Learning Online Algorithms with Distributional Advice

Ilias Diakonikolas [*1] **Vasilis Kontonis** [*1] **Christos Tzamos** [*1] **Ali Vakilian** [*2] **Nikos Zarifis** [*1]

Abstract

We study the problem of designing online algorithms given advice about the input. While prior work had focused on deterministic advice, we only assume distributional access to the instances of interest, and the goal is to learn a competitive algorithm given access to i.i.d. samples. We aim to be competitive against an adversary with prior knowledge of the distribution, while also performing well against worst-case inputs.

We focus on the classical online problems of ski-rental and prophet-inequalities, and provide sample complexity bounds for the underlying learning tasks. First, we point out that for general distributions it is information-theoretically impossible to beat the worst-case competitive-ratio with any finite sample size. As our main contribution, we establish strong positive results for well-behaved distributions. Specifically, for the broad class of log-concave distributions, we show that $\mathrm{poly}(1/\epsilon)$ samples suffice to obtain $(1+\epsilon)$-competitive ratio. Finally, we show that this sample upper bound is close to best possible, even for very simple classes of distributions.

1. Introduction

Uncertainty in the input is a central challenge in the area of algorithm design. A number of both classical and recent paradigms have been introduced to capture this issue. *Online computation* is a prototypical domain where dealing with *uncertainty of future data* naturally arises. To get around the inherent uncertainty of the online setting, many different models have been introduced: online algorithms with lookahead (Grove, 1995), with statistical adversary (Raghavan, 1992), or with diffused adversary (Koutsoupias & Papadimitriou, 2000). A more recent line of work studies online algorithms with *predictions* or *advice* (also known as learning-based/data-driven online algorithms) (Renault & Rosén, 2015; Angelopoulos et al., 2015; Lykouris & Vassilvtiskii, 2018; Purohit et al., 2018; Gollapudi & Panigrahi, 2019; Angelopoulos et al., 2020; Dütting et al., 2020; Lattanzi et al., 2020; Anand et al., 2020; Bamas et al., 2020).

In this work, we introduce a new model of *learning-based algorithms*. Our goal is to design algorithms that can leverage the *learnable* structure of the input. Specifically, we assume sample access to the underlying distribution over inputs. Given i.i.d. samples from this distribution, we want to design methods that are competitive against the optimal performance of any algorithm with *exact* knowledge of the distribution over inputs; and at the same time are robust to model misspecification or adversarial predictions. In other words, when the predictions are *almost perfect*, we expect the algorithm to solve the problem almost optimally; and if the predictions are misleading, the algorithm is required to attain guarantees comparable to the best possible worst-case bounds. This work falls in the active domain of *beyond worst-case analysis* (see the recent book (Roughgarden, 2020) and the related chapter on algorithms with predictions (Mitzenmacher & Vassilvitskii, 2020)).

While most of the existing results in learning-based online algorithms only consider *deterministic* predictions, in this work we aim to design such algorithms when predictions are random and drawn from an underlying distribution. This setting can accurately model the type of predictions that arise in real applications. In many scenarios, we have access to past data, i.e., distributional advice, that can be used to design an algorithm that performs well on unseen data.

In this paper, we focus on two basic online problems with distributional information. In particular, we investigate the trade-off between the *efficiency* and the *sample complexity* of the *ski-rental* and the *prophet inequality* problems in an online model when the input distribution is assumed to belong to a known class of distributions. Unlike the classical online setting, we consider the *diffused adversary model* proposed by Koutsoupias and Papadimitriou (2000), which assumes that the given input distribution belongs to

[*]Equal contribution [1]Department of Computer Sciences, University of Wisconsin, Madison, Wisconsin, USA [2]Toyota Technological Institute at Chicago (TTIC), Chicago, Illinois, USA. This work was done when the author was at University of Wisconsin-Madison. Correspondence to: Ali Vakilian <vakilian@ttic.edu>.

Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021.

a *known* class $\mathcal{C}$ of distributions. We note that Koutsoupias and Papadimitriou study the generalized competitive ratio of online problems such as paging and k-server in the defused adversary model only assuming knowledge of the class of distributions. In contrast, we also assume *sample access* to the input distribution.

1.1. Problem Statement

Given an online minimization problem f, a distribution D on inputs for f, and an algorithm $\mathcal{A}$, we denote $\mathrm{cost}(\mathcal{A}; t)$ the cost of $\mathcal{A}$ for f on input t and $\mathrm{cost}(\mathcal{A}; D) := \mathbf{E}_{t\sim D}[\mathrm{cost}(\mathcal{A}; t)]$, i.e., the average cost of $\mathcal{A}$ when the input is drawn from D. We remark that when $\mathcal{A}$ is a randomized algorithm, the expected cost is computed with respect to the randomness of both $\mathcal{A}$ and D. We also define the optimal cost under the distribution D to be the minimum achievable cost for any algorithm, i.e., $\mathrm{OPT}_D = \min_{\mathcal{A}} \mathrm{cost}(\mathcal{A}; D)$.

Observe that apart from the computational hardness of optimizing over possible algorithms, computing OPT_D requires *exact knowledge of the input distribution* D. In our setting, we only assume sample access to D.

For maximization problems, we define OPT_D similarly where instead of minimizing $\mathrm{cost}(\cdot)$ the goal is to maximize $\mathrm{gain}(\cdot)$, i.e., $\mathrm{OPT}_D = \max_{\mathcal{A}} \mathrm{gain}(\mathcal{A}; D)$.

Robustness and Consistency. Two measures of interest in the domain of algorithms with predictions are *consistency* and *robustness*. At a high-level, the consistency requirement implies that, if the samples that we observe are accurate, then the generalized competitive ratio should approach one. The robustness requirement implies that in any case (i.e., no matter how inaccurate the predictions are), the (generalized) competitive ratio should not be much worse compared to the best pure online algorithm.

Fix a family of distributions $\mathcal{C}$ and some distribution D not necessarily in $\mathcal{C}$. Let $\mathcal{X}$ be a finite set of i.i.d. samples from D and let $\mathcal{A}_{\mathcal{X}}$ be the algorithm that we learn given the set of samples $\mathcal{X}$. We say that the algorithm $\mathcal{A}_{\mathcal{X}}$ is α-*consistent* and β-*robust* under the class $\mathcal{C}$ if the following hold.

- If $D \in \mathcal{C}$, then $\mathrm{cost}(\mathcal{A}_{\mathcal{X}}; D)/\mathrm{OPT}_D \le \alpha$.
- Otherwise, $\mathrm{cost}(\mathcal{A}_{\mathcal{X}}; D)/\mathrm{OPT}_{\mathrm{ONL}} \le \beta$, where $\mathrm{OPT}_{\mathrm{ONL}}$ is the cost of an optimal classical online algorithm on D.

Similarly, we can define the notion of robust and consistent algorithms for maximization problems by considering the inverse ratios, e.g., $\frac{\mathrm{OPT}_D}{\mathrm{gain}(\mathcal{A}_{\mathcal{X}}; D)} \le \alpha$.

In this paper, our goal is to answer the following question.

"Given finite samples from a distribution D, can we efficiently design consistent and robust algorithms?"

1.2. Our Results

Ski Rental. In the ski-rental problem, each day the player has to decide whether to rent skis for this day, which costs one unit, or buy skis for the rest of the season at a cost of b units. The goal is to minimize the total cost paid by the player. Ski-rental is a well-studied problem and admits an algorithm with competitive ratio equal to $\frac{e}{e-1}$, which is known to be tight (Karlin et al., 1994).

Before we present our result for the ski-rental setting, we note that it is not possible to improve over the existing competitive ratio bounds without distributional assumptions.

Observation 1.1. *Fix any $\alpha < e/(e-1)$. There is no algorithm that for any input distribution D, draws finitely many samples from D and returns an α-consistent strategy for the ski-rental problem.*

We show that we can bypass this negative result when the input distributions are assumed to have some "nice" structure. In previous literature for the ski-rental problem with predictions (Purohit et al., 2018; Gollapudi & Panigrahi, 2019; Wang et al., 2020), the authors assume that the advice is a guess for the exact number of the days that we are going to ski. A distributional generalization of this would be to assume that number of ski-days is distributed uniformly in some interval of the real line. More generally, a natural model for the number of ski-days is that it follows a *log-concave distribution*. Log-concave distributions are a general non-parametric class of distributions that have been used as a model extensively in statistics and machine learning (Bagnoli & Bergstrom, 2005). In particular, they include the uniform distribution on any interval, but also distributions with infinite support, including Gaussians and exponentials.

Definition 1.2 (Log-Concave Distributions)**.** *We say that a distribution is log-concave if its density can be written as $p(x) = e^{f(x)}$, for some concave function $f : \mathbb{R} \to \mathbb{R}$.*

As our first main contribution, we show that under any log-concave distribution we can design consistent and robust algorithms for the ski-rental problem.

Result 1. *For any $\lambda > 1$, there exists an algorithm that draws $\widetilde{O}(1/\varepsilon^2)$ samples, runs in sample near-linear time, and outputs a $(\lambda(1+\varepsilon))$-consistent and $(\frac{\lambda}{\lambda-1})$-robust strategy for ski-rental under log-concave distributions (see Theorem 2.9).*

To prove the above result, we first show that we can obtain ε-additive strategies, i.e., strategies with cost $\mathrm{OPT}_D + \varepsilon$, for general distributions, see Section 2.1. Next, we show that the optimal solution under log-concave distributions is structured: it either corresponds to buying the skis initially or renting them indefinitely. Then, by an application of our result on ε-additive strategies, we derive a $(1+\varepsilon)$-multiplicative strategy on log-concave distributions with tight sample complexity, see Section 2.2. Finally, by using an algorithm of Mahdian et al. (2012) in black-box fashion, we obtain our main result for ski-rental.

We also prove that the sample complexity of Result 1 is essentially optimal: in Section 2.4 we show that $\Omega(1/\varepsilon^2)$ are necessary in order to obtain $(1+\varepsilon)$-consistent algorithm, even when the underlying distributions are exponentials.

Prophet Inequality. In the prophet inequality problem, there are n distributions $D_1, \dots, D_n$. A gambler knows both the distributions D_i and their order. Each "day", the gambler observes a value $X_i \sim D_i$ and has to decide whether to accept and get X_i as reward or to reject and continue with X_{i+1}, losing X_i irrevocably. The gambler's goal is to maximize the gain. In the standard setting, the gambler is competing against the "prophet" who knows the exact realizations $X_1, \dots, X_n$ and therefore achieves gain equal to $\mathbf{E}[\max(X_1, \dots, X_n)]$. In this setting, it is known that the best possible competitive ratio is $1/2$ (Krengel & Sucheston, 1978) and it is achievable by simple stopping rules such as the median of $\max_i X_i$ (i.e., a value T s.t. $\Pr[\max_i X_i > T] = 1/2$) (Samuel-Cahn, 1984) or $\mathbf{E}[\max_i X_i]/2$ (Kleinberg & Weinberg, 2012). In our distributional setting, the gambler does not know exactly the distributions $D_1, \dots, D_n$, but only has sample access to them. Similar to the ski-rental setting, the gambler is competing against an adversary who knows the distributions $D_1, \dots, D_n$ (not their exact realizations $X_1, \dots, X_n$).

Similarly to Observation 1.1 for ski-rental, we cannot improve over the existing, worst case, competitive ratio of $1/2$ for prophet inequality without distributional assumptions (see the full version of the paper for the formal statement). For log-concave distributions, we show:

Result 2. *For any $\lambda > 1$, there exists an algorithm that draws $\widetilde{O}(n^3/\varepsilon^2)$ samples from the distributions $D_1, \dots, D_n$ and, in sample near-linear time, outputs a $((1+\varepsilon)\lambda)$-consistent and $(2\lambda/(\lambda-1))$-robust strategy for prophet inequality under log-concave distributions (see Theorem 3.3).*

We remark that $\Omega(n)$ samples are necessary to guarantee any non-trivial error guarantees in the prophet inequality setting: if we do not observe at least one sample from each D_i we may ignore one with very high average value. We did not attempt to optimize the sample complexity of our algorithm with respect to the number of distributions n. This is left as an interesting question for future work.

Stronger Sampling Models. We show that, assuming access to the stronger *conditional sampling* oracle as introduced in (Canonne et al., 2015; Chakraborty et al., 2016), we can achieve a $(1+\varepsilon)$-multiplicative guarantee on any distribution *without structural assumptions*. We work in the ski-rental setting. Given an unknown distribution D supported on $[0, +\infty)$, for any subset $S \subseteq [0, +\infty)$, the conditional sampling oracle returns a sample with probability proportional to the conditional distribution of D restricted to S. In this paper, we use a weaker version of this oracle where the target sets S are of form $[\alpha, +\infty)$. Due to space limitations, we include the proof of the following result in the final version of the paper.

Result 3. *For any input distribution D, $O(1/\varepsilon^4)$ conditional samples from D suffice to design a strategy $\mathcal{A}$ that achieves* $\mathrm{cost}(\mathcal{A}; D) \le (1+\varepsilon)\mathrm{OPT}_D$.

We remark that under some natural anti-concentration properties of the distribution, the conditional sampling access we require can be simulated via traditional samples and rejection sampling with a small overhead.

1.3. Related Work

Online Algorithms with Advice. Incorporating machine learning advice in the design of online algorithm has received a lot of attention in recent years (Mahdian et al., 2012; Angelopoulos et al., 2015; Esfandiari et al., 2018; Lykouris & Vassilvtiskii, 2018; Purohit et al., 2018; Gollapudi & Panigrahi, 2019; Kodialam, 2019; Indyk et al., 2020; Anand et al., 2020; Angelopoulos et al., 2020; Rohatgi, 2020; Lattanzi et al., 2020; Dütting et al., 2020; Lavastida et al., 2020; Antoniadis et al., 2020; Bamas et al., 2020). In particular, the ski-rental problem has served as one of the prominent standard test beds for most of developed techniques and proposed models in this area, e.g., (Purohit et al., 2018; Gollapudi & Panigrahi, 2019; Kodialam, 2019; Anand et al., 2020; Bamas et al., 2020).

Algorithms with predictions have been also studied extensively in other domains such as online learning (Alabi et al., 2019; Bhaskara et al., 2020), data structures (Kraska et al., 2018; Mitzenmacher, 2018; Dai & Shrivastava, 2019; Vaidya et al., 2021), streaming and sketching (Hsu et al., 2019; Indyk et al., 2019; Jiang et al., 2020; Cohen et al., 2020) and combinatorial problems (Gupta & Roughgarden, 2017; Dai et al., 2017; Balcan et al., 2017; 2018).

Comparison with (Anand et al., 2020). We provide a comparison of our framework to the recent result of Anand, Ge, and Panigrahi, in which the authors design a "learning to rent" framework. In their setting, the algorithm receives a feature vector x which can be considered as "prediction", and they assume that there exists an unknown underlying joint distribution between x and the actual number of ski-

Table 1. This table shows the sample complexity of our algorithms in different settings of ski-rental and prophet inequality. In ski-rental, b denotes the price of ski. In prophet inequality, for ε-additive approximation guarantee, we assume $D_1, \dots, D_n$ are supported on $[0, b]$.

	$(1+\varepsilon)$-Multiplicative		Consistency and Robustness	ε-Additive
	General	Log-Concave		
Ski-Rental	Inapprox.	$\widetilde{O}(\varepsilon^{-2})$	Theorem 2.9	$\widetilde{O}(b^2\varepsilon^{-2})$
Prophet Inequality	Inapprox.	$\widetilde{O}(n^3\varepsilon^{-2})$	Theorem 3.3	$\widetilde{O}(b^2 n^2\, \varepsilon^{-2})$

days y; $(x, y) \sim \mathcal{K}$. Then, their goal is to learn an algorithm $\theta(x)$, which decides the number of days to rent. The goal of Anand et al. is to output a learning rule θ, which can be thought as a mapping from the "distributional" prediction to a stopping time, and their objective is to minimize the expected value of the competitive ratios over prediction x. In contrast, in our work, we design a single strategy for the given unknown distribution and our measure of efficiency is the competitive ratio of the expected costs. While the objective of both papers is to design learning-based algorithms under distributional information, the settings and approaches are quite different.

2. Ski-Rental

In this section, our main goal is to prove Result 1. In Subsection 2.1 we show that we can obtain additive approximations to OPT_D under any distribution with $\widetilde{O}(1/\varepsilon^2)$ samples. In Subsection 2.2 we show that using the additive approximation result we can obtain multiplicative approximations assuming that the underlying distribution is log-concave. In Subsection 2.4 we prove that $\Omega(1/\varepsilon^2)$ samples are necessary in order to obtain $(1+\varepsilon)$-multiplicative estimates for log-concave distributions.

2.1. Sample Complexity of Additive Approximation

In this subsection, we study the sample complexity of ε-additive approximation algorithms for ski-rental. We remark that in this problem any deterministic algorithm/strategy $\mathcal{A}$ is equivalent to a threshold T where we decide to buy the skis. For example, if our strategy is to buy the skis initially we have $T = 0$ or, on other extreme, renting them indefinitely corresponds to $T = \infty$. Any randomized algorithm for this problem corresponds to a distribution over thresholds. The cost of an algorithm $\mathcal{A}$ corresponding to a distribution q over thresholds with respect to some distribution D is then defined as[1]

$$\mathrm{cost}(\mathcal{A}; D) = \mathrm{cost}(q; D) = \mathop{\mathbf{E}}_{T\sim q}\left[\mathop{\mathbf{E}}_{x\sim D}[\mathbf{1}\{x < T\}x + (b+T)\mathbf{1}\{x \ge T\}]\right],$$

where the first term corresponds to the case where the actual number of ski-days $x \sim D$ is smaller than T: we just pay amount x for renting the skis for those days. The second term corresponds to $x \ge T$ in which case we buy the skis at day T and pay $b + T$ overall. We denote by OPT_D the cost achieved by the optimal threshold for a given distribution D, i.e.,

$$\mathrm{OPT}_D = \min_{T\in[0,+\infty]}\left[\mathop{\mathbf{E}}_{x\sim D}[\mathbf{1}\{x < T\}x + (b+T)\mathbf{1}\{x > T\}]\right].$$

Observe that the optimal cost is given by a deterministic threshold T. We first define the notion of ε-additive approximation algorithms.

Definition 2.1 (ε-Additive Approximation)**.** *Given an instance of ski-rental in which the number of ski-days is drawn from an unknown distribution D, and the ski price is b, we say that an algorithm $\mathcal{A}$ achieves ε-additive approximation if,* $\mathrm{cost}(\mathcal{A}; D) \le \mathrm{OPT}_D + \varepsilon$, *where* OPT_D *denotes the optimal cost when D is known.*

The main result of this section is the following theorem, where we show that for any value of $\varepsilon > 0$, $\widetilde{O}(1/\varepsilon^2)$ samples from D suffices to design an ε-additive approximation algorithm for ski-rental over D without any assumption on the distribution D.

Theorem 2.2. *Let D denote the distribution of the number of ski-days. There exists an algorithm (see Algorithm 1) that for any $\varepsilon, \delta \in (0, 1]$, draws $\widetilde{O}(b^2\varepsilon^{-2}\log(1/\delta))$ samples from D, runs in time linear in the number of samples, and, with probability at least $1-\delta$, outputs a strategy $\mathcal{A}$ that satisfies* $\mathrm{cost}(\mathcal{A}; D) \le \mathrm{OPT}_D + \varepsilon$.

Proof. The proof follows from an application of Dvoretzky, Kiefer, and Wolfowitz (DKW) inequality. Let $X_1, \dots, X_n$ be i.i.d. samples from D and let D_n denote the (empirical) distribution constructed from the sampled points. Formally, the cdf of D_n is defined as follows:

$$P_{D_n}(x) = \frac{1}{n}\sum_{i=1}^{n}\mathbf{1}\{X_i \le x\} \quad x \in \mathbb{R}.$$

By DKW inequality, for $n = \Omega(b^2\varepsilon^{-2}\log(1/\delta))$, with probability at least $1-\delta$, it holds $d_K(D_n, D) \le \varepsilon/b$. Recall

[1] We denote by $\mathbf{1}\{x < t\}$ the indicator function that is 0 for all $x < t$ and 1 otherwise.

that, the Kolmogorov distance of distributions D and D' with cdf respectively F and F' is defined as $d_K(D, D') := \sup_{x\in\mathbb{R}} |F'(x) - F(x)|$.

For any threshold $\theta \in \mathbb{R}$, let $\mathcal{A}_\theta$ denote the algorithm that buys at θ. First, we prove the following result that relates the cost of an algorithm of the form $\mathcal{A}_\theta$ for ski-rental on D to its cost on a distribution D' where $d_K(D, D') \le \varepsilon/b$. The proof of the following claim can be found in the Supplementary Material.

Claim 2.3. *Suppose that $d_K(D, D') \le \varepsilon/b$. If there exists T' and $t^* \le T$ such that $\mathrm{cost}(\mathcal{A}_{T'}; D') \le \mathrm{OPT}_{D'} + \varepsilon$ and $\mathrm{cost}(\mathcal{A}_{t^*}; D) \le \mathrm{OPT}_D + \varepsilon$, then $\mathrm{cost}(\mathcal{A}_{T'}; D) \le \mathrm{OPT}_D + O(\varepsilon\ (T + T' + b)/b)$.*

Next, we bound the value T in Claim 2.3. In other words, we show that for any distribution there exists an ε-additive approximation strategy of form $\mathcal{A}_{t^*}$ with $t^* = O(b \log \frac{b}{\varepsilon})$.

Claim 2.4. *For any D, we can find a $t^* \le 2b \log \frac{b}{\varepsilon}$ such that $\mathrm{cost}(\mathcal{A}_{t^*}; D) \le \mathrm{OPT}_D + O(\varepsilon)$.*

Now, we are ready to describe our algorithm (Algorithm 1). By Claim 2.4, recall that $n = \Theta(b \log(1/\delta)/\varepsilon^2)$, we can find an ε-additive approximation strategy $\mathcal{A}_{\theta_\varepsilon}$, for D_n with $\theta_\varepsilon \le 2b \log \frac{b}{\varepsilon}$. Moreover, by another application of Claim 2.4, there exists $t^* = 2b \log \frac{b}{\varepsilon}$ such that $\mathrm{cost}(\mathcal{A}_{t^*}; D) \le \mathrm{OPT}_D + \varepsilon$. Hence, since with probability at least $1 - \delta$, $d_K(D, D_n) \le \varepsilon/b$, by an application of Claim 2.3 with $D' = D_n$, we obtain

$$\begin{aligned}\mathrm{cost}(\mathcal{A}_{\theta_\varepsilon}; D) &\le \mathrm{OPT}_D + O(\varepsilon \cdot (\theta_\varepsilon + t^* + b)/b) \\ &\le \mathrm{OPT}_D + O(\varepsilon\ \log(b/\varepsilon)).\end{aligned}$$

Hence, $\widetilde{O}(b^2\varepsilon^{-2} \log(1/\delta))$ samples from the input suffices to design an ε-additive approximation strategy for ski-rental on D. □

Algorithm 1 Algorithm for ε-Additive Approximation

1: **Input:** i.i.d. samples $X_1, \ldots, X_n \sim D$
2: **Output:** A buying time T
3: D_n is empirical distribution computed from $X_1, \ldots, X_n$
4: $T' \leftarrow$ optimal buying time for ski-rental on D_n
5: **return** $\min\{b \log \frac{1}{\varepsilon}, T'\}$

2.2. Multiplicative Guarantee for Log-Concave Distributions

In this subsection, we focus on the class of *log-concave* distributions and design an efficient algorithm that achieves $(1 + \varepsilon)$-multiplicative approximation for the case the number of days is drawn from a log-concave distribution. The main result of this section is the following theorem.

Theorem 2.5. *Assume that the number of ski-days follows a log-concave distribution D. There exists an algorithm that for any $\varepsilon, \delta \in (0, 1]$, draws $\widetilde{O}(\log(1/\delta)/\varepsilon^2)$ samples from D, runs in time linear in the number of samples, and, with probability at least $1 - \delta$, outputs a strategy $\mathcal{A}$ s.t.*

$$\mathrm{cost}(\mathcal{A}; D) \le (1 + \varepsilon)\mathrm{OPT}_D.$$

Proof. First, we show that the ε-additive approximation algorithm, Algorithm 1, can be used for designing $(1 + \varepsilon)$-multiplicative approximation algorithms on distributions whose optimal strategy is to either buy initially or rent indefinitely. The proof of the following claim can be found in the Supplementary Material.

Claim 2.6. *Let D be a distribution whose optimal strategy is either to buy initially or rent indefinitely. Fix $\varepsilon, \delta \in (0, 1]$. There exists an algorithm that draws $\widetilde{O}(\log(1/\delta)/\varepsilon^2)$ samples from D and with with probability at least $1 - \delta$ outputs a strategy $\mathcal{A}$ such that $\mathrm{cost}(\mathcal{A}; D) \le (1 + \varepsilon)\mathrm{OPT}_D$.*

Next, we show that the optimal strategy for log-concave distributions is to either buy initially or rent indefinitely. Let p_D and P_D respectively denote the pdf and the cdf of D. Let $q(\cdot)$ be the pdf corresponding to a randomized strategy where for each $x > 0$, $q(x)$ denotes the probability of stopping at time x. The cost of the strategy corresponding to q is,

$$\begin{aligned}\mathrm{cost}(\mathcal{A}; D) = \int_0^\infty q(x)\Big(\int_0^x y p_D(y)\, dy \\ + (b + x)(1 - P_D(x))\Big)\, dx\ . \qquad (1)\end{aligned}$$

Therefore, to minimize the cost, we need to set $q(\cdot)$ to be a point mass on x that minimizes

$$g(x) := \int_0^x y p_D(y)\, dy + (b + x)(1 - P_D(x))\ .$$

To find the minimizer of g, we compute the derivative $g'(x) = (1 - P_D(x)) - b p_D(x) = (1 - P_D(x))(1 - b\, \mathrm{hr}_D(x))$, where $\mathrm{hr}_D(x) = p_D(x)/(1 - P_D(x))$ is the *hazard rate* function of the distribution D. Since D is a log-concave distribution, we have that its hazard rate $\mathrm{hr}_D(x)$ is an increasing function of x.

1. If for all $x \in \mathbb{R}$, it holds $\mathrm{hr}_D(x) < 1/b$, then $g(\cdot)$ is an increasing function and the optimal strategy is to buy the skis initially. Similarly, if for all $x \in \mathbb{R}$, it holds $\mathrm{hr}_D(x) > 1/b$, then g is minimized at $+\infty$, i.e., the optimal strategy is to rent the skis indefinitely.

2. If for some $x_0 \in \mathbb{R}$, it holds $\mathrm{hr}_D(x_0) = 1/b$, then $g(\cdot)$ is increasing in $[0, x_0]$ and decreasing in $[x_0, +\infty)$. This means that g is either minimized for $x = 0$, or $x = +\infty$.

Therefore, when D is a log-concave distribution, the optimal strategy is always to either buy the skis initially or rent them indefinitely. The result follows from Claim 2.6. □

2.3. Consistency-Robustness Trade-Off of Ski-Rental in Distributional Setting

Here, we show that for any $\lambda > 1$, we can achieve$(\lambda, \frac{\lambda}{\lambda-1})$ consistency-robustness trade-off for ski-rental by applying the result of Mahdian et al. (2012) that works for the more general problem of *online resource allocation* in a black-box fashion.

In online resource allocation, we are given a sequence of jobs $\mathcal{J}$ and a set of servers $\mathcal{S}$. Jobs arrive one by one, and at each time t, the task is to assign the job J_t to a set of servers. Each server S_i has an activation cost c_i and assigns each job J_t to a server S_i along with a serving cost d_i. The goal is to find an assignment that minimizes the cost of activating servers plus the serving costs. The ski-rental problem is a special case of this problem where we have two servers "buy" (S_1) and "rent" (S_2) with ($c_1 = b, d_1 = 0$) and ($c_2 = 0, d_2 = 1$). Moreover, each ski-day is a job.

Lemma 2.7 (Theorem 3.1 of Mahdian et al.). *For any $\gamma > 1$, there exists an algorithm $\mathcal{H}(\gamma)$ that given two algorithms $\mathcal{A}, \mathcal{B}$ for an online resource allocation problem $\mathcal{P}$, satisfies*

$$\mathrm{cost}(\mathcal{H}(\gamma); \mathcal{I}) \leq \min\{\lambda\, \mathrm{cost}(\mathcal{A}; \mathcal{I}), (\frac{\lambda}{\lambda - 1})\, \mathrm{cost}(\mathcal{B}; \mathcal{I})\}.$$

Corollary 2.8. *Given two strategies $\mathcal{A}, \mathcal{B}$ for ski-rental, for any instance $\mathcal{I}$, and for any $\lambda > 1$, the cost the strategy* SKI-RENTAL*($\mathcal{A}, \mathcal{B}, \lambda$), Algorithm 2, on instance $\mathcal{I}$ is at most* $\min\{\lambda\, \mathrm{cost}(\mathcal{A}; \mathcal{I}), (\frac{\lambda}{\lambda-1})\, \mathrm{cost}(\mathcal{B}; \mathcal{I})\}$.

Algorithm 2 SKI-RENTAL: consistent and robust algorithm for ski-rental adapted from (Mahdian et al., 2012).

Input: Two strategies $\mathcal{A}, \mathcal{B}$ for ski-rental
Output: A buying time T
$t \leftarrow 0$ {number of days}
while ski-day **do**
  $t \leftarrow t + 1$
  if $\mathrm{cost}_t(\mathcal{A}) \leq (1 - \gamma)\mathrm{cost}_t(\mathcal{B})$ **then**
    if $\mathcal{A}$ buys at t **buy**; otherwise, **rent**
  else
    if $\mathcal{B}$ buys at t **buy**; otherwise, **rent**
  end if
end while

Now we are ready to we prove the main result of the section which is a formal restatement of Result 1.

Theorem 2.9. *For any $\lambda > 1, \delta \in (0, 1]$, there exists an algorithm that draws $\widetilde{O}(\log(1/\delta)/\varepsilon^2)$ samples from the input distribution D, runs in time linear in the number of samples and, with probability at least $1 - \delta$, returns a $(\lambda(1 + \varepsilon))$-consistent and $(\frac{\lambda}{\lambda-1})$-robust strategy $\mathcal{A}$ for ski-rental under log-concave distributions, i.e., if D is a log-concave distribution,* $\mathrm{cost}(\mathcal{A}; D) \leq \lambda(1 + \varepsilon)\mathrm{OPT}_D$*; otherwise,* $\mathrm{cost}(\mathcal{A}; D) \leq \frac{\lambda}{\lambda-1}\mathrm{OPT}_{\mathrm{ONL}}$.

Proof. Let $\mathcal{A}_\mathcal{X}$ be the strategy guaranteed by Theorem 2.5 that receives samples $\mathcal{X} = \{X_1, \ldots, X_{\widetilde{O}(1/\varepsilon^2 \log(1/\delta))}\}$ from D. Moreover, let $\mathcal{A}_{\mathrm{ONL}}$ be a worst-case optimal algorithm (i.e., with competitive ratio $\frac{e}{e-1}$) for ski-rental.

By Theorem 2.5, for log-concave distributions, with probability at least $1 - \delta$, $\mathrm{cost}(\mathcal{A}_\mathcal{X}; D) \leq (1 + \varepsilon)\, \mathrm{OPT}_D$. Then, by an application of Corollary 2.8, SKI-RENTAL($\mathcal{A}_\mathcal{X}, \mathcal{A}_{\mathrm{ONL}}$) satisfied the desired consistency and robustness guarantees. □

2.4. Sample Complexity Lower Bound for Exponential Distributions

In this subsection, we bound from below the sample complexity for any algorithm that learns a strategy that achieves $(1 + \varepsilon)\mathrm{OPT}_D$ cost under log-concave distributions. Interestingly, the lower bound holds for the simple case of exponential distributions which are a (very small) subset of log-concave distributions.

Theorem 2.10. *Suppose that there exists an algorithm that for any log-concave distribution D and any $\varepsilon \in (0, 1]$, draws k samples from D and, with probability at least $2/3$, outputs a strategy $\mathcal{A}$ such that* $\mathrm{cost}(\mathcal{A}; D) \leq (1 + \varepsilon)\mathrm{OPT}_D$ *. Then $k = \Omega(1/\varepsilon^2)$.*

Proof. We reduce the above optimization problem to a distribution testing problem. In particular, we show that given an algorithm $\mathcal{A}$ that satisfies the guarantees of Theorem 2.10, we can construct an algorithm that identifies the distribution D that generated the samples. Set the buying cost of the ski-rental problem $b = 1$. For our proof, we only need to consider two distributions. We set D_1 to be the exponential distribution with rate $\lambda_1 = 1/(1 + 2\varepsilon)$ and D_2 to be the exponential distribution with rate $\lambda_2 = 1/(1 - 2\varepsilon)$. Observe that since exponential distributions are log-concave, their optimal strategy is to either buy initially or rent indefinitely, see the proof of Theorem 2.5. In particular, the optimal strategy for D_1 is to buy initially and $\mathrm{OPT}_{D_1} = 1$, while the optimal strategy for D_2 is to rent indefinitely and $\mathrm{OPT}_{D_2} = 1 - 2\varepsilon$. Let $X_1, \ldots, X_k$ be k samples from D_i and denote by q the distribution/strategy that algorithm $\mathcal{A}(X_1, \ldots, X_k)$ outputs. Notice that this captures also randomized algorithms that decide to buy at some time t with density $q(t)$. We consider the following testing algorithm $\mathcal{T}$:

- If $\mathrm{cost}(q; D_2) \geq (1 - 2\varepsilon)\mathrm{cost}(q; D_1)$ then output 1, i.e., the distribution that generated the samples is D_1.

- Otherwise, output 2.

Our proof crucially relies on the following claim. We show that any strategy q that performs well under distribution D_1 cannot perform well under distribution D_2. We provide its proof in the Supplementary Material.

Claim 2.11. *Let q be any distribution on $[0, \infty]$ such that $\mathrm{cost}(q; D_1) \le (1+\varepsilon)\mathrm{OPT}_{D_1}$. Then, for ε sufficiently small it holds $\mathrm{cost}(q; D_2) > (1+\varepsilon)\mathrm{OPT}_{D_2}$.*

Assume, in order to reach to contradiction, that the testing algorithm $\mathcal{T}$ outputs 2 and the underlying distribution is D_1, that is it holds $\mathrm{cost}(q; D_2) < (1-2\varepsilon)\mathrm{cost}(q; D_1)$. By the guarantee for the strategy $\mathcal{A}$ we have that $\mathrm{cost}(q; D_1) \le (1+\varepsilon)\mathrm{OPT}_{D_1}$. This implies

$$\mathrm{cost}(q; D_2) < (1-2\varepsilon)\mathrm{cost}(q; D_1) \\ = (1-2\varepsilon)(1+\varepsilon)\mathrm{OPT}_{D_1} = (1-2\varepsilon)(1+\varepsilon)\,.$$

From Claim E.1, we have that $\mathrm{cost}(q; D_2) > (1+\varepsilon)\mathrm{OPT}_{D_2} = (1+\varepsilon)(1-2\varepsilon)$. Therefore, we conclude $(1-2\varepsilon)(1+\varepsilon) > \mathrm{cost}(q; D_2) > (1+\varepsilon)(1-2\varepsilon)$, which is a contradiction. The case where the algorithm outputs 1 and the underlying distribution is D_2 is similar. Therefore, our testing algorithm can distinguish the two exponential distributions whose mean differ by $O(\varepsilon)$. It is not hard to see that, to distinguish between these two exponential distributions, $\Omega(1/\varepsilon^2)$ samples are required. For the details, see the full proof provided in the Supplementary material. □

3. Prophet Inequality

In this section we prove our result for prophet inequalities, Result 2. Any strategy in this setting can be described with a set of thresholds $T_1, \ldots T_n$, see Algorithm 3. Since we should always accept the last value X_n, the last threshold T_n is redundant and should always be equal to 0. To keep notation simple we assume that this is the case and use n-thresholds.

Algorithm 3 Threshold-Algorithm

```
1: Input: Thresholds {T_i}_{i∈[n]}, values {X_i ~ D_i}_{i∈[n]}
2: Output: One of the values X_i's
3: i ← 1
4: while X_i < T_i do i ← i + 1 end while
5: return X_i.
```

We first define the gain of a strategy $T_1, \ldots, T_n$.

Definition 3.1 (Gain of a Strategy)**.** *Fix distributions $D_1, \ldots, D_n$. Let $T_1, \ldots, T_n$ be a set of thresholds. We define the $\mathrm{gain}(T_1, \ldots, T_n; D_1, \ldots, D_n)$ to be the expected output of Algorithm 3 with thresholds $T_1, \ldots, T_n$. When, it is clear from the context, we may drop the distributions and simply write $\mathrm{gain}(T_1, \ldots, T_n)$.*

The following fact gives the optimal way to choose the thresholds $T_1, \ldots, T_n$ given that we know exactly $D_1, \ldots, D_n$. For the proof, see Supplementary Material.

Fact 3.2. *Algorithm 3 is optimal with respect to $D_1, \ldots, D_n$ when the thresholds are $T_i = \mathbf{E}_{X_i \sim D_i}[\max(X_i, T_{i+1})]$, $T_{n-1} = \mathbf{E}_{X_n \sim D_n}[X_n]$, $T_n = 0$. We use OPT to denote the gain of the optimal algorithm that knows the distributions, see Fact 3.2.*

For example, with two distributions D_1 and D_2, the optimal gain according to the fact above is $\mathrm{OPT} = \mathbf{E}_{X_1 \sim D_1}[\max(X_1, \mathbf{E}_{X_2 \sim D_2}[X_2])]$.

We present below the main theorem of this section, i.e., the formal statement of Result 2.

Theorem 3.3. *For any $\lambda > 1$ and $\varepsilon, \delta \in (0, 1]$, there exists a randomized algorithm that draws $\widetilde{O}(n^3/\varepsilon^2)$ samples from the distributions $D_1, \ldots, D_n$ and, in sample near-linear time, with probability at least $1-\delta$, outputs an $((1+\varepsilon)\lambda)$-consistent and $(2\lambda/(\lambda-1))$-robust strategy for prophet inequality under log-concave distributions.*

For the proof of Theorem 3.3 we rely on two main components. The first is the following lemma that shows that when the underlying distributions are log-concave we can obtain multiplicative approximations to the optimal gain.

Lemma 3.4. *Let $X_1, \ldots, X_n$ be non-negative independent random variables with log-concave densities. There is an algorithm such that for any $\varepsilon, \delta \in (0, 1]$ draws $O(\frac{n^2}{\varepsilon^2} \log(n/\delta))$ samples from each distribution D_i, and computes in near-linear sample time a set of thresholds $T_1, \ldots, T_n$ such that, with probability at least $1-\delta$, it holds $\mathrm{OPT}/\mathrm{gain}(T_1, \ldots, T_n) \le 1 + \varepsilon$.*

Theorem 3.3 follows by combining our multiplicative approximation result of Lemma 3.4 that works under the assumption that the distributions are log-concave with a recent result from (Rubinstein et al., 2019) for the prophet inequality problem showing that with just one sample from each of $D_1, \ldots, D_n$ we can obtain value $(1/2)\mathrm{OPT}_{\mathrm{ONL}}$.

Lemma 3.5 (Theorem 1 of (Rubinstein et al., 2019))**.** *There is an algorithm that draws n samples from $D_1, \ldots, D_n$, runs in sample linear time, and outputs a strategy $\mathcal{A}$ such that $\mathrm{gain}(\mathcal{A}) \ge (1/2)\mathrm{OPT}_{\mathrm{ONL}}$.*

We now proceed with the proof of Lemma 3.4.

Proof sketch of Lemma 3.4. Our proof crucially relies on the following result where we prove that, when the underlying distribution of a random variable $X \ge 0$ is log-concave, we can obtain multiplicative estimates of thresholds, i.e., of quantities of $\max(X, c)$ with a sample complexity independent of the support size of X.

Lemma 3.6. *Let X be a non-negative random variable with log-concave density and let $c > 0$. Fix $\varepsilon, \delta \in (0, 1]$, then by drawing $N = O(\frac{1}{\varepsilon^2}\log(1/\delta))$ samples, we can compute, in sample linear time, an estimate $\widehat{T}$ such that*

$$\frac{1}{1+\varepsilon} \leq \frac{\widehat{T}}{\mathbf{E}[\max(X,c)]} \leq 1+\varepsilon\ .$$

Proof. Denote $\mu = \mathbf{E}[\max(X, c)]$. We draw n i.i.d. samples $X_1, \ldots, X_n$ from the log-concave distribution and use the empirical average $\hat{T} = \frac{1}{n}\sum_{i=1}^{n}\max(c, X_i)$. By Chebyshev's inequality we obtain

$$\Pr\left[\left|\hat{T}-\mu\right| \geq \epsilon\mu\right] \leq \frac{\mathbf{Var}[\hat{T}]}{\varepsilon^2\mu^2} = \frac{\mathbf{Var}[\max(c,X)]}{n\varepsilon^2\mu^2}\ . \quad (2)$$

To bound $\mathbf{Var}[\max(c, X)]$ consider X' to be an independent copy of X. We have

$$\begin{aligned}\mathbf{Var}[\max(c,X)] &= \frac{1}{2}\,\mathbf{E}[(\max(c,X)-\max(c,X'))^2]\\ &\leq \frac{1}{2}\,\mathbf{E}[(X-X')^2] = \mathbf{Var}[X]\,.\end{aligned}$$

In order to bound the variance of X we are going to use the following reverse Hölder inequality for log-concave distributions, see, for example, (Lovász & Vempala, 2007).

Lemma 3.7 (Reverse Hölder for Log-concave Measures)**.** *Let X be a random variable distributed according to some log-concave density on $\mathbb{R}$. It holds $\mathbf{E}[|X|^k| \leq (2k)^k(\mathbf{E}[|X|])^k$.*

Since X is distributed according to a log-concave density we have that $\mathbf{Var}[X] \leq \mathbf{E}[X^2] \leq 16\,\mathbf{E}[|X|]^2 = 16\,\mathbf{E}[X]^2$, where the last equality follows from the fact that X is non-negative. Finally, we have that $\mathbf{E}[X] \leq \mathbf{E}[\max(c, X)] = \mu$ and therefore, from Equation (2), we obtain that $\Pr\left[\left|\hat{T}-\mu\right| \geq \epsilon\mu\right] \leq \frac{16}{n\varepsilon^2}$. Therefore, with $O(1/\varepsilon^2)$ samples we have that with probability at least $2/3$ it holds $|\hat{T}/\mu - 1| \leq \varepsilon$ which also implies that $1/(1+\varepsilon) \leq \hat{T}/\mu \leq 1+\varepsilon$ by the fact that $1-\varepsilon \leq 1/(1+\varepsilon)$ for all $\varepsilon \in [0, 1]$. To amplify the success probability to $1-\delta$ for any $\delta > 0$ we can use the "median-trick", i.e., repeat the above process $M = O(\log(1/\delta))$ times and keep the median of the estimates $\hat{T}_1, \ldots, \hat{T}_M$. Since each one of them satisfies the error guarantee with probability at least $2/3$ we have that the probability that the median violates the same error guarantee is at most $(2/3)^{M/2} \leq \delta$. This holds because for the median to be outside the interval $[\mu - \varepsilon\mu, \mu + \varepsilon\mu]$ we need at least half of $\hat{T}_1, \ldots, \hat{T}_M$ to fall outside the same interval. Overall, we obtain that with $O(1/\varepsilon^2\log(1/\delta)$ we can compute an estimate $\hat{T}$ such that $1/(1+\varepsilon) \leq \hat{T}/\mu \leq 1+\varepsilon$. ☐

We now return to the proof of Lemma 3.4. For simplicity assume that we have two distributions D_1 and D_2. Let $\hat{T}$ be an $(1+\varepsilon)$-multiplicative approximation of the optimal threshold $T = \mathbf{E}[X_2]$, calculated using $O(1/\varepsilon^2\log(1/\delta))$ samples, see Lemma 3.6 (set $c = 0$).

$$\begin{aligned}\mathrm{gain}(\hat{T}) &= \mathbf{E}[X_1\mathbf{1}\{X_1 \geq \hat{T}\} + X_2\mathbf{1}\{X_1 < \hat{T}\}]\\ &= \mathbf{E}[X_1\mathbf{1}\{X_1 \geq \hat{T}\} + \mathbf{E}[X_2]\mathbf{1}\{X_1 < \hat{T}\}]\\ &\geq \mathbf{E}[X_1\mathbf{1}\{X_1 \geq \hat{T}\} + \frac{\hat{T}}{1+\varepsilon}\mathbf{1}\{X_1 < \hat{T}\}]\\ &\geq 1/(1+\varepsilon)\,\mathbf{E}[\max(X_1, \hat{T})]\\ &\geq 1/(1+\varepsilon)^2\mathrm{OPT}\end{aligned}$$

where we used $T \geq \hat{T}/(1+\varepsilon) \geq T/(1+\varepsilon)^2$. Similar analysis can be applied when we have n distributions to obtain $(1+\varepsilon)^{2n}\mathrm{gain}(\hat{T}_1, \ldots, \hat{T}_n) \geq \mathrm{OPT}$. By setting $\varepsilon' = \Theta(\varepsilon/n)$, we have $(1+\varepsilon')\mathrm{gain}(\hat{T}_1, \ldots, \hat{T}_n) \geq \mathrm{OPT}$. For this value of ε' we obtain that $\widetilde{O}(n^2/\varepsilon^2\log(1/\delta))$ samples from each D_i suffice to estimate the thresholds $\hat{T}_1, \ldots, \hat{T}_n$. The full proof is deferred to Supplementary Material. ☐

Proof of Theorem 3.3. Let $\mathcal{A}$ be the randomized strategy that with probability $1/\lambda$ uses the $(1+\varepsilon)$-multiplicative strategy for log-concave distributions (Lemma 3.4), otherwise, it uses the worst-case optimal algorithm for the online setting given in Lemma 3.5, which has $(1/2)$-competitive ratio. If the underlying distributions are log-concave, the expected gain of $\mathcal{A}$ is with probability $1/\lambda$, $\mathrm{OPT}/(1+\varepsilon)$ and with probability $(\lambda-1)/\lambda$ at least zero. Thus, the algorithm $\mathcal{A}$ is $((1+\varepsilon)\lambda)$-consistent. On the other case, where the underlying distributions are arbitrary, the expected gain of $\mathcal{A}$ is $(1/2)\mathrm{OPT}_{\mathrm{ONL}}$ with probability $1-1/\lambda$, and at least zero, otherwise. Thus, $\mathcal{A}$ is $(2\lambda/(\lambda-1))$-robust. ☐

Similar to the additive result for ski-rental, we remark that we can obtain additive approximation guarantees in the prophet inequality setting. We remark that we assume nothing about the distributions $D_1, \ldots, D_n$ apart from their support being bounded in some interval $[0, b]$.

Theorem 3.8. *Let $X_1, \ldots, X_n$ be non-negative independent random variables with maximum value $b > 0$. There is an algorithm such that for any $\varepsilon, \delta \in (0, 1]$, draws $\widetilde{O}(\frac{nb^2}{\varepsilon^2}\log(1/\delta))$ samples from each one, and in sample polynomial time, computes a set of thresholds $T_1, \ldots, T_n$ such that* $\mathrm{OPT} - \mathrm{gain}(T_1, \ldots, T_n) \leq \varepsilon\,.$

Acknowledgements

This project was supported by NSF award #2006206. The authors would like to thank the anonymous reviewers for their insightful comments and suggestions.

References

Alabi, D., Kalai, A. T., Liggett, K., Musco, C., Tzamos, C., and Vitercik, E. Learning to prune: Speeding up repeated computations. In *Conference on Learning Theory*, pp. 30–33. PMLR, 2019.

Anand, K., Ge, R., and Panigrahi, D. Customizing ML predictions for online algorithms. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 303–313. PMLR, 2020.

Angelopoulos, S., Dürr, C., Kamali, S., Renault, M., and Rosén, A. Online bin packing with advice of small size. In *Workshop on Algorithms and Data Structures*, pp. 40–53. Springer, 2015.

Angelopoulos, S., Dürr, C., Kamali, S., Jin, S., and Renault, M. Online computation with untrusted advice. In *11th Innovations in Theoretical Computer Science Conference (ITCS 2020)*, volume 151, pp. 52–1. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2020.

Antoniadis, A., Gouleakis, T., Kleer, P., and Kolev, P. Secretary and online matching problems with machine learned advice. *Advances in Neural Information Processing Systems*, 2020.

Bagnoli, M. and Bergstrom, T. Log-concave probability and its applications. *Economic theory*, 26(2):445–469, 2005.

Balcan, M.-F., Nagarajan, V., Vitercik, E., and White, C. Learning-theoretic foundations of algorithm configuration for combinatorial partitioning problems. In *Conference on Learning Theory*, pp. 213–274. PMLR, 2017.

Balcan, M.-F., Dick, T., Sandholm, T., and Vitercik, E. Learning to branch. In *International conference on machine learning*, pp. 344–353. PMLR, 2018.

Bamas, E., Maggiori, A., and Svensson, O. The primal-dual method for learning augmented algorithms. *Advances in Neural Information Processing Systems*, 2020.

Bhaskara, A., Cutkosky, A., Kumar, R., and Purohit, M. Online learning with imperfect hints. In *International Conference on Machine Learning*, pp. 822–831. PMLR, 2020.

Canonne, C. L., Ron, D., and Servedio, R. A. Testing probability distributions using conditional samples. *SIAM Journal on Computing*, 44(3):540–616, 2015.

Chakraborty, S., Fischer, E., Goldhirsh, Y., and Matsliah, A. On the power of conditional samples in distribution testing. *SIAM Journal on Computing*, 45(4):1261–1296, 2016.

Cohen, E., Geri, O., and Pagh, R. Composable sketches for functions of frequencies: Beyond the worst case. In *International Conference on Machine Learning*, pp. 2057–2067. PMLR, 2020.

Dai, H., Khalil, E. B., Zhang, Y., Dilkina, B., and Song, L. Learning combinatorial optimization algorithms over graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 6351–6361, 2017.

Dai, Z. and Shrivastava, A. Adaptive learned bloom filter (ada-bf): Efficient utilization of the classifier. *arXiv preprint arXiv:1910.09131*, 2019.

Dütting, P., Lattanzi, S., Leme, R. P., and Vassilvitskii, S. Secretaries with advice. *arXiv preprint arXiv:2011.06726*, 2020.

Dvoretzky, A., Kiefer, J., and Wolfowitz, J. Asymptotic minimax character of the sample distribution function and of the classical multinomial estimator. *The Annals of Mathematical Statistics*, pp. 642–669, 1956.

Esfandiari, H., Korula, N., and Mirrokni, V. Allocation with traffic spikes: Mixing adversarial and stochastic models. *ACM Transactions on Economics and Computation (TEAC)*, 6(3-4):1–23, 2018.

Gollapudi, S. and Panigrahi, D. Online algorithms for rent-or-buy with expert advice. In *International Conference on Machine Learning*, pp. 2319–2327, 2019.

Grove, E. F. Online bin packing with lookahead. In *SODA*, volume 95, pp. 430–436, 1995.

Gupta, R. and Roughgarden, T. A pac approach to application-specific algorithm selection. *SIAM Journal on Computing*, 46(3):992–1017, 2017.

Hsu, C.-Y., Indyk, P., Katabi, D., and Vakilian, A. Learning-based frequency estimation algorithms. In *International Conference on Learning Representations*, 2019.

Indyk, P., Vakilian, A., and Yuan, Y. Learning-based low-rank approximations. *arXiv preprint arXiv:1910.13984*, 2019.

Indyk, P., Mallmann-Trenn, F., Mitrović, S., and Rubinfeld, R. Online page migration with ml advice. *arXiv preprint arXiv:2006.05028*, 2020.

Jiang, T., Li, Y., Lin, H., Ruan, Y., and Woodruff, D. P. Learning-augmented data stream algorithms. *ICLR*, 2020.

Karlin, A. R., Manasse, M. S., McGeoch, L. A., and Owicki, S. Competitive randomized algorithms for nonuniform problems. *Algorithmica*, 11(6):542–571, 1994.

Kleinberg, R. and Weinberg, S. M. Matroid prophet inequalities. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pp. 123–136, 2012.

Kodialam, R. Optimal algorithms for ski rental with soft machine-learned predictions. *arXiv preprint arXiv:1903.00092*, 2019.

Koutsoupias, E. and Papadimitriou, C. H. Beyond competitive analysis. *SIAM Journal on Computing*, 30(1): 300–317, 2000.

Kraska, T., Beutel, A., Chi, E. H., Dean, J., and Polyzotis, N. The case for learned index structures. In *Proceedings of the 2018 International Conference on Management of Data*, pp. 489–504, 2018.

Krengel, U. and Sucheston, L. On semiamarts, amarts, and processes with finite value. *Advances in Probability and Related Topics, 4:197-266*, 1978.

Lattanzi, S., Lavastida, T., Moseley, B., and Vassilvitskii, S. Online scheduling via learned weights. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1859–1877. SIAM, 2020.

Lavastida, T., Moseley, B., Ravi, R., and Xu, C. Learnable and instance-robust predictions for online matching, flows and load balancing. *arXiv preprint arXiv:2011.11743*, 2020.

Le Cam, L. *Asymptotic methods in statistical decision theory*. Springer Science & Business Media, 1987.

Lovász, L. and Vempala, S. The geometry of logconcave functions and sampling algorithms. *Random Structures & Algorithms*, 30(3):307–358, 2007.

Lykouris, T. and Vassilvtiskii, S. Competitive caching with machine learned advice. In *International Conference on Machine Learning*, pp. 3296–3305, 2018.

Mahdian, M., Nazerzadeh, H., and Saberi, A. Online optimization with uncertain information. *ACM Transactions on Algorithms (TALG)*, 8(1):1–29, 2012.

Mitzenmacher, M. A model for learned bloom filters, and optimizing by sandwiching. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 462–471, 2018.

Mitzenmacher, M. and Vassilvitskii, S. Algorithms with predictions. *arXiv preprint arXiv:2006.09123*, 2020.

Purohit, M., Svitkina, Z., and Kumar, R. Improving online algorithms via ml predictions. In *Advances in Neural Information Processing Systems*, pp. 9661–9670, 2018.

Raghavan, P. A statistical adversary for on-line algorithms. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 7:79–83, 1992.

Renault, M. P. and Rosén, A. On online algorithms with advice for the k-server problem. *Theory of Computing Systems*, 56(1):3–21, 2015.

Rohatgi, D. Near-optimal bounds for online caching with machine learned advice. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1834–1845. SIAM, 2020.

Roughgarden, T. Beyond the worst-case analysis of algorithms (introduction). *arXiv preprint arXiv:2007.13241*, 2020.

Rubinstein, A., Wang, J. Z., and Weinberg, S. M. Optimal single-choice prophet inequalities from samples. *arXiv preprint arXiv:1911.07945*, 2019.

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

Vaidya, K., Knorr, E., Kraska, T., and Mitzenmacher, M. Partitioned learned bloom filter. *International Conference on Learning Representations*, 2021.

Wang, S., Li, J., and Wang, S. Online algorithms for multi-shop ski rental with machine learned advice. *Advances in Neural Information Processing Systems*, 33, 2020.