

Stochastic optimization with decision-dependent distributions

Dmitriy Drusvyatskiy*

Lin Xiao†

Abstract

Stochastic optimization problems often involve data distributions that change in reaction to the decision variables. This is the case for example when members of the population respond to a deployed classifier by manipulating their features so as to improve the likelihood of being positively labeled. Recent works on performative prediction have identified an intriguing solution concept for such problems: find the decision that is optimal with respect to the static distribution that the decision induces. Continuing this line of work, we show that in the strongly convex setting, typical stochastic algorithms—originally designed for static problems—can be applied directly for finding such equilibria with little loss in efficiency. The reason is simple to explain: the main consequence of the distributional shift is that it corrupts algorithms with a *bias* that decays linearly with the distance to the solution. Using this perspective, we obtain convergence guarantees for popular algorithms, such as stochastic gradient, clipped gradient, prox-point, and dual averaging methods, along with their accelerated and proximal variants. In realistic applications, deployment of a decision rule is often much more expensive than sampling. We show how to modify the aforementioned algorithms so as to maintain their sample efficiency while performing only logarithmically many deployments.

Keywords: stochastic optimization, distributional shift, Wasserstein distance, performative prediction, stochastic gradient, proximal point method.

1 Introduction

Stochastic optimization plays a central role in statistical sciences and large-scale data-driven computing. The goal of stochastic optimization in these settings is to learn a decision rule (e.g. classifier) from a limited data sample that generalizes well to the entire population. In simplest circumstances, this task amounts to the optimization problem

$$\mathbf{St}(\mathcal{P}) : \quad \min_x \mathbb{E}_{z \sim \mathcal{P}} \ell(x, z) + r(x). \quad (1.1)$$

Here, z encodes the population data, which is assumed to follow some fixed probability distribution \mathcal{P} that is accessible only through sampling. The functions ℓ and r play qualitatively different roles. Typically, $\ell(x, z)$ evaluates the loss of the decision rule parametrized by x on a data point z . In contrast, the function $r: \mathbf{R}^d \rightarrow \mathbf{R} \cup \{\infty\}$ models constraints or promotes some low-dimensional structure in x , such as sparsity or low-rank. We refer to the problem (1.1) as $\mathbf{St}(\mathcal{P})$ to emphasize

*Department of Mathematics, University of Washington, Seattle, WA 98195; www.math.washington.edu/~ddrusv. Research of Drusvyatskiy was supported by the NSF DMS 1651851 and CCF 1740551 awards.

†Facebook AI Research, Seattle, WA 98109. Email: linx@fb.com.

‡Part of this work was done when both authors were affiliated with Microsoft Research.

Algorithms	Iterate update with $z_t \sim \mathcal{D}(x_t)$
Proximal point	$x_{t+1} = \arg \min_x \ell(x, z_t) + r(x) + \frac{1}{2\eta_t} \ x - x_t\ ^2$
Prox-gradient	$x_{t+1} = \text{prox}_{\eta_t r}(x_t - \eta_t \nabla \ell(x_t, z_t))$
Accel. prox-grad.	$\left\{ \begin{array}{l} x_t = \text{prox}_{\eta_t r}(y_{t-1} - \eta_t \nabla \ell(y_{t-1}, z'_t)) \\ y_t = x_t + \beta_t(x_t - x_{t-1}) \end{array} \right\} \quad \text{with } z'_t \sim \mathcal{D}(y_{t-1})$
Clipped gradient	$x_{t+1} = \arg \min_x (\ell(x_t, z_t) + \langle \nabla \ell(x_t, z_t), x - x_t \rangle)^+ + r(x) + \frac{1}{2\eta_t} \ x - x_t\ ^2$
Dual averaging	$x_{t+1} = \arg \min_x \left\langle \frac{1}{t} \sum_{i=1}^t \nabla \ell(x_i, z_i), x \right\rangle + r(x) + \frac{1}{2\eta_t} \ x - x_0\ ^2$

Table 1: Stochastic algorithms with state-dependent distributions.

its dependence on the distribution \mathcal{P} , while “St” abbreviates “static”—a term whose significance will become clear shortly.

Stochastic approximation algorithms are often the methods of choice for $\text{St}(\mathcal{P})$. In every iteration $t = 1, 2, \dots$, such methods draw a fresh sample $z_t \in \mathcal{P}$ and update the iterate x_t using the randomly selected function $\ell(\cdot, z_t)$. Among such algorithms, the stochastic proximal gradient method is the most popular; in each iteration, the method simply takes a step from x_t in the direction opposite to the gradient $\nabla \ell(x_t, z_t)$, followed by a proximal operation of r . Other common stochastic algorithms include proximal point, clipped gradient, and dual averaging methods, along with their inertial variants.

Convergence guarantees of stochastic optimization algorithms crucially rely on the sampling distribution \mathcal{P} being fixed throughout the run of an algorithm. This assumption, however, is violated in applications where the distribution evolves along the iterations. There are two main sources of such distributional shifts. The first is temporal, where the distribution varies in time due to reasons that are independent of the iterates x_t . This setting has been extensively studied in the machine learning literature; see e.g. [5, 6, 11, 12, 25, 34]. The second common source is due to a feedback mechanism, wherein the distribution generating the data in iteration t may depend on, or react to, the current “state” x_t . For example, deployment of a classifier by a learning system, when made public, often causes the population to adapt their attributes in order to increase the likelihood of being positively labeled—a process called “gaming”. Even when the population is agnostic to the classifier, the decisions made by the learning system (e.g. loan approval) may inadvertently alter the profile of the population (e.g. credit score). The goal of the learning system therefore is to find a classifier that generalizes well under the response distribution. Recent research in strategic classification [7, 13, 16, 28] and performative prediction [39, 45] has highlighted the prevalence of this phenomenon.

1.1 Problem setting

The focus of this work is stochastic optimization under decision-dependent distributions. Our approach to such problems builds on the framework of “performative prediction” proposed in [39, 45]. Namely, we consider optimization problems of the form

$$\min_x \mathbb{E}_{z \sim \mathcal{D}(x)} \ell(x, z) + r(x), \quad (1.2)$$

where $\mathcal{D}(x)$ is a distribution indexed by the decision variable $x \in \mathbf{R}^d$. In contrast to $\mathbf{St}(\mathcal{P})$ in (1.1), the data distribution is now decision-dependent. Thus, the quality of a decision rule x is judged by its performance according to the induced distribution $\mathcal{D}(x)$. A direct solution of (1.2) is out of reach in general, even for convex loss functions. Nonetheless, an appealing and widely used heuristic for such problems is to apply standard stochastic optimization algorithms to the static problem $\mathbf{St}(\mathcal{D}(x_t))$ for one or more iterations, then update the distribution $\mathcal{D}(x_{t+1})$ based on the generated iterates, and repeat. Motivated by the recent works [39, 45], we ask the following question:

What can one expect from standard stochastic algorithms (Table 1) when the sampling distribution $\mathcal{D}(x_t)$ used in iteration t depends on the iterate x_t ?

The answer we present relies on a certain distinguished point \bar{x} , highlighted in [39, 45]. A point \bar{x} is at *equilibrium* for the family of distributions $\mathcal{D}(x)$ if \bar{x} solves the static problem $\mathbf{St}(\mathcal{P})$ with $\mathcal{P} = \mathcal{D}(\bar{x})$. Thus \bar{x} is at equilibrium if it solves the static problem that the distribution $\mathcal{D}(\bar{x})$ induces. Equilibrium points are sure to exist under mild continuity and convexity assumptions [45]. Our main contribution can be summarized as follows:

Under mild conditions, stochastic optimization algorithms that sample according to a state-dependent distribution $\mathcal{D}(x_t)$ can be viewed as inexact analogues of the same algorithms applied to the static problem $\mathbf{St}(\mathcal{D}(\bar{x}))$. The inexactness manifests in a bias/error that decays linearly with the distance to \bar{x} .

Thus, stochastic optimization algorithms under decision-dependent distributions automatically search for the equilibrium point \bar{x} by implicitly solving the problem $\mathbf{St}(\mathcal{D}(\bar{x}))$. The strength of the distribution’s dependence on the state—measured by the Lipschitz constant of the map $\mathcal{D}(\cdot)$ —directly impacts the decay rate in the bias/error. We show therefore that if the state dependence is sufficiently weak, standard stochastic algorithms (Table 1) exhibit the same efficiency estimates as if they were directly applied to the static problem $\mathbf{St}(\mathcal{D}(\bar{x}))$ —an a priori impossible task, since the distribution $\mathcal{D}(\bar{x})$ is unknown and inaccessible.

1.2 Application: strategic classification

An important application arena for the developed techniques is the framework of *strategic classification* [28, 40]. This problem class can model a variety of settings with state-dependent distributions, such as fraud detection, traffic prediction, spam filtering, and service recommendations. See [55] and [45, Appendix B] for detailed applications.

Strategic classification is a two player game between an “institution” that deploys a classifier and a population of “agents” who can adapt their features in response in order to increase their likelihood of being positively classified. The game proceeds as an iterative process where the institution and the population take turns to adjust the classifier and features respectively. Specifically, let $z = (a, b)$ denote the features-label pairs of the population that follow a base distribution $z \sim \mathcal{P}$. The institution begins the game by deploying a classifier h_x , parametrized by $x \in \mathbf{R}^d$, and learned from data sampled from \mathcal{P} . Each agent responds to the classifier h_x by greedily modifying their features a to increase their chance of being favorably labeled:

$$\Delta(h_x, a) := \arg \max_{a'} \{u(h_x, a') - c(a, a')\}. \quad (1.3)$$

Here $u(\cdot, \cdot)$ is some utility function and $c(\cdot, \cdot)$ is the cost of altering the features. Thus the samples available to the institution, $(\Delta(h_x, a), b)$, in the next stage of the game, follow a distribution that

depends on x , which we denote by $(\Delta, b) \sim \mathcal{D}(x)$. The goal of the institution is to find the decision variable x that minimizes the classification error with respect to the response distribution:

$$\Pr_{(\Delta, b) \sim \mathcal{D}(x)} [h_x(\Delta) \neq b] = \mathbb{E}_{(\Delta, b) \sim \mathcal{D}(x)} [\mathbf{1}(h_x(\Delta) \neq b)].$$

In practice, one often replaces the 0/1 loss $\mathbf{1}(\cdot)$ with a convex surrogate $\ell(x, z)$, such as the logistic loss, to facilitate large-scale optimization. Thus the problem of strategic classification is an instance of (1.2) under a specific family of distributions $\mathcal{D}(x)$.

In practice, the agents are unlikely to actually play the best-response solutions when modifying their features or that the utility and cost functions are common to all agents. Moreover, in some cases, the dependence of the distribution on the state can be passive; for example, when a bank uses a classifier to approve loan applications, the credit scores of the population are automatically impacted for downstream tasks, even in absence of feature manipulation. In line with the recent works [39, 45], we do not restrict to the model of strategic classification, and instead only require the distribution map $\mathcal{D}(\cdot)$ to have a relatively small Lipschitz constant.

The nature of the dynamics of interaction between the institution and the agents may require different forms of algorithms. For applications such as credit assessment and traffic management, the institution cannot react to each individual changes made by the agents, but can refresh the decision policy periodically after observing sufficiently many new samples and carrying out an (extensive) offline optimization process. On the other hand, applications such as spam filtering and online fraud detection may require immediate update of decision rules whenever some classification errors are made, which correspond to online or streaming stochastic algorithms. We provide analysis of both types of algorithms.

1.3 Related work

Our work is closely related to a number of research themes in optimization, statistics, and machine learning. We now highlight these relationships.

Performative prediction and distributional shift. The two seminal papers on performative prediction [39, 45] motivate and guide much of our work. In particular, the problem setting and assumptions we use (Section 3) are identical to that of [39, 45]. “Performative prediction” is an evocative name for the problem class in machine learning settings since it nicely contrasts the problem class with supervised learning. To stay consistent with the stochastic optimization literature, however, we do not use this terminology here and instead refer to the qualifier “decision-dependent distributions” when needed.

The earlier paper [45] introduces the notion of an equilibrium point (therein called “performatively stable”) and identified regimes in which retraining and gradient descent algorithms converge linearly. The follow up paper [39], in turn, analyzes two variants of the projected stochastic gradient method (called greedy and lazy), and establishes convergence of the last iterate to the equilibrium. Our current work complements [39, 45], aiming to provide a systematic and transparent treatment, based on controlling bias/error. The convergence guarantees we develop apply to a wide class of stochastic algorithms, including those given in Table 1, along with their accelerated and proximal variants. Comments and comparisons with [39, 45] appear throughout the text. Aside from performative prediction, there is a long history of problems with distributional shift in machine learning, whether due to time drift (e.g. [5, 6, 11, 12, 25, 34]) or deployment of classifiers (e.g. [7, 13, 16, 28]). We believe that the techniques developed here may be useful in these contexts as well.

Stochastic programming. Stochastic optimization problems with decision-dependent uncertainties have appeared in the classical stochastic programming literature, such as [1, 20, 31, 48, 52]. We

refer the reader to the recent paper [30], which discusses taxonomy and various models of decision dependent uncertainties. An important theme of these works is to utilize structural assumptions on how the decision variables impact the distributions. Consequently, these works sharply deviate from the framework explored in [39, 45] and from our paper. Time-varying problems have also been studied under the title “nonstationary optimization problems” in, e.g., [22, 24], where it is assumed that the time varying functions converges to a limit but there is no explicit assumption on decision or state-feedback.

Convex stochastic optimization. Many of the techniques used here are rooted in convex optimization; recent monographs on the subject include [9, 14, 44]. The arguments we present are most closely related to the literature on (accelerated) stochastic gradient methods [26, 35], dual averaging [43, 53], and model-based minimization [3, 17].

Online convex optimization. Online convex optimization is a more general framework than stochastic convex optimization in the sense that the loss function at each iteration need not follow any probability distribution and can even be adversarial. On the other hand, online convex optimization requires stronger assumptions such as bounded domain and gradients; see, e.g., [29, 50, 54]. Under these conditions, any online convex optimization algorithm can be applied to static stochastic optimization problems. Similarly, we show by a simple reduction that any online algorithm can be applied to stochastic optimization problems with state-dependent distributions.

Error and bias in gradient oracles. Biased stochastic gradients play a central role in this work. There is a vast literature on errors/bias in gradient computations. For example, [18, 21, 51] analyze first-order methods under inexact oracles and discuss convergence rates. The bias that appears in the current work is fundamentally different, however, in that it decays linearly with the distance to the solution. The closest error model we are aware of is based on relative errors in the gradient (e.g. [10, Section 1.2.1] and [2]). The bias we encounter is more restrictive still and therefore facilitates stronger guarantees.

1.4 Outline of the paper

Section 2 presents two fundamental lemmas that characterize the sensitivity of the expected loss function and its gradient to arbitrary distributional shift. Section 3 formalizes the assumptions that we will use throughout the paper, in particular emphasizing Lipschitz continuity of the distributions relative to variations in the decision variables. Section 4 outlines the main results and previews technical contributions—formally developed in Sections 5-9.

Notation. Throughout, we consider a Euclidean space, denoted for simplicity as \mathbf{R}^d . The symbol $\langle \cdot, \cdot \rangle$ will denote the inner product in \mathbf{R}^d , while $\|x\| = \sqrt{\langle x, x \rangle}$ will denote the induced norm. We equip \mathbf{R}^d with the Borel σ -algebra, throughout. The *proximal map* of any function $f: \mathbf{R}^d \rightarrow \mathbf{R} \cup \{\infty\}$ is defined as

$$\text{prox}_{\eta f}(x) = \arg \min_y \left\{ f(y) + \frac{1}{2\eta} \|y - x\|^2 \right\},$$

where $\eta > 0$ is an arbitrary constant.

We will be interested in random variables taking values in a metric space. Therefore, throughout the paper, we fix a metric space Z with metric $d(\cdot, \cdot)$ and equip Z with the Borel σ -algebra. The symbol \mathbb{P} will denote the set of Radon probability measures on Z with a finite first moment $\mathbb{E}_{z \sim \mathcal{P}}[d(z, z_0)] < \infty$ for some $z_0 \in Z$. We measure the deviation between two measures $\mu, \nu \in \mathbb{P}$ using the Wasserstein-1 distance:

$$W_1(\mu, \nu) = \sup_{g \in \text{Lip}_1} \left\{ \mathbb{E}_{X \sim \mu}[g(X)] - \mathbb{E}_{Y \sim \nu}[g(Y)] \right\}, \quad (1.4)$$

where Lip_1 denotes the set of 1-Lipschitz continuous functions $g: Z \rightarrow \mathbf{R}$. The equivalence of this definition with the description of $W_1(\mu, \nu)$ using couplings is the Kantorovich-Rubinstein duality theorem [33].

Throughout the paper, we fix an arbitrary function $r: \mathbf{R}^d \rightarrow \mathbf{R} \cup \{\infty\}$ and a measurable loss function $\ell: \mathbf{R}^d \times Z \rightarrow \mathbf{R}$. The symbol $\nabla \ell(x, z)$ will always refer to the gradient of $\ell(x, z)$ in the variable x . Throughout the paper, we impose the following assumption.

Assumption 1 (Smoothness). *The loss $\ell(x, z)$ is C^1 -smooth in x for all $z \in Z$ and the map $z \mapsto \nabla \ell(x, z)$ is β -Lipschitz continuous for any $x \in \mathbf{R}^d$.*

An important consequence (following from the dominated convergence theorem) is that for any measure $\mu \in \mathbb{P}$, the expected loss $\mathbb{E}_{z \sim \mu} \ell(x, z)$ is differentiable in x with gradient $\mathbb{E}_{z \sim \mu} [\nabla \ell(x, z)]$.

Lemma 1.1 (Expected gradient). *Fix a measure $\mu \in \mathbb{P}$ such that the loss $\ell(x, \cdot)$ is μ -integrable for every $x \in \mathbf{R}^d$. Then the expected loss $\mathbb{E}_{z \sim \mu} \ell(x, z)$ is differentiable in x with gradient $\mathbb{E}_{z \sim \mu} [\nabla \ell(x, z)]$.*

Proof. The definition of \mathbb{P} yields a point $z_0 \in Z$ satisfying $\mathbb{E}_{z \sim \mu} d(z, z_0) < \infty$. Then for any x and $z \in Z$, Assumption 1 implies $\|\nabla \ell(x, z)\| \leq \|\nabla \ell(x, z_0)\| + \beta \cdot d(z, z_0)$. Since $\nabla \ell(\cdot, z_0)$ is continuous, the norm $\|\nabla \ell(\cdot, z_0)\|$ is bounded on any compact set. An application of the dominated convergence theorem (e.g., [23, Theorem 2.14]) completes the proof. \square

2 Sensitivity to distributional shift

Optimization algorithms that rely on state-dependent sampling, in essence, perform updates on a sequence of static problems that slowly vary along the iterations. An appealing strategy for analyzing such algorithms—and the one we follow here—leverages the stability of the problem $\text{St}(\mathcal{P})$ to perturbations in \mathcal{P} . Formalizing this viewpoint, define the expected loss

$$f_\mu(x) := \mathbb{E}_{z \sim \mu} \ell(x, z),$$

for any measure $\mu \in \mathbb{P}$. The main question we aim to answer in this section is how variations in $\mu \in \mathbb{P}$ impact the function f_μ and its gradient ∇f_μ . The two Lemmas 2.1 and 2.2 provide the answer that guides much of our development.

In order to ensure that the expected losses we encounter are well-defined, throughout this section, we fix two measures $\mu, \nu \in \mathbb{P}$ such that for all $x \in \mathbf{R}^d$, the loss $\ell(x, \cdot)$ is integrable with respect to μ and ν . We begin with Lemma 2.1, which shows that the distance $W_1(\mu, \nu)$ uniformly bounds the deviation $\nabla f_\mu(x) - \nabla f_\nu(x)$. This result is well-known and widely used; we provide a short proof for completeness.

Lemma 2.1 (Gradient deviation). *Under Assumption 1, the estimate holds:*

$$\sup_{x \in \mathbf{R}^d} \|\nabla f_\mu(x) - \nabla f_\nu(x)\| \leq \beta \cdot W_1(\mu, \nu). \quad (2.1)$$

Proof. Fix a unit vector $v \in \mathbf{R}^d$ and define the function $g(z) := v^T \nabla \ell(x, z)$. Clearly by Assumption 1, g is β -Lipschitz continuous in z and therefore by (1.4) we deduce

$$v^T (\nabla f_\mu(x) - \nabla f_\nu(x)) = \mathbb{E}_{z \sim \mu} g(z) - \mathbb{E}_{z \sim \nu} g(z) \leq \beta \cdot W_1(\mu, \nu).$$

Taking the supremum over unit vectors v yields the result. \square

Similarly, it is tempting to use the distance $W_1(\mu, \nu)$ to linearly bound the functional error $|f_\mu(x) - f_\nu(x)|$; such an estimate, however, is far too crude for analyzing algorithms. Instead, the key insight is that convergence analysis of algorithms does not rely on function values in absolute terms, but rather on their differences. The following lemma provides a *multiplicative bound* on the error in the function gaps $\Delta_\mu(x, y) := f_\mu(x) - f_\mu(y)$.

Lemma 2.2 (Function gap deviation). *Under Assumption 1, all points $x, y \in \mathbf{R}^d$ satisfy:*

$$\Delta_\mu(x, y) - \Delta_\nu(x, y) \leq \beta \cdot \|y - x\| \cdot W_1(\mu, \nu). \quad (2.2)$$

Proof. Fix two points $x, y \in \mathbf{R}^d$. For any $s \in [0, 1]$, set $x_s := x + s(y - x)$, which leads to $\frac{d}{ds} f_\mu(x_s) = \langle \nabla f_\mu(x_s), y - x \rangle$. The fundamental theorem of calculus allows to write

$$f_\mu(y) - f_\mu(x) = \int_0^1 \langle \nabla f_\mu(x_s), y - x \rangle ds \quad \text{and} \quad f_\nu(y) - f_\nu(x) = \int_0^1 \langle \nabla f_\nu(x_s), y - x \rangle ds.$$

Subtracting the two estimates yields

$$\begin{aligned} |[f_\mu(y) - f_\mu(x)] - [f_\nu(y) - f_\nu(x)]| &= \left| \int_0^1 \langle \nabla f_\mu(x_s) - \nabla f_\nu(x_s), y - x \rangle ds \right| \\ &\leq \|y - x\| \cdot \int_0^1 \|\nabla f_\mu(x_s) - \nabla f_\nu(x_s)\| ds \end{aligned} \quad (2.3)$$

$$\leq \beta \cdot \|y - x\| \cdot W_1(\mu, \nu), \quad (2.4)$$

where (2.3) follows from Cauchy–Schwarz and (2.4) follows from Lemma 2.1. \square

Importantly, the right side of (2.2) is proportional to the *product* $\|y - x\| \cdot W_1(\mu, \nu)$. In particular, if the distance $W_1(\mu, \nu)$ is on the order of $\|y - x\|$ —the setting we will pass to shortly—then the right-side is proportional to the quadratic error $\|y - x\|^2$. Both Lemmas 2.1 and 2.2 will play a central role in the later sections.

The topic of this section is related to the more general literature on sensitivity of optimal value and solution sets of stochastic programming problems, where a variety of distance measures between probability distributions other than the Wasserstein distance are often used; see, e.g., [47] and references therein.

3 Assumptions under state-dependent sampling

We next record the assumptions that we will use to analyze stochastic optimization under state-dependent distributions—the main content of the work. The imposed conditions are identical to those used in the seminal work [45].

Consider a family of probability measures $\mathcal{D}(x) \in \mathbb{P}$ indexed by $x \in \mathbf{R}^d$. It is instructive throughout the discussion to keep in mind the corresponding problem (1.2). The first assumption we make asserts Lipschitz control on the assignment $x \mapsto \mathcal{D}(x)$.

Assumption 2 (Lipschitz distributions). *There exists $\gamma > 0$ satisfying*

$$W_1(\mathcal{D}(x), \mathcal{D}(y)) \leq \gamma \cdot \|x - y\| \quad \text{for all } x, y \in \mathbf{R}^d.$$

As an example, consider the framework of strategic classification in Section 1.2. If the agents employ a linear utility function $u(x, a) = \langle x, a \rangle$ and a quadratic cost for modifying the features $c(a, a') = \frac{1}{2\gamma} \|a - a'\|^2$, then $W_1(\mathcal{D}(x), \mathcal{D}(y)) \leq \gamma \|x - y\|$; see [45, Appendix G].

The Lipschitz condition on the distribution map $\mathcal{D}(\cdot)$, recorded in Assumption 2, quantifies how far the map is away from being constant—the strength of the state-dependence. The guarantees of Lemmas 2.1 and 2.2 become especially potent under Assumption 2. We record them in Corollary 3.1, and will use it often. To simplify the exposition, we introduce the notation

$$f_x(y) := \mathbb{E}_{z \sim \mathcal{D}(x)} \ell(y, z) \quad \text{and} \quad \nabla f_x(y) := \mathbb{E}_{z \sim \mathcal{D}(x)} \nabla \ell(y, z).$$

In particular, for $f_x(y)$ to be well-defined, we suppose throughout that $\ell(y, \cdot)$ is integrable with respect to $\mathcal{D}(x)$ for all x and y . Note that in light of Section 2, we may equivalently write $f_x(y) = f_{\mathcal{D}(x)}(y)$. Observe also that $\nabla f_x(x)$ is the gradient of the function $y \mapsto f_x(y)$ evaluated at $y = x$.

Corollary 3.1 (Gradient and function gap deviations). *Suppose Assumptions 1 and 2 hold. Then for all points $x, y, u, v \in \mathbf{R}^d$ the estimates hold:*

$$\sup_{w \in \mathbf{R}^d} \|\nabla f_x(w) - \nabla f_y(w)\| \leq \gamma\beta \cdot \|x - y\|, \quad (3.1)$$

$$|(f_x(u) - f_x(v)) - (f_y(u) - f_y(v))| \leq \gamma\beta \cdot \|x - y\| \cdot \|u - v\|. \quad (3.2)$$

We will see that a variety of algorithms with state-dependent sampling are implicitly solving a certain static problem that is at “equilibrium”. A formal description of this phenomenon relies on the notion of an equilibrium point from [45].

Definition 3.2 (Equilibrium point). A point $\bar{x} \in \mathbf{R}^d$ is at *equilibrium with respect to* $\mathcal{D}(\cdot)$ if \bar{x} solves the static problem $\text{St}(\mathcal{D}(\bar{x}))$, or equivalently

$$\bar{x} \in \arg \min_x \{f_{\bar{x}}(x) + r(x)\}.$$

Thus a point \bar{x} is at equilibrium with respect to $\mathcal{D}(\cdot)$ if \bar{x} solves the static problem that it entails. In the context of strategic classification, the decision variable \bar{x} is at equilibrium if the institution has no incentive to deploy another classifier based purely on the population’s response to \bar{x} . Equilibrium points are distinct from minimizers of (1.2) in general, though the distance between the two can be bounded under strong convexity assumptions [45, Theorem 4.3].

Observe that equilibrium points are precisely the fixed points of the *repeated minimization* procedure

$$x_{t+1} = \arg \min_x \{f_{x_t}(x) + r(x)\}. \quad (3.3)$$

This algorithm is largely conceptual since it requires access to the entire data set (to evaluate the expectation) in every iteration. Nonetheless, such “retraining heuristics” are ubiquitous in practice for recovering from distributional shifts, regardless of their origin. Repeated minimization will play an important role in later sections.

Equilibrium points are sure to exist under fairly weak assumptions. The following two sufficient conditions were proved in Theorem 3.5 and Proposition 4.1 of [45], respectively.

Proposition 3.3 (Existence of equilibrium points). *The family of distributions $\mathcal{D}(\cdot)$ is sure to admit an equilibrium point under either of the two conditions:*

- (a) *Assumptions 1 and 2 hold and the functions f_x are α -strongly convex for all $x \in \mathbf{R}^d$ with $\frac{\gamma\beta}{\alpha} < 1$. Moreover, in this case, the equilibrium point is unique.*

(b) The loss $\ell(x, z)$ is jointly continuous in (x, z) and convex in x , the distribution map $\mathcal{D}(\cdot)$ is continuous, and the domain of r is compact.

In light of Proposition 3.3, we make the blanket assumption throughout:

The family of distributions $\mathcal{D}(\cdot)$ admits an equilibrium point, denoted by \bar{x} .

The final two ingredients are assumptions on strong convexity and smoothness: we will use three variants of the former and two variants of the latter depending on context.

Assumption 3 (Strong convexity). *The regularizer r is convex and there exists $\alpha > 0$ satisfying one of the following three properties:*

- (a) $f_{\bar{x}}$ is α -strongly convex,
- (b) f_x is α -strongly convex for all $x \in \mathbf{R}^d$,
- (c) the loss function $\ell(\cdot, z)$ is α -strongly convex for all $z \in Z$.

In this case, define the ratio $\rho := \frac{\gamma\beta}{\alpha}$.

Clearly, the implications (c) \Rightarrow (b) \Rightarrow (a) hold in Assumption 3. We will see that the constant $\rho = \frac{\gamma\beta}{\alpha}$ sharply characterizes the regime of convergence of basic stochastic algorithms with state-dependent sampling.

Assumption 4 (Smoothness). *There exists $L > 0$ satisfying one of the following:*

- (a) $\nabla f_{\bar{x}}$ is L -Lipschitz continuous,
- (b) ∇f_x is L -Lipschitz continuous for all $x \in \mathbf{R}^d$.

If in addition either of Assumptions 3(a)-(c) hold, define the condition number $\kappa := \frac{L}{\alpha}$.

We note that the previous work [39, 45] use a common smoothness parameter β with respect to z and x . But in this paper, we distinguish them as β and L respectively, and our results reveal more nuanced trade-offs between different parameters.

3.1 The interesting parameter regime $\rho < 1$

The proof of Proposition 3.3(a), presented in [45, Theorem 3.5], is particularly instructive algorithmically. Namely, observe that repeated minimization (3.3) is a fixed point iteration of the map $S(x) := \arg \min_y \{f_x(y) + r(y)\}$. The authors of [45] show that S is Lipschitz continuous with parameter ρ . Consequently, in the regime $\rho < 1$, the map S is a contraction and has a fixed point, which by definition is at equilibrium relative to $\mathcal{D}(\cdot)$. Conversely, they show in the same paper that repeated minimization can easily diverge if $\rho \geq 1$. In this sense, the parameter regime $\rho < 1$ is the natural setting for analyzing algorithms under state-dependent distributions. In this work, we aim to show that in the setting $\rho < 1$, typical stochastic algorithms (Table 1) implicitly solve the static problem $\text{St}(\mathcal{D}(\bar{x}))$, where \bar{x} is an equilibrium point. We end the section with an illuminating informal argument supporting this claim for the stochastic gradient method.

Suppose that Assumptions 1, 2, 3(a), 4(a) hold and for simplicity $r \equiv 0$. In each iteration, the stochastic gradient method draws a sample $z_t \sim \mathcal{D}(x_t)$ and uses $\nabla \ell(x_t, z_t)$ to advance. Clearly $\nabla \ell(x_t, z_t)$ is an unbiased estimator of $\nabla f_{x_t}(x_t)$ but is biased with respect to the true gradient

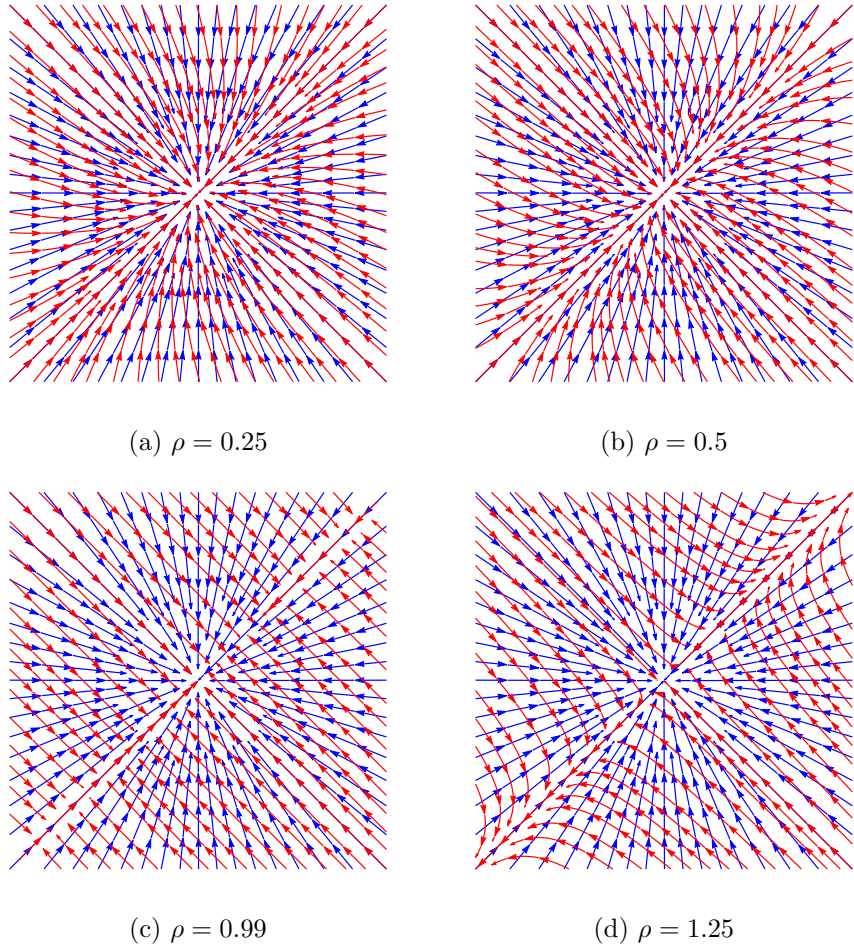


Figure 1: Alignments of vector fields $x \mapsto -\nabla f_x(x)$ (red) and $x \mapsto -\nabla f_{\bar{x}}(x)$ (blue). Each picture corresponds to a different choice of $\rho \in \{0.25, 0.5, 0.99, 1.25\}$.

$\nabla f_{\bar{x}}(x_t)$. The estimate (3.1) directly bounds the bias $\|\nabla f_x(x) - \nabla f_{\bar{x}}(x)\|$, which combined with strong convexity, yields the relative error guarantee:

$$\|\nabla f_x(x) - \nabla f_{\bar{x}}(x)\| \leq \rho \cdot \|\nabla f_{\bar{x}}(x)\|. \quad (3.4)$$

A simple consequence is that in the regime $\rho < 1$, the two vectors $\nabla f_x(x)$ and $\nabla f_{\bar{x}}(x)$ are well-aligned in the sense that they span an angle with cosine $\sqrt{1 - \rho^2}$; see Appendix E for a quick justification. Therefore, in this parameter regime, numerical methods that use unbiased estimators of $\nabla f_x(x)$ are effectively solving the static problem $\text{St}(\mathcal{D}(\bar{x}))$ using biased gradients. A simple numerical example will illustrate this viewpoint.

Example 3.1 (Illustration). We describe now a synthetic two-dimensional example of mean estimation of a moving Gaussian. Specifically, fix a parameter $\rho \geq 0$ and consider the problem (1.2) with losses $\ell(x, z) = \frac{1}{2}\|x - z\|^2$, no regularizer $r = 0$, and the Gaussian distribution $\mathcal{D}(x_1, x_2) = N(\rho(x_2, x_1), I)$. A quick computation yields the expression

$$\nabla f_y(x) = x - \mathbb{E}_{z \sim \mathcal{D}(y)}(z) = \begin{bmatrix} x_1 - \rho y_2 \\ x_2 - \rho y_1 \end{bmatrix}.$$

It is straightforward to see that the origin $\bar{x} = \{0\}$ is the unique equilibrium point of $\mathcal{D}(\cdot)$, provided $\rho^2 \neq 1$. A quick computation shows the equalities $\alpha = \beta = 1$ and $\gamma = \rho$. According to the estimate (3.4), the vector fields $x \mapsto -\nabla f_{\bar{x}}(x)$ and $x \mapsto -\nabla f_x(x)$ span an acute angle pointwise for any $\rho \in (0, 1)$. Figure 1 illustrates that this is indeed the case for $\rho = \{0.25, 0.5, 0.99\}$. Moreover with these parameters, the integral curves of the two vector fields converge to the origin. When $\rho = 1.25$, the two vector fields span an obtuse angle in some regions and integral curves of $x \mapsto -\nabla f_x(x)$ may even diverge.

The intuition highlighted in Example 3.1 can be made precise in the setting $r = 0$. When regularization is present, the estimate (3.4) is not useful because the right-hand side might be uniformly bounded away from zero. Instead, our arguments in Section 7 will use (3.1) directly.

3.2 Measurability issues.

Before outlining our main results, we must address a technical measurability issue. The reader may safely skip this discussion upon first reading. In the rest of the paper, we will be interested in stochastic algorithms that generate iterates x_t based on samples z_t drawn from $\mathcal{D}(x_t)$. Therefore, in order to formally speak of convergence guarantees, we must isolate a single probability space describing the randomness of the entire process.

One simple way to do that is to assume that all the measures $\mathcal{D}(x)$ are absolutely continuous with respect to some probability measure μ on Z and such that the Radon–Nikodym derivative $\frac{d\mathcal{D}(x)}{d\mu}(z)$ is measurable in (x, z) . In particular, for all x and y , we may then write $f_x(y)$ as an expectation over a single measure $f_x(y) = \mathbb{E}_{z \sim \mu} \ell(y, z) \frac{d\mathcal{D}(x)}{d\mu}(z)$. For the stochastic algorithms that we consider, the probability space with respect to which we take expectations is then the countable product $(Z, \mathcal{F}, \mu)^\infty$, where \mathcal{F} is the Borel σ -algebra, equipped with a natural ascending-product filtration. An important example of this set-up is the setting when $Z = \mathbf{R}^d$ and $\mathcal{D}(x)$ admits a density $p_x(z)$ relative to the Lebesgue measure such that the map $(x, z) \mapsto p_x(z)$ is measurable.

Alternative, we may assume that there exists a fixed probability space $(\mathcal{W}, \mathcal{F}, \mu)$ and a measurable map $\theta: \mathbf{R}^d \times \mathcal{W} \rightarrow \mathcal{Z}$ such that $\mathcal{D}(x)$ is the pushforward measure of μ under the map $w \mapsto \theta(x, w)$. Formally this means that for every Borel set $V \subset \mathcal{Z}$, the $\mathcal{D}(x)$ -measure of V is equal to the μ -measure of the set $\{w \in \mathcal{W} : \theta(x, w) \in V\}$. Thus the random variable $z \sim \mathcal{D}(x)$ has the same law as $\theta(x, w)$ where $w \sim \mu$. In this case, we may write $f_x(y) = \mathbb{E}_{w \sim \mu} \ell(y, \theta(x, w))$. As previously, the probability space with respect to which we take expectations when discussing stochastic algorithms is the countable product $(\mathcal{W}, \mathcal{F}, \mu)^\infty$, equipped with an ascending filtration.

4 Outline of the main results

This section outlines the main results of this work. Throughout, the reader should keep in mind the running theme of the paper: a variety of stochastic algorithms under state-dependent distributions are implicitly solving the static problem $\mathbf{St}(\mathcal{D}(\bar{x}))$, where \bar{x} is the equilibrium point. We impose Assumptions 1, 2, 3(c), 4(b) throughout the section. Define the objective function of the static problems $\mathbf{St}(\mathcal{D}(x))$ as

$$\min_y \varphi_x(y) := f_x(y) + r(y).$$

To shorten the notation, we drop the subscript \bar{x} from $\varphi_{\bar{x}}$ and set $\varphi := \varphi_{\bar{x}}$. To better conceptualize the theoretical guarantees, we augment the discussion with numerical illustrations on Example 3.1 and on a strategic classification problem explored in [39, 45]. The implementation details for strategic classification appear in Appendix F.

4.1 Calm and contractive algorithms (Section 5)

We begin the algorithmic development by analyzing a number of conceptual algorithms under state-dependent distributions, the most important being repeated minimization (3.3). More generally, the classical proximal point and gradient methods extend to the state-dependent setting as follows:

$$\begin{aligned} \text{(Proximal point)} \quad & x_{t+1} = \arg \min_x \left\{ f_{x_t}(x) + r(x) + \frac{1}{2\eta} \|x - x_t\|^2 \right\}, \\ \text{(Proximal gradient)} \quad & x_{t+1} = \text{prox}_{\eta r}(x_t - \eta \nabla f_{x_t}(x_t)), \end{aligned}$$

where $\eta > 0$ is a user-specified step size. Thus in each iteration t , the two algorithms simply take a proximal point step and a proximal gradient step, respectively, on the static problem $\text{St}(\mathcal{D}(x_t))$. The proximal point method in the extreme case $\eta = \infty$ coincides with repeated minimization (3.3). Setting notation, let us denote either of these updates as $x_{t+1} = S_{x_t}(x_t)$; more generally, the symbol $S_y(x)$ will denote the update of a point x by an algorithm acting on the static problem $\text{St}(\mathcal{D}(y))$.

The paper [45] showed that when r is the indicator function of a closed convex set and $\rho < 1$, repeated minimization and the projected gradient method converge linearly to the equilibrium point \bar{x} . Section 5 provides an alternative and transparent explanation based on stability to distributional shifts. Namely, linear convergence is a direct consequence of the two independent phenomena:

1. **(Calmness to distribution)** The updates $S_y(x)$ are τ -calm relative to $\mathcal{D}(\bar{x})$, meaning there exists $\tau > 0$ satisfying $\sup_x \|S_y(x) - S_{\bar{x}}(x)\| \leq \tau \cdot W_1(\mathcal{D}(y), \mathcal{D}(\bar{x}))$ for all $y \in \mathbf{R}^d$.
2. **(Contraction at equilibrium)** The updates $S_{\bar{x}}(\cdot)$ are q -contractive on the static problem $\text{St}(\mathcal{D}(\bar{x}))$, meaning $\|S_{\bar{x}}(x) - S_{\bar{x}}(y)\| \leq q \cdot \|x - y\|$ for all $x, y \in \mathbf{R}^d$.

The first property asserts control on how the update $S_y(x)$ varies with respect to the distribution $\mathcal{D}(y)$, while the contraction property asserts that the update is contractive when applied to the static problem induced by $\mathcal{D}(\bar{x})$. It is elementary to see that these two conditions imply that the update $x \mapsto S_x(x)$ contracts towards \bar{x} with ratio $q + \gamma\tau$. By computing the calmness and contraction parameters for different algorithms (in terms of $\rho > 0$), we obtain the following theorem.

Theorem 4.1 (Informal). *Repeated minimization and proximal point methods converge linearly to \bar{x} in the regime $\rho < 1$, while the proximal gradient method converges linearly to \bar{x} in the regime $\rho < \frac{1}{2}$. (A more careful argument in Section 7 shows that the proximal gradient method converges linearly in the optimal regime $\rho < 1$.)*

In theory, repeated minimization and the proximal-point method work within the same optimal parameter regime $\rho \in (0, 1)$. We have seen experimentally, however, that the proximal point method can succeed in a much wider parameter regime whereas repeated minimization can exhibit wild oscillatory behavior. As an example, Figure 2 depicts the performance of both algorithms with different proximal parameters on a problem of strategic classification with $\rho > 1$. It is an interesting question to identify the theoretic justification for this behavior.

The three algorithms described so far (repeated minimization, prox-point, gradient descent) are largely conceptual since they require access to the entire data set in every iteration. Implementable algorithms under state-dependent distributions fall into two categories—those that update the sampling distribution in every step and those that run multiple iterations of stochastic methods on the current static problem before updating the sampling distribution. The former algorithms are called “greedy” and the latter are called “lazy” in [39]. As remarked in [39], lazy algorithms can be interpreted as performing inexact repeated minimization. Such algorithms can be advantageous

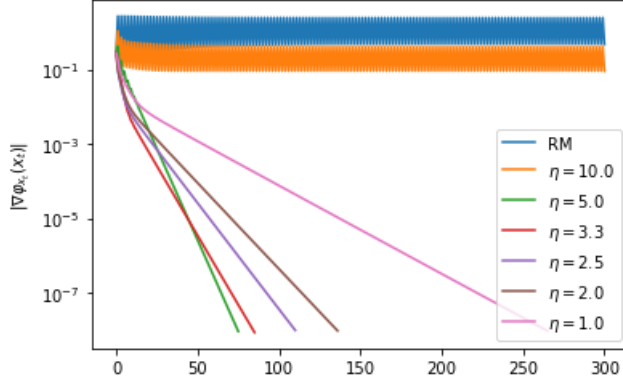


Figure 2: We implement repeated minimization and the proximal point method on a problem of strategic classification detailed in Appendix F (the problem parameters are $n = 200$, $\alpha = 10/n$, $\gamma = 5$.) “RM” refers to repeated minimization while the rest of the curves refer to the proximal-point method with parameter η . Repeated minimization and the prox-point method with $\eta = 10$ exhibit wild oscillatory behavior. The prox-point method with parameters $\eta \in \{1, 2, 2.5, 3.3, 5.0\}$ succeeds at finding the equilibrium point, with a rate that degrades with decreasing $\eta > 0$.

in applications where the action of updating the distribution is much costlier than sampling from the currently available distribution. For example, this is the case in applications where it takes significant amount of time for the population to adjust to a newly unveiled learning rule.

We analyze both types of algorithms in this work. Most of the paper, however, focuses on greedy algorithms, while inexact repeated minimization is deferred to Section 9. Though the two types of methods generate entirely different trajectories, the convergence arguments we present are slight modifications of each other—thereby underscoring the utility of the developed techniques.

4.2 Reduction to online convex optimization (Section 6)

We begin by showing that virtually any algorithm developed for “online convex optimization” can be used to find the equilibrium point \bar{x} in the stochastic setting under state-dependent distributions. The framework of online convex optimization can be interpreted as a repeated game (e.g., [29, 50, 54]). At each iteration t , the player chooses a point x_t from a convex set $K \subset \mathbf{R}^d$, then a convex cost function $\ell_t : K \rightarrow \mathbf{R}$ is revealed and the player incurs the cost $\ell_t(x_t)$. The goal of the player is to minimize the *regret*, defined as the difference between the total cost $\sum_{i=1}^t \ell_i(x_i)$ incurred up to round t and the minimum cost of any fixed decision from hindsight, $\min_{x \in K} \sum_{i=1}^t \ell_i(x)$. Here we consider a regularized version of online convex optimization and define the regret up to round t as

$$R_t := \sum_{i=1}^t (\ell_i(x_i) + r(x_i)) - \min_x \sum_{i=1}^t (\ell_i(x) + r(x)),$$

where r is a convex function that represents the indicator function of the set K or a more general regularization. In this setting, we can use the online proximal gradient method,

$$x_{t+1} = \text{prox}_{\eta_t r}(x_t - \eta_t \nabla \ell_t(x_t)), \quad (4.1)$$

as studied in [19]. Other suitable algorithms include the regularized dual average method [43, 53] and variants of FTRL (Follow-The-Regularized-Leader) method [38]. Under various Lipschitz and

strong convexity assumptions, the regret R_t scales as $\mathcal{O}(\log t)$. We establish the following reduction by leveraging the gap deviation inequality (3.2).

Theorem 4.2 (Informal). *Suppose that we are in the regime $\rho < \frac{1}{2}$. Then any online algorithm with regret $R_t = \mathcal{O}(\log t)$ can be used under state-dependent distributions by declaring $\ell_i(x_i) = \ell(x_i, z_i)$ with $z_i \sim \mathcal{D}(x_i)$ in each iteration. Then the average iterate $\hat{x}_t := \frac{1}{t} \sum_{i=1}^t x_i$ satisfies*

$$\mathbb{E}[\varphi(\hat{x}_t) - \varphi(\bar{x})] \leq \mathcal{O}\left(\frac{\log t}{(1-2\rho)t}\right), \quad \text{for all } t \geq 0.$$

In particular, known regret bounds for the proximal gradient, regularized dual averaging, and FTRL algorithms directly yield a converge rate $\mathcal{O}(\log(t)/(1-2\rho)t)$ under state-dependent sampling. Though the theorem is attractive in its generality, it is far from satisfactory. Indeed, the framework of online convex optimization is much more general than stochastic optimization, since the loss function ℓ_t at each iteration may not follow any probability distribution and can even be adversarial. As a consequence, online convex optimization requires stringent assumptions in order to have meaningful regret analysis, most notably that the encountered gradients of the loss functions and the domain $\text{dom } r$ be bounded. Moreover smoothness of the loss function does not play a significant role. We will see that much finer convergence guarantees hold for stochastic optimization under state-dependent distributions. In particular, bounds on the gradient of the loss function required in online convex optimization will be replaced by a finite variance assumption.

4.3 Stochastic gradient methods (Section 7)

The simplest and most widely used stochastic algorithm is the stochastic gradient method. We begin by investigating its extension under decision-dependent distributions:

$$\text{SG} : \quad \left\{ \begin{array}{l} \text{Sample } z_t \sim \mathcal{D}(x_t) \text{ and set } g_t = \nabla \ell(x_t, z_t), \\ \text{Set } x_{t+1} = \text{prox}_{\eta_t r}(x_t - \eta_t g_t). \end{array} \right\} \quad (4.2)$$

Rather than reducing the method SG to online convex optimization, as mentioned previously, we will analyze it directly under the following finite variance assumption.

Assumption 5 (Finite variance). *There is a constant $\sigma > 0$ satisfying*

$$\mathbb{E}_{z \sim \mathcal{D}(x)} \|\nabla \ell(x, z) - \nabla f_x(x)\|^2 \leq \sigma^2 \quad \forall x \in \text{dom } r.$$

Notice that contrary to the static setting, the vector $g_t = \nabla \ell(x_t, z_t)$ is a biased estimator for the true gradient $\nabla f_{\bar{x}}(x_t)$, where the bias is proportional to $\|x_t - \bar{x}\|$; recall the gradient deviation inequality (3.1). We will prove the following key recursion.

Lemma 4.3 (Key recursion). *Suppose that the step-size sequence satisfies $\eta_t < \frac{1}{2L}$. Then the iterates $\{x_t\}$ generated by the SG Algorithm in (4.2) satisfy*

$$2\eta_t \mathbb{E}[\varphi(x_{t+1}) - \varphi(\bar{x})] \leq (1 - \alpha(1-2\rho)\eta_t + 2\gamma^2\beta^2\eta_t^2) \mathbb{E}\|x_t - \bar{x}\|^2 - \mathbb{E}\|x_{t+1} - \bar{x}\|^2 + 2\eta_t^2\sigma^2.$$

Observe that the contraction factor multiplying $\mathbb{E}\|x_t - \bar{x}\|^2$ for small η_t scales as $1 - \alpha(1-2\rho)\eta_t$. It follows that in the regime $\rho < \frac{1}{2}$, one can drive the gap $\mathbb{E}[\varphi(x_{t+1}) - \varphi(\bar{x})]$ to zero at a controlled rate, with an appropriate choice of η_t . Moreover, a quick argument shows that the parameter regime of convergence becomes larger if we focus on the rate at which the square distance $\|x_{t+1} - \bar{x}\|^2$

decays. Indeed, lower-bounding the left side in Lemma 4.3 using strong convexity and rearranging yields the one-step progress guarantee

$$\mathbb{E}\|x_{t+1} - \bar{x}\|^2 \leq (1 - \alpha(1 - \rho)\eta_t + \frac{1}{2}\eta_t^2\gamma^2\beta^2) \mathbb{E}\|x_t - \bar{x}\|^2 + \eta_t^2\sigma^2.$$

For small η_t , the contraction factor multiplying $\|x_t - \bar{x}\|^2$ roughly scales as $1 - \alpha(1 - \rho)\eta$. Therefore in the regime $\rho < 1$, one can drive the square distance $\mathbb{E}[\|x_{t+1} - \bar{x}\|^2]$ to zero at a controlled rate, with an appropriate choice of η_t .

With Lemma 4.3 at hand, obtaining guarantees with various choices of parameters η_t is standard. The following theorem presents one such guarantee. The first part of the theorem when r is an indicator function of a closed convex set was proved in [39] using a different argument.

Theorem 4.4 (Informal). *In the regime $\rho < 1$, the proximal stochastic gradient method with appropriate parameters η_t will generate a point x satisfying $\mathbb{E}\|x - \bar{x}\|^2 \leq \varepsilon$ using*

$$\mathcal{O}\left(\left(\left(\frac{\rho}{1-\rho}\right)^2 + \frac{\kappa}{1-\rho}\right) \cdot \log\left(\frac{\|x_0 - \bar{x}\|^2}{\varepsilon}\right) + \frac{\sigma^2}{(1-\rho)^2\alpha^2\varepsilon}\right) \quad \text{samples.}$$

Moreover, in the regime $\rho < \frac{1}{2}$, the method will generate x satisfying $\mathbb{E}[\varphi(x) - \varphi(\bar{x})] \leq \varepsilon$ using

$$\mathcal{O}\left(\left(\left(\frac{\rho}{1-2\rho}\right)^2 + \frac{\kappa}{1-2\rho}\right) \cdot \log\left(\frac{\varphi(x_0) - \varphi(\bar{x})}{\varepsilon}\right) + \frac{\sigma^2}{(1-2\rho)\alpha\varepsilon}\right) \quad \text{samples.}$$

In the static setting $\gamma = 0$ (and hence $\rho = 0$), Theorem 4.4 recovers the classical guarantees for the proximal stochastic gradient method [26,27]. Figure 3 illustrates the performance of SG on Example 3.1. We note that the follow-up paper [15] establishes an extension of Theorem 4.4 that holds with high probability under light tail assumptions on the noise.

Accelerated gradient methods, famously introduced by Nesterov [42] and extended to the proximal setting by Beck and Teboulle [8], play a central role in convex optimization. Such methods are best in class for smooth convex optimization. Ghadimi and Lan [27,37] proposed an accelerated method for the stochastic setting, which is best in class for smooth convex stochastic problems. Continuing the theme of the paper, we ask whether acceleration is possible with state-dependent distributions. We show an affirmative answer in the slightly suboptimal regime $\rho \lesssim \kappa^{-1/4}$.

Rather than analyzing the original method of Ghadimi and Lan [27], we focus on the more recent variant of Kulunchakov and Mairal [35]. With decision-dependent distributions, it reads as:

$$\text{ASG : } \left\{ \begin{array}{l} \text{Sample } z_t \sim \mathcal{D}(y_{t-1}) \text{ and set } g_t = \nabla\ell(y_{t-1}, z_t), \\ \text{Set } x_t = \text{prox}_{\eta_t r}(y_{t-1} - \eta_t g_t), \\ \text{Set } y_t = x_t + \frac{1 - \sqrt{\eta\alpha(1-2\rho)}}{1 + \sqrt{\eta\alpha(1-2\rho)}}(x_t - x_{t-1}). \end{array} \right\}. \quad (4.3)$$

In the deterministic and unregularized ($r \equiv 0$) setting, the method reduces to the classical procedure in [41], derived through estimate sequences. The following theorem summarizes the convergence guarantees of the ASG algorithm under state dependent sampling.

Theorem 4.5 (Informal). *In the regime $\rho \lesssim \kappa^{-1/4}$, the stochastic ASG method (4.3) with appropriate parameter choices will generate a point x satisfying $\mathbb{E}[\varphi(x) - \varphi(\bar{x})] \leq \varepsilon$ using*

$$\mathcal{O}\left(\sqrt{\kappa} \cdot \log\left(\frac{\varphi(x_0) - \varphi(\bar{x})}{\varepsilon}\right) + \frac{\sigma^2}{\alpha\varepsilon}\right) \quad \text{samples.}$$

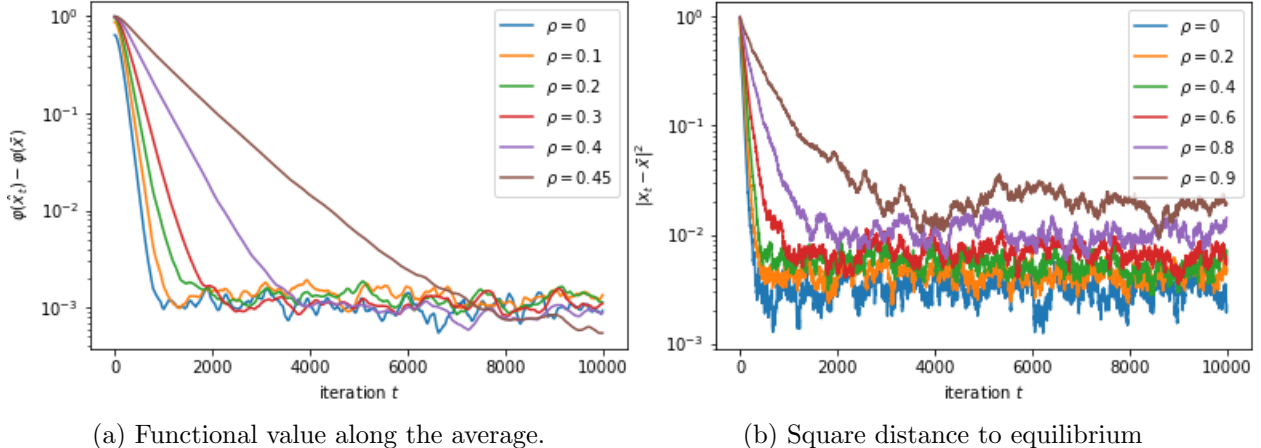


Figure 3: Returning to Example 3.1, we implement a stochastic gradient method with a constant parameter $\eta = 0.01$ averaged over 20 runs for various parameters ρ . Figure 3a shows the function gap $\varphi(\hat{x}_t) - \varphi(\bar{x})$ along the “average iterate” on the equilibrium problem. As the theoretical results (Theorem 7.3) suggest, the gap tends linearly to a noise level controlled by η , with a linear rate that degrades as ρ tends to $1/2$. Figure 3b depicts the square distance of the current iterate $\|x_t - \bar{x}\|^2$, which also tends linearly to a noise level that now depends both on η and on ρ . Indeed, Figure 3b is fully justified by the results of [39], and we include it here only as an illustration.

Notice that the regime when the accelerated method is guaranteed to work $\rho \lesssim \kappa^{-1/4}$ is “sub-optimal” by the small factor $\kappa^{-1/4}$. On the other hand, it is surprising that there is any regime $\rho > 0$ where the accelerated method works at all, since it is well known that accelerated methods suffer from error accumulation [18]. The reason there is no contradiction here is that the gradient bias that we encounter tends to zero linearly as one approaches the solution (2.1). It would be interesting to know whether the extra factor $\kappa^{-1/4}$ is really necessary or is an artifact of the proof.

The acceleration phenomenon is most prominent in the nearly noiseless setting $\sigma \approx 0$. As an illustration, Figure 4 compares the performance of the vanilla gradient method and the accelerated gradient method in the batch setting ($\sigma = 0$) on a problem of strategic classification. Experimentally, we see that acceleration leads to an impressive speedup even in very ill-conditioned settings, thereby suggesting that the parameter regime $\rho \lesssim \kappa^{-1/4}$ in Theorem 4.5 may be loose.¹

Proof technique: gradient deviation as a measure of bias. The results in this section follow from the following transparent geometric reasoning. Recall that the gradient deviation inequality (3.1) shows that contrary to the static setting, the vector $\nabla\ell(x, z)$ with $z \sim \mathcal{D}(x)$ is a biased estimator of the gradient $\nabla f_{\bar{x}}(x)$, with bias scaling as $\|x - \bar{x}\|$. Nonetheless, a quick computation shows that the mean of the estimator $\nabla f_x(x) = \mathbb{E}_{z \sim \mathcal{D}(x)}[\nabla\ell(x, z)]$ furnishes a strong convexity inequality between x and \bar{x} given by

$$f_{\bar{x}}(\bar{x}) \geq f_{\bar{x}}(x) + \langle \nabla f_x(x), \bar{x} - x \rangle + \frac{\alpha(1 - 2\rho)}{2} \|x - \bar{x}\|^2.$$

¹Note, that the orange curve in Figure 4b is highly nonmonotone, as is expected with accelerated methods.

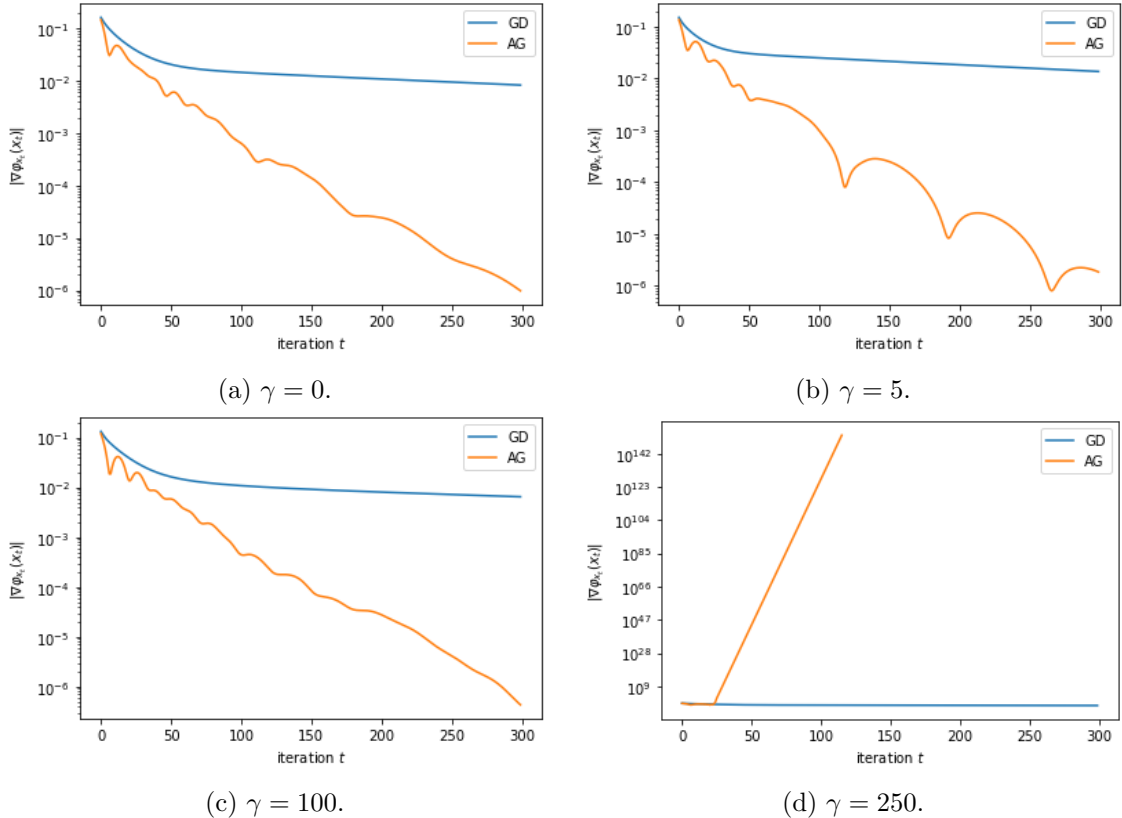


Figure 4: We implement accelerated gradient descent on strategic classification (Appendix F) with parameters $\alpha = 10/n$ and $n = 2000$. In this example $L = 5.05$ and $\beta = 2$ and therefore the inclusion $\rho \in (0, 1)$ holds as long as $\gamma \in (0, \frac{\alpha}{2})$. In each iteration, the method uses the full gradient $\nabla f_{y_t}(y_t)$ and we set $y_t = x_t + \frac{1 - \sqrt{\alpha/L}}{1 + \sqrt{\alpha/L}}(x_t - x_{t-1})$, thereby heuristically ignoring scaling by $1 - 2\rho$. Acceleration leads to an impressive speedup in settings far beyond the threshold $\gamma \in (0, \frac{\alpha}{2})$.

This inequality suffices to establish convergence guarantees for the proximal stochastic gradient method; the reason is simply that strong convexity is used in the classical argument only to compare the function values along the iterates with the minimal value. Perhaps more surprisingly, the accelerated variant of the method can also be understood from this viewpoint. Section 7 analyzes the proximal stochastic gradient method and its accelerated variant under a biased stochastic oracle model. This oracle model is broader than the setting of state-dependent sampling and may be of independent interest.

4.4 Model-based minimization: stochastic proximal point and clipped gradient methods (Section 8)

Though the stochastic gradient method is popular in practice, it has well-documented deficiencies. Notably, the method is highly sensitive to algorithmic parameters, with small misspecifications often drastically degrading performance. Recent works [3, 49] have suggested that algorithms based on tighter models than linear may lead to more robust algorithms. Following [3, 17], we consider a class of stochastic algorithms that proceed as follows. In each iteration t , the methods draw a sample

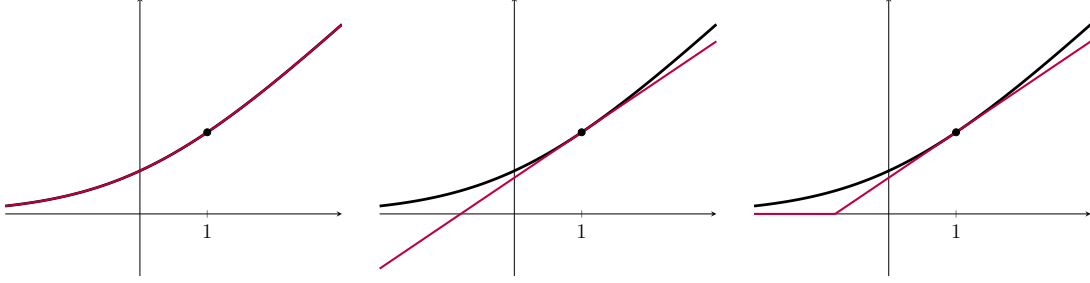


Figure 5: Illustration of the three models for the function $\ell(y) = \ln(1 + e^y)$; black curve depicts the graph of ℓ , the red curves depict the models $\ell_1(y) = \ell(y)$ (proximal point), $\ell_1(y) = \ell(1) + \ell'(1)(y-1)$ (gradient), $\ell_1(y) = \max\{\ell(1) + \ell'(1)(y-1), 0\}$ (clipped gradient).

$z_t \in \mathcal{D}(x_t)$ and approximate the loss function $\ell(\cdot, z_t)$ by a simpler model $\ell_{x_t}(\cdot, z_t)$ formed at the basepoint x_t .² The next iterate x_{t+1} is then the minimizer of the function $\ell_{x_t}(\cdot, z_t) + r + \frac{1}{2\eta_t} \|\cdot - x_t\|^2$. Thus, the model-based algorithm repeats the steps

$$\left\{ \begin{array}{l} \text{Sample } z_t \sim \mathcal{D}(x_t) \\ \text{Set } x_{t+1} = \arg \min_y \left\{ \ell_{x_t}(y, z_t) + r(y) + \frac{1}{2\eta_t} \|y - x_t\|^2 \right\} \end{array} \right\} \quad (\text{MBA})$$

For example, the stochastic proximal gradient method (4.2) uses the linear model $\ell_x(y, z) = \ell(x, z) + \langle \nabla \ell(x, z), y - x \rangle$, while the stochastic proximal point method uses the loss function itself $\ell_x(y, z) = \ell(y, z)$. Often, the proximal point subproblem can be solved in closed form since it depends only on a single data point. Yet another interesting algorithm is the clipped gradient method, which uses the truncated models $\ell_x(y, z) = \max\{\ell(x, z) + \langle \nabla \ell(x, z), y - x \rangle, 0\}$ under the assumption that the losses are nonnegative. See Fig. 5 for an illustration. Tighter models often lead to better performing algorithms.

Section 8 presents convergence guarantees for the entire class of model-based algorithms. It will be important for the clipped stochastic gradient method to assume that r is μ -strongly convex for some $\mu \geq 0$. The efficiency guarantees specialized for the stochastic proximal gradient, proximal point, and clipped gradient methods read as follows.

Theorem 4.6 (Informal). *Define $\tilde{\alpha} := \alpha + \mu$ for stochastic proximal gradient and proximal point methods and set $\tilde{\alpha} := \mu$ for the clipped gradient method; in addition, set $\tilde{\rho} = \frac{\gamma\beta}{\tilde{\alpha}}$. In the regime $\tilde{\rho} < 1$, the three methods with appropriate parameters η_t will find x satisfying $\mathbb{E}\|x - \bar{x}\|^2 \leq \varepsilon$ using*

$$\mathcal{O} \left(\frac{L + \tilde{\alpha}}{\tilde{\alpha}(1 - \tilde{\rho})} \cdot \log \left(\frac{\|x_0 - \bar{x}\|^2}{\varepsilon} \right) + \frac{\sigma^2}{\tilde{\alpha}^2(1 - \tilde{\rho})^2 \varepsilon} \right) \quad \text{samples.}$$

Moreover, in the regime $\tilde{\rho} < \frac{1}{2}$, the methods will generate x satisfying $\mathbb{E}[\varphi(x) - \varphi(\bar{x})] \leq \varepsilon$ using

$$\mathcal{O} \left(\frac{L + \tilde{\alpha}}{\tilde{\alpha}(1 - 2\tilde{\rho})} \cdot \log \left(\frac{\varphi(x_0) - \varphi(\bar{x})}{\varepsilon} \right) + \frac{\sigma^2}{\tilde{\alpha}(1 - 2\tilde{\rho})\varepsilon} \right) \quad \text{samples.}$$

As an illustration, Figure 6 illustrates the performance of the stochastic gradient, clipped gradient, and proximal point methods on a problem of strategic classification. We use a sublinearly

²Formally, one must assume that the assignment $(x, y, z) \mapsto \ell_x(y, z)$ is measurable.

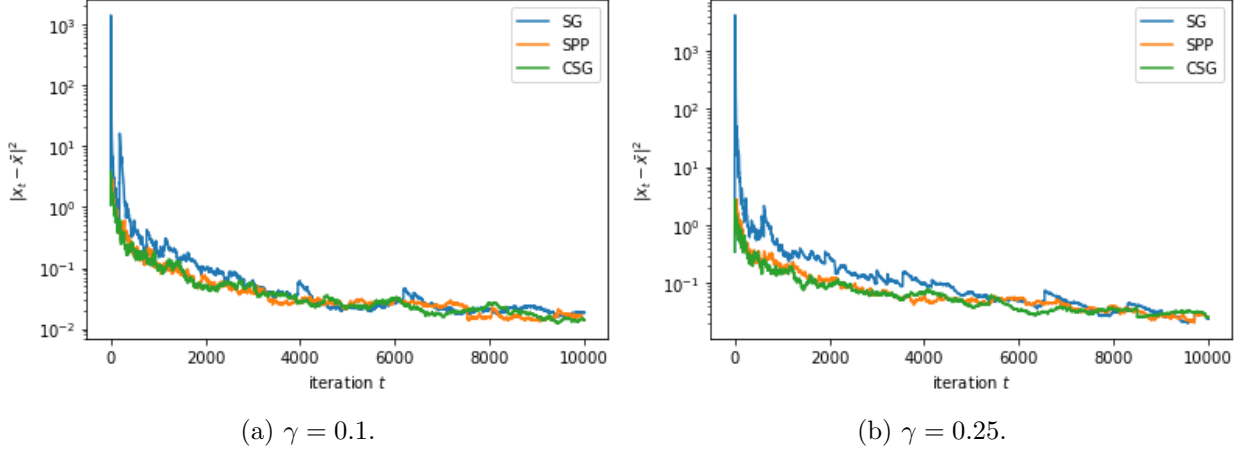


Figure 6: Stochastic gradient (SG), clipped gradient (CSG), proximal point (SPP) methods on strategic classification (Appendix F) with parameters $\alpha = 100/n$, $n = 2000$, and $\eta_t = \frac{2}{\alpha(t+1)}$.

decaying stepsize-sequence $\eta_t = \frac{2}{\alpha(t+1)}$. The three methods perform similarly asymptotically. In the initial stage, however, the subgradient method generates iterates that are highly suboptimal due to a large initial step-size. Consequently, the clipped gradient and proximal-point methods may be preferable. The drastically different performance in the early stages between the subgradient method and the clipped gradient/proximal point methods in the static setting was investigated in [3]. Similar guarantees likely extend to the setting of decision-dependent distributions, though we do not pursue this line of work here.

Proof technique: function gap deviation (3.2) & Lyapunov analysis. The proofs of the outlined results rely on the interplay between the function gap inequality (3.2) and typical Lyapunov arguments used in stochastic optimization. Namely, classical convergence arguments for stochastic methods on a static problem $\min_x \psi(x)$ rely on one-step improvement bounds of the form:

$$\eta_t \mathbb{E}_t[\psi(x_{t+1}) - \psi(x)] \leq \frac{1 - c_1 \eta_t}{2} \|x_t - x\|^2 - \frac{1 + c_2 \eta_t}{2} \|x_{t+1} - x\|^2 + c_3 \eta_t^2 \quad \forall x. \quad (4.4)$$

Here, $c_1, c_2, c_3 \in \mathbf{R}$ are some constants and $\eta_t > 0$ is a user-specified sequence; see also Lemma 4.3. As long as the sum $c_1 + c_2$ is positive, one may drive the gap $\mathbb{E}[\psi(x_{t+1}) - \psi(x)]$ below any fixed tolerance by choosing $\eta_t > 0$ appropriately. The condition $c_1 + c_2 > 0$ typically holds under strong convexity assumptions. Standard efficiency estimates follow by setting x to be the minimizer of ψ ; for our purposes, however, it is important that x can be arbitrary.

Returning to stochastic optimization with state-dependent distributions, recall that the losses themselves are assumed to be smooth and strongly convex. Therefore, stochastic methods on the static problem $\min \varphi_x$ (for any x) likely enjoys the estimate (4.4) with $c_1 + c_2 > 0$. Imagine now that given a current iterate x_t we take a single step of such an algorithm on the problem $\mathbf{St}(\mathcal{D}(x_t))$. Setting $x = \bar{x}$ (with $\psi \equiv \varphi_{x_t}$) and applying the function gap inequality implies the one-step improvement:

$$\eta_t \mathbb{E}_t[\varphi(x_{t+1}) - \varphi(\bar{x})] \leq \frac{1 - (c_1 - \gamma\beta)\eta_t}{2} \|x_t - \bar{x}\|^2 - \frac{1 + (c_2 - \gamma\beta)\eta_t}{2} \|x_{t+1} - \bar{x}\|^2 + c_3 \eta_t^2.$$

Therefore in the regime $c_1 + c_2 - 2\gamma\beta > 0$, we can drive the gap $\mathbb{E}_t[\varphi(x_{t+1}) - \varphi(\bar{x})]$ to zero with an appropriate choice of $\eta_t > 0$. Moreover, if φ is $\tilde{\alpha}$ -strongly convex (for some $\tilde{\alpha} > 0$), then we

may lower bound the left side by $(\eta_t \tilde{\alpha}/2)\mathbb{E}\|x_{t+1} - \bar{x}\|^2$. Elementary algebraic manipulations then show that the parameter regime of convergence in $\mathbb{E}\|x_{t+1} - \bar{x}\|^2$ improves to $c_1 + c_2 + \tilde{\alpha} - 2\gamma\beta > 0$. In summary, the function gap inequality (3.2) allows to translate one-step improvements on static problems $\min \varphi_{x_t}$ into one-step improvements on the target problem $\min \varphi$. We will show that model based algorithms on well-conditioned static problems satisfy a one-step improvement bound of the form (4.4) with $c_1 + c_2 > 0$, and then apply the outlined argument using (3.2).

4.5 Inexact repeated minimization (Section 9)

All the aforementioned stochastic algorithms draw a single sample in between every change in distribution. In practice, however, modifying the sampling distribution may be much more expensive than drawing a sample from the current distribution $\mathcal{D}(x_t)$. Following [45], we call the process of modifying the distribution *deployment*. It is an interesting question if it is possible to maintain the sample efficiencies of the aforementioned algorithms for (implicitly) solving $\mathbf{St}(\mathcal{D}(\bar{x}))$ while decreasing the number of deployments. An answer to this question for the projected stochastic gradient method appears in [45, Theorem 3.3]. Namely, consider running inexact repeated minimization for iterations $t = 0, 1, \dots$ with a projected stochastic gradient method applied for J_t iterations on each subproblem. The authors show that setting $J_t = \mathcal{O}(t^{1.1p})$ ensures that iterates x_t converge to \bar{x} at the rate $\mathcal{O}(t^{-p})$. Here $p > 0$ is a tuning parameter that controls the tradeoff between sampling and deployment. We prove a closely related result for all the algorithms analyzed in the previous sections; concisely, we show that the number of deployments can be reduced to be logarithmic in problem parameters without sacrificing sample efficiency.

We begin with the following theorem, which provides guarantees on the efficiency of repeated minimization with model-based algorithms used as inexact subsolvers.

Theorem 4.7 (Informal). *Define $\tilde{\alpha} = \alpha + \mu$ for stochastic proximal gradient and proximal point methods and set $\tilde{\alpha} = \mu$ for the clipped gradient method; in addition, set $\tilde{\rho} = \frac{\gamma\beta}{\tilde{\alpha}}$. In the regime $\tilde{\rho} < 1$, the three methods may be used as inexact solvers within repeated minimization. The resulting methods will generate a point x satisfying $\mathbb{E}\|x - \bar{x}\|^2 \leq \varepsilon$ using*

$$\mathcal{O}\left(\frac{1}{1-\tilde{\rho}} \cdot \left(\log\left(\frac{\varphi(x_0) - \varphi(\bar{x})}{\varepsilon}\right) + \log\left(\frac{\sigma^2}{1-\tilde{\rho}} \cdot \frac{1}{L\varepsilon}\right)\right)\right) \quad \text{deployments,}$$

and

$$\mathcal{O}\left(\frac{\log((1-\tilde{\rho})^{-1})}{1-\tilde{\rho}} \cdot \left(\frac{L}{\tilde{\alpha}(1-\tilde{\rho})} \log\left(\frac{\varphi(x_0) - \varphi(\bar{x})}{\varepsilon}\right) + \frac{\sigma^2}{\tilde{\alpha}(1-\tilde{\rho})^2\varepsilon}\right)\right) \quad \text{samples.}$$

Observe that the sample complexity in Theorems 4.6 and 4.7 are essentially the same. The key difference is that the number of deployments in the latter is only logarithmic in the problem parameters. As a simple illustration, Figure 7 depicts the performance of inexact repeated minimization on Example 3.1 with the stochastic gradient method for approximately solving the inner problems. We cap the number of deployments at 20 and adjust the number of inner iterations according to the schedule in Corollary 9.3. Experimentally, we see that online algorithms (see Figure 3) and those based on repeated minimization perform similarly on this example.

The following theorem provides guarantees on the efficiency of repeated minimization with an accelerated stochastic gradient method used as an inexact solver.

Theorem 4.8 (Informal). *In the regime $\rho < \frac{1}{2}$, the accelerated stochastic proximal gradient method may be used as an inexact solver within repeated minimization. The resulting method will generate*

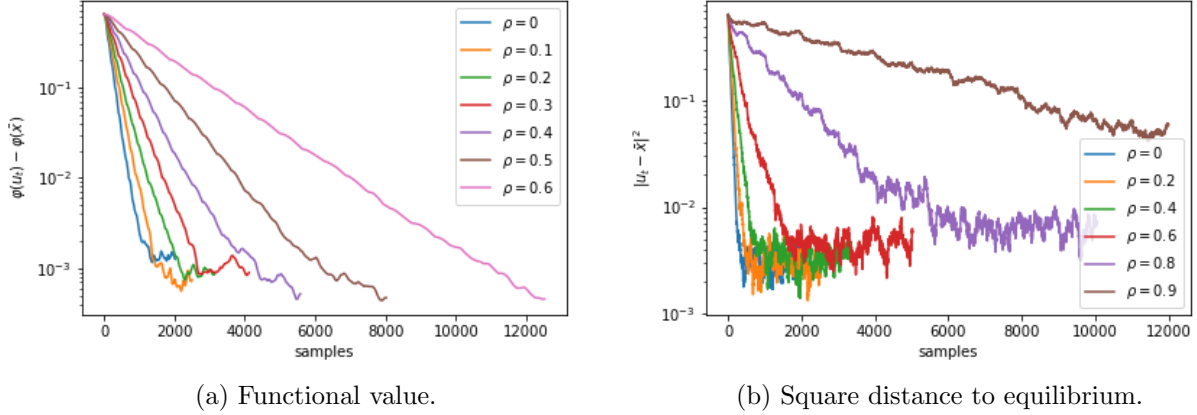


Figure 7: Inexact repeated minimization on Example 3.1.

a point x satisfying $\mathbb{E}[\varphi(x) - \varphi(\bar{x})] \leq \varepsilon$ using

$$\mathcal{O} \left(\left(1 - \frac{1}{2(1-\rho)}\right)^{-1} \left(\log \left(\frac{\varphi(x_0) - \varphi(\bar{x})}{\varepsilon} \right) + \log \left(\frac{\sigma^2}{(1 - \frac{\rho}{1-\rho})\varepsilon\sqrt{\alpha L}} \right) \right) \right) \quad \text{deployments,}$$

and

$$\mathcal{O} \left(\left(1 - \frac{1}{2(1-\rho)}\right)^{-1} \cdot \left(\sqrt{\kappa} \cdot \log \left(\frac{\varphi(x_0) - \varphi(\bar{x})}{\varepsilon} \right) + \frac{\sigma^2}{(1 - \frac{\rho}{1-\rho})\alpha\varepsilon} \right) \right) \quad \text{samples.}$$

Comparing Theorems 4.5 and 4.8, we see that the latter takes hold in the nearly optimal parameter regime $\rho < \frac{1}{2}$ without the extra factor of $\kappa^{-1/4}$. In addition, the number of deployments is only logarithmic in the problem parameters. The proof strategy for Theorems 4.7 and 4.8 again relies heavily on using the function gap inequality (3.2) to perturb Lyapunov type arguments.

Sections 5-9 formally justify the results outlined in this section.

5 Calm and contractive methods

In this section, we analyze two conceptual algorithms with state-dependent distributions:

$$\text{(Proximal point)} \quad x_{t+1} = \arg \min_x \left\{ f_{x_t}(x) + r(x) + \frac{1}{2\eta} \|x - x_t\|^2 \right\},$$

$$\text{(Proximal gradient)} \quad x_{t+1} = \text{prox}_{\eta r}(x_t - \eta \nabla f_{x_t}(x_t)).$$

Thus in each iteration t , the two methods simply take a proximal point and proximal gradient steps, respectively, on the static problem $\text{St}(\mathcal{D}(x_t))$. The proximal point method in the extreme case $\eta = \infty$ is called *repeated minimization* in [45]. The paper [45] showed that when r is the indicator function of a closed convex set and $\rho < 1$, repeated minimization and the proximal gradient method converge linearly to \bar{x} . In this section, we provide a different and complimentary viewpoint based on stability to distributional shifts.

Updates	$S_\nu(x) = \arg \min_y \{ \dots \}$	Calmness $\tau > 0$	Contraction $q \in [0, 1)$
Minimization	$f_\nu(y) + r(y)$	β/α	0
Prox-point	$f_\nu(y) + r(y) + \frac{1}{2\eta}\ y - x\ ^2$	$\frac{\eta\beta}{1+\eta\alpha}$	$\frac{1}{1+\eta\alpha}$
Prox-gradient	$\langle \nabla f_\nu(x), y \rangle + r(y) + \frac{1}{2\eta}\ y - x\ ^2$	$\eta\beta$	$\sqrt{1 - \alpha\eta}$ ($\eta \leq 1/L$)

Table 2: Calm and contractive updates under Assumption 1, 6: first column lists the names of the algorithms, second column specifies the updates, third column lists the calmness constants, the fourth columns lists the contraction factor of the map $x \mapsto S_\mu(x)$.

5.1 An interlude: calmness of algorithms for $\text{St}(\nu)$

We begin with an interlude quantifying the stability of algorithmic updates on the parametric family of problems $\text{St}(\nu)$ with $\nu \in \mathbb{P}$. To this end, let $S_\nu(x)$ denote an update of a point x by an algorithm on the static problem $\text{St}(\nu)$. Recall the notation $f_\nu(x) = \mathbb{E}_{z \sim \nu} \ell(x, z)$ from Section 2. Table 2 lists three basic examples that are worth keeping in mind: full minimization, proximal-point, and proximal gradient updates. A desirable property of an algorithm is that for any fixed x , the update map $S_\nu(x)$ is Lipschitz continuous with respect to variations in ν .

The following definition summarizes this stability property, relative to perturbations of a fixed distribution μ , which will later correspond to the equilibrium distribution $\mathcal{D}(\bar{x})$.

Definition 5.1 (Calmness). Fix a distribution $\mu \in \mathbb{P}$. We say that a map $S_\nu(x)$ is τ -calm relative to μ if the estimate holds:

$$\|S_\nu(x) - S_\mu(x)\| \leq \tau \cdot W_1(\nu, \mu), \quad \forall \nu \in \mathbb{P}.$$

Without further assumptions, standard algorithms can easily fail to be calm even if the loss function is convex.³ As a remedy, we impose strong convexity assumptions on the problem data.

Assumption 6 (Strong convexity). *Suppose that r is closed and convex and that f_μ is α -strongly convex for some distribution $\mu \in \mathbb{P}$ and some constant $\alpha > 0$.*

Theorem 5.2 verifies that under Assumptions 1 and 6, the three basic updates in Table 2 are indeed calm relative to μ . The fourth column in Table 2 also lists the well known contraction factors of the updates $x \mapsto S_\mu(x)$.

Theorem 5.2 (Calmness of updates). *Suppose that Assumptions 1 and 6 hold. Then the updates of the repeated minimization, proximal point, and proximal gradient methods are τ -calm with respect to the fixed distribution μ in Assumption 6, for the constants $\tau > 0$ appearing in Table 2.*

Proof. Our goal is to establish an upper bound $\sup_x \|S_\nu(x) - S_\mu(x)\| \leq \tau W_1(\mu, \nu)$ for the three algorithms. Consequently, let us fix a point x throughout the proof and define the updates corre-

³Consider the univariate function $l(x, z) = \frac{1}{4}x^4 + zx$ and define μ and ν_k to be point masses at zero and $\frac{1}{k}$, respectively. Set $x_\mu = \arg \min f_\mu$ and $x_{\nu_k} = \arg \min f_{\nu_k}$. A quick computation shows $|x_\mu - x_{\nu_k}|/W_1(\mu, \nu_k) = k^{2/3} \rightarrow \infty$ as $k \rightarrow \infty$.

sponding to repeated minimization, proximal point, and proximal gradient updates, respectively:

$$\begin{aligned} S_\nu^1(x) &= \arg \min_y \{f_\nu(y) + r(y)\}, \\ S_\nu^2(x) &= \arg \min_y \{f_\nu(y) + r(y) + \frac{1}{2\eta}\|y - x\|^2\}, \\ S_\nu^3(x) &= \text{prox}_{\eta r}(x - \eta \nabla f_\nu(x)). \end{aligned}$$

We first verify the calmness constant for repeated minimization, S_ν^1 . To this end, define the function $\varphi_\nu(y) := f_\nu(y) + r(y)$ and its minimizer $y_\nu := \arg \min \varphi_\nu$. First-order optimality conditions guarantee the inclusions $0 \in \partial\varphi_\mu(y_\mu)$ and $0 \in \partial\varphi_\nu(y_\nu) = \nabla f_\nu(y_\nu) + \partial r(y_\nu)$. In other words, there exist $\xi \in \partial r(y_\nu)$ such that $\nabla f_\nu(y_\nu) + \xi = 0$.

On the other hand, strong convexity of φ_μ guarantees $\alpha\|y - y'\| \leq \|w - w'\|$ for all $w \in \partial\varphi_\mu(y)$ and $w' \in \partial\varphi_\mu(y')$. We set

$$y = y_\mu, \quad w = 0 \in \partial\varphi_\mu(y_\mu), \quad y' = y_\nu, \quad w' = \nabla f_\mu(y_\nu) + \xi \in \partial\varphi_\mu(y_\nu).$$

Since $\nabla f_\nu(y_\nu) + \xi = 0$, we have $w' = \nabla f_\mu(y_\nu) - \nabla f_\nu(y_\nu)$ and thus can deduce

$$\alpha\|y_\mu - y_\nu\| \leq \|\nabla f_\mu(y_\nu) - \nabla f_\nu(y_\nu)\| \leq \beta \cdot W_1(\mu, \nu),$$

where the last inequality follows from Lemma 2.1. Noticing that $S_\nu^1(x) = y_\nu$ and $S_\mu^2(x) = y_\mu$, we thus arrive at the claimed estimate $\tau = \beta/\alpha$ for repeated minimization.

Calmness of the proximal point update S_ν^2 and proximal gradient update S_ν^3 follow by applying the above procedure with the different loss functions

$$\begin{aligned} \varphi_\nu^2(y) &= f_\nu(y) + r(y) + \frac{1}{2\eta}\|y - x\|^2, \\ \varphi_\nu^3(y) &= \langle \nabla f_\nu(x), y \rangle + r(y) + \frac{1}{2\eta}\|y - x\|^2, \end{aligned}$$

and recognizing $S_\nu^i(x)$ as the minimizers $S_\nu^i(x) = \arg \min_y \varphi_\nu^i(y)$ for $i = 2, 3$ respectively. They give the desired calmness constant in Table 2. \square

5.2 Linear convergence of conceptual algorithms

We next pass to the setting where the distribution governing the data is state-dependent. To this end, suppose that Assumptions 1, 2, 3(a), 4(a) hold. We will show that linear convergence of repeated minimization, proximal point, and proximal gradient methods is a direct consequence of the two independent phenomenon:

1. **(Calm)** The updates are τ -calm (Theorem 5.2).
2. **(Contractive)** The algorithms are q -contractive on the static problem $\text{St}(\mathcal{D}(\bar{x}))$. That is, there exists $q \in [0, 1)$ such that $\|S_{\bar{x}}(x) - S_{\bar{x}}(y)\| \leq q\|x - y\|$ for all $x, y \in \text{dom } r$.

It is elementary to see that an algorithm satisfying these two properties is automatically $(q + \tau\gamma)$ -contractive under state dependent sampling. This is the content of the following lemma.

Lemma 5.3 (Calm and contractive). *Fix a map $S_\nu(x)$ that is τ -calm and such that the map $x \mapsto S_{\mathcal{D}(\bar{x})}(x)$ is q -contractive with \bar{x} as its fixed point. Then the estimate holds:*

$$\|S_{\mathcal{D}(x)}(x) - \bar{x}\| \leq (q + \tau\gamma)\|x - \bar{x}\| \quad \forall x \in \mathbf{R}^d.$$

Proof. Abusing notation slightly and setting $S_x(y) := S_{\mathcal{D}(x)}(y)$, we compute

$$\|S_x(x) - \bar{x}\| = \|S_x(x) - S_{\bar{x}}(\bar{x})\| \leq \|S_x(x) - S_{\bar{x}}(x)\| + \|S_{\bar{x}}(x) - S_{\bar{x}}(\bar{x})\| \quad (5.1)$$

$$\leq \tau \cdot W_1(\mathcal{D}(x), \mathcal{D}(\bar{x})) + q\|x - \bar{x}\| \quad (5.2)$$

$$\leq (q + \tau\gamma)\|x - \bar{x}\|. \quad (5.3)$$

Here (5.1) uses the triangle inequality, (5.2) follows from calmness and contractiveness, while (5.3) follows from Assumption 2. The proof is complete. \square

Combining Lemma 5.3 with the constants τ and q specified in Table 2 immediately yields linear convergence guarantees for the three conceptual algorithms.

Corollary 5.4 (Repeated minimization, proximal point, and proximal gradient methods). *Suppose that Assumptions 1, 2, 3(a), 4(a) hold. Then for any point x , the following estimates hold:*

$$\frac{\|x^+ - \bar{x}\|}{\|x - \bar{x}\|} \leq \begin{cases} \frac{\gamma\beta}{\alpha} & \text{if } x^+ = \arg \min_y \{f_x(y) + r(y)\}, \\ \frac{1+\gamma\eta\beta}{1+\eta\alpha} & \text{if } x^+ = \arg \min_y \{f_x(y) + r(y) + \frac{1}{2\eta}\|y - x\|^2\}, \\ \sqrt{1 - \eta\alpha} + \gamma\eta\beta & \text{if } x^+ = \text{prox}_{\eta r}(x - \eta\nabla f_x(x)) \text{ and } \eta \leq \frac{1}{L}. \end{cases}$$

Thus iterated minimization and the prox-point methods converge linearly to \bar{x} in the regime $\rho < 1$, while the prox-gradient method converges linearly to \bar{x} in the regime $\rho < \frac{1}{2}$.

Corollary 5.4 shows that iterated minimization and proximal point methods converge linearly in the parameter regime $\rho < 1$. Moreover, as shown in [45], repeated minimization can easily diverge outside this parameter regime. The regime of convergence $\rho < \frac{1}{2}$ for the proximal gradient method therefore appears slightly suboptimal, most likely due to our decomposition of the analysis into two separate properties of calmness and contraction. This regime can be trivially enlarged when no regularization is present, i.e., when $r = 0$. Indeed, with the choice $\eta = \frac{2}{\alpha+L}$, the gradient method is $\frac{L-\alpha}{L+\alpha}$ contractive on the static problem $\text{St}(\mathcal{D}(\bar{x}))$ (e.g. [14, Theorem 3.12]). Lemma 5.3 therefore guarantees that the update x^+ satisfies

$$\frac{\|x^+ - \bar{x}\|}{\|x - \bar{x}\|} \leq \frac{L - \alpha}{L + \alpha} + \frac{2\gamma\beta}{L + \alpha}.$$

Clearly, the right side is smaller than one if and only if $\rho < 1$. A similar guarantee based on a different argument appears in [39, Proposition 2.5].

In general regularized settings, the regime of convergence of the proximal gradient method can be enlarged to $\rho < 1$ through a different argument. See the forthcoming guarantees for the proximal stochastic gradient method (Theorem 7.3 with $\sigma^2 = 0$).

6 Reduction to online convex optimization

This section shows that virtually any algorithm designed for online convex optimization can be applied under state-dependent sampling. Recall the setting of online convex optimization in Section 4.2. We assume that the feasible set K is bounded, say with diameter D , and the gradients of the cost functions are also bounded by a constant G , i.e., $\|\nabla \ell_t(x)\| \leq G$ for all $x \in K$ and

all t . Under these assumptions, typical online algorithms, including the online proximal gradient method (4.1), have bounded regret against any reference point $x \in \text{dom } r$:

$$R_t(x) := \sum_{i=1}^t ((\ell_i(x_i) + r(x_i)) - (\ell_i(x) + r(x))) \leq U_t, \quad (6.1)$$

where $U_t = \mathcal{O}(GD\sqrt{t})$ for convex losses with $\eta_t = D/(G\sqrt{t})$ and $U_t = \mathcal{O}((G^2/\alpha)\ln(t))$ for α -strongly convex losses with $\eta_t = 1/(\alpha t)$. See [19], [53] and [38] for the details.

We claim that one can directly apply online optimization algorithms in the setting of state-dependent sampling, and derive their convergence rate from the regret bounds (6.1). As usual, we let \bar{x} be an equilibrium point with respect to $\mathcal{D}(\cdot)$, set $\varphi_x(y) = \mathbb{E}_{z \sim \mathcal{D}(x)}[\ell(y, z)] + r(y)$, and use the shorthand $\varphi = \varphi_{\bar{x}}$.

Theorem 6.1 (Reduction). *Suppose that Assumptions 1, 2, 3(a) hold. Consider an online algorithm that in each iteration $t \geq 1$ encounters the loss function $\ell_t(x_t) = \ell(x_t, z_t)$ where $z_t \sim \mathcal{D}(x_t)$. Suppose that the uniform regret bound (6.1) holds for each $t \geq 1$. Then the average iterate $\hat{x}_t := \frac{1}{t} \sum_{i=1}^t x_i$ satisfies*

$$\mathbb{E}[\varphi(\hat{x}_t)] - \varphi(\bar{x}) \leq \frac{U_t}{(1 - 2\rho)t} \quad \forall t \geq 1.$$

Proof. Taking the expectation in (6.1) and setting $x = \bar{x}$ yields

$$\mathbb{E}[R_t(\bar{x})] = \sum_{i=1}^t \mathbb{E}[\varphi_{x_i}(x_i) - \varphi_{x_i}(\bar{x})] \leq U_t.$$

The function gap inequality (3.2) in turn implies

$$\varphi(x_i) - \varphi_{\bar{x}}(\bar{x}) \leq \varphi_{x_i}(x_i) - \varphi_{x_i}(\bar{x}) + \gamma\beta \cdot \|x_i - \bar{x}\|^2.$$

Strong convexity implies $\|x_i - \bar{x}\|^2 \leq \frac{2}{\alpha}(\varphi(x_i) - \varphi(\bar{x}))$ and hence

$$(1 - 2\rho)(\varphi(x_i) - \varphi(\bar{x})) \leq \varphi_{x_i}(x_i) - \varphi_{x_i}(\bar{x}).$$

Therefore, in the setting $\rho < 1/2$, we have

$$\mathbb{E}\left[\sum_{i=1}^t (\varphi(x_i) - \varphi(\bar{x}))\right] \leq \frac{1}{1 - 2\rho} \mathbb{E}[R_t(\bar{x})] \leq \frac{1}{1 - 2\rho} U_t.$$

Dividing both sides by t and using convexity of φ completes the proof. \square

Known regret bounds for the proximal gradient [19], regularized dual averaging [53], and FTRL [38] algorithms directly yield a converge rate $O(\log(t)/(1 - 2\rho)t)$ under state-dependent sampling and under the assumptions used in the aforementioned papers.

Though Theorem 6.1 is attractive in its generality, it is far from satisfactory. Indeed, the framework of online convex optimization is much more general than stochastic optimization, since the loss function ℓ_t at each iteration may not follow any probability distribution and can even be adversarial. As a consequence, online convex optimization requires stringent assumptions in order to have meaningful regret analysis, most notably boundedness of the encountered gradients of the loss functions. Moreover smoothness of the loss function does not play a significant role. We will see in the next section that much finer convergence guarantees hold for stochastic optimization under state-dependent distributions.

7 Stochastic gradient methods

In this section, we directly analyze the (accelerated) stochastic proximal gradient method, without relying on regret bounds. The only consequence of state-dependent distribution that will be relevant is that the stochastic estimator of the gradient is biased and the bias is bounded as in (3.1). As a result, we work with a broader model with a biased stochastic gradient oracle, formalized next in Assumption 7.

Assumption 7 (Biased stochastic gradient oracle). *Consider the optimization problem*

$$\min_x \varphi(x) := f(x) + r(x),$$

and denote its minimizer by \bar{x} . We suppose that there exist constants $L, B, \alpha > 0$ satisfying the following.

- (a) The function $f: \mathbf{R}^d \rightarrow \mathbf{R}$ is α -strongly convex and differentiable with L -Lipschitz gradient.
- (b) The function $r: \mathbf{R}^d \rightarrow \mathbf{R} \cup \{\infty\}$ is closed and convex.
- (c) For every point x , we may draw a realization of a random vector $g(x) \in \mathbf{R}^d$ from a common probability space satisfying the bias/variance bounds:

$$\|\mathbb{E}[g(x)] - \nabla f(x)\| \leq B\|x - \bar{x}\| \quad \text{and} \quad \mathbb{E}\|g(x) - \mathbb{E}[g(x)]\|^2 \leq \sigma^2.$$

Throughout, we define the condition number $\kappa := \frac{L}{\alpha}$ and set $\rho := \frac{B}{\alpha}$.

It is clear that Assumptions 1, 2, 3(a) and 4(a) under the setting of state-dependent distributions imply Assumption 7 with $g(x) := \nabla \ell(x, z)$ where $z \sim \mathcal{D}(x)$, and $B := \gamma\beta$. Consequently, all convergence guarantees presented in this section can be interpreted under the setting of state-dependent distributions by simply setting $B = \gamma\beta$.

The main consequence of the particular form of the bias in Assumption 7 (c) is summarized in the following lemma. In essence, the lemma states that a strong convexity type inequality holds between x and \bar{x} , with $\nabla f(x)$ replaced by $\mathbb{E}[g(x)]$.

Lemma 7.1 (Approximate subgradient inequality). *Under Assumption 7, it holds that*

$$f(\bar{x}) \geq f(x) + \langle \mathbb{E}[g(x)], \bar{x} - x \rangle + \frac{\alpha(1 - 2\rho)}{2} \|x - \bar{x}\|^2 \quad \forall x \in \mathbf{R}^d.$$

Proof. Strong convexity of f guarantees

$$\begin{aligned} f(\bar{x}) &\geq f(x) + \langle \nabla f(x), \bar{x} - x \rangle + \frac{\alpha}{2} \|x - \bar{x}\|^2 \\ &= f(x) + \langle \mathbb{E}[g(x)], \bar{x} - x \rangle + \frac{\alpha}{2} \|x - \bar{x}\|^2 + \langle \nabla f(x) - \mathbb{E}[g(x)], \bar{x} - x \rangle. \end{aligned}$$

The Cauchy-Schwarz inequality in turn yields the estimate

$$\langle \nabla f(x) - \mathbb{E}[g(x)], \bar{x} - x \rangle \geq -\|\mathbb{E}[g(x)] - \nabla f(x)\| \cdot \|\bar{x} - x\| \geq -B\|x - \bar{x}\|^2.$$

Combining the two inequalities above yields the desired result. \square

7.1 Stochastic proximal gradient method

We are now ready to analyze the stochastic proximal gradient method, summarized in Algorithm 1, under Assumption 7.

Algorithm 1: Stochastic proximal gradient method	
Input: initial x_0 and sequence $\{\eta_t\}_{t=0}^T \subset (0, \infty)$.	
Step $t = 0, 1, \dots, T$:	Sample $g_t = g(x_t)$
	Set $x_{t+1} = \text{prox}_{\eta_t r}(x_t - \eta_t g_t)$

Throughout, we let $\mathbb{E}_t[\cdot]$ be the conditional expectation given the iterates x_0, \dots, x_t . The following lemma provides a key recursion quantifying the one-step progress of the algorithm. The argument closely parallels the proof in the unbiased setting [26].

Lemma 7.2 (Key recursion). *Suppose that Assumption 7 holds and the step-size sequence satisfies $\eta_t < \frac{1}{L}$. Then the iterates $\{x_t\}$ generated by Algorithm 1 satisfy*

$$2\eta_t \mathbb{E}_t[\varphi(x_{t+1}) - \varphi(\bar{x})] \leq \left(1 - \alpha\eta_t(1 - 2\rho) + \frac{\eta_t^2 B^2}{1 - \eta_t L}\right) \|x_t - \bar{x}\|^2 - \mathbb{E}_t \|x_{t+1} - \bar{x}\|^2 + \frac{\eta_t^2 \sigma^2}{1 - \eta_t L}.$$

Proof. Since by Assumption 7 the gradient ∇f is L -Lipschitz, we have

$$\begin{aligned} \varphi(x_{t+1}) &= f(x_{t+1}) + r(x_{t+1}) \\ &\leq f(x_t) + \langle \nabla f(x_t), x_{t+1} - x_t \rangle + r(x_{t+1}) + \frac{L}{2} \|x_{t+1} - x_t\|^2 \\ &= f(x_t) + \langle g_t, x_{t+1} - x_t \rangle + r(x_{t+1}) + \frac{L}{2} \|x_{t+1} - x_t\|^2 - \langle g_t - \nabla f(x_t), x_{t+1} - x_t \rangle. \end{aligned}$$

Let $\delta_t > 0$ be an arbitrary positive sequence. We use Young's inequality to bound the last inner-product term in the above inequality, which results in

$$\begin{aligned} \varphi(x_{t+1}) &\leq f(x_t) + \langle g_t, x_{t+1} - x_t \rangle + r(x_{t+1}) + \frac{L}{2} \|x_{t+1} - x_t\|^2 \\ &\quad + \frac{\delta_t}{2} \|g_t - \nabla f(x_t)\|^2 + \frac{1}{2\delta_t} \|x_{t+1} - x_t\|^2 \\ &= f(x_t) + \langle g_t, x_{t+1} - x_t \rangle + r(x_{t+1}) + \frac{1}{2\eta_t} \|x_{t+1} - x_t\|^2 \\ &\quad + \frac{\delta_t}{2} \|g_t - \nabla f(x_t)\|^2 + \frac{\delta_t^{-1} - \eta_t^{-1} + L}{2} \|x_{t+1} - x_t\|^2 \\ &\leq f(x_t) + \langle g_t, \bar{x} - x_t \rangle + r(\bar{x}) + \frac{1}{2\eta_t} \|\bar{x} - x_t\|^2 - \frac{1}{2\eta_t} \|x_{t+1} - \bar{x}\|^2 \\ &\quad + \frac{\delta_t}{2} \|g_t - \nabla f(x_t)\|^2 + \frac{\delta_t^{-1} - \eta_t^{-1} + L}{2} \|x_{t+1} - x_t\|^2, \end{aligned} \tag{7.1}$$

where the last inequality (7.1) follows from the fact that x_{t+1} is by construction the minimizer of the η_t^{-1} -strongly convex function $f(x_t) + \langle g_t, \cdot - x_t \rangle + r + \frac{1}{2\eta_t} \|\cdot - x_t\|^2$.

Next set $\delta_t := \frac{\eta_t}{1-\eta_t L}$ to make the last term of (7.1) zero. Taking conditional expectations and using Lemma 7.1, we deduce

$$\begin{aligned}\mathbb{E}_t[\varphi(x_{t+1})] &\leq \varphi(\bar{x}) - \frac{\alpha(1-2\rho)}{2} \|x_t - \bar{x}\|^2 \\ &\quad + \frac{1}{2\eta_t} \|\bar{x} - x_t\|^2 - \frac{1}{2\eta_t} \mathbb{E}_t \|\bar{x} - x_{t+1}\|^2 + \frac{\delta_t}{2} \mathbb{E}_t \|g_t - \nabla f(x_t)\|^2.\end{aligned}\quad (7.2)$$

Next, we upper bound the last term in (7.2) as follows:

$$\mathbb{E}_t \|g_t - \nabla f(x_t)\|^2 = \mathbb{E} \|g_t - \mathbb{E}[g_t]\|^2 + \|\mathbb{E}[g_t] - \nabla f(x_t)\|^2 \leq \sigma^2 + B^2 \|x_t - \bar{x}\|^2.$$

Combining this estimate with (7.2) and grouping like terms completes the proof. \square

With Lemma 7.2 at hand, obtaining convergence guarantees under various choices of the control sequence $\eta_t > 0$ is completely standard. We highlight one such result based on using a constant sequence.

Theorem 7.3 (Constant step size). *Suppose Assumption 7 holds. Let x_t be the iterates generated by Algorithm 1 with a fixed parameter $\eta > 0$. Then the following are true.*

1. *Suppose we are in the regime $\rho < 1$ and set $\hat{\alpha} := \alpha - B$. Then with $\eta \leq \min \left\{ \frac{1}{2L}, \frac{1}{B^2/\hat{\alpha}+L} \right\}$, the estimate holds:*

$$\mathbb{E} \|x_t - \bar{x}\|^2 \leq \left(1 - \frac{2\eta\hat{\alpha}}{3}\right)^t \mathbb{E} \|x_0 - \bar{x}\|^2 + \frac{2\sigma^2\eta}{\hat{\alpha}}. \quad (7.3)$$

2. *Suppose we are in the regime $\rho < \frac{1}{2}$ and set $\hat{\alpha} := \alpha - 2B$. Then with $\eta \leq \frac{1}{2(B^2/\hat{\alpha}+L)}$, the estimate holds:*

$$\mathbb{E}[\varphi(\hat{x}_t) - \varphi(\bar{x})] \leq 2 \left(1 - \frac{\eta\hat{\alpha}}{2}\right)^t (\varphi(x_0) - \varphi(\bar{x})) + \sigma^2\eta. \quad (7.4)$$

where the average iterate is defined recursively by $\hat{x}_t = (1 - \frac{\hat{\alpha}\eta}{2})\hat{x}_{t-1} + \frac{\hat{\alpha}\eta}{2}x_t$.

Proof. Observe first that the assumption $\eta \leq \frac{1}{2L}$ implies $\frac{\eta^2\sigma^2}{1-\eta L} \leq 2\eta^2\sigma^2$. Therefore Lemma 7.2 yields the one step improvement:

$$2\eta\mathbb{E}[\varphi(x_{t+1}) - \varphi(\bar{x})] \leq \left(1 - \alpha\eta(1-2\rho) + \frac{\eta^2 B^2}{1-\eta L}\right) \mathbb{E} \|x_t - \bar{x}\|^2 - \mathbb{E} \|x_{t+1} - \bar{x}\|^2 + 2\eta^2\sigma^2. \quad (7.5)$$

Proof of claim 1: Lower bounding the left side of (7.5) using strong convexity, $\varphi(x_{t+1}) - \varphi(\bar{x}) \geq \frac{\alpha}{2} \|x_{t+1} - \bar{x}\|^2$, and rearranging yields the guarantee:

$$\mathbb{E} \|x_{t+1} - \bar{x}\|^2 \leq \left(1 - \frac{2\alpha\eta(1-\rho) - \frac{\eta^2 B^2}{1-\eta L}}{1 + \eta\alpha}\right) \mathbb{E} \|x_t - \bar{x}\|^2 + \frac{2\eta^2\sigma^2}{1 + \eta\alpha}.$$

The assumption $\eta < \frac{1}{B^2/\hat{\alpha}+L}$ directly implies $\frac{\eta^2 B^2}{1-\eta L} \leq \alpha\eta(1-\rho)$, and therefore we deduce

$$\mathbb{E} \|x_{t+1} - \bar{x}\|^2 \leq \left(1 - \frac{\alpha\eta(1-\rho)}{1 + \eta\alpha}\right) \mathbb{E} \|x_t - \bar{x}\|^2 + \frac{2\eta^2\sigma^2}{1 + \eta\alpha}.$$

Unrolling the recursion and using the estimate $\eta\alpha \leq \frac{1}{2}$ completes the proof of (7.3).

Proof of Claim 2: The assumption $\eta < \frac{1}{2(B^2/\hat{\alpha}+L)}$ by construction guarantees $\frac{\eta^2 B^2}{1-\eta L} \leq \frac{\hat{\alpha}\eta}{2}$. Therefore, the inequality (7.5) implies

$$2\eta\mathbb{E}_t[\varphi(x_{t+1}) - \varphi(\bar{x})] \leq \left(1 - \frac{\hat{\alpha}\eta}{2}\right) \|\bar{x} - x_t\|^2 - \mathbb{E}_t\|\bar{x} - x_{t+1}\|^2 + 2\eta^2\sigma^2.$$

Applying Corollary A.3 for the recursion completes the proof of (7.4). \square

Let us now translate Theorem 7.3 into an efficiency guarantee for finding an approximate minimizer of φ . There are two standard restarting techniques that achieve this goal. The first is based on restarting the constant step algorithm with exponentially increasing mini-batches of gradients $\frac{1}{m} \sum_{i=1}^m g_t^i$ in order to decrease the variance. The second approach restarts the constant step algorithm with geometrically decreasing parameters η . Both techniques are standard and are detailed in Appendix B. The following corollary (formal version of Theorem 4.4) focuses on the latter strategy for simplicity, though both strategies enjoy the same sample complexity guarantees.

Corollary 7.4 (Efficiency of Algorithm 1 with geometrically decaying schedule of η).

Suppose Assumption 7 holds. Then the following statements hold.

1. **(Distance)** *Suppose $\rho < 1$ and that we have available an estimate $\Delta \geq \|x_0 - \bar{x}\|^2$. Define the modified convexity parameter $\hat{\alpha} = \alpha - B$. Then Algorithm 1 may be augmented with the geometric decay schedule in Algorithm 8 (Appendix B) under the identification*

$$c = \frac{2\hat{\alpha}}{3}, \quad C = 1, \quad h(x) = \|x - \bar{x}\|^2, \quad \delta_0 = \frac{1}{B^2/\hat{\alpha} + L}, \quad D = \frac{2\sigma^2}{\hat{\alpha}}.$$

The resulting procedure will generate a point x satisfying $\mathbb{E}[\|x - \bar{x}\|^2] \leq \varepsilon$ using

$$\mathcal{O}\left(\left(\frac{B^2}{\hat{\alpha}^2} + \frac{L}{\hat{\alpha}}\right) \log\left(\frac{\Delta}{\varepsilon}\right) + \frac{\sigma^2}{\hat{\alpha}^2\varepsilon}\right) \quad \text{samples.}$$

2. **(Function value)** *Suppose $\rho < \frac{1}{2}$ and that we have available an estimate $\Delta \geq \varphi(x_0) - \varphi(\bar{x})$. Define the modified convexity parameter $\hat{\alpha} = \alpha - 2B$. Then Algorithm 1 may be augmented with the geometric decay schedule in Algorithm 8 (Appendix B) under the identification*

$$c = \frac{\hat{\alpha}}{2}, \quad C = 2, \quad h(x) = \varphi(x) - \varphi(\bar{x}), \quad \delta_0 = \frac{1}{2(B^2/\hat{\alpha} + L)}, \quad D = \sigma^2.$$

The resulting procedure will generate a point x satisfying $\mathbb{E}[\varphi(x) - \varphi(\bar{x})] \leq \varepsilon$ using

$$\mathcal{O}\left(\left(\frac{B^2}{\hat{\alpha}^2} + \frac{L}{\hat{\alpha}}\right) \log\left(\frac{\Delta}{\varepsilon}\right) + \frac{\sigma^2}{\hat{\alpha}\varepsilon}\right) \quad \text{samples.}$$

Proof. The stated sample complexity can be obtained directly by substituting the specified values of c, C, h, δ_0 and D into the results of Lemma B.1. For example, in the case of distance, we have

$$\frac{1}{c\delta_0} = \frac{1}{\frac{2}{3}\hat{\alpha} \cdot \frac{1}{B^2/\hat{\alpha}+L}} = \frac{3}{2} \frac{B^2 + L\hat{\alpha}}{\hat{\alpha}^2} = \frac{3}{2} \left(\frac{B^2}{\hat{\alpha}^2} + \frac{L}{\hat{\alpha}}\right).$$

The rest of the substitutions are more straightforward. \square

7.2 Accelerated Stochastic proximal gradient method

We next discuss the accelerated stochastic proximal gradient method in the oracle model (Assumption 7). To state the algorithm, we require a few auxiliary quantities. Setting the stage, define $\hat{\alpha} = \alpha - 2B$ and choose an arbitrary $\gamma_0 \geq \hat{\alpha}$ and stepsize parameters $\eta_t > 0$. Define the two auxiliary sequences

$$\begin{aligned} \delta_t &= \sqrt{\eta_t \gamma_t} && \text{for all } t \geq 0. \\ \gamma_t &= (1 - \delta_t)\gamma_{t-1} + \delta_t \hat{\alpha} && \text{for all } t \geq 1. \end{aligned} \tag{7.6}$$

Algorithm 2 summarizes the accelerated stochastic gradient method, proposed in [35].

Algorithm 2: Accelerated stochastic gradient method	
Input: Initial $x_0 \in \mathbf{R}^d$, $\gamma_0 > 0$, sequence $\{\eta_t\}_{t \geq 0} \in [0, \infty)$, count $T \in \mathbb{N}$.	
Step $t = 0, 1, \dots, T$: Set	
$x_t = y_{t-1} - \eta_t g_t,$	
$y_t = x_t + \beta_t(x_t - x_{t-1})$	where $\beta_t = \frac{\delta_t(1 - \delta_t)\eta_{t+1}}{\eta_t\delta_{t+1} + \eta_{t+1}\delta_t^2}.$

Notice that here the sequence β_t is not related to β in Assumption 1. If we set $\gamma_0 = \hat{\alpha}$ and $\eta_t = \frac{1}{4L}$ for all $t \geq 0$, then for all $t \geq 0$,

$$\gamma_t = \hat{\alpha}, \quad \delta_t = \frac{1}{2}\sqrt{\frac{\hat{\alpha}}{L}}, \quad \beta_t = \frac{1 - (1/2)\sqrt{\hat{\alpha}/L}}{1 + (1/2)\sqrt{\hat{\alpha}/L}}.$$

Our main result of analyzing Algorithm 2 is the following theorem.

Theorem 7.5 (Constant step accelerated method). *Suppose that we are in the regime $\rho \leq \frac{1/2}{1 + \sqrt{32 + 64\sqrt{3\kappa}}}$. Set $\gamma_0 = \hat{\alpha}$ and $\eta_t = \frac{1}{4L}$ for all $t \geq 0$. Then the iterates generated by Algorithm 2 satisfy*

$$\mathbb{E}[\varphi(x_t) - \varphi(\bar{x})] \leq 2 \left(1 - \sqrt{\frac{\hat{\alpha}}{4L}}\right)^t (\varphi(x_0) - \varphi(\bar{x})) + \frac{9\sigma^2}{16\sqrt{L\hat{\alpha}}}.$$

Theorem 7.5 immediately yields an accelerated rate of convergence in the setting $\rho = \frac{B}{\alpha} \leq \frac{1/2}{1 + \sqrt{32 + 64\sqrt{3\kappa}}}$. Indeed, Lemma B.1 trivially implies that combining Algorithm 2 with a minibatch restart strategy (Algorithm 7) yields a procedure that will find a point x satisfying $\mathbb{E}[\varphi(x) - \min \varphi] \leq \varepsilon$ using

$$\mathcal{O}\left(\sqrt{\kappa} \cdot \log\left(\frac{\Delta}{\varepsilon}\right) + \frac{\sigma^2}{\alpha\varepsilon}\right)$$

stochastic gradient samples, where $\Delta \geq \varphi(x_0) - \varphi(\bar{x})$ is user specified (see Theorem 4.5).

The proof of Theorem 7.5 is quite long and technical, relying on the machinery of stochastic estimate sequences [35]. Therefore, we have placed it in Appendix C. The high level idea of the argument is as follows. Existing arguments (in the unbiased setting) based on estimate sequences $d_t(x) = d_t^* + \frac{\eta_t}{2}\|x - v_t\|^2$ rely on lower bounding the error $\mathbb{E}[d_t^* - \varphi(x_t)]$; see e.g. [35, 41]. The usual path is through a sequence of clever algebraic manipulations. During these manipulations, there is a term $\|y_t - v_t\|^2$ that appears, which is lower-bounded by zero and ignored. In contrast, we show that this term balances the incurred bias of the stochastic gradients.

8 Model-based algorithms

This section presents convergence guarantees for a wide class of algorithms of a “proximal point type”, introduced in the two recent papers [3, 17] in the static setting.

8.1 Model-based algorithms for a static problem

We begin by reviewing model-based algorithms for static problems and refine the available convergence guarantees. Namely, the convergence guarantees developed in the papers [3, 17] require the second moment of subgradients to be bounded. We will show that when the loss function is smooth, we may instead assume a bound on the variance.

Setting the stage, consider the static optimization problem

$$\min_x F(x) = f(x) + r(x) \quad \text{with} \quad f(x) = \mathbb{E}_{z \sim \mathcal{P}} \ell(x, z),$$

where \mathcal{P} is some probability distribution, the loss $\ell(x, z)$ is differentiable in x , and $r: \mathbf{R}^d \rightarrow \mathbf{R} \cup \{\infty\}$ is a closed function. In each iteration t , the algorithms we consider draw an i.i.d set of m samples $S_t \stackrel{i.i.d}{\sim} \mathcal{P}$ and approximate the loss function $\ell(\cdot, z_t)$ by a simpler model $\ell_{x_t}(\cdot, S_t)$ formed at the basepoint x_t . The next iterate x_{t+1} is then declared to be the minimizer of the function $\ell_{x_t}(\cdot, S_t) + r + \frac{1}{2\eta_t} \|\cdot - x_t\|^2$. The formal procedure is stated in Algorithm 3.

Algorithm 3: Model-based algorithm for static problems
<p>Input: initial x_0 and sequence $\{\eta_t\}_{t=0}^T \subset (0, \infty)$.</p> <p>Step $t = 0, 1, \dots, T$:</p> <p style="padding-left: 40px;">Sample $S_t \stackrel{i.i.d}{\sim} \mathcal{P}$</p> <p style="padding-left: 40px;">Set $x_{t+1} = \arg \min_y \left\{ \ell_{x_t}(y, S_t) + r(y) + \frac{1}{2\eta_t} \ y - x_t\ ^2 \right\}$</p>

Clearly, convergence guarantees of Algorithm 4 must depend both on the regularity of the models $\ell_x(y, S)$ individually and on how well the models approximate f . The following assumption formalizes these two properties. It will be useful to keep track of two parameters $\alpha_1, \alpha_2 \geq 0$, which measure “strong convexity” type properties, along with a variance bound $\sigma_0 > 0$.

Assumption 8 (Models for static problems). *There exist constants $\alpha_1, \alpha_2, \sigma_0 \geq 0$ such that the following properties hold for all $x, y \in \mathbf{R}^d$ and for almost all samples $S \stackrel{i.i.d}{\sim} \mathcal{P}$:*

- (a) **(Measurability)** *The function $(x, y, z) \mapsto \ell_x(y, z)$ is measurable.*
- (b) **(Convexity)** *The model $\ell_x(\cdot, S)$ is convex and the sum $\ell_x(\cdot, S) + r$ is α_1 -strongly convex.*
- (c) **(Bias/variance)** *The model $\ell_x(\cdot, S)$ is differentiable at x and satisfies*

$$\mathbb{E}_S[\nabla \ell_x(x, S)] = \nabla f(x) \quad \text{and} \quad \mathbb{E}_S \|\nabla \ell_x(x, S) - \nabla f(x)\|^2 \leq \sigma_0^2.$$

- (d) **(Accuracy)** *The estimate holds:*

$$\mathbb{E}_S[\ell_x(x, S) - \ell_x(y, S)] \geq f(x) - f(y) + \frac{\alpha_2}{2} \|x - y\|^2.$$

The convexity assumption (b) is self-explanatory. The Bias/variance property (c) asserts that $\nabla \ell_x(x, S)$ is an unbiased estimator of $\nabla f(x)$ and has finite variance σ_0^2 . The accuracy assumption (d) simply states that the gap $\ell_x(x, S) - \ell_x(y, S)$ is lower bounded in expectation by the true gap $f(x) - f(y)$. Note that assumption (d) is trivially implied by the two intuitive conditions:

$$\mathbb{E}_S[\ell_x(x, S)] = f(x) \quad \text{and} \quad \mathbb{E}_S[\ell_x(y, S)] + \frac{\alpha_2}{2} \|x - y\|^2 \leq f(y),$$

holding for all $x, y \in \mathbf{R}^d$. The first simply says that the model $\ell_x(\cdot, S)$ evaluated at the basepoint x coincides with $f(x)$ in expectation, while the second asserts that the model $\ell_x(\cdot, S)$ lower bounds f in expectation with an additional quadratic gap.

An important consequence of Assumption 8 is that F is strongly convex with parameter $\alpha_1 + \alpha_2$.

Lemma 8.1. *The function F is $(\alpha_1 + \alpha_2)$ -strongly convex.*

Proof. We will show that F satisfies a strong form of a subgradient inequality. To this end, fix a sample set $S \stackrel{i.i.d.}{\sim} \mathcal{P}$, and a point $x \in \mathbf{R}^d$ along with a subgradient $v \in \partial F(x)$. Notice that the inclusion $v - \nabla f(x) \in \partial r(x)$ holds. Therefore, the vector $v - \nabla f(x) + \nabla \ell_x(x, S)$ is a subgradient of $\ell_x(\cdot, S) + r$ at x . We compute

$$F(y) - F(x) \geq \mathbb{E}_S[\ell_x(y, S) + r(y) - \ell_x(x, S) - r(x)] + \frac{\alpha_2}{2} \|y - \bar{x}\|^2 \quad (8.1)$$

$$\geq \mathbb{E}_S[\langle v - \nabla f(x) + \nabla \ell_x(x, S), y - x \rangle] + \frac{\alpha_1 + \alpha_2}{2} \|y - x\|^2 \quad (8.2)$$

$$= \langle v, y - x \rangle + \frac{\alpha_1 + \alpha_2}{2} \|y - x\|^2, \quad (8.3)$$

where (8.1), (8.2), and (8.3) follow from conditions (d), (b), and (c) of Assumption 8, respectively. It follows immediately that ∂F is $(\alpha_1 + \alpha_2)$ -strongly monotone. Therefore [46, Theorem 12.17] directly implies that F is $(\alpha_1 + \alpha_2)$ -strongly convex, as claimed. \square

8.1.1 Examples: proximal point, proximal gradient, and clipped gradient

Let us look at three examples of models satisfying Assumption 8 and the corresponding algorithms; see the accompanying Figure 5 and Table 3. Specifically, for the three examples, we assume the following two conditions on the loss and the regularizer:

1. **(Variance)** There exists $\sigma > 0$ satisfying

$$\mathbb{E}_{z \sim \mathcal{P}} \|\nabla \ell(x, z) - \nabla f(x)\|^2 \leq \sigma^2 \quad \text{for all } x \in \mathbf{R}^d.$$

2. **(Strong convexity)** There exist $\alpha, \mu \geq 0$ such that r is μ -strongly convex and for every $z \in Z$, the loss $\ell(\cdot, z)$ is α -strongly convex.

Algorithm 3 clearly treats ℓ and r differently, and therefore α and μ play distinct roles. Notice that one can always modify the constants α and μ while maintaining the sum $\alpha + \mu$, simply by adding/subtracting a fixed quadratic $\frac{\lambda}{2} \|\cdot\|^2$ from ℓ and r .

Example 8.1 (Stochastic proximal point). The simplest stochastic model of a loss is the loss itself:

$$\ell_x(y, z) = \ell(y, z).$$

Algorithm 3 equipped with these models is the proximal point method. Assumption 8 trivially holds with parameters $(\alpha_1, \alpha_2, \sigma_0) = (\alpha + \mu, 0, \sigma)$. More generally, given a sample set $S \stackrel{i.i.d.}{\sim} \mathcal{P}$, we may use the average model $\ell_x(y, S) = \frac{1}{|S|} \sum_{z \in S} \ell(y, z)$ whose gradient has the smaller variance $\sigma_0^2 = \sigma^2/|S|$.

Method	Model $\ell_x(y, z)$	$(\alpha_1, \alpha_2, \sigma_0)$
Prox-point	$\ell(y, z)$	$(\alpha + \mu, 0, \sigma)$
Gradient	$\ell(x, z) + \langle \nabla \ell(x, z), y - x \rangle$	(μ, α, σ)
Clipped-gradient	$\max\{\ell(x, z) + \langle \nabla \ell(x, z), y - x \rangle, 0\}$	$(\mu, 0, \sigma)$

Table 3: Stochastic proximal point, gradient, and clipped gradient methods.

Example 8.2 (Stochastic proximal gradient). Another class of models is induced by linearizations

$$\ell_x(y, z) = \ell(x, z) + \langle \nabla \ell(x, z), y - x \rangle.$$

Algorithm 3 equipped with these models reduces to the stochastic proximal gradient method. Assumption 8(b) clearly holds with $\alpha_1 = \mu$. Assumption 8(c) holds trivially. Assumption 8(d) with $\alpha_2 = \alpha$ follows from the expression $\mathbb{E}_{z \sim \mathcal{D}(x)} \ell_x(x, z) = f_x(x)$ and strong convexity of f_x . More generally, given a sample set $S \stackrel{i.i.d}{\sim} \mathcal{P}$, we may declare $\ell_x(y, S) = \frac{1}{|S|} \sum_{z \in S} \ell_x(y, z)$ thereby decreasing the variance $\sigma_0^2 = \sigma^2/|S|$.

Example 8.3 (Stochastic clipped proximal gradient). An interesting middle ground between proximal point and gradient models was proposed in [3]. Namely, suppose that each loss $\ell(\cdot, z)$ is lower bounded by some known constant, which we may without loss of generality assume is zero. This assumption is completely innocuous in data scientific contexts. Then we may use the clipped linear models

$$\ell_x(y, z) = \max\{\ell(x, z) + \langle \nabla \ell(x, z), y - x \rangle, 0\}.$$

Algorithm 3 equipped with these models is the proximal clipped stochastic gradient method. A quick computation shows that Assumption 8 holds with $(\alpha_1, \alpha_2, \sigma_0) = (\mu, 0, \sigma)$. Given a sample set $S \stackrel{i.i.d}{\sim} \mathcal{P}$, we can form the two models,

$$\ell_x(x, S) = \frac{1}{|S|} \sum_{z \in S} \ell(x, z) \quad \text{and} \quad \ell_x(y, z) = \max\left\{\frac{1}{|S|} \sum_z \ell(x, z) + \langle \nabla \ell(x, z), y - x \rangle, 0\right\}.$$

It is straightforward to verify that both models $\ell_x(\cdot, S)$ are differentiable at x with gradient $\nabla \ell_x(x, S) = \frac{1}{|S|} \sum_{z \in S} \nabla \ell(x, z)$. Thus the variance of the gradient improves to $\sigma_0^2 = \sigma^2/|S|$.

8.1.2 Convergence guarantees.

We are now ready to analyze Algorithm 3. The main idea of the convergence proof is to establish a one-step improvement guarantee on the function F . Henceforth, we let $\mathbb{E}_t[\cdot]$ denote the expectation conditioned on x_t .

Lemma 8.2 (One step improvement). *Suppose that Assumption 8 holds and that ∇f is L -Lipschitz continuous. Fix a sequence $\eta_t < \frac{1}{L}$ for all $t \geq 0$. Then the iterates generated by Algorithm 3 satisfy:*

$$\eta_t \mathbb{E}_t[F(x_{t+1}) - F(y)] \leq \frac{1 - \alpha_2 \eta_t}{2} \|x_t - y\|^2 - \frac{1 + \alpha_1 \eta_t}{2} \mathbb{E}_t \|x_{t+1} - y\|^2 + \frac{\sigma_0^2 \eta_t^2}{2(1 - \eta_t L)}, \quad (8.4)$$

for all indices $t \geq 0$ and all $y \in \mathbf{R}^d$.

Proof. Observe that the inequality we wish to prove is conditioned on x_t . Therefore to simplify notation, set $x := x_t$, $S := S_t$, and $x^+ := x_{t+1}$. Since x^+ is the minimizer of a $(\eta_t^{-1} + \alpha_1)$ -strongly convex function $\ell_x(\cdot, S) + r + \frac{1}{2\eta_t} \|\cdot - x\|^2$, we deduce

$$\frac{\eta_t^{-1} + \alpha_1}{2} \|x^+ - y\|^2 \leq \left(\ell_x(y, S) + r(y) + \frac{1}{2\eta_t} \|y - x\|^2 \right) - \left(\ell_x(x^+, S) + r(x^+) + \frac{1}{2\eta_t} \|x^+ - x\|^2 \right). \quad (8.5)$$

We next lower bound $\ell_x(x^+, S)$ using convexity: Assumption 8 guarantees

$$\ell_x(x^+, S) \geq \ell_x(x, S) + \langle \nabla \ell_x(x, S), x^+ - x \rangle.$$

Combining this estimate with (8.5), multiplying through by $2\eta_t$, and taking the expectation yields

$$(1 + \alpha_1 \eta_t) \mathbb{E}_S \|x^+ - y\|^2 \leq \|x - y\|^2 - 2\eta_t \mathbb{E}_S [\ell_x(x, S) - \ell_x(y, S)] - \mathbb{E}_S \|x^+ - x\|^2 - 2\eta_t \mathbb{E}_S [\langle \nabla \ell_x(x, S), x^+ - x \rangle] - 2\eta_t \mathbb{E}_S [r(x^+) - r(y)]. \quad (8.6)$$

Assumption 8(d) implies

$$\mathbb{E}_S [\ell_x(x, S) - \ell_x(y, S)] \geq f(x) - f(y) + \frac{\alpha_2}{2} \|x - y\|^2.$$

Combining this estimate with (8.6) yields

$$(1 + \alpha_1 \eta_t) \mathbb{E}_S \|x^+ - y\|^2 \leq (1 - \alpha_2 \eta_t) \|x - y\|^2 - 2\eta_t \mathbb{E}_S [f(x) + r(x^+) - f(y) - r(y)] - \mathbb{E}_S \|x^+ - x\|^2 - 2\eta_t \mathbb{E}_S [\langle \nabla \ell_x(x, S), x^+ - x \rangle]. \quad (8.7)$$

Next, smoothness of f guarantees

$$f(x) \geq f(x^+) - \langle \nabla f(x), x^+ - x \rangle - \frac{L}{2} \|x^+ - x\|^2. \quad (8.8)$$

Taking expectations in (8.8) and combining the estimate with (8.7) yields

$$(1 + \alpha_1 \eta_t) \mathbb{E}_S \|x^+ - y\|^2 \leq (1 - \alpha_2 \eta_t) \|x - y\|^2 - 2\eta_t \mathbb{E}_S [F(x^+) - F(y)] + 2\eta_t \mathbb{E}_S [\langle \nabla f(x) - \nabla \ell_x(x, S), x^+ - x \rangle] - (1 - \eta_t L) \mathbb{E}_S \|x^+ - x\|^2.$$

Young's inequality in turn guarantees

$$2\eta_t \langle \nabla f(x) - \nabla \ell_x(x, S), x^+ - x \rangle \leq \frac{\eta_t^2 \|\nabla f(x) - \nabla \ell_x(x, S)\|^2}{1 - \eta_t L} + (1 - \eta_t L) \|x^+ - x\|^2.$$

Combining the last two inequalities with the finite variance assumption 8(c) yields

$$(1 + \alpha_1 \eta_t) \mathbb{E}_S \|x^+ - y\|^2 \leq (1 - \alpha_2 \eta_t) \|x - y\|^2 - 2\eta_t \mathbb{E}_S [F(x^+) - F(y)] + \frac{\sigma_0^2 \eta_t^2}{1 - \eta_t L}.$$

Rearranging yields (8.4) as claimed. \square

Using a constant parameter $\eta > 0$ yields the following guarantee.

Theorem 8.3 (Constant step). *Suppose that Assumption 8 holds and that ∇f is L -Lipschitz continuous. Fix a sequence $\eta < \min\{\frac{1}{2L}, \frac{1}{\alpha_1}, \frac{1}{\alpha_2}\}$. Then for all points $y \in \mathbf{R}^d$, the estimate holds*

$$\mathbb{E}[F(\hat{x}_t) - F(y)] \leq 2 \left(1 - \frac{(\alpha_1 + \alpha_2)\eta}{2}\right)^t (F(x_0) - F(y)) + \sigma_0^2 \eta,$$

where we recursively define $\hat{x}_t = \left(1 - \frac{\eta(\alpha_1 + \alpha_2)}{1 + c_2 \eta}\right) \hat{x}_{t-1} + \frac{\eta(\alpha_1 + \alpha_2)}{1 + c_2 \eta} x_t$.

Proof. The assumption $\eta < \frac{1}{2L}$ guarantees $\frac{\sigma_0^2 \eta^2}{2(1 - \eta L)} \leq \sigma_0^2 \eta$, and therefore Lemma 8.2 implies

$$\eta \mathbb{E}_t[F(x_{t+1}) - F(y)] \leq \frac{1 - \alpha_2 \eta}{2} \|x_t - y\|^2 - \frac{1 + \eta \alpha_1}{2} \mathbb{E}_t \|x_{t+1} - y\|^2 + \sigma_0^2 \eta.$$

Corollary A.3 directly implies

$$\mathbb{E}[F(\hat{x}_t) - F(y)] \leq \left(1 - \frac{(\alpha_1 + \alpha_2)\eta}{1 + \alpha_1 \eta}\right)^t \left(F(x_0) - F(y) + \frac{\alpha_1 + \alpha_2}{2} \|x_0 - y\|^2\right) + \sigma_0^2 \eta.$$

Taking into account of $\alpha_1 \eta \leq 1$ and $\alpha_2 \eta \leq 1$ and applying Lemma 8.1 complete the proof. \square

Combining Algorithm 3 with the geometric decay schedule (Algorithm 8 in Appendix B) yields an algorithm with the following efficiency guarantee, which is immediate from Lemma B.2.

Corollary 8.4 (Efficiency). *Suppose that Assumption 8 holds and that ∇f is L -Lipschitz continuous. Fix an estimate $\Delta \geq F(x_0) - \min F$. Then Algorithm 3 may be augmented with the geometric decay schedule (Algorithm 8) under the identification*

$$c = \frac{\alpha_1 + \alpha_2}{2}, \quad C = 2, \quad h(x) = F(x) - \min F, \quad \delta_0 = \min \left\{ \frac{1}{2L}, \frac{1}{\alpha_1}, \frac{1}{\alpha_2} \right\}, \quad D = \sigma_0^2.$$

The resulting procedure will generate a point x satisfying $\mathbb{E}[F(x) - \min F] \leq \varepsilon$ after

$$\mathcal{O} \left(\frac{L}{\alpha_1 + \alpha_2} \log \left(\frac{\Delta}{\varepsilon} \right) + \frac{\sigma_0^2}{(\alpha_1 + \alpha_2)\varepsilon} \right) \quad \textit{iterations}.$$

8.2 Model-based algorithms under state-dependent distributions

Model-based algorithms easily adapt to the setting with state-dependent distributions by allowing the sampling distribution to vary along the iterations. The formal procedure is stated in Algorithm 4.

Algorithm 4: Model-based algorithm under state-dependent sampling

Input: initial x_0 and sequence $\{\eta_t\}_{t=0}^T \subset (0, \infty)$.

Step $t = 0, 1, \dots, T$:

Sample $S_t \stackrel{i.i.d.}{\sim} \mathcal{D}(x_t)$

Set $x_{t+1} = \arg \min_y \left\{ \ell_{x_t}(y, S_t) + r(y) + \frac{1}{2\eta_t} \|y - x_t\|^2 \right\}$

Naturally, we will impose Assumption 8 for all static problems (1.1) with $\mathcal{P} = \mathcal{D}(x)$.

Assumption 9 (Stochastic models under state feedback). *Suppose that Assumption 8 holds for the static problem $\text{St}(\mathcal{D}(x))$ for all $x \in \mathbf{R}^d$, with the same set of parameters $(\alpha_1, \alpha_2, \sigma_0)$.*

For the rest of the section, we suppose that Assumptions 2, 4(b), and 9 hold. We are now ready to analyze Algorithm 4. The main idea of the convergence proof is to combine the one-step improvement guarantee (8.4) for the intermediate function φ_{x_t} with the function gap inequality (3.2) to obtain a one-step improvement on φ .

Corollary 8.5 (Key recursion). *Suppose the inequality $\eta_t < \frac{1}{L}$ holds for all $t \geq 0$. Then the iterates generated by Algorithm 4 satisfy the estimate:*

$$\eta_t \mathbb{E}_t[\varphi(x_{t+1}) - \varphi(\bar{x})] \leq \frac{1 - (\alpha_2 - \gamma\beta)\eta_t}{2} \|x_t - \bar{x}\|^2 - \frac{1 + (\alpha_1 - \gamma\beta)\eta_t}{2} \|x_{t+1} - \bar{x}\|^2 + \frac{\sigma_0^2 \eta_t^2}{2(1 - \eta_t L)}. \quad (8.9)$$

Proof. The function gap inequality (3.2) directly yields

$$\begin{aligned} \varphi_{x_t}(x_{t+1}) - \varphi_{x_t}(\bar{x}) &\geq \varphi_{\bar{x}}(x_{t+1}) - \varphi_{\bar{x}}(\bar{x}) - \gamma\beta \|x_{t+1} - \bar{x}\| \cdot \|x_t - \bar{x}\| \\ &\geq \varphi_{\bar{x}}(x_{t+1}) - \varphi_{\bar{x}}(\bar{x}) - \frac{\gamma\beta}{2} \|x_{t+1} - \bar{x}\|^2 - \frac{\gamma\beta}{2} \|x_t - \bar{x}\|^2, \end{aligned}$$

where the last inequality follows from Young's inequality. Using this estimate to lower bound the left side of (8.4) and rearranging completes the proof. \square

The following theorem summarizes the convergence guarantees of the method with a constant parameter $\eta > 0$.

Theorem 8.6 (Convergence guarantees). *Suppose that Assumptions 2, 4(b), and 9 hold. Let x_t be the iterates generated by Algorithm 4 with a fixed parameter $\eta > 0$. Then the following are true.*

1. *Suppose, we are in the regime $\frac{\gamma\beta}{\alpha_1 + \alpha_2} < 1$ and set $\hat{\alpha} := \alpha_1 + \alpha_2 - \gamma\beta$. Then with the parameter $\eta \leq \min\{\frac{1}{2L}, \frac{1}{\alpha_1}, \frac{1}{\alpha_2}\}$, the estimate holds:*

$$\mathbb{E}\|x_t - \bar{x}\|^2 \leq \left(1 - \frac{\hat{\alpha}\eta}{2}\right)^t \|x_0 - \bar{x}\|^2 + \frac{\sigma_0^2 \eta}{2\hat{\alpha}}. \quad (8.10)$$

2. *Suppose we are in the regime $\frac{\gamma\beta}{\alpha_1 + \alpha_2} < \frac{1}{2}$ and set $\hat{\alpha} := \alpha_1 + \alpha_2 - 2\gamma\beta$. Then with the parameter $\eta \leq \min\{\frac{1}{2L}, \frac{1}{(\alpha_1 - \gamma\beta)^+}, \frac{1}{(\alpha_2 - \gamma\beta)^+}\}$ where $(\cdot)^+ = \max\{\cdot, 0\}$, the estimate holds:*

$$\mathbb{E}[\varphi(\hat{x}_t) - \varphi(\bar{x})] \leq 2 \left(1 - \frac{\hat{\alpha}\eta}{2}\right)^t (\varphi(x_0) - \varphi(\bar{x})) + \sigma_0^2 \eta,$$

where we recursively define $\hat{x}_t = \left(1 - \frac{\hat{\alpha}\eta}{1 + (\alpha_1 - \gamma\beta)\eta}\right) \hat{x}_{t-1} + \frac{\hat{\alpha}\eta}{1 + (\alpha_1 - \gamma\beta)\eta} x_t$.

Proof. Notice that the assumption $\eta \leq \frac{1}{2L}$ implies $\frac{\sigma_0^2 \eta^2}{2(1 - \eta L)} \leq \sigma_0^2 \eta^2$. Combining this estimate with Corollary 8.5 yields:

$$\eta \mathbb{E}[\varphi(x_{t+1}) - \varphi(\bar{x})] \leq \frac{1 - (\alpha_2 - \gamma\beta)\eta}{2} \mathbb{E}\|x_t - \bar{x}\|^2 - \frac{1 + (\alpha_1 - \gamma\beta)\eta}{2} \mathbb{E}\|x_{t+1} - \bar{x}\|^2 + \sigma_0^2 \eta^2. \quad (8.11)$$

Proof of Claim 1: Lower-bounding the left-side using Lemma 8.1 and rearranging yields

$$\frac{1 + (2\alpha_1 + \alpha_2 - \gamma\beta)\eta}{2} \mathbb{E}\|x_{t+1} - \bar{x}\|^2 \leq \frac{1 - (\alpha_2 - \gamma\beta)\eta}{2} \mathbb{E}\|x_t - \bar{x}\|^2 + \sigma_0^2 \eta^2. \quad (8.12)$$

The assumption $\alpha_1 + \alpha_2 - \gamma\beta > 0$ ensures that the left side is positive, and therefore

$$\mathbb{E}\|x_{t+1} - \bar{x}\|^2 \leq \left(1 - \frac{2(\alpha_1 + \alpha_2 - \gamma\beta)\eta}{1 + (2\alpha_1 + \alpha_2 - \gamma\beta)\eta}\right) \mathbb{E}\|x_t - \bar{x}\|^2 + \frac{\sigma_0^2\eta^2}{1 + (2\alpha_1 + \alpha_2 - \gamma\beta)\eta}.$$

The assumption $\eta \leq \frac{1}{\alpha_2}$ ensures that the coefficient multiplying $\mathbb{E}\|x_t - \bar{x}\|^2$ lies in $(0, 1)$. Iterating the recursion and taking into account $(\hat{\alpha} + \alpha_1)\eta \leq 3$ directly yields (8.10).

Proof of Claim 2: In light of the expression (8.11), we aim to apply Corollary A.3 with $h(x) := \varphi(x) - \varphi(\bar{x})$, $D_t := \frac{1}{2}\mathbb{E}\|x_t - \bar{x}\|^2$, $c_1 := \alpha_2 - \gamma\beta$, and $c_2 := \alpha_1 - \gamma\beta$. The assumption $\eta \leq \min\{\frac{1}{(\alpha_1 - \gamma\beta)^+}, \frac{1}{(\alpha_2 - \gamma\beta)^+}\}$ directly implies $1 - c_1\eta > 0$ and $1 + c_2\eta > 0$. An application of Corollary A.3 yields the estimate:

$$\mathbb{E}[\varphi(\hat{x}_t) - \varphi(\bar{x})] \leq \left(1 - \frac{\hat{\alpha}\eta}{1 + (\alpha_1 - \gamma\beta)\eta}\right)^t \left(\varphi(x_0) - \varphi(\bar{x}) + \frac{\hat{\alpha}}{2}\|x_0 - \bar{x}\|^2\right) + \sigma_0^2\eta.$$

Lemma 8.1 in turn implies $\frac{\hat{\alpha}}{2}\|x_0 - \bar{x}\|^2 \leq \varphi(x_0) - \varphi(\bar{x})$. Taking into account $(\alpha_1 - \gamma\beta)\eta \leq 1$ completes the proof. \square

The following corollary (formal version of Theorem 4.6) obtains efficiency guarantees by combining Algorithm 4 with the geometric decay schedule (Algorithm 8).

Corollary 8.7 (Efficiency of Algorithm 4 with geometrically decaying schedule).

Suppose that Assumptions 2, 4(b), and 9 hold. Then the following are true.

1. **(Distance)** *Suppose we are in the regime $\frac{\gamma\beta}{\alpha_1 + \alpha_2} < 1$ and that we have available an estimate $\Delta \geq \|x_0 - \bar{x}\|^2$. Define the augmented strong convexity parameter $\hat{\alpha} = \alpha_1 + \alpha_2 - \gamma\beta$. Then Algorithm 4 may be augmented with the geometric decay schedule (Algorithm 8) under the identification*

$$c = \frac{\hat{\alpha}}{2}, \quad C = 1, \quad h(x) = \|x - \bar{x}\|^2, \quad \delta_0 = \min\left\{\frac{1}{2L}, \frac{1}{\alpha_1}, \frac{1}{\alpha_2}\right\}, \quad D = \frac{\sigma_0^2}{2\hat{\alpha}}.$$

The resulting procedure will generate a point x satisfying $\mathbb{E}\|x - \bar{x}\|^2 \leq \varepsilon$ using

$$\mathcal{O}\left(\left(\frac{L + \alpha_1 + \alpha_2}{\hat{\alpha}}\right) \log\left(\frac{\Delta}{\varepsilon}\right) + \frac{\sigma_0^2}{\hat{\alpha}^2\varepsilon}\right) \quad \text{samples.}$$

2. **(Function value)** *Suppose we are in the regime $\frac{\gamma\beta}{\alpha_1 + \alpha_2} < \frac{1}{2}$ and that we have available an estimate $\Delta \geq \varphi(x_0) - \varphi(\bar{x})$. Define the augmented strong convexity parameter $\hat{\alpha} = \alpha_1 + \alpha_2 - 2\gamma\beta$. Then Algorithm 4 may be augmented with the geometric decay schedule (Algorithm 8) under the identification*

$$c = \frac{\hat{\alpha}}{2}, \quad C = 2, \quad h(x) = \varphi(x) - \varphi(\bar{x}), \quad \delta_0 = \min\left\{\frac{1}{2L}, \frac{1}{(\alpha_1 - \gamma\beta)^+}, \frac{1}{(\alpha_2 - \gamma\beta)^+}\right\}, \quad D = \sigma_0^2.$$

The resulting procedure will generate a point x satisfying $\varphi(x) - \varphi(\bar{x}) \leq \varepsilon$ using

$$\mathcal{O}\left(\frac{L + \alpha_1 + \alpha_2}{\hat{\alpha}} \log\left(\frac{\Delta}{\varepsilon}\right) + \frac{\sigma_0^2}{\hat{\alpha}\varepsilon}\right) \quad \text{samples.}$$

Proof. This is immediate from Theorem 8.6 and Lemma B.2. \square

Efficiency guarantees of the stochastic proximal point, proximal gradient, and clipped gradient methods under Assumptions 1, 2, 3(c), 4(b) follow directly from Corollary 8.7 and the values of α_1 and α_2 in Table 3.

9 Inexact repeated minimization

All stochastic algorithms we have discussed so far in each iteration t apply a single step of a standard algorithm on the static problem $\text{St}(\mathcal{D}(x_t))$. An alternative strategy is to apply an algorithm to the static problem $\text{St}(\mathcal{D}(x_t))$ for a moderate number of iterations and switch the distribution at the end of the run. Algorithms based on this principle can be interpreted as inexact repeated minimization. Such algorithms can be superior in the typical settings where changing the distribution governing the static problem may be much costlier than drawing a sample. We will informally call the process of changing the distribution from $\mathcal{D}(x)$ to $\mathcal{D}(y)$ as *deployment*. In this section, we will show that it is possible to maintain the sample complexity derived in previous sections, while using a number of deployments that is only logarithmic in the problem parameters.

9.1 Model-based algorithms

We begin by analyzing Algorithm 5, which effectively performs repeated minimization with each subproblem solved approximately by a model-based algorithm (Algorithm 4). We describe two versions of the procedure. The first version uses the last iterate of each run to warmstart the next run, while the second version instead uses the running average of the iterates. Throughout the section, we suppose that Assumptions 2, 4(b), and 9 hold.

<p>Algorithm 5: Stagewise based Minimization</p> <p>Input: initial u_0, sequences $\{\eta_j\}_{j=0}^J \subset (0, \infty)$ and $\{J_t\}_{t=0}^T \subset \mathbb{N}$.</p> <p>for $t = 0, 1, \dots, T$ do</p> <p style="padding-left: 20px;">Set $x_0 = u_t$</p> <p style="padding-left: 20px;">for $j = 1, \dots, J_t$ do</p> <p style="padding-left: 40px;">Sample $S_j \stackrel{i.i.d.}{\sim} \mathcal{D}(x_0)$</p> <p style="padding-left: 40px;">Set $x_j = \arg \min_x \left\{ \ell_{x_0}(x, S_j) + r(x) + \frac{1}{2\eta_j} \ x - x_{j-1}\ ^2 \right\}$</p> <p style="padding-left: 20px;">end</p> <p style="padding-left: 20px;">Set</p> $u_{t+1} = \begin{cases} x_{J_t} & , \text{ if Version I} \\ \hat{\Gamma}_{J_t} x_0 + \sum_{j=1}^{J_t} \frac{\hat{\Gamma}_{J_t} \hat{\delta}_j}{\hat{\Gamma}_j} x_j & , \text{ if Version II} \end{cases}$ <p style="padding-left: 20px;">where we define $\hat{\delta}_j = \frac{(\alpha_1 + \alpha_2 - \gamma\beta)\eta_j}{1 + (\alpha_1 - \gamma\beta)\eta_j}$ and $\hat{\Gamma}_j = \prod_{i=1}^j (1 - \hat{\delta}_i)$.</p> <p>end</p>

The following theorem shows a one-step improvement guarantee for Algorithm 5, which explicitly balances the impact of the number of inner iterations J_t and the step-size parameters η_j on the overall performance.

Theorem 9.1 (One epoch improvement). *Suppose we are in the regime $\frac{\gamma\beta}{\alpha_1 + \alpha_2} < 1$.*

1. **(Version I)** Choose a sequence $\eta_j \leq \frac{1}{2L}$. Then the iterates u_t generated by Version I of

Algorithm 5 satisfy

$$\mathbb{E}\|u_{t+1} - \bar{x}\|^2 \leq \left(\Gamma_{J_t} + \frac{\gamma\beta}{2\alpha_1 + 2\alpha_2 - \gamma\beta}\right) \mathbb{E}\|u_t - \bar{x}\|^2 + \frac{2\sigma_0^2}{2\alpha_1 + 2\alpha_2 - \gamma\beta} \sum_{j=1}^{J_t} \frac{\eta_j \delta_j \Gamma_{J_t}}{\Gamma_j}, \quad (9.1)$$

where we set $\delta_j = \frac{(2\alpha_1 + 2\alpha_2 - \gamma\beta)\eta_j}{1 + \eta_j(2\alpha_1 + \alpha_2 - \gamma\beta)}$ and $\Gamma_j = \prod_{i=1}^j (1 - \delta_i)$.

2. **(Version II)** Choose a sequence $\eta_j \leq \min\{\frac{1}{2L}, \frac{1}{(\gamma\beta - \alpha_1)^+}, \frac{1}{\alpha_2}\}$. Then the iterates u_t generated by Version II of Algorithm 5 satisfy

$$\mathbb{E}[\varphi(u_{t+1}) - \varphi(\bar{x})] \leq \left(2\hat{\Gamma}_{J_t} + \frac{\gamma\beta}{\alpha_1 + \alpha_2}\right) \mathbb{E}[\varphi(u_t) - \varphi(\bar{x})] + \sigma_0^2 \sum_{j=1}^{J_t} \frac{\eta_j \hat{\delta}_j \hat{\Gamma}_{J_t}}{\hat{\Gamma}_j}. \quad (9.2)$$

Proof. Fix an index $t \in \{0, \dots, T\}$. We will consider the outcome of the inner loop $j = 1, \dots, J_t$. To simplify notation therefore let us drop the subscript t from u_t . Setting $y = \bar{x}$ in Lemma 8.2 yields the guarantee:

$$\eta_j \mathbb{E}[\varphi_u(x_j) - \varphi_u(\bar{x})] \leq \frac{1 - \alpha_2 \eta_j}{2} \mathbb{E}\|x_{j-1} - \bar{x}\|^2 - \frac{1 - \alpha_1 \eta_j}{2} \mathbb{E}\|x_j - \bar{x}\|^2 + \sigma_0^2 \eta_j^2, \quad (9.3)$$

for all $j = 1, \dots, J_t$. The function gap inequality (3.2) in turn implies

$$\begin{aligned} \varphi_u(x_j) - \varphi_u(\bar{x}) &\geq \varphi(x_j) - \varphi(\bar{x}) - \gamma\beta \|x_j - \bar{x}\| \cdot \|y - \bar{x}\| \\ &\geq \varphi(x_j) - \varphi(\bar{x}) - \frac{\gamma\beta}{2} \|x_j - \bar{x}\|^2 - \frac{\gamma\beta}{2} \|u - \bar{x}\|^2, \end{aligned}$$

where the last estimate follows from Young's inequality. Combining with (9.3) and rearranging, we conclude

$$\eta_j \mathbb{E}[\varphi(x_j) - \varphi(\bar{x})] \leq \frac{1 - \alpha_2 \eta_j}{2} \mathbb{E}\|x_{j-1} - \bar{x}\|^2 - \frac{1 + (\alpha_1 - \gamma\beta)\eta_j}{2} \mathbb{E}\|x_j - \bar{x}\|^2 + \frac{2\sigma_0^2 \eta_j^2 + \gamma\beta \eta_j \mathbb{E}\|u - \bar{x}\|^2}{2}. \quad (9.4)$$

Suppose now that we are running version I of Algorithm 5. Lower-bounding the left-side of (9.4) using strong convexity, $\varphi(x_j) - \varphi(\bar{x}) \geq \frac{\alpha_1 + \alpha_2}{2} \|x_j - \bar{x}\|^2$ (Lemma 8.1), and rearranging yields

$$\mathbb{E}\|x_j - \bar{x}\|^2 \leq \left(\frac{1 - \alpha_2 \eta_j}{1 + \eta_j(2\alpha_1 + \alpha_2 - \gamma\beta)}\right) \mathbb{E}\|x_{j-1} - \bar{x}\|^2 + \frac{2\sigma_0^2 \eta_j^2 + \eta_j \gamma\beta \mathbb{E}\|u - \bar{x}\|^2}{1 + \eta_j(2\alpha_1 + \alpha_2 - \gamma\beta)}.$$

The coefficient of $\mathbb{E}\|x_{j-1} - \bar{x}\|^2$ is precisely $1 - \delta_j$. Unrolling the recursion, we conclude

$$\mathbb{E}\|x_{J_t} - \bar{x}\|^2 \leq \Gamma_{J_t} \mathbb{E}\|x_0 - \bar{x}\|^2 + \sum_{j=1}^{J_t} \frac{2\sigma_0^2 \eta_j^2 + \eta_j \gamma\beta \mathbb{E}\|u - \bar{x}\|^2}{1 + \eta_j(2\alpha_1 + \alpha_2 - \gamma\beta)} \cdot \frac{\Gamma_{J_t}}{\Gamma_j}. \quad (9.5)$$

Next to simplify the sum, algebraic manipulations yield

$$\sum_{j=1}^{J_t} \frac{\eta_j}{1 + \eta_j(2\alpha_1 + \alpha_2 - \gamma\beta)} \cdot \frac{\Gamma_{J_t}}{\Gamma_j} = \frac{1}{2\alpha_1 + 2\alpha_2 - \gamma\beta} \sum_{j=1}^{J_t} \frac{\delta_j \Gamma_{J_t}}{\Gamma_j} = \frac{1 - \Gamma_{J_t}}{2\alpha_1 + 2\alpha_2 - \gamma\beta}, \quad (9.6)$$

where the last equality uses Lemma A.1. Note also the expression

$$\sum_{j=1}^{J_t} \frac{\eta_j^2}{1 + \eta_j(2\alpha_1 + \alpha_2 - \gamma\beta)} \cdot \frac{\Gamma_{J_t}}{\Gamma_j} = \frac{1}{2\alpha_1 + 2\alpha_2 - \gamma\beta} \sum_{j=1}^{J_t} \frac{\eta_j \delta_j \Gamma_{J_t}}{\Gamma_j}. \quad (9.7)$$

Combining (9.6), (9.7), (9.5), and the equality $x_0 = u$ completes the proof of (9.1).

Suppose now that we are running version II of Algorithm 5. Then applying Lemma A.2 to the recursion (9.4) directly implies

$$\mathbb{E}[\varphi(u_{t+1}) - \varphi(\bar{x})] \leq \hat{\Gamma}_{J_t} \left(\mathbb{E}[\varphi(x_0) - \varphi(\bar{x})] + \frac{\alpha_1 + \alpha_2 - \gamma\beta}{2} \|x_0 - \bar{x}\|^2 + \sum_{j=1}^{J_t} \frac{(\alpha_1 + \alpha_2 - \gamma\beta)(2\sigma_0^2 \eta_j^2 + \gamma\beta \eta_j \|u - \bar{x}\|^2)}{2\hat{\Gamma}_j(1 + (\alpha_1 - \gamma\beta)\eta_j)} \right).$$

To simplify the right side, observe first $\frac{\alpha_1 + \alpha_2 - \gamma\beta}{2} \|x_0 - \bar{x}\|^2 \leq \varphi(x_0) - \varphi(\bar{x})$ (Lemma 8.1). Next the same argument that justified (9.6) and (9.7) implies the expressions:

$$\sum_{j=1}^{J_t} \frac{(\alpha_1 - \alpha_2 - \gamma\beta)\eta_j}{1 + (\alpha_1 - \gamma\beta)\eta_j} \cdot \frac{\hat{\Gamma}_{J_t}}{\hat{\Gamma}_j} = 1 - \hat{\Gamma}_{J_t} \quad \text{and} \quad \sum_{j=1}^{J_t} \frac{(\alpha_1 + \alpha_2 - \gamma\beta)\eta_j^2}{1 + (\alpha_1 - \gamma\beta)\eta_j} \cdot \frac{\hat{\Gamma}_{J_t}}{\hat{\Gamma}_j} = \sum_{j=1}^{J_t} \frac{\eta_j \hat{\delta}_j \hat{\Gamma}_{J_t}}{\hat{\Gamma}_j}.$$

The claimed estimate (9.2) follows immediately. \square

Looking at the estimate (9.2), it is natural to choose J_t such that the estimate, $2\hat{\Gamma}_{J_t} + \frac{\gamma\beta}{\alpha_1 + \alpha_2} \leq \frac{1}{2}(1 + \frac{\gamma\beta}{\alpha_1 + \alpha_2})$, holds. The following result summarizes the guarantees of such a procedure when the stepsizes η_j are constant.

Corollary 9.2. *Suppose we are in the regime $\frac{\gamma\beta}{\alpha_1 + \alpha_2} < 1$.*

1. **(Version I)** *Choose a constant parameter $\eta \leq \frac{1}{2L}$ and set*

$$J_t = \left\lceil \left(1 + \frac{1}{(2\alpha_1 + 2\alpha_2 - \gamma\beta)\eta} \right) \log \left(\frac{2\alpha_1 + 2\alpha_2 - \gamma\beta}{\alpha_1 + \alpha_2 - \gamma\beta} \right) \right\rceil \quad \forall t \geq 0.$$

Then the iterates u_t generated by Version I of Algorithm 5 satisfy

$$\mathbb{E}\|u_t - \bar{x}\|^2 \leq \left(\frac{1}{2} \left(1 + \frac{\gamma\beta}{2\alpha_1 + 2\alpha_2 - \gamma\beta} \right) \right)^t \mathbb{E}\|u_0 - \bar{x}\|^2 + \frac{2\sigma_0^2 \eta}{\alpha_1 + \alpha_2 - \gamma\beta}. \quad (9.8)$$

2. **(Version II)** *Choose $\eta \leq \min\{\frac{1}{2L}, \frac{1}{(\gamma\beta - \alpha_1)^+}, \frac{1}{\alpha_2}\}$ and set*

$$J_t = \left\lceil \left(1 + \frac{1}{(\alpha_1 + \alpha_2 - \gamma\beta)\eta} \right) \log \left(\frac{2(\alpha_1 + \alpha_2)}{\alpha_1 + \alpha_2 - \gamma\beta} \right) \right\rceil \quad \forall t \geq 0.$$

Then the iterates u_t generated by Version II of Algorithm 5 satisfy

$$\mathbb{E}[\varphi(u_t) - \varphi(\bar{x})] \leq \left(\frac{1}{2} \left(1 + \frac{\gamma\beta}{\alpha_1 + \alpha_2} \right) \right)^t \mathbb{E}[\varphi(u_0) - \varphi(\bar{x})] + \frac{2\sigma_0^2 \eta}{1 - \frac{\gamma\beta}{\alpha_1 + \alpha_2}}. \quad (9.9)$$

Proof. With the constant parameter η , the estimate (9.1) becomes

$$\mathbb{E}\|u_t - \bar{x}\|^2 \leq \left(\left(1 - \frac{(2\alpha_1 + 2\alpha_2 - \gamma\beta)\eta}{1 + \eta(2\alpha_1 + \alpha_2 - \gamma\beta)} \right)^{j_{t-1}} + \frac{\gamma\beta}{2\alpha_1 + 2\alpha_2 - \gamma\beta} \right) \mathbb{E}\|u_{t-1} - \bar{x}\|^2 + \frac{2\sigma_0^2 \eta}{2\alpha_1 + 2\alpha_2 - \gamma\beta},$$

where we used the equality $\sum_{j=1}^{j_{t-1}} \frac{\delta_j \Gamma_{j_{t-1}}}{\Gamma_j} = 1 - \Gamma_{j_{t-1}}$ (Lemma A.1). The definition of j_{t-1} ensures that the coefficient of $\mathbb{E}\|u_{t-1} - \bar{x}\|^2$ is at most $\frac{1}{2}(1 + \frac{\gamma\beta}{2\alpha_1 + 2\alpha_2 - \gamma\beta})$ and therefore

$$\mathbb{E}\|u_t - \bar{x}\|^2 \leq \frac{1}{2} \left(1 + \frac{\gamma\beta}{2\alpha_1 + 2\alpha_2 - \gamma\beta} \right) \mathbb{E}\|u_{t-1} - \bar{x}\|^2 + \frac{2\sigma_0^2 \eta}{2\alpha_1 + 2\alpha_2 - \gamma\beta},$$

Unrolling the recursion in t completes the proof of (9.8). Similarly, with the constant parameter η , the estimate (9.2) becomes

$$\mathbb{E}[\varphi(u_t) - \varphi(\bar{x})] \leq \left(2 \left(1 - \frac{\eta(\alpha_1 + \alpha_2 - \gamma\beta)}{1 + (\alpha_1 - \gamma\beta)\eta} \right)^{j_{t-1}} + \frac{\gamma\beta}{\alpha_1 + \alpha_2} \right) \mathbb{E}[\varphi(u_{t-1}) - \varphi(\bar{x})] + \sigma_0^2 \eta.$$

The definition of j_{t-1} ensures that the coefficient of $\mathbb{E}[\varphi(u_{t-1}) - \varphi(\bar{x})]$ is at most $\frac{1}{2}(1 + \frac{\gamma\beta}{\alpha_1 + \alpha_2})$. Iterating the recursion in t completes the proof of (9.9). \square

Consequently, we may equip Algorithm 5 with a geometrically decaying stepsize. The efficiency estimates of the resulting procedure are recorded in the following corollary (formal version of Theorem 4.7).

Corollary 9.3. *Suppose that Assumptions 2, 4(b), and 9 hold, and that we are in the regime $\frac{\gamma\beta}{\alpha_1 + \alpha_2} < 1$.*

1. **(Version I)** *Suppose we have available an estimate $\Delta \geq \|x_0 - \bar{x}\|^2$. Define the augmented strong convexity parameter $\hat{\alpha} = \alpha_1 + \alpha_2 - \gamma\beta$. Then Algorithm 5 (Version I) may be augmented with the geometric decay schedule (Algorithm 8) with parameters:*

$$\psi(\delta) \equiv \frac{1}{2} \left(1 - \frac{\gamma\beta}{2\alpha_1 + 2\alpha_2 - \gamma\beta} \right), \quad C = 1, \quad h = \|\cdot - \bar{x}\|^2, \quad D = \frac{2\sigma_0^2}{\alpha_1 + \alpha_2 - \gamma\beta}, \quad \delta_0 = \frac{1}{2L}.$$

The resulting procedure will generate a point x satisfying $\mathbb{E}[\|x - \bar{x}\|^2] \leq \varepsilon$ using

$$\mathcal{O} \left(\left(1 - \frac{\gamma\beta}{2\alpha_1 + 2\alpha_2 - \gamma\beta} \right)^{-1} \cdot \left(\log \left(\frac{\Delta}{\varepsilon} \right) + \log \left(\frac{\sigma_0^2}{\hat{\alpha}L\varepsilon} \right) \right) \right) \quad (9.10)$$

deployments and

$$\mathcal{O} \left(\left(1 - \frac{\gamma\beta}{2\alpha_1 + 2\alpha_2 - \gamma\beta} \right)^{-1} \log \left(\left(1 - \frac{\gamma\beta}{2\alpha_1 + 2\alpha_2 - \gamma\beta} \right)^{-1} \cdot \left(\frac{L}{\hat{\alpha}} \log \left(\frac{\Delta}{\varepsilon} \right) + \frac{\sigma_0^2}{\hat{\alpha}^2\varepsilon} \right) \right) \right) \quad (9.11)$$

samples.

2. **(Version II)** *Suppose we have available an estimate $\Delta \geq \varphi(x_0) - \min \varphi$. Define the augmented strong convexity parameter $\hat{\alpha} = \alpha_1 + \alpha_2 - \gamma\beta$. Then Algorithm 5 (Version II) may be augmented with the geometric decay schedule (Algorithm 8) with parameters:*

$$\psi(\delta) \equiv \frac{1}{2} \left(1 - \frac{\gamma\beta}{\alpha_1 + \alpha_2} \right), \quad C = 1, \quad h(\cdot) = \varphi(\cdot) - \varphi(\bar{x}), \quad D = \frac{2\sigma_0^2}{1 - \frac{\gamma\beta}{\alpha_1 + \alpha_2}}, \quad \delta_0 = \min \left\{ \frac{1}{2L}, \frac{1}{(\gamma\beta - \alpha_1)^+}, \frac{1}{\alpha_2} \right\}.$$

The resulting procedure will generate a point x satisfying $\mathbb{E}[\varphi(x) - \min \varphi] \leq \varepsilon$ using

$$\mathcal{O} \left(\left(1 - \frac{\gamma\beta}{\alpha_1 + \alpha_2} \right)^{-1} \cdot \left(\log \left(\frac{\Delta}{\varepsilon} \right) + \log \left(\frac{\sigma_0^2}{1 - \frac{\gamma\beta}{\alpha_1 + \alpha_2}} \cdot \frac{1}{L\varepsilon} \right) \right) \right)$$

deployments and

$$\mathcal{O} \left(\left(\left(1 - \frac{\gamma\beta}{\alpha_1 + \alpha_2} \right)^{-1} \right) \cdot \log \left(\left(1 - \frac{\gamma\beta}{\alpha_1 + \alpha_2} \right)^{-1} \right) \cdot \left(\frac{1}{\hat{\alpha}\delta_0} \log \left(\frac{\Delta}{\varepsilon} \right) + \frac{\sigma_0^2}{\left(1 - \frac{\gamma\beta}{\alpha_1 + \alpha_2} \right) \hat{\alpha}\varepsilon} \right) \right)$$

samples.

Proof. We prove the efficiency of Version I of Algorithm 5; the proof for version II is completely analogous. The procedure invokes the algorithm in $k = 0, 1, \dots, K$ stages, where $K = \lceil 1 + \log_2(\frac{D\eta_0}{\varepsilon}) \rceil$. Applying Lemma B.2, the number of deployments coincides with the sum

$$\sum_{k=0}^K T_k = \left\lceil \frac{1}{\psi(\delta_0)} \cdot \log\left(\frac{2C\Delta}{\varepsilon}\right) \right\rceil + \sum_{k=1}^K \left\lceil \frac{\log(4C)}{\psi(2^{-k}\delta_0)} \right\rceil.$$

Simplifying yields (9.10). The total number of samples used in stage k

$$T_k \cdot \left[\left(1 + \frac{1}{(2\alpha_1 + 2\alpha_2 - \gamma\beta)\eta_k} \right) \log\left(\frac{2\alpha_1 + 2\alpha_2 - \gamma\beta}{\alpha_1 + \alpha_2 - \gamma\beta}\right) \right] \quad (9.12)$$

where $T_0 = \left\lceil \frac{1}{\psi(\delta_0)} \cdot \log\left(\frac{2C\Delta}{\varepsilon}\right) \right\rceil$, $T_k = \left\lceil \frac{1}{\psi(\delta_k)} \cdot \log(4C) \right\rceil$ for $k \geq 1$, and $\eta_k = 2^{-k}\eta_0$. Observe $\sum_{k=1}^K \frac{1}{\eta_k} = \sum_{k=1}^K 2^{k+1}L \leq 2^{K+2}L \leq 16\frac{D\eta_0}{\varepsilon} \cdot L = \frac{16\sigma_0^2}{\hat{\alpha}\varepsilon}$. Summing the expressions (9.12) across $k = 0, 1, \dots, K$ yields (9.11). \square

9.2 Accelerated method

We next explore applying an accelerated stochastic gradient method within inexact repeated minimization. Throughout the section, we suppose that Assumptions 1, 2, 3(b), 4(b) hold. The accelerated method is summarized as Algorithm 6.

Algorithm 6: Stagewise Stochastic Accelerated Gradient	
Input: initial u_0 , integers $J, T \in \mathbb{N}$.	
for $t = 0, 1, \dots, T$ do	
Set $x_0 = u_t$	
for $j = 1, \dots, J$ do	
	Sample $z_t \sim \mathcal{D}(u_t)$
	$x_j = \text{prox}_{r/L}(y_{j-1} - \eta \nabla \ell(y_{j-1}, z_t))$
	$y_j = x_j + \frac{1 - \sqrt{\alpha/L}}{1 + \sqrt{\alpha/L}}(x_t - x_{t-1})$
end	
Set $u_{t+1} = x_J$	
end	

The idea of the argument is as follows. The convergence guarantees for the static stochastic accelerated gradient method [36, Corollary 13] directly imply

$$\mathbb{E}_t[\varphi_x(x_J) - \varphi_x(y)] \leq \left(1 - \sqrt{\frac{\alpha}{L}}\right)^J \cdot \left(\varphi_x(x_0) - \varphi_x(y) + \frac{\alpha}{2}\|x_J - y\|^2\right) + \frac{\sigma^2}{\sqrt{\alpha L}},$$

for all points y , where \mathbb{E}_t denotes the conditional expectation on u_t . The following general lemma shows how to translate efficiency estimates of this type for φ_x to a similar estimate for φ .

Theorem 9.4 (Reduction). *Suppose we are in the parameter regime $\rho < \frac{1}{2}$. Fix a point x and counter $j \in \mathbb{N}$ and suppose that an algorithm applied to the static problem $\text{St}(\mathcal{D}(x))$ generates a random point x_j satisfying*

$$\mathbb{E}[\varphi_x(x_j) - \varphi_x(y)] \leq C(1 - q)^j \left(\varphi_x(x) - \varphi_x(y) + \frac{\alpha}{2}\|x_j - y\|^2\right) + B \quad \forall y \in \mathbf{R}^d, \quad (9.13)$$

where $B, C \geq 0$ and $q \in (0, 1)$ are some constants. Then the estimate holds:

$$\mathbb{E}[\varphi(x_j) - \varphi(\bar{x})] \leq \frac{\frac{\gamma\beta}{\alpha} + 2C(1-q)^j}{1 - \frac{\gamma\beta}{\alpha} - C(1-q)^j} \cdot (\varphi(x) - \varphi(\bar{x})) + \frac{B}{1 - \frac{\gamma\beta}{\alpha} - C(1-q)^j}$$

Proof. Appealing to the gap deviation inequality (3.2), we compute

$$\begin{aligned} \varphi(x_j) - \varphi(\bar{x}) &\leq \varphi_x(x_j) - \varphi_x(\bar{x}) + \gamma\beta\|x_j - \bar{x}\| \cdot \|x - \bar{x}\| \\ &\leq \varphi_x(x_j) - \varphi_x(\bar{x}) + \frac{\gamma\beta}{2}\|x_j - \bar{x}\|^2 + \frac{\gamma\beta}{2}\|x - \bar{x}\|^2 \end{aligned} \quad (9.14)$$

$$\leq \frac{\gamma\beta}{\alpha}(\varphi(x) - \varphi(\bar{x})) + \varphi_x(x_j) - \varphi_x(\bar{x}) + \frac{\gamma\beta}{2}\|x_j - \bar{x}\|^2, \quad (9.15)$$

where (9.14) follows from Young's inequality and (9.15) follows from strong convexity of φ . Next, taking into account (9.13) with $y = \bar{x}$, we conclude

$$\begin{aligned} \mathbb{E}[\varphi(x_j) - \varphi(\bar{x})] &\leq \frac{\gamma\beta}{\alpha}(\varphi(x) - \varphi(\bar{x})) + C(1-q)^j \left(\varphi_x(x) - \varphi_x(\bar{x}) + \frac{\alpha}{2}\|x - \bar{x}\|^2 \right) \\ &\quad + B + \frac{\gamma\beta}{2}\mathbb{E}\|x_j - \bar{x}\|^2. \end{aligned}$$

The gap deviation inequality (3.2) guarantees

$$[\varphi_x(x) - \varphi_x(\bar{x})] - [\varphi(x) - \varphi(\bar{x})] \leq \gamma\beta\|x - \bar{x}\|^2 \leq \frac{2\gamma\beta}{\alpha}(\varphi(x) - \varphi(\bar{x})),$$

while strong convexity of φ implies $\frac{1}{2}\|x_j - \bar{x}\|^2 \leq \frac{1}{\alpha}(\varphi(x_j) - \varphi(\bar{x}))$. We therefore deduce

$$\mathbb{E}[\varphi(x_j) - \varphi(\bar{x})] \leq \frac{\frac{\gamma\beta}{\alpha} + C(1-q)^j \left(1 + \frac{2\gamma\beta}{\alpha}\right)}{1 - \frac{\gamma\beta}{\alpha} - C(1-q)^j} \cdot (\varphi(x) - \varphi(\bar{x})) + \frac{B}{1 - \frac{\gamma\beta}{\alpha} - C(1-q)^j}.$$

Invoking the upper bound $\frac{2\gamma\beta}{\alpha} \leq 1$ completes the proof. \square

Combining Algorithm 6 with the minibatch restart strategy (Algorithm 7) yields an overall scheme with the following efficiency estimates (formal version of Theorem 4.8).

Theorem 9.5 (Accelerated Stochastic Gradient). *Suppose that Assumptions 1, 2, 3(b), 4(b) hold and that we are in the regime $\rho < \frac{1}{2}$. Set the number of inner iterations $J = \sqrt{\frac{L}{\alpha}} \log\left(\frac{4}{\frac{1}{2}-\rho}\right)$. Then the iterates u_t generated by Algorithm 6 satisfy*

$$\mathbb{E}[\varphi(u_t) - \varphi(u_0)] \leq \left(\frac{1}{2(1-\rho)}\right)^t \cdot \mathbb{E}[\varphi(u_0) - \varphi(\bar{x})] + \frac{32\sigma^2}{5(1 - \frac{\rho}{1-\rho})\sqrt{\alpha L}}. \quad (9.16)$$

Consequently, if we have available an estimate $\Delta \geq \|x_0 - \bar{x}\|^2$, then Algorithm 6 may be combined with the minibatch restart strategy (Algorithm 7) by setting

$$h(\cdot) = \varphi(\cdot) - \varphi(\bar{x}), \quad \tau = 1 - \frac{1}{2(1-\rho)}, \quad C = 1, \quad B = \frac{32\sigma^2}{5(1 - \frac{\rho}{1-\rho})\sqrt{\alpha L}}.$$

The resulting procedure will generate a point x satisfying $\mathbb{E}[\varphi(x) - \min \varphi] \leq \varepsilon$ using

$$\mathcal{O}\left(\left(1 - \frac{1}{2(1-\rho)}\right)^{-1} \left(\log\left(\frac{\Delta}{\varepsilon}\right) + \log\left(\frac{\sigma^2}{(1 - \frac{\rho}{1-\rho})\varepsilon\sqrt{\alpha L}}\right)\right)\right) \quad (9.17)$$

deployments and

$$\mathcal{O} \left(\left(1 - \frac{1}{2(1-\rho)}\right)^{-1} \cdot \left(\sqrt{\frac{L}{\alpha}} \log \left(\frac{\Delta}{\varepsilon} \right) + \frac{\sigma^2}{(1 - \frac{\rho}{1-\rho})\alpha\varepsilon} \right) \right) \quad (9.18)$$

samples.

Proof. Fix an index $t = 0, 1, \dots, T$ and set $x = u_t$. The convergence guarantees for the static stochastic accelerated gradient method [36, Corollary 13] directly imply

$$\mathbb{E}_t[\varphi_x(x_J) - \varphi_x(y)] \leq \left(1 - \sqrt{\frac{\alpha}{L}}\right)^J \cdot \left(\varphi_x(x_0) - \varphi_x(y) + \frac{\alpha}{2} \|x_J - y\|^2 \right) + \frac{\sigma^2}{\sqrt{\alpha L}},$$

for all points y , where \mathbb{E}_t denotes the conditional expectation on u_t . Theorem 9.4 along with the tower rule for expectation therefore implies

$$\mathbb{E}_t[\varphi(u_{t+1}) - \varphi(\bar{x})] \leq \frac{\frac{\gamma\beta}{\alpha} + 2(1 - \sqrt{\alpha/L})^J}{1 - \frac{\gamma\beta}{\alpha} - (1 - \sqrt{\alpha/L})^J} \cdot (\varphi(u_t) - \varphi(\bar{x})) + \frac{\sigma^2/\sqrt{\alpha L}}{1 - \frac{\gamma\beta}{\alpha} - (1 - \sqrt{\alpha/L})^J}.$$

The choice J guarantees that the coefficient in front of $\varphi(u_t) - \varphi(\bar{x})$ is at most $\frac{1}{2(1-\rho)}$ and the inequality $1 - \rho - (1 - \tau)^J \geq \frac{3}{4} - \frac{7}{8}\rho \geq \frac{5}{16}$ holds. We therefore conclude

$$\mathbb{E}_t[\varphi(u_{t+1}) - \varphi(\bar{x})] \leq \frac{1}{2(1-\rho)} \cdot (\varphi(u_t) - \varphi(\bar{x})) + \frac{16\sigma^2}{5\sqrt{\alpha L}}.$$

Using the tower rule for expectations and iterating the recursion directly yields (9.16).

Following the notation of Lemma B.1, the number of deployments is given by

$$\sum_{k=0}^K T_k = \left\lceil \sqrt{\frac{1}{\tau}} \log \left(\frac{2\Delta}{\varepsilon} \right) \right\rceil + \left\lceil 1 + \log_2 \left(\frac{32\sigma^2}{5(1 - \frac{\rho}{1-\rho})\varepsilon\sqrt{\alpha L}} \right) \right\rceil \cdot \left\lceil \sqrt{\frac{1}{\tau}} \log(4) \right\rceil.$$

The estimate (9.17) follows immediately. The total number of samples used is

$$J \cdot \sum_{k=0}^K m_k T_k = \mathcal{O} \left(\frac{1}{\tau\sqrt{\kappa}} \log \left(\frac{4}{\frac{1}{2} - \rho} \right) \left(\log \left(\frac{\Delta}{\varepsilon} \right) + \frac{B}{\varepsilon} \right) \right).$$

The claimed estimate (9.18) follows immediately. \square

Thus, in the nearly optimal parameter regime $\rho < \frac{1}{2}$, Algorithm 6 enjoys the same sample efficiency as its online counterpart, while requiring a number of deployments that is only logarithmic in the problem parameters.

Acknowledgements: We thank Celestine Mendler-Dünner, Moritz Hardt, Juan C. Perdomo, and Tijana Zrnic for useful comments and suggestions.

References

- [1] Shabbir Ahmed. *Strategic planning under uncertainty: Stochastic integer programming approaches*. University of Illinois at Urbana-Champaign, 2000.

- [2] Ahmad Ajalloeian and Sebastian U Stich. Analysis of sgd with biased gradient estimators. *arXiv preprint arXiv:2008.00051*, 2020.
- [3] Hilal Asi and John C Duchi. Stochastic (approximate) proximal point methods: Convergence, optimality, and adaptivity. *SIAM Journal on Optimization*, 29(3):2257–2290, 2019.
- [4] Necdet Serhat Aybat, Alireza Fallah, Mert Gurbuzbalaban, and Asuman Ozdaglar. A universally optimal multistage accelerated stochastic gradient method. In *Advances in Neural Information Processing Systems*, pages 8525–8536, 2019.
- [5] Peter L Bartlett. Learning with a slowly changing distribution. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 243–252, 1992.
- [6] Peter L Bartlett, Shai Ben-David, and Sanjeev R Kulkarni. Learning changing concepts by exploiting the structure of change. *Machine Learning*, 41(2):153–174, 2000.
- [7] Yahav Bechavod, Katrina Ligett, Zhiwei Steven Wu, and Juba Ziani. Causal feature discovery through strategic modification. *arXiv preprint arXiv:2002.07024*, 2020.
- [8] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.*, 2(1):183–202, 2009.
- [9] Amir Beck. *First-order methods in optimization*. SIAM, 2017.
- [10] Dimitri P Bertsekas. Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334, 1997.
- [11] Omar Besbes, Yonatan Gur, and Assaf Zeevi. Non-stationary stochastic optimization. *Operations research*, 63(5):1227–1244, 2015.
- [12] Omar Besbes, Yonatan Gur, and Assaf Zeevi. Optimal exploration–exploitation in a multi-armed bandit problem with non-stationary rewards. *Stochastic Systems*, 9(4):319–337, 2019.
- [13] Michael Brückner, Christian Kanzow, and Tobias Scheffer. Static prediction games for adversarial learning problems. *The Journal of Machine Learning Research*, 13(1):2617–2654, 2012.
- [14] Sébastien Bubeck. Convex optimization: Algorithms and complexity. *arXiv preprint arXiv:1405.4980*, 2014.
- [15] J. Cutler, D. Drusvyatskiy, and Z. Harchaoui. Stochastic optimization under time drift: iterate averaging, step decay, and high probability guarantees. Accepted at NeurIPS 2021, arXiv:2108.07356, 2021.
- [16] Nilesh Dalvi, Pedro Domingos, Sumit Sanghai, and Deepak Verma. Adversarial classification. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 99–108, 2004.
- [17] Damek Davis and Dmitriy Drusvyatskiy. Stochastic model-based minimization of weakly convex functions. *SIAM Journal on Optimization*, 29(1):207–239, 2019.
- [18] Olivier Devolder, François Glineur, and Yurii Nesterov. First-order methods of smooth convex optimization with inexact oracle. *Mathematical Programming*, 146(1-2):37–75, 2014.

- [19] John Duchi and Yoram Singer. Efficient online and batch learning using forward backward splitting. *Journal of Machine Learning Research*, 10:2899–2934, 2009.
- [20] Jitka Dupacová. Optimization under exogenous and endogenous uncertainty. *University of West Bohemia in Pilsen*, 2006.
- [21] Pavel Dvurechensky, Alexander Gasnikov, and Dmitry Kamzolov. Universal intermediate gradient method for convex problems with inexact oracle. *arXiv preprint arXiv:1712.06036*, 2017.
- [22] Yu. Ermoliev. Stochastic quasigradient methods. In Yu. Ermoliev and R. J-B Wets, editors, *Numerical Techniques for Stochastic Optimization*, chapter 6, pages 141–185. Springer, 1988.
- [23] Gerald B Folland. *A Guide to Advanced Real Analysis*. Number 37. American Mathematical Soc., 2014.
- [24] A. A. Gaivoronskii. Nonstationary stochastic programming problems. *Cybernetics*, 14:575–579, 1978.
- [25] João Gama, Indrè Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM computing surveys (CSUR)*, 46(4):1–37, 2014.
- [26] Saeed Ghadimi and Guanghui Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization i: A generic algorithmic framework. *SIAM Journal on Optimization*, 22(4):1469–1492, 2012.
- [27] Saeed Ghadimi and Guanghui Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, ii: shrinking procedures and optimal algorithms. *SIAM Journal on Optimization*, 23(4):2061–2089, 2013.
- [28] Moritz Hardt, Nimrod Megiddo, Christos Papadimitriou, and Mary Wootters. Strategic classification. In *Proceedings of the 2016 ACM conference on innovations in theoretical computer science*, pages 111–122, 2016.
- [29] Elad Hazan. Introduction to online convex optimization. *Foundations and Trends in Optimization*, 2(3-4):157–325, 2016.
- [30] Lars Hellemo, Paul I Barton, and Asgeir Tomasgard. Decision-dependent probabilities in stochastic programs with recourse. *Computational Management Science*, 15(3):369–395, 2018.
- [31] Tore W Jonsbråten, Roger JB Wets, and David L Woodruff. A class of stochastic programs with decision dependent random elements. *Annals of Operations Research*, 82:83–106, 1998.
- [32] Kaggle. *Give me some credit*. Featured prediction competition, 2012. <https://www.kaggle.com/c/GiveMeSomeCredit/data>.
- [33] Leonid Vitalyevich Kantorovich and Gennady S Rubinstein. On a space of completely additive functions. *Vestnik Leningrad. Univ*, 13(7):52–59, 1958.
- [34] Anthony Kuh, Thomas Petsche, and Ronald L Rivest. Learning time-varying concepts. In *Advances in Neural Information Processing Systems*, pages 183–189, 1991.

- [35] Andrei Kulunchakov and Julien Mairal. Estimate sequences for stochastic composite optimization: Variance reduction, acceleration, and robustness to noise. *arXiv preprint arXiv:1901.08788*, 2019.
- [36] Andrei Kulunchakov and Julien Mairal. A generic acceleration framework for stochastic composite optimization. In *Advances in Neural Information Processing Systems*, pages 12556–12567, 2019.
- [37] Guanghui Lan. An optimal method for stochastic composite optimization. *Mathematical Programming*, 133(1-2):365–397, 2012.
- [38] Brendan McMahan. A survey of algorithms and analysis for adaptive online learning. *Journal of Machine Learning Research*, 18:1–50, 2017.
- [39] Celestine Mendler-Dünner, Juan C Perdomo, Tijana Zrnic, and Moritz Hardt. Stochastic optimization for performative prediction. *arXiv preprint arXiv:2006.06887*, 2020.
- [40] Smitha Milli, John Miller, Anca D. Dragan, and Moritz Hardt. The social cost of strategic classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 230–239, January 2019.
- [41] Y. Nesterov. *Introductory lectures on convex optimization*, volume 87 of *Applied Optimization*. Kluwer Academic Publishers, Boston, MA, 2004. A basic course.
- [42] Yu. Nesterov. A method for solving the convex programming problem with convergence rate $O(1/k^2)$. *Dokl. Akad. Nauk SSSR*, 269(3):543–547, 1983.
- [43] Yurii Nesterov. Primal-dual subgradient methods for convex problems. *Mathematical Programming*, 120(1):221–259, 2009.
- [44] Yurii Nesterov. *Lectures on convex optimization*, volume 137. Springer, 2018.
- [45] Juan C Perdomo, Tijana Zrnic, Celestine Mendler-Dünner, and Moritz Hardt. Performative prediction. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2020. arXiv preprint arXiv:2002.06673.
- [46] R.T. Rockafellar and R.J-B. Wets. *Variational Analysis*. Grundlehren der mathematischen Wissenschaften, Vol 317, Springer, Berlin, 1998.
- [47] Werner Römisch. Stability of stochastic programming problems. In A. Ruszczyński and A. Shapiro, editors, *Handbooks in Operations Research and Management Science: Stochastic Programming*, volume 10, pages 483–554. Elsevier, 2003.
- [48] Reuven Y Rubinfeld and Alexander Shapiro. *Discrete event systems: Sensitivity analysis and stochastic optimization by the score function method*, volume 13. Wiley, 1993.
- [49] Ernest K Ryu and Stephen Boyd. Stochastic proximal iteration: a non-asymptotic improvement upon stochastic gradient descent. *Author website, early draft*, 2014.
- [50] Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2012.

- [51] Fedor Stonyakin, Alexander Gasnikov, Alexander Tyurin, Dmitry Pasechnyuk, Artem Agafonov, Pavel Dvurechensky, Darina Dvinskikh, Alexey Kroshnin, and Victorya Piskunova. Inexact model: A framework for optimization and variational inequalities. *arXiv preprint arXiv:1902.00990*, 2019.
- [52] Pravin Varaiya and RJ-B Wets. Stochastic dynamic optimization approaches and computation. 1988.
- [53] Lin Xiao. Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research*, 11:2543–2596, 2010.
- [54] Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th International Conference on Machine Learning (ICML)*, pages 928–936, Washington DC, 2003.
- [55] Indrė Žiliobaitė, Mykola Pechenizkiy, and João Gama. An overview of concept drift applications. In N. Japkowica and J. Stefanowski, editors, *Big Data Analysis: New Algorithms for a New Society*, pages 91–114. Springer, 2016.

A Averaging lemma

In this section, we investigate recursions of the form

$$\delta_t \cdot \mathbb{E}[h(x_t)] \leq (1 - c_1 \delta_t) D_{t-1} - (1 + c_2 \delta_t) D_t + \omega_t \quad \forall t \geq 1, \quad (\text{A.1})$$

where $h(\cdot)$ is a convex function, $c_1, c_2 \in \mathbf{R}$ are some constants, and the sequences $\{\delta_t\}_{t \geq 1}$, $\{D_t\}_{t \geq 0}$, $\{\omega_t\}_{t \geq 1}$ are nonnegative. The expectation is taken over randomness in the points x_t , which are usually the iterates produced by a stochastic algorithm. In typical circumstances $\delta_t > 0$ is a user-specified sequence (e.g. stepsize). Our goal is to determine the rate at which the value $h(\hat{x}_t)$ tends to zero, where \hat{x}_t is a running average of the iterates. Most of the material in this section follows the discussion in [35, Section A.2, A.3] and [26]. We begin with the following elementary lemma.

Lemma A.1. *Consider a sequence of weights $\{\delta_t\}_{t \geq 0}$ in $(0, 1)$. Then the partial products $\Gamma_t = \prod_{i=1}^t (1 - \delta_i)$ satisfy the equation $\sum_{i=1}^t \frac{\delta_i}{\Gamma_i} + 1 = \frac{1}{\Gamma_t}$.*

Proof. We proceed by induction. For $t = 1$, we have $\Gamma_1 = 1 - \delta_1$, which is the same as $\Gamma_1 + \delta_1 = 1$. Therefore, $\frac{\delta_1}{\Gamma_1} + 1 = \frac{\delta_1 + \Gamma_1}{\Gamma_1} = \frac{1}{\Gamma_1}$, meaning that the base case is true.

Now suppose that $\sum_{i=1}^t \frac{\delta_i}{\Gamma_i} + 1 = \frac{1}{\Gamma_t}$ holds. By definition, $\Gamma_{t+1} = \Gamma_t(1 - \delta_{t+1})$, which is the same as $\Gamma_{t+1} + \Gamma_t \delta_{t+1} = \Gamma_t$. Based on this equality, we have

$$\sum_{i=1}^{t+1} \frac{\delta_i}{\Gamma_i} + 1 = \sum_{i=1}^t \frac{\delta_i}{\Gamma_i} + 1 + \frac{\delta_{t+1}}{\Gamma_{t+1}} = \frac{1}{\Gamma_t} + \frac{\delta_{t+1}}{\Gamma_{t+1}} = \frac{\Gamma_{t+1} + \Gamma_t \delta_{t+1}}{\Gamma_t \Gamma_{t+1}} = \frac{\Gamma_t}{\Gamma_t \Gamma_{t+1}} = \frac{1}{\Gamma_{t+1}},$$

where the second equality used the induction hypothesis. This completes the proof. \square

The convergence guarantees will be stated in terms of the augmented weights

$$\hat{\delta}_t := \frac{\delta_t(c_1 + c_2)}{1 + c_2 \delta_t} \quad \text{and} \quad \hat{\Gamma}_t := \prod_{i=1}^t (1 - \hat{\delta}_i), \quad (\text{A.2})$$

and average iterates that are recursively defined as

$$\hat{x}_0 := x_0 \quad \text{and} \quad \hat{x}_t := (1 - \hat{\delta}_t)\hat{x}_{t-1} + \hat{\delta}_t x_t \quad \forall t \geq 1.$$

The following lemma establishes the sought upper bound on the values $h(\hat{x}_t)$. This result follows quickly by reducing to the special case $c_1 = 1$, $c_2 = 0$, for which the lemma was proved in [35, Lemma 12].

Lemma A.2 (Averaging). *Consider a convex function $h: \mathbf{R}^d \rightarrow \mathbf{R} \cup \{\infty\}$ and let $\{x_t\}_{t \geq 0} \subset \mathbf{R}^d$ be a sequence of random vectors in $\text{dom } h$. Suppose that there are constants $c_1, c_2 \in \mathbf{R}$ and nonnegative sequences $\{\delta_t\}_{t \geq 1}$, $\{D_t\}_{t \geq 0}$, and $\{\omega_t\}_{t \geq 1}$ satisfying (A.1). Suppose moreover the relations $c_1 + c_2 > 0$, $1 - c_1\delta_t > 0$, and $1 + c_2\delta_t > 0$ hold for all $t \geq 1$. Then the estimate holds:*

$$\frac{\mathbb{E}[h(\hat{x}_t)]}{c_1 + c_2} + D_t \leq \hat{\Gamma}_t \left(\frac{h(x)}{c_1 + c_2} + D_0 + \sum_{i=1}^t \frac{\omega_i}{\hat{\Gamma}_i(1 + c_2\delta_i)} \right). \quad (\text{A.3})$$

Proof. The special case of this lemma in the setting $c_1 = 1$ and $c_2 = 0$ was proved in [35, Lemma 12]. We will reduce the general case to this setting. To this end, dividing the inequality (A.1) through by $1 + c_2\delta_t$ yields

$$\hat{\delta}_t \cdot \mathbb{E}[\hat{h}(x_t)] \leq (1 - \hat{\delta}_t) D_{t-1} - D_t + \hat{\omega}_t,$$

where we set $\hat{h}(x) := \frac{h(x)}{c_1 + c_2}$ and $\hat{\omega}_t := \frac{\omega_t}{1 + c_2\delta_t}$. Noting the inclusion $\hat{\delta}_t \in (0, 1)$, an application of [35, Lemma 12] completes the proof. \square

Observe that the efficiency estimate (A.3) is guided by the augmented weights $\hat{\delta}_i$ defined in (A.2). Therefore, when applying Lemma A.2, it is most convenient to specify $\hat{\delta}_i$ rather than δ_i . For any desired value $\hat{\delta} \in (0, 1)$, we may then simply set $\delta = \frac{\hat{\delta}}{c_1 + c_2 - c_2\hat{\delta}}$. Typical choices of the sequence $\hat{\delta}_t$ and the corresponding products $\hat{\Gamma}_t$ are summarized in Table 4.

$\hat{\delta}_t$	$\hat{\delta}$	$\frac{1}{t+1}$	$\frac{2}{t+2}$	$\min\{\frac{1}{t+1}, \hat{\delta}\}$	$\min\{\frac{2}{t+2}, \hat{\delta}\}$
$\hat{\Gamma}_t$	$(1 - \hat{\delta})^t$	$\frac{1}{t+1}$	$\frac{2}{(t+1)(t+2)}$	$\begin{cases} (1 - \hat{\delta})^t & \text{if } t < t_0 \\ \frac{\hat{\Gamma}_{t_0-1} t_0}{t+1} & \text{if } t \geq t_0 \end{cases}$	$\begin{cases} (1 - \hat{\delta})^t & \text{if } t < t'_0 \\ \frac{\hat{\Gamma}_{t'_0-1} t'_0 (t'_0+1)}{(t+1)(t+2)} & \text{if } t \geq t'_0 \end{cases}$

Table 4: The table describes the values of $\hat{\Gamma}_t = \prod_{i=1}^t (1 - \hat{\delta}_i)$ under various common choices of $\hat{\delta}_t$. Here, we assume $\hat{\delta} \in (0, 1)$ and set $t_0 = \lceil \frac{1}{\hat{\delta}} - 1 \rceil$, $t'_0 = \lceil \frac{2}{\hat{\delta}} - 2 \rceil$.

It is now straightforward to evaluate the right side of (A.3) under the different choices of $\hat{\delta}_t$, as specified in Table 4. For simplicity, we record the resulting estimate only in the case that $\hat{\delta}_t$ is constant across the iterations.

Corollary A.3 (Constant parameter). *Assume the setting of Lemma A.2, and suppose that $\delta_t = \delta$ and $\omega_i = \omega$ are constant. Then the estimate holds:*

$$\mathbb{E}[h(\hat{x}_t)] + (c_1 + c_2)D_t \leq \left(\frac{1 - c_1\delta}{1 + c_2\delta} \right)^t (h(x) + (c_1 + c_2)D_0) + \frac{\omega}{\delta}. \quad (\text{A.4})$$

Proof. The result follows directly from Lemma A.2 and algebraic manipulations. Namely, we compute

$$\sum_{i=1}^t \frac{\hat{\Gamma}_i \omega}{\hat{\Gamma}_i(1+c_2\delta)} = \frac{\omega}{\delta(c_1+c_2)} \sum_{i=1}^t \frac{\hat{\Gamma}_i \delta(c_1+c_2)}{\hat{\Gamma}_i(1+c_2\delta)} = \frac{\omega}{\delta(c_1+c_2)} \sum_{i=1}^t \frac{\hat{\Gamma}_i \hat{\delta}}{\hat{\Gamma}_i} = \frac{\omega}{\delta(c_1+c_2)} \cdot (1 - \hat{\Gamma}_t),$$

where the last equality follows from Lemma A.1. An application of Lemma A.2 completes the proof. \square

B Stagewise scheme for improved efficiency

This section describes two restart schemes for improving the efficiency of constant-step stochastic algorithms. We follow the discussion in [36, Appendix B.1], though closely related ideas can be found for example in [4, 27]. There are two complementary approaches. The first is based on restarting the constant step algorithm with exponentially increasing minibatches in order to decrease the variance of gradient estimators. The second approach restarts the algorithm with geometrically decreasing step-sizes. We discuss these two strategies in turn.

B.1 Minibatch restart

Suppose that we have available a stochastic algorithm $\mathcal{A}(y, m, T)$ that generates a point y_T satisfying

$$\mathbb{E}[h(y_T)] \leq C(1-\tau)^T h(y_0) + \frac{B}{m}, \tag{B.1}$$

where h is a nonnegative function and $C, B > 0$ and $\tau \in (0, 1)$ are some constants. In concrete circumstances, $h(y)$ may denote the function gap $\varphi(y) - \min \varphi$ or the square distance to the solution $\|y - \bar{x}\|^2$; the point y_0 specifies the initialization and T is the number of iterations; $B > 0$ is proportional to the variance σ^2 of stochastic gradients used by the algorithm, which can be reduced by a factor of $m \in \mathbb{N}$ by using a minibatch of size m . Thus the overall sample complexity of the execution is proportional to $m \cdot T$. For example, the stochastic gradient method on a smooth strongly convex function φ satisfies (B.1) with $h(y) = \varphi(y) - \min \varphi$, $C = 1$, $B = \sigma^2/L$, and $\tau = \alpha/L$. If the stochastic gradients are used in minibatches of size m , the variance σ^2 shrinks by a factor of m .

The minibatch restart procedure, formally described in Algorithm 7, runs in stages by repeatedly calling $\mathcal{A}(y_k, m_k, T_k)$. Specifically, in stage k , the method sets $m_k = 2^k$ and finds a point y satisfying $\mathbb{E}[h(y_T)] \leq 2 \cdot \frac{B}{m_k}$, while using the output from the previous stage as a warmstart.

Algorithm 7: Minibatch restart
<p>Input: $y_0 \in \mathbf{R}^d$, $B, C > 0$, $\tau \in (0, 1)$, estimate $\Delta \geq h(y_0)$, accuracy $\varepsilon > 0$, algorithm $\mathcal{A}(y, m, T)$ satisfying (B.1).</p> <p>Initialize: Set $y = \mathcal{A}(y_0, m_0, T_0)$ with $m_0 = 1$ and $T_0 = \tau^{-1} \cdot \log(\frac{2C\Delta}{\varepsilon})$.</p> <p>Set $K = \lceil 1 + \log_2(\frac{B}{\varepsilon}) \rceil$.</p> <p>Step $k = 1, \dots, K$:</p> <p style="padding-left: 40px;">Set $y = \mathcal{A}(y, m_k, T_k)$ with $m_k = 2^k$, $T_k = \lceil \tau^{-1} \cdot \log(4C) \rceil$.</p> <p>Return y.</p>

The overall efficiency of the scheme is summarized in the following lemma. The proof can be found in [36, Appendix B.1].

Lemma B.1 (Minibatch restart). *The point y returned by Algorithm 7 satisfies $\mathbb{E}[h(y)] \leq \varepsilon$ and the efficiency estimate holds:*

$$\sum_{k=0}^K m_k T_k = \mathcal{O} \left(\frac{1}{\tau} \cdot \log \left(\frac{2C\Delta}{\varepsilon} \right) + \frac{B \log(4C)}{\tau \varepsilon} \right).$$

Algorithm 7 is effective when working with minibatches of the loss functions is computationally feasible. There are, however, important situations where each iteration using a minibatch becomes prohibitively expensive computationally. For example, a proximal point update on a single loss function $\ell(\cdot, z)$ may be computable in closed form, whereas computing the proximal point of an average $\frac{1}{s} \sum_{i=1}^s \ell(x, z_i)$ may be computationally challenging. The following section provides an alternative strategy to minibatching that is based on geometrically decreasing the step-size used by the algorithm.

B.2 Geometric decay schedule

Suppose that we have available a stochastic algorithm $\mathcal{A}(y_0, \delta, T)$, such that as long as $\delta < \delta_0$, the method generates a point y_T satisfying

$$\mathbb{E}[h(y_T)] \leq C(1 - \psi(\delta))^T h(y_0) + D\delta, \quad (\text{B.2})$$

where h is a nonnegative function, $C, D > 0$ and $\delta_0 \in (0, 1)$ are some constants that are specific to the algorithm, and ψ is a function mapping $[0, \delta_0]$ into $(0, 1)$. In typical circumstances, $h(y)$ may denote the function gap $\varphi(y) - \min \varphi$ or the square distance to the solution $\|y - \bar{x}\|^2$; the coefficient δ is proportional to a step-size parameter that the user is free to choose; the function $\psi(\delta) = c\delta$ is linear for some constant $c > 0$. The procedure, formally described in Algorithm 8, runs in stages by repeatedly calling $\mathcal{A}(y_k, \delta_k, T_k)$. Specifically, in stage k , the method sets $\delta_k = \delta_0 2^{-k}$ and finds a point y satisfying $\mathbb{E}[h(y_T)] \leq 2 \cdot D\delta_k$, while using the output from the previous stage as a warmstart.

Algorithm 8: Geometric decay schedule

Input: $y_0 \in \mathbf{R}^d$, $C, D > 0$, $\delta_0 \in (0, 1)$, estimate $\Delta \geq h(y_0)$, accuracy $\varepsilon > 0$, algorithm $\mathcal{A}(y, \delta, T)$ satisfying (B.2).

Initialize: Set $y_0 = \mathcal{A}(y_0, \delta_0, T_0)$ with $T_0 = \frac{1}{\psi(\delta_0)} \cdot \log\left(\frac{2C\Delta}{\varepsilon}\right)$.

Set $K = \left\lceil 1 + \log_2 \left(\frac{D\delta_0}{\varepsilon} \right) \right\rceil$.

Step $k = 1, \dots, K$:

$$\text{Set } y_k = \mathcal{A}(y_{k-1}, \delta_k, T_k) \quad \text{with} \quad \delta_k = 2^{-k} \delta_0, \quad T_k = \left\lceil \frac{1}{\psi(\delta_k)} \cdot \log(4C) \right\rceil.$$

Return y_K .

The overall efficiency of the scheme is summarized in the following lemma. The proof can be found in [36, Appendix B.1].

Lemma B.2 (Geometric decay). *The point y returned by Algorithm 8 satisfies $\mathbb{E}[h(y_K)] \leq \varepsilon$ and the efficiency estimate holds:*

$$\sum_{k=0}^K T_k = \left\lceil \frac{1}{\psi(\delta_0)} \cdot \log \left(\frac{2C\Delta}{\varepsilon} \right) \right\rceil + \sum_{k=1}^K \left\lceil \frac{\log(4C)}{\psi(2^{-k} \delta_0)} \right\rceil.$$

In particular, when ψ has the form $\psi(\delta) = c\delta$ for some constants $c \in (0, \frac{1}{\delta_0})$, the estimate becomes

$$\sum_{k=0}^K T_k = \mathcal{O} \left(\frac{1}{c\delta_0} \cdot \log \left(\frac{2C\Delta}{\varepsilon} \right) + \frac{D \log(4C)}{\varepsilon c} \right). \quad (\text{B.3})$$

C Proof of Theorem 7.5

The argument we present is based on the technique of estimate sequences, originally introduced by Nesterov [41, Section 2.2.1] for deterministic algorithms, and recently extended by [35] to stochastic settings. In particular, we closely follow the notation and the proof outline of [35, Section 4.1].

Recall that Algorithm 2 amounts to the recursion

$$\left\{ \begin{array}{l} x_t = \arg \min_x \langle g_t, x \rangle + r(x) + \frac{1}{2\eta_t} \|x - y_{t-1}\|^2, \\ \beta_t = \frac{\delta_t(1-\delta_t)\eta_{t+1}}{\eta_t\delta_{t+1} + \eta_{t+1}\delta_t^2} \\ y_t = x_t + \beta_t(x_t - x_{t-1}) \end{array} \right\},$$

where δ_t and γ_t are defined in (7.6), with $\gamma_0 \geq \hat{\alpha}$. Henceforth, define $\Gamma_t := \prod_{i=1}^t (1 - \delta_i)$.

Estimate Sequences. We next set up the machinery of estimate sequences. To this end, define the auxiliary vectors

$$\tilde{g}_t := \eta_t^{-1}(y_{t-1} - x_t) \quad \text{and} \quad r'_t := \tilde{g}_t - g_t \in \partial r(x_t),$$

and the “local models”

$$l_t(x) := f(y_{t-1}) + \langle g_t, x - y_{t-1} \rangle + \frac{\hat{\alpha}}{2} \|x - y_{t-1}\|^2 + r(x_t) + \langle r'_t, x - x_t \rangle.$$

Note that Lemma 7.1 guarantees the bound $\mathbb{E}[l_t(\bar{x})] \leq \varphi(\bar{x})$. This is worth emphasizing: in expectation, the models l_t evaluated at \bar{x} lower bound the minimal value $\varphi(\bar{x})$.

Estimate sequences are constructed by aggregating the “local models” along the iterations. Namely, fix a vector $v_0 \in \mathbf{R}^d$ and define the sequence of functions:

$$d_0(x) := \varphi(x_0) + \frac{\gamma_0}{2} \|x - v_0\|^2, \quad (\text{C.1})$$

$$d_t(x) := (1 - \delta_t)d_{t-1}(x) + \delta_t l_t(x) \quad \forall t \geq 1, \quad (\text{C.2})$$

Since each function d_t is a spherical quadratic, we may write it in standard form

$$d_t(x) = d_t^* + \frac{\gamma_t}{2} \|x - v_t\|^2, \quad (\text{C.3})$$

for some vectors $v_t \in \mathbf{R}^d$ and real $d_t^* \in \mathbf{R}$. The following relationship between the sequences v_t , x_t , and y_t follows from algebraic manipulations and is classical [35, Lemma 1]:

$$(x_{t-1} - y_{t-1}) + \frac{\delta_t \gamma_{t-1}}{\gamma_t} (v_{t-1} - y_{t-1}) = 0 \quad \forall t \geq 1. \quad (\text{C.4})$$

Henceforth, we will let $\mathbb{E}_t[\cdot]$ denote the expectation conditioned on g_0, \dots, g_{t-1} . We begin with the following basic lemma showing that lower bounds on the deviations $\mathbb{E}[d_t^*] - \mathbb{E}[\varphi(x_t)]$ directly control the progress of the algorithm.

Lemma C.1 (Basic estimate sequence bound). *Suppose that there exists a sequence of numbers $\xi_t \geq 0$ satisfying $\mathbb{E}[d_t^*] \geq \mathbb{E}[\varphi(x_t)] - \xi_t$ for each $t \geq 0$. Then for each $t \geq 0$ the estimate holds:*

$$\mathbb{E} \left[\varphi(x_t) - \varphi(\bar{x}) + \frac{\gamma_t}{2} \|\bar{x} - v_t\|^2 \right] \leq \Gamma_t \left[\varphi(x_0) - \varphi(\bar{x}) + \frac{\gamma_0}{2} \|x_0 - \bar{x}\|^2 \right] + \xi_t.$$

Proof. Lemma 7.1 guarantees the bound $\mathbb{E}_t[l_t(\bar{x})] \leq \varphi(\bar{x})$. Therefore from the definition of the estimate sequence, we have

$$\mathbb{E}_t[d_t(\bar{x})] = (1 - \delta_t)d_{t-1}(\bar{x}) + \delta_t \mathbb{E}_t[l_t(\bar{x})] \leq (1 - \delta_t)d_{t-1}(\bar{x}) + \delta_t \varphi(\bar{x})$$

Using the tower rule for expectations, subtracting $\varphi(\bar{x})$ from both sides, and unrolling the recursion, yields

$$\mathbb{E}[d_t(\bar{x})] - \varphi(\bar{x}) \leq (1 - \delta_t)(\mathbb{E}[d_{t-1}(\bar{x})] - \varphi(\bar{x})) \leq \Gamma_t(d_0(\bar{x}) - \varphi(\bar{x})).$$

Using the expression (C.3) for d_t therefore gives

$$\mathbb{E} \left[d_t^* - \varphi(\bar{x}) + \frac{\gamma_t}{2} \|\bar{x} - v_t\|^2 \right] \leq \Gamma_t(d_0(\bar{x}) - \varphi(\bar{x})).$$

Taking into account the assumed bound, $\mathbb{E}[d_t^*] \geq \mathbb{E}[\varphi(x_t)] - \xi_t$, completes the proof. \square

In light of Lemma C.1, our immediate task is to find conditions guaranteeing the inequality $\mathbb{E}[d_t^*] \geq \mathbb{E}[\varphi(x_t)] - \xi_t$ for some deterministic sequence ξ_t . This is the content of the following theorem, whose proof we postpone to Section C.1.

Theorem C.2 (Accelerated stochastic gradient method). *Suppose $\eta_t \leq \frac{1}{4L}$ and that for all $t = 0, \dots, T$ the inequality holds:*

$$\sqrt{1 + \frac{\delta_t}{\hat{\alpha}\eta_t}} \leq \frac{\hat{\alpha}/B}{\sqrt{32}(1 + \eta_t B)}. \quad (\text{C.5})$$

Then for all $t = 0, \dots, T$, the estimate $\mathbb{E}[d_t^] \geq \mathbb{E}[\varphi(x_t)] - \xi_t$ holds, where ξ_t are defined recursively as $\xi_t = (1 - \delta_t)\xi_{t-1} + \frac{9}{8}\eta_t\sigma^2$.*

Let us now deduce Theorem 7.5 from Theorem C.2. Namely, setting $\eta_t = \frac{1}{4L}$ and $\gamma_0 = \hat{\alpha}$ results in the expressions $\delta_t = \sqrt{\hat{\alpha}\eta_t} = (1/2)\sqrt{\hat{\alpha}/L}$ for all indices t . Therefore the sufficient condition (C.5) amount to

$$\sqrt{1 + 2\sqrt{\frac{L}{\hat{\alpha}}}} = \sqrt{1 + 2\sqrt{\frac{L}{\alpha - 2B}}} = \sqrt{1 + 2\sqrt{\frac{L}{\alpha} \cdot \frac{1}{1 - 2\rho}}} \leq \frac{\frac{1-2\rho}{\rho}}{\sqrt{32}(1 + \frac{\alpha}{4L} \cdot \rho)}.$$

Using the estimate $1 + \frac{\alpha}{4L} \cdot \rho \leq 2$ on the right and $\rho \leq \frac{1}{3}$ on the left, it suffices to ensure

$$\sqrt{1 + 2\sqrt{3\kappa}} \leq \frac{1}{2\sqrt{32}} \frac{1 - 2\rho}{\rho}.$$

Solving for ρ yields exactly the parameter regime assumed in Theorem 7.5. An application of Lemma C.1 yields therefore the guarantee

$$\mathbb{E}[\varphi(x_t) - \varphi(\bar{x})] \leq 2\Gamma_t[\varphi(x_0) - \varphi(\bar{x})] + \xi_t.$$

Unrolling the recursion for ξ_t and using Lemma A.1 yields

$$\xi_t = \frac{9}{8}\Gamma_t \sum_{i=1}^t \frac{\eta_i\sigma^2}{\Gamma_i} = \frac{9\sigma^2}{16\sqrt{L\hat{\alpha}}}(1 - \Gamma_t).$$

This completes the proof of Theorem 7.5.

C.1 Proof of Theorem C.2

The argument we present closely parallels that of [35, Theorem 3], which in turn builds on Nesterov's original treatment in [41, p. 78]. Assume by induction $\mathbb{E}[d_{t-1}^*] \geq \mathbb{E}[\varphi(x_{t-1})] - \xi_{t-1}$ for some constant $\xi_{t-1} \geq 0$. We aim to establish an analogous estimate for the next iterate. The same algebraic manipulations as in the very beginning of [35, Theorem 3] apply verbatim, yielding the lower bound

$$\begin{aligned} d_t^* &\geq (1 - \delta_t)d_{t-1}^* + \delta_t l_t(y_{t-1}) - \frac{\eta_t}{2} \|\tilde{g}_t\|^2 \\ &\quad + \frac{\delta_t(1-\delta_t)\gamma_{t-1}}{\gamma_t} \left(\langle \tilde{g}_t, v_{t-1} - y_{t-1} \rangle + \frac{\hat{\alpha}}{2} \|v_{t-1} - y_{t-1}\|^2 \right). \end{aligned} \quad (\text{C.6})$$

In the static setting (both deterministic [41] and stochastic [35]), the term $\|v_{t-1} - y_{t-1}\|^2$ is lower-bounded by zero and ignored. We will instead carry this term forward in order to offset the bias. To simplify notation, define the bias $\Delta_t := \nabla f(y_{t-1}) - \mathbb{E}[g_t]$. Combining (C.2), (C.6), and the inductive hypothesis yields

$$\begin{aligned} \mathbb{E}[d_t^*] &\geq (1 - \delta_t)[\mathbb{E}[\varphi(x_{t-1})] - \xi_{t-1}] + \delta_t \mathbb{E}[l_t(y_{t-1})] - \frac{\eta_t}{2} \mathbb{E}[\|\tilde{g}_t\|^2] \\ &\quad + \frac{\delta_t(1-\delta_t)\gamma_{t-1}}{\gamma_t} \mathbb{E} \left(\langle \tilde{g}_t, v_{t-1} - y_{t-1} \rangle + \frac{\hat{\alpha}}{2} \|y_{t-1} - v_{t-1}\|^2 \right). \end{aligned} \quad (\text{C.7})$$

Next, using convexity, we conclude

$$\begin{aligned} \mathbb{E}[\varphi(x_{t-1})] &\geq \mathbb{E} \left[f(y_{t-1}) + \langle \nabla f(y_{t-1}), x_{t-1} - y_{t-1} \rangle + r(x_t) + \langle r'_t, x_{t-1} - x_t \rangle \right] \\ &= \mathbb{E} \left[l_t(y_{t-1}) + \langle \tilde{g}_t, x_{t-1} - y_{t-1} \rangle + \langle \Delta_t, x_{t-1} - y_{t-1} \rangle \right]. \end{aligned}$$

Combining this estimate with (C.7) yields

$$\begin{aligned} \mathbb{E}[d_t^*] &\geq \mathbb{E}[l_t(y_{t-1})] - (1 - \delta_t)\xi_{t-1} - \frac{\eta_t}{2} \mathbb{E}[\|\tilde{g}_t\|^2] \\ &\quad + (1 - \delta_t) \mathbb{E} \left[\langle \tilde{g}_t, (x_{t-1} - y_{t-1}) + \frac{\delta_t \gamma_{t-1}}{\gamma_t} (v_{t-1} - y_{t-1}) \rangle \right] \\ &\quad + (1 - \delta_t) \left(\frac{\hat{\alpha} \delta_t \gamma_{t-1}}{2\gamma_t} \mathbb{E}[\|y_{t-1} - v_{t-1}\|^2] + \mathbb{E} \langle \Delta_t, x_{t-1} - y_{t-1} \rangle \right). \end{aligned}$$

The quantity $(x_{t-1} - y_{t-1}) + \frac{\delta_t \gamma_{t-1}}{\gamma_t} (v_{t-1} - y_{t-1})$ is zero by construction (C.4). Using the expression $x_{t-1} - y_{t-1} = \frac{\delta_t \gamma_{t-1}}{\gamma_t} (y_{t-1} - v_{t-1})$ and Young's inequality, we deduce

$$\begin{aligned} \frac{\hat{\alpha} \gamma_{t-1} \delta_t}{2\gamma_t} \|y_{t-1} - v_{t-1}\|^2 + \langle \Delta_t, x_{t-1} - y_{t-1} \rangle &= \frac{\gamma_{t-1} \delta_t}{\gamma_t} \left(\frac{\hat{\alpha}}{2} \|y_{t-1} - v_{t-1}\|^2 + \langle \Delta_t, y_{t-1} - v_{t-1} \rangle \right) \\ &\geq -\frac{\gamma_{t-1} \delta_t}{2\gamma_t \hat{\alpha}} \cdot \|\Delta_t\|^2. \end{aligned}$$

Therefore, we conclude that

$$\mathbb{E}[d_t^*] \geq \mathbb{E} \left[l_t(y_{t-1}) \right] - (1 - \delta_t)\xi_{t-1} - \frac{\eta_t}{2} \mathbb{E}[\|\tilde{g}_t\|^2] - (1 - \delta_t) \frac{\gamma_{t-1} \delta_t}{2\gamma_t \hat{\alpha}} \mathbb{E} [\|\Delta_t\|^2]. \quad (\text{C.8})$$

The next lemma lower bounds $\mathbb{E}[l_t(y_{t-1})]$ appearing in (C.8). Its proof is given in Section D.1.

Lemma C.3. *For every index t , the estimate holds:*

$$\mathbb{E}[\varphi(x_t)] \leq \mathbb{E}[l_t(y_{t-1})] + \left(\frac{L\eta_t^2}{2} - \frac{3\eta_t}{4} \right) \mathbb{E}[\|\tilde{g}_t\|^2] + \eta_t \sigma^2 + \eta_t \mathbb{E}[\|\Delta_t\|^2].$$

Combining the estimate (C.8) and Lemma C.3, we obtain

$$\mathbb{E}[d_t^*] \geq \mathbb{E}\varphi(x_t) - (1 - \delta_t)\xi_{t-1} - \eta_t\sigma^2 - P_t,$$

where the error term P_t is defined as

$$P_t := \left(\frac{L\eta_t^2}{2} - \frac{\eta_t}{4} \right) \mathbb{E}[\|\tilde{g}_t\|^2] + \eta_t \left(1 + \frac{(1-\delta_t)\gamma_{t-1}\delta_t}{2\gamma_t\hat{\alpha}\eta_t} \right) \mathbb{E}\|\Delta_t\|^2.$$

Our final goal is to show $P_t \leq c\eta_t\sigma^2$ for some constant c , which will complete the induction by setting

$$\xi_t := (1 - \delta_t)\xi_{t-1} + (1 + c)\sigma^2\eta_t.$$

To this end, let us simplify the expression for P_t by using (7.6) to write

$$\frac{(1 - \delta_t)\gamma_{t-1}\delta_t}{2\gamma_t\hat{\alpha}\eta_t} = \frac{1}{2} \cdot \frac{(1 - \delta_t)\gamma_{t-1}}{\gamma_t} \cdot \frac{\delta_t}{\hat{\alpha}\eta_t} = \frac{1}{2} \left(1 - \frac{\delta_t\hat{\alpha}}{\gamma_t} \right) \cdot \frac{\delta_t}{\hat{\alpha}\eta_t} = \frac{1}{2} \left(\frac{\delta_t}{\hat{\alpha}\eta_t} - \frac{\delta_t^2}{\gamma_t\eta_t} \right) = \frac{1}{2} \left(\frac{\delta_t}{\hat{\alpha}\eta_t} - 1 \right).$$

Thus we arrive at the expression

$$P_t := \left(\frac{L\eta_t^2}{2} - \frac{\eta_t}{4} \right) \mathbb{E}[\|\tilde{g}_t\|^2] + \frac{\eta_t}{2} \left(1 + \frac{\delta_t}{\hat{\alpha}\eta_t} \right) \mathbb{E}\|\Delta_t\|^2.$$

The strategy is now to show that $\mathbb{E}[\|\tilde{g}_t\|^2]$ is much larger than $\|y_{t-1} - \bar{x}\|^2$ while $\mathbb{E}\|\Delta_t\|^2$ is much smaller than $\|y_{t-1} - \bar{x}\|^2$. This is the content of the following lemma, which is proved in Section D.2.

Lemma C.4. *For each index $t \geq 1$, the estimate holds:*

$$\mathbb{E}[\|\tilde{g}_t\|^2] \geq \frac{\hat{\alpha}^2}{8(1 + \eta_t B)^2} \mathbb{E}[\|y_{t-1} - \bar{x}\|^2] - \sigma^2,$$

Using Lemma C.4 and the estimates $\eta_t \leq \frac{1}{4L}$ and $\mathbb{E}\|\Delta_t\|^2 \leq B^2\mathbb{E}\|y_{t-1} - \bar{x}\|^2$ we conclude

$$\begin{aligned} P_t &= \left(\frac{L\eta_t^2}{2} - \frac{\eta_t}{4} \right) \mathbb{E}[\|\tilde{g}_t\|^2] + \eta_t \left(1 + \frac{(1-\delta_t)\gamma_{t-1}\delta_t}{2\gamma_t\hat{\alpha}\eta_t} \right) \mathbb{E}\|\Delta_t\|^2 \\ &\leq \eta_t \left(-\frac{1}{8}\mathbb{E}[\|\tilde{g}_t\|^2] + \frac{1}{2} \left(1 + \frac{\delta_t}{\hat{\alpha}\eta_t} \right) \mathbb{E}\|\Delta_t\|^2 \right) \\ &\leq \frac{\eta_t}{8}\sigma^2 + \eta_t \left(-\frac{\hat{\alpha}^2}{64(1+\eta_t B)^2} + \frac{1}{2} \left(1 + \frac{\delta_t}{\hat{\alpha}\eta_t} \right) B^2 \right) \mathbb{E}\|y_{t-1} - \bar{x}\|^2 \\ &\leq \frac{\eta_t}{8}\sigma^2. \end{aligned}$$

This completes the induction by setting $\xi_t = (1 - \delta_t)\xi_{t-1} + \frac{9}{8}\eta_t\sigma^2$.

D Proofs of technical lemmas in Section C

D.1 Proof of Lemma C.3

The argument follows similar reasoning as [35, Lemma 2], with a careful accounting for the bias. Using smoothness, we estimate

$$\begin{aligned} \varphi(x_t) &\leq \mathbb{E}[f(y_{t-1}) + \langle \nabla f(y_{t-1}), x_t - y_{t-1} \rangle + \frac{L}{2}\|x_t - y_{t-1}\|^2 + r(x_t)] \\ &= \mathbb{E}[f(y_{t-1}) + \langle g_t, x_t - y_{t-1} \rangle + \frac{L}{2}\|x_t - y_{t-1}\|^2 + r(x_t)] + \mathbb{E}[\langle \mathbb{E}[g_t] - g_t, x_t - y_{t-1} \rangle] \\ &\quad + \mathbb{E}[\langle \nabla f(y_{t-1}) - \mathbb{E}[g_t], x_t - y_{t-1} \rangle], \end{aligned} \tag{D.1}$$

where the last equality follows from algebraic manipulations. Next, we compute

$$\mathbb{E}\langle \mathbb{E}[g_t] - g_t, x_t - y_{t-1} \rangle = \mathbb{E}\langle \mathbb{E}[g_t] - g_t, x_t \rangle = \mathbb{E}\langle \mathbb{E}[g_t] - g_t, x_t - w_t \rangle,$$

where we define $w_t := \text{prox}_{\eta_t r}(y_{t-1} - \eta_t \mathbb{E}[g_t])$. Taking into account that the proximal map $\text{prox}_{\eta_t r}(\cdot)$ is nonexpansive, we deduce

$$\|x_t - w_t\| = \|\text{prox}_{\eta_t r}(y_{t-1} - \eta_t g_t) - \text{prox}_{\eta_t r}(y_{t-1} - \eta_t \mathbb{E}[g_t])\| \leq \eta_t \|g_t - \mathbb{E}[g_t]\|.$$

Therefore continuing (D.1) we obtain

$$\begin{aligned} \varphi(x_t) &\leq \mathbb{E}[f(y_{t-1}) + \langle g_t, x_t - y_{t-1} \rangle + \frac{L}{2} \|x_t - y_{t-1}\|^2 + r(x_t)] + \eta_t \sigma^2 \\ &\quad + \mathbb{E}[\langle \nabla f(y_{t-1}) - \mathbb{E}[g_t], x_t - y_{t-1} \rangle] \\ &= \mathbb{E}[l_t(y_{t-1}) + \langle \tilde{g}_t, x_t - y_{t-1} \rangle + \frac{L}{2} \|x_t - y_{t-1}\|^2] + \eta_t \sigma^2 \\ &\quad + \mathbb{E}[\langle \nabla f(y_{t-1}) - \mathbb{E}[g_t], x_t - y_{t-1} \rangle] \\ &= \mathbb{E}[l_t(y_{t-1})] + \left(\frac{L\eta_t^2}{2} - \eta_t \right) \mathbb{E}[\|\tilde{g}_t\|^2] + \eta_t \sigma^2 \\ &\quad + \mathbb{E}[\langle \nabla f(y_{t-1}) - \mathbb{E}[g_t], x_t - y_{t-1} \rangle]. \end{aligned}$$

Finally, Young's inequality yields

$$\begin{aligned} \mathbb{E}[\langle \nabla f(y_{t-1}) - \mathbb{E}[g_t], x_t - y_{t-1} \rangle] &\leq \eta_t \|\nabla f(y_{t-1}) - \mathbb{E}[g_t]\|^2 + \frac{1}{4\eta} \|x_t - y_{t-1}\|^2 \\ &= \eta \|\nabla f(y_{t-1}) - \mathbb{E}[g_t]\|^2 + \frac{\eta_t}{4} \|\tilde{g}_t\|^2, \end{aligned}$$

thereby completing the proof.

D.2 Proof of Lemma C.4

The result will follow quickly from the following stand-alone lemma.

Lemma D.1. *Fix a constant $\eta \leq 2/L$, a point $y \in \mathbf{R}^d$, and a vector $v \in \mathbf{R}^d$ satisfying $\|v - \nabla f(y)\| \leq B\|y - \bar{x}\|$. Define the proximal gradient update and the displacement vector:*

$$y^+ = \text{prox}_{\eta r}(y - \eta v) \quad \text{and} \quad \tilde{g} = \eta^{-1}(y - y^+).$$

Then the estimate holds:

$$\|\tilde{g}\| \geq \frac{\hat{\alpha}}{2(1 + \eta B)} \|y - \bar{x}\|. \quad (\text{D.2})$$

Before proving Lemma D.1, let us see how it implies Lemma C.4—the result we are after. Setting $y = y_{t-1}$ and $v = \mathbb{E}[g_t]$ in Lemma D.1 yields

$$\|\eta_t^{-1}(y_{t-1} - \text{prox}_{\eta_t r}(y_{t-1} - \eta_t \mathbb{E}[g_t]))\| \geq \frac{\hat{\alpha}}{2(1 + \eta_t B)} \|y_{t-1} - \bar{x}\| \quad \forall t \geq 1. \quad (\text{D.3})$$

Therefore, we now compute

$$\begin{aligned} \|\tilde{g}_t\| &= \eta_t^{-1} \|y_{t-1} - \text{prox}_{\eta_t r}(y_{t-1} - \eta_t g_t)\| \\ &\geq \eta_t^{-1} \|y_{t-1} - \text{prox}_{\eta_t r}(y_{t-1} - \eta_t \mathbb{E}[g_t])\| - \eta_t^{-1} \|\text{prox}_{\eta_t r}(y_{t-1} - \eta_t g_t) - \text{prox}_{\eta_t r}(y_{t-1} - \eta_t \mathbb{E}[g_t])\| \\ &\geq \eta_t^{-1} \|y_{t-1} - \text{prox}_{\eta_t r}(y_{t-1} - \eta_t \mathbb{E}[g_t])\| - \|g_t - \mathbb{E}[g_t]\|, \\ &\geq \frac{\hat{\alpha}}{2(1 + \eta_t B)} \|y_{t-1} - \bar{x}\| - \|g_t - \mathbb{E}[g_t]\|, \end{aligned}$$

where the first inequality follows from the reverse triangle inequality, the second inequality uses that the proximal map is nonexpansive, and the third follows from (D.3). Rearranging and using the inequality $2a^2 + 2b^2 \geq (a + b)^2$ yields

$$2\|\tilde{g}_t\|^2 + 2\|g_t - \mathbb{E}[g_t]\|^2 \geq (\|\tilde{g}_t\| + \|g_t - \mathbb{E}[g_t]\|)^2 \geq \left(\frac{\hat{\alpha}}{2(1 + \eta B)} \|y_{t-1} - \bar{x}\| \right)^2.$$

Taking expectations of both sides completes the proof of Lemma C.4. It remains to verify Lemma D.1.

Proof of Lemma D.1. To simplify notation, define the error $\Delta = v - \nabla f(y)$. Since y^+ is the minimizer of the η^{-1} -strongly convex function $\langle v, \cdot - y \rangle + r + \frac{1}{2\eta} \|\cdot - y\|^2$, we deduce for every $x \in \mathbf{R}^d$ the estimate holds:

$$\langle v, y^+ - y \rangle + \frac{1}{2\eta} \|y^+ - y\|^2 + r(y^+) \leq \langle v, x - y \rangle + \frac{1}{2\eta} \|x - y\|^2 + r(x) - \frac{1}{2\eta} \|y^+ - x\|^2. \quad (\text{D.4})$$

Next, algebraic manipulations yield the equality

$$\begin{aligned} \|y^+ - y\|^2 + \|y^+ - x\|^2 - \|x - y\|^2 &= 2\langle y - y^+, x - y \rangle + 2\|y - y^+\|^2 \\ &= 2\eta \langle \tilde{g}, x - y \rangle + 2\eta^2 \|\tilde{g}\|^2. \end{aligned}$$

Combining this estimate with (D.4) yields

$$\langle v, y^+ - y \rangle + r(y^+) \leq \langle v, x - y \rangle + r(x) + \langle \tilde{g}, y - x \rangle - \eta \|\tilde{g}\|^2. \quad (\text{D.5})$$

Smoothness of f in turn guarantees

$$\begin{aligned} \langle v, y^+ - y \rangle &= \langle \nabla f(y), y^+ - y \rangle + \langle v - \nabla f(y), y^+ - y \rangle \\ &\geq f(y^+) - f(y) - \frac{L}{2} \|y^+ - y\|^2 - \eta \langle \Delta, \tilde{g} \rangle. \end{aligned}$$

Combining this estimate with (D.5) therefore yields

$$\varphi(y^+) \leq f(y) + \langle v, x - y \rangle + r(x) + \langle \tilde{g}, y - x \rangle - \left(\eta - \frac{L\eta^2}{2} \right) \|\tilde{g}\|^2 + \eta \langle \Delta, \tilde{g} \rangle. \quad (\text{D.6})$$

Set now $x = \bar{x}$ in (D.6) and using strong convexity observe the estimate

$$\begin{aligned} \langle v, \bar{x} - y \rangle &= \langle \nabla f(y), \bar{x} - y \rangle + \langle v - \nabla f(y), \bar{x} - y \rangle \\ &\leq f(\bar{x}) - f(y) - \frac{\hat{\alpha}}{2} \|y - \bar{x}\|^2. \end{aligned}$$

We thus deduce

$$\varphi(y^+) \leq \varphi(\bar{x}) + \langle \tilde{g}, y - \bar{x} \rangle - \frac{\hat{\alpha}}{2} \|y - \bar{x}\|^2 - \left(\eta - \frac{L\eta^2}{2} \right) \|\tilde{g}\|^2 + \eta \langle \Delta, \tilde{g} \rangle. \quad (\text{D.7})$$

Lower-bounding $\varphi(y^+)$ by $\varphi(\bar{x})$ and rearranging yields

$$\langle \tilde{g}, y - \bar{x} \rangle \geq \frac{\hat{\alpha}}{2} \|y - \bar{x}\|^2 + \left(\eta - \frac{L\eta^2}{2} \right) \|\tilde{g}\|^2 - \eta \langle \Delta, \tilde{g} \rangle.$$

Using Cauchy-Schwarz, lower bounding the term $\left(\eta - \frac{L\eta^2}{2} \right) \|\tilde{g}\|^2$ by zero, and dividing through by $\|y - \bar{x}\|$ yields

$$\|\tilde{g}\| \geq \frac{\hat{\alpha}}{2} \|y - \bar{x}\| - \frac{\eta \|\Delta\| \|\tilde{g}\|}{\|y - \bar{x}\|} \geq \frac{\hat{\alpha}}{2} \|y - \bar{x}\| - \eta B \|\tilde{g}\|.$$

Rearranging completes the proof of (D.2). \square

E Estimate (3.4) implies an angle condition

Squaring both sides of (3.4), expanding, and dividing by $\|\nabla f_x(x)\| \cdot \|\nabla f_{\bar{x}}(x)\|$ yields

$$2 \left\langle \frac{\nabla f_x(x)}{\|\nabla f_x(x)\|}, \frac{\nabla f_{\bar{x}}(x)}{\|\nabla f_{\bar{x}}(x)\|} \right\rangle \geq \frac{\|\nabla f_x(x)\|}{\|\nabla f_{\bar{x}}(x)\|} + (1 - \rho^2) \frac{\|\nabla f_{\bar{x}}(x)\|}{\|\nabla f_x(x)\|}.$$

Lower bounding the right side with the estimate $a + b \geq 2\sqrt{ab}$, we conclude $\left\langle \frac{\nabla f_x(x)}{\|\nabla f_x(x)\|}, \frac{\nabla f_{\bar{x}}(x)}{\|\nabla f_{\bar{x}}(x)\|} \right\rangle \geq \sqrt{1 - \rho^2}$, as claimed.

F Numerical experiments on strategic classification

This section describes the experimental setup of strategic classification used in [39, 45] and in the current work. Specifically, we begin with the Kaggle data set [32], which contains a historical financial record of 15000 individuals. Each individual is described by ten features $a \in \mathbf{R}^{10}$, with the label $b \in \{0, 1\}$ indicating whether 90 days have passed since the person experienced delinquency. In the experiments, we subsample $n/2$ records labeled with 0 and with 1, respectively. We normalize all features to have zero mean and unit variance. The goal is to learn a classifier parametrized by $x \in \mathbf{R}^{10}$ that accurately predicts the label. As in [39, 45] we use the regularized logistic loss:

$$\frac{1}{n} \sum_{i=1}^n -b_i \cdot x^\top a_i + \log(1 + \exp(x^\top a_i)) + \frac{\alpha}{2} \|x\|^2,$$

for some parameter $\alpha > 0$. We isolate three features (utilization of credit lines, number of open credit lines, number of real estate loans) that the individuals will strategically adapt in reaction to a deployed classifier. Namely, when a classifier parametrized by x is deployed, each individual updates the features a as

$$a_S = a_S - \gamma \cdot x_S,$$

where S is the index set of the three strategic features. This update is equivalent to (1.3) with the linear utility function $u(x, a) = -\langle a, x \rangle$ and quadratic cost $c(a', a) = \frac{1}{2\gamma} \|a' - a\|^2$. The label of each individual stay the same. It is shown in [45, Section B.2] that the gradient of the population objective is Lipschitz continuous in x with constant $L = \frac{1}{4n} \sum_{i=1}^n \|a_i\|^2 + \alpha$ and is Lipschitz continuous in (a, b) with parameter $\beta = 2$. Therefore, we may estimate $\rho = \frac{2\gamma}{\alpha}$. Therefore the interesting parameter for the strategic effects is $\gamma \in (0, \frac{\alpha}{2})$. This estimate is fairly crude and experimentally we see that numerical methods can work well in a much wider parameter regime.