \tau_n = max(X_{\pi_1}, \ldots, X_{\pi_{\frac{n}{e}}})$, which leads to the optimal competitive ratio of $\frac{1}{e} \simeq 0.36$. Therefore our goal is to beat the $\frac{1}{2}$ barrier. Since the order is no longer adversarial, it is natural to expect that algorithms with competitive ratio larger than $\frac{1}{2}$ will exist.

5.4.3 Achieving the $1 - \frac{1}{e}$-competitive ratio

Esfandiari et al. [20] demonstrated that, unlike the prophet inequality in the prophet secretary problem, at least two thresholds are required to surpass the $\frac{1}{2}$ barrier. Firstly, they established that the competitive ratio of an online algorithm employing only one threshold is bounded by $0.5 + \frac{1}{2n}$.

Next, they devised an algorithm that utilizes only two thresholds. For the first half of the steps, they employ one threshold, and for the remaining half, they switch to a different one. It's important to note that both thresholds must be proportional to OPT (the optimal solution value). Initially, they adopt an optimistic approach and set a higher threshold, but if they fail to pick a value during the first half, they lower the threshold. By implementing this strategy, they achieve a competitive ratio of approximately $\frac{5}{9} \simeq 0.55$.

Finally, they designed an algorithm that attains a $1 - \frac{1}{e} \simeq 0.63$-competitive ratio by employing n distinct thresholds. This algorithm is based on the following idea:

Algorithm 8 0.63-competitive ratio algorithm [20]

Let $< \tau_1, \tau_2, \ldots, \tau_n >$ be a sequence of thresholds.
for $k = 1$ to n **do**
  if $X_{\pi_k} \geq \tau_k$ **then**
    Let $Y = X_{\pi_k}$ and exit the for loop.
  end if
end for

The above mentioned thresholds are non-adaptive(the algorithms is oblivious to the history) and non-increasing. This choice is made intuitively because as we move to the end of available variables, we should be more pessimistic and lower the thresholds so as to ensure we select one of the remaining higher values.

Later, Correa et al.[12] proved that the same factor of $1-\frac{1}{e}$ can be obtained with a personalized but nonadaptive sequence of thresholds, that is thresholds $\tau_1, ..., \tau_n$ such that whenever variable V_i is shown the gambler stops if its value is above τ_i. In the case in which the probability of having two X_i's being the maximum is zero the algorithm is quite simple:

Algorithm 9 $1 - \frac{1}{e}$-competitive ratio algorithm [12]

Compute q_i = probability that Xi is the maximum.
Discard variable X_i with probability $1 - \frac{2}{2+(e-2)q_i}$.
Set threshold $\tau_i = F_i^{-1}(1-q_i)$.
Keep first random variable whose realization is at least τ_i.

In the general situation we apply an arbitrary tie-breaking rule so that $\sum q_i = 1$.

5.4.4 Surpassing the $1 - \frac{1}{e}$ barrier

In 2018 Azar et al. [9] proved that the lower bound of $1 - \frac{1}{e}$ that was proposed by Esfandiari et al. [20] is not tight. More specifically they showed that there exists an algorithm for the Prophet Secretary with competitive ratio larger than $1 - \frac{1}{e} + \frac{1}{400}$. The algorithm proposed by Esfandiari et al. is oblivious to the probability distributions of $X_1, \ldots, X_n$, only requires knowledge of $E[max_i X_i]$, which it competes against and chooses its thresholds deterministically. Therefore it is called a deterministic distribution-insensitive algorithm and as proved by Azar et al. [9] such algorithm cannot have a competitive ratio larger than $\frac{11}{15} \simeq 0.733$. This observation improves the upper bound of 0.746 that was found by Hill and Hertz [26] for the IID Prophet Inequality, which as we mentioned above if the variables are IID then the Prophet Inequality is equivalent to the Prophet Secretary.

Azar et al. [9] improved Esfandiari et al.'s algorithm by taking advantage of the fact that $E[max_i X_i] = \int_0^\infty Pr[max_i X_i \geq x]dx$ and that every interval $I \subseteq \mathbb{R}$ contributes the value $\int_{x \in I} Pr[max_i X_i \geq x]dx$ to $E[max_i X_i]$. Their approach involves categorizing Prophet Secretary instances into 3 groups:

1. The first category encompasses instances in which the contribution of the interval $[0, 1 - \frac{1}{e}]$ to $E[max_i X_i]$ is minimal.
2. In the second category they include instances, in which, in expectation, more than one X_i's exceed a certain threshold.
3. The third category, the most pivotal, comprises all remaining instances. They demonstrate that one of the X_i's (wlog X_1) is larger than the rest with high probability and possesses a sufficiently high expectation. For these instances, their algorithm applies the same threshold to all samples. They establish that with high probability, the algorithm selects a sample after encountering X_1. As a result, it extracts most of the expected value of X_1 as its profit. Furthermore, the algorithm typically encounters roughly half of the other X_i's before encountering X_1 due to the uniformly random order in which samples arrive. Consequently, even in the unlikely event that one of the other X_i's exceeds the threshold, the algorithm still manages to capture its value with a probability close to $\frac{1}{2}$.

As we can see, the improved algorithm proposed by Azar et al. is sensitive to the distributions of $X_1, X_2, \ldots, X_n$.

5.4.5 Utilizing Blind Strategies

Correa et al. [14] introduced a novel class of algorithms known as blind strategies, which enable them to achieve a competitive ratio of approximately $1 - \frac{1}{e} + \frac{1}{27} \simeq 0.669$ in the context of the prophet secretary problem. Additionally, their research includes a proof demonstrating that the upper bound cannot exceed $\sqrt{3} - 1 \simeq 0.732$. This result effectively distinguishes the prophet secretary problem from the IID prophet inequality, as the upper bound for the prophet secretary problem is strictly lower than what can be achieved in the IID case.

Essentialy blind strategies set a nonincreasing sequence of thresholds that depends only on the distribution of the maximum of the random variables, and the gambler stops the first time a sample surpasses the threshold of the stage. The algorithms proposed by Correa et al. exhibit remarkable robustness, constituting a generalization of single-threshold algorithms into multi-threshold strategies. Their inspiration draws from Ehsani et al.'s pioneering work [18], which initially computed a threshold value, denoted as τ, satisfying the condition: $P(max\{V_1, V_2, \ldots, V_n\} \leq \tau) = \frac{1}{e}$. This τ was employed as a single-threshold strategy, prompting the gambler to stop the moment any observed value exceeded τ. However, Ehsani et al. noted that this strategy exclusively applied to random variables with continuous distributions. Nevertheless, by incorporating a degree of randomization, they extended the strategy to accommodate general random variables. Instead of fixing a single acceptance probability, they introduced a function $\alpha : [0, 1] \to [0, 1]$. This function was employed to establish a sequence of thresholds as follows: Given an instance with n continuous distributions they draw uniformly and independently n random values in $[0, 1]$, and reorder them as $u[1] < \cdots < u[n]$. Then they set thresholds $\tau_1, \ldots, \tau_n$ such that $P(max\{V_1, \ldots, V_n\} \leq \tau_i) = \alpha(u[i])$ and the gambler stops at time i if $V_{\sigma_i} > \tau_i$. Notably, if the function α was nonincreasing, this led to a nonincreasing sequence of thresholds.

The concept of blind strategies originates from the aforementioned algorithm. What distinguishes these strategies is that, despite decisions being time-dependent, this temporal reliance solely hinges on the selection of the function α, which remains independent of the specific instance. From a technical perspective, these strategies leverage Schur convexity [34]. Notably, the algorithms mentioned earlier correspond to a blind strategy with $a(\cdot) = \frac{1}{e}$. Through their rigorous analysis, Correa et al. establish that the probability of the gambler obtaining a value exceeding τ is at least as great as the probability of the maximum exceeding τ, rescaled by a factor of $1-\frac{1}{e}$. This result draws upon Schur convexity to deduce that if a value surpasses the threshold τ, the gambler's probability of selecting it is no less than $1-\frac{1}{e}$. They extend their analysis to work with more general functions α,which require precise bounds on the distribution of the stopping time corresponding to a function α.

They present two lower bounds on the performance of blind strategies:

1. In the first case, they optimize the choice of α by solving an ordinary differential equation, resulting in a guarantee of 0.665.

2. In the second case, employing a refined analysis, they derive the stated bound of 0.669.

While their general approach may seem to have room for further enhancement, they prove that blind strategies cannot surpass a factor of 0.675. This is demonstrated by selecting two specific instances where no blind strategy can excel in both cases.

Furthermore, they establish an upper bound on the performance of any algorithm. By constructing a non-i.i.d. instance, they demonstrate that no algorithm can outperform $\sqrt{3} - 1 \simeq 0.732$. This surpasses the previously known best bound of 0.745, which applies to the i.i.d. case and was proven by Hill and Kertz [26]. Additionally, their result improves and extends Azar et al.'s recent bound of 11/15 ≈ 0.733 for the more restricted class of Deterministic distribution-insensitive algorithms. Their work establishes a previously unknown separation between the prophet secretary problem and the i.i.d. prophet inequality.

5.4.6 Improving hardness results

More recently, Bubna et al.[10] by utilizing a brute force numerical simulation and increasing the support size of the IID variables improved the hardness result form 0.732 to 0.7254. Based on this result Giambartolomei et al.[22] used the same idea to show a hardness of 0.7235, however, this result only applies to the order-unaware setting. To show the separation result they rely on an innovative asymptotic analysis of the optimal algorithm's acceptance thresholds, computed via backward induction, in a random arrival order setting, so as to obtain upper bounds on the competitive ratio of the optimal algorithm.

5.4.7 Trying to beat the online optimal

Up until now, we put the gambler against a prophet who knows all the data in advance and thus is able to make an optimal decision. More recently Dutting et al. [16] proposed a different benchmark, the online optimal. This benchmark does not assume any prior knowledge of the future; the gambler competes against an algorithm that has the same information as him at every step, but infinite computation power. This new concept enables us to measure the potential loss that arises due to computational limitations, rather than quantifying the loss that's due to the fact that the algorithm has to make decisions online. Dutting et al. [16] managed to approximate the expected value of the online optimal to within a factor of 1 − ε, which immediately translates into an algorithm that achieves a $(1-\epsilon)$-approximation.

Dutting et al. [16] leverage the observation that when you cluster "similar" variables into g groups and handle variables in each group uniformly, you can construct a dynamic program to monitor the count of variables in each group. However, this approach results in exponential complexity in g. The key challenge is then to demonstrate the existence of a concise grouping that achieves a $(1-\epsilon)$-approximation. For the QPTAS, it is adequate to provide a grouping of size polylog(n), but for the PTAS, it is essential to further reduce this to O(1).

- **QPTAS**

  Dutting et al. start by addressing a special case, where each random variable has binary support as follows: variable X_i is either v_i or zero, with probabilities p_i and $1-p_i$ respectively. Firstly, they scale the values appropriately so that OPT falls into some small constant interval $[c, 1]$. Secondly, they show that they don't lose more than $O(\epsilon)$ if they ignore variables with low value $(v_i \leq \epsilon)$ or low expected value $(v_i p_i \leq \frac{\epsilon}{n})$. The remaining variables with high values $v_i \geq \frac{1}{\epsilon^2}$ are compressed by adjusting their values so that they all have the same value and their probabilities fall in a O(poly n) range. Variables with small values (from ϵ to $\frac{1}{\epsilon^2}$) and probabilities ranging from $\frac{\epsilon^3}{n}$ to 1 are discretized into powers of $(1+\epsilon)$.

This construction can be readily extended to constant-size support cases by treating high realizations similarly to the single case. To handle support sizes that are not necessarily constant, an argument is presented for collapsing all high realizations (above OPT) of a variable into a single point. Combined with discretization for low realizations (from ϵ to 1), this brings us back to the constant-size support case.

- **PTAS**

  In a manner similar to the QPTAS case, the approach taken for the PTAS involves considering scenarios where all variables have binary values, which encapsulates the general strategy. Starting with the discretization techniques from the QPTAS, the challenge lies in managing variables with individually small realization probabilities (< poly(ϵ)) that cannot be disregarded since their combined impact on the optimal reward can be significant. This step is crucial to reduce the number of distinct probabilities to poly($\frac{1}{\epsilon}$).

  To address this, a novel method called ”frontloading” is introduced. It works like this: We fix a support value v, and consider the variables of interest (those with neither high nor low probabilities) with that support. If it is the case that k many of these variables have a total realization probability that is not very high, then we claim that imagining these k variables as a single box, with the total realization probability equal to that of the k boxes, and as an ”outside option” always available through the interval where these variables arrive, does not affect the reward much. This innovation significantly reduces the variety of probabilities that must be tracked in the dynamic program to find the optimal solution.

  To extend this approach to variables with support sizes greater than one, it is shown that a variable with multiple support values can be envisioned as a collection of binary variables (each of them having one of the support values with its corresponding probability that is not too high, and the remaining probability on value 0) and an additional variable with high probability for each support value that occurs sequentially. This simplifies the problem back to the binary case, as there are only a limited number of variable types with higher probabilities on each support value, allowing for distinct treatment.

5.4.8 Minimization variant of Prophet Secretary

The result mentioned in section 4.4 for the I.I.D. case of the cost minimization prophet inequality also holds for the for the minimization variant of the prophet secretary problem as well. Lastly, it was also proved that for any large number C, there is no C-competitive algorithm for minimization prophet secretary with one exchange.

5.5 Variables drawn from unknown distributions

Up to this point, we have studied the Prophet Inequality problem with the assumption that we have knowledge of the distributions of the variables. The decision about when to stop and accept the realization of a variable depends only on the values of the random variables $X_1, \ldots, X_t$ and on the distribution D. The case where D is unknown, such that the decision may depend only on on the values of the random variables is equally interesting but it has received little attention.

The primary drive for investigating this issue stems from its relevance in contemporary applications of prophet inequalities, particularly their role in evaluating posted-price mechanisms and reserve pricing in advertising auctions. In these scenarios, it is typical to depict valuations as samples drawn from an underlying distribution. However, it might not be feasible to presume that the auctioneer is aware of this distribution. Nonetheless, the auctioneer may choose to learn the distribution on the fly as opportunities arrive, or may possess some limited historical information in the form of additional samples.

Despite the obvious appeal of the problem, which was noted by Azar et al.[8], little was known about this setting up until recently. The latest research on this setting is due to Correa et al. [15], who considered the prophet problem in which values are drawn independently from a single unknown distribution, and asked which approximation guarantees can be obtained relative to the expected maximum value in hindsight. Unlike the known distribution setting where an optimal stopping rule can be obtained via backwards induction, in the unknown distribution setting, there is no clear candidate. The challenges of this problem stem from the fact that we hope to be able to learn something for future values from earlier ones.

5.5.1 Sublinear number of samples

For $o(n)$ samples the prophet problem behaves like the secretary porblem.

5.5.1.1 Achieving a $\frac{1}{e}$-bound without samples

Correa et al. [15] achieved a guarantee of $\frac{1}{e}$ by employing the optimal stopping rule for the secretary problem. This rule ensures that it will stop on the maximum value with a probability of at least $\frac{1}{e}$. As demonstrated in section 5.4, this implies a 1/e-approximation concerning the expected maximum. To provide a quick recap of the strategy used for the secretary problem: we discard a $\frac{1}{e}$ fraction of the values and then accept the first value that surpasses the maximum among the discarded values.

This algorithm doesn't require any samples, is guaranteed to stop at the maximum of the sequence with a probability of $\frac{1}{e}$, and can also be demonstrated to offer a $\frac{1}{e}$ approximation for our objective. However, this analysis does not consider the fact that all values originate from the same distribution, thus neglecting any potential learning opportunities.

Finally, the authors prove that no learning of the distribution is feasible, and the straightforward guarantee of $\frac{1}{e}$ is, in fact, the best possible in the prophet setting. In simpler words, $o(n)$ samples are not enough to improve on the bound of $\frac{1}{e}$.

5.5.2 Linear number of samples

They proceeded to show that there is a sharp phase transition when going from $o(n)$ samples to Ω(n) samples, by giving an algorithm that uses as few as $n-1$ samples and improves the lower bound from $\frac{1}{e}$ to $1 - \frac{1}{e} \simeq 0.632$. This bound is tight for two different classes of algorithms that share certain features of their proposed algorithm.

5.5.2.1 Obtaining a $\frac{1}{2}$-approximation with $n-1$ samples

With the help of the below algorithm they show that if the stopping rule has access to $n-1$ samples, then we can simply take the maximum of these samples as a single, non-adaptive threshold for all random variables to obtain a factor $\frac{1}{2}$-approximation.

Algorithm 10 Fresh-looking samples [15]

Data: Sequence of i.i.d. random variables $X_1, \ldots, X_n$ sampled from an unknown distribution D, sample access to D
Result: Stopping time τ
$S_1, \ldots, S_{n-1} \leftarrow n-1$ independent samples from D
$S \leftarrow \{S_1, S_2, \ldots, S_{n-1}\}$
for $t = 1$ to n **do**
  if $X_t \geq maxS$ **then**
    return t
  else
    $S \leftarrow$ random subset size $n-1$ of $\{S_1, \ldots, S_{n-1}, X_1, \ldots, X_t\}$
  end if
end for
return $n+1$

In order to gain some intuition they designed an algorithm that samples $n-1$ values $S_1, \ldots, S_{n-1}$ from D , uses the maximum of these as a uniform threshold for all of the random variables $X_1, \ldots, X_n$, and accepts the first random variable that exceeds this threshold. The expected value that can be collected from any random variable X_t conditioned on stopping at that random variable is at least $E[max\{X_1, \ldots, X_n\}]$, since under this condition X_t is the maximum of at least n i.i.d. random variables. The approximation guarantee of this algorithm is $\frac{1}{2} + \frac{1}{4n-2}$.

5.5.2.2 Achieving a $1 - \frac{1}{e}$ approximation with $n-1$ samples

Correa et al. [15] demonstrated that it is feasible to achieve an enhanced bound of $1 - \left(1 - \frac{1}{n}\right)^n \geq 1 - \frac{1}{e} \approx 0.632$ with just $n-1$ samples. They accomplished this by refining the naive algorithm mentioned above, by increasing the probability of stopping, all while maintaining the property that the expected value collected when stopping is at least $E[max\{X_1, \ldots, X_n\}]$.

The stopping rule they use to achieve this bound is as follows: The rule starts by drawing $n-1$ samples. Then, when considering the ith random variable for $i \geq 1$, it also considers a random subset of size $n-1$ drawn uniformly from the $n-1$ initial samples and the $i-1$ random variables seen so far. If the ith random variable is greater than the maximum of that random subset the rule stops, otherwise it continues with the next random variable. While the stopping rule itself is easy to describe, its analysis relies on an insight that is somewhat subtle. Indeed, each of the sets of random variables used to set a threshold for acceptance is distributed like a set of $n-1$ fresh samples from the distribution. The expected value collected from each random variable, conditioned on its acceptance, thus equals the expected maximum value of n independent draws from the distribution, and the probability of accepting a random variable conditioned on reaching it is exactly $\frac{1}{n}$. The approximation guarantee is then equal to the overall probability of stopping, which is at least $1 - \frac{1}{e}$.

They claim that while an improvement over the bound of $1-\frac{1}{e} \approx 0.632$ remains conceivable via more complicated stopping rules, such an improvement cannot go beyond $ln(2) \approx 0.693$.

5.5.3 Superlinear number of samples

Correa et al. [15] also considered the case where we have access to $O(n^2)$ samples. With this larger sample size, it becomes possible to approach the optimal guarantee of approximately 0.745, which is achievable when the distribution is fully known. This is achieved by mimicking the stopping rule that attains that bound, which uses a decreasing sequence of thresholds corresponding to conditional acceptance probabilities that increase over time. However, instead of using the true distribution, they use quantiles from the empirical distribution. By initially discarding a constant fraction of the values and leveraging the DKW inequality to demonstrate the simultaneous concentration of all empirical quantiles, they manage to reduce the required number of samples from $O(n^4)$ to $O(n^2)$. This is in contrast to the more straightforward approach, which potentially stops on any of the values and relies on Chernoff and union bounds to establish concentration.

5.5.4 Final results

In conclusion, the results they obtained are:

- A straightforward guarantee for the case of $\alpha \geq \frac{1}{e} \approx 0.368$ can be derived from the well-known optimal solution to the secretary problem and this bound is actually tight
- The stopping time may depend on a limited number of samples from D, and even with $o(n)$ samples we get an appoxrimation of $\alpha \leq \frac{1}{e}$, meaning we cannot improve on the bound of $\frac{1}{e}$, with just $o(n)$ samples
- If we have access to n samples, the approximation improves significantly: $\alpha \geq 1-\frac{1}{e} \approx 0.632$ and $\alpha \leq ln(2) \approx 0.693$
- If we have $O(n^2)$ samples, this is equivalent to knowledge of the distribution and we obtain $\alpha \geq 0.745 - \epsilon$

6. CONCLUSIONS AND FUTURE WORK

Throughout this thesis, the problem of Prophet Inequality was explored, which at its core is a combination of online decision-making and stochasticity. One of the most basic economic problems is that of eliciting information to make optimal decisions. The prophet inequality problem is a natural extension of this problem, where the decision-maker is faced with a sequence of random variables and must decide when to stop and accept the realization of a random variable.

In Chapter 2, we established foundational theoretical knowledge essential for comprehending the prophet inequality and its significance in contemporary decision-making processes. Specifically, we outlined the core challenge of optimal stopping theory, which focuses on determining the ideal timing for taking actions based on sequentially observed random variables to either maximize rewards or minimize costs. Additionally, we elucidated the distinction between game theory and mechanism design: the former explores how entities can influence multiple outcomes, while the latter centers on how to achieve specific outcomes. Our exploration extended to price mechanisms, particularly highlighting the connection between posted price mechanisms and the prophet inequality. Furthermore, we conducted an explanation of competitive analysis, by delineating offline and online optimization and introducing the competitive ratio as a crucial metric employed throughout the thesis to evaluate the efficiency of proposed online algorithms. Finally, we emphasized the disparity between adaptive and non-adaptive algorithms and presented key statistical functions and definitions sourced from the problem's bibliography.

In chapter 3, we explored in depth the Prophet Inequality problem for the reward maximization setting. We started by presenting the classic algorithm of Krengel and Sucheston [30], who established a fundamental lower bound wherein the gambler can consistently secure at least half of the expected prophet's reward. This lower bound is the foundation of all further research. In section 3.3 we extensively compared the selection between order types: the gambler's power to determine the arrival order of each item versus the random order variant. The latter, when it is employing independent and identically distributed (iid) variables, aligns with the Prophet Secretary problem and is detailed in section 5.4. As fas as order selection is concerned, we highlighted its NP-hardness even under a special case of a 3-point distribution, where the highest and lowest points of the support are the same for all the distributions. At first, the lower bound of the optimal factor was 0.669 and the upper bound 0.745, but as more studies focused on order selection this gap has began to get smaller. We explained that the bound of 0.632 is the best possible for both random and free-order prophet inequalities and it is also the best ratio attainable for threshold stopping rules. When we have iid variables, we present that the first to surpass the $\frac{1}{2}$ barrier of prophet inequality was a 0.6321-competitive algorithm based on complicated recursive functions, later an approximate single threshold 0.7380-competitive algorithm was found, after which the tight optimal competitive ratio of 0.7451 was finally attained. In the non-iid case we detailed an algorithm that achieves a 0.7380-competitive ratio when each distribution in the instance occurs m times, for a sufficiently large m. Additionally, we showcased that offering the gambler a set of predetermined permutations for indexing the random variables enhanced the competitive ratio to the inverse of the golden ratio, even with just 2 permutations. However, despite having the choice among the $n!$ permutations, the ratio could not surpass the bound of $1 - \frac{1}{e}$. We then demonstrated an algorithm centered on arrival time design, which improves the competitive ratio of order selection to 0.725, reaching 0.745 for the iid setting. Finally, we explained how the competitve ratio for order selection was improved from 0.7251 to 0.7258, and that it can be more beneficial for the gambler to choose the order as random order was proven to be bounded by 0.7325.

In chapter 4, we turned our focus on the cost minimization variant. Contrary to our initial intuition, the cost minimization setting proves to be significantly more challenging than the maximization one, primarily due to its constraints being upwards-closed in nature. At first, we mentioned that algorithms fail to achieve any bounded approximation for the non-iid advarsarial or random order instance. However, in the iid case, it was demonstrated that the competitive ratio of any online algorithm is bounded by $\frac{(1.11)^n}{6}$. Following this result, we presented an algorithm that achieves a $O(polylogn)$-factor approximation using only one threshold. Furthermore, we note that when employing multiple threshold algorithms, it is possible to attain a distribution-dependent constant factor CPI for the class of Entire distributions, and a 2-factor CPI for Entire MHR distributions.

In chapter 5 we discuss the progress that has been made in various variations of the original problem. For the multiple choices prophet inequality, the current state-of-the-art algorithm achieves a competitive ratio of $(1-\frac{1}{\sqrt{k+3}})$. In the matroid prophet inequality it has been established that there exists an online algorithm whose expected payoff is at least half of the expected weight of the maximum weight basis. Furthermore, in the context of the matching prophet inequality, the most effective algorithm discovered to date offers an approximation that is nearly tight, approaching $(1-\frac{1}{\sqrt{k+3}})$. Additionally, we presented the prophet secretary problem, which equates to the random order prophet inequality. Initially, an algorithm was devised establishing a bound of $1-\frac{1}{e}$, which defines a nonincreasing sequence of n thresholds $\tau_1, \dots, \tau_n$ that only depend on the expectation of the maximum of the V_i's and on n. Subsequently, it was proved that the same factor can be obtained with a personalized but nonadaptive sequence of thresholds and with a single threshold algorithm(having to randomize to break ties in some situations). However, this bound was proven to be not tight, as an algorithm that relies on some subtle analysis obtains $1-\frac{1}{e}+\frac{1}{400}$. Notably, blind strategies improved upon the previous results by obtaining a constant of 0.669, but cannot surpass 0.675. Additionally, it was established that no online algorithm could achieve better than $\sqrt{3}-1$, marking the first distinction between the prophet secretary and the iid prophet inequality. Finally, we examined scenarios where a gambler faces a computer possessing equivalent knowledge but has infinite computational power. In this setting, it was proven that an algorithm exists with a $(1-\epsilon)$ -approximation. Regarding the minimization variant of the prophet secretary, analogous results to those presented in section 4.4 of the iid prophet inequality apply. The last variation we studied was the one where variables are drawn from unknown distributions. A direct guarantee of 0.368 stems from the optimal solution to the secretary problem in this context. Additionally, we examined cases where the stopping time relies on a limited number of samples from F, demonstrating that even with $o(n)$ samples, $\alpha \leq \frac{1}{e}$. Conversely, leveraging n samples yields a notable enhancement: $\alpha \geq 1-\frac{1}{e} \approx 0.632$ and $\alpha \leq ln(2) \approx 0.693$. Finally, possessing $O(n^2)$ samples equates to having knowledge of the distribution, resulting in $\alpha \geq 0.745-\epsilon$ for any $\epsilon > 0$.

These detailed examinations underscore the multifaceted nature of the Prophet Inequality Problem, shedding light on its complexities and potential applications. While this research contributes significantly to understanding and addressing this intricate problem, avenues for future research remain, including exploring more practical applications and refining algorithms to achieve tighter competitive ratios.

ABBREVIATIONS - ACRONYMS

GT	Game Theory
MD	Mechanism Design
QPTAS	Quasi Polynomial Time Approximation Scheme
PTAS	Polynomial Time Approximation Scheme
FPTAS	Fully Polynomial Time Approximation Scheme
MHRD	Monotone Hazard Rate Distribution
CPI	Cost Prophet Inequality
PPMs	Posted-Price Mechanisms

BIBLIOGRAPHY

[1] Melika Abolhasani, Soheil Ehsani, Hossein Esfandiari, Mohammad Taghi HajiAghayi, Robert Kleinberg, and Brendan Lucier. Beating 1-1/e for ordered prophets. *STOC 2017: Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, (61-71), June 2017.

[2] Shipra Agrawal, Jay Sethuraman, and Xingyu Zhang. On optimal ordering in the optimal stopping problem. *EC '20: Proceedings of the 21st ACM Conference on Economics and Computation*, pages 187–188, July 2020.

[3] S. Alaei, M. Hajiaghayi, and V. Liaghat. Online prophet-inequality matching with applications to ad allocation. *EC'12*, pages 18–35, June 2012.

[4] Saeed Alaei. Bayesian combinatorial auctions: Expanding single buyer mechanisms to many buyers. *FOCS'11*, pages 512–521, October 2011.

[5] K.J Arrow, D. Blackwell, and M.A. Girschick. Bayes and minimax solutions of sequential decision problems. *Econometrica*, 13(3/4):213–244, July-October 1949.

[6] Makis Arsenis, Odysseas Drosis, and Robert Kleinberg. Constrained-order prophet inequalities. *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, (2034-2046), 2021.

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

[8] P. D. Azar, R. Kleinberg, and S. M. Weinberg. Prophet inequalities with limited information. *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1358–1377, 2014.

[9] Yossi Azar, Ashish Chiplunkar, and Haim Kaplan. Prophet secretary: Surpassing the 1-1/e barrier. *EC '18*, pages 303–318, June 2018.

[10] Archit Bubna and Ashish Chiplunkar. Prophet inequality: Order selection beats random order. *EC '23: Proceedings of the 24th ACM Conference on Economics and Computation*, pages 302–336, May 2023.

[11] S. Chawla, J.D. Hartline, D.L. Malec, and B. Sivan. Multi-parameter mechanism design and sequential posted pricing. *STOC'10*, 2010.

[12] José Correa, Patricio Foncea, Ruben Hoeksma, Tim Oosterwijk, and Tjark Vredeveldd. Posted price mechanisms for a random stream of customers. *EC '17: Proceedings of the 2017 ACM Conference on Economics and Computation*, pages 169–186, June 2017.

[13] Jose Correa, Patricio Foncea, Dana Pizarro, and Victor Verdugo. From pricing to prophets, and back! *Operations Research Letters*, 44(1):25–29, January 2019.

[14] Jose Correa, Raimundo Saona, and Bruno Ziliotto. Prophet secretary through blind strategies. *Mathematical Programming*, 190(483-521), 2021.

[15] Jose R. Correa, Paul Dutting, Felix Fischer, and Kevin Schewior. Prophet inequalities for i.i.d. random variables from an unknown distribution. *EC '19*, pages 3–17, June 2019.

[16] Paul Dutting, Evangelia Gergatsouli, Rojin Rezvan, Yifeng Teng, and Alexandros Tsiigonias-Dimitriadis. Prophet secretary against the online optimal. *EC '23*, pages 561–581, July 2023.

[17] E. B. Dynkin. Optimal choice of the stopping moment of a markov process. *Sov. Math Dokl.*, 4:627–629, 1963.

[18] S. Ehsani, M. Hajiaghayi, T. Kesselheim, and S. Singla. Prophet secretary for combinatorial auctions and matroids. *Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms*, (700-714), 2018.

[19] E.Samuel-Cahn. Comparison of threshold stop rules and maximum for independent nonnegative random variables. *The Annals of Probability*, pages 1213–1216, 1984.

[20] Hossein Esfandiari, Mohammad Taghi Hajiaghayi, Vahid Liaghat, and Morteza Monemizadesh. Prophet secretary. *CoRR*, abs/1507.01155, 2015.

[21] Thomas S. Ferguson. *Mathematical Statistics: A Decision Theoretic Approach*. 1967.

[22] Giordano Giambartolomei, Frederik Mallmann-Trenn, and Raimundo Saona. Prophet inequalities: Separating random order from order selection. *arXiv:2304.04024*, August 2023.

[23] M. Hajiaghayi, R. Kleinberg, and T. Sandholm. Automated online mechanism design and prophet inequalities. *AAAI*, 7:58–65, 2007.

[24] Theodore P Hill and Arie Hordijk. Selection of order of observation in optimal stopping problems. *Journal of applied probability*, 22(1):177–184, 1985.

[25] T.P. Hill. Prophet inequalities and order selection in optimal stopping problems. *Proceedings of the American Mathematical Society*, 88(1), May 1983.

[26] T.P. Hill and Robert P. Kertz. Comparisons of stop rule and supremum expectations of i. i. d. random variables. *The Annals of Probability*, 10(2):336–345, May 1982.

[27] D.P Kennedy. Prophet-type inequalities for multi-choice optimal stopping. *Stochastic Processes and their Applications*, 24(1):77–88, February 1987.

[28] R. P. Kertz. Stop rule and supremum expectations of i.i.d. random variables: A complete comparison by conjugate duality. *Journal of Multivariate Analysis*, 19:88–112, 1986.

[29] Robert Kleinberg and S. Matthew Weinberg. Matroid prophet inequalities. *STOC'12*, pages 123–136, May 2012.

[30] Ulrich Krengel and Louis Sucheston. Semiamarts and finite values. *Bulletin of the American Mathematical Society*, 83(4), July 1977.

[31] Vasilis Livanos and Ruta Mehta. Prophet inequalities for cost minimization. *https://doi.org/10.48550/ARXIV.2209.07988*, https://arxiv.org/abs/2209.07988 2022.

[32] M. S. Manasse, L. A. McGeoch, and D. D. Sleator. Competitive algorithms for on-line problems. *STOC'88*, pages 322–333, January 1988.

[33] Roger B. Myerson. Optimal auction design. *Mathematics of Operations Research*, 6(1):58–73, 1981.

[34] J. Pecaric, F. Proshman, and Y. Tong. Convex functions, partial orderings, and statistical applications. *Academic Press*, 1992.

[35] Bo Peng and Zhihao Gavin Tang. Order selection prophet inequality: From threshold optimization to arrival time design. *FOCS 2022*, pages 171–178, 2022.

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

[37] Abraham Wald. *Sequential Analysis*. J. Wiley and sons, 1945.

[38] Abraham Wald. Statistical decision functions. *The annals of Mathematical Statistics*, 20(2):165–205, June 1949.

[39] Q. Yan. Mechanism design via correlation gap. *Proceedings of the Twenty- Second Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 710–719, January 2011.