Decision-Dependent Risk Minimization in Geometrically Decaying Dynamic Environments

Mitas Ray, Lillian J. Ratliff, Dmitriy Drusvyatskiy, Maryam Fazel

University of Washington, Seattle

Abstract

This paper considers the problem of expected loss minimization given a data distribution that is dependent on the decision-maker's action and evolves dynamically in time according to a geometric decay process. Motivated by practice, empirical information settings are considered: namely, the decision-maker either has oracle access, for a fixed batch size, to the empirical gradient of the loss (first order setting), or the empirical loss function (zero order setting). Novel algorithms for each of these settings are introduced, both of which operate on the same underlying principle: the decision-maker repeatedly deploys a fixed decision over a fixed length epoch, thereby allowing the dynamically changing environment to sufficiently mix before updating the decision. The proposed algorithms are shown to converge to the optimal point. Specifically, high-probability sample complexity guarantees are given, which depend exponentially on the epoch length and logarithmically on the batch size. The algorithms are evaluated on a "semi-synthetic" example using real world data from the SFpark dynamic pricing pilot study; it is shown that the announced prices result in an improvement for the institution's objective (target occupancy), while achieving an overall reduction in parking rates.

1 Introduction

Traditionally, supervised machine learning algorithms are trained based on past data under the assumption that the past data is representative of the future. However, machine learning algorithms are increasingly being used in settings where the output of the algorithm changes the environment and hence, the data distribution. Indeed, loan procurement processes, online labor markets (Anagnostopoulos et al. 2018; Horton 2010), predictive policing (Lum and Isaac 2016), on-street parking (Pierce and Shoup 2018; Dowling, Ratliff, and Zhang 2020), and vehicle sharing markets (Banerjee, Riquelme, and Johari 2015) are all examples of real-world settings in which the algorithm's decisions change the underlying data distribution in large part due to the fact that the algorithm interacts with strategic users.

To address this problem, the machine learning community introduced the problem of *performative prediction* which models the data distribution as being *decision-dependent* thereby accounting for feedback induced distributional shift (Perdomo et al. 2020; Miller, Perdomo, and Zrnic 2021; Drusvyatskiy and Xiao 2020; Brown, Hod, and Kalemaj 2020; Mendler-Dünner et al. 2020). With the exception of (Brown, Hod, and Kalemaj 2020), this work has focused on static environments.

In many of the aforementioned application domains, however, the underlying data distribution also may have memory or even be changing dynamically in time. When a decision-making mechanism is announced it may take time to see the full effect of the decision as the environment and strategic data sources respond given their prior history or interactions.

For example, many municipalities announce quarterly a new quasi-static set of prices for on-street parking. In this setting, the institution may adjust parking rates for certain blocks in order to to achieve a desired occupancy range to reduce cruising phenomena and increase business district vitality (Fiez et al. 2018; Dowling et al. 2017; Pierce and Shoup 2013; Shoup 2006). For instance, in high traffic areas, the institution may announce increased parking rates to free up parking spots and redistribute those drivers to less populated blocks. However, upon announcing a new price, the population may react slowly, whether it be from initially being unaware of the price change, to facing natural inconveniences from changing one's parking routine. This introduces dynamics into our setting; hence, the data distribution takes time to equilibrate after the pricing change is made.

Motivated by such scenarios, we study the problem of decision-dependent risk minimization (or, synonymously, performative prediction) in dynamic settings wherein the underlying decision-dependent distribution evolves according to a geometrically decaying process. Taking into account the time it takes for a decision to have the full effect on the environment, we devise an algorithmic framework for finding the optimal solution in settings where the decision maker has access to different types of gradient information.

For both information settings (gradient access and loss function access, via the appropriate oracle), the decision-maker deploys the current decision for a fixed number of steps (the length of an epoch) allowing the dynamically evolving distribution to approach the fixed point distribution for that announced decision. At the end of the epoch, the decision is updated using a first-order or zeroth-order oracle.

One interpretation of this procedure is that the environ-

ment is operating on a faster timescale compared to the update of the decision-maker's action. For instance, consider the dynamically changing distribution as the data distribution corresponding to a population of strategic data sources. The phase during which the same decision is deployed for a fixed number of steps can be interpreted as the population of agents adapting at a faster rate than the update of the decision. This in fact occurs in many practical settings such as on-street parking, wherein prices and policies more generally are *quasi-static*, meaning they are updated infrequently relative to actual curbspace utilization.

1.1 Contributions

For the decision-dependent learning problem in geometrically decaying environments, we propose first-order or zeroth-order oracle algorithms (and characterize their sample complexity) that converge to the optimal point under appropriate assumptions, which make the performative risk minimization problem strongly convex.

- **Zero Order Oracle** (Algorithm 1, Section 3): with probability $1-\rho$, for $n \geq \Omega(\log \eta)$, $\eta > 1/(3\alpha t)$, and $m \geq \Omega(d \log d)$, the sample complexity is $O(t^{-1/3} + \sqrt{d \log(1/\rho) \log(m)/m})$.
- **First Order Oracle** (Algorithm 2, Section 3): with probability $1-\rho$, for arbitrary error tolerance $\epsilon > 0$, epoch length $n \geq \Omega(\log \epsilon)$, time horizon t, and batch size $m \geq \Omega(d \log d)$, the sample complexity is

  $$O((1-\alpha/M)^t + \epsilon + \sqrt{d \log(1/\rho) \log(m)/m}),$$

  where the loss is α-strongly convex and M-smooth, and d is the dimension of the decision space.

The technical novelty arises from bounding the error between expected and empirical gradients, considering that the distribution with respect to which gradient information is available is changing dynamically over time. We show that with high probability, the sample complexity is nearly the optimal rate in the setting where the expected loss is available, and characterize how it depends on the batch size m.

The algorithms are applied to a set of *semi-synthetic* experiments using real data from the SFpark pilot study on the use of dynamic pricing to manage curbside parking (Section 4). The experiments demonstrate that optimizing taking into consideration feedback-induced distribution shift even in a dynamic environment leads to the institution—and perhaps surprisingly, the user as well—experiencing lower expected cost. Moreover, there are important secondary effects of this improvement including increased access to parking—hence, business district vitality—and reduced circling for parking and congestion which not only saves users time but also reduces carbon emissions (Shoup 2006). A more comprehensive set of experiments are contained in Appendix D.

1.2 Related work

Dynamic Decision-Dependent Optimization. As hinted above, dynamic decision-dependent optimization has been considered quite extensively in the stochastic optimization literature wherein the problem of *recourse* arises due to decision-makers being able to make a secondary decision after some information has been revealed (Jonsbråten, Wets, and Woodruff 1998; Goel and Grossmann 2004; Varaiya and Wets 1988). In this problem, the goal of the institution is to solve a multi-stage stochastic program, in which the probability distribution of the population is a function of the decision announced by the institution. This multi-stage procedure models a dynamic process. Unlike the setting considered in this paper, the institution has the ability to make a recourse decision upon observing full or partial information about the stochastic components.

Reinforcement Learning. Reinforcement learning is a more closely related problem in the sense that a decision is being made over time where the environment dynamically changes as a function of the state and the decision-maker's actions (Sutton and Barto 2018). A subtle but important difference is that the setting we consider is such that the decision maker's objective is to find the action which optimizes the decision-dependent expected risk at the fixed point distribution (cf. Definition 1, Section 2) induced by the optimal action and the environment dynamics. This is in contrast to finding a policy which is a state-dependent distribution over actions given an accumulated cost over time. Our setting can be viewed as a special case of the general reinforcement learning problem, however with additional structure that is both practically well-motivated, and beneficial to exploit in the design and analysis of algorithms. More concretely, we crucially exploit the assumed model of environment dynamics (in this case, the geometric decay), the distribution dependence, and convexity to obtain strong convergence guarantees for the algorithms proposed herein.

Performative prediction. As alluded to in the introductory remarks, the most closely related body of literature is on performative prediction wherein the decision-maker or optimizer takes into consideration that the underlying data distribution depends on the decision. Performative prediction has been studied in both static and dynamic environments. In the static setting, as the distribution shifts, a naïve strategy is to re-train the model on this new distribution using heuristics to determine when to trigger the retraining process. Under the guise that if retraining is repeated, eventually the distribution will stabilize, early works on performative prediction—such as the works of Perdomo et al. (2020) and Mendler-Dünner et al. (2020)—studied this equilibrium notion, and called these points *performatively stable*. Mendler-Dünner et al. (2020) and Drusvyatskiy and Xiao (2020) study stochastic optimization algorithms applied to the performative prediction problem and recover optimal convergence guarantees to the performatively stable point. Yet, performatively stable points may differ from the optimal solution of the decision-dependent risk minimization problem as was shown in Perdomo et al. (2020). Taking this gap between stable and optimal points into consideration, Miller, Perdomo, and Zrnic (2021) characterize when the performative prediction problem is strongly convex, and devise a two-stage algorithm for finding the so-called *performatively optimal* solution—that is, the optimal solution to the decision-dependent risk minimization problem—when the decision-dependent distribu-

tion is from the location-scale family.

None of the aforementioned works consider dynamic environments. Brown, Hod, and Kalemaj (2020) is the first paper, to our knowledge, to investigate the dynamic setting for performative prediction. Assuming regularity properties of the dynamics, they show that classical retraining algorithms (repeated gradient descent and repeated risk minimization) converge to the performatively stable point of the expected risk at the corresponding fixed point distribution. Counter to this, in this paper we propose algorithms for the dynamic setting which target performatively optimal points.

2 Preliminaries

As noted in the introduction, the algorithms we propose proceed in epochs, wherein the decision-maker deploys the current decision for a fixed number of steps (i.e., an epoch) allowing the dynamically evolving distribution to *mix*. For such an approach to work, it needs to be the case that when the same decision is deployed repeatedly, the distribution converges to a fixed point. Further, convexity (in the decision variable) of the expected loss function given the fixed point distribution ensures the optimal solution is unique and obtainable via gradient-based learning. This motivates the assumptions we introduce in this section.

We consider the problem of a single decision-maker facing a decision dependent learning problem in a geometrically decaying environment. The decision-maker seeks to minimize its expected loss at the fixed point distribution which is given by

$$\mathcal{L}(x) = \mathbb{E}_{z\sim\mathcal{D}(x)}[\ell(z, x)].$$

The loss function is denoted $\ell_i : \mathbb{R}^d \times \mathcal{Z} \to \mathbb{R}$, where $\mathcal{Z}$ is some q finite dimensional metric space, and $\mathcal{D}(x) \in \mathcal{P}(\mathcal{Z})$ is a probability measure that depends on the decision $x \in \mathcal{X}$ where $\mathcal{X} \subset \mathbb{R}^d$ is closed and convex, and there exists constants $r, R > 0$ satisfying $r\mathbb{B} \subseteq \mathcal{X} \subseteq R\mathbb{B}$.

As noted, the environment is evolving in time according to a geometrically decaying process. That is, the random variable z depends not only on the decision $x_t \in \mathcal{X}$ at time t, but also explicitly on the time instant t. In particular, the random variable z is governed by the distribution p_t which is the probability measure at time t generated by the process $p_{t+1} = \mathcal{T}(p_t, x_t)$ where

$$\mathcal{T}(p, x) = \mu p + (1 - \mu)\mathcal{D}(x), \tag{1}$$

and $\mu \in [0, 1)$ is the geometric decay rate. Observe that given the geometrically decaying dynamics in (1), for any $x \in \mathcal{X}$, $\mathcal{D}(x)$ is trivially a fixed point—i.e., $\mathcal{T}(\mathcal{D}(x), x) = \mathcal{D}(x)$. Moreover, for any $\mu \in (0, 1)$ and fixed $x \in \mathcal{X}$, p_t converges at an exponential rate to $\mathcal{D}(x)$.

One interpretation of this transition map is that it captures the phenomenon that for each time, a $1 - \mu$ fraction of the population becomes aware of the machine learning model x being used by the institution. Another interpretation is that the environment (and strategic data sources in the environment) has memory based on past interactions which is captured in the 'state' of the distribution, and the effects of the past decay geometrically at a rate of μ.

Performative optimality is defined as follows.

Definition 1 (Performatively optimal point)**.** *For a given distribution $\mathcal{D}(x)$, the decision vector $x^* \in \mathcal{X}$ is a performatively optimal point if*

$$x^* \in \arg\min_{x\in\mathcal{X}} \mathcal{L}(x) = \arg\min_{x\in\mathcal{X}} \mathbb{E}_{z\sim\mathcal{D}(x)}[\ell(z, x)].$$

Throughout we use the notation $\nabla\mathcal{L}$ to denote the derivative differentiating through both the x dependence in the loss and the x dependence in the distribution $\mathcal{D}$. Moreover, the notation $\nabla_x\ell$ and $\nabla_z\ell$ denote the partial derivative of ℓ with respect to x and z, respectively.

We also make the following standing assumptions on the loss ℓ and distribution $\mathcal{D}$.

Assumption 1 (Standing)**.** *The loss ℓ and distribution $\mathcal{D}$ satisfy the following:*

1. *The loss $\ell(z, x)$ is C^1 smooth in x, and L-Lipschitz in (x, z).*
2. *The map $x \mapsto \nabla_x\ell(z, x)$ is β-Lipschitz continuous in z.*
3. *The loss $\ell(z, x)$ is ξ-strongly convex in x.*

We assume that the distribution $\mathcal{D}(x)$ is γ-Lipschitz with respect to the Wasserstein-1 distance denoted by W_1.

Assumption 2. *There exists an $\gamma > 0$ such that for any $x, x' \in \mathcal{X}$, $\mathcal{W}_1(\mathcal{D}(x), \mathcal{D}(x')) \le \gamma\|x - x'\|$.*

The following assumption implies a convex ordering on the random variables on which the loss is dependent.

Assumption 3 (Mixture Dominance)**.** *The distribution map $\mathcal{D}$ and loss ℓ satisfy mixture dominance—i.e., for any $x \in \mathcal{X}$ and $s \in (0, 1)$, $\mathbb{E}_{z\sim\mathcal{D}(sx'+(1-s)x'')}[\ell(z, x)] \le \mathbb{E}_{z\sim s\mathcal{D}(x')+(1-s)\mathcal{D}(x'')}[\ell(z, x)]$, for all $x', x'' \in \mathcal{X}$.*

Under Assumptions 2, 1, and 3, the expected loss $\mathcal{L}(x)$ is $\alpha := (\xi - 2\gamma\beta)$ strongly convex (cf. Theorem 3.1 Miller, Perdomo, and Zrnic (2021)), so that the performatively optimal point is unique.

We make the following assumption on the regularity of the expected loss.

Assumption 4. *The map $x \mapsto \nabla L(x)$ is M-Lipschitz continuous.*

An important class of distributions in the performative prediction literature that satisfy this assumption are location-scale distribution.

Example 1 (Losses with Decision-Dependent Location-Scale Distributions are Smooth)**.** *We say $\mathcal{D}(\theta)$ belongs to the family of location-scale distributions if*

$$z \sim \mathcal{D}(x) \iff z = \Sigma_0 z_0 + a_0 + A^\top x,$$

where the location depends on x, the base random variable $z_0 \sim \mathcal{D}_0$ is a sample from a zero-mean distribution, and A is norm bounded. Then, under the Assumption 1.2, the expected risk $\mathcal{L}(x)$ is $\beta(\|A\|_{\text{op}}^2 + 2\|A\|_{\text{op}} + 1)$–smooth.

3 Algorithms & Sample Complexity Analysis

As alluded to in the introduction, the algorithms we propose for each of the information settings are similar in spirit—namely, they each operate by holding fixed a decision for the length n of each epoch and sampling or querying the

Algorithm 1: Epoch-Based Zeroth Order Algorithm

Initialization: epoch length n, step-size η, initial point x_1, query radius δ, horizon T, batch size m, initial distribution p_0;
for $t = 1, 2, \ldots, T$ **do**
  // Step 1: Mixing
  Sample vector v_t from the unit sphere;
  Run $x_t + \mu v_t$ for n steps, so
  $p_t = \mathcal{T}(\ldots \mathcal{T}(\mathcal{T}(p_{t-1}, x_t + \delta v_t), x_t + \delta v_t) \ldots, x_t + \delta v_t)$;
  // Step 2: Update
  Oracle reveals $\hat{g}_t = \frac{d}{\delta} \frac{1}{m} \sum_{k=1}^{m} \ell(z_k, x_t + \delta v_t) v_t$, $z_k \sim \mathcal{D}(x_t + \delta v_t)$;
  Update $x_{t+1} = \Pi_{(1-\delta)\mathcal{X}}(x_t - \eta \hat{g}_t)$;
end

Algorithm 2: Epoch-Based First Order Algorithm

Initialization: epoch length n, step-size η, initial point x_1, horizon T, batch size m, initial distribution p_0;
for $t = 1, 2, \ldots, T$ **do**
  // Step 1: Mixing
  Run x_t for n steps, so
  $p_t = \mathcal{T}(\ldots \mathcal{T}(\mathcal{T}(p_{t-1}, x_t), x_t) \ldots, x_t)$;
  // Step 2: Update
  Oracle reveals $\hat{g}_t = \frac{1}{m} \sum_{k=1}^{m} \nabla \ell(z_k, x_t)$, $z_k \sim \mathcal{D}(x_t)$;
  Update $x_{t+1} = \Pi_{\mathcal{X}}(x_t - \eta \hat{g}_t)$;
end

environment until the distribution dynamics have mixed sufficiently towards the fixed point distribution.

In practice, different information may be available to the decision maker. We consider two settings: zero order and first order oracle.

To characterize the sample complexity in both of these settings, we need to bound the error between the empirical gradient information and the expected gradient information. Towards this end, we need the following assumption that now explicitly depends on the algorithm structure—namely, that it proceeds in epochs of length n.

Assumption 5. *Given an epoch length m, for every $p_t = \mu^n p_{t-1} + (1-\mu^n)\mathcal{D}(x)$ induced by the initial distribution and the mapping $\mathcal{D}(\cdot)$, the loss $\ell(z, x)$ is G-Lipschitz continuous and H-Hessian Lipschitz continuous.*

The location-scale family of distributions satisfies Assumption 5. The following example shows what the value of G is and H can similarly be derived.

Example 2 (Location-Scale Distributions)**.** *We say $\mathcal{D}(x)$ belongs to the family of location-scale distributions if*

$$z \sim \mathcal{D}(x) \iff z = \Sigma_0 z_0 + a_0 + A^\top x,$$

where the location depends on x, $z_0 \sim \mathcal{D}_0$ is a sample from a zero mean distribution, and A is norm bounded. If p_0 is additionally a location-scale distribution, then p_t is a location-scale family for all t, such that $\ell(z, x)$ is G-Lipschitz with $G = \zeta((1-\mu^n)\|A\|_{\text{op}} + 1)$.

This class encompasses a broad set of distributions that are commonplace in the performative prediction literature. This class of distributions is also γ-sensitive and satisfies the mixture dominance condition when ℓ is convex, observations that appeared in Miller, Perdomo, and Zrnic (2021).

3.1 Zero Order Oracle

In this setting, the decision-maker has access only to the empirical loss at the current distribution p_t. This is a more realistic setting given that the form of the data distribution p_t—and more specifically, $\mathcal{D}(\cdot)$—may be a priori unknown. For example, if the data is generated by strategic data sources having their own private utility functions and preferences, then the decision-maker does not necessarily have access to the distribution map $\mathcal{D}(x)$ in practice.

A decision-maker using Algorithm 1 updates with gradient estimates

$$\hat{g}_t = \frac{d}{\delta} \frac{1}{m} \sum_{k=1}^{m} \ell(z_k, x_t + \delta v_t) v_t,$$

where v_t is a unit vector and $z_k \sim p_t = \mu^n p_{t-1} + (1-\mu)^n \mathcal{D}(x_t + \delta v_t)$ is the k-th sample from p_t. This is a one-point gradient estimate of the expected loss at p_t (cf. Flaxman, Kalai, and McMahan (2004)). As is well known, for a given function $f : \mathbb{R}^d \to \mathbb{R}$ and query radius $\delta > 0$,

$$\mathbb{E}_{v \sim \mathbb{S}}[f(x + \delta v) v] = \frac{\delta}{d} \nabla \hat{f}(x),$$

where $\hat{f}(x) = \mathbb{E}_{v \sim \mathbb{B}}[f(x + \delta v)]$ and $\mathbb{B}$ and $\mathbb{S}$ are the Euclidean unit ball and sphere, respectively, in dimension d.

In each iteration t, the zero-order algorithm samples $v_t \sim \mathbb{S}$ uniformly at random and then declares

$$x_{t+1} = \Pi_{(1-\delta)\mathcal{X}}\left(x_t - \eta \hat{g}_t\right).$$

The reason for projecting onto the set $(1-\delta)\mathcal{X}$ is to ensure that in the next iteration, the decision is in the feasible set.

In the zero order setting, we also need the loss to be uniformly bounded.

Assumption 6. *The quantity $\ell_* := \sup\{|\ell(z, x)| : x \in \mathcal{X},\ z \in \mathcal{Z}\}$ is finite.*

In order to analyze Algorithm 1, we bound the difference in the expected gradients at $z \sim p_t = \mu^n p_{t-1} + (1-\mu^n)\mathcal{D}(x_t + \delta v_t)$ and $z \sim \mathcal{D}(x_t + \delta v)$ (Lemma 1), and we bound the difference between the empirical gradient at p_t of the smoothed loss and the gradient of the smoothed loss at p_t (Lemma 5, Appendix A). Define the smoothed expected risk as follows:

$$\mathcal{L}^\delta(x) = \mathbb{E}_{v \sim \mathbb{B}}[\mathbb{E}_{z(x+\delta v) \sim \mathcal{D}(x+\delta v)}[\ell(z(x + \delta v), x + \delta v)]].$$

Lemma 1. *Under Assumptions 1–3 and 6, the error between the gradient smoothed loss at p_t and the gradient of the smoothed expected risk at the fixed point $\mathcal{D}$ satisfies*

$$\|\nabla \mathbb{E}_{v \sim \mathbb{B}}[\mathbb{E}_{z \sim p_t}[\ell(z, x_t + \delta v)]] - \nabla \mathcal{L}^\delta(x_t)\| \leq L \cdot \left(\mu^n \overline{W}(p_0) + 2\mu^n \delta \gamma + \ell_* \frac{\gamma \eta}{\mu} \frac{\mu^n}{1-\mu^n}\right)$$

where $p_t = \mu^n p_{t-1} + (1-\mu^n)\mathcal{D}(x_t + \delta v_t)$*, and* $\overline{W}(d_0) = \max_{x\in\mathcal{X}} W_1(p_0, \mathcal{D}(x))$.

We defer the proof to Appendix B.1. We apply Lemma 5 to get a bound on the error between the empirical gradient of the smoothed loss, which is given by

$$\mathcal{L}_t^{\delta,(m)}(x_t) = \frac{1}{m}\sum_{k=1}^{m} \mathop{\mathbb{E}}_{v\sim\mathbb{B}}[\ell(z, x_t + \delta v)],$$

and the gradient of the smoothed loss

$$\mathcal{L}_t^{\delta}(x_t) = \mathop{\mathbb{E}}_{v\sim\mathbb{B}, z\sim p_t}[\ell(z, x_t + \delta v)]$$

at the current distribution p_t.

Under Assumption 5, we have that with probability $1-\rho$, the error $\|\nabla\mathcal{L}_t^{\delta}(x) - \nabla\mathcal{L}_t^{\delta,(m)}(x)\|$ uniformly converges—i.e.,

$$\sup_{x\in\mathcal{X}} \|\nabla\mathcal{L}_t^{\delta}(x) - \nabla\mathcal{L}_t^{\delta,(m)}(x)\| \le \frac{G}{2}\sqrt{\frac{Cd\log m}{m}} \tag{2}$$

for $m \ge \Omega(d\log d)$ and where C depends on ρ as detailed in the following theorem which characterizes the sample complexity of Algorithm 1. This follows from a small modification of Theorem 1 in Mei, Bai, and Montanari (2016) to apply to norm-subGaussian random vectors, so we leave it to the appendix. For a fixed time t, let $\mathcal{E}_t$ be the event that (2) for all $s \in \{0, \dots, t\}$.

Theorem 1. *Suppose that Assumptions 1-6 hold. There exists a universal constant* C_0 *such that with probability* $1-\rho$*, Algorithm 1 with epoch length* $n \ge \Omega(\log\eta)$*, step-size* $\eta = \eta_0/t$ *with* $\eta_0 > 1/(3\alpha)$*, and query radius* $\delta = (d^2\ell_*^2\eta/(2MR))^{1/3}$*, returns iterates satisfying*

$$\mathbb{E}[\tfrac{1}{2}\|x_t - x^*\|^2 | \mathcal{E}_t] \le O\left(d^2 t^{-\frac{1}{3}} + \sqrt{\log\left(\frac{1}{\rho}\right)\frac{d\log m}{m}}\right)$$

for $m \ge Cd\log d$*,* $C = C_0\max\{c_h, \log(RG/\rho), 1\}$ *and* $c_h = \frac{1}{\log d}\max\left\{\log\left(\frac{4M}{G^2}\right), \log\left(\frac{8H}{G^3}\right)\right\}$.

The proof is deferred to Appendix B.2. Unlike the standard constrained zero order setting for which it has been shown that the approach of Flaxman, Kalai, and McMahan (2004) obtains a $O(d^2 t^{-1/3})$ rate (Agarwal, Dekel, and Xiao 2010), we have an additional error term due to the environment dynamics that results in having to bound the error between the empirical and expected gradients at p_t to their counterparts at $\mathcal{D}(x)$—i.e., the distribution which defines the performatively optimal point (cf. Definition 1). In expectation with respect to p_t, the rate is $O(d^2 t^{-1/3})$.

3.2 First Order Oracle

In this setting, we assume access to an oracle that reveals the empirical gradient of the performative risk at the current distribution—i.e.,

$$\hat{g}_t = \frac{1}{m}\sum_{k=1}^{m} \nabla\ell(z_k, x_t)$$

where z_k is the k-th sample from p_t. To analyze the performance of Algorithm 2, there are two crucial steps: (1) Obtain a bound on the difference between the gradient of the expected performative risk at the current distribution p_t and the gradient of the expected risk at $\mathcal{D}(x_t)$ (Lemma 2); (2) Obtain a bound on the difference between the empirical gradient and the expected gradient (Lemma 5, Appendix A).

Lemma 2. *Under Assumptions 1–3, the gradient error satisfies*

$$\begin{aligned}&\|\nabla\mathbb{E}_{z_t\sim p_t}[\ell(z_t, x_t)] - \nabla\mathbb{E}_{z_t\sim\mathcal{D}(x_t)}[\ell(z_t, x_t)]\| \\ &\quad\le \mu^{tn}\beta W_1(p_0, \mathcal{D}(x_1)) + \epsilon L(1+\gamma)\frac{\beta\eta\mu^n}{1-\mu^n} + \mu^n L\gamma,\end{aligned}$$

where $p_t = \mu^n p_{t-1} + (1-\mu^n)\mathcal{D}(x_t)$.

We defer the proof to Appendix C.1. The first term $\mu^{tn}\beta\mathcal{W}_1(p_0, \mathcal{D}(x_1))$ captures how the dependence on the initial decision x_1 in the gradient error exponentially decays to zero as t approaches infinity. The second term $\beta\eta\gamma L(1+\gamma)\mu^n/(1-\mu^n)$ captures the difference between subsequent decisions of Algorithm 2. The third term $\mu^n L\gamma$ captures the effect of the decision on the loss function. Observe that as the epoch length n becomes large, the bound approaches zero, which is equivalent to the static setting—i.e., there is no past state dependence (equivalently, $\mu = 0$), and hence, the gradient error is zero.

To bound the difference between the empirical gradient and the expected gradient we apply Lemma 5 (Appendix A) to the empirical loss

$$\mathcal{L}_t^{(m)}(x_t) = \frac{1}{m}\sum_{k=1}^{m} \ell(z_k, x_t)$$

and the expected loss

$$\mathcal{L}_t(x_t) = \mathop{\mathbb{E}}_{z\sim p_t}[\ell(z, x_t)].$$

Indeed, Assumptions 4 and 5 are enough so that the assumptions of Lemma 5 hold. Hence, with probability $1-\rho$, we have that

$$\sup_{x\in\mathcal{X}} \|\nabla\mathcal{L}_t^{(m)}(x) - \nabla\mathcal{L}_t(x)\| \le \frac{G}{2}\left(\frac{Cd\log m}{m}\right)^{\frac{1}{2}} \tag{3}$$

for $m \ge \Omega(d\log d)$ and C depends on ρ as detailed in the following theorem which characterizes the sample complexity of Algorithm 2 given the two gradient error bounds.

Theorem 2. *Suppose that Assumptions 1–5 hold. Fix arbitrary* $\epsilon > 0$ *and let* $\eta = 1/M$ *and* $m \ge \log(\epsilon\alpha/\tilde{C})/\log(\mu)$ *where* $\tilde{C} = \beta W_1(\mathcal{D}(x_1), d_0) + \gamma\beta L(1+\gamma)/M + \gamma L + \epsilon\alpha$*. There exists a universal constant* C_0 *such that with probability* $1-\rho$*, Algorithm 2 returns iterates that satisfy*

$$\begin{aligned}\|x_{t+1} - x^*\| \le \left(1 - \frac{\alpha}{M}\right)^t \|x_1 - x^*\| + \epsilon \\ + \left(\log\left(\frac{1}{\rho}\right)\frac{d\log m}{m}\right)^{\frac{1}{2}}\end{aligned}$$

for $m \ge Cd\log d$ *where* $C = C_0\max\{c_h, \log(RG/\rho), 1\}$ *with* $c_h = \frac{1}{\log d}\max\left\{\log\left(\frac{4M}{G^2}\right), \log\left(\frac{8H}{G^3}\right)\right\}$.

We defer the proof to Appendix C.2. The idea is to massage the problem into stochastic gradient descent with biased gradients where there are two sources of bias. While the size of the error $\epsilon > 0$ can be arbitrarily chosen, the lower bound on the number of samples m depends logarithmically on the choice of ϵ and hence, there is a tradeoff between the error tolerance and epoch size. If access to an expected gradient oracle is available, then the convergence rate is analogous to Ajalloeian and Stich (2020, Theorem 6) since the batch-size n dependent term drops out.

4 Numerical Experiments

In this section, we apply our aforementioned algorithms to a semi-synthetic example based on real data from the dynamic pricing experiment—namely, SFpark[1]—for on-street parking in San Francisco. Parking availability, location, and price are some of the most important factors when people choose whether or not to use a personal vehicle to make a trip (Shoup 2006, 2021; Fiez and Ratliff 2020). The primary goal of the SFpark pilot project was to make it easy to find a parking space. To this end, SFpark targeted a range of 60–80% occupancy in order to ensure some availability at any given time, and devised a controlled experiment for demand responsive pricing. Meter operational hours are split into distinct rate periods, and rates are adjusted on a block-by-block basis, using occupancy data from parking sensors in on-street parking spaces in the pilot areas. We focus on weekdays in the numerical experiments; for weekdays, distinct rate periods are 900–1200, 1200–1500, and 1500–1800. Excluding special events, SFpark adjusted hourly rates as follows: a) 80–100% occupancy, rates are increased by \$0.25; b) $60-80\%$ occupancy, no adjustment is made; c) $30-60\%$ occupancy, rate is decreased by \$0.25; d) occupancy below 30%, rate is decreased by \$0.50. When a price change is deployed it takes time for users to become aware of the price change through signage and mobile payment apps (Pierce and Shoup 2013).

Given the target occupancy, the dynamic decision-dependent loss (or performative risk) is given by

$$\mathbb{E}_{z\sim p_t}[\ell(z,x)] = \mathbb{E}_{z\sim p_t}[(z-0.7)^2 + \tfrac{\nu}{2}x^2],$$

for each block, where z is the occupancy (which is between zero and one), x is the change in price from the nominal price at the beginning of the SFpark study, and ν is the regularization parameter. For the initial distribution p_0, we sample from the data at the beginning of the pilot study where the price is at the nominal (or initial) price. The distribution $\mathcal{D}(x)$ is defined as follows:

$$z \sim \mathcal{D}(x) \iff z = \zeta_0 + ax$$

where ζ_0 follows the same distribution as p_0 described above, and a is a proxy for the price elasticity which is estimated by fitting a line to the final and initial occupancy and price (cf. Appendix D.1).[2]

[1] SFpark: `tinyurl.com/dwtf7wwn`

[2] Price elasticity is the change in percentage occupancy for a given percentage change in price.

Comparing Performative Optimum to SFpark. We run Algorithms 2 and 1 (using parameters as dictated by Theorems 2 and 1, respectively) for Beach ST 600, a representative block in the Fisherman's Wharf sub-area, in the time window of 1200–1500 as depicted in Figure 1. Beach ST is frequently visited by tourists and local residents. For Beach ST 600, we compute $a \approx -0.157$, which means that a \$1.00 increase in the parking rate will lead to a 15% decrease in parking occupancy at the fixed point distributions. Additionally, we use the data to compute the geometric decay rate of $\delta \approx 0.959$ (computations described in Appendix D). Since the initial price is \$3 per hour for this block, we take $\mathcal{X} = [-3, 5]$, since the maximum price that SFpark charges is \$8 per hour, and the minimum price is zero dollars. Additionally, we set $\nu =$ 1e-3.

The first and third plots in Figure 1 show prices announced and corresponding occupancy, respectively, for Algorithm 2, on 600 Beach Street, with different choices of n and T; and, they show the prices announced and corresponding occupanices by SFpark as compared to the performatively optimal point (computed offline). Similarly, the second and fourth plots in Figure 1 show this same information for Algorithm 1. Since Algorithm 1 is zero order, convergence requires more time and has variance coming from the randomness of the query directions.

The SFpark experiment changed prices approximately every eight weeks. As observed in Figure 1, this choice of n is reasonable—the estimated δ value is close to one—and leads to convergence to the performatively optimal point for both the first order and zero order (more realistic for this application) algorithms. As n increases, the performance degrades. This observation holds for this street generally; however, in our experimentation we found that other frequencies (i.e. epoch length) were optimal suggesting that a non-uniform (across blocks) price update schedule may lead to better outcomes. Appendix D.2 contains additional experiments.

Moreover, the prices under the performatively optimal solution obtained by the proposed algorithms are lower than the SFpark solution for the entire trajectory, and the algorithms both reach the target occupancy while SFpark is far from it. The third and fourth plots of Figure 1 show the effect of the negative price elasticity on the occupancy; an increased price causes a decreased occupancy. An interesting observation is that for Algorithm 2, a larger choice of n, and consequently a smaller choice of T, allows for convergence closer to the performatively optimal price, but for Algorithm 1, a smaller choice of n, and consequently, a larger choice of T, allows for quicker (and with lower variance) convergence to the performatively optimal price. This is due to the randomness in the query direction for the gradient estimator used in Algorithm 1, meaning that a larger T is needed to converge quickly to the optimal solution. This suggests that in the more realistic case of zero order feedback, the institution should make more price announcements.

Redistributing Parking Demand. In this semi-synthetic experiment, we set $\nu =$ 1e-3 and take $\mathcal{X} = [-3.5, 4.5]$ since the base distribution for these blocks has a nominal price of \$3.50. We also use the estimated δ and m values

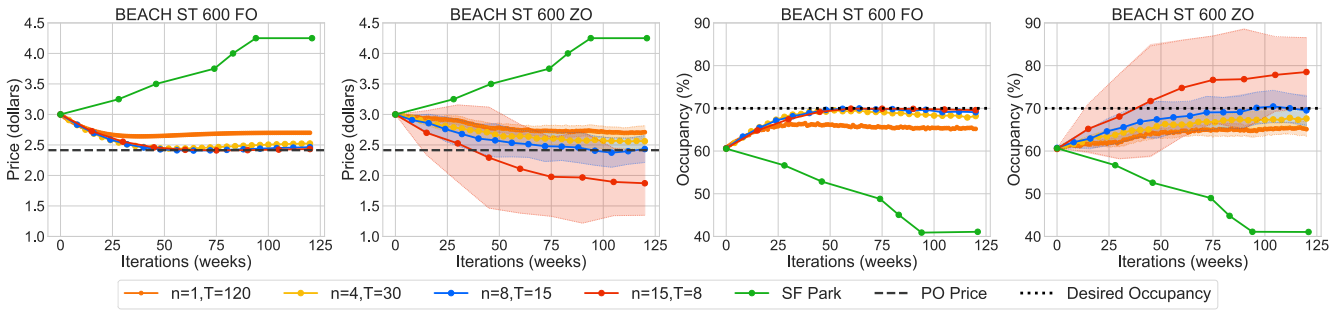

Figure 1: Results of Algorithm 2 (first and third plots) and Algorithm 1 (second and fourth plots) with different (n, T) pairs for 600 Beach ST and time window 1200–1500. Each marker represents a price announcement, and the plots show the prices and corresponding predicted occupancies. The SFpark prices and occupancies are far from the target and performative optimal price, whereas the proposed algorithms obtain both points up to theoretical error bounds.

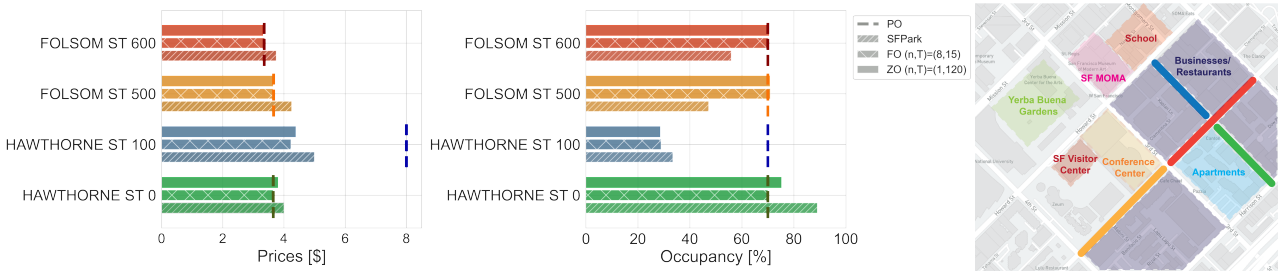

Figure 2: Final prices announced by first and zero order algorithms (Algorithms 2 and 1) run with $(n, T) = (8, 15)$ and $(n, T) = (1, 120)$, respectively, as compared to SFpark for streets depicted in the right graphic (color coded to the bar charts) during the 900–1200 time period. The center plot shows the corresponding predicted occupancies. The dotted lines represent performatively optimal price and target occupancy of 70%, in the left and center plots, respectively. The average price overall is lower for both proposed methods, the occupancy is better distributed, and the average occupancy closer to the desired range.

(described in more detail in Appendix D.3). We run Algorithms 2 and 1 (using parameters as dictated by the corresponding sample complexity theorems) for a collection of blocks during the time period 900–1200 in a highly mixed use area (i.e., with tourist attractions, a residential building, restaurants and other businesses). The results are depicted in Figure 2.

Hawthorne ST 0 is a very high demand street; the occupancy is around 90% on average during the initial distribution and remains high for SFpark (cf. center, Figure 2). The performatively optimal point, on the other hand, reduces this occupancy to within the target range 60–80% for both the first and zeroth order methods. This occupancy can be seen as being redistributed to the Folsom ST 500-600 block, as depicted in Figure 2 (center) for our proposed methods: the SFpark occupancy is much below the 70% target average for these blocks, while both the decision-dependent algorithms lead to occupancy at the target average. Interestingly, this also comes at a lower price (not just on average, but for each block) than SFpark.

Hawthorne ST 100 is an interesting case in which both our approach and SFpark do not perform well. This is because the performatively optimal price in the *unconstrained case* is \$9.50 an hour which is well above the maximum price of \$8 in the constrained setting we consider. In addition, the price elasticity is positive for this block; together these facts explain the low occupancy. Potentially other control knobs available to SFpark, such as time limits, can be used in conjunction with price to manage occupancy; this is an interesting direction of future work.

5 Discussion and Future Directions

This work is an important step in understanding performative prediction in dynamic environments. Moving forward there are a number of interesting future directions. We consider one class of well-motivated dynamics. Another practically motivated class of dynamics are period dynamics; indeed, in many applications there is an external context which evolves periodically such as seasonality or other temporal effects. Devising algorithms for such cases is an interesting direction of future work. As compared to classical reinforcement learning problems, in this work, we exploit the structure of the dynamics along with convexity to devise convergent algorithms. However, we only considered general conditions on the class of distributions $\mathcal{D}(x)$; it may be possible to exploit additional structure on $\mathcal{D}(x)$ in improving the sample complexity of the proposed algorithms or devising more appropriate algorithms that leverage this structure.

References

Agarwal, A.; Dekel, O.; and Xiao, L. 2010. Optimal Algorithms for Online Convex Optimization with Multi-Point Bandit Feedback. In *Conference on Learning Theory*, 28–40. Citeseer.

Ajalloeian, A.; and Stich, S. U. 2020. Analysis of SGD with biased gradient estimators. *arXiv preprint arXiv:2008.00051*.

Anagnostopoulos, A.; Castillo, C.; Fazzone, A.; Leonardi, S.; and Terzi, E. 2018. Algorithms for hiring and outsourcing in the online labor market. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1109–1118.

Banerjee, S.; Riquelme, C.; and Johari, R. 2015. Pricing in ride-share platforms: A queueing-theoretic approach. *Available at SSRN 2568258*.

Bravo, M.; Leslie, D. S.; and Mertikopoulos, P. 2018. Bandit learning in concave N-person games. In *Advances in Neural Information Processing Systems*.

Brown, G.; Hod, S.; and Kalemaj, I. 2020. Performative Prediction in a Stateful World. In *Advances in Neural Information Processing Systems*.

Dowling, C.; Fiez, T.; Ratliff, L.; and Zhang, B. 2017. Optimizing curbside parking resources subject to congestion constraints. In *Proceedings of the IEEE 56th Annual Conference on Decision and Control (CDC)*, 5080–5085.

Dowling, C. P.; Ratliff, L. J.; and Zhang, B. 2020. Modeling Curbside Parking as a Network of Finite Capacity Queues. *IEEE Transactions on Intelligent Transportation Systems*, 21(3): 1011–1022.

Drusvyatskiy, D.; and Xiao, L. 2020. Stochastic optimization with decision-dependent distributions. *arXiv preprint arXiv:2011.11173*.

Fiez, T.; and Ratliff, L. J. 2020. Gaussian Mixture Models for Parking Demand Data. *IEEE Transactions on Intelligent Transportation Systems*, 21(8): 3571–3580.

Fiez, T.; Ratliff, L. J.; Dowling, C.; and Zhang, B. 2018. Data driven spatio-temporal modeling of parking demand. In *2018 Annual American Control Conference (ACC)*, 2757–2762. IEEE.

Flaxman, A. D.; Kalai, A. T.; and McMahan, H. B. 2004. Online convex optimization in the bandit setting: gradient descent without a gradient. *arXiv preprint cs/0408007*.

Goel, V.; and Grossmann, I. E. 2004. A stochastic programming approach to planning of offshore gas field developments under uncertainty in reserves. *Computers & chemical engineering*, 28(8): 1409–1429.

Horton, J. J. 2010. Online labor markets. In *International workshop on internet and network economics*, 515–522. Springer.

Jin, C.; Netrapalli, P.; Ge, R.; Kakade, S. M.; and Jordan, M. I. 2019. A short note on concentration inequalities for random vectors with subgaussian norm. *arXiv preprint arXiv:1902.03736*.

Jonsbråten, T. W.; Wets, R. J.; and Woodruff, D. L. 1998. A class of stochastic programs withdecision dependent random elements. *Annals of Operations Research*, 82: 83–106.

Lum, K.; and Isaac, W. 2016. To predict and serve? *Significance*, 13(5): 14–19.

Mei, S.; Bai, Y.; and Montanari, A. 2016. The landscape of empirical risk for non-convex losses. *arXiv preprint arXiv:1607.06534*.

Mendler-Dünner, C.; Perdomo, J.; Zrnic, T.; and Hardt, M. 2020. Stochastic Optimization for Performative Prediction. *Advances in Neural Information Processing Systems*, 33.

Miller, J.; Perdomo, J. C.; and Zrnic, T. 2021. Outside the Echo Chamber: Optimizing the Performative Risk. *arXiv preprint arXiv:2102.08570*.

Perdomo, J.; Zrnic, T.; Mendler-Dünner, C.; and Hardt, M. 2020. Performative Prediction. In III, H. D.; and Singh, A., eds., *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, 7599–7609. PMLR.

Pierce, G.; and Shoup, D. 2013. Getting the prices right: an evaluation of pricing parking by demand in San Francisco. *Journal of the american planning association*, 79(1): 67–81.

Pierce, G.; and Shoup, D. 2018. *Sfpark: Pricing parking by demand*. Routledge.

Schmidt, M.; Roux, N. L.; and Bach, F. 2011. Convergence rates of inexact proximal-gradient methods for convex optimization. In *Advances in Neural Information Processing Systems*.

Shoup, D. C. 2006. Cruising for parking. *Transport policy*, 13(6): 479–486.

Shoup, D. C. 2021. *The high cost of free parking*. Routledge.

Sutton, R. S.; and Barto, A. G. 2018. *Reinforcement learning: An introduction*. MIT press.

Varaiya, P.; and Wets, R.-B. 1988. Stochastic dynamic optimization approaches and computation.

Vershynin, R. 2018. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press.