An Introduction to Prophet Inequalities

Vasilis Livanos

Theory and Algorithms Group
Department of Computer Science
University of Illinois at Urbana-Champaign

livanos3@illinois.edu

October 5th, 2020

Overview

Motivation

Motivation

- Suppose we want to sell an orange. We know we will see n potential buyers, one after the other in some order, and we have to decide whether to sell to buyer i before we see buyer $i + 1$.

Motivation

- Suppose we want to sell an orange. We know we will see n potential buyers, one after the other in some order, and we have to decide whether to sell to buyer i before we see buyer $i+1$.
- The buyers have some private valuations $v_1, \ldots, v_n$ for the orange. How do we decide on what prices to offer?

Motivation

Motivation

- If buyer i buys the orange at price p and values it at v_i, then they receive utility $v_i - p$, and we receive revenue p.

Motivation

- If buyer i buys the orange at price p and values it at v_i, then they receive utility $v_i - p$, and we receive revenue p.
- We want to maximize the social welfare (the sum of utilities, which includes our revenue), and want to compare against the best possible offline decision.

Motivation

- If buyer i buys the orange at price p and values it at v_i, then they receive utility $v_i - p$, and we receive revenue p.
- We want to maximize the social welfare (the sum of utilities, which includes our revenue), and want to compare against the best possible offline decision.
    1. If we assume the buyers arrive in *worst-case order* and their valuations for the orange are *arbitrary*, then we cannot achieve any meaningful competitive ratio.

Motivation

- If buyer i buys the orange at price p and values it at v_i, then they receive utility $v_i - p$, and we receive revenue p.
- We want to maximize the social welfare (the sum of utilities, which includes our revenue), and want to compare against the best possible offline decision.
  1. If we assume the buyers arrive in *worst-case order* and their valuations for the orange are *arbitrary*, then we cannot achieve any meaningful competitive ratio.
  2. Assume they arrive in *worst-case order* but their valuations are drawn independently from distributions $D_1, \ldots, D_n$. $\implies$ *Prophet Inequality Problem*

Motivation

- If buyer i buys the orange at price p and values it at v_i, then they receive utility $v_i - p$, and we receive revenue p.
- We want to maximize the social welfare (the sum of utilities, which includes our revenue), and want to compare against the best possible offline decision.
  1. If we assume the buyers arrive in *worst-case order* and their valuations for the orange are *arbitrary*, then we cannot achieve any meaningful competitive ratio.
  2. Assume they arrive in *worst-case order* but their valuations are drawn independently from distributions $D_1, \ldots, D_n$. $\implies$ *Prophet Inequality Problem*
  3. Assume their valuations are *arbitrary*, but they arrive in *random order*. $\implies$ *Secretary Problem**

Prophet Inequality Problem (1/2)

Prophet Inequality Problem (1/2)

- We are given n non-negative random variables $X_1, \ldots, X_n$ and their distributions $D_1, \ldots, D_n$. We see a realization from each X_i in *adversarial order*.

Prophet Inequality Problem (1/2)

- We are given n non-negative random variables $X_1, \ldots, X_n$ and their distributions $D_1, \ldots, D_n$. We see a realization from each X_i in *adversarial order*.
- At every step i, when we see the realization of X_i, we have to immediately and irrevocably decide whether to
  1. select X_i and stop, or
  2. ignore X_i and continue to the next step.

Prophet Inequality Problem (1/2)

- We are given n non-negative random variables $X_1, \ldots, X_n$ and their distributions $D_1, \ldots, D_n$. We see a realization from each X_i in *adversarial order*.
- At every step i, when we see the realization of X_i, we have to immediately and irrevocably decide whether to
  1. select X_i and stop, or
  2. ignore X_i and continue to the next step.
- We want to select the highest possible value, and compare against $X^* = \max_{1 \leq i \leq n} X_i$ on expectation.

Prophet Inequality Problem (1/2)

- We are given n non-negative random variables $X_1, \ldots, X_n$ and their distributions $D_1, \ldots, D_n$. We see a realization from each X_i in *adversarial order*.
- At every step i, when we see the realization of X_i, we have to immediately and irrevocably decide whether to
  1. select X_i and stop, or
  2. ignore X_i and continue to the next step.
- We want to select the highest possible value, and compare against $X^* = \max_{1 \leq i \leq n} X_i$ on expectation.
- There exists an algorithm which selects a value V such that $\mathbb{E}[V] \geq \frac{1}{2}\,\mathbb{E}[X^*]$, and no algorithm can achieve better competitive ratio [KS77].

Prophet Inequality Problem (2/2)

Prophet Inequality Problem (2/2)

- Simple and elegant problem in optimal stopping theory, solved since the '70s. Unsurprisingly, there exist many algorithms for it.

Prophet Inequality Problem (2/2)

- Simple and elegant problem in optimal stopping theory, solved since the '70s. Unsurprisingly, there exist many algorithms for it.

$\mathcal{A}_T$: "Fixed-Threshold" Algorithm

Select a threshold T based on $D_1, \ldots, D_n$, and accept the first $X_i \geq T$.

Prophet Inequality Problem (2/2)

- Simple and elegant problem in optimal stopping theory, solved since the '70s. Unsurprisingly, there exist many algorithms for it.

$\mathcal{A}_T$: "Fixed-Threshold" Algorithm

Select a threshold T based on $D_1, \ldots, D_n$, and accept the first $X_i \geq T$.

- Select $T =$ median of the distribution of X^*, i.e. $\Pr[X^* \geq T] = \frac{1}{2}$ (assuming no point mass on T) [Sam84].

Prophet Inequality Problem (2/2)

- Simple and elegant problem in optimal stopping theory, solved since the '70s. Unsurprisingly, there exist many algorithms for it.

$\mathcal{A}_T$: "Fixed-Threshold" Algorithm

Select a threshold T based on $D_1, \ldots, D_n$, and accept the first $X_i \geq T$.

- Select $T =$ median of the distribution of X^*, i.e. $\Pr[X^* \geq T] = \frac{1}{2}$ (assuming no point mass on T) [Sam84].
- Select $T = \frac{1}{2}\,\mathbb{E}[X^*]$ [KW12].

A Grocer's Dilemma - Posted-Price Mechanisms (1/2)

A Grocer's Dilemma - Posted-Price Mechanisms (1/2)

- The grocer will set a price for the orange at the beginning of the day. The threshold T corresponds to the price posted on the orange by the grocer. At this point, the grocer might as well leave the shop.

A Grocer's Dilemma - Posted-Price Mechanisms (1/2)

- The grocer will set a price for the orange at the beginning of the day. The threshold T corresponds to the price posted on the orange by the grocer. At this point, the grocer might as well leave the shop.
- The first person who sees the orange and has valuation at least T will buy it.

A Grocer's Dilemma - Posted-Price Mechanisms (1/2)

- The grocer will set a price for the orange at the beginning of the day. The threshold T corresponds to the price posted on the orange by the grocer. At this point, the grocer might as well leave the shop.
- The first person who sees the orange and has valuation at least T will buy it.
- Prophet inequalities provide guarantees for *posted-price mechanisms* in online auctions. Crucially, PPMs do not require bidding.

A Grocer's Dilemma - Posted Price Mechanisms (2/2)

A Grocer's Dilemma - Posted Price Mechanisms (2/2)

- Posted-price mechanisms are

A Grocer's Dilemma - Posted Price Mechanisms (2/2)

- Posted-price mechanisms are
  1. **Anonymous:** All buyers are offered the same price, regardless of their type distribution.

A Grocer's Dilemma - Posted Price Mechanisms (2/2)

- Posted-price mechanisms are
  1. **Anonymous:** All buyers are offered the same price, regardless of their type distribution.
  2. **Static:** The choice of which price to offer which buyer does not change as the mechanism progresses.

A Grocer's Dilemma - Posted Price Mechanisms (2/2)

- Posted-price mechanisms are
  1. **Anonymous:** All buyers are offered the same price, regardless of their type distribution.
  2. **Static:** The choice of which price to offer which buyer does not change as the mechanism progresses.
  3. **Order-Oblivious:** The pricing rule does not depend on the order in which the buyers arrive, and in fact the order can be chosen by an adaptive adversary.

A Grocer's Dilemma - Posted Price Mechanisms (2/2)

- Posted-price mechanisms are
  1. **Anonymous:** All buyers are offered the same price, regardless of their type distribution.
  2. **Static:** The choice of which price to offer which buyer does not change as the mechanism progresses.
  3. **Order-Oblivious:** The pricing rule does not depend on the order in which the buyers arrive, and in fact the order can be chosen by an adaptive adversary.
  4. **Ex-post Individually Rational:** No buyer is worse if they come to the grocery store and see the orange than if they did not participate at all.

A Grocer's Dilemma - Posted Price Mechanisms (2/2)

- Posted-price mechanisms are
  1. **Anonymous:** All buyers are offered the same price, regardless of their type distribution.
  2. **Static:** The choice of which price to offer which buyer does not change as the mechanism progresses.
  3. **Order-Oblivious:** The pricing rule does not depend on the order in which the buyers arrive, and in fact the order can be chosen by an adaptive adversary.
  4. **Ex-post Individually Rational:** No buyer is worse if they come to the grocery store and see the orange than if they did not participate at all.
  5. **Strategy-Proof:** There is no incentive for buyers to misreport, because we do not even ask them for a bid!

Proof of the Prophet Inequality

$\Pr[X^* \geq T] = \frac{1}{2}$. Let $\mathcal{E}_i$ be the event we "reach" the i-th random variable.

Proof of the Prophet Inequality

$\Pr[X^* \geq T] = \frac{1}{2}$. Let $\mathcal{E}_i$ be the event we "reach" the i-th random variable.

$$\mathbb{E}[V] = T \Pr[X^* \geq T] + \sum_{i=1}^{n} \Pr[\mathcal{E}_i] \cdot \mathbb{E}[(X_i - T)^+]$$

Proof of the Prophet Inequality

$\Pr[X^* \geq T] = \frac{1}{2}$. Let $\mathcal{E}_i$ be the event we "reach" the i-th random variable.

$$\mathbb{E}[V] = T \Pr[X^* \geq T] + \sum_{i=1}^{n} \Pr[\mathcal{E}_i] \cdot \mathbb{E}[(X_i - T)^+]$$

$$= T \Pr[X^* \geq T] + \sum_{i=1}^{n} \Pr\left[\max_{1 \leq j \leq i-1} X_i < T\right] \cdot \mathbb{E}[(X_i - T)^+]$$

Proof of the Prophet Inequality

$\Pr[X^* \geq T] = \frac{1}{2}$. Let $\mathcal{E}_i$ be the event we "reach" the i-th random variable.

$$\begin{aligned}
\mathbb{E}[V] &= T \Pr[X^* \geq T] + \sum_{i=1}^{n} \Pr[\mathcal{E}_i] \cdot \mathbb{E}[(X_i - T)^+] \\
&= T \Pr[X^* \geq T] + \sum_{i=1}^{n} \Pr\left[\max_{1 \leq j \leq i-1} X_i < T\right] \cdot \mathbb{E}[(X_i - T)^+] \\
&\geq T \Pr[X^* \geq T] + \sum_{i=1}^{n} \Pr[X^* < T] \cdot \mathbb{E}[(X_i - T)^+]
\end{aligned}$$

Proof of the Prophet Inequality

$\Pr[X^* \geq T] = \frac{1}{2}$. Let $\mathcal{E}_i$ be the event we "reach" the i-th random variable.

$$
\begin{aligned}
\mathbb{E}[V] &= T \Pr[X^* \geq T] + \sum_{i=1}^{n} \Pr[\mathcal{E}_i] \cdot \mathbb{E}[(X_i - T)^+] \\
&= T \Pr[X^* \geq T] + \sum_{i=1}^{n} \Pr\left[\max_{1 \leq j \leq i-1} X_i < T\right] \cdot \mathbb{E}[(X_i - T)^+] \\
&\geq T \Pr[X^* \geq T] + \sum_{i=1}^{n} \Pr[X^* < T] \cdot \mathbb{E}[(X_i - T)^+] \\
&= \frac{1}{2} T + \frac{1}{2} \mathbb{E}\left[\sum_{i=1}^{n} (X_i - T)^+\right]
\end{aligned}
$$

Proof of the Prophet Inequality

$\Pr[X^* \geq T] = \frac{1}{2}$. Let $\mathcal{E}_i$ be the event we "reach" the i-th random variable.

$$\begin{aligned}
\mathbb{E}[V] &= T\Pr[X^* \geq T] + \sum_{i=1}^{n} \Pr[\mathcal{E}_i] \cdot \mathbb{E}[(X_i - T)^+] \\
&= T\Pr[X^* \geq T] + \sum_{i=1}^{n} \Pr\left[\max_{1 \leq j \leq i-1} X_j < T\right] \cdot \mathbb{E}[(X_i - T)^+] \\
&\geq T\Pr[X^* \geq T] + \sum_{i=1}^{n} \Pr[X^* < T] \cdot \mathbb{E}[(X_i - T)^+] \\
&= \frac{1}{2}T + \frac{1}{2}\mathbb{E}\left[\sum_{i=1}^{n} (X_i - T)^+\right] \\
&\geq \frac{1}{2}T + \frac{1}{2}\mathbb{E}\left[(X^* - T)^+\right]
\end{aligned}$$

Proof of the Prophet Inequality

$\Pr[X^* \geq T] = \frac{1}{2}$. Let $\mathcal{E}_i$ be the event we "reach" the i-th random variable.

$$\begin{aligned}
\mathbb{E}[V] &= T \Pr[X^* \geq T] + \sum_{i=1}^{n} \Pr[\mathcal{E}_i] \cdot \mathbb{E}[(X_i - T)^+] \\
&= T \Pr[X^* \geq T] + \sum_{i=1}^{n} \Pr\left[\max_{1 \leq j \leq i-1} X_i < T\right] \cdot \mathbb{E}[(X_i - T)^+] \\
&\geq T \Pr[X^* \geq T] + \sum_{i=1}^{n} \Pr[X^* < T] \cdot \mathbb{E}[(X_i - T)^+] \\
&= \frac{1}{2} T + \frac{1}{2} \mathbb{E}\left[\sum_{i=1}^{n} (X_i - T)^+\right] \\
&\geq \frac{1}{2} T + \frac{1}{2} \mathbb{E}\left[(X^* - T)^+\right] \\
&\geq \frac{1}{2} \mathbb{E}[X^*]
\end{aligned}$$

$\frac{1}{2}$ is Tight

$\frac{1}{2}$ is Tight

- Consider two random variables X_1 and X_2, where $X_1 = 1$ deterministically, and $X_2 = \frac{1}{\varepsilon}$ w.p. ε and $X_2 = 0$ w.p. $1 - \varepsilon$, for some small $\varepsilon > 0$.

$\frac{1}{2}$ is Tight

- Consider two random variables X_1 and X_2, where $X_1 = 1$ deterministically, and $X_2 = \frac{1}{\varepsilon}$ w.p. ε and $X_2 = 0$ w.p. $1 - \varepsilon$, for some small $\varepsilon > 0$.
- Every algorithm will receive value 1 on expectation, regardless of which element it picks.

$\frac{1}{2}$ is Tight

- Consider two random variables X_1 and X_2, where $X_1 = 1$ deterministically, and $X_2 = \frac{1}{\varepsilon}$ w.p. ε and $X_2 = 0$ w.p. $1 - \varepsilon$, for some small $\varepsilon > 0$.
- Every algorithm will receive value 1 on expectation, regardless of which element it picks.
- The expected value of the prophet is

$$\mathbb{E}[X^*] = \frac{1}{\varepsilon} \cdot \varepsilon + 1 \cdot (1 - \varepsilon) = 2 - \varepsilon.$$

Overview

Secretary Problem (1/3)

Secretary Problem (1/3)

- Suppose now that we have no information on the distributions of the buyers. However, we know that they arrive in *random order*.

Secretary Problem (1/3)

- Suppose now that we have no information on the distributions of the buyers. However, we know that they arrive in *random order*.
- At every step i, we see a value v_i, and we have to immediately and irrevocably decide whether to select v_i or not.

Secretary Problem (1/3)

- Suppose now that we have no information on the distributions of the buyers. However, we know that they arrive in *random order*.
- At every step i, we see a value v_i, and we have to immediately and irrevocably decide whether to select v_i or not.
- Our objective is to maximize the probability with which we select $v^* = \max_{1 \leq i \leq n} v_i$. We assume distinct values for simplicity. What is the optimal strategy?

Secretary Problem (1/3)

- Suppose now that we have no information on the distributions of the buyers. However, we know that they arrive in *random order*.
- At every step i, we see a value v_i, and we have to immediately and irrevocably decide whether to select v_i or not.
- Our objective is to maximize the probability with which we select $v^* = \max_{1 \leq i \leq n} v_i$. We assume distinct values for simplicity. What is the optimal strategy?
- Simple problem, with an elegant and striking solution.

Secretary Problem (2/3)

Secretary Problem (2/3)

- When we see v_i, we should never select it if $v_i < \max_{1 \leq j \leq i-1} v_j$, because then v_i definitely won't be the best overall.

Secretary Problem (2/3)

- When we see v_i, we should never select it if $v_i < \max_{1 \leq j \leq i-1} v_j$, because then v_i definitely won't be the best overall.
- Also, the decision whether to accept v_i or not can only depend on $\{v_1, \ldots, v_i\}$. These imply that we should reject the first r values, for some r, and accept the first v_i where $i > r$ such that $v_i > \max_{1 \leq j \leq i-1} v_i$.

Secretary Problem (2/3)

- When we see v_i, we should never select it if $v_i < \max_{1 \leq j \leq i-1} v_j$, because then v_i definitely won't be the best overall.
- Also, the decision whether to accept v_i or not can only depend on $\{v_1, \ldots, v_i\}$. These imply that we should reject the first r values, for some r, and accept the first v_i where $i > r$ such that $v_i > \max_{1 \leq j \leq i-1} v_i$.
- Let $r = \frac{n}{2}$ and v_2^* denote the second-highest value. Then,
  1. with probability $1/2$, v^* is in the second half of the elements and,
  2. with probability $1/2$, v_2^* is in the first half of the elements.

Secretary Problem (2/3)

- When we see v_i, we should never select it if $v_i < \max_{1 \le j \le i-1} v_j$, because then v_i definitely won't be the best overall.
- Also, the decision whether to accept v_i or not can only depend on $\{v_1, \ldots, v_i\}$. These imply that we should reject the first r values, for some r, and accept the first v_i where $i > r$ such that $v_i > \max_{1 \le j \le i-1} v_i$.
- Let $r = \frac{n}{2}$ and v_2^* denote the second-highest value. Then,
  1. with probability $1/2$, v^* is in the second half of the elements and,
  2. with probability $1/2$, v_2^* is in the first half of the elements.
- If that happens, we will select v^*. Therefore, for $r = \frac{n}{2}$,

$$\Pr[v_{\text{sel}} = v^*] \ge \frac{1}{4}.$$

Secretary Problem (3/3)

Secretary Problem (3/3)

- In fact, the optimal policy is to set $r \approx \frac{n}{e}$. Then, one can show that $\Pr[v_{\text{sel}} = v^*] \geq \frac{1}{e}$, and this bound is tight [Lin61; Dyn63].

Secretary Problem (3/3)

- In fact, the optimal policy is to set $r \approx \frac{n}{e}$. Then, one can show that $\Pr[v_{\text{sel}} = v^*] \geq \frac{1}{e}$, and this bound is tight [Lin61; Dyn63].
- What about our grocer? In this setting, the grocer does not want money, but gives the orange away for free.

Secretary Problem (3/3)

- In fact, the optimal policy is to set $r \approx \frac{n}{e}$. Then, one can show that $\Pr[v_{\text{sel}} = v^*] \geq \frac{1}{e}$, and this bound is tight [Lin61; Dyn63].
- What about our grocer? In this setting, the grocer does not want money, but gives the orange away for free.
- If we have probability p of giving the orange to the maximum-valued buyer, then this immediately gives a p-competitive algorithm that selects a value v_{sel} such that

$$\mathbb{E}[v_{\text{sel}}] \geq p \cdot v^*.$$

Secretary Problem (3/3)

- In fact, the optimal policy is to set $r \approx \frac{n}{e}$. Then, one can show that $\Pr[v_{\mathrm{sel}} = v^*] \geq \frac{1}{e}$, and this bound is tight [Lin61; Dyn63].
- What about our grocer? In this setting, the grocer does not want money, but gives the orange away for free.
- If we have probability p of giving the orange to the maximum-valued buyer, then this immediately gives a p-competitive algorithm that selects a value v_{sel} such that

$$\mathbb{E}[v_{\mathrm{sel}}] \geq p \cdot v^*.$$

- The bound of $\frac{1}{e}$ is tight in this case, although the tightness proof does not follow as easily.

Overview

1. Introduction
   - The Prophet Inequality Problem
   - The Secretary Problem
2. Generalizations and Constraints
   - Selecting Multiple Values
   - Online Contention Resolution Schemes
3. Variations and Open Problems

k-Prophet (Uniform Matroid)

k-Prophet (Uniform Matroid)

- Can we derive simple algorithms for more complex settings? A natural generalization is the setting where we can accept up to k values, for a given k.

k-Prophet (Uniform Matroid)

- Can we derive simple algorithms for more complex settings? A natural generalization is the setting where we can accept up to k values, for a given k.
- For the prophet inequality problem, we compare against $\mathbb{E}\left[\max_{S:|S|\leq k}\sum_{i\in S}X_i\right]$. For this setting, we differentiate between "fixed-threshold" and "adaptive-threshold" algorithms.

k-Prophet (Uniform Matroid)

- Can we derive simple algorithms for more complex settings? A natural generalization is the setting where we can accept up to k values, for a given k.
- For the prophet inequality problem, we compare against $\mathbb{E}\left[\max_{S:|S|\leq k} \sum_{i\in S} X_i\right]$. For this setting, we differentiate between "fixed-threshold" and "adaptive-threshold" algorithms.

$\mathcal{A}$: "Adaptive-Threshold" Algorithm

For every $i \in [n]$, at step i, set a threshold T_i, based on $D_1, \ldots, D_n$ and $X_1, \ldots, X_{i-1}$, and accept every $X_i \geq T_i$ until we have selected k values.

Fixed-Threshold Algorithm for k-Prophet

Fixed-Threshold Algorithm for k-Prophet

- One can give a simple fixed-threshold algorithm for this setting, which achieves a $1 - O\left(\sqrt{\frac{\log k}{k}}\right)$-competitive ratio.

Fixed-Threshold Algorithm for k-Prophet

- One can give a simple fixed-threshold algorithm for this setting, which achieves a $1 - O\left(\sqrt{\frac{\log k}{k}}\right)$-competitive ratio.
- **Idea:** Select a threshold T such that the expected number of values $\geq T$ are $k - \delta$ for some δ.

Fixed-Threshold Algorithm for k-Prophet

- One can give a simple fixed-threshold algorithm for this setting, which achieves a $1 - O\left(\sqrt{\frac{\log k}{k}}\right)$-competitive ratio.
- **Idea:** Select a threshold T such that the expected number of values $\geq T$ are $k - \delta$ for some δ.
- Since the realizations of the X_i's are independent, for an appropriately chosen δ, one can show that the number of realizations that are at least T are between $k - 2\delta$ and k, with high probability (Hoeffding bound).

Fixed-Threshold Algorithm for k-Prophet

- One can give a simple fixed-threshold algorithm for this setting, which achieves a $1 - O\left(\sqrt{\frac{\log k}{k}}\right)$-competitive ratio.
- **Idea:** Select a threshold T such that the expected number of values $\geq T$ are $k - \delta$ for some δ.
- Since the realizations of the X_i's are independent, for an appropriately chosen δ, one can show that the number of realizations that are at least T are between $k - 2\delta$ and k, with high probability (Hoeffding bound).
- For fixed realizations, let $S_T = \{i \in [n] \mid X_i \geq T\}$. Then

$$\sum_{i \in S_T} X_i = \sum_{i \in S_T} T + (X_i - T) = T \cdot |S_T| + \sum_{i \in S_T} (X_i - T).$$

Fixed-Threshold Algorithm for k-Prophet

- One can give a simple fixed-threshold algorithm for this setting, which achieves a $1 - O\left(\sqrt{\frac{\log k}{k}}\right)$-competitive ratio.
- **Idea:** Select a threshold T such that the expected number of values $\geq T$ are $k - \delta$ for some δ.
- Since the realizations of the X_i's are independent, for an appropriately chosen δ, one can show that the number of realizations that are at least T are between $k - 2\delta$ and k, with high probability (Hoeffding bound).
- For fixed realizations, let $S_T = \{i \in [n] \mid X_i \geq T\}$. Then

$$\sum_{i \in S_T} X_i = \sum_{i \in S_T} T + (X_i - T) = T \cdot |S_T| + \sum_{i \in S_T} (X_i - T).$$

- For $\delta = \sqrt{2k \log k}$, we get

$$\sum_{i \in S_T} X_i \geq \left(1 - \sqrt{\frac{8 \log k}{k}}\right) OPT.$$

Adaptive-Threshold Algorithms

Adaptive-Threshold Algorithms

- One can get a better competitive ratio via an adaptive-threshold algorithm. The best known bound is $1 - \frac{1}{\sqrt{k+3}}$ [Ala14], and is asymptotically tight.

Adaptive-Threshold Algorithms

- One can get a better competitive ratio via an adaptive-threshold algorithm. The best known bound is $1 - \frac{1}{\sqrt{k+3}}$ [Ala14], and is asymptotically tight.
- For the secretary problem, [Kle05] showed that we can get a $1 - O\left(\frac{1}{\sqrt{k}}\right)$ competitive ratio, and this is also asymptotically tight.

Adaptive-Threshold Algorithms

- One can get a better competitive ratio via an adaptive-threshold algorithm. The best known bound is $1 - \frac{1}{\sqrt{k+3}}$ [Ala14], and is asymptotically tight.
- For the secretary problem, [Kle05] showed that we can get a $1 - O\left(\frac{1}{\sqrt{k}}\right)$ competitive ratio, and this is also asymptotically tight.
- The upper bound is obtained using a recursive algorithm. It first partitions the sequence into an initial segment and final segment of approximately equal length.

Adaptive-Threshold Algorithms

- One can get a better competitive ratio via an adaptive-threshold algorithm. The best known bound is $1 - \frac{1}{\sqrt{k+3}}$ [Ala14], and is asymptotically tight.
- For the secretary problem, [Kle05] showed that we can get a $1 - O\left(\frac{1}{\sqrt{k}}\right)$ competitive ratio, and this is also asymptotically tight.
- The upper bound is obtained using a recursive algorithm. It first partitions the sequence into an initial segment and final segment of approximately equal length.
- Then, it recursively chooses at most $\frac{k}{2}$ elements from the initial segment, and sets a threshold value equal to the $\frac{k}{2}$-th largest element of the initial segment.

Adaptive-Threshold Algorithms

- One can get a better competitive ratio via an adaptive-threshold algorithm. The best known bound is $1 - \frac{1}{\sqrt{k+3}}$ [Ala14], and is asymptotically tight.
- For the secretary problem, [Kle05] showed that we can get a $1 - O\left(\frac{1}{\sqrt{k}}\right)$ competitive ratio, and this is also asymptotically tight.
- The upper bound is obtained using a recursive algorithm. It first partitions the sequence into an initial segment and final segment of approximately equal length.
- Then, it recursively chooses at most $\frac{k}{2}$ elements from the initial segment, and sets a threshold value equal to the $\frac{k}{2}$-th largest element of the initial segment.
- Finally, it chooses all elements of the final segment that meet this threshold until exhausting its k allotted choices.

Overview

Matroid Constraint

Matroid Constraint

- We can generalize our problems even further by requiring the selected r.v.'s to be independent with respect to a constraint family $\mathcal{F} = ([n], \mathcal{I})$. Here, we compare against $\mathbb{E}\left[\max_{S \in \mathcal{I}} \sum_{i \in S} X_i\right]$ in the prophet inequality setting.

Matroid Constraint

- We can generalize our problems even further by requiring the selected r.v.'s to be independent with respect to a constraint family $\mathcal{F} = ([n], \mathcal{I})$. Here, we compare against $\mathbb{E}\left[\max_{S \in \mathcal{I}} \sum_{i \in S} X_i\right]$ in the prophet inequality setting.
- A natural case to consider is when $\mathcal{F}$ is a matroid on $[n]$. This is the first setting for which our problems differ significantly (as of yet).

Matroid Constraint

- We can generalize our problems even further by requiring the selected r.v.'s to be independent with respect to a constraint family $\mathcal{F} = ([n], \mathcal{I})$. Here, we compare against $\mathbb{E}\left[\max_{S \in \mathcal{I}} \sum_{i \in S} X_i\right]$ in the prophet inequality setting.
- A natural case to consider is when $\mathcal{F}$ is a matroid on $[n]$. This is the first setting for which our problems differ significantly (as of yet).
- [KW12] showed that there exists an (adaptive-threshold) algorithm for the *Matroid Prophet Inequality Problem* which matches the $\frac{1}{2}$-competitive ratio of the single-item case!

Matroid Constraint

- We can generalize our problems even further by requiring the selected r.v.'s to be independent with respect to a constraint family $\mathcal{F} = ([n], \mathcal{I})$. Here, we compare against $\mathbb{E}\left[\max_{S \in \mathcal{I}} \sum_{i \in S} X_i\right]$ in the prophet inequality setting.
- A natural case to consider is when $\mathcal{F}$ is a matroid on $[n]$. This is the first setting for which our problems differ significantly (as of yet).
- [KW12] showed that there exists an (adaptive-threshold) algorithm for the *Matroid Prophet Inequality Problem* which matches the $\frac{1}{2}$-competitive ratio of the single-item case!
- In contrast, no constant-competitive algorithm is known for the *Matroid Secretary Problem* as of yet. The best known algorithm gives a $O\left(\frac{1}{\log \log r}\right)$-competitive ratio, where r is the rank of the matroid [Lac14; FSZ18].

General Constraints

General Constraints

- How to generalize to different types or combinations of constraints?

General Constraints

- How to generalize to different types or combinations of constraints?
- **Idea:** Find a function g that is an upper bound on OPT. Model the problem as an LP

$$\begin{aligned} \max. \quad & g(\boldsymbol{y}) \\ \text{s.t.} \quad & \boldsymbol{y} \in \mathcal{P}_{\mathcal{I}} \quad (LP) \\ & \boldsymbol{y} \geq 0 \end{aligned}$$

  where $\mathcal{P}_{\mathcal{I}}$ is a convex relaxation of $\mathcal{I}$.

General Constraints

- How to generalize to different types or combinations of constraints?
- **Idea:** Find a function g that is an upper bound on OPT. Model the problem as an LP

$$\begin{aligned} \text{max.} \quad & g(\boldsymbol{y}) \\ \text{s.t.} \quad & \boldsymbol{y} \in \mathcal{P}_{\mathcal{I}} \quad (LP) \\ & \boldsymbol{y} \geq 0 \end{aligned}$$

  where $\mathcal{P}_{\mathcal{I}}$ is a convex relaxation of $\mathcal{I}$.
- Solving (LP) yields a fractional point $\boldsymbol{x}$, which we want to round, subject to our constraints $\mathcal{F}$, but also in an online fashion.

Contention Resolution Schemes

Contention Resolution Schemes

- How can we round subject to constraints such that we always return a feasible set?

Contention Resolution Schemes

- How can we round subject to constraints such that we always return a feasible set?

Contention Resolution Scheme (informally) [CVZ11]

A (b, c)-balanced *Contention Resolution Scheme (CRS)* is a procedure which receives a point $\boldsymbol{x} \in b \cdot \mathcal{P}_{\mathcal{I}}$ as input and returns a set $S \in \mathcal{I}$ which contains every $i \in [n]$ with probability at least $c \cdot x_i$.

Contention Resolution Schemes

- How can we round subject to constraints such that we always return a feasible set?

Contention Resolution Scheme (informally) [CVZ11]

A (b, c)-balanced *Contention Resolution Scheme (CRS)* is a procedure which receives a point $\boldsymbol{x} \in b \cdot \mathcal{P}_{\mathcal{I}}$ as input and returns a set $S \in \mathcal{I}$ which contains every $i \in [n]$ with probability at least $c \cdot x_i$.

- This guarantee yields a bc-approximation w.r.t. OPT_{LP}, and thus also OPT.

Contention Resolution Schemes

- How can we round subject to constraints such that we always return a feasible set?

Contention Resolution Scheme (informally) [CVZ11]

A (b, c)-balanced *Contention Resolution Scheme (CRS)* is a procedure which receives a point $\boldsymbol{x} \in b \cdot \mathcal{P}_{\mathcal{I}}$ as input and returns a set $S \in \mathcal{I}$ which contains every $i \in [n]$ with probability at least $c \cdot x_i$.

- This guarantee yields a bc-approximation w.r.t. OPT_{LP}, and thus also OPT.
- While CRSs are great, they are of no help for our problems, since we want to round the x_i's in an online fashion.

Online Contention Resolution Schemes

Online Contention Resolution Schemes

- Surprisingly, this can be done with little loss in the approximation guarantees via Online Contention Resolution Schemes (OCRSs) [FSZ16]!.

Online Contention Resolution Schemes

- Surprisingly, this can be done with little loss in the approximation guarantees via Online Contention Resolution Schemes (OCRSs) [FSZ16]!.
- Essentially, an OCRSs that gives an α-approximation w.r.t. OPT_{LP} for a constraint $\mathcal{F}$, yields an equivalent α-competitive algorithm for the prophet inequality problem w.r.t. $\mathcal{F}$.

Online Contention Resolution Schemes

Online Contention Resolution Schemes

- OCRSs (and CRSs) exist for matroids, matchings, knapsacks, etc.

Online Contention Resolution Schemes

- OCRSs (and CRSs) exist for matroids, matchings, knapsacks, etc.
- They are nice because we can combine them to obtain OCRSs for more complicated constraints.

Online Contention Resolution Schemes

- OCRSs (and CRSs) exist for matroids, matchings, knapsacks, etc.
- They are nice because we can combine them to obtain OCRSs for more complicated constraints.
- Recently, prophet inequalities have been used to give optimal OCRSs for simple settings, implying the connection between the two is deeper.

Overview

Prophet Secretary and I.I.D. Setting

Prophet Secretary and I.I.D. Setting

- What if we could get the best of both worlds?

Prophet Secretary and I.I.D. Setting

- What if we could get the best of both worlds?
- In the *Prophet Secretary Problem*, we are given n non-negative random variables $X_1, \ldots, X_n$ and their distributions $D_1, \ldots, D_n$, and we observe a realization from each X_i in *random order*.

Prophet Secretary and I.I.D. Setting

- What if we could get the best of both worlds?
- In the *Prophet Secretary Problem*, we are given n non-negative random variables $X_1, \ldots, X_n$ and their distributions $D_1, \ldots, D_n$, and we observe a realization from each X_i in *random order*.
- Can we do better than $\frac{1}{2}$? In fact, yes! [Esf+15] showed that there exists a $1 - \frac{1}{e}$-competitive algorithm, and recently, a $1 - \frac{1}{e} + d$-competitive algorithm was discovered for some small constant $d > 0$ [ACK17; CSZ18].

Prophet Secretary and I.I.D. Setting

- What if we could get the best of both worlds?
- In the *Prophet Secretary Problem*, we are given n non-negative random variables $X_1, \ldots, X_n$ and their distributions $D_1, \ldots, D_n$, and we observe a realization from each X_i in *random order*.
- Can we do better than $\frac{1}{2}$? In fact, yes! [Esf+15] showed that there exists a $1 - \frac{1}{e}$-competitive algorithm, and recently, a $1 - \frac{1}{e} + d$-competitive algorithm was discovered for some small constant $d > 0$ [ACK17; CSZ18].
- What if we knew that all random variables in the prophet inequality setting were i.i.d.? Clearly the optimal bound in this case is not worse than the prophet secretary problem.

Prophet Secretary and I.I.D. Setting

- What if we could get the best of both worlds?
- In the *Prophet Secretary Problem*, we are given n non-negative random variables $X_1, \dots, X_n$ and their distributions $D_1, \dots, D_n$, and we observe a realization from each X_i in *random order*.
- Can we do better than $\frac{1}{2}$? In fact, yes! [Esf+15] showed that there exists a $1 - \frac{1}{e}$-competitive algorithm, and recently, a $1 - \frac{1}{e} + d$-competitive algorithm was discovered for some small constant $d > 0$ [ACK17; CSZ18].
- What if we knew that all random variables in the prophet inequality setting were i.i.d.? Clearly the optimal bound in this case is not worse than the prophet secretary problem.
- [Cor+17] showed that the optimal ratio is ≈ 0.7451, and is actually tight.

General Objectives

General Objectives

- Other objective functions have been considered as well. In these settings, the objective is to select a set $S \in \mathcal{I}$ (for some constraint family $\mathcal{F}$) to maximize $\mathbb{E}[f(S)]$, and we compare against $\mathbb{E}[\max_{T \in \mathcal{I}} f(T)]$.

General Objectives

- Other objective functions have been considered as well. In these settings, the objective is to select a set $S \in \mathcal{I}$ (for some constraint family $\mathcal{F}$) to maximize $\mathbb{E}[f(S)]$, and we compare against $\mathbb{E}[\max_{T \in \mathcal{I}} f(T)]$.
- When f is a submodular function, we can use OCRSs and obtain constant-competitive algorithms for the *Submodular Prophet Inequality Problem* [RS16]. More general functions (e.g. monotone subadditive) have been studied as well [Rub16; RS16].

General Objectives

- Other objective functions have been considered as well. In these settings, the objective is to select a set $S \in \mathcal{I}$ (for some constraint family $\mathcal{F}$) to maximize $\mathbb{E}[f(S)]$, and we compare against $\mathbb{E}[\max_{T \in \mathcal{I}} f(T)]$.
- When f is a submodular function, we can use OCRSs and obtain constant-competitive algorithms for the *Submodular Prophet Inequality Problem* [RS16]. More general functions (e.g. monotone subadditive) have been studied as well [Rub16; RS16].
- Furthermore, when f is a submodular function, [FZ18] showed that any α-competitive algorithm for the Matroid Secretary Problem yields a $O(\alpha)$-competitive algorithm for the *Submodular Matroid Secretary Problem*.

Future Directions

Future Directions

1. Constant-competitive algorithm for the Matroid Secretary Problem.

Future Directions

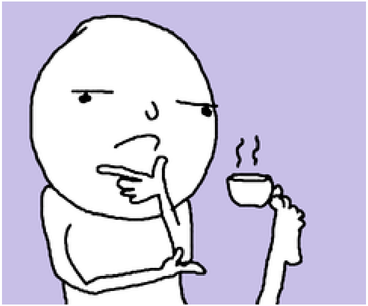
1. Constant-competitive algorithm for the Matroid Secretary Problem.
2. Is there a deeper connection between OCRSs and prophet inequalities?

Future Directions

1. Constant-competitive algorithm for the Matroid Secretary Problem.
2. Is there a deeper connection between OCRSs and prophet inequalities?
3. What is the best constant for the Prophet Secretary Problem?

and more...

QUESTIONS ?



Fixed-Threshold Algorithm for k-Prophet (Proof 1/2)

Fixed-Threshold Algorithm for k-Prophet (Proof 1/2)

- One can give a simple fixed-threshold algorithm for this setting, which achieves a $1 - O\left(\sqrt{\frac{\log k}{k}}\right)$-competitive ratio.
- **Idea:** Select a threshold T such that the expected number of values $\geq T$ are $k - \delta$ for some δ.
- Since the realizations of the X_i's are independent, for an appropriately chosen δ, one can show that the number of realizations that are at least T are between $k - 2\delta$ and k, with high probability (Hoeffding bound).
- For fixed realizations, let $S_T = \{i \in [n] \mid X_i \geq T\}$. Then

$$\sum_{i \in S_T} X_i = \sum_{i \in S_T} T + (X_i - T) = T \cdot |S_T| + \sum_{i \in S_T} (X_i - T).$$

Fixed-Threshold Algorithm for k-Prophet (Proof 1/2)

- One can give a simple fixed-threshold algorithm for this setting, which achieves a $1 - O\left(\sqrt{\frac{\log k}{k}}\right)$-competitive ratio.
- **Idea:** Select a threshold T such that the expected number of values $\geq T$ are $k - \delta$ for some δ.
- Since the realizations of the X_i's are independent, for an appropriately chosen δ, one can show that the number of realizations that are at least T are between $k - 2\delta$ and k, with high probability (Hoeffding bound).
- For fixed realizations, let $S_T = \{i \in [n] \mid X_i \geq T\}$. Then

$$\sum_{i \in S_T} X_i = \sum_{i \in S_T} T + (X_i - T) = T \cdot |S_T| + \sum_{i \in S_T} (X_i - T).$$

- Since $|S_T| \geq k - 2\delta$, our revenue is at least $(k - 2\delta)T$.

Fixed-Threshold Algorithm for k-Prophet (Proof 2/2)

Fixed-Threshold Algorithm for k-Prophet (Proof 2/2)

- Let S^* be the optimal set selected by the prophet. Then

$$OPT = \sum_{i \in S^*} X_i \leq \sum_{i \in S^*} T + (X_i - T) \leq kT + \sum_{i=1}^{n} (X_i - T),$$

Fixed-Threshold Algorithm for k-Prophet (Proof 2/2)

- Let S^* be the optimal set selected by the prophet. Then

$$OPT = \sum_{i \in S^*} X_i \leq \sum_{i \in S^*} T + (X_i - T) \leq kT + \sum_{i=1}^{n} (X_i - T),$$

- Since $|S_T| \leq k$, we accepted every value that was at least T. Thus

$$\sum_{i \in S_T} (X_i - T) = \sum_{i=1}^{n} (X_i - T) \geq OPT - kT \geq \frac{k - 2\delta}{k} (OPT - kT)$$
$$= \left(1 - \frac{2\delta}{k}\right) OPT - (k - 2\delta) T.$$

Fixed-Threshold Algorithm for k-Prophet (Proof 2/2)

- Let S^* be the optimal set selected by the prophet. Then

$$OPT = \sum_{i \in S^*} X_i \le \sum_{i \in S^*} T + (X_i - T) \le kT + \sum_{i=1}^{n} (X_i - T),$$

- Since $|S_T| \le k$, we accepted every value that was at least T. Thus

$$\sum_{i \in S_T} (X_i - T) = \sum_{i=1}^{n} (X_i - T) \ge OPT - kT \ge \frac{k - 2\delta}{k} (OPT - kT)$$
$$= \left(1 - \frac{2\delta}{k}\right) OPT - (k - 2\delta) T.$$

- For $\delta = \sqrt{2k \log k}$, we get

$$\sum_{i \in S_T} X_i \ge \left(1 - \frac{2\delta}{k}\right) OPT = \left(1 - \sqrt{\frac{8 \log k}{k}}\right) OPT.$$

References I

Yossi Azar, Ashish Chiplunkar, and Haim Kaplan. "Prophet Secretary: Surpassing the 1-1/e Barrier". In: *CoRR* abs/1711.01834 (2017). arXiv: 1711.01834. URL: http://arxiv.org/abs/1711.01834.

Saeed Alaei. "Bayesian Combinatorial Auctions: Expanding Single Buyer Mechanisms to Many Buyers". In: *SIAM Journal on Computing* 43.2 (2014), pp. 930–972. DOI: 10.1137/120878422. eprint: https://doi.org/10.1137/120878422. URL: https://doi.org/10.1137/120878422.

José Correa et al. "Posted Price Mechanisms for a Random Stream of Customers". In: *Proceedings of the 2017 ACM Conference on Economics and Computation*. EC '17. Cambridge, Massachusetts, USA: ACM, 2017, pp. 169–186. ISBN: 978-1-4503-4527-9. DOI: 10.1145/3033274.3085137. URL: http://doi.acm.org/10.1145/3033274.3085137.

References II

José R. Correa, Raimundo Saona, and Bruno Ziliotto. "Prophet Secretary Through Blind Strategies". In: *CoRR* abs/1807.07483 (2018). arXiv: 1807.07483. URL: http://arxiv.org/abs/1807.07483.

Chandra Chekuri, Jan Vondrák, and Rico Zenklusen. "Submodular Function Maximization via the Multilinear Relaxation and Contention Resolution Schemes". In: *Proceedings of the Forty-third Annual ACM Symposium on Theory of Computing*. STOC '11. San Jose, California, USA: ACM, 2011, pp. 783–792. ISBN: 978-1-4503-0691-1. DOI: 10.1145/1993636.1993740. URL: http://doi.acm.org/10.1145/1993636.1993740.

E. B. Dynkin. "The optimum choice of the instant for stopping a Markov process". In: *Soviet Math. Dokl* 4 (1963).

References III

Hossein Esfandiari et al. "Prophet Secretary". In: *CoRR* abs/1507.01155 (2015). arXiv: 1507.01155. URL: http://arxiv.org/abs/1507.01155.

Moran Feldman, Ola Svensson, and Rico Zenklusen. "Online Contention Resolution Schemes". In: *Proceedings of the Twenty-seventh Annual ACM-SIAM Symposium on Discrete Algorithms*. SODA '16. Arlington, Virginia: Society for Industrial and Applied Mathematics, 2016, pp. 1014–1033. ISBN: 978-1-611974-33-1. URL: http://dl.acm.org/citation.cfm?id=2884435.2884507.

References IV

Moran Feldman, Ola Svensson, and Rico Zenklusen. "A Simple O(log log(rank))-Competitive Algorithm for the Matroid Secretary Problem". In: *Mathematics of Operations Research* 43.2 (2018), pp. 638–650. DOI: 10.1287/moor.2017.0876. eprint: https://doi.org/10.1287/moor.2017.0876. URL: https://doi.org/10.1287/moor.2017.0876.

Moran Feldman and Rico Zenklusen. "The Submodular Secretary Problem Goes Linear". In: *SIAM Journal on Computing* 47.2 (2018), pp. 330–366. DOI: 10.1137/16M1105220. eprint: https://doi.org/10.1137/16M1105220. URL: https://doi.org/10.1137/16M1105220.

Robert Kleinberg. "A Multiple-Choice Secretary Algorithm with Applications to Online Auctions". In: *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms*. SODA '05. Vancouver, British Columbia: Society for Industrial and Applied Mathematics, USA, 2005, pp. 630–631. ISBN: 0898715857.

Ulrich Krengel and Louis Sucheston. "Semiamarts and finite values". In: *Bull. Amer. Math. Soc.* 83.4 (July 1977), pp. 745–747. URL: https://projecteuclid.org:443/euclid.bams/1183538915.

Robert Kleinberg and S. Matthew Weinberg. "Matroid Prophet Inequalities". In: *CoRR* abs/1201.4764 (2012). arXiv: 1201.4764. URL: http://arxiv.org/abs/1201.4764.

References VI

O. Lachish. "O(log log Rank) Competitive Ratio for the Matroid Secretary Problem". In: *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*. 2014, pp. 326–335.

D. V. Lindley. "Dynamic Programming and Decision Theory". In: *Journal of the Royal Statistical Society. Series C (Applied Statistics)* 10.1 (1961), pp. 39–51. ISSN: 00359254, 14679876. URL: http://www.jstor.org/stable/2985407.

Aviad Rubinstein and Sahil Singla. "Combinatorial Prophet Inequalities". In: *CoRR* abs/1611.00665 (2016). arXiv: 1611.00665. URL: http://arxiv.org/abs/1611.00665.

Aviad Rubinstein. "Beyond Matroids: Secretary Problem and Prophet Inequality with General Constraints". In: *Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing*. New York, NY, USA: Association for Computing Machinery, 2016, pp. 324–332. ISBN: 9781450341325. URL: https://doi.org/10.1145/2897518.2897540.

Ester Samuel-Cahn. "Comparison of Threshold Stop Rules and Maximum for Independent Nonnegative Random Variables". In: *The Annals of Probability* 12.4 (1984), pp. 1213–1216. ISSN: 00911798. URL: http://www.jstor.org/stable/2243359.