Learning Rate Schedules in the Presence of Distribution Shift

Matthew Fahrbach [1] **Adel Javanmard** [1 2] **Vahab Mirrokni** [1] **Pratik Worah** [1]

Abstract

We design learning rate schedules that minimize regret for SGD-based online learning in the presence of a changing data distribution. We fully characterize the optimal learning rate schedule for online linear regression via a novel analysis with stochastic differential equations. For general convex loss functions, we propose new learning rate schedules that are robust to distribution shift, and we give upper and lower bounds for the regret that only differ by constants. For non-convex loss functions, we define a notion of regret based on the gradient norm of the estimated models and propose a learning schedule that minimizes an upper bound on the total expected regret. Intuitively, one expects changing loss landscapes to require more exploration, and we confirm that optimal learning rate schedules typically increase in the presence of distribution shift. Finally, we provide experiments for high-dimensional regression models and neural networks to illustrate these learning rate schedules and their cumulative regret.

1. Introduction

A fundamental question when training neural networks is how much of the weight space to explore and when to stop exploring. For stochastic gradient descent (SGD)-based training algorithms, this is primarily governed by the learning rate. If the learning rate is high, then we explore more of the weight space and vice versa. Learning rates are typically decreased over time in order to converge to a local optimum, and there is now a substantial literature focused on how fast learning rates should decay for fixed data distributions (see, e.g., Tripuraneni et al. (2018) and Fang et al. (2018), and the references therein).

However, what should we do if the data distribution is constantly changing? This is the case in many large-scale online learning systems where (1) the data arrives in a stream, (2) the model continuously makes predictions and computes the loss, and (3) it always updates its weights based on the new data it sees (Anil et al., 2022). The goal of such a system is to always keep the loss low. In this setting, convergence is less of a priority since the model needs to be able to adapt to distribution shifts. Intuitively, if the loss landscape is consistently changing (either gradually or due to infrequent sudden spikes), then it is sensible for the model to always explore its weight space. We formalize this idea in our work.

Such an investigation naturally leads to the question of how high the learning rate should be, and what an optimal learning rate schedule is in an online learning scenario? These questions are critical because while tuning the learning rate can lead to improved accuracy in many applications, it can also make the online learner widely inaccurate if the wrong learning rate is used as the distribution changes.

Formally, we study learning rate schedules in the presence of distribution shifts by considering *dynamic regret*, a well-known notion in online optimization that measures the performance against a dynamic comparator sequence. This regret framework captures the lifetime performance of an online learning system that makes predictions on incoming examples as they arrive (possibly from a time-varying distribution) before using this data to update its weights.

Our main contributions can be summarized as follows:

Linear regression. We consider a linear regression setup with time-varying coefficients $\{\theta_t^*\}_{t\geq 1}$, which are chosen upfront by an adversary such that $\|\theta_t^* - \theta_{t+1}^*\|_2 \leq \gamma_t$ for a sequence of positive numbers $\{\gamma_t\}_{t\geq 1}$. The variation in the model coefficients results in response shift (while the covariates distribution remains the same across time). We consider a learner who updates their model estimates via mini-batch SGD with an adaptive step size sequence $\{\eta_t\}_{t\geq 1}$ chosen in an online manner (i.e., only with access to previous data points). We derive a novel stochastic differential equation (SDE) that approximates the dynamics of SGD under distribution shift, and by analyzing it, we derive the optimal learning rate schedule.

Convex loss functions. We generalize our problem formulation along the following directions: (i) We consider

[1]Google Research [2]University of Southern California. Correspondence to: Matthew Fahrbach <fahrbach@google.com>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

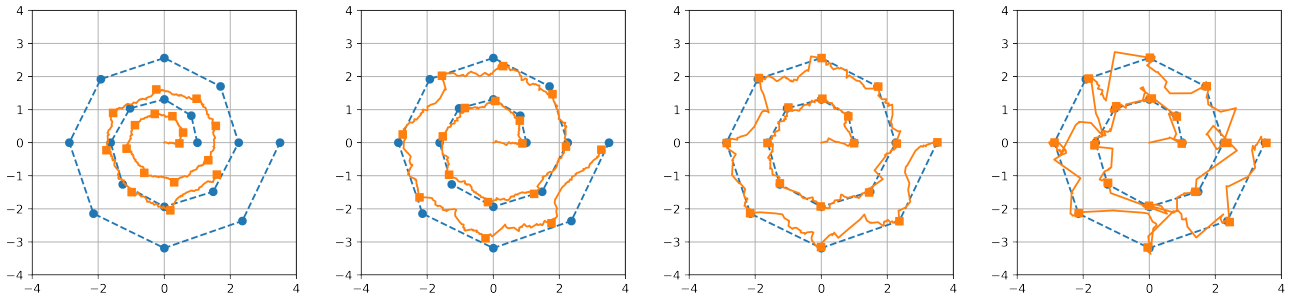

Figure 1: SGD trajectories for online linear regression with different constant learning rates. The discrete blue spirals are the optimal model weights $\theta_t^* \in \mathbb{R}^2$, which start at $(1, 0)$ and jump clockwise every 100 steps. The orange paths are the learned weights θ_t, starting at $\theta_0 = 0$ for $0 \leq t \leq 17 \cdot 100$. The orange squares depict the position every 100 steps. We use batch size $B_t = 1$ and step sizes $\eta_t \in \{0.003, 0.01, 0.03, 0.1\}$ from left to right. The rightmost SGD is the most out of control, but it incurs the least regret because it adapts to changes in θ_t^* the fastest without diverging.

general convex loss functions $\ell(\theta, z)$ that measure the loss of a model $\theta \in \mathbb{R}^p$ on the data point $z \in \mathbb{R}^d$. (ii) At each step the learner observes a batch of data points $\{z_{t,k}\}$ drawn from a time-varying distribution P_t, which means it can model both response shift and covariate shift. (iii) An adversary can choose the distributions P_t adaptively at each step by observing the history (i.e., the data and model estimates from previous rounds), in contrast to the linear regression setup where the sequence of models are time-varying but fixed a priori. For strongly convex loss functions, we give a lower bound for the total expected regret that is of the same form as our upper bound and differs only in the constants, demonstrating that our regret analysis is nearly tight. We then propose a learning rate schedule to minimize the derived upper bound on the regret. This schedule is adaptive, resulting in a time-dependent learning rate that tries to catch up with the amount of distribution shift in the moment. We refer to Section 1.1 for a detailed comparison to the literature on online convex optimization in dynamic environments.

Non-convex loss functions. For settings with non-convex loss functions, we modify the notion of regret to use the gradient norm of the estimated model. We derive an upper bound for the expected cumulative regret and propose a learning rate schedule that minimizes it. In our experiments in Appendix E, we use neural networks and dynamic learning rates to continuously classify cells arriving in a stream of small condition RNA data (Bastidas-Ponce et al., 2019). This work simulates an online and deep learning-based *flow cytometry* algorithm. We refer the reader to Li et al. (2019) for more details about this application. One take-away message from our analysis and experiments in all three settings is that an optimal learning rate schedule typically increases in the presence of distribution shift.

The organization of the paper is as follows. In Section 1.1, we proceed with a literature review. In Section 1.2, we present an overview of our tools, techniques, and informal statements of our theoretical results. We formally define the problem in Section 2. We present our results for linear regression in Section 3, convex losses in Section 4, and non-convex losses in Section 5. In Section 6, we present experiments to study the effect of the proposed learning rate schedules, including high-dimensional regression and a medical application to flow cytometry. We defer the proofs of our technical results to the appendix.

1.1. Related work

With deep neural networks now being used in countless applications and SGD remaining the dominant algorithm for training these models, there has been a surge of effort to understand how learning rates affect the behavior of stochastic optimization methods (Bengio, 2012; Smith, 2015). Most of the existing literature, however, assumes no shift in the underlying distribution across the iterations of SGD. Various trade-offs between learning rate and batch size have been studied (Keskar et al., 2016; Smith et al., 2018). In particular, Smith et al. (2018) proposes that instead of the decaying learning rate, one can increase the batch size during training and empirically show that it results in near-identical model performance with significantly fewer parameter updates. Shi et al. (2020) analyze the effect of learning rate on SGD by studying its continuum formulation given by a stochastic differential equation (SDE) and show that for a broad class of losses, this SDE converges to its stationary distribution at a linear rate, further revealing the dependence of a linear convergence rate on the learning rate. Learning rate schedules for SGD, under fixed distribution, and for the setting of least squares has been studied in (Ge et al., 2019; Jain et al., 2019). Decaying learning rate via cyclical schedules has also been proposed for training deep neural models (see, e.g., Smith (2015); Loshchilov & Hutter (2016); Li & Arora (2019)).

The effects of SGD hyperparameters (e.g., batch size and learning rate) have also been studied for the adversarial robustness of the resulting models (Yao et al., 2018; Kamath et al., 2020). In this setting, a model is trained on unperturbed samples, but at test time the sample features are slightly perturbed. In contrast, this paper considers settings where *the data distribution is constantly changing*—even during training—and studies the effect of learning rates in presence of such distribution shifts.

Connections to online optimization. The notion of dynamic regret has been used in online convex optimization to evaluate the performance of a learner against a dynamic target, as opposed to the classical single best action in hindsight (Zinkevich, 2003; Yang et al., 2016; Jadbabaie et al., 2015; Besbes et al., 2015). In this setting, nature chooses a sequence of convex functions $f_1, \dots, f_T$ and the learner chooses a model (i.e, action) θ_t at each round and incurs the loss $f_t(\theta_t)$. Our problem is closer to non-stationary approximation (Besbes et al., 2015), in which the learner only has noisy access to gradients $\nabla f_t(\theta_t)$. There is often a notion of variation to capture the change in the comparator. For example, Yang et al. (2016) works with "path variation," which measures how fast the minimizers of the sequence of loss functions change, and Besbes et al. (2015) defines a "functional variation" based on the supremum distance between consecutive loss functions.

Yang et al. (2016) give a bound for the cumulative dynamic regret when a constant step size $\eta \propto \sqrt{\mathcal{V}_T/T}$ is used, where T is the horizon length and $\mathcal{V}_T$ is the variation budget that controls the power that nature has in choosing the sequence of loss functions (see Theorem 7 therein). Besbes et al. (2015) propose a restarting procedure, which for batch size Δ_T restarts an online gradient descent algorithm every Δ_T periods. Their analysis suggests to take $\Delta_T = (T/\mathcal{V}_T)^{2/3}$ and $\eta \propto 1/\sqrt{\Delta_T}$ (see Theorem 3 therein).

While these results also suggest that in a changing environment the learning rate should be in general set higher, our formulation and analysis for the convex setting departs from these works in the following ways: (i) Instead of constant or a pre-determined learning rate, our framework allows for *adaptive schedules* where the learning rate at every epoch can be set based on the history; (ii) The notion of dynamic regret is often defined with respect to an arbitrary but fixed sequence of loss functions that satisfy a variation budget constraint. In contrast, we allow the data distribution to shift adaptively at each step after observing the history, and so the expected loss changes adaptively over time; (iii) Besbes et al. (2015) and Yang et al. (2016) establish lower bounds on the dynamic regret, but these lower bounds are for the worst-case regret over the choice of loss function sequences that satisfy the variation budget constraint. The lower bounds are obtained by carefully crafting a sequence that is hard to optimize in an online manner. However, there is a subtle difference in our setting: the loss function $\ell(\theta, z)$ is fixed and the change in the expected loss across time comes from a shift in the data distribution z. The lower bound we develop for dynamic regret uses the same fixed loss function $\ell(\theta, z)$.

1.2. Overview of techniques

To analyze the behavior of SGD in this linear regression setting, we derive a novel stochastic differential equation (SDE) that approximates the dynamics of SGD in the presence of distribution shift. Using Grönwall's inequality (Gronwall, 1919), we control the deviation of the SGD trajectory from the continuous process and relate the regret of SGD to the second moment of the continuous process, which we characterize using the celebrated Itô's lemma from stochastic calculus (Oksendal, 2013) (see Lemma D.2). With this characterization, we derive an optimal learning rate schedule in a sequential manner.

Our results for general convex loss functions are based on an intricate treatment of the regret terms, taking the expectation with respect to a proper filtration and using several properties of convex functions and SGD itself.

Non-convex loss functions can have a complicated landscape with potentially many local minima and saddle points. Even without distribution shifts, first-order methods like SGD are not guaranteed to converge to a global minimum. To deal with this, we modify the definition of regret to use the norm of the gradient of the loss for the estimated models. Thus, a trajectory that stays close to local minima of the loss functions has low total regret. To upper bound the cumulative regret in this setting, we follow a similar proof technique as in the convex case, but rely only on the SGD update formulation and first-order optimality conditions on the sequence of optimal weights $\{\theta_t^*\}_{t\geq 1}$.

2. Problem formulation: Dynamic regret

We consider an online sequential learning setting where at each step t the learner observes a batch of size B_t data points $\boldsymbol{z}_t = \{z_{t,k}\}_{k=1}^{B_t}$ drawn independently from a distribution P_t. The distributions P_t can vary with time and are defined on $\mathbb{R}^d$. The batch loss incurred at step t is $\frac{1}{B_t}\sum_{k=1}^{B_t} \ell(\theta_t, z_{t,k})$ for a function $\ell : \mathbb{R}^p \times \mathbb{R}^d \to \mathbb{R}_{\geq 0}$. The learner then updates its model weights $\theta_t \to \theta_{t+1} \in \mathbb{R}^p$.

Define the expected loss as $\bar{\ell}_t(\theta) := \mathbb{E}_{P_t}[\ell(\theta, z_{t,k})]$. Letting $(\theta_1, \theta_2, \dots)$ denote the sequence of learned models, the total expected loss up to time T is $\sum_{t=1}^{T} \bar{\ell}_t(\theta_t)$. The goal of the learner is to minimize the above objective. For each step t, we define an *oracle model* with weights

$$\theta_t^* := \arg\min_{\theta \in \mathbb{R}^p} \bar{\ell}_t(\theta). \tag{1}$$

Since the distributions P_t can vary with time, the weights θ_t^* also shift over time.

Instead of minimizing the total loss, we equivalently work with the *regret* of the learner defined with respect to the comparator sequence $(\theta_1^*, \theta_2^*, \dots)$ below:

$$\mathsf{Reg}(T) := \sum_{t=1}^{T} \mathsf{reg}_t, \quad \mathsf{reg}_t := \bar{\ell}_t(\theta_t) - \bar{\ell}_t(\theta_t^*). \tag{2}$$

Note that $\mathsf{Reg}(t)$ and reg_t are random variables that depend on θ_t. This framework can be seen as a game between nature, who chooses the distributions P_t (and thus the sequence of oracle models θ_t^*), and the learner, who must choose the sequence of models θ_t for $t \geq 1$.

The learner updates its weights using projected mini-batch stochastic gradient descent (mini-batch SGD) given by

$$\theta_{t+1} = \Pi_\Theta \left(\theta_t - \eta_t \nabla \ell_t^B(\theta_t)\right) \tag{3}$$

$$\nabla \ell_t^B(\theta_t) := \frac{1}{B_t} \sum_{k=1}^{B_t} \nabla \ell(\theta_t, z_{t,k}), \tag{4}$$

where $\nabla \ell(\theta_k, z_k)$ are stochastic gradients, Θ is a bounded convex set, and Π_Θ is the projection onto the admissible weight set $\Theta \subseteq \mathbb{R}^p$. Observe that $\mathbb{E}[\nabla \ell(\theta_k, z_k)] = \nabla \bar{\ell}(\theta_k)$, and therefore the sample average gradient above is an unbiased estimate of the gradient of the expected loss.

Nature is allowed to be *adaptive* in that she can set θ_t^* after observing the history of the data

$$\boldsymbol{z}_{[t-1]} := \{(z_{k,1}, z_{k,2}, \dots, z_{k,B_k}) : 1 \leq k \leq t-1\}. \tag{5}$$

The step sizes η_t, called the *learning rate schedule*, can also change over time in an adaptive manner, i.e., the learning rate η_t is a function of $\boldsymbol{z}_{[t-1]}$. Note that by the SGD update, θ_t is a function of $\boldsymbol{z}_{[t-1]}$, and so θ_t^* can depend on the previously learned models θ_s for $s < t$. The learning rate schedule controls how the step size changes across iterations.

Definition 2.1 (Distribution shift)**.** Recall the definition of oracle models θ_t^* in (1). We quantify the *distribution shift* (variation of P_t over time) in terms of the variation in oracle models, namely

$$\gamma_t := \|\theta_t^* - \theta_{t+1}^*\|_2 . \tag{6}$$

This is similar to the notion of path variation introduced in Yang et al. (2016), except that path variation considers the total variation in the minimizers of the sequence of loss functions, whereas we focus on individual changes after each gradient update.

3. Linear regression

We start by studying the linear regression setting with a time-varying coefficient model (Fan & Zhang, 2008; Hastie & Tibshirani, 1993). Each sample $z_{t,k} = (x_{t,k}, y_{t,k})$ is a pair of covariates $x_{t,k} \in \mathbb{R}^d$ and a label $y_{t,k}$, with

$$y_{t,k} = \langle x_{t,k}, \theta_t^* \rangle + \varepsilon_{t,k} , \tag{7}$$

where $\varepsilon_{t,k} \sim \mathsf{N}(0, \sigma^2)$ is random noise. The covariate distribution is assumed to be the same across time, and for simplicity assumed as $x_{t,k} \sim \mathsf{N}(0, I)$. The model θ_t^* changes over time, so we have label distribution shift. We consider least squares loss $\ell(\theta, z) = \frac{1}{2}(y - \langle x, \theta \rangle)^2$, for $z = (x, y)$.

To provide theoretical insight on the dependence of SGD on the learning rate under distribution shift, we follow a recent line of work that studies optimization algorithms via the analysis of their behavior in continuous-time limits (Krichene & Bartlett, 2017; Li et al., 2017; Chaudhari et al., 2018; Shi et al., 2020). Specifically, for SGD this amounts to studying stochastic differential equations (SDEs) as an approximation for discrete stochastic updates. The construction of this correspondence is based on the Euler–Maruyama method. We assume that the step sizes in SGD are given by $\eta_t = \varepsilon \zeta(\varepsilon t)$, where $\zeta(t) \in [0, 1]$ is the adjustment factor and ε is the maximum allowed learning rate. In addition, the batch sizes are given by $B_t = \varepsilon \nu(\varepsilon t)$, for sufficiently regular functions $\zeta, \nu : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$.[1]

We use t to denote the iteration number of SGD and use τ as the continuous time variable for the corresponding SDE. We show that the trajectory of SGD updates can be approximated by the solution of the following SDE:

$$\begin{aligned} \mathrm{d}X(\tau) &= -(\zeta(\tau)X(\tau) + Y'(\tau))\mathrm{d}\tau \\ &+ \tfrac{\zeta(\tau)}{\sqrt{\nu(\tau)}} \left((\|X(\tau)\|^2 + \sigma^2)I + X(\tau)X(\tau)^\mathsf{T}\right)^{1/2} \mathrm{d}W(\tau) , \end{aligned} \tag{8}$$

where $X(0) = \theta_0 - \theta_0^*$ and $Y(\tau)$ is a sufficiently smooth curve so that $Y(\varepsilon t) = \theta_t^*$. Further, $W(\tau)$ is d-dimensional vector with each entry being a standard Brownian motion, independent from other entries. To make this connection, we posit the following assumptions:

A1. The functions $\zeta(\tau)$ and $\zeta(\tau)/\sqrt{\nu(\tau)}$ are bounded Lipschitz: $\|\zeta\|_\infty, \|\zeta\|_{\mathrm{Lip}}, \|\zeta/\sqrt{\nu}\|_\infty, \|\zeta/\sqrt{\nu}\|_{\mathrm{Lip}} \leq K$.

A2. The function $Y(\tau)$ is bounded Lipschitz: $\|Y(\tau)\| \leq K$ and $\|Y'(\tau)\| \leq \Gamma/\varepsilon$, for constants $K, \Gamma > 0$. Recall that $Y(\tau)$ is the continuous interpolation of the sequence models θ_t^* and therefore $Y'(\tau)$ controls how fast θ_t^* are changing and is a measure of distribution shift in the response variable y_{tk} in (7).

[1] More precisely, $B_t = \lceil \varepsilon \nu(\varepsilon t) \rceil$ must be an integer, however, the rounding effect is negligible in the continuous time analysis.

In (A1) we use the notation $\|f\|_{\mathrm{Lip}} := \sup_{x\neq y} |f(x) - f(y)|/|x-y|$ to indicate the Lipschitz norm of a function and $\|f\|_\infty := \sup_x |f(x)|$.

Proposition 3.1. *For any fixed $T, u > 0$, there exists a constant $C = C(K, \Gamma, d, \sigma, T, u)$, with parameters K, Γ given in Assumptions A1-A2, such that with probability at least $1 - e^{-u^2}$ we have*

$$\sup_{t\in[0,T/\varepsilon]\cap\mathbb{Z}_{\geq 0}} \left| \|X_{t\varepsilon}\|^2 - \|\theta_t - \theta_t^*\|^2 \right| \leq C\sqrt{\varepsilon}\,.$$

We defer the proof of this proposition and the exact expression for the constant C to Appendix A.1.

The expected regret at time t works out as:

$$\begin{aligned}\mathbb{E}[\mathsf{reg}_t] &= \mathbb{E}[\bar{\ell}_t(\theta_t) - \bar{\ell}_t(\theta_t^*)] \\ &= \frac{1}{2}\mathbb{E}[(\langle x_{tk}, \theta_t - \theta_t^*\rangle + \varepsilon_{tk})^2] - \frac{1}{2}\mathbb{E}[\varepsilon_{tk}^2] \\ &= \frac{1}{2}\mathbb{E}[\|\theta_t - \theta_t^*\|^2]\,.\end{aligned}$$

Using Proposition 3.1, $|\mathbb{E}[\mathsf{reg}_t] - \frac{1}{2}\mathbb{E}[\|X(t\varepsilon)\|^2]| \leq C\sqrt{\varepsilon}$. Henceforth, we focus on analyzing the second moment of the process X, as ε can be fixed to an arbitrarily small value.

For $X(\tau)$ the solution of SDE (8), we define

$$m_\tau := \mathbb{E}[X(\tau)] \in \mathbb{R}^d, \quad v_\tau := \mathbb{E}[\|X(\tau)\|^2]\,. \tag{9}$$

In our next theorem, we derive an ODE for m_τ and v_τ, using Itô's lemma from stochastic calculus (Oksendal, 2013). The proof is deferred to Section A.2.

Theorem 3.2. *Consider the SDE problem (8), and the moments m_τ and v_τ given by (9). We have*

$$m_\tau' = -\zeta(\tau)m_\tau - Y'(\tau)\,, \tag{10}$$

$$\begin{aligned} v_\tau' &= \Big((d+1)\frac{\zeta(\tau)^2}{\nu(\tau)} - 2\zeta(\tau)\Big)v_\tau \\ &\quad + \frac{\zeta(\tau)^2}{\nu(\tau)}\sigma^2 d - 2m_\tau^\mathsf{T} Y'(\tau)\,. \end{aligned} \tag{11}$$

It is worth noting that from the above ODEs, larger distribution shift (quantified by the $Y'(\tau)$ term) increases the drift in m_τ as well as the drift in v_τ via the term $m_\tau^\mathsf{T} Y'(\tau)$. In this case, the learner needs to choose a larger step size $\zeta(\tau)$ to reduce the drift in m_τ, which is consistent with our message that in dynamic environments the learning rate should often be set higher.

The problem of finding an optimal learning rate can be seen as an optimal control problem, where the state of the system (m_τ, v_τ) evolves according to ODEs (10)–(11), the control variables ζ can take values in the set of Borel-measurable functions from $[0, T]$ to $[0, 1]$, and the goal is to minimize the cost functional $\int_0^T v_\tau \mathrm{d}\tau$. The optimal learning rate schedule can then be solved exactly by dynamic programming, using the Hamilton–Jacobi–Bellman equation (Bellman, 1956). However, the optimal learning rate will depends on $Y'(\tau)$, which is a d-dimensional time-varying vector. We next do a simplification to reduce the dependence to $\|Y'(\tau)\|$.

Note that $|m_\tau^\mathsf{T} Y'(\tau)| \leq \|Y'(\tau)\|\ \|m_\tau\| \leq \|Y'(\tau)\|\sqrt{v_\tau}$. The first inequality becomes tight if the shift $Y'(\tau)$ is aligned with the expected error m_τ. The second inequality becomes tighter as the batch size grows, since it reduces the variance in $X(\tau)$, which by (9) is given by $v_\tau - \|m_\tau\|^2$. Therefore, we have

$$v_\tau' \leq \Big((d{+}1)\frac{\zeta(\tau)^2}{\nu(\tau)} - 2\zeta(\tau)\Big)v_\tau + \frac{\zeta(\tau)^2}{\nu(\tau)}\sigma^2 d + 2\|Y'(\tau)\|\sqrt{v_\tau}.$$

With this observation and the fact that our objective is to minimize the cost $\int_0^T v_\tau \mathrm{d}\tau$, we consider the process $\tilde{v}_\tau$ defined using the upper bound on v_τ', namely

$$\begin{aligned}\tilde{v}_\tau' &= \Big((d+1)\frac{\zeta(\tau)^2}{\nu(\tau)} - 2\zeta(\tau)\Big)\tilde{v}_\tau \\ &\quad + \frac{\zeta(\tau)^2}{\nu(\tau)}\sigma^2 d + 2\|Y'(\tau)\|\sqrt{\tilde{v}_\tau}\,.\end{aligned} \tag{12}$$

Our next result characterizes an optimal learning rate with respect to process $\tilde{v}_\tau$.

Theorem 3.3. *Consider the control problem*

$$\underset{\zeta:[0,T]\to[0,1]}{\text{minimize}} \int_0^T \tilde{v}_\tau \mathrm{d}\tau\,, \qquad \text{subject to constraint (12)}\,.$$

The optimal policy ζ is given by

$$\zeta^*(\tau) = \min\left\{1, \left(\frac{d+1}{\nu(\tau)}\tilde{v}_\tau + \frac{\sigma^2 d}{\nu(\tau)}\right)^{-1}\tilde{v}_\tau\right\}. \tag{13}$$

Using the policy $\zeta^*(\tau)$ given by (13) and (12), we get an ODE that can be solved for v_τ and then plugged back into (13) to obtain an optimal policy $\zeta^*(\tau)$ and hence optimal learning rate. We formalize this approach in Algorithm 1, where we solve the ODE for $\tilde{v}_\tau$ (after substituting for optimal $\zeta^*(\tau)$) using the (forward) Euler method. Translating from the continuous domain to the discrete domain, we use the relations $\eta_t = \varepsilon\zeta(\varepsilon t)$, $B_t = \varepsilon\nu(\varepsilon t)$, and $\|Y'(\varepsilon t)\| \approx \|\theta_{t+1}^* - \theta_t^*\|/\varepsilon = \gamma_t/\varepsilon$.

Remark 3.4. The learning rate schedule proposed in Algorithm 1 is an online schedule in the sense that η_t is determined based on the history up to time t, i.e., it does does not look into future.

Remark 3.5. The proposed learning rate in Algorithm 1 depends on the distribution shifts γ_t. In settings where γ_t

Algorithm 1 Optimal learning rate schedule for linear regression undergoing distribution shift.

Input: max step size ε, discretization scale $\kappa \in (0, 1]$
Output: step sizes η_t^*
Initialization: $v \leftarrow 0$
for $t = 1, 2, \ldots$ **do**
  for $j = 1, 2, \ldots, \lceil 1/\kappa \rceil$ **do**
    $r \leftarrow \min\left(\frac{vB_t}{(d+1)v+\sigma^2 d}, \varepsilon\right)$
    $v \leftarrow v + \kappa\left(\frac{d+1}{B_t}r^2 - 2r\right)v + \kappa\frac{\sigma^2 d}{B_t}r^2 + 2\kappa\gamma_{t-1}\sqrt{v}$
  end for
  $\eta_t^* \leftarrow \min\left(\frac{vB_t}{(d+1)v+\sigma^2 d}, \varepsilon\right)$
end for

is not revealed (even after the learner proceeds to the next round), we estimate γ_t using an exponential moving average of the drifts in the consecutive estimated models θ_t, namely $\hat{\gamma}_t = \beta\hat{\gamma}_{t-1} + (1-\beta)\|\theta_t - \theta_{t-1}\|$, with a factor $\beta \in (0, 1)$.

Figure 2 shows the learning rate schedule η_t^* given by Algorithm 1:

- **Bursty shifts.** The left subplot corresponds to the setting where γ_t follows a jump process. At the beginning of each episode (40 steps each), γ_t jumps to a value s and then becomes zero for the rest of the episode. Therefore, the distribution remains the same within an episode but then switches to another distribution in the next episode. In this case, we see the learning rate restart at the beginning of each episode with a more aggressive step size (capped at $\varepsilon = 0.1$) but then decrease within the episode as there is no shift.
- **Smooth shifts.** The right subplot illustrates the setting where γ_t changes continuously as $\gamma_t = 1/t^\alpha$ for a constant value α. We see that a smaller value of α (i.e., larger distribution shift) induces a larger learning rate.

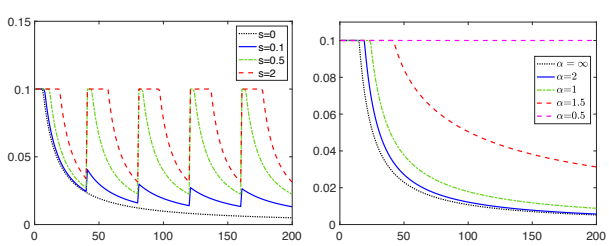

Figure 2: Learning rate schedules η_t^* devised in Algorithm 1 for online linear regression. The batch size is $B_t = 100$ for all $1 \le t \le 200$, dimension $d = 100$, max step size $\varepsilon = 0.1$, and $\sigma = 2$.

3.1. Case study: No distribution shift

To build further insight about the proposed schedule, we study the behavior of Algorithm 1 when there is no shift in the data distribution and the batch size is the same across SGD iterations. Note that in this case, $Y'(\tau) = 0$ and $\nu(\tau) = B/\varepsilon$. The behavior of the learning rate schedule η_t^* is described in the next lemma.

Lemma 3.6. *Consider the following ODE:*

$$\tilde{v}_\tau' = \Big(\mathsf{a}\zeta(\tau)^2 - 2\zeta(\tau)\Big)\tilde{v}_\tau + \mathsf{b}\zeta(\tau)^2 \qquad (14)$$
$$\mathsf{a} := \varepsilon\frac{d+1}{B}, \quad \mathsf{b} := \varepsilon\frac{\sigma^2 d}{B},$$

with optimal $\zeta(\tau)$ given by (13). Define

$$\tau_* := \left[\frac{1}{2-\mathsf{a}}\log\left((1-\mathsf{a})\left(\tilde{v}_0\frac{2-\mathsf{a}}{\mathsf{b}} - 1\right)\right)\right]_+ ,$$
$$C = \mathsf{a}\ln\left(\frac{1-\mathsf{a}}{\mathsf{b}}\right) + 1 - \mathsf{a} - \tau_* .$$

We then have the following:

- *If $\tau \le \tau_*$, then*
$$\tilde{v}_\tau = \left(\tilde{v}_0 - \frac{\mathsf{b}}{2-\mathsf{a}}\right)e^{-(2-\mathsf{a})\tau} + \frac{\mathsf{b}}{2-\mathsf{a}}, \quad \zeta(\tau) = 1 .$$
- *As $\tau \to \infty$, we have*
$$\lim_{\tau\to\infty}\frac{\tilde{v}_\tau}{\frac{b}{\tau+C}} = 1 , \quad \lim_{\tau\to\infty}\frac{\zeta(\tau)}{\frac{1}{\mathsf{a}+C+\tau}} = 1.$$

Recalling the relation $\eta_t = \varepsilon\zeta(\varepsilon t)$ and using Lemma 3.6, we have $\eta_t^* = \varepsilon$ for $t \le t_* := \lceil \tau_*/\varepsilon \rceil$ and

$$\lim_{t\to\infty}\frac{\eta_t^*}{\frac{\varepsilon}{\mathsf{a}+C+\varepsilon t}} = 1.$$

In words, η_t^* asymptotically has the rate $1/t$. In Figure 3, we plot an example of processes $\tilde{v}_\tau$ and the optimal learning rate η_t^* for linear regression without any distribution shift.

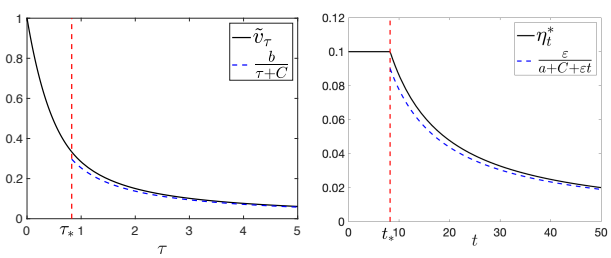

Figure 3: The process $\tilde{v}_\tau$ defined by ODE (14) if there is no distribution shift (left). Here we have $\varepsilon = 0.1$, $\mathsf{a} := \varepsilon(d+1)/B = 0.1$, $\mathsf{b} := \varepsilon\sigma^2 d/B = 0.3$, and initialization $\tilde{v}_0 = 1$. Behavior of the learning rate schedule η_t^* given by Algorithm 1, which asymptotically has the rate $1/t$ (right).

4. General convex loss

4.1. Upper bound on the total regret

Here we derive an upper bound on the total regret for general convex loss functions. We use this bound to study the behavior of optimal learning rates (by minimizing the regret upper bound) with respect to distribution shifts. We proceed by making the following assumption.

Assumption 4.1. Suppose that

(i) We have $\mathbb{E}_{P_t}[\|\nabla\ell(\theta_t, z_{t,k}) - \nabla\bar{\ell}_t(\theta_t)\|^2] \leq \sigma^2$, for some parameter $\sigma \geq 0$. Since the data points in each batch are sampled i.i.d., this implies that

$$\mathbb{E}_{P_t}\left[\left\|\frac{1}{B_t}\sum_{k=1}^{B_t}\nabla\ell(\theta_t, z_{t,k}) - \nabla\bar{\ell}_t(\theta_t)\right\|^2\right] \leq \frac{\sigma^2}{B_t}\,.$$

(ii) We have $\nabla^2\bar{\ell}_t(\theta) \preceq LI$ for $\theta \in \Theta$, or a weaker L-smooth condition

$$\|\nabla\bar{\ell}_t(\theta_1) - \nabla\bar{\ell}_t(\theta_2)\| \leq L\|\theta_1 - \theta_2\|,$$

for $\theta_1, \theta_2 \in \Theta$.

(iii) We assume the oracle models θ_t^* are in Θ and that the diameter of Θ is bounded by $D_{\max}$. Alternatively, we assume that $\theta_t^* \in \Theta'$ for all t, and $D_{\max} = \max\{\|\theta - \theta'\| : \theta \in \Theta, \theta' \in \Theta'\}$.

Note that for all steps t, $\nabla\ell(\theta_t, z_{t,k})$ is an unbiased estimator of $\nabla\bar{\ell}_t(\theta_t)$ and Assumption (i) bounds its variance. Assumption (ii) is for technical analysis and is satisfied if the loss function has a continuous Hessian. Assumption (iii) assumes that the oracle models θ_t^* remain in a bounded set as t grows. Since in practice the SGD is run for a finite number of iterations, this is not a restricting assumption, e.g., $D_{\max}$ can depend on the horizon length T.

Theorem 4.2. *Suppose the loss function $\ell(\theta, z)$ is convex in θ, and assume that the oracle model θ_t^* and the learning rate η_t are adapted to the history $\boldsymbol{z}_{t-1}$, defined by (5). Let $D_t := \|\theta_t^* - \theta_t\|$ and $a_t := 2\eta_t - L\eta_t^2 > 0$ for $t \geq 1$. Under Assumption 4.1, and assuming $\eta_t \leq \frac{1}{L}$, for all $t \geq 1$, the following bound holds on the total regret of SGD:*

$$\mathbb{E}[\mathsf{Reg}(T)] \leq \sum_{t=1}^{T}\mathbb{E}\Bigg[\left(\frac{D_t^2}{a_t} - \frac{D_{t+1}^2}{a_t}\right) + \frac{\sigma^2\eta_t^2}{B_t a_t} + \frac{\|\theta_t^* - \theta_{t+1}^*\|^2}{a_t} + \frac{2}{a_t}\langle\theta_t^* - \theta_{t+1}^*, \theta_{t+1} - \theta_t^*\rangle\Bigg]\,. \tag{15}$$

Here, the expectation is with respect to the randomness in data points observed in the T steps.

We next discuss how the regret bound (15) can be used to derive optimal learning rate schedules. We would like to derive optimal rates η_t^* by minimizing the bound (15) in a sequential manner. However, the bound depends on D_t and θ_{t+1}, which are not observable. Indeed, θ_{t+1} is defined at step $t+1$ where η_t should have already been determined. To address this issue, we use the fact that the projected SGD updates remain in the set Θ and by invoking Assumption (iii), we have $D_t \leq D_{\max}$ and $\|\theta_{t+1} - \theta_t^*\| \leq D_{\max}$. Also recall our notation $\gamma_t = \|\theta_t^* - \theta_{t+1}^*\|$ for the distribution shift. Therefore, by rearranging the terms in (15) and telescope summing over $1/a_t$, we have

$$\mathbb{E}[\mathsf{Reg}(T)] \leq D_{\max}^2\mathbb{E}\left[\frac{1}{a_1} + \sum_{t=2}^{T}\left(\frac{1}{a_t} - \frac{1}{a_{t-1}}\right)_+\right] + \sum_{t=1}^{T}\mathbb{E}\left[\frac{1}{a_t}\left(\frac{\sigma^2\eta_t^2}{B_t} + \gamma_t^2 + 2D_{\max}\gamma_t\right)\right], \tag{16}$$

where $x_+ = \max(x, 0)$ indicates the positive part of x.

We next discuss the choice of learning rates that minimizes the upper bound (16) in a sequential manner. Conditioned on $\boldsymbol{z}_{[t-1]}$, the optimal η_t is given by

$$\eta_t^* := \operatorname*{argmin}_{0\leq\eta\leq\frac{1}{L}}\left\{D_{\max}^2\left(\frac{1}{2\eta - L\eta^2} - \frac{1}{2\eta_{t-1} - L\eta_{t-1}^2}\right)_+ + \frac{\sigma^2}{B_t}\cdot\frac{\eta^2}{2\eta - L\eta^2} + \frac{\gamma_t^2 + 2D_{\max}\gamma_t}{2\eta - L\eta^2}\right\}. \tag{17}$$

Our next proposition characterizes η_t^*.

Proposition 4.3 (Learning rate schedule)**.** *Define the thresholds $\tau_{1,t}$ and $\tau_{2,t}$ as follows:*

$$\tau_{1,t} := \tfrac{B_t}{2\sigma^2}\left(\sqrt{b_{1,t}^2L^2 + \tfrac{4\sigma^2}{B_t}b_{1,t}} - b_{1,t}L\right), \tag{18}$$

$$\tau_{2,t} := \tfrac{B_t}{2\sigma^2}\left(\sqrt{b_{2,t}^2L^2 + \tfrac{4\sigma^2}{B_t}b_{2,t}} - b_{2,t}L\right), \tag{19}$$

$$b_{1,t} := \gamma_t^2 + 2D_{\max}\gamma_t, \quad b_{2,t} := (\gamma_t + D_{\max})^2\,.$$

The optimal learning rate η_t^* defined by (17) is given by:

$$\eta_t^* = \begin{cases}\tau_{1,t} & \text{if } \eta_{t-1}^* \leq \tau_{1,t},\\ \eta_{t-1}^* & \text{if } \tau_{1,t} \leq \eta_{t-1}^* \leq \tau_{2,t}\\ \tau_{2,t} & \text{if } \eta_{t-1}^* \geq \tau_{2,t}\,.\end{cases} \tag{20}$$

Remark 4.4. The proposed learning rate in (17) depends on σ, L and shifts γ_t. Having access to the loss function $\ell(\theta, z)$, the learner can use sample estimates for σ, L. Also note that we can use any upper bound on γ_t in the bound (16) and obtain a similar schedule. Of course, if the upper bound is crude, it results in a conservative learning rate schedule. In settings where an upper bound on the shifts γ_t is not available, we estimate γ_t using an exponential moving average of the drifts in the consecutive estimated models θ_t, namely $\hat{\gamma}_t = \beta\hat{\gamma}_{t-1} + (1-\beta)\|\theta_t - \theta_{t-1}\|$, with a factor $\beta \in (0, 1)$.

Remark 4.5. The values $b_{1,t}$ and $b_{2,t}$ in (18) and (19) are increasing in the distribution shift γ_t and it is easy to see that the thresholds $\tau_{1,t}, \tau_{2,t}$ are also increasing in γ_t. As a result for every value of η_{t-1}, higher distribution shift γ_t increases the optimal learning rate η_t^*.

Note that Theorem 4.2 and Remark 4.5 are optimized with respect to the upper envelope of the optimal regret. We also prove a corresponding lower envelope result for SGD.

4.2. Lower bound on the total regret

The learning rate schedule in 4.3 is optimized with respect to the upper bound derived for the cumulative dynamic regret. We next prove a corresponding lower bound result for SGD, which matches the upper bound and only differs by constants. Thus, our analysis of the optimal learning rate schedules for SGD is tight up to constants.

Before we begin, we make an additional assumption.

Assumption 4.6. We assume that the loss function $\ell(\theta, z)$ is μ-strongly convex in θ, for some $\mu > 0$, i.e., $\ell(\theta) - \frac{\mu}{2}\|\theta\|^2$ is convex in θ.

Theorem 4.7. *Suppose the oracle model θ_t^* and the learning rate η_t are adapted to the history $\boldsymbol{z}_{t-1}$, defined by (5). Let $D_t := \|\theta_t^* - \theta_t\|$, $\gamma_t := \|\theta_t^* - \theta_{t+1}^*\|$, and $a_t' := 2(\eta_t + \frac{L}{\mu}\eta_t - \eta_t^2 L)$. Under Assumptions 4.1 and 4.6, and assuming $\eta_t \le \frac{1}{\mu}$, for all $t \ge 1$, we have the following bound on the total regret of the batch SGD:*

$$\mathbb{E}[\mathsf{Reg}(T)] \ge \sum_{t=1}^{T} \mathbb{E}\Bigg[\left(\frac{D_t^2}{a_t'} - \frac{D_{t+1}^2}{a_t'} \right) + \frac{\sigma^2 \eta_t^2}{B_t a_t'} + \frac{\|\theta_t^* - \theta_{t+1}^*\|^2}{a_t'} + \frac{2}{a_t'} \langle \theta_t^* - \theta_{t+1}^*, \theta_{t+1} - \theta_t^* \rangle \Bigg], \quad (21)$$

where the expectation is with respect to the randomness in data points observed in the T steps.

5. Non-convex loss

When the loss function ℓ is non-convex, SGD like any other first order method can get trapped in a local minimum or a saddle point of the landscape. When there is no distribution shift, there is a line of work showing that SGD can efficiently escape saddle points if the step size is large enough (Lee et al., 2016; Jin et al., 2017). This superiority of SGD in non-convex settings is often attributed to the stochasticity of the gradients, which significantly accelerates the escape from saddle points.

In non-convex settings one cannot control convergence to a global minimum without making further structural assumption on the optimization landscape and the initialization of SGD. In view of that, we propose to consider the following notion of regret based on the cumulative gradient norm of the SGD trajectory:

$$\mathsf{Reg}(T) := \sum_{t=1}^{T} \|\nabla \bar{\ell}_t(\theta_t)\|^2 . \quad (22)$$

In words, the regret is defined with respect to the norm of gradient at the sequence of estimated models. This notion does not differentiate between local or global minima.

Further, due to the complex landscapes of non-convex loss, we work with a more holistic measure of distribution shift, namely

$$\gamma_t := \sup_{\theta \in \mathbb{R}^p} |\bar{\ell}_t(\theta) - \bar{\ell}_{t+1}(\theta)| . \quad (23)$$

Recall that $\bar{\ell}_t = \mathbb{E}_{P_t}[\ell(\theta, z_{t,k})]$ and obviously if there is no shift at step t, i.e., $P_t = P_{t+1}$ then $\gamma_t = 0$. In contrast, in the convex setting, we measure the distribution shift only in terms of the difference between the global minimizers of $\bar{\ell}_t$ and $\bar{\ell}_{t+1}$, cf. Definition 2.1.

We can now state our regret bound in the non-convex setting.

Theorem 5.1. *Suppose the learning rates η_t are adapted to the history $\boldsymbol{z}_{t-1}$, defined by (5). Let γ_t be defined as (23), and define $a_t := 2\eta_t - L\eta_t^2$, for $t \ge 1$. Under Assumption 4.1 $(i), (ii)$, and assuming $\eta_t \le \frac{1}{L}$, for all $t \ge 1$, we have the following bound on the total regret of batch SGD:*

$$\mathbb{E}[\mathsf{Reg}(T)] \le \mathbb{E}\left[\frac{2\bar{\ell}_1(\theta_1)}{a_1^2} + \sum_{t=2}^{T} 2\bar{\ell}_t(\theta_t) \left(\frac{1}{a_t} - \frac{1}{a_{t-1}^2} \right) \right] + \sum_{t=1}^{T} \mathbb{E}\left[\frac{1}{a_t} \cdot \left(\frac{L\sigma^2\eta_t^2}{B_t} + 2\gamma_t \right) \right] . \quad (24)$$

The theorem above has a very similar format to the bound derived in Theorem 4.2. By minimizing the regret of the upper bound (24) in sequential manner conditioned on $\boldsymbol{z}_{[t-1]}$, the optimal learning rate is given by

$$\eta_t^* := \arg\min_{0 \le \eta \le \frac{1}{L}} \frac{2\bar{\ell}_t(\theta_t) + 2\gamma_t}{2\eta - L\eta^2} + \frac{L\sigma^2}{B_t} \cdot \frac{\eta^2}{2\eta - L\eta^2} . \quad (25)$$

The optimal η_t^* admits a closed form solution given below:

$$\eta_t^* = \frac{B_t}{L\sigma^2} \left(\sqrt{b_t^2 + 2\frac{\sigma^2}{B_t} b_t} - b_t \right) , \; b_t = L(\gamma_t + \bar{\ell}_t(\theta_t)) .$$

The above characterization is derived by noticing that the function in (25) is convex in η, for $\eta \in (0, 1/L]$ and the stationary point of the function η^* satisfies the boundary condition $0 \le \eta^* \le 1/L$.

It is easy to see that the learning rate η_t^* is increasing in the distribution shift γ_t. To implement this learning rate, we estimate $\bar{\ell}_t(\theta_t)$ by $\ell^{B_t}(\theta_t)$, its sample average over the batch at time t. The proofs are deferred to the supplementary materials due to the space constraint.

6. Experiments

We implement these experiments using TensorFlow (Abadi et al., 2016) and Keras (Chollet et al., 2015).[2] We study high-dimensional regression in Section 6.1 and an application of neural networks to flow cytometry in Appendix E.

6.1. High-dimensional regression

We use the learning rate schedules in Algorithm 1 and Proposition 4.3 for linear and logistic regression, respectively. We consider paths $\{\theta_t^*\}_{t=1}^T$ such that for $\theta_t^* \in \mathbb{R}^d, i \in [d]$,

$$\theta_t^*(i) = \begin{cases} r_{a,b}(t)^3 \cos(\lceil i/2 \rceil 2k\pi\alpha(t)) & \text{if } i \text{ odd}, \\ r_{a,b}(t)^3 \sin(\lceil i/2 \rceil 2k\pi\alpha(t)) & \text{if } i \text{ even}, \end{cases} \tag{26}$$

where $r_{a,b}(t) = \texttt{linspace}(a, b, T)$ controls the radius, $\alpha(t) = \texttt{linspace}(0, 1, T)$, and k is the base frequency. These paths have linearly independent components due to their trigonometric frequencies and phases (useful for high dimensions), and move at non-monotonic speeds if $a \neq b$.

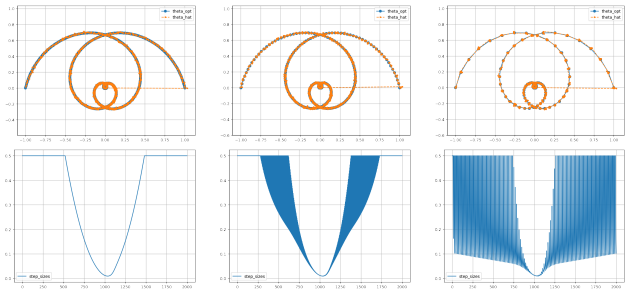

Figure 4: SGD trajectories of Algorithm 1 (top); and oscillating learning rates η_t as we discretize the path defined by θ_t^* where $\eta_{\max} = 0.5$ (bottom).

Linear regression. We start by investigating Algorithm 1 for online least squares. Setting $\theta_0 = 0$, at each step t we generate $X \in \mathbb{R}^{B_t \times d}$ for $x_{ij} \sim \mathsf{N}(0, 1)$ and get back the response $y = X\theta_t^* + \varepsilon$ for $\varepsilon_i \sim \mathsf{N}(0, 0.1)$.

Consider the 2-dimensional trajectory in Figure 4 defined by $r_{1,-1}(t)$, $k = 4$, and $B_t = 256$. For $T = 2000$, the path starts at $\theta_1^* = (1, 0)$, spirals into the origin, and returns to $\theta_T^* = (-1, 0)$. To study the effect of *continuous vs discrete distribution shifts*, we downsample the points by $\ell \in \{1, 4, 16\}$ to get the discretized paths

$$\hat{\theta}_t^* = \theta_{\lceil t/\ell \rceil \ell}^*,$$

for $t \in [T]$. As ℓ increases (i.e., from left to right in Figure 4), the learning rate η_t of Algorithm 1 starts to oscillate—decreasing when θ_t is near θ_t^* and returning to $\eta_{\max} = 0.5$ when θ_t^* shifts.

[2]The source code is available at https://github.com/fahrbach/learning-rate-schedules.

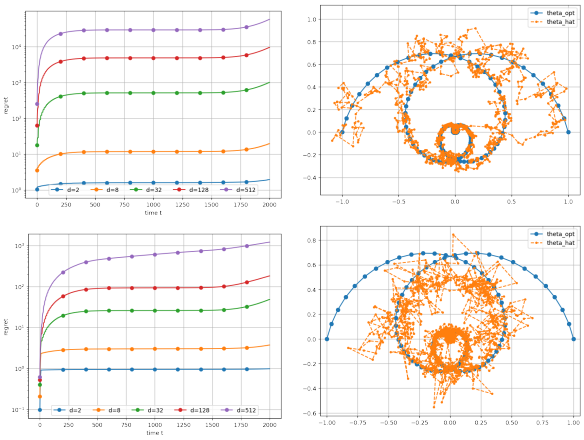

Figure 5: Cumulative regret of Algorithm 1 with $\eta_{\max} = 1/\sqrt{d}$ for increasing dimensions d (top-left); and the first and second coordinates of the SGD for $d = 128$ and batch size $B_t = 256$ (top-right). Cumulative regret of Proposition 4.3 for d-dimensional logistic regression (bottom-left); and the first and second coordinates of the SGD for $d = 128$ and batch size $B_t = 256$ (bottom-right).

Next, we increase the dimension d and plot the cumulative regret of Algorithm 1 in Figure 5. We use the same $\ell = 8$ discretized paths and set $\eta_{\max} = 1/\sqrt{d}$. Note that for all values of d, the total regret increases, levels off, and then increases again. This corresponds to θ_t^* spiraling into the origin, spending time there, and exiting. The initial spike in regret is due to finding the θ_t^* path, i.e., the first few steps when θ_t moves from the origin to θ_t^*.

Logistic regression. We also empirically study the learning rate schedule in Proposition 4.3 for d-dimensional logistic regression with binary cross entropy loss. Similar to the linear regression experiments, at each step t we generate the covariates $X \in \mathbb{R}^{B_t \times d}$, but now we get back $y = \text{sigmoid}(X\theta_t^* + \varepsilon)$. We note that the learning rate schedule in Proposition 4.3 is largely parameter-free for generalized linear models. For example, setting $\sigma^2 = d/4$ and $L = 1/4$ minimizes the upper bound on the regret in (16) for logistic regression with log loss, so the only hyperparameter we set is $D_{\max} = d$.

Conclusion

This work explores learning rate schedules that minimize regret for online SGD-based learning in the presence of distribution shifts. We derive a novel stochastic differential equation to approximate the SGD path for linear regression with model shifts, and we derive new adaptive schedules for general convex and non-convex losses that minimize regret upper bounds. These learning rate schedules can increase in the presence of distribution shifts and allow for more aggressive optimization.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. TensorFlow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation*, pp. 265–283, 2016.

Anil, R., Gadanho, S., Huang, D., Jacob, N., Li, Z., Lin, D., Phillips, T., Pop, C., Regan, K., Shamir, G. I., et al. On the factory floor: ML engineering for industrial-scale ads recommendation models. *arXiv preprint arXiv:2209.05310*, 2022.

Bastidas-Ponce, A., Sophie Tritschler, L. D., Scheibner, K., Tarquis-Medina, M., Salinno, C., Schirge, S., Burtscher, I., Böttcher, A., Theis, F. J., Lickert, H., and Bakht, M. Comprehensive single cell mRNA profiling reveals a detailed roadmap for pancreatic endocrinogenesis. *Development*, 146(12):dev173849, 2019.

Bellman, R. Dynamic programming and lagrange multipliers. *Proceedings of the National Academy of Sciences*, 42(10):767–769, 1956.

Bengio, Y. *Practical Recommendations for Gradient-Based Training of Deep Architectures*, pp. 437–478. Springer Berlin Heidelberg, 2012.

Bergen, V., Lange, M., Peidli, S., Wolf, F. A., and Theis, F. Generalizing rna velocity to transient cell states through dynamical modeling. *Nature Biotechnology*, 38:1408–1414, 2020.

Besbes, O., Gur, Y., and Zeevi, A. Non-stationary stochastic optimization. *Operations Research*, 63(5):1227–1244, 2015.

Chaudhari, P., Oberman, A., Osher, S., Soatto, S., and Carlier, G. Deep relaxation: partial differential equations for optimizing deep neural networks. *Research in the Mathematical Sciences*, 5(3):1–30, 2018.

Chollet, F. et al. Keras. https://github.com/fchollet/keras, 2015.

Coleman, B., Kang, W.-C., Fahrbach, M., Wang, R., Hong, L., Chi, E. H., and Cheng, D. Z. Unified Embedding: Battle-tested feature representations for web-scale ML systems. *arXiv preprint arXiv:2305.12102*, 2023.

Fan, J. and Zhang, W. Statistical methods with varying coefficient models. *Statistics and its Interface*, 1(1):179–195, 2008.

Fang, Y., Xu, J., and Yang, L. Online bootstrap confidence intervals for the stochastic gradient descent estimator. *Journal of Machine Learning Research*, 19(78):1–21, 2018.

Ge, R., Kakade, S. M., Kidambi, R., and Netrapalli, P. The step decay schedule: A near optimal, geometrically decaying learning rate procedure for least squares. *Advances in neural information processing systems*, 32, 2019.

Gronwall, T. H. Note on the derivatives with respect to a parameter of the solutions of a system of differential equations. *Annals of Mathematics*, pp. 292–296, 1919.

Hastie, T. and Tibshirani, R. Varying-coefficient models. *Journal of the Royal Statistical Society: Series B (Methodological)*, 55(4):757–779, 1993.

Hu, Z., Tang, A., Singh, J., and Butte, A. J. A robust and interpretable end-to-end deep learning model for cytometry data. *PNAS*, 2020.

Hu, Z., Bhattacharya, S., and Butte, A. J. Application of machine learning for cytometry data. *Frontiers in Immunology*, 2022.

Jadbabaie, A., Rakhlin, A., Shahrampour, S., and Sridharan, K. Online optimization: Competing with dynamic comparators. In *Artificial Intelligence and Statistics*, pp. 398–406. PMLR, 2015.

Jain, P., Nagaraj, D., and Netrapalli, P. Making the last iterate of SGD information theoretically optimal. In *Conference on Learning Theory*, pp. 1752–1755. PMLR, 2019.

Jin, C., Ge, R., Netrapalli, P., Kakade, S. M., and Jordan, M. I. How to escape saddle points efficiently. In *International Conference on Machine Learning*, pp. 1724–1732. PMLR, 2017.

Johnson, R. and Zhang, T. Accelerating stochastic gradient descent using predictive variance reduction. *Advances in Neural Information Processing Systems*, 26, 2013.

Kamath, S., Deshpande, A., and Subrahmanyam, K. How do SGD hyperparameters in natural training affect adversarial robustness? *arXiv preprint arXiv:2006.11604*, 2020.

Keskar, N. S., Mudigere, D., Nocedal, J., Smelyanskiy, M., and Tang, P. T. P. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*, 2016.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Krichene, W. and Bartlett, P. L. Acceleration and averaging in stochastic descent dynamics. *Advances in Neural Information Processing Systems*, 30, 2017.

Lee, J. D., Simchowitz, M., Jordan, M. I., and Recht, B. Gradient descent only converges to minimizers. In *Conference on learning theory*, pp. 1246–1257. PMLR, 2016.

Li, Q., Tai, C., and Weinan, E. Stochastic modified equations and adaptive stochastic gradient algorithms. In *International Conference on Machine Learning*, pp. 2101–2110. PMLR, 2017.

Li, Y., Mahjoubfar, A., Chen, C. L., Niazi, K. R., Pei, L., and Jalali, B. Deep cytometry: Deep learning with real-time inference in cell sorting and flow cytometry. *Scientific Reports*, 2019.

Li, Z. and Arora, S. An exponential learning rate schedule for deep learning. *arXiv preprint arXiv:1910.07454*, 2019.

Loshchilov, I. and Hutter, F. SGDR: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

Oksendal, B. *Stochastic Differential Equations: An Introduction with Applications*. Springer Science & Business Media, 2013.

Polyak, B. T. and Juditsky, A. B. Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, 30(4):838–855, 1992.

Shi, B., Su, W. J., and Jordan, M. I. On learning rates and Schrödinger operators. *arXiv preprint arXiv:2004.06977*, 2020.

Smith, L. N. No more pesky learning rate guessing games. *CoRR, abs/1506.01186*, 5:575, 2015.

Smith, S. L., Kindermans, P.-J., Ying, C., and Le, Q. V. Don't decay the learning rate, increase the batch size. In *International Conference on Learning Representations*, 2018.

Tripuraneni, N., Flammarion, N., Bach, F., and Jordan, M. I. Averaging stochastic gradient descent on Riemannian manifolds. In *Proceedings of the 31st Conference On Learning Theory*, volume 75, pp. 650–687. PMLR, 2018.

Yang, T., Zhang, L., Jin, R., and Yi, J. Tracking slowly moving clairvoyant: Optimal dynamic regret of online learning with true and noisy gradient. In *International Conference on Machine Learning*, pp. 449–457. PMLR, 2016.

Yao, Z., Gholami, A., Lei, Q., Keutzer, K., and Mahoney, M. W. Hessian-based analysis of large batch training and robustness to adversaries. *Advances in Neural Information Processing Systems*, 31, 2018.

Zhang, S., Choromanska, A. E., and LeCun, Y. Deep learning with elastic averaging SGD. *Advances in Neural Information Processing Systems*, 28, 2015.

Zhou, X. On the Fenchel duality between strong convexity and Lipschitz continuous gradient. *arXiv preprint arXiv:1803.06573*, 2018.

Zinkevich, M. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th International Conference on Machine Learning*, pp. 928–936, 2003.

A. Proof of theorems and technical lemmas for linear regression

A.1. Proof of Proposition 3.1

The integration form of the stochastic differential equation (8) reads as

$$X(\tau) = X_0 + Y_0 - \int_0^\tau \zeta(s)X(s)\mathrm{d}s - Y(\tau) + \int_0^\tau \frac{\zeta(s)}{\sqrt{\nu(s)}} D_s^{1/2}\mathrm{d}W(s)\,, \tag{27}$$

where $D_s = ((\|X(s)\|^2 + \sigma^2)I + X(s)X(s)^\mathsf{T})$. We start by proving some useful bounds on the solution of $X(\tau)$ process.

Lemma A.1. *Consider the process $X(\tau)$ given by (27) with initialization X_0 satisfying $\|X_0\| \le K$. Under Assumptions* A1-A2*, with probability at least $1 - e^{-u^2}$ we have*

$$\sup_{\tau\in[0,T]} \|X(\tau)\| \le C\sqrt{T}(\sqrt{d}+u)\exp\Big[C\Big(T^2 + (\sqrt{d}+u)^2 T\Big)\Big]\,. \tag{28}$$

and

$$\sup_{t\in[0,T/\varepsilon]\cap\mathbb{Z}_{\ge 0}} \sup_{u\in[0,\varepsilon]} \|X(t\varepsilon+u) - X(t\varepsilon)\| \le C'\sqrt{T\varepsilon}(\sqrt{d}+u)^2\exp\Big[C\Big(T^2 + (\sqrt{d}+u)^2 T\Big)\Big]\,, \tag{29}$$

for any fixed $u > 0$, and constants $C = C(K,\sigma)$, $C' = C'(K,\sigma,\Gamma)$.

Proof of Lemma A.1. Define $V(\tau) := \int_0^\tau \frac{\zeta(s)}{\sqrt{\nu(s)}} D_s^{1/2}\mathrm{d}W(s)$. We have

$$\mathrm{Cov}(V(\tau)) = \int_0^\tau \frac{\zeta(s)^2}{\nu(s)} D_s \mathrm{d}s\,,$$

so then

$$\|\mathrm{Cov}(V(\tau))\|_{\mathrm{op}} \le K^2 \int_0^\tau \|D_s\|_{\mathrm{op}}\mathrm{d}s \le A_\tau := K^2\int_0^\tau (2\|X_s\|^2 + \sigma^2)\mathrm{d}s\,. \tag{30}$$

Note that $\exp\alpha\|V(\tau)\|^2$ is a submartingale, and by virtue of Doob's martingale inequality, we have

$$\mathbb{P}\Big(\sup_{\tau\le T}\|V(\tau)\| \ge \lambda\Big) \le \mathbb{E}[\exp\{\alpha\|V(T)\|/2\}]\exp\{-\alpha\lambda^2/2\} \le (1 - A_T\alpha)^{-d/2}\exp\{-\alpha\lambda^2/2\}\,.$$

Take $\alpha = 1/(2A_T)$ and $\lambda = 2\sqrt{A_T}(\sqrt{d}+u)$ to obtain

$$\mathbb{P}\left(\sup_{\tau\le T}\|V(\tau)\| \ge 2\sqrt{A_T}(\sqrt{d}+u)\right) \le 2^{d/2}\exp(-(\sqrt{d}+u)^2) \le e^{-u^2}\,. \tag{31}$$

Using (27) and recalling Assumptions A1-A2, we get

$$\begin{aligned}\|X(\tau)\| &\le \|X_0\| + \|Y_0\| + \|Y_\tau\| + \int_0^\tau \zeta(s)\|X(s)\|\mathrm{d}s + \|V(\tau)\| \\ &\le 3K + \int_0^\tau K\|X(s)\|\mathrm{d}s + \|V(\tau)\|\,.\end{aligned}$$

We next use the inequality $(a+b+c)^2 \le 3(a^2+b^2+c^2)$ to get

$$\begin{aligned}\|X(\tau)\|^2 &\le 27K^2 + 3K^2\Big(\int_0^\tau \|X(s)\|\mathrm{d}s\Big)^2 + 3\|V(\tau)\|^2 \\ &\le 27K^2 + 3K^2\tau\int_0^\tau \|X(s)\|^2\mathrm{d}s + 3\|V(\tau)\|^2\,,\end{aligned}$$

where in the second line we used Cauchy–Shwarz inequality. Define $\Delta_T = \sup_{\tau \le T} \|X(\tau)\|^2$. Taking the supremum over $\tau \le T$ of both sides of the previous inequality and using the bound (31), we arrive at

$$\begin{aligned}
\Delta_T &\le 27K^2 + 3K^2 T \int_0^T \Delta_s \mathrm{d}s + 12 A_T (\sqrt{d} + u)^2 \\
&\le 27K^2 + 3K^2 T \int_0^T \Delta_s \mathrm{d}s + 12 A_T (\sqrt{d} + u)^2 \\
&\le 27K^2 + 3K^2 T \int_0^T \Delta_s \mathrm{d}s + 12 (2K^2 \int_0^T \Delta_s \mathrm{d}s + \sigma^2 T K^2)(\sqrt{d} + u)^2 \\
&= 27K^2 + 12\sigma^2 T K^2 (\sqrt{d} + u)^2 + (3T + 24(\sqrt{d} + u)^2) K^2 \int_0^T \Delta_s \mathrm{d}s.
\end{aligned}$$

Using Gronwall's inequality, the above relation implies that

$$\Delta_T \le K^2 (27 + 12\sigma^2 T (\sqrt{d} + u)^2) \exp((3T + 24(\sqrt{d} + u)^2) K^2 T).$$

Taking square root of both sides and using $\sqrt{a+b} \le \sqrt{a} + \sqrt{b}$, we get

$$\sup_{\tau \le T} \|X(\tau)\| \le K(\sqrt{27} + \sqrt{12T}\sigma(\sqrt{d} + u)) \exp((3T + 24(\sqrt{d} + u)^2) K^2 T/2)\,,$$

which completes the proof of (28).

We next proceed with proving (29). Define $\tilde{\Delta}(t, \varepsilon) = \sup_{h \in [0, \varepsilon]} \|X(t\varepsilon + h) - X(t\varepsilon)\|$. Using (27), we have

$$\begin{aligned}
\tilde{\Delta}(t, \varepsilon) &\le \sup_{h \in [0, \varepsilon]} \left\{ \left\| \int_{t\varepsilon}^{t\varepsilon + h} \zeta(s) X(s) \mathrm{d}s \right\| + \|Y(t\varepsilon + h) - Y(t\varepsilon)\| + \left\| \int_{t\varepsilon}^{t\varepsilon + h} \frac{\zeta(s)}{\sqrt{\nu(s)}} D_s^{1/2} \mathrm{d}W(s) \right\| \right\} \\
&\le K\varepsilon \sup_{s \le T} \|X(s)\| + \sup_{h \in [0, \varepsilon], \tau \in [t\varepsilon, t\varepsilon + h]} \|Y'(\tau)\| h + \sup_{h \in [0, \varepsilon]} \|V(t, h, \varepsilon)\| \\
&\le K\varepsilon \sup_{s \le T} \|X(s)\| + \Gamma + \sup_{h \in [0, \varepsilon]} \|V(t, h, \varepsilon)\|\,,
\end{aligned} \tag{32}$$

with $V(t, h, \varepsilon) := \int_{t\varepsilon}^{t\varepsilon + h} \frac{\zeta(s)}{\sqrt{\nu(s)}} D_s^{1/2} \mathrm{d}W(s)$. In the last step, we used Assumption A2, by which $\|Y'(\tau)\| \le \Gamma/\varepsilon$. Similar to the derivation of (31), we have

$$\mathbb{P}\left(\sup_{t \in [0, T/\varepsilon] \cap \mathbb{Z}_{\ge 0}} \sup_{h \in [0, \varepsilon]} \|V(t, h, \varepsilon)\| \ge 2\sqrt{B_\varepsilon}(\sqrt{d} + u) \right) \le 2^{d/2} \exp(-(\sqrt{d} + u)^2) \le e^{-u^2}\,,$$

with $B_\varepsilon := \sup_{h \le \varepsilon} K^2 \int_{t\varepsilon}^{t\varepsilon + h} (2\|X(s)\|^2 + \sigma^2) \mathrm{d}s$. Plugging in (32), we have

$$\begin{aligned}
\tilde{\Delta}(t, \varepsilon) &\le K\varepsilon \sup_{s \le T} \|X(s)\| + 2K \sqrt{(2 \sup_{s \le T} \|X(s)\|^2 + \sigma^2)\varepsilon}\, (\sqrt{d} + u) \\
&= \Gamma + K \left(\varepsilon + 2\sqrt{2\varepsilon}(\sqrt{d} + u) \right) (\sup_{s \le T} \|X(s)\|) + 2K\sigma\sqrt{\varepsilon}(\sqrt{d} + u) \\
&\le C' \sqrt{T\varepsilon} (\sqrt{d} + u)^2 \exp\left[C \left(T^2 + (\sqrt{d} + u)^2 T \right) \right].
\end{aligned} \tag{33}$$

This concludes the proof of Equation (29).

We next rewrite the stochastic gradient descent update as follows:

$$\begin{aligned}\theta_{t+1} &= \theta_t - \eta_t \frac{1}{B_t}\sum_{k=1}^{B_t} \nabla \ell(\theta_t, z_{k,t}) \\ &= \theta_t + \eta_t \frac{1}{B_t}\sum_{k=1}^{B_t} (y_{tk} - \langle x_{tk}, \theta\rangle) x_{tk} \\ &= \theta_t + \eta_t(\theta_t^* - \theta_t) + \frac{\eta_t}{B_t}\sum_{k=1}^{B_t}\big((x_{tk}x_{tk}^\mathsf{T} - I)(\theta_t^* - \theta_t) + \varepsilon_{tk}x_{tk}\big) \\ &= \theta_t + \eta_t(\theta_t^* - \theta_t) - \eta_t \xi_t\,, \end{aligned} \tag{34}$$

where the noise term ξ_t has mean zero, given that the data points $z_{t,k}$ are sampled independently at each step t.

Note that ξ_t in (34) is the average of B_t zero mean variables and thus can be approximated by a normal distribution with covariance $(1/B_t)D_t$, with

$$\begin{aligned} D_t &= \big\{\mathbb{E}\left[(x_{tk}x_{tk}^\mathsf{T} - I)(\theta_t^* - \theta_t)(\theta_t^* - \theta_t)^\mathsf{T}(x_{tk}x_{tk}^\mathsf{T} - I)^\mathsf{T}\right] + \sigma^2 I\big\} \\ &= \big((\|\theta_t^* - \theta_t\|^2 + \sigma^2)I + (\theta_t^* - \theta_t)(\theta_t^* - \theta_t)^\mathsf{T}\big)\,, \end{aligned} \tag{35}$$

where the above identity follows from Lemma D.1. We let $\xi_t = -D_t^{1/2} g_t$ with $g_t \sim \mathsf{N}(0, I_d)$. Iterating update (34) recursively, we have

$$\begin{aligned}\theta_t - \theta_t^* &= \theta_0 - \theta_t^* + \sum_{\ell=0}^{t-1}\eta_\ell(\theta_\ell^* - \theta_\ell) + \sum_{\ell=0}^{t-1}\frac{\eta_\ell}{\sqrt{B_\ell}} D_\ell^{1/2} g_\ell \\ &= \theta_0 - \theta_t^* + \varepsilon\sum_{\ell=0}^{t-1}\zeta(\ell\varepsilon)(\theta_\ell^* - \theta_\ell) + \int_0^{t\varepsilon}\frac{s\zeta([s])}{\sqrt{s\nu([s])}} D_{[s]}^{1/2}\frac{\mathrm{d}W(s)}{\sqrt{s}} \\ &= \theta_0 - Y(\varepsilon t) + \varepsilon\sum_{\ell=0}^{t-1}\zeta(\ell\varepsilon)(\theta_\ell^* - \theta_\ell) + \int_0^{t\varepsilon}\frac{\zeta([s])}{\sqrt{\nu([s])}} D_{[s]}^{1/2}\mathrm{d}W(s)\,, \end{aligned} \tag{36}$$

where we adopt the notation $[s] = \varepsilon\lfloor s/\varepsilon\rfloor$, and $W(s)$ represents the standard Brownian motion.

We take the difference of (27) and (36). Since $\theta_0 = \theta_0 - \theta_0^* + \theta_0^* = X_0 + Y_0$, for $\tau \in \mathbb{Z}_{\geq 0}\varepsilon \cap [0, T]$, we have:

$$\begin{aligned}\|(\theta_{\tau/\varepsilon} - \theta^*) - X(\tau)\| \leq &\left\|\varepsilon\sum_{\ell=0}^{\tau/\varepsilon-1}\zeta(\ell\varepsilon)(\theta_\ell - \theta_\ell^*) - \int_0^\tau \zeta(s)X(s)\mathrm{d}s\right\| \\ &+ \left\|\int_0^\tau\Big(\frac{\zeta([s])}{\sqrt{\nu([s])}} - \frac{\zeta(s)}{\sqrt{\nu(s)}}\Big)\mathrm{d}W(s)\right\|. \end{aligned} \tag{37}$$

We first treat the first term. We have

$$\begin{aligned}&\varepsilon\sum_{\ell=0}^{t-1}\zeta(\ell\varepsilon)(\theta_\ell - \theta_\ell^*) - \int_0^\tau \zeta(s)X(s)\mathrm{d}s \\ &= \int_0^\tau \zeta([s])(\theta_{\lfloor s/\varepsilon\rfloor} - \theta_{\lfloor s/\varepsilon\rfloor}^*) - \int_0^\tau \zeta(s)X(s)\mathrm{d}s \\ &= \int_0^\tau \zeta([s])\Big(\theta_{\lfloor s/\varepsilon\rfloor} - \theta_{\lfloor s/\varepsilon\rfloor}^* - X([s])\Big)\mathrm{d}s + \int_0^\tau \zeta([s])(X([s]) - X(s))\mathrm{d}s + \int_0^\tau(\zeta([s]) - \zeta(s))X(s)\mathrm{d}s. \end{aligned}$$

We have

$$\left\|\int_0^\tau \zeta([s])(X([s]) - X(s))\mathrm{d}s\right\| \leq K\tau \sup_{t\in[0,T/\varepsilon]\cap\mathbb{Z}_{\geq 0}}\sup_{h\in[0,\varepsilon]}\|X(t\varepsilon + h) - X(t\varepsilon)\|\,. \tag{38}$$

Also,

$$\left\| \int_0^\tau (\zeta([s]) - \zeta(s)) X(s) \mathrm{d}s \right\| \le K\varepsilon\tau \sup_{\tau \in [0,T]} \|X(\tau)\| \,. \tag{39}$$

Note that the right-hand side of (38) and (39) are bounded in Lemma A.1.

We next bound the second term on the right-hand side of (37). Define

$$E(\tau) := \int_0^\tau \Big(\frac{\zeta([s])}{\sqrt{\nu([s])}} - \frac{\zeta(s)}{\sqrt{\nu(s)}} \Big) \mathrm{d}W(s).$$

Note that $E(\tau) \sim \mathsf{N}(0, \alpha^2 I_d)$, where

$$\alpha^2 = \int_0^\tau \Big(\frac{\zeta([s])}{\sqrt{\nu([s])}} - \frac{\zeta(s)}{\sqrt{\nu(s)}} \Big)^2 \mathrm{d}s \le K^2 \varepsilon\tau \,,$$

using Assumption A2 by which $\|\zeta/\sqrt{v}\|_{\mathrm{Lip}} \le K$. By applying Doob's inequality to the martingale $\exp(\frac{1}{2\tau}\|E(\tau)\|)$, similar to derivation of (31), we obtain

$$\mathbb{P}\left(\sup_{\tau \le T} \|E(\tau)\| \ge 2K\sqrt{\varepsilon T}(\sqrt{d} + u) \right) \le e^{-u^2/2} \,. \tag{40}$$

Now we define

$$\Delta(\tau) := \sup_{t \in [0, \tau/\varepsilon] \cap \mathbb{Z}_{\ge 0}} \|X(t\varepsilon) - (\theta_t - \theta_t^*)\| \,.$$

Using Lemma A.1 to bound (38) and (39) and then combining that with (40) into (37) we arrive at

$$\begin{aligned}
\Delta(\tau) &\le K \int_0^\tau \Delta(s) \mathrm{d}s + K\tau C' \sqrt{T\varepsilon} (\sqrt{d} + u)^2 \exp\Big[C \Big(T^2 + (\sqrt{d} + u)^2 T \Big) \Big] \\
&\quad + K\varepsilon\tau C \sqrt{T} (\sqrt{d} + u) \exp\Big[C \Big(T^2 + (\sqrt{d} + u)^2 T \Big) \Big] + 2K\sqrt{\varepsilon T}(\sqrt{d} + u) \\
&\le K \int_0^\tau \Delta(s) \mathrm{d}s + C'' T^{3/2} \sqrt{\varepsilon} (\sqrt{d} + u)^2 \exp\Big[C \Big(T^2 + (\sqrt{d} + u)^2 T \Big) \Big].
\end{aligned} \tag{41}$$

Using Gronwall's inequality we obtain

$$\Delta(T) \le C'' T^{3/2} \sqrt{\varepsilon} (\sqrt{d} + u)^2 \exp\Big[C \Big(T^2 + (\sqrt{d} + u)^2 T \Big) + KT \Big] \,. \tag{42}$$

We derive the final claim by noting that

$$\begin{aligned}
\sup_{t \in [0, \tau/\varepsilon] \cap \mathbb{Z}_{\ge 0}} \Big| \|X(t\varepsilon)\|^2 - \|\theta_t - \theta_t^*\|^2 \Big| &\le \Delta(T)^2 + 2\Delta(T) \sup_{t \in [0, \tau/\varepsilon] \cap \mathbb{Z}_{\ge 0}} \|X(t\varepsilon)\| \\
&\le C_1 \sqrt{\varepsilon} (\sqrt{d} + u)^4 T^3 \exp\Big[C_2 \Big(T^2 + (\sqrt{d} + u)^2 T \Big) \Big] \,,
\end{aligned} \tag{43}$$

for some constants C_1, C_2, depending on K, σ, Γ. This completes the proof. □

A.2. Proof of Theorem 3.2

Recall the SDE for process $X(\tau)$ given by

$$\mathrm{d}X(\tau) = -(\zeta(\tau)X(\tau) + Y'(\tau))\mathrm{d}\tau + \frac{\zeta(\tau)}{\sqrt{\nu(\tau)}} \left((\|X(\tau)\|^2 + \sigma^2) I + X(\tau) X(\tau)^\mathsf{T} \right)^{1/2} \mathrm{d}W(\tau) \,,$$

Let $m_\tau := \mathbb{E}[X(\tau)]$. Taking expectation of the above SDE, we obtain

$$m_\tau' = -\zeta(\tau) m_\tau - Y'(\tau) \,.$$

Next we define the stochastic process $Z(\tau) = \|X(\tau)\|^2$. By Ito's lemma (cf. Lemma D.2), we have

$$\begin{aligned}\mathrm{d}Z(\tau) = &\left(-2\zeta(\tau)\|X(\tau)\|^2 - 2X(\tau)^\mathsf{T}Y'(\tau) + \frac{\zeta(\tau)^2}{\nu(\tau)}\left((d+1)\|X(\tau)\|^2 + d\sigma^2\right)\right)\mathrm{d}\tau \\ &+ 2\frac{\zeta(\tau)}{\sqrt{\nu(\tau)}}X(\tau)^\mathsf{T}\left((\|X(\tau)\|^2+\sigma^2)I + X(\tau)X(\tau)^\mathsf{T}\right)^{1/2}\,\mathrm{d}W(\tau)\,.\end{aligned}$$

Taking expectation of both sides, we arrive at the following ODE for $v_\tau = \mathbb{E}[Z(\tau)] = \mathbb{E}[\|X(\tau)\|^2]$:

$$\begin{aligned}v_\tau' &= -2\zeta(\tau)v_\tau - 2m_\tau^\mathsf{T}Y'(\tau) + \frac{\zeta(\tau)^2}{\nu(\tau)}((d+1)v_\tau + d\sigma^2) \\ &= \left((d+1)\frac{\zeta(\tau)^2}{\nu(\tau)} - 2\zeta(\tau)\right)v_\tau + \frac{\zeta(\tau)^2}{\nu(\tau)}\sigma^2 d - 2m_\tau^\mathsf{T}Y'(\tau)\,. \end{aligned} \tag{44}$$

A.3. Proof of Theorem 3.3

We start by giving a brief overview of the Hamilton–Jacobi–Bellman (HJB) equation (Bellman, 1956).

Consider the following value function:

$$V(z(\tau_0),\tau_0) = \min_{\zeta:[\tau_0,T]\to\mathcal{A}} \int_{\tau_0}^{T} C(z(\tau),\zeta(\tau))\mathrm{d}\tau + D(z(T))\,, \tag{45}$$

where $z(\tau)$ is the vector of the system state, $\zeta(\tau)$, for $\tau \in [\tau_0, T]$ is the control policy we aim to optimize over and takes value in a set $\mathcal{A}$, $C(\cdot)$ is the scalar cost function and $D(\cdot)$ gives the bequest value at the final state $z(T)$.

Suppose that the system is also subject to the constraint

$$\frac{\mathrm{d}}{\mathrm{d}\tau}z(\tau) = \Phi(z(\tau),\zeta(\tau))\,, \quad \forall \tau \in [\tau_0, T]\,, \tag{46}$$

with Φ describing the evolution of the system state over time. The dynamic programming principle allows us to derive a recursion on the value function V, in the form of a partial differential equation (PDE). Namely, the the Hamilton–Jacobi–Bellman PDE is given by

$$\begin{aligned}\partial_\tau V(z,\tau) + \min_{\zeta\in\mathcal{A}}\left[\partial_z V(z,\tau)\cdot\Phi(z,\zeta) + C(z,\zeta)\right] = 0\,, \\ \text{subject to} \quad V(z,T) = D(z)\,.\end{aligned} \tag{47}$$

The above PDE can be solved backward in time and then the optimal control $\zeta^*(\tau)$ is given by

$$\zeta^*(\tau) = \arg\min_{\zeta\in\mathcal{A}}\left[\partial_z V(z(\tau),\tau)\cdot\Phi(z(\tau),\zeta) + C(z(\tau),\zeta)\right]\,. \tag{48}$$

We are now ready to prove the claim of Theorem 3.3, using the HJB equation.

Consider $\tilde{v}_\tau$ as the system state at time τ (i.e., $z(\tau) = \tilde{v}_\tau$), and the cost function $C(\tilde{v}_\tau, \zeta(\tau)) = \tilde{v}_\tau$. Also set $D(\cdot)$ to be the zero everywhere. The control variable $\zeta(\tau)$ takes values in $\mathcal{A} = [0, 1]$.

The function $\Phi(\cdot,\cdot)$ in (46) is given by (12), which we recall here:

$$\Phi(\tilde{v}_\tau,\zeta) := \left((d+1)\frac{\zeta^2}{\nu(\tau)} - 2\zeta\right)\tilde{v}_\tau + \frac{\zeta^2}{\nu(\tau)}\sigma^2 d + 2\|Y'(\tau)\|\sqrt{\tilde{v}_\tau}\,.$$

Note that in our case, the cost function C does not depend on $\zeta(\tau)$. Also, it is easy to see that $\partial_z V(\tilde{v}_\tau,\tau) > 0$ because larger $\tilde{v}_\tau$ means we are further from the sequence of models and so the minimum cost achievable in tracking the sequence of models will be higher. Therefore, (48) reduces to

$$\zeta^*(\tau) = \arg\min_{\zeta\in[0,1]} \Phi(\tilde{v}_\tau,\zeta).$$

Since Φ is quadratic in ζ, solution to the above optimization has a closed form given by

$$\zeta^*(\tau) = \min\left\{1, \left(\frac{d+1}{\nu(\tau)}\tilde{v}_\tau + \frac{\sigma^2 d}{\nu(\tau)}\right)^{-1}\tilde{v}_\tau\right\},$$

which completes the proof.

A.4. Proof of Lemma 3.6

Substituting for $\zeta(\tau)$ from (13), it is easy to verify that $\tilde{v}'(\tau) \le 0$ and so $\tilde{v}(\tau)$ is decreasing in τ.

Define the shorthand $\mathsf{a} := (d+1)/\nu(\tau)$ and $\mathsf{b} := \sigma^2 d/\nu(\tau)$. Note that if $\tilde{v}_\tau \ge \mathsf{b}/(1-\mathsf{a})$, then by (13), $\zeta(\tau) = 1$ and in this case ODE (12) reduces to $\tilde{v}'_\tau = (\mathsf{a}-2)\tilde{v}_\tau + \mathsf{b}$, with the solution

$$\tilde{v}_\tau = \left(\tilde{v}_0 + \frac{\mathsf{b}}{\mathsf{a}-2}\right)e^{(\mathsf{a}-2)\tau} - \frac{\mathsf{b}}{\mathsf{a}-2}.$$

However, the above solution is valid until $\tilde{v}_\tau \ge \mathsf{b}/(1-\mathsf{a})$, which is the assumption we started with, which using the above characterization is equivalent to

$$\tau \le \tau_* := \left[\frac{1}{2-\mathsf{a}}\log\left((1-\mathsf{a})\left(\tilde{v}_0\frac{2-\mathsf{a}}{\mathsf{b}} - 1\right)\right)\right]_+.$$

For $\tau > \tau_*$, we have $\tilde{v}_\tau \le \mathsf{b}/(1-\mathsf{a})$ and so $\zeta(\tau) = \tilde{v}_\tau/(\mathsf{a}\tilde{v}_\tau + \mathsf{b})$ by (13). In this case, ODE (12) reduces to

$$\tilde{v}'_\tau = -\frac{\tilde{v}_\tau^2}{\mathsf{a}\tilde{v}_\tau + \mathsf{b}}.$$

By rearranging the terms and integrating, the solution to above ODE satisfies

$$\mathsf{a}\ln\left(\frac{1}{\tilde{v}_\tau}\right) + \frac{\mathsf{b}}{\tilde{v}_\tau} = \tau + C, \tag{49}$$

where C can be obtained by the continuity condition of $\tilde{v}_\tau$ at τ_*, i.e.,

$$C = \mathsf{a}\ln\left(\frac{1-\mathsf{a}}{\mathsf{b}}\right) + 1 - \mathsf{a} - \tau_*.$$

From (49) we observe that as $\tau \to \infty$, $\tilde{v}_\tau \to 0$ and the term $\mathsf{b}/\tilde{v}_\tau$ becomes dominant by which we obtain

$$\lim_{\tau\to\infty}\frac{\tilde{v}_\tau}{\frac{\mathsf{b}}{\tau+C}} = 1.$$

In addition, invoking definition of optimal policy $\zeta(\tau)$, we obtain

$$\lim_{\tau\to\infty}\frac{\zeta(\tau)}{\frac{1}{\mathsf{a}+C+\tau}} = 1,$$

which completes the proof.

B. Proof of theorems and technical lemmas for convex loss

B.1. Proof of Theorem 4.2

We define the shorthand $D_t^2 = \|\theta_t^* - \theta_t\|^2$ and let $v_t = \theta_t^* - \theta_{t+1}^*$ be shifts in the optimal models. We also define the shorthand

$$\nabla\ell_t^B(\theta_t) := \frac{1}{B_t}\sum_{k=1}^{B_t}\nabla\ell(\theta_t, z_{t,k}).$$

Since projection on a convex set is contraction, we have

$$\|\Pi_\Theta(u) - w\| \leq \|u - w\|\,,$$

for any $w \in \Theta$. Using this property, we have

$$\begin{aligned}
D_{t+1}^2 &= \|\Pi_\Theta(\theta_t - \eta_t \nabla \ell_t^B(\theta_t)) - \theta_{t+1}^*\|^2 \\
&= \|\Pi_\Theta(\theta_t - \eta_t \nabla \ell_t^B(\theta_t)) - \theta_t^* + \theta_t^* - \theta_{t+1}^*\|^2 \\
&= \|\Pi_\Theta(\theta_t - \eta_t \nabla \ell_t^B(\theta_t)) - \theta_t^*\|^2 + \|v_t\|^2 + 2\langle v_t, \Pi_\Theta(\theta_t - \eta_t \nabla \ell_t^B(\theta_t)) - \theta_t^*\rangle \\
&\leq \|\theta_t - \eta_t \nabla \ell_t^B(\theta_t) - \theta_t^*\|^2 + \|v_t\|^2 + 2\langle v_t, \Pi_\Theta(\theta_t - \eta_t \nabla \ell_t^B(\theta_t)) - \theta_t^*\rangle \\
&= D_t^2 - 2\eta_t \langle \nabla \ell_t^B(\theta_t), \theta_t - \theta_t^*\rangle + \|v_t\|^2 + 2\langle v_t, \Pi_\Theta(\theta_t - \eta_t \nabla \ell_t^B(\theta_t)) - \theta_t^*\rangle + \eta_t^2 \|\nabla \ell_t^B(\theta_t)\|^2.
\end{aligned}$$

Define

$$\delta_t := \nabla \ell_t^B(\theta_t) - \nabla \bar{\ell}_t(\theta_t)\,,$$

as the difference between the gradient of the expected loss (at step t) and the gradient of the batch average loss at that step. Writing the above bound in terms of this notation, we get

$$\begin{aligned}
D_{t+1}^2 \leq D_t^2 &- 2\eta_t \langle \nabla \bar{\ell}_t(\theta_t) + \delta_t, \theta_t - \theta_t^*\rangle + \|v_t\|^2 + 2\langle v_t, \Pi_\Theta(\theta_t - \eta_t \nabla \ell_t^B(\theta_t)) - \theta_t^*\rangle \\
&+ \eta_t^2 \Big(\|\nabla \bar{\ell}_t(\theta_t)\|^2 + \|\delta_t\|^2 + 2\langle \delta_t, \nabla \bar{\ell}_t(\theta_t)\rangle \Big)\,.
\end{aligned} \tag{50}$$

By Zhou (2018, Lemma 4) for any L-smooth convex function f, we have

$$\frac{1}{L}\|\nabla f(y) - \nabla f(x)\|^2 \leq \langle \nabla f(y) - \nabla f(x), y - x\rangle\,. \tag{51}$$

Since the loss function $\ell(\theta, z)$ is convex, the expected loss functions $\bar{\ell}_t(\theta)$ are also convex for $t = 1, \ldots, T$. Using (51) together with the fact that $\nabla \bar{\ell}_t(\theta_t^*) = 0$ by optimality of θ_t^*, we get

$$\frac{1}{L}\|\nabla \bar{\ell}_t(\theta_t)\|^2 \leq \langle \nabla \bar{\ell}_t(\theta_t), \theta_t - \theta_t^*\rangle\,. \tag{52}$$

Using the above bound, we obtain

$$\begin{aligned}
D_{t+1}^2 \leq D_t^2 &- (2\eta_t - L\eta_t^2)\langle \nabla \bar{\ell}_t(\theta_t), \theta_t - \theta_t^*\rangle + \|v_t\|^2 + 2\langle v_t, \Pi_\Theta(\theta_t - \eta_t \nabla \ell_t^B(\theta_t)) - \theta_t^*\rangle \\
&+ \eta_t^2 \|\delta_t\|^2 - 2\eta_t \langle \delta_t, \theta_t - \theta_t^* - \eta_t \nabla \bar{\ell}_t(\theta_t)\rangle.
\end{aligned}$$

Recall our assumption $\eta_t \leq 2/L$. Using the convexity of $\bar{\ell}_k$, we have

$$\bar{\ell}_t(\theta_t) - \bar{\ell}_t(\theta_t^*) \leq \langle \nabla \bar{\ell}_t(\theta_t), \theta_t - \theta_t^*\rangle, \tag{53}$$

which along with the above bound implies that

$$\begin{aligned}
D_{t+1}^2 \leq D_t^2 &- (2\eta_t - L\eta_t^2)(\bar{\ell}_t(\theta_t) - \bar{\ell}_t(\theta_t^*)) + \|v_t\|^2 + 2\langle v_t, \Pi_\Theta(\theta_t - \eta_t \nabla \ell_t^B(\theta_t)) - \theta_t^*\rangle \\
&+ \eta_t^2 \|\delta_t\|^2 - 2\eta_t \langle \delta_t, \theta_t - \theta_t^* - \eta_t \nabla \bar{\ell}_t(\theta_t)\rangle\,.
\end{aligned}$$

Note that $\Pi_\Theta(\theta_t - \eta_t \nabla \ell_t^B(\theta_t)) - \theta_t^* = \theta_{t+1} - \theta_t^*$. We let $a_t := 2\eta_t - L\eta_t^2 > 0$, and by rearranging the terms in the above equation we obtain

$$\bar{\ell}_t(\theta_t) - \bar{\ell}_t(\theta_t^*) \leq \frac{D_t^2}{a_t} - \frac{D_{t+1}^2}{a_t} + \frac{\|v_t\|^2}{a_t} + \frac{2}{a_t}\langle v_t, \theta_{t+1} - \theta_t^*\rangle + \frac{\eta_t^2 \|\delta_t\|^2}{a_t} - \frac{2\eta_t}{a_t}\langle \delta_t, \theta_t - \theta_t^* - \eta_t \nabla \bar{\ell}_t(\theta_t)\rangle\,. \tag{54}$$

We next note that $\theta_t, \theta_t^*, \eta_t$ are adapted to the filtration $\boldsymbol{z}_{[t-1]}$, and therefore,

$$\mathbb{E}[\langle \delta_t, \theta_t - \theta_t^* - \eta_t \nabla \bar{\ell}_t(\theta_t)\rangle | \boldsymbol{z}_{[t-1]}] = \langle \mathbb{E}[\delta_t | \boldsymbol{z}_{[t-1]}], \theta_t - \theta_t^* - \eta_t \nabla \bar{\ell}_t(\theta_t)\rangle = 0\,.$$

Taking iterated expectations of both sides of (54) with respect to filtration $\boldsymbol{z}_t$ (first conditional on $\boldsymbol{z}_{[t-1]}$ and then with respect to $\boldsymbol{z}_{[t-1]}$), we get

$$\mathbb{E}[\mathsf{reg}_t] \le \mathbb{E}\left[\frac{D_t^2 - D_{t+1}^2}{a_t} + \frac{\sigma^2}{B_t}\frac{\eta_t^2}{a_t} + \frac{\|v_t\|^2}{a_t} + \frac{2}{a_t}\langle v_t, \theta_{t+1} - \theta_t^*\rangle\right]\,, \tag{55}$$

with $\mathsf{reg}_t = \bar{\ell}_t(\theta_t) - \bar{\ell}_t(\theta_t^*)$. Summing both sides over $t = 1, \dots, T$, we obtain the desired result.

B.2. Proof of Proposition 4.3

Recall the optimization problem for η^* given below:

$$\eta_t^* := \arg\min_{0 \le \eta \le \frac{1}{L}} D_{\max}^2 \left(\frac{1}{2\eta - L\eta^2} - \frac{1}{2\eta_{t-1} - L\eta_{t-1}^2}\right)_+ + \frac{\sigma^2}{B_t} \cdot \frac{\eta^2}{2\eta - L\eta^2} + \frac{\gamma_t^2 + 2D_{\max}\gamma_t}{2\eta - L\eta^2}\,. \tag{56}$$

Note that the functions $1/(2\eta - L\eta^2)$ and $\eta^2/(2\eta - L\eta^2)$ are convex for $\eta \le 1/L$. Also the pointwise maximum of convex functions is convex, which implies that the objective function above is convex. With that, we first derive the stationary points of the objective function and then compare them to the boundary points 0 and $1/L$.

Setting the subgradient of the objective to zero we arrive at the following equation:

$$\frac{2\sigma^2}{B_t} \cdot \frac{1}{(2 - L\eta)^2} + 2\Big(\gamma_t^2 + 2D_{\max}\gamma_t + D_{\max}^2 \mathbb{I}(\eta < \eta_{t-1})\Big)\frac{L\eta - 1}{(2\eta - L\eta^2)^2} = 0\,. \tag{57}$$

We consider the two cases below:

- $\eta \ge \eta_{t-1}$: In this case, (57) reduces to

  $$\frac{\sigma^2}{B_t} + \Big(\gamma_t^2 + 2D_{\max}\gamma_t\Big)\frac{L\eta - 1}{\eta^2} = 0\,,$$

  which is a quadratic equation in η. Solving for η, the positive solution is given by τ_1 (18). This case happens only when the solution satisfies the condition of the case, namely $\eta_{t-1} \le \tau_{1,t}$.

- $\eta \le \eta_{t-1}$. In this case, (57) reduces to

  $$\frac{\sigma^2}{B_t} + \Big(\gamma_t^2 + 2D_{\max}\gamma_t + D_{\max}^2\Big)\frac{L\eta - 1}{\eta^2} = 0\,,$$

  which admits the positive solution $\tau_{2,t}$ (19). This case happens only when the solution satisfies the condition of the case, namely $\tau_{2,t} \le \eta_{t-1}$.

If $\tau_{1,t} < \eta_{t-1} < \tau_{2,t}$, then in both of the above cases, the solution happens at the boundary value η_{t-1}. This brings us to the following characterization for η_t^*:

$$\eta_t^* = \begin{cases} \tau_{1,t} & \text{if } \eta_{t-1}^* \le \tau_{1,t}, \\ \eta_{t-1}^* & \text{if } \tau_{1,t} \le \eta_{t-1}^* \le \tau_{2,t} \\ \tau_{2,t} & \text{if } \eta_{t-1}^* \ge \tau_{2,t}\,. \end{cases} \tag{58}$$

Note that the above characterization was based on the stationary points of the objective. we next examine if the above solution satisfies the boundary conditions. Obviously $\eta_t^* > 0$. We also claim that $\eta_t^* \le 1/L$. For this, we only need to show that $\tau_{2,t} \le 1/L$ (because $\eta_t^* \le \tau_{2,t}$ for all values of η_{t-1}). Invoking definition of $\tau_{2,t}$, we have

$$\tau_{2,t} := \frac{B_t}{2\sigma^2}\left(\sqrt{b_{2,t}^2 L^2 + \frac{4\sigma^2}{B_t} b_{2,t}} - b_{2,t} L\right), \quad b_{2,t} := (\gamma_t + D_{\max})^2\,.$$

It is easy to see that $\tau_{2,t} \le 1/L$ follows simply from $b_{2,t}^2 L^2 + 4\frac{\sigma^2}{B_t} b_{2,t} < (\frac{2\sigma^2}{LB_t} + b_{2,t} L)^2$.

B.3. Proof of Theorem 4.7

Recall that

$$\delta_t := \nabla \ell_t^B(\theta_t) - \nabla \bar{\ell}_t(\theta_t)\,,$$

as the difference between the gradient of the expected loss (at step t) and the gradient of the batch average loss at that step. Writing D_{t+1} in terms of this notation, we get

$$\begin{aligned} D_{t+1}^2 = D_t^2 &- 2\eta_t\langle \nabla\bar{\ell}_t(\theta_t) + \delta_t, \theta_t - \theta_t^*\rangle + \|v_t\|^2 + 2\langle v_t, (\theta_t - \eta_t\nabla\ell_t^B(\theta_t)) - \theta_t^*\rangle \\ &+ \eta_t^2\Big(\|\nabla\bar{\ell}_t(\theta_t)\|^2 + \|\delta_t\|^2 + 2\langle\delta_t, \nabla\bar{\ell}_t(\theta_t)\rangle\Big)\,. \end{aligned} \tag{59}$$

Since the loss function $\ell(\theta, z)$ is L-smooth and μ-strongly convex, the expected loss $\bar{\ell}_t(\theta)$ is also L-smooth and μ-strongly convex and by invoking Zhou (2018, Lemma 3(iii)), we have

$$\langle \nabla\bar{\ell}_t(\theta_t), \theta_t - \theta_t^*\rangle \le \bar{\ell}_t(\theta_t) - \bar{\ell}_t(\theta_t^*) + \frac{1}{2\mu}\|\nabla\bar{\ell}_t(\theta_t)\|^2\,.$$

Using this bound in (59), we obtain

$$\begin{aligned} D_{t+1}^2 \ge D_t^2 &- 2\eta_t(\bar{\ell}_t(\theta_t) - \bar{\ell}_t(\theta_t^*)) + \|v_t\|^2 + 2\langle v_t, (\theta_t - \eta_t\nabla\ell_t^B(\theta_t)) - \theta_t^*\rangle \\ &+ \Big(\eta_t^2 - \frac{\eta_t}{\mu}\Big)\|\nabla\bar{\ell}_t(\theta_t)\|^2 + \eta_t^2\|\delta_t\|^2 - 2\eta_t\langle\delta_t, \theta_t - \theta_t^* - \eta_t\nabla\bar{\ell}_t(\theta_t)\rangle. \end{aligned} \tag{60}$$

We next use Zhou (2018, Lemma 4, item 5) and the fact that $\nabla\bar{\ell}_t(\theta_t^*) = 0$ to get

$$\|\nabla\bar{\ell}_t(\theta_t)\|^2 \le 2L(\bar{\ell}_t(\theta_t) - \bar{\ell}_t(\theta_t^*))\,. \tag{61}$$

Using the above bound into (62), for $\eta_t \le 1/\mu$, we obtain

$$\begin{aligned} D_{t+1}^2 \ge D_t^2 &- 2\Big(\eta_t + \frac{L}{\mu}\eta_t - \eta_t^2 L\Big)(\bar{\ell}_t(\theta_t) - \bar{\ell}_t(\theta_t^*)) + \|v_t\|^2 + 2\langle v_t, (\theta_t - \eta_t\nabla\ell_t^B(\theta_t)) - \theta_t^*\rangle \\ &+ \eta_t^2\|\delta_t\|^2 - 2\eta_t\langle\delta_t, \theta_t - \theta_t^* - \eta_t\nabla\bar{\ell}_t(\theta_t)\rangle\,. \end{aligned} \tag{62}$$

We recognize that $\theta_t - \eta_t\nabla\ell_t^B(\theta_t) = \theta_{t+1}$ by the SGD update, and let $a_t' := 2(\eta_t + \frac{L}{\mu}\eta_t - \eta_t^2 L)$, with $\eta_t \le 1/\mu$.

Next we obtain a telescoping series for $\mathsf{Reg}(T)$ as before. Continuing as before (in Theorem 4.2), we can (1) isolate $\bar{\ell}_t(\theta_t) - \bar{\ell}_t(\theta_t^*)$ on the left-hand side, and (2) take expectations: first conditioned on the filtration $\boldsymbol{z}_{[t-1]}$ and then an unconditioned expectation, to get:

$$\mathbb{E}[\mathsf{Reg}(T)] = \sum_{t=1}^T \mathbb{E}[\mathsf{reg}_t] \ge \mathbb{E}\left[\sum_{t=1}^T\left(\frac{D_t^2}{a_t'} - \frac{D_{t+1}^2}{a_{t+1}'}\right) + \frac{\sigma^2\eta_t^2}{B_t a_t'} + \frac{\|v_t\|^2}{a_t'} + 2\frac{\langle v_t, \theta_{t+1} - \theta_t^*\rangle}{a_t'}\right],$$

which completes the proof of theorem.

C. Proof of theorems and technical lemmas for non-convex loss

C.1. Proof of Theorem 5.1

We note that by Assumption 4.1,

$$\Big|\bar{\ell}_t(\theta_{t+1}) - \bar{\ell}_t(\theta_t) - \langle\nabla\ell_t(\theta_t), \theta_{t+1} - \theta_t\rangle\Big| \le \frac{L}{2}\|\theta_{t+1} - \theta_t\|^2 = \frac{L}{2}\eta_t^2\|\nabla\ell_t^B(\theta_t)\|^2\,. \tag{63}$$

Therefore,

$$\begin{aligned} \bar{\ell}_t(\theta_{t+1}) - \bar{\ell}_t(\theta_t) &\le \langle\nabla\bar{\ell}_t(\theta_t), \theta_{t+1} - \theta_t\rangle + \frac{L}{2}\eta_t^2\|\nabla\ell_t^B(\theta_t)\|^2 \\ &\le -\eta_t\langle\nabla\bar{\ell}_t(\theta_t), \nabla\ell_t^B(\theta_t)\rangle + \frac{L}{2}\eta_t^2\|\nabla\ell_t^B(\theta_t)\|^2. \end{aligned}$$

Recall the notation $\delta_t := \nabla \ell_t^B(\theta_t) - \nabla \bar{\ell}_t(\theta_t)$, by which we get

$$\begin{aligned}\bar{\ell}_t(\theta_{t+1}) - \bar{\ell}_t(\theta_t) &\le -\eta_t \|\nabla \bar{\ell}_t(\theta_t)\|^2 - \eta_t \langle \nabla \bar{\ell}_t(\theta_t), \delta_t \rangle + \frac{L}{2}\eta_t^2 \left(\|\nabla \bar{\ell}_t(\theta_t)\|^2 + 2\langle \nabla \bar{\ell}_t(\theta_t), \delta_t \rangle + \|\delta_t\|^2 \right) \\ &= -\left(\eta_t - \frac{L}{2}\eta_t^2 \right) \|\nabla \bar{\ell}_t(\theta_t)\|^2 - (\eta_t - L\eta_t^2) \langle \nabla \bar{\ell}_t(\theta_t), \delta_t \rangle + \frac{L}{2}\eta_t^2 \|\delta_t\|^2 .\end{aligned}$$

By condition $\eta_t \le 1/L$, we have $a_t = \eta_t - L\eta_t^2 > 0$. Rearranging the terms in the above inequality, we obtain

$$\|\nabla \bar{\ell}_t(\theta_t)\|^2 \le 2\frac{\bar{\ell}_t(\theta_t) - \bar{\ell}_t(\theta_{t+1})}{a_t} - 2\langle \nabla \bar{\ell}_t(\theta_t), \delta_t \rangle + \frac{L\eta_t^2}{a_t}\|\delta_t\|^2 . \tag{64}$$

Since $\theta_t, \theta_t^*, \eta_t$ are adapted to the filtration $\boldsymbol{z}_{[t-1]}$, we have

$$\mathbb{E}[\langle \nabla \bar{\ell}_t(\theta_t), \delta_t \rangle | \boldsymbol{z}_{[t-1]}] = \langle \nabla \bar{\ell}_t(\theta_t), \mathbb{E}[\delta_t] | \boldsymbol{z}_{[t-1]]} = 0 .$$

Therefore, by taking expectation from the both sides of (65), first conditional on $\boldsymbol{z}_{[t-1]}$ and then with respect to $\boldsymbol{z}_{[t-1]}$ we get

$$\begin{aligned}\mathbb{E}[\|\nabla \bar{\ell}_t(\theta_t)\|^2] &\le 2\frac{\bar{\ell}_t(\theta_t) - \bar{\ell}_t(\theta_{t+1})}{a_t} + \frac{L\eta_t^2}{a_t}\frac{\sigma^2}{B_t} \\ &\le 2\frac{\bar{\ell}_t(\theta_t) - \bar{\ell}_{t+1}(\theta_{t+1})}{a_t} + \frac{|\bar{\ell}_{t+1}(\theta_{t+1}) - \bar{\ell}_t(\theta_{t+1})|}{a_t} + \frac{L\eta_t^2}{a_t}\frac{\sigma^2}{B_t} \\ &= 2\frac{\bar{\ell}_t(\theta_t) - \bar{\ell}_{t+1}(\theta_{t+1})}{a_t} + \frac{\gamma_t}{a_t} + \frac{L\eta_t^2}{a_t}\frac{\sigma^2}{B_t} .\end{aligned} \tag{65}$$

Summing both sides over $t = 1, \dots, T$, we have

$$\begin{aligned}\mathbb{E}[\mathsf{Reg}(T)] &= \sum_{t=1}^{T} \mathbb{E}[\|\nabla \bar{\ell}_t(\theta_t)\|^2] \\ &\le \sum_{t=1}^{T} \mathbb{E}\left[\left(\frac{2\bar{\ell}_t(\theta_t)}{a_t} - \frac{2\bar{\ell}_{t+1}(\theta_{t+1})}{a_t} \right) + L\frac{\sigma^2\eta_t^2}{B_t a_t} + \frac{\gamma_t}{a_t} \right] \\ &= \sum_{t=2}^{T} \mathbb{E}\left[2\bar{\ell}_t(\theta_t) \left(\frac{1}{a_t} - \frac{1}{a_{t-1}} \right) \right] + \mathbb{E}\left[\frac{2\bar{\ell}_1(\theta_1)}{a_1} - \frac{2\bar{\ell}_{T+1}(\theta_T)}{a_{T+1}} \right] + \sum_{t=1}^{T} \mathbb{E}\left[L\frac{\sigma^2\eta_t^2}{B_t a_t} + \frac{\gamma_t}{a_t} \right] .\end{aligned}$$

The result follows by noting that $\bar{\ell}_{T+1}(\theta_{T+1}) \ge 0$.

D. Auxiliary lemmas

Lemma D.1. *Let $x \in \mathbb{R}^d$ such that $x \sim \mathsf{N}(0, I_d)$. For any fixed vector $u \in \mathbb{R}^d$, we have*

$$\mathbb{E}[(xx^\mathsf{T} - I)uu^\mathsf{T}(xx^\mathsf{T} - I)^\mathsf{T}] = \|u\|^2 I + uu^\mathsf{T} .$$

Proof. By Stein's lemma, for any function $g : \mathbb{R}^d \to \mathbb{R}$ we have

$$\mathbb{E}[(xx^\mathsf{T} - I)g(x)] = \mathbb{E}[\nabla^2 g(x)] .$$

Using the above identity with $g(x) = (u^\mathsf{T} x)^2$ we obtain

$$\mathbb{E}[xx^\mathsf{T}(u^\mathsf{T} x)^2] = 2uu^\mathsf{T} + \|u\|^2 I . \tag{66}$$

Using the above characterization, we get

$$\begin{aligned}\mathbb{E}[(xx^\mathsf{T} - I)uu^\mathsf{T}(xx^\mathsf{T} - I)^\mathsf{T}] &= \mathbb{E}[xx^\mathsf{T}(u^\mathsf{T} x)^2] - u(u^\mathsf{T} x)x^\mathsf{T} - x(x^\mathsf{T} u)u^\mathsf{T} + uu^\mathsf{T} \\ &= 2uu^\mathsf{T} + \|u\|^2 I - 2uu^\mathsf{T} + uu^\mathsf{T} \\ &= uu^\mathsf{T} + \|u\|^2 I ,\end{aligned}$$

which completes the proof. □

We next present Ito's lemma, which allows to find the differential of a time-dependent function of a stochastic process.

Lemma D.2 (Itô's lemma, (Oksendal, 2013))**.** *Let $X_t \in \mathbb{R}^p$ be a vector of Itô drift-diffusion process, such that*

$$\mathrm{d}X_t = f(t, X_t)\mathrm{d}t + g(t, X_t)\mathrm{d}W_t\,,$$

with W_t being an q-dimensional standard Brownian motion and $f(t, X_t) \in \mathbb{R}^p$ and $g(t, X_t) \in \mathbb{R}^{p\times q}$. Consider a scalar process $Y(t)$ defined by $Y(t) = \phi(t, X(t))$, where $\phi(t, X)$ is a scalar function which is continuously differentiable with respect to t and twice continuously differentiable with respect to X. We then have

$$\begin{aligned}
\mathrm{d}Y_t &= \tilde{f}(t, X_t)\mathrm{d}t + \tilde{g}(t, X_t)\mathrm{d}W_t\,,\\
\tilde{f}(t, X_t) &= \phi_t(t, X_t) + \phi_x(t, X_t)^\mathsf{T} f(t, X_t) + \frac{1}{2}\mathrm{tr}\left(g(t, X_t)^\mathsf{T}\phi_{xx}(t, X_t)g(t, X_t)\right)\\
\tilde{g}(t, X_t) &= \phi_x(t, X_t)^\mathsf{T} g(t, X_t)\,.
\end{aligned}$$

E. Experiments: Flow cytometry

In this section, we explore a medical application called *flow cyotometry*, which uses neural networks and online stochastic optimization to classify cells as they arrive in a stream from a shifting data distribution. The features this model receives as input are measurements based on the RNA expressions of each cell (see, e.g., Li et al. (2019); Hu et al. (2020; 2022) and the references therein for details). This induces a learning problem with a non-convex loss landscape that changes with time, where we do not have a tight characterization for an optimal learning rate schedule.

E.1. Background

We start with background on flow cytometry to give more context for this application. A sample of cells from a tissue is prepared and a small number of selected RNA sequences in the cells are bound to different fluorescent markers. A laser then illuminates the incoming stream of cells, which can now be separated based on the intensity of the signals from different fluorescent markers. Using fluorescent markers, however, comes at a cost as they can interfere with normal cellular functioning. In contrast, marker-free systems that use large convolutional neural nets are often more accurate, but can be slower to adapt to distribution shifts. See Li et al. (2019) for further details.

We study a two-step system that does initial classification with an inexpensive "student" neural network and only relies on a small number of fluorescent markers. This is followed by additional analysis using a large pretrained convolutional neural network (CNN) with near real-time feedback. As a simplification, we assume the expensive CNN is a "teacher" model whose predictions are ground truth labels. We can achieve real-time feedback for the initial classifier that first sees the cells by replicating the teacher across servers to increase its inference throughput. The goal is to optimize the (inexpensive) classifier online and minimize its loss, i.e., the number of misclassified cells.

The distribution of the arriving cells can change based on the sample preparation and tissue characteristics. For example, for pancreatic tissue, if we stream the cells starting from anterior to posterior, the initial mixture of cells consists of more non-secreting cells but later will have a higher proportion of secreting cells. Thus, as a simplification, it is worth exploring the effect of different learning rate schedules for a simple online neural network that classifies the input stream of cells into different cell types based on a small number of RNA expression markers in each cell. We use the pancreatic RNA expression data in (Bastidas-Ponce et al., 2019; Bergen et al., 2020).[3]

Specifically, we use the expression levels of ten RNA molecules (corresponding to genes Pyy, Meg3, Malat1, Gcg, Gnas, Actb, Ghrl, Rsp3, Ins2 and Hspa8) for the 4000 murine pancreatic cells in the `scVelo` repository. The expression levels of these genes determines the cell types completely. We slightly perturb the expression levels to generate a stream of cells, and within this stream we vary the distribution of secreting cells (i.e., alpha, beta, and delta) and non-secreting cells (i.e., ductal), starting from non-secreting cells dominating the distribution and ending with secreting cells dominating the distribution. Figure 6 (left) is a two-dimensional embedding of these ten signals labeled by their cell-type. In practice, any stream of cells undergoes a similar distribution shift depending on how the samples are prepared.

E.2. Experimental setup: Model and cytometry simulation

The following is a description of our simulation setup:

[3]This data is available at `https://scvelo.readthedocs.io/scvelo.datasets.pancreas/`.

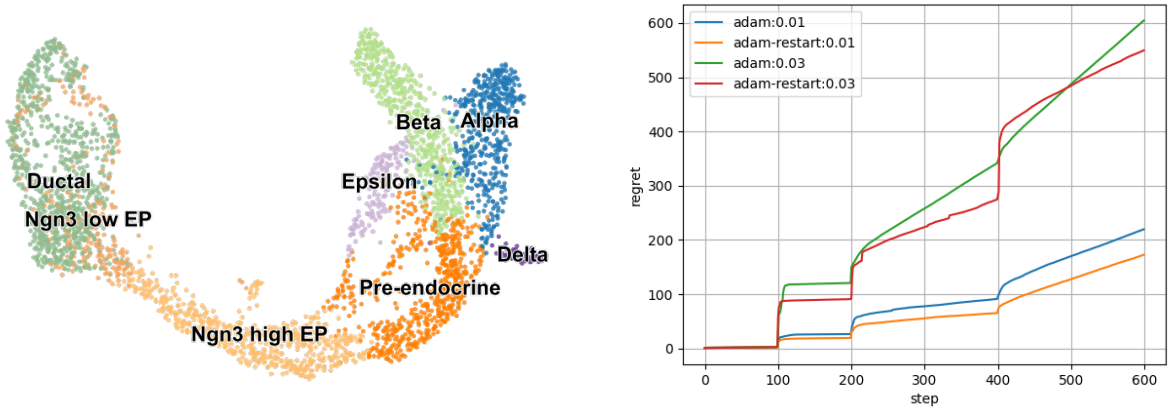

Figure 6: Visualization of the 10-dimensional cytometry data and their ground truth labels (left). Cumulative regret of online models using different initial learning rates and optional Adam restarts at the beginning of each distribution shift (right).

- *Training data and distribution shift:* Each training example is a 10-dimensional vector $x \in \mathbb{R}^{10}$ drawn from a mixture distribution of 4000 murine pancreatic cells and updated by randomly perturbing each of its RNA expressions by a factor $U \sim [0.9, 1.1]$ drawn i.i.d. The label y is the cell type: ductal, alpha, beta, delta. We consider a shift between four different mixture distributions:

  1. $P_1(y) = (0.0, 0.0, 0.0, 1.0)$ for 100 steps
  2. $P_2(y) = (0.0, 0.0, 0.1, 0.9)$ for 100 steps
  3. $P_3(y) = (0.1, 0.0, 0.2, 0.7)$ for 200 steps
  4. $P_4(y) = (0.3, 0.5, 0.1, 0.1)$ for 200 steps

  The first distribution only contains perturbed non-secretory (ductal) cells. Then, each successive mixture distribution increases the probability of a secretory cell, simulating the cell arrival statistics as we sweep from right to left over a section of the pancreas for this data.

- *Neural network:* The input is a 10-dimensional vector of RNA expression levels for the cell. We then use a feedforward neural network with five hidden layer and dimensions $(64, 32, 16, 8, 4)$. Each hidden layer uses an ELU activation, and the last 4-dimensional embedding after activation are the logits for the cell type.

- *Loss and optimizer:* We use categorical cross entropy loss with `from_logits=true` for stability. Each step uses a batch of $B_t = 64$ new examples to simulate the data stream. We optimize this model in an online manner using Adam (Kingma & Ba, 2014) for different initial learning rates and by optionally resetting its parameters at the beginning of a distribution shift. We plot the cumulative regret in Figure 6 (right), where the regret for each step is defined in (2).

We draw several conclusions from the results of this experiment. First, while larger learning rates are often better for minimizing the regret of an online SGD-based system, there is a normally a sweet spot before the first step size that causes the SGD to diverge. In this experiment, an initial learning rate of 0.1 for Adam caused the model to diverge but the total regret is minimized with an initial learning rate of 0.01, achieving less regret than $\eta_0 \in \{0.001, 0.003, 0.03\}$. Second, resetting the Adam optimizer at the beginning of each distribution shift (which increases its step size) allows us to achieve less cumulative regret, as these models can more quickly adjust to the new data distributions. Finally, the models get stuck in local minima without adaptive and increasing learning rate schedules, as evident by the $\eta_0 = 0.03$ plots in Figure 6 (right), which have different slopes in the final two phases.

Future works

We propose extending our SDE framework to develop adaptive adjustment schemes for other hyperparameters in SGD variants such as Polyak averaging (Polyak & Juditsky, 1992), SVRG (Johnson & Zhang, 2013), and elastic averaging SGD (Zhang et al., 2015), as well as deriving effective adaptive momentum parameter adjustment policies. We also propose

studying a "model hedging" question to quantify how neutral a model should remain at a given time to optimally trade off between underfitting the current distribution and being able to quickly adapt to a (possibly adversarial) future distribution. We believe this area of designing adaptive learning rate schedules is a fruitful and exciting area that combines control theory, online optimization, and large-scale recommender systems (Anil et al., 2022; Coleman et al., 2023).