Decision-Dependent Risk Minimization in Geometrically Decaying Dynamic Environments

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

University of Washington, Seattle
{mitasray, ratliffl, ddrusv, mfazel}@uw.edu

Abstract

This paper studies 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. Novel algorithms for both the information setting in which the decision-maker has a first order gradient oracle and the setting in which they have simply a loss function oracle are are introduced. The algorithms operate on the same underlying principle: the decision-maker repeatedly deploys a fixed decision repeatedly over the length of an epoch, thereby allowing the dynamically changing environment to sufficiently mix before updating the decision. The iteration complexity in each of the settings is shown to match existing rates for first and zero order stochastic gradient methods up to logarithmic factors. 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.

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, 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 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 repeatedly for the duration of an epoch, thereby 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 environment 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 curb space utilization.

Contributions

For the decision-dependent learning problem in geometrically decaying environments, we propose first-order or zeroth-order oracle algorithms that converge to the optimal point under appropriate assumptions, which make the risk minimization problem strongly convex. We obtain the following iteration complexity guarantees:

- **Zero Order Oracle** (Algorithm 1, Section): We show that the sample complexity in the zeroth order setting is $\tilde{O}(\frac{d^2}{\varepsilon^2})$ which matches the optimal rate for single query zeroth order methods in strongly convex settings up to logarithmic factors.
- **First Order Oracle** (Algorithm 2, Section): We show that the same complexity in the first order setting is $\tilde{O}(\frac{1}{\varepsilon})$ again matching the known rates for first order stochastic gradient methods up to logarithmic factors.

The technical novelty arises from bounding the error between expected gradient at the fixed point distribution corresponding to the current decision and the stochastic gradient at the current distribution at time t.

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). 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 is contained in Appendix D, including purely synthetic simulations and other semi-synthetic simulations using the 'Give Me Some Credit' data set (Kaggle 2011).

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) 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. A naïve strategy is to re-train the model after using heuristics to determine when there is sufficient distribution shift. 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 distribution 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.

Preliminaries

We consider the problem of a single decision-maker facing a decision dependent learning problem in a geometrically decaying environment.

Towards formally defining the optimization problem the decision-maker faces, we first introduce some notation. Throughout, we let $\mathbb{R}^d$ denote a d–dimensional Euclidean space with inner product $\langle \cdot, \cdot \rangle$ and induced norm $\|x\| = \sqrt{\langle x, x \rangle}$. The projection of a point $y \in \mathbb{R}^d$ onto a set $\mathcal{X} \subset \mathbb{R}^d$ is denoted $\mathrm{proj}_{\mathcal{X}}(y) = \mathrm{argmin}_{x \in \mathcal{X}} \|x - y\|$. We are in interested in random variables taking values in a metric space. Given a metric space $\mathcal{Z}$ with metric $\mathrm{d}(\cdot, \cdot)$ the symbol $\mathbb{P}(\mathcal{Z})$ denotes the set of Radon probability measures ν on $\mathcal{Z}$ with a finite first moment $\mathbb{E}_{z \sim \nu}[\mathrm{d}(z, z')] < \infty$ for some $z' \in \mathcal{Z}$. We measure the deviation between two measures $\nu, \nu' \in \mathbb{P}(\mathcal{Z})$ using the Wasserstein-1 distance:

$$W_1(\nu, \mu) = \sup_{h \in \mathrm{Lip}_1} \{\mathbb{E}_{X \sim \nu}[h(X)] - \mathbb{E}_{Y \sim \mu}[h(Y)]\},$$

where Lip_1 denotes the set of 1–Lipschitz continuous functions $h : \mathcal{Z} \to \mathbb{R}$.

The decision-maker seeks to solve

$$\min_{x \in \mathcal{X}} \mathcal{L}(x) \tag{1}$$

where $\mathcal{L}(x) = \mathbb{E}_{z \sim \mathcal{D}(x)}[\ell(x, z)]$ is the expected loss. The decision-space $\mathcal{X}$ lies in the Euclidean space $\mathbb{R}^d$, is closed and convex, and there exists constants $r, R > 0$ satisfying $r\mathbb{B} \subseteq \mathcal{X} \subseteq R\mathbb{B}$ where $\mathbb{B}$ is the unit ball in dimension d. The loss function is denoted $\ell : \mathbb{R}^d \times \mathcal{Z} \to \mathbb{R}$, and $\mathcal{D}(x) \in \mathbb{P}(\mathcal{Z})$ is a probability measure that depends on the decision $x \in \mathcal{X}$.

Definition 1. *For a given probability measure $\mathcal{D}(x)$ induced by action $x \in \mathcal{X}$, the decision vector $x^* \in \mathcal{X}$ is optimal if*

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

The main challenge to finding an optimal point is that 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

$$\mathcal{T}(p, x) = \lambda p + (1 - \lambda)\mathcal{D}(x), \tag{2}$$

and $\lambda \in [0, 1)$ is the geometric decay rate. Observe that given the geometrically decaying dynamics in (2), for any $x \in \mathcal{X}$, the distribution $\mathcal{D}(x)$ is trivially a fixed point—i.e., $\mathcal{T}(\mathcal{D}(x), x) = \mathcal{D}(x)$. Let $\mathcal{T}^n := \mathcal{T} \circ \cdots \circ \mathcal{T}$ denote the n-times composition of the map $\mathcal{T}$ so that, given the form in (2), we have $\mathcal{T}^n(p, x) = \lambda^n p + (1 - \lambda^n)\mathcal{D}(x)$.

One interpretation of this transition map is that it captures the phenomenon that for each time, a $(1 - \lambda)$ 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 λ. For instance, it is known in behavioral economics that humans often compare their decisions to a reference point, and that reference point may evolve in time and represent an accumulation of past outcomes (Nar, Ratliff, and Sastry 2017; Kahneman and Tversky 2013).

Throughout we use the notation $\nabla \mathcal{L}$ to denote the derivative of $\mathcal{L}$ with respect to x. The notation $\nabla_x \ell$ and $\nabla_z \ell$ denotes the partial derivative of ℓ with respect to x and z, respectively. Further, let $\nabla_{x,z} \ell = (\nabla_x \ell, \nabla_z \ell)$ denote the vector of partial derivatives. We also make the following standing assumptions on the loss ℓ and the probability measure $\mathcal{D}(x)$.

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

a. The loss $\ell(x, z)$ is C^1 smooth in x, and L-Lipschitz continuous in (x, z).
b. The map $(x, z) \mapsto \nabla_{x,z} \ell(x, z)$ is β-Lipschitz continuous.
c. The loss $\ell(x, z)$ is ξ-strongly convex in x.
d. There exists a constant $\gamma > 0$ such that

$$W_1(\mathcal{D}(x), \mathcal{D}(x')) \leq \gamma \|x - x'\| \quad \forall\, x, x' \in \mathcal{X}.$$

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

Assumption 2 (Mixture Dominance)**.** *The probability measure $\mathcal{D}(x)$ and loss ℓ satisfy mixture dominance—i.e., for any $x \in \mathcal{X}$ and $s \in (0, 1)$, $\mathbb{E}_{z \sim \mathcal{D}(sv + (1-s)w)}[\ell(z, x)] \leq \mathbb{E}_{z \sim s\mathcal{D}(v) + (1-s)\mathcal{D}(w)}[\ell(z, x)]$, for all $v, w \in \mathcal{X}$.*

Under Assumptions 1 and 2, 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 optimal point is unique.

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

Assumption 3 (Smoothness)**.** *The map $x \mapsto \nabla \mathcal{L}(x)$ is G-Lipschitz continuous, and the map $x \mapsto \nabla^2 \mathcal{L}(x)$ is H-Lipschitz continuous.*

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

Assumption 4 (Parametric family)**.** *There exists a probability measure $\mathcal{P}$ and matrix A such that*

$$z \sim \mathcal{D}(x) \iff z = \zeta + Ax,$$

and where ζ has mean $\mu := \mathbb{E}_{\zeta \sim \mathcal{P}}[\zeta]$ and co-variance $\Sigma := \mathbb{E}_{\zeta \sim \mathcal{P}}[(\zeta - \mu)(\zeta - \mu)^\top]$, respectively.

This class encompasses a broad set of distributions that are commonplace in the performative prediction literature. As observed in Miller, Perdomo, and Zrnic (2021), this class of probability measures is also γ-Lipschitz continuous and satisfies the mixture dominance condition when ℓ is convex.

Algorithm 1: Epoch-Based Zeroth Order Algorithm

Initialization: epoch length n_t, step-size $\eta_t = \frac{4}{t\alpha}$, initial point x_1, query radius δ, horizon T, initial distribution p_0;
for $t = 1, 2, \ldots, T$ **do**
  // Step 1: Query-Mix
  Sample vector v_t from the unit sphere;
  Query with $x_t + \delta v_t$ for n_t steps, so that $p_t = \mathcal{T}^{n_t}(p_{t-1}, x_t + \delta v_t)$;
  // Step 2: Update
  Oracle reveals $\hat{g}_t = \frac{d}{\delta}\ell(z, x_t + \delta v_t)v_t$, $z \sim p_t$;
  Update $x_{t+1} = \mathrm{proj}_{(1-\delta)\mathcal{X}}(x_t - \eta_t \hat{g}_t)$;
end

Algorithm 2: Epoch-Based First Order Algorithm

Initialization: epoch length n_t, step-size η_t, initial point x_1, horizon T, initial distribution p_0;
for $t = 1, 2, \ldots, T$ **do**
  // Step 1: Query-Mix
  Query with x_t for n_t steps, so that $p_t = \mathcal{T}^{n_t}(p_{t-1}, x_t)$;
  // Step 2: Update
  Oracle reveals $\hat{g}_t = \nabla\ell(x_t, z)$, $z \sim p_t$;
  Update $x_{t+1} = \mathrm{proj}_{\mathcal{X}}(x_t - \eta_t \hat{g}_t)$;
end

Lemma 1 (Sufficient conditions for Assumption 3)**.** *Suppose that Assumption 4 holds and there exists a constants $\beta, \rho \geq 0$ such that the map $(x, z) \mapsto \nabla_{x,z}\ell(x, z)$ is β-Lipschitz continuous and has a ρ-Lipschitz continuous gradient. Then, Assumption 3 holds with constants*

$$G := \sqrt{\beta^2 \max\{1, \|A\|_{\mathrm{op}}^2\} \cdot (1 + \|A\|_{\mathrm{op}}^2)},$$

$$H := \sqrt{\rho^2 \max\{1, \|A\|_{\mathrm{op}}^4\} \cdot (1 + \|A\|_{\mathrm{op}}^2)}.$$

The proof is contained in Appendix A.

Algorithms & Sample Complexity Analysis

As alluded to in the introduction, the algorithms we propose for each of the information settings are similar in spirit: they each operate in epochs by holding fixed a decision for n steps and querying the environment until the distribution dynamics have mixed sufficiently towards the fixed point distribution corresponding to the current action.

Zero Order Stochastic Gradient Method

The most general information setting we consider is such that the decision-maker has only "bandit feedback". That is, they only have access to a loss function evaluation oracle. This does not require the decision-maker to have access to the decision-dependent probability measure $\mathcal{D}(x)$. This is a more realistic setting given that the form of $\mathcal{D}(\cdot)$—may be a priori unknown. For example, if the data is generated by strategic data sources having their own private utility functions and preferences (e.g., as in strategic classification or prediction, or incentive/pricing design problems), then the decision-maker does not necessarily have access to the distribution map $\mathcal{D}(x)$ in practice.

The zero-order stochastic gradient method proceeds as follows. Fix a parameter $\delta > 0$. In each epoch t, the Algorithm 1 samples v_t is a unit vector uniformly from the unit sphere $\mathbb{S}$ in dimension d, queries the environment for n_t iterations with $x_t + \delta v_t$, and then the loss oracle reveals $\ell(x_t + \delta v_t, z_t)$ where $z_t \sim \lambda^{n_t} p_{t-1} + (1 - \lambda^{n_t})\mathcal{D}(x_t + \delta v_t)$ which the decision maker uses to update x_t as follows:

$$x_{t+1} = \operatorname*{proj}_{(1-\delta)\mathcal{X}} \left(x_t - \eta \hat{g}_t\right),$$

where

$$\hat{g}_t = \frac{d}{\delta}\ell(x_t + \delta v_t, z_t)v_t. \tag{3}$$

This is a one-point gradient estimate of the expected loss at p_t. It can be shown that (3) is an unbiased estimate of the gradient of the smoothed loss function

$$\mathcal{L}_t^\delta(x) = \mathbb{E}_{v\sim\mathbb{S}}\left[\mathbb{E}_{z\sim p_t}\ell(x, z)\right]$$

at time t (e.g., in the general setting without decision-dependence this follows from Flaxman, Kalai, and McMahan (2004)). 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.

Define the smoothed expected risk as follows:

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

It is straightforward to show that $\mathcal{L}^\delta$ is strongly convex with parameter $(1-c)\alpha$ for some $c \in (0, 1)$ in the regime where $\delta \leq c\alpha/H$ (cf. Lemma 6, Appendix B).

To obtain convergence guarantees we need the following additional assumption.

Assumption 5. *The quantity $\ell_* := \sup\{|\ell(x, z)| : x \in \mathcal{X},\ z \in \mathcal{Z}\}$ is finite.*

The next lemma provides a crucial step in the proof of our main convergence result for the bandit feedback setting: it provides a bound on the bias due to the dynamics.

Lemma 2. *Under Assumptions 1, 2, 3, and 5, the error between the gradient smoothed loss $\mathcal{L}_t^\delta$ at p_t and the gradient of the smoothed expected loss $\mathcal{L}^\delta$ satisfies*

$$\begin{aligned}&\|\nabla\mathbb{E}_{v\sim\mathbb{B}}[\mathbb{E}_{z\sim p_t}[\ell(z, x_t + \delta v)]] - \nabla\mathcal{L}^\delta(x_t)\| \\ &\leq L \cdot \left(\lambda^{n_t}\overline{W}(p_0) + \lambda^{n_t}\frac{4\gamma d}{\alpha\delta}\frac{\lambda\ell_*}{(1-\lambda)^2}\right)\end{aligned}$$

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

We defer the proof to Appendix B.1.

To obtain the convergence rate, let $\bar{x}^\delta$ be the optimal point for $\mathcal{L}^\delta$ on $(1-\delta)\mathcal{X}$.

Theorem 1. *Suppose that Assumptions 1, 2, 3, and 5 hold. Let $\delta \le \min\{r, \frac{\alpha}{2H}\}$, and set step size $\eta_t = \frac{4}{\alpha t}$ and epoch length*

$$n_t \ge \log\left(\frac{\overline{W}(p_0) + \frac{4\gamma d}{\alpha\delta}\frac{\lambda \ell_*}{(1-\lambda)^2}}{\left(\eta_t \frac{\alpha}{L^2}\frac{\ell_*^2 d^2}{4\delta^2}\right)^{1/2}}\right)\frac{1}{\log(1/\lambda)}.$$

Then the estimate holds:

$$\mathbb{E}\,\|x_t - x^*\|^2 \le \frac{\max\{\alpha^2\delta^2\|x_1 - \bar{x}^\delta\|^2, 16d^2\ell_*^2\}}{t\alpha^2\delta^2} + 2\delta^2\left(\left(1+\frac{G}{\alpha}\right)\|x^*\| + \frac{G}{\alpha}\right)^2$$

The following corollary states the convergence rate.

Corollary 1 (Main result for zero-order oracle)**.** *Suppose the assumptions of Theorem 1 hold. Fix a target accuracy*

$$\varepsilon < 4r^2((1+\tfrac{G}{\alpha})R + \tfrac{G}{\alpha})^2,$$

and set $\delta = \alpha\sqrt{\varepsilon/4}/((\alpha+G)R+G)$ and $\eta_t = 4/(\alpha t)$. Then, the estimate $\mathbb{E}\,\|x_t - x^*\|^2 \le \varepsilon$ holds for all

$$t \ge \frac{\max\{8\alpha^2\varepsilon R^2, 128((\alpha+G)R+G)^2\ell_*^2 d^2\}}{\alpha^4\varepsilon^2}.$$

In the proceeding corollary, the lower bound on t is in terms of the number of epochs that Algorithm 1 needs to be run to obtain the target accuracy. In terms of total iterations across all epochs (i.e., $\sum_{k=1}^t n_k$), the rate is thus $O\left(\frac{d^2}{\varepsilon^2}\log\left(\frac{1}{\varepsilon}\right)\right)$.

First Order Stochastic Gradient Method

In many situations, the decision maker has access to a parametric description of the decision-dependent probability measure $\mathcal{D}(x)$ in which case the decision-maker can employ a stochastic gradient method. The challenge of having the distribution changing in time still remains, and hence the novelty of the results in this section.

To this end, let the expected loss at time t be given by

$$\mathcal{L}_t(x) = \mathbb{E}_{z\sim p_t}\ell(x_t, z). \tag{4}$$

Under Assumption 4 and mild smoothness assumptions, differentiating (4) we see that the gradient of $\mathcal{L}_t$ is simply

$$\nabla\mathcal{L}_t(x) = \mathop{\mathbb{E}}_{z\sim p_t}[\nabla_x\ell(x,z) + (1-\lambda^n)A^\top\nabla_z\ell(x,z)].$$

Therefore, given a point x, the decision-maker may draw $z \sim p_t$ and form the vector

$$\hat{g}_t = \nabla\ell(x_t, z) = \nabla_x\ell(x_t,z) + (1-\lambda^n)A^\top\nabla_z\ell(x_t,z).$$

By definition, $\hat{g}_t$ is an unbiased estimator of $\nabla\mathcal{L}_t(x)$, that is

$$\mathbb{E}_{z\sim p_t}[\hat{g}_t] = \nabla\mathcal{L}_t(x).$$

Algorithm 2 proceeds as follows. In round t, the decision maker queries the environment with x_t for n steps so that $p_t = \lambda^n p_{t-1} + (1-\lambda^n)\mathcal{D}(x_t)$. Then, the gradient oracle reveals $\hat{g}_t$ as defined above where $z \sim p_t$, and the decision maker updates x_t using $x_{t+1} = \mathrm{proj}_{\mathcal{X}}(x_t - \eta_t\hat{g}_t)$.

The following lemma is completely analogous to Lemma 2, and provides a bound on the gradient error due to the dynamics.

Lemma 3. *Under Assumptions 1, 2, and 4, the gradient error satisfies*

$$\|\nabla\mathbb{E}_{z\sim p_t}[\ell(z,x_t)]] - \nabla\mathcal{L}(x_t)\|^2 \le L^2\cdot\left(\lambda^n\overline{W}_1(p_0) + \lambda^n\gamma\eta\frac{L(1+\|A\|_{\mathrm{op}})\lambda}{(1-\lambda)^2}\right)^2$$

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

We defer the proof to Appendix C.1.

Assumption 6 (Finite Variance)**.** *There exists a constant $\sigma > 0$ satisfying*

$$\mathop{\mathbb{E}}_{z\sim p_t}[\|\hat{g}_t - \mathop{\mathbb{E}}_{z'\sim p_t}\nabla\ell(x,z')\|^2] \le \sigma^2 \quad \forall x\in\mathcal{X},\ \forall t \ge 1.$$

To justify the above assumption, we provide sufficient conditions for the above assumption to hold in terms of the variance of the partial gradients $\nabla_{x,z}\ell$.

Lemma 4 (Sufficient Conditions for Assumption 6)**.** *Suppose there exists constants $s_1, s_2 \ge 0$ such that for all $x \in \mathcal{X}$ the estimates hold:*

$$\mathop{\mathbb{E}}_{z\sim p_t}\|\nabla_x\ell(x,z) - \mathop{\mathbb{E}}_{z'\sim p_t}\nabla_x\ell(x,z')\|^2 \le s_1^2$$
$$\mathop{\mathbb{E}}_{z\sim p_t}\|\nabla_z\ell(x,z) - \mathop{\mathbb{E}}_{z'\sim p_t}\nabla_z\ell(x,z')\|^2 \le s_2^2$$

Then Assumption 6 holds with $\sigma_t^2 = 2(s_1^2 + \|A\|_{\mathrm{op}}^2 s_2^2)$.

Theorem 2. *Suppose that Assumptions 1, 2, 3, and 4 hold. For step-size $\eta \le \frac{\alpha}{2G^2}$ and epoch length*

$$n \ge \log\left(L\frac{\overline{W}_1(p_0) + \gamma\eta L(1+\|A\|_{\mathrm{op}})\frac{\lambda}{(1-\lambda)^2}}{(\alpha\eta)^{1/2}\sigma}\right)\frac{1}{\log(1/\lambda)},$$

the estimate holds:

$$\mathbb{E}\|x_{t+1} - x^*\|^2 \le \frac{1}{1+\eta\alpha}\mathbb{E}\|x_t - x^*\|^2 + \frac{4\eta^2\sigma^2}{1+\eta\alpha}.$$

We defer the proof to Appendix C.2. Applying a step-decay schedule on η yeilds the following corollary, the proof of which follows directly from the recursion in Theorem 2 and generic results on step decay schedules (see, e.g., Drusvyatskiy and Xiao (2020, Lemma B.2)).

Corollary 2 (Main result for first order oracle)**.** *Suppose the assumptions of Theorem 2 hold, and that Algorithm 2 is run in super-epochs indexed by $k = 1,\dots,K$ wherein each super-epoch is run for T_k epochs with constant step-size $\eta_k = \frac{\alpha}{2G^2}\cdot 2^{-k}$, and such that the last iterate of super-epoch k is used as the first iterate in super-epoch $k+1$. Fix a target accuracy $\varepsilon > 0$ and suppose $R > \|x_1 - x^*\|^2$ is available. Set*

$$T_1 = \left\lceil \tfrac{2}{\alpha\eta_1}\log(\tfrac{2R}{\varepsilon})\right\rceil,\ T_k = \left\lceil \tfrac{2\log(4)}{\alpha\eta_k}\right\rceil,\ \text{for } k \ge 2,$$

and $K = \lceil 1 + \log_2(\frac{2\eta_1\sigma^2}{\alpha\varepsilon})\rceil$. The final iterate x produced satisfies $\mathbb{E}\,\|x - x^*\|^2 \le \varepsilon$, while the total number of epochs is at most

$$O\left(\frac{G^2}{\alpha^2}\log\left(\frac{2R}{\varepsilon}\right) + \frac{\sigma^2}{\alpha^2\varepsilon}\right).$$

It is straightforward to show that the total number of iterations is $O\left(\frac{G^2}{\alpha^2}\log\left(\frac{2R}{\varepsilon}\right) + \frac{\sigma^2}{\alpha^2\varepsilon}\log(\frac{1}{\varepsilon})\right)$.

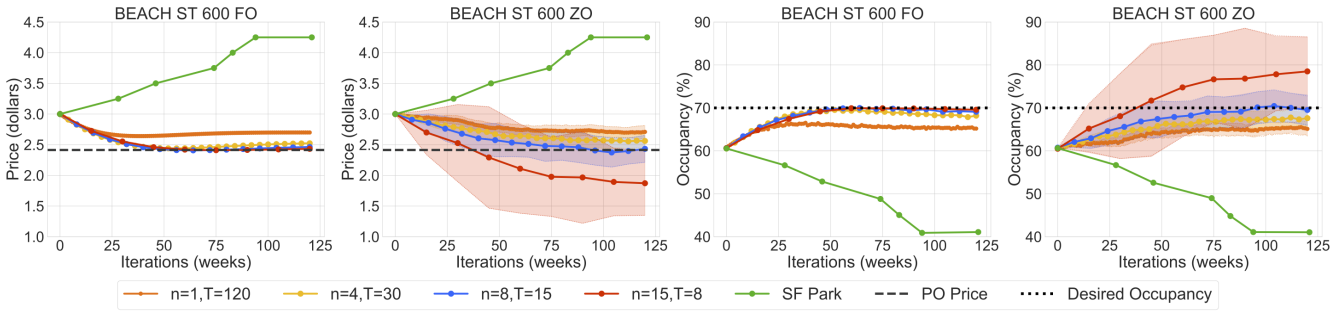

Figure 1: Results of Algorithm 2 (first and third plots) and Algorithm 1 (second and fourth plots) with different (n, T) pairs for 600 Beach ST and time window 1200–1500. Each marker represents a price announcement, and the plots show the prices and corresponding predicted occupancies. The SFpark prices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

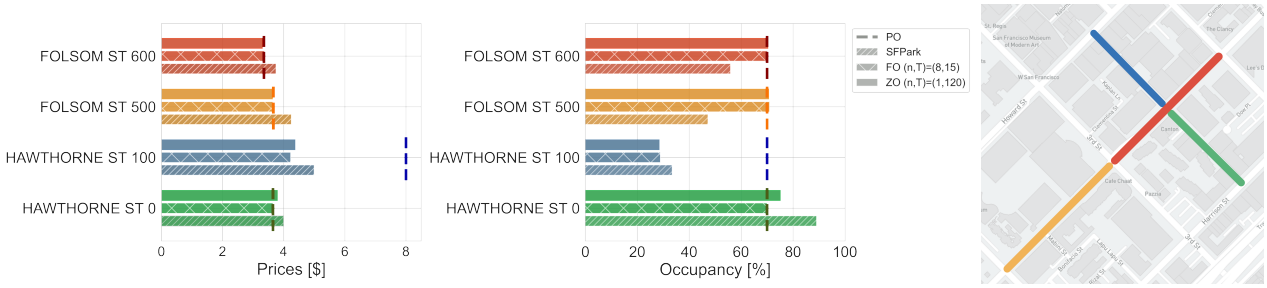

Figure 2: Final prices announced by first and zero order algorithms (Algorithms 2 and 1) run with $(n, T) = (8, 15)$ and $(n, T) = (1, 120)$, respectively, as compared to SFpark for streets depicted in the right graphic (color coded to the bar charts) during the 900–1200 time period. The center plot 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.

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).[2] 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. 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 is given by

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

where z is the vector of curb occupancies (which is between zero and one), x is the vector of changes in price from the nominal price at the beginning of the SFpark study for each curb, 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:

$$z \sim \mathcal{D}(x) \iff z = \zeta + Ax$$

where ζ 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).[3]

[1] SFpark: tinyurl.com/dwtf7wwn

[2] Code: https://github.com/ratlifflj/D3simulator.git

[3] 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 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 $\lambda \approx 0.959$ (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 the regularization parameter $\nu =$ 1e-3. The algorithms are run using parameters as dictated by Theorems 1 and 2, respectively, with the exception of epoch length. The epoch length we set to reasonable values as dictated by the parking application. In particular, the unit of time for an iteration is weeks, and we set the epoch length in terms of the number of weeks the price is held fixed. For instance, the SFpark study changed prices every eight weeks.[4]

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 occupancies 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.

SFpark changed prices approximately every eight weeks. As observed in Figure 1, this choice of n is reasonable—the estimated λ value is close to one—and leads to convergence to the optimal price change for both the first order and zero order algorithms. As n increases, the performance degrades, an observation that holds more generally for this curb. However, in our experiments, we found that different curbs had different optimal epoch lengths, thereby suggesting that a non-uniform price update schedule may lead to better outcomes. Appendix D.2 contains additional experiments.

Moreover, the prices under the 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 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 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.

[4]In Appendix D, we run synthetic experiments wherein the epoch length is chosen according to the theoretical results.

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 λ and A values (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.

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

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*.

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.

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.

Kaggle. 2011. Give me some credit dataset. https://www.kaggle.com/c/GiveMeSomeCredit. Accessed: 2022-03-27.

Kahneman, D.; and Tversky, A. 2013. Prospect theory: An analysis of decision under risk. In *Handbook of the fundamentals of financial decision making: Part I*, 99–127. World Scientific.

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

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*.

Nar, K.; Ratliff, L. J.; and Sastry, S. 2017. Learning prospect theory value function and reference point of a sequential decision maker. In *Proceedings of the IEEE 56th Annual Conference on Decision and Control (CDC)*, 5770–5775.

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.

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. *IIASA Working Paper*.

Technical Lemmas and Notation

Notation. Throughout we will use the following derivative and partial derivative notation. For a given function $\ell(z, x)$, the partial derivative of ℓ with respect to z is denoted $\nabla_z \ell(z, x)$ and the partial derivative with respect to x is denoted $\nabla_x \ell(z, x)$. For the expected risk $\mathbb{E}_{z\sim\mathcal{D}(x)}[\ell(z, x)]$, the total derivative with respect to x is denoted

$$\nabla \mathop{\mathbb{E}}_{z\sim\mathcal{D}(x)}[\ell(z,x)] = \nabla\left(\int \ell(z,x)\boldsymbol{p}_x(z)dz\right) = \mathop{\mathbb{E}}_{z\sim\mathcal{D}(x)}[\nabla_x\ell(z,x)] + \mathop{\mathbb{E}}_{z\sim\mathcal{D}(x)}[\ell(z,x)\nabla_x \log(\boldsymbol{p}_x(z))]$$

where $\boldsymbol{p}_x(z)$ is the density function for $\mathcal{D}(x)$ and in the last equality we have applied the so-called 'log trick'. Throughout, we use the notation $\|\cdot\|$ for the Euclidean norm.

Technical Lemmas. The following lemma is a direct consequence of dual form of the Wasserstein-1 distance.

Lemma 5. *Let $f : \mathbb{R}^n \to \mathbb{R}^n$ be β-Lipschitz, and let X, X' be random vectors with distributions p and p', respectively. Then,*

$$\|\mathbb{E}[f(X)] - \mathbb{E}[f(X')]\| \le \beta W_1(p, p').$$

Lemma 1 (Sufficient conditions for Assumption 3)**.** *Suppose that Assumption 4 holds and there exists a constants $\beta, \rho \ge 0$ such that the map $(x, z) \mapsto \nabla_{x,z}\ell(x, z)$ is β-Lipschitz continuous and has a ρ-Lipschitz continuous gradient. Then, Assumption 3 holds with constants*

$$G := \sqrt{\beta^2 \max\{1, \|A\|_{\mathrm{op}}^2\} \cdot (1 + \|A\|_{\mathrm{op}}^2)},$$

$$H := \sqrt{\rho^2 \max\{1, \|A\|_{\mathrm{op}}^4\} \cdot (1 + \|A\|_{\mathrm{op}}^2)}.$$

Proof. Observe that we may write

$$\nabla\mathcal{L}(x) = \mathop{\mathbb{E}}_{\zeta\sim\mathcal{P}} V^\top \nabla_{x,z}\ell(z, \zeta + Ax) \quad \text{where } V = \begin{bmatrix} I & 0 \\ 0 & A \end{bmatrix}.$$

Therefore, we deduce

$$\begin{aligned}
\|\nabla\mathcal{L}(x) - \nabla\mathcal{L}(y)\| &\le \|V\|_{\mathrm{op}} \mathop{\mathbb{E}}_{\zeta\sim\mathcal{P}} \|\nabla_{x,z}\ell(z, \zeta + Ax) - \nabla_{x,z}\ell(y, \zeta + Ay)\| \\
&\le \max\{1, \|A\|_{\mathrm{op}}\} \cdot \beta \cdot \mathop{\mathbb{E}}_{\zeta\sim\mathcal{P}} \|(z, \zeta + Ax) - (y, \zeta + Ay)\| \\
&= \max\{1, \|A\|_{\mathrm{op}}\} \cdot \beta \cdot \sqrt{\|x - y\|^2 + \|A(x - y)\|^2} \\
&\le \max\{1, \|A\|_{\mathrm{op}}\} \cdot \beta \cdot \sqrt{(1 + \|A\|_{\mathrm{op}}^2)} \cdot \|x - y\|.
\end{aligned}$$

Analogously, observe that

$$\nabla^2\mathcal{L}(x) = \mathop{\mathbb{E}}_{\zeta\sim\mathcal{P}} V^\top \nabla^2\ell(x, \zeta + Ax)V$$

where

$$\nabla^2\ell(x, z) = \begin{bmatrix} \nabla_x^2\ell(x, z) & \nabla_{xz}\ell(x, z) \\ \nabla_{zx}\ell(x, z) & \nabla_z^2\ell(x, z) \end{bmatrix}$$

is the Hessian of ℓ with respect to (x, z). Therefore, we deduce

$$\begin{aligned}
\|\nabla^2\mathcal{L}(x) - \nabla^2\mathcal{L}(y)\| &\le \|V\|_{\mathrm{op}}^2 \mathop{\mathbb{E}}_{\zeta\sim\mathcal{P}} \|\nabla^2\ell(z, \zeta + Ax) - \nabla^2\ell(y, \zeta + Ay)\| \\
&\le \max\{1, \|A\|_{\mathrm{op}}^2\} \cdot \rho \cdot \mathop{\mathbb{E}}_{\zeta\sim\mathcal{P}} \|(z, \zeta + Ax) - (y, \zeta + Ay)\| \\
&= \max\{1, \|A\|_{\mathrm{op}}^2\} \cdot \rho \cdot \sqrt{\|x - y\|^2 + \|A(x - y)\|^2} \\
&\le \max\{1, \|A\|_{\mathrm{op}}^2\} \cdot \rho \cdot \sqrt{(1 + \|A\|_{\mathrm{op}}^2)} \cdot \|x - y\|.
\end{aligned}$$

The proof is complete. □

Proofs for Zero Order Oracle Setting

Technical Lemmas

Recall that

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

Lemma 6. *Suppose that Assumptions 1, 2, and 3 hold. Choose $\delta \leq c\alpha/H$ for some constant $c \in (0,1)$. Then the map $\mathcal{L}^\delta$ is strongly convex over $\mathcal{X}$ with parameter $(1-c)\alpha$.*

Proof. We first estimate the Lipschitz constant of the difference map

$$h(x) := \nabla\mathcal{L}^\delta(x) - \nabla\mathcal{L}(x).$$

To this end, we compute

$$\nabla h(x) = \mathbb{E}_{w\sim\mathbb{B}}[\nabla^2\mathcal{L}(x+\mu w) - \nabla^2\mathcal{L}(x)].$$

Taking into account that the map $x \mapsto \nabla^2\mathcal{L}(x)$ is H-Lipschitz continuous, we deduce

$$\|\nabla h(x)\|_{\text{op}} \leq \mathbb{E}_{w\sim\mathbb{B}}[\|\nabla^2\mathcal{L}(x+\delta w) - \nabla^2\mathcal{L}(x)\|_{\text{op}}] \leq \delta H\mathbb{E}_{w\sim\mathbb{B}}\|w\| \leq \delta H.$$

Thus the map h is Lipschitz continuous with parameter δH. We therefore compute

$$\langle\nabla\mathcal{L}^\delta(x) - \nabla\mathcal{L}^\delta(x'), x-x'\rangle = \langle\nabla\mathcal{L}(x) - \nabla\mathcal{L}(x'), x-x'\rangle - \langle V(x)-V(x'), x-x'\rangle \geq (\alpha - H\delta)\|x-x'\|^2,$$

which completes the proof. □

Lemma 2. *Under Assumptions 1, 2, 3, and 5, the error between the gradient smoothed loss $\mathcal{L}_t^\delta$ at p_t and the gradient of the smoothed expected loss $\mathcal{L}^\delta$ satisfies*

$$\begin{aligned}&\|\nabla\mathbb{E}_{v\sim\mathbb{B}}[\mathbb{E}_{z\sim p_t}[\ell(z, x_t+\delta v)]] - \nabla\mathcal{L}^\delta(x_t)\| \\ &\leq L\cdot\left(\lambda^{n_t}\overline{W}(p_0) + \lambda^{n_t}\frac{4\gamma d}{\alpha\delta}\frac{\lambda\ell_*}{(1-\lambda)^2}\right)\end{aligned}$$

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

Proof of Lemma 2. Observe that using Jensen's inequality along with Lemma 5, we deduce

$$\begin{aligned}\|\nabla\mathbb{E}_{v\sim\mathbb{B}}[\mathbb{E}_{z\sim p_t}[\ell(z, x_t+\delta v)]] - \nabla\mathcal{L}^\delta(x_t)\|^2 &\leq \mathbb{E}_{v\sim\mathbb{B}}\left[\|\nabla\mathbb{E}_{z\sim p_t}[\ell(z, x_t+\delta v)] - \nabla\mathbb{E}_{z\sim\mathcal{D}(x_t+\delta v)}[\ell(z, x_t+\delta v)]\|^2\right] \\ &\leq \mathbb{E}_{v\sim\mathbb{B}}[L^2(W_1(p_t, \mathcal{D}(x_t+\delta v)))^2].\end{aligned}$$

Hence, we need an an upper bound on $\mathcal{W}_1(p_t, \mathcal{D}(x_t+\delta v_t))$ which is the Wasserstein-1 distance between the distribution at time t and the fixed point distribution for the query point $x_t + \delta v_t$.

Upper bound on $W_1(p_t, \mathcal{D}(x_t+\delta v))$. Using the fact that $p_t = \lambda^{n_t}p_{t-1} + (1-\lambda^{n_t})\mathcal{D}(x_t+\delta v)$, we expand $W_1(\mathcal{D}(x_t+\delta v), p_t)$ as follows:

$$\begin{aligned}W_1(p_t, \mathcal{D}(x_t+\delta v)) &= W_1(\lambda^{n_t}p_{t-1} + (1-\lambda^{n_t})\mathcal{D}(x_t+\delta v), \mathcal{D}(x_t+\delta v),) \\ &\leq \lambda^{n_t}W_1(p_{t-1}, \mathcal{D}(x_t+\delta v)) + (1-\lambda^{n_t})W_1(\mathcal{D}(x_t+\delta v), \mathcal{D}(x_t+\delta v)) \\ &= \lambda^{n_t}W_1(p_{t-1}, \mathcal{D}(x_t+\delta v)) \\ &= \lambda^{n_t}W_1(\lambda^{n_{t-1}}p_{t-2} + (1-\lambda^{n_{t-1}})\mathcal{D}(x_{t-1}+\delta v), \mathcal{D}(x_t+\delta v)) \\ &\leq \lambda^{n_t}\cdot\lambda\cdot W_1(p_{t-2}, \mathcal{D}(x_t+\delta v)) + \lambda^{n_t}(1-\lambda^{n_t})W_1(\mathcal{D}(x_{t-1}+\delta v), \mathcal{D}(x_t+\delta v)) \\ &\leq \lambda^{n_t}\cdot\lambda\cdot W_1(p_{t-2}, \mathcal{D}(x_t+\delta v)) + \lambda^{n_t}(1-\lambda^{n_t})\cdot\gamma\cdot\|x_t - x_{t-1}\| \\ &\leq \lambda^{n_t}\cdot\lambda\cdot W_1(p_{t-2}, \mathcal{D}(x_t+\delta v)) + \lambda^{n_t}\cdot\gamma\cdot\|x_t - x_{t-1}\|,\end{aligned}$$

where we have used the triangle inequality, Assumption 1(d), and the fact that $\lambda > \lambda^{n_t}$ for any $t \geq 1$, $\lambda^{n_t} < \lambda^{n_{t-i}}$ for any $t \in \{1, \ldots, t-1\}$, and $1 - \lambda^{n_t} < 1$. Continuing to unroll the recursion, we have that

$$\begin{aligned}\mathbb{E}_v W_1(p_t, \mathcal{D}(x_t+\delta v)) &\leq \lambda^{n_t}\cdot\lambda\mathbb{E}_v W_1(\lambda^{n_{t-2}}p_{t-3} + (1-\lambda^{n_{t-2}})\mathcal{D}(x_{t-2}+\delta v_{t-2}), \mathcal{D}(x_t+\delta v_t)) + \lambda^{n_t}\cdot\gamma\cdot\|x_t - x_{t-1}\| \\ &\leq \lambda^{n_t}\lambda^2\mathbb{E}_v W_1(p_{t-3}, \mathcal{D}(x_t+\delta v)) + \lambda\lambda^{n_t}\mathbb{E}_v\mathcal{W}_1(\mathcal{D}(x_t+\delta v), \mathcal{D}(x_{t-2}+\delta v))) + \lambda^{n_t}\cdot\gamma\|x_t - x_{t-1}\| \\ &\leq \lambda^{n_t}\cdot\lambda^2\mathbb{E}_v W_1(p_{t-3}, \mathcal{D}(x_t+\delta v)) + \lambda^{n_t}\cdot\gamma\mathbb{E}_v(\|x_t - x_{t-1}\| + \lambda\cdot\|x_t - x_{t-2}\|) \\ &\leq \lambda^{n_t}\lambda^{t-1}\mathbb{E}_v W_1(\mathcal{D}(x_t+\delta v), p_0) + \lambda^{n_t}\gamma\sum_{i=1}^{t-1}\lambda^{(i-1)}\mathbb{E}_v\|x_t - x_{t-i}\|. \qquad (5)\end{aligned}$$

Hence, we need a bound on $\|x_t - x_{t-i}\|$ for each $i \in \{1, \dots, t-1\}$. Using the fact that $x_t = x_{t-1} - \eta_t \frac{d}{\delta} \ell(x_{t-1} + \delta v_{t-1}, z_{t-1}) v_{t-1}$ where $z_{t-1} \sim p_{t-1}$, we have that

$$\begin{aligned}
\|x_t - x_{t-i}\| &= \|x_{t-1} - \eta_{t-1} \tfrac{d}{\delta} \ell(x_{t-1} + \delta v_{t-1}, z) v_{t-1} - x_{t-i}\| \\
&= \|x_{t-2} - \eta_{t-2} \frac{d}{\delta} \ell(x_{t-2} + \delta v_{t-2}, z_{t-2}) v_{t-2} - \eta_{t-1} \frac{d}{\delta} \ell(x_{t-1} + \delta v_{t-1}, z_{t-1}) v_{t-1} - x_{t-i}\| \\
&\leq \frac{d}{\delta} \eta_1 \sum_{j=t-i}^{t-1} |\ell(x_j + \delta v_j, z_j)| \|v_j\| \\
&\leq \eta_1 \frac{d}{\delta} \ell_*(i-1).
\end{aligned}$$

Hence, we have that

$$\begin{aligned}
\mathbb{E}_v W_1(p_t, \mathcal{D}(x_t + \delta v)) &\leq \lambda^{n_t} \lambda^{t-1} \mathbb{E}_v W_1(\mathcal{D}(x_t + \delta v), p_0) + \lambda^{n_t} \gamma \sum_{i=1}^{t-1} \lambda^{(i-1)} \|x_t - x_{t-i}\| \\
&\leq \lambda^{n_t} \lambda^{t-1} \mathbb{E}_v W_1(\mathcal{D}(x_t + \delta v), p_0) + \lambda^{n_t} \gamma \sum_{i=1}^{t-1} \lambda^{(i-1)} \eta_1 \frac{d}{\delta} \ell_*(i-1) \\
&\leq \lambda^{n_t} \overline{W}(p_0) + \lambda^{n_t} \frac{4\gamma d}{\alpha \delta} \frac{\ell_* \lambda}{(1-\lambda)^2},
\end{aligned}$$

where the last inequality holds using the fact that $\sum_{i=1}^{t-1} \lambda^{(i-1)}(i-1) \leq \frac{\lambda}{(1-\lambda)^2}$.

Bounding gradient error. Using this bound, we deduce

$$\begin{aligned}
\|\nabla \mathbb{E}_{v \sim \mathbb{B}}[\mathbb{E}_{z \sim p_t}[\ell(z, x_t + \delta v)]] - \nabla \mathcal{L}^\delta(x_t)\|^2 &\leq \mathbb{E}_{v \sim \mathbb{B}}[L^2 (W_1(p_t, \mathcal{D}(x_t + \delta v)))^2] \\
&\leq L^2 \left(\lambda^{n_t} \overline{W}(p_0) + \lambda^{n_t} \frac{4\gamma d}{\alpha \delta} \frac{\lambda \ell_*}{(1-\lambda)^2} \right)^2.
\end{aligned}$$

This concludes the proof. □

Lemma 7. *Suppose that Assumptions 1 and 3 hold. The loss $\mathcal{L}^\delta(x)$ is differentiable and the map $x \mapsto \nabla \mathcal{L}^\delta(x)$ is G-Lipschitz continuous. Moreover, the estimate holds:*

$$\|\nabla \mathcal{L}(x) - \nabla \mathcal{L}^\delta(x)\| \leq G\delta \quad \forall\, x \in \mathcal{X}.$$

Proof. For any point $x, x' \in \mathcal{X}$, we successively estimate

$$\|\nabla \mathcal{L}^\delta(x) - \nabla \mathcal{L}^\delta(x')\| \leq \mathop{\mathbb{E}}_{w \sim \mathbb{B}}[\|\nabla \mathcal{L}(x + \delta w) - \nabla \mathcal{L}(x' + \delta w)\|] \leq G\|x - x'\|$$

Thus $\nabla \mathcal{L}^\delta$ is G-Lipschitz continuous. Next, we estimate

$$\|\nabla \mathcal{L}(x) - \nabla \mathcal{L}^\delta(x)\| \leq \mathop{\mathbb{E}}_{w \sim \mathbb{B}}[\|\nabla \mathcal{L}(x + \delta w) - \nabla \mathcal{L}(x)\|] \leq G \cdot \mathop{\mathbb{E}}_{w \sim \mathbb{B}} \|w\| \leq G \cdot \delta,$$

which concludes the proof. □

Define the smoothed loss at p_t is defined as

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

Let $\bar{x}^\delta$ the optimal point of $\mathcal{L}^\delta$ on $(1-\delta)\mathcal{X}$, and x^δ be the optimal point of $\mathcal{L}^\delta$ on $\mathcal{X}$. We have the following bound on the distance between the optimum of the performative prediction problem defined by L on $\mathcal{X}$ and the optimum of the perturbed problem defined by $\mathcal{L}^\delta$ on $(1-\delta)\mathcal{X}$.

The normal cone to a convex set $\mathcal{X}$ at $x \in \mathcal{X}$, denoted by $N_{\mathcal{X}}(x)$ is the set

$$N_{\mathcal{X}}(x) = \{v \in \mathbb{R}^d : \langle v, y - x \rangle \leq 0 \ \ \forall y \in \mathcal{X}\}.$$

Lemma 8. *Choose $\delta < \min\{r, \frac{\alpha}{H}\}$. Then the estimate holds:*

$$\|x^* - \bar{x}^\delta\| \leq \delta \left(\left(1 + \frac{G}{\alpha}\right) \|x^*\| + \frac{G}{\alpha} \right).$$

Proof. There are two sources of perturbation: one replacing $\mathcal{X}$ with $(1-\delta)\mathcal{X}$ and the other in replacing $\mathcal{L}$ with $\mathcal{L}^\delta$. We will deal with each one individually. To do so, set $\tau := 1 - \delta$ and let $\tilde{x}$ be the optimal point for $\mathcal{L}$ on the shrunken set $\tau\mathcal{X}$. Thus $\tilde{x}$ satisfies the inclusion $0 \in \nabla\mathcal{L}(\tilde{x}) + N_{\tau\mathcal{X}}(\tilde{x})$ where $N_{\tau\mathcal{X}}(\tilde{x})$ denotes the normal cone to $\tau\mathcal{X}$ at $\tilde{x}$. The triangle inequality directly gives

$$\|x^* - \bar{x}^\delta\| \leq \|x^* - \tilde{x}\| + \|\tilde{x} - \bar{x}^\delta\|. \tag{6}$$

Let us bound the first term on the right hand side of (6). To this end, since the map $x \mapsto \nabla\mathcal{L}(x) + N_{\tau\mathcal{X}}(x)$ is α-strongly monotone, we deduce

$$\alpha\|\tilde{x} - \tau x^*\| \leq \operatorname{dist}(0, \nabla\mathcal{L}(\tau x^*) + N_{\tau\mathcal{X}}(\tau x^*)). \tag{7}$$

Let use estimate the right hand side of (7). Since x^* is optimal, the inclusion $0 \in \nabla\mathcal{L}(x^*) + N_{\mathcal{X}}(x^*)$ holds. Taking into account the identity $N_{\tau\mathcal{X}}(\tau^*) = N_{\mathcal{X}}(x^*)$, we deduce

$$\operatorname{dist}(0, \nabla\mathcal{L}(\tau x^*) + N_{\tau\mathcal{X}}(\tau x^*)) = \operatorname{dist}(0, \nabla\mathcal{L}(\tau x^*) + N_{\mathcal{X}}(x^*)) \leq \|\nabla\mathcal{L}(\tau x^*) - \nabla\mathcal{L}(x^*)\| \leq \delta \cdot G \cdot \|x^*\|,$$

where the last inequality holds since $\nabla\mathcal{L}$ is G-Lipschitz continuous. Appealing to (7) and using the triangle inequality, we therefore deduce

$$\|x^* - \tilde{x}\| \leq \|\tilde{x} - \tau x^*\| + \delta\|x^*\| \leq \delta \left(1 + \frac{G}{\alpha}\right) \|x^*\|.$$

It remains to upper bound $\|\tilde{x} - x^*\|$. Since $\tilde{x}$ is optimal, we have that

$$\langle -\nabla\mathcal{L}(\tilde{x}), x - \tilde{x} \rangle \leq 0, \ \ \forall\, x \in \tau\mathcal{X}. \tag{8}$$

Analogously, since $\bar{x}^\delta$ is also optimal, we have that

$$\langle -\nabla\mathcal{L}^\delta(\bar{x}^\delta), x - \bar{x}^\delta \rangle \leq 0, \ \ \forall\, x \in \tau\mathcal{X}. \tag{9}$$

Then, by strong convexity of the game and estimates (8) and (9), we get that

$$\begin{aligned}
\alpha\|\tilde{x} - \bar{x}^\delta\|^2 &\leq \langle \nabla\mathcal{L}(\tilde{x}) - \nabla\mathcal{L}(\bar{x}^\delta), \tilde{x} - \bar{x}^\delta \rangle \\
&\leq \langle \nabla\mathcal{L}^\delta(\bar{x}^\delta) - \nabla\mathcal{L}(\bar{x}^\delta), \tilde{x} - \bar{x}^\delta \rangle \\
&\leq \|\nabla\mathcal{L}^\delta(\bar{x}^\delta) - \nabla\mathcal{L}(\bar{x}^\delta)\| \|\tilde{x} - \bar{x}^\delta\| \\
&\leq G \cdot \delta \cdot \|\tilde{x} - \bar{x}^\delta\|
\end{aligned}$$

where the last inequality follows from Lemma 7. □

The following lemma holds by a simple inductive argument.

Lemma 9. *Consider a sequence $D_t \geq 0$ for $t \geq 1$ and constants $t_0 \geq 0$, $a > 0$ satisfying*

$$D_{t+1} \leq \left(1 - \frac{2}{t + t_0}\right) D_t + \frac{a}{(t + t_0)^2}.$$

Then the estimate holds:

$$D_t \leq \frac{\max\{(1 + t_0)D_1, a\}}{t + t_0} \quad \forall t \geq 1.$$

Proof of Theorem 1

Theorem 1. *Suppose that Assumptions 1, 2, 3, and 5 hold. Let $\delta \leq \min\{r, \frac{\alpha}{2H}\}$, and set step size $\eta_t = \frac{4}{\alpha t}$ and epoch length*

$$n_t \geq \log\left(\frac{\overline{W}(p_0) + \frac{4\gamma d}{\alpha\delta} \frac{\lambda \ell_*}{(1-\lambda)^2}}{\left(\eta_t \frac{\alpha}{L^2} \frac{\ell_*^2 d^2}{4\delta^2}\right)^{1/2}} \right) \frac{1}{\log(1/\lambda)}.$$

Then the estimate holds:

$$\begin{aligned}
\mathbb{E}\,\|x_t - x^*\|^2 \leq{}& \frac{\max\{\alpha^2\delta^2\|x_1 - \bar{x}^\delta\|^2, 16 d^2 \ell_*^2\}}{t\alpha^2\delta^2} \\
&+ 2\delta^2 \left(\left(1 + \frac{G}{\alpha}\right) \|x^*\| + \frac{G}{\alpha} \right)^2
\end{aligned}$$

Proof. Adding and subtracting appropriately, we have that

$$\begin{aligned}\frac{1}{2}\|x_{t+1}-x^*\|^2 &\le \|x_{t+1}-\bar{x}^\delta\|^2+\|\bar{x}^\delta-x^*\|^2\\ &\le \|x_{t+1}-\bar{x}^\delta\|^2+\delta^2\left(\left(1+\frac{G}{\alpha}\right)\|x^*\|+\frac{G}{\alpha}\right)^2\end{aligned}$$

Now, to bound $\|x_{t+1}-\bar{x}^\delta\|$, we have that

$$\begin{aligned}\mathbb{E}[\|x_{t+1}-\bar{x}^\delta\|^2] &\le \mathbb{E}[\|x_t-\bar{x}^\delta-\eta_t\hat{g}_t(x_t)\|^2]\\ &\le \mathbb{E}[\|x_t-\bar{x}^\delta\|^2]-2\eta_t\,\mathbb{E}\langle\hat{g}_t(x_t),x_t-\bar{x}^\delta\rangle+\eta_t^2\,\mathbb{E}\,\|\hat{g}_t(x_t)\|^2\\ &= \mathbb{E}[\|x_t-\bar{x}^\delta\|^2]-2\eta_t\,\mathbb{E}\langle\nabla\mathcal{L}_t^\delta(x_t),x_t-\bar{x}^\delta\rangle+\eta_t^2\,\mathbb{E}\,\|\hat{g}_t(x_t)\|^2\end{aligned}$$

where the last equality holds since $\mathbb{E}[\hat{g}_t(x_t)]=\nabla\mathcal{L}_t^\delta(x_t)$. We rewrite the smoothed gradient of the loss at time t as

$$\nabla\mathcal{L}_t^\delta(x_t)=\nabla\mathcal{L}^\delta(x_t)+\nabla\mathcal{L}_t^\delta(x_t)-\nabla\mathcal{L}^\delta(x_t).$$

Hence

$$\begin{aligned}\mathbb{E}[\|x_{t+1}-\bar{x}^\delta\|^2] &\le \mathbb{E}[\|x_t-\bar{x}^\delta-\eta_t\hat{g}_t(x_t)\|^2]\\ &\le \mathbb{E}[\|x_t-\bar{x}^\delta\|^2]-2\eta_t\,\mathbb{E}\langle\nabla\mathcal{L}^\delta(x_t),x_t-\bar{x}^\delta\rangle-2\eta_t\,\mathbb{E}\langle\nabla\mathcal{L}_t^\delta(x_t)-\nabla\mathcal{L}^\delta(x_t),x_t-\bar{x}^\delta\rangle+\eta_t^2\,\mathbb{E}\,\|\hat{g}_t(x_t)\|^2\\ &\le (1-\eta_t\alpha)\,\mathbb{E}[\|x_t-\bar{x}^\delta\|^2]-2\eta_t\,\mathbb{E}\langle\nabla\mathcal{L}_t^\delta(x_t)-\nabla\mathcal{L}^\delta(x_t),x_t-\bar{x}^\delta\rangle+\eta_t^2\frac{\ell_*^2d^2}{2\delta^2},\end{aligned}$$

where we used the fact that the smoothed loss is $(1-c)\alpha$ strongly convex for any $c\in(0,1)$ and we let $c:=1/2$. Using the fact that

$$\mathbb{E}\,|\langle\nabla\mathcal{L}_t^\delta(x_t)-\nabla\mathcal{L}^\delta(x_t),x_t-\bar{x}^\delta\rangle| \le \frac{1}{2\Delta_1}L^2\left(\lambda^{n_t}\overline{W}(p_0)+\lambda^{n_t}\frac{4\gamma d}{\alpha\delta}\frac{\ell_*}{(1-\lambda)^2}\right)^2+\frac{\Delta_1\,\mathbb{E}\,\|x_t-\bar{x}^\delta\|^2}{2},$$

we have that

$$\begin{aligned}\mathbb{E}[\|x_{t+1}-\bar{x}^\delta\|^2] &\le (1-\eta_t\alpha)\,\mathbb{E}[\|x_t-\bar{x}^\delta\|^2]+\eta_t^2\frac{\ell_*^2d^2}{2\delta^2}\\ &\quad+2\eta_t\left(\frac{1}{2\Delta_1}L^2\left(\lambda^{n_t}\overline{W}(p_0)+\lambda^{n_t}\frac{4\gamma d}{\alpha\delta}\frac{\ell_*}{(1-\lambda)^2}\right)^2+\frac{\Delta_1\,\mathbb{E}\,\|x_t-\bar{x}^\delta\|^2}{2}\right)\\ &= (1-\eta_t(\alpha-\Delta_1))\,\mathbb{E}[\|x_t-\bar{x}^\delta\|^2]+\eta_t^2\frac{\ell_*^2d^2}{2\delta^2}+2\eta_t\left(\frac{1}{2\Delta_1}L^2\left(\lambda^{n_t}\overline{W}(p_0)+\lambda^{n_t}\frac{4\gamma d}{\alpha\delta}\frac{\ell_*}{(1-\lambda)^2}\right)^2\right)\\ &\le (1-\eta_t\frac{\alpha}{2})\,\mathbb{E}[\|x_t-\bar{x}^\delta\|^2]+\eta_t^2\frac{\ell_*^2d^2}{2\delta^2}+2\eta_t\left(\frac{1}{\alpha}L^2\left(\lambda^{n_t}\overline{W}(p_0)+\lambda^{n_t}\frac{4\gamma d}{\alpha\delta}\frac{\ell_*}{(1-\lambda)^2}\right)^2\right)\end{aligned}$$

where we use $\Delta_1:=\alpha/2$. Now, since

$$n_t\ge\log\left(\frac{\overline{W}(p_0)+\frac{4\gamma d}{\alpha\delta}\frac{\ell_*}{(1-\lambda)^2}}{\left(\eta_t\frac{\alpha}{L^2}\frac{\ell_*^2d^2}{4\delta^2}\right)^{1/2}}\right)\frac{1}{\log(1/\lambda)},$$

we have that

$$\lambda^{n_t}\left(\overline{W}(p_0)+\frac{4\gamma d}{\alpha\delta}\frac{\ell_*}{(1-\lambda)^2}\right)\le\left(\eta_t\frac{\alpha}{L^2}\frac{\ell_*^2d^2}{4\delta^2}\right)^{1/2}.$$

so that

$$2\eta_t\left(\frac{L^2}{\alpha}\left(\lambda^{n_t}\overline{W}(p_0)+\lambda^{n_t}\frac{4\gamma d}{\alpha\delta}\frac{\ell_*}{(1-\lambda)^2}\right)^2\right)\le\eta_t^2\frac{\ell_*^2d^2}{2\delta^2}.$$

Therefore, we deduce

$$\mathbb{E}[\|x_{t+1}-\bar{x}^\delta\|^2]\le\left(1-\eta_t\frac{\alpha}{2}\right)\mathbb{E}[\|x_t-\bar{x}^\delta\|^2]+\eta_t^2\frac{\ell_*^2d^2}{\delta^2}$$

Since $\eta_t=4/(\alpha t)$, we apply Lemma 9 to deduce that

$$\mathbb{E}\,\|x_{t+1}-\bar{x}^\delta\|^2\le\frac{\max\{\alpha^2\delta^2\|x_1-\bar{x}^\delta\|^2,16\ell_*^2d^2\}}{\delta^2\alpha^2t}\quad\forall t\ge1.$$

This concludes the proof.

□

Proof of Corollary 1

Corollary 1 (Main result for zero-order oracle)**.** *Suppose the assumptions of Theorem 1 hold. Fix a target accuracy*

$$\varepsilon < 4r^2((1+\tfrac{G}{\alpha})R + \tfrac{G}{\alpha})^2,$$

and set $\delta = \alpha\sqrt{\varepsilon/4}/((\alpha+G)R+G)$ *and* $\eta_t = 4/(\alpha t)$*. Then, the estimate* $\mathbb{E}\,\|x_t - x^*\|^2 \le \varepsilon$ *holds for all*

$$t \ge \frac{\max\{8\alpha^2\varepsilon R^2, 128((\alpha+G)R+G)^2\ell_*^2 d^2\}}{\alpha^4\varepsilon^2}.$$

Proof. The assumed upper bound on ε directly implies that $\delta \le \frac{\alpha}{2G}$ and $\delta < r$. An application of Theorem 1 yeilds the estimate

$$\mathbb{E}[\|x_t - x^*\|^2] \le \frac{\max\{\delta^2\alpha^2\|x_1 - \bar{x}^\delta\|^2, 16d^2\ell_*^2\}}{t\alpha^2\delta^2} + \frac{\varepsilon}{2}.$$

Setting the right side to ε, solving for t, and using the trivial upper bound $\|x_1 - \bar{x}^\delta\| \le 2R$ completes the proof. □

Proofs for First Order Oracle Setting

Proof of Lemma 3

Lemma 3. *Under Assumptions 1, 2, and 4, the gradient error satisfies*

$$\begin{aligned}&\|\nabla\mathbb{E}_{z\sim p_t}[\ell(z,x_t)]] - \nabla\mathcal{L}(x_t)\|^2\\ &\quad\le L^2\cdot\left(\lambda^n\overline{W}_1(p_0) + \lambda^n\gamma\eta\frac{L(1+\|A\|_{\mathrm{op}})\lambda}{(1-\lambda)^2}\right)^2\end{aligned}$$

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

Proof. Observe that using Jensen's inequality along with Lemma 5, we deduce

$$\begin{aligned}\|\nabla\mathbb{E}_{z\sim p_t}[\ell(z,x_t)]] - \nabla\mathcal{L}(x_t)\|^2 &= \left[\|\nabla\mathbb{E}_{z\sim p_t}[\ell(z,x_t)] - \nabla\mathbb{E}_{z\sim\mathcal{D}(x_t)}[\ell(z,x_t)]\|^2\right]\\ &\le L^2(W_1(p_t,\mathcal{D}(x_t)))^2.\end{aligned}$$

The remainder of the proof is identical to the proof of Lemma 2. Indeed, we have that

$$W_1(p_t,\mathcal{D}(x_t)) \le \lambda^n\lambda^{t-1}W_1(\mathcal{D}(x_t),p_0) + \lambda^n\gamma\sum_{i=1}^{t-1}\lambda^{(i-1)}\|x_t - x_{t-i}\|.$$

Hence, we need a bound on $\|x_t - x_{t-i}\|$ for each $i \in \{1,\dots,t-1\}$. Recall that $x_t = x_{t-1} - \eta\hat{g}_{t-1}$ where

$$\hat{g}_{t-1} = \nabla_x\ell(x_{t-1},z_{t-1}) + (1-\lambda^n)A^\top\nabla_z\ell(x_{t-1},z_{t-1}), \quad \text{and} \quad z_{t-1}\sim p_{t-1}.$$

Moreover,

$$\|\hat{g}_t\| \le L(1+\|A\|_{\mathrm{op}})$$

since ℓ is L-Lipschitz continuous. Hence, we have the following bound:

$$\begin{aligned}\|x_t - x_{t-i}\| &= \|x_{t-1} - \eta\hat{g}_{t-1} - x_{t-i}\|\\ &= \|x_{t-2} - \eta\hat{g}_{t-2} - \eta_{t-1}\hat{g}_{t-1} - x_{t-i}\|\\ &\le \eta\sum_{j=t-i}^{t-1}\|\hat{g}_j\|\\ &\le \eta\cdot L\cdot(1+\|A\|_{\mathrm{op}})\cdot(i-1).\end{aligned}$$

Therefore, we deduce

$$\begin{aligned}W_1(p_t,\mathcal{D}(x_t)) &\le \lambda^n\lambda^{t-1}W_1(\mathcal{D}(x_t),p_0) + \lambda^{n_t}\gamma\sum_{i=1}^{t-1}\lambda^{(i-1)}\eta\cdot L\cdot(1+\|A\|_{\mathrm{op}})\cdot(i-1)\\ &\le \lambda^n\lambda^{t-1}W_1(\mathcal{D}(x_t),p_0) + \lambda^n\gamma\eta\cdot L\cdot(1+\|A\|_{\mathrm{op}})\cdot\frac{\lambda}{(1-\lambda)^2},\end{aligned}$$

where the last inequality follows from the fact that $\sum_{i=1}^{t-1}\lambda^{(i-1)}(i-1) \le \frac{\lambda}{(1-\lambda)^2}$. Using this bound on the Wasserstein-1 distance between the current probability distribution p_t at time t and the fixed point probability distribution $\mathcal{D}(x_t)$ induced by x_t, we have that

$$\|\nabla\mathbb{E}_{z\sim p_t}[\ell(z,x_t)]] - \nabla\mathcal{L}(x_t)\|^2 \le L^2\cdot\left(\lambda^n\overline{W}_1(p_0) + \lambda^n\gamma\eta\cdot L\cdot(1+\|A\|_{\mathrm{op}})\cdot\frac{\lambda}{(1-\lambda)^2}\right)^2$$

since $\lambda^{t-1} \le 1$. This concludes the proof. □

Proof of Theorem 2

We restate the theorem for convienience.

Theorem 2. *Suppose that Assumptions 1, 2, 3, and 4 hold. For step-size $\eta \leq \frac{\alpha}{2G^2}$ and epoch length*

$$n \geq \log\left(L\frac{\overline{W}_1(p_0) + \gamma\eta L(1+\|A\|_{\mathrm{op}})\frac{\lambda}{(1-\lambda)^2}}{(\alpha\eta)^{1/2}\sigma}\right)\frac{1}{\log(1/\lambda)},$$

the estimate holds:

$$\mathbb{E}\|x_{t+1} - x^*\|^2 \leq \frac{1}{1+\eta\alpha}\mathbb{E}\|x_t - x^*\|^2 + \frac{4\eta^2\sigma^2}{1+\eta\alpha}.$$

Note that the gradient $\hat{g}_t$ approximates the gradient $\mathcal{G}(x) := \nabla\mathcal{L}(x) = \nabla\,\mathbb{E}_{z\sim\mathcal{D}(x)}\,\ell(x,z)$.

$$\|x_{t+1} - x_t\| = \|x_t - \eta\hat{g}_t - x_t\| = \|x_t - \eta\mathcal{G}(x_t) - \eta(\hat{g}_t - \mathcal{G}(x_t)) - x_t\|$$

Noting that x_{t+1} is the minimizer of the 1–strongly convex function $x \mapsto \frac{1}{2}\|x_t - \eta\hat{g}_t - x\|^2$ over $\mathcal{X}$, we deduce

$$\frac{1}{2}\|x_{t+1} - x^*\|^2 \leq \frac{1}{2}\|x_t - \eta\hat{g}_t - x^*\|^2 - \frac{1}{2}\|x_t - \eta\hat{g}_t - x_{t+1}\|^2.$$

Expanding the squares on the right hand side and combining terms yields

$$\begin{aligned}\frac{1}{2}\|x_{t+1} - x^*\|^2 &\leq \frac{1}{2}\|x_t - x^*\|^2 - \eta\langle\hat{g}_t, x_{t+1} - x^*\rangle - \frac{1}{2}\|x_{t+1} - x_t\|^2 \\ &= \frac{1}{2}\|x_t - x^*\|^2 - \eta\langle\hat{g}_t, x_t - x^*\rangle - \frac{1}{2}\|x_{t+1} - x^*\|^2 - \eta\langle\hat{g}_t, x_{t+1} - x_t\rangle.\end{aligned}$$

Setting $\mu_t := \mathbb{E}_t[\hat{g}_t]$, we successively compute

$$\begin{aligned}\frac{1}{2}\mathbb{E}_t\|x_{t+1} - x^*\|^2 &\leq \frac{1}{2}\|x_t - x^*\|^2 - \eta\langle\mathbb{E}_t\hat{g}_t, x_t - x^*\rangle - \frac{1}{2}\mathbb{E}_t\|x_{t+1} - x^*\|^2 - \eta\mathbb{E}_t\langle\hat{g}_t, x_{t+1} - x_t\rangle \\ &\leq \frac{1}{2}\|x_t - x^*\|^2 - \eta\langle\mu_t, x_t - x^*\rangle - \frac{1}{2}\mathbb{E}_t\|x_{t+1} - x^*\|^2 - \eta\mathbb{E}_t\langle\hat{g}_t, x_{t+1} - x_t\rangle \\ &= \frac{1}{2}\|x_t - x^*\|^2 - \eta\mathbb{E}_t\langle\mathcal{G}(x_{t+1}), x_{t+1} - x^*\rangle - \frac{1}{2}\mathbb{E}_t\|x_{t+1} - x^*\|^2 \\ &\quad + \eta\underbrace{\mathbb{E}_t\langle\hat{g}_t - \mu_t, x_t - x_{t+1}\rangle}_{P_1} + \eta\underbrace{\mathbb{E}_t[\langle\mu_t - \mathcal{G}(x_{t+1}), x^* - x_{t+1}\rangle]}_{P_2}.\end{aligned}$$

Strong convexity of $\mathcal{L}(x)$ implies that $\langle\mathcal{G}(x_{t+1}), x_{t+1} - x^*\rangle \geq \alpha\|x_{t+1} - x^*\|^2$ so that

$$\frac{1+2\eta\alpha}{2}\mathbb{E}_t\|x_{t+1} - x^*\|^2 \leq \frac{1}{2}\|x_t - x^*\|^2 - \frac{1}{2}\mathbb{E}_t\|x_{t+1} - x_t\|^2 + \eta(P_1 + P_2).$$

Using Young's inequality, we upper bound P_1 as follows:

$$\begin{aligned}P_1 &\leq \frac{1}{2\Delta_1}\mathbb{E}_t\|\hat{g}_t - \mu_t\|^2 + \frac{\Delta_1\mathbb{E}_t\|x_{t+1} - x_t\|^2}{2} \\ &\leq \frac{\sigma^2}{2\Delta_1} + \frac{\Delta_1\mathbb{E}_t\|x_{t+1} - x_t\|^2}{2}\end{aligned}$$

using Assumption 6. Using Yong's inequality again, we have that

$$P_2 \leq \frac{\mathbb{E}_t\|\mu_t - \mathcal{G}(x_{t+1})\|^2}{2\Delta_2} + \frac{\Delta_2\mathbb{E}_t\|x_{t+1} - x^*\|^2}{2}.$$

Next observe that

$$\begin{aligned}\mathbb{E}_t\|\mu_t - \mathcal{G}(x_{t+1})\|^2 &\leq 2\mathbb{E}_t\|\mu_t - \mathcal{G}(x_t)\|^2 + 2\mathbb{E}_t\|\mathcal{G}(x_t) - \mathcal{G}(x_{t+1})\|^2 \\ &\leq 2C^2 + 2G^2\mathbb{E}_t\|x_t - x_{t+1}\|^2,\end{aligned}$$

where

$$C^2 := L^2 \cdot \left(\lambda^n\overline{W}_1(p_0) + \lambda^n\gamma\eta \cdot L \cdot (1+\|A\|_{\mathrm{op}}) \cdot \frac{\lambda}{(1-\lambda)^2}\right)^2.$$

Therefore

$$P_2 \leq \frac{2C^2 + 2G^2\|x_t - x_{t+1}\|^2}{2\Delta_2} + \frac{\Delta_2 \mathbb{E}_t\|x_{t+1} - x^*\|^2}{2}. \tag{10}$$

Now we have that

$$\frac{1+\eta(2\alpha - \Delta_2)}{2}\mathbb{E}_t\|x_{t+1} - x^*\|^2 \leq \frac{1}{2}\|x_t - x^*\|^2 + \frac{\eta\sigma^2}{2\Delta_1} + \frac{\eta C^2}{\Delta_2} - \frac{1 - 2\eta G^2 \Delta_2^{-1} - \eta\Delta_1}{2}\mathbb{E}_t\|x_{t+1} - x_t\|^2. \tag{11}$$

Setting $\Delta_2 = \alpha$ and $\Delta_1 = \frac{1}{\eta} - \frac{2G^2}{\alpha}$ ensures the last term on the right hand side is zero. We also have that $\eta \leq \alpha/(4G^2)$ implies that $\Delta_1 \geq \frac{1}{2\eta}$. Rearranging (11) we get that

$$\mathbb{E}_t\|x_{t+1} - x^*\|^2 \leq \frac{1}{1+\eta\alpha}\|x_t - x^*\|^2 + \frac{2\eta^2\sigma^2}{1+\eta\alpha} + \frac{2\eta C^2}{\alpha(1+\eta\alpha)}.$$

Next we verify that our choice of n is large enough so that $\frac{C^2}{\alpha} \leq \eta\sigma^2$. Indeed, this is equivalent to

$$\left(\lambda^n \overline{W}_1(p_0) + \lambda^n \gamma \frac{4}{\alpha G^2} \cdot L \cdot (1 + \|A\|_{\mathrm{op}}) \cdot \frac{\lambda}{(1-\lambda)^2}\right) \leq \frac{\alpha^{1/2}}{L}\eta^{1/2}\sigma$$

which is in turn equivalent to

$$n \geq \log\left(L\frac{\overline{W}_1(p_0) + \gamma\frac{4}{\alpha G^2}L(1+\|A\|_{\mathrm{op}})\frac{\lambda}{(1-\lambda)^2}}{(\alpha\eta)^{1/2}\sigma}\right)\frac{1}{\log(1/\lambda)}.$$

Hence, for our choice of n, we have that

$$\mathbb{E}_t\|x_{t+1} - x^*\|^2 \leq \frac{1}{1+\eta\alpha}\|x_t - x^*\|^2 + \frac{4\eta^2\sigma^2}{1+\eta\alpha}.$$

Which completes the proof.

Numerical Simulations

In this section, we start by describing the SFpark data and experiment set-up. Then we provide additional figures and details for each of the two experiments conducted in the main. Finally, we introduce a synthetic data example which abstracts strategic classification in settings where agents have memory.

SFPark Data Description

In this section, we provide more details on our data cleaning strategies and our model for the SFpark dataset.

Data cleaning. We start by discussing our data cleaning strategy. Of the many features in the dataset, the key ones of interest to us were the street name, district name, total time available (number of parking spots multiplied by number of seconds per hour), total time occupied, and rate. Many of the rates were unavailable in the original dataset, but the rate charged for the day before and day after were. If we encountered a missing rate, we replaced it with the rate before and after, if those rates were equal. We only worked with blocks where we could successfully fill in each of the missing rates. This process can be found in the accompanying code.

Estimating price sensitivity. The model we consider is explained in the main body. To provide more intuition and details, as an example, consider the 600 block of Beach Street for the time window between 1200–1500. The initial distribution, d_0, is sampled from the data at the initial price for parking along the 600 block of Beach ST (Beach ST 600), which in this case is $x_0 = \$3$ per hour. As described in Section , we assume that for an announced price difference of $x = \tilde{x} - x_0$, $\tilde{x}$ is the charged price and x is the variable of optimization. The occupancy follows a distribution of $\zeta + A(\tilde{x} - x_0)$, where ζ follows the same distribution as p_0.

The *price sensitivity* A is a proxy for the price elasticity, in that it provides us a relationship between the change in price and the change in occupancy mapped to a $(0, 1)$ scale. Indeed, recall that price elasticity is a change in the percentage occupancy for a given change in percentage price. Hence, price sensitivity as we have defined it has the same sign as price sensitivity except that it is in the right units of our mathematical abstraction for the problem, and is in this sense a proxy thereof. We compute A by considering the following:

a. The average occupancy for the initial price over every weekday in the beginning of the pilot study until the price is changed.
b. The average occupancy over every weekday in the final week of the last price announcement.

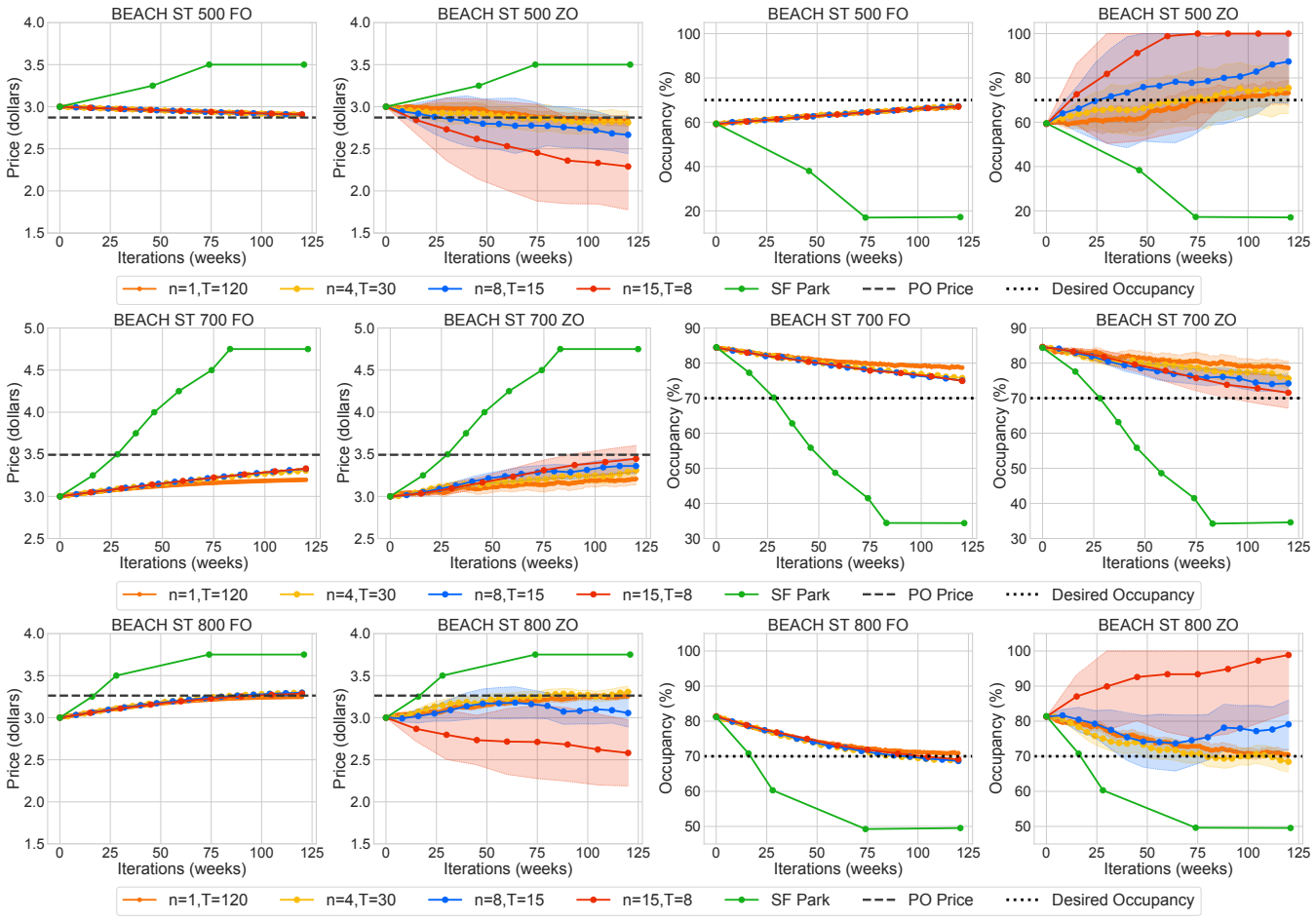

Figure 3: Results of Algorithm 2 (first and third plots of each row) and Algorithm 1 (second and fourth plots of each row) with different (n, T) pairs for the 500, 700 and 800 blocks of Beach ST for time window 1200–1500. Each marker represents a price announcement, and the plots show the prices and 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.

As an example, for the 600 block of Beach ST, the initial price was \$3.00 per hour and the average occupancy before a new price was announced was approximately 60.6%, the final price announced during the pilot study was \$4.25, and the average occupancy for the final week was approximately 41.1%. Therefore, for the 600 block of Beach ST, we estimate that

$$A \approx \frac{0.411 - 0.606}{4.25 - 3} = -0.156,$$

where occupancy percentage is mapped to the $[0, 1]$ scale. It was shown in Pierce and Shoup (2018) that price elasticity is in general a small negative number on average for the SFpark pilot study and experiment. This is consistent with prior studies on price elasticity for on-street parking where information about price and location plays a crucial role (Fiez and Ratliff 2020; **?**). However, for the SFpark pilot study, the price elasticity also depends highly on the block and neighborhood.

Estimating geometric decay parameter λ. We also use this data to estimate the geometric decay rate, λ. As described in Section , when a new rate is posted, the effect on the occupancy is not immediate, and so the geometric decay rate, λ, in this context represents the speed at which this new announced price travels through the population (and consequently affects the parking occupancy). We group the occupancy data by day of week, in order to account for different traffic patterns on different weekdays. We assume that the week before a new price is announced is the fixed point distribution of the previous rate. For example, for the 600 block of Beach ST, a rate of \$3.50 per hour was announced on February 14, 2012, which means that we assumed that the occupancies on February 7–13, 2012 were the fixed point distributions of the previous rate \$3.25. We now fix a day of the week (e.g., Monday), a block (e.g., Beach ST 600), and a time window (e.g., 1200–1500). Suppose the prices $\{x_i\}$ are announced and $\mathcal{D}(x_i)$ represents the fixed point distribution of announcing x_i, where the price x_i is in effect for K_i weeks.

Then, for the k-th week after announcing x_i, we assume that the occupancy is represented by $\lambda^k \mathcal{D}(x_{i-1}) + (1-\lambda^k)\mathcal{D}(x_i)$. For each week k, and for price x_i, the occupancy for the specified day is represented as $z_{i,k}$. To find the value of λ, for the specified day and block, we solve the following optimization problem:

$$\underset{\lambda \in [0,1]}{\text{minimize}} \quad \sum_i \sum_{k=1}^{K_i} (\lambda^k \mathcal{D}(x_{i-1}) + (1-\lambda^k)\mathcal{D}(x_i) - z_{i,k})^2.$$

We perform projected gradient descent to solve this problem. For the final value of λ that we use for the specified block, we average the estimated values of delta for each day.

Comparing Performative Optimum to SFpark

Here, we provide experiments for other blocks on Beach Street (beyond just the 600 block in Section). Each row in Figure 3 shows prices and corresponding occupancies for Algorithm 2 and Algorithm 1 for the 500, 700, and 800 blocks of Beach ST, respectively. In each instance, we make similar observations to those in Section for the 600 block on Beach ST, namely, that SFpark consistently overshot the price to reach the target occupancy, and that the choice of $n = 8$ is reasonable, in that a time period of 8 weeks is sufficient for the population to equilibriate before announcing a new price.

An interesting observation from Figure 3 comes from the fact that the 500 block of Beach ST has a price sensitivity of $A \approx -0.844$, and the 800 block of Beach ST has a price sensitivity of $A \approx -0.424$. Since both of these values have large magnitudes, we observe that for a small price reduction, the estimated occupancy increases to 100%. Therefore, for blocks where the magnitude of the price sensitivity is large, our experiments suggest using a smaller choice of n, and consequently a larger choice of T, in order to reduce the variance for the price announcements to prevent having large fluctuations in occupancy. All four of the blocks on Beach Street have very similar estimated λ values. Table 1 indicates that each block adjusts to new

Beach ST Block Number	(estimated) $\approx \lambda$ value
500	0.993
600	0.959
700	0.993
800	0.984

Table 1: Estimated decay rate λ for each block along Beach ST.

price announcements at similar rates. This makes sense given that each of the blocks are on the same street all next to each other, and located near similar landmarks.

Redistributing Parking Demand

In this appendix subsection, we describe the details for the experiment on redistributing parking demand. The study includes the four connected blocks Hawthorne ST 0, Hawthorne ST 100, Folsom ST 500, and Folsom ST 600 because the blocks are adjacent to one another as shown in Figure 2. Thus, we wanted to investigate whether price changes would redistribute the traffic such that each block had an occupancy closer to the target of 70%. An interesting note is that while Folsom ST 500 and Folsom ST 600 both have negative price sensitivity values of of $A \approx -0.399$ and $A \approx -0.284$ respectively, Hawthorne ST 0 and Hawthorne ST 100 have positive price sensitivity values of $A \approx 0.454$ and $A \approx 0.044$ respectively. Since Hawthorne ST has a very high initial average occupancy, SFpark should consider decreasing prices on this street in order to shift demand to the nearby streets. This is exactly what we see done by both Algorithms 1 and 2 so that both streets are closer to the target occupancy. Although the price sensitivity is very different for these blocks, the estimated λ values are very similar. Hawthorne ST 0 has $\lambda \approx 0.853$, Hawthorne ST 100 has $\lambda \approx 0.979$, Folsom ST 500 has $\lambda \approx 0.996$, and Folsom ST 600 has $\lambda \approx 0.793$, so each block adjusts to new price announcements at similar rates.

Synthetic Data: Strategic Classification in Dynamic Environments

In this appendix subsection, we apply our algorithm to a synthetic strategic classification problem—which was considered in the dynamic setting in Brown, Hod, and Kalemaj (2020) and in the static setting in Drusvyatskiy and Xiao (2020); Miller, Perdomo, and Zrnic (2021); Perdomo et al. (2020), e.g.—where there is memory in the agent population. For simplicity (and to support visualization of the classifier performance), each data point contains a feature vector, $\phi_i \in \mathbb{R}^2$, and a corresponding label, $y_i \in \{-1, 1\}$ where $i \in \{1, \ldots, m\}$ and m is the number of strategic users. The loss incurred by the institution is given by an ℓ_2-regularized logistic loss:

$$\frac{1}{2} \sum_{i=1}^{m} -y_i \langle x, \phi_i \rangle + \log(1 + \exp(\langle x, \phi_i \rangle)) + \frac{\nu}{2} \|x\|^2,$$

where we set $m = 1000$. The agents are non-strategic (meaning they do not perturb their true feature vector $\bar{\phi}_i$) if they have label $y_i = 1$, and otherwise 'best respond' to the announced classifier according to the model

$$\phi_i = \arg\max_w -\langle x, w\rangle - \frac{1}{2\tilde{\epsilon}}\|w - \bar{\phi}_i\|^2 = \bar{\phi}_i - \tilde{\epsilon}x.$$

We take $\tilde{\epsilon} = 0.1$, but the observations we make hold more generally with the exception of very large magnitude perturbations for which the problem (even in the static setting) becomes untenable. We randomly select a subset of the two features to treat as strategic. We also randomly generate a ground truth data set by drawing $m \times 2$ samples from a normal distribution, drawing the ground truth ϕ_{gt} from a (2 dimensional) normal distribution and then assigning labels according to

$$y_i = (\text{sign}(\phi_i^\top \phi_{\text{gt}} + 0.1v) + 1)/2,\ v \sim \mathcal{N}(0, 1).$$

Specifically, agents are allowed to perturb in the x_1 direction as can be seen in Figure 4. Moreover, we take the initial data distribution p_0 to be far from the base distribution for users' true preferences $\bar{\phi}_i$ even with performative effects; specifically, p_0 is a Gaussian distribution with a mean of 1.0 and scale (standard deviation) of 45. More details on the implementation can be found in the accompanying code.

We divide the data into a training and test set with a $(2/3)$–$(1/3)$ split. We set the regularization parameter to $\nu = 1/m_{\text{train}}$ where m_{train} is the size of the training data set. The inner product can be interpreted as the utility of the agent and the norm difference as the cost of manipulation. We present results for a modest value of $n = 20$; similar or lower values are consistent with our observations and as our theory suggests, as $n \to \infty$, the solution obtained by Algorithm 2 approaches the performatively optimal solution.

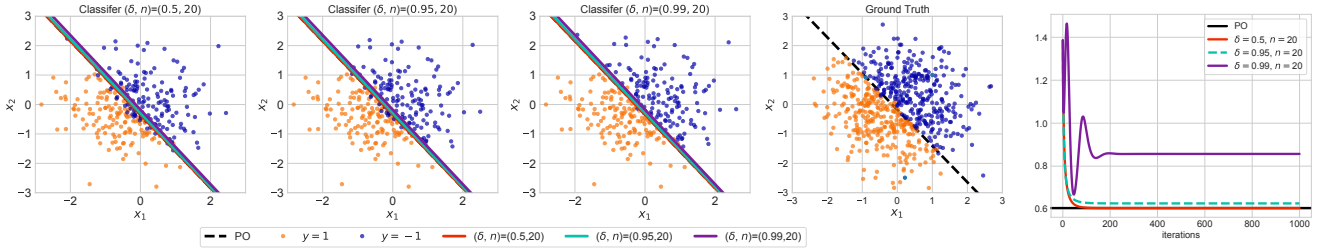

Figure 4: Classifiers and losses for different values of λ and $n = 20$. In order of appearance from left to right, the first three plots show the learned classifiers with the data at the distribution $\mathcal{D}(x)$ induced by the learned classifier for $(\lambda, n) = (0.5, 20)$, $(\lambda, n) = (0.95, 20)$, $(\lambda, n) = (0.99, 20)$. The fourth plot from the left is the ground truth data distribution without performative effects. The differences in the data distributions are subtle, but one can see that the different learned classifiers evoke different responses from the strategic users. The far right plot shows the losses as a function of iterations.

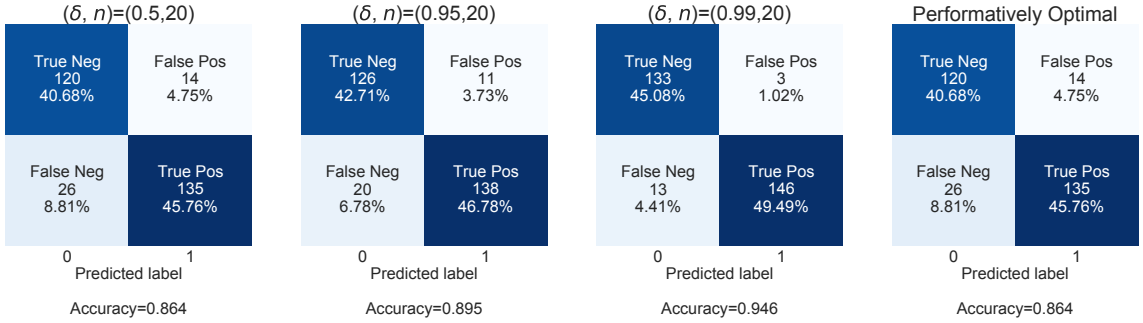

Figure 5: Accuracy of the classifiers (via confusion matrix) learned for the data distribution and setting shown in Figure 4. For this randomly sampled data distribution, λ plays a significant role on the generalization capability (as measured by accuracy on the test set). Surprisingly, accuracy improves as the mixing parameter λ increases (meaning longer time to mix) and this also has an impact on auxiliary but related metrics such as the false positive and false negative rates. This observation depends highly on the data distribution, but exposes interesting directions for future theoretical work on understanding how performative optimality translates to generalization and robustness guarantees.

We explore different values of λ and n—i.e., the mixing parameter of the geometric dynamics and the epoch length of Algorithm 2—on not just convergence but also on accuracy. The observations we report actually lead to a number of interesting

open questions for this field including how performative optimality relates to generalization. We find that depending on the skew of the data distribution and the strength of the perturbation power of the strategic agents—namely, $\tilde{\epsilon}$—that surprisingly, the performatively optimal point may not generalize very well as compared to the solution obtained by Algorithm 2 when the mixing parameter λ is large. The latter has better accuracy as can be seen in Figure 5; the loss value per iteration and the classifiers for different λ values are shown in Figure 4.

In other settings (e.g., with different ground truth data), the solution obtained by Algorithm 2, even with different values of λ and different choices of epoch length n, performs just as well as the performatively optimal solution as depicted in Figure 7, the data for which has original distribution depicted in Figure 6, which also contains the learned classifiers and losses per iteration for different λ values.

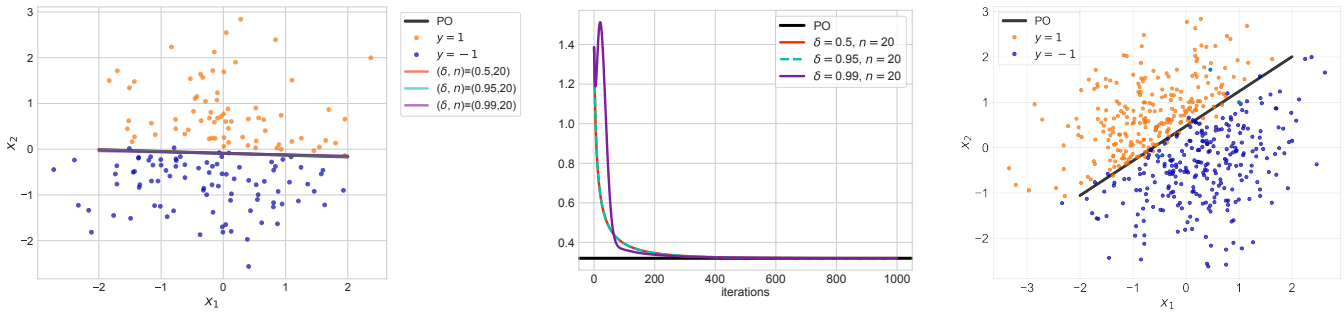

Figure 6: Classifiers and losses for different values of λ and n, for the given original data distribution shown in the far right plot. (**left**) Different classifiers (as a function of λ and n) and the data distribution given the strategic best response at the performatively optimal point. (**center**) Losses for the different (λ, n) pairs as a function of iteration. (**right**) original data distribution and ground truth classifier.

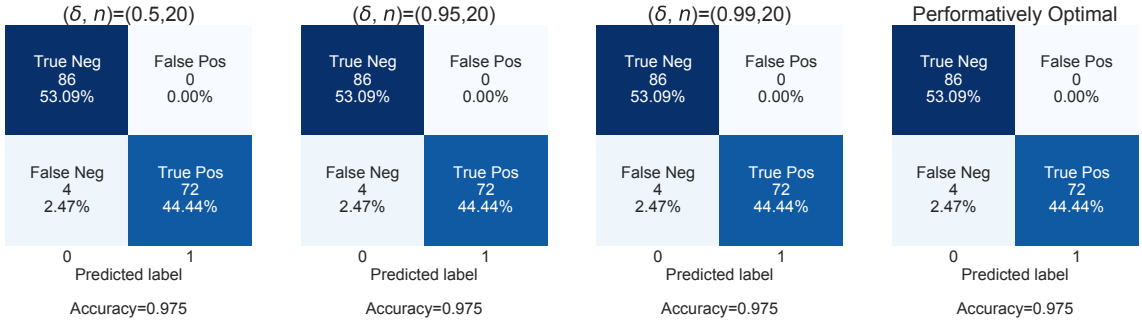

Figure 7: Accuracy of the classifiers (via confusion matrix) learned for the data distribution and setting shown in Figure 6. For this randomly sampled data distribution, the value of λ **does not** play a significant role on the generalization capability (as measured by accuracy on the test set). Accuracy remains the same across the learned classifiers in each setting.

These observations about the generalization performance of the obtained solution under our proposed algorithm (for different values of the geometric process or mixing constant λ) as compared to the (performatively) optimal point, while highly dependent on the underlying data distribution, open up a number of interesting directions for future work on understanding precisely when the optimal point gives good generalization and robustness guarantees.

Semi-Synthetic Data: Strategic Classification in Dynamic Environments

As a point of comparison to the existing literature, we perform additional numerical experiments on a strategic classification simulator from the Kaggle *Give Me Some Credit* dataset discussed in Perdomo et al. (2020) and Brown, Hod, and Kalemaj (2020). In this dataset, each data point contains a feature vector, $\phi \in \mathbb{R}^d$, which represents historical information about an individual, and the label, $y \in \{0, 1\}$, represents whether or not the individual has defaulted on a loan. For more details on the dataset itself, see Appendix B.2 in Perdomo et al. (2020).

Let S be the subset of features that an individual can strategically manipulate. We assume that the best response of every individual to an announced x is given by $\phi^S - \tilde{\varepsilon} x^S$, where we use the notation x^S to be the restriction of x to the subset S and similarly for ϕ^S. The remaining features of the individual stay the same as the original data.

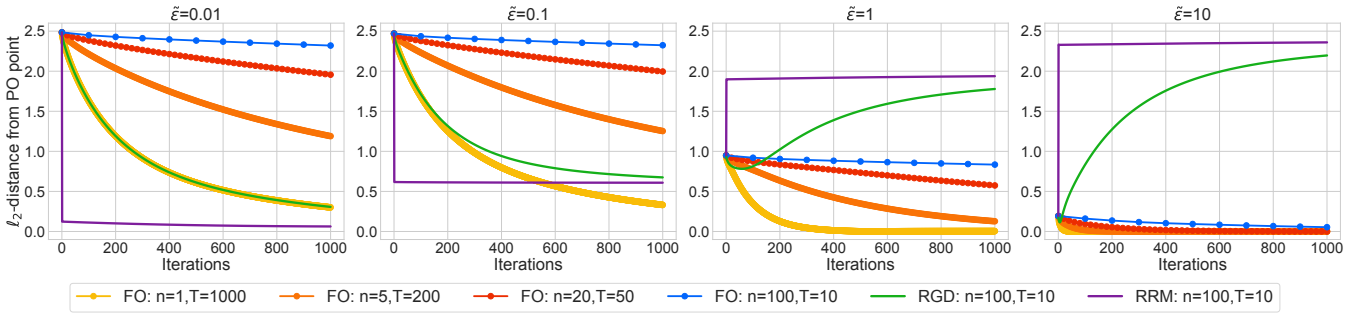

Figure 8: **'Give Me Some Credit' Experiment 1**: Results of Algorithm 2 called with different (n, T) pairs along with standard implementations of repeated risk minimization (RRM) and repeated gradient descent (RGD) wherein the dynamics and classifier are updated at each iteration. Each marker represents a new x announcement, and the plots show the Euclidean distance from the performatively optimal point. Algorithm 2 converges to the performatively optimal point for each value of $\tilde{\varepsilon}$ while RRM and RGD converge to the performatively stable point. The latter may be far from the performatively optimal point for large perturbation values $\tilde{\epsilon}$ as indicated in the plot, going from left to right.

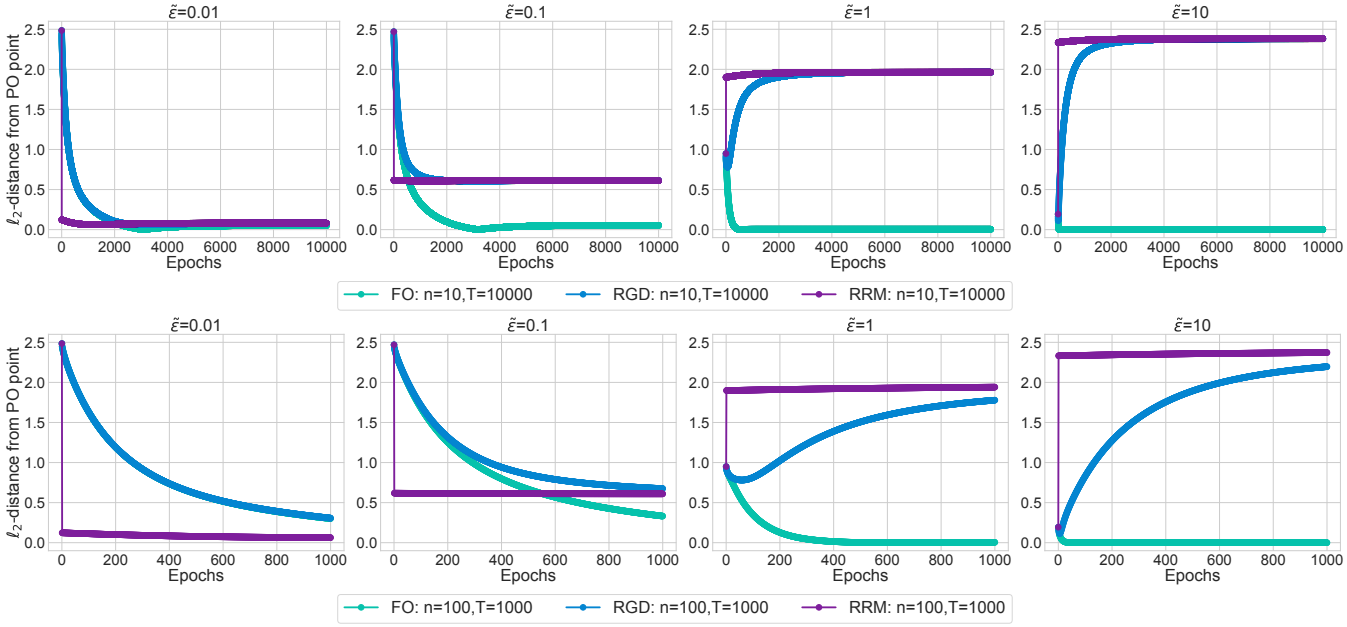

Figure 9: **'Give Me Some Credit' Experiment 2**: Results of Algorithm 2 compared to epoch-based implementations of RRM and RGD—i.e., where in each epoch the dynamics are updated n times with the same classifier deployed—each called with $(n, T) \in \{(10, 1000), (100, 1000)\}$. Each marker represents a new x announcement, and the plots show the Euclidean distance from the performatively optimal point.

We conduct two sets of experiments. In the first set, we compare our algorithm on the total number of iterations—i.e., epochs n multiplied by T—to risk minimization (RRM) (Perdomo et al. 2020; Brown, Hod, and Kalemaj 2020), and repeated gradient descent (RGD) (Perdomo et al. 2020)—implemented for the dynamic environment which was not considered in Perdomo et al. (2020)—both of which, notably update x at every iteration in $[0, nT]$ where as our approach (Algorithm 2) only updates at every n steps in that same interval.

In the second set of experiments, we compare our approach to an epoch based implementation of both RRM and RGD where in these implementations the dynamics are also allowed to "mix" and the decision maker updates only every n steps as in our method. These later experiments are more comparable even though the epoch based implementations of RRM and RGD have not been studied theoretically. For both experiments, we plot the ℓ_2 distance to the optimal point.

Experiment 1: Comparison to Iteration-Based (Classical) RRM and RGD. Figure 8 shows the results of the first set of experiments, for which we have taken $\lambda = 0.9$, which is relatively large meaning that the mixing time for the geometric process is large. Neither RRM nor RGD target the performatively optimal point, but instead the *performatively stable* point, i.e., the

point at which repeated retraining will stabilize. As shown in Figure 8, a performatively stable point (the point RRM was shown to converge to in Brown, Hod, and Kalemaj (2020)) may be far from the peformatively optimal point. Interestingly, we also observe that for small values of $\tilde{\varepsilon}$ (i.e. on the order of 1e-2), the performatively optimal point and the performatively stable point are very close, and so RGD behaves nearly identically to calling Algorithm 2 with $n = 1$. This seems to imply that when performative effects (i.e., size of $\tilde{\varepsilon}$ in this set of experiments) are very low, the naïve strategies of RRM or RGD suffice when trying to find the optimal point. On the other hand, for values of $\tilde{\varepsilon}$ on the order of 1e-1 or larger, RRM and RGD do not converge to the performatively optimal point while Algorithm 2 does, albeit with worse iteration complexity to convergence to the stable point of the respective algorithm.

Experiment 2: Comparison to Epoch-Based RRM and RGD. Figure 9 shows the results of the second set of experiments. As noted above, in this set of experiments, we compare to epoch based implementations of RRM and RGD to Algorithm 2 which is also an epoch-based algorithm, the idea here being that these are more comparable algorithms in a sense. As can be seen in Figure 9, the observations are analogous to the first set of experiments. Epoch-based RRM and RGD converge to the performatively stable point (as defined in (Perdomo et al. 2020) and (Brown, Hod, and Kalemaj 2020), for the dynamic setting). For $\tilde{\varepsilon}$ on the order of 1e-2, the performatively stable point is close to the performatively optimal point (although still not equal to it), and for $\tilde{\varepsilon}$ on the order of 1e-1 or larger, the performatively stable point is considerably farther away from the performatively optimal point. On the other hand, Algorithm 2 converges to the performatively optimal point for all shown values of $\tilde{\epsilon}$, the size of the strategic perturbation.

We note that we did not compare to the zero-th order method since it has different information than both the RRM and RGD and is thus less comparable. We expect the same observations about non-convergence of RRM and RGD for large $\tilde{\varepsilon}$ to persist and Algorithm 1 will converge as the theory predicts, albeit at a much slower rate than Algorithm 2 due to the bandit feedback.