Journal of Machine Learning Research 27 (2026) 1-31 Submitted 8/25; Revised 3/26; Published 4/26

Minimax Optimal Convergence of Gradient Descent in Logistic Regression via Large and Adaptive Stepsizes

Ruiqi Zhang RQZHANG@BERKELEY.EDU
Department of Statistics
University of California, Berkeley

Jingfeng Wu UUUJF@BERKELEY.EDU
Simons Institute for the Theory of Computing
University of California, Berkeley

Licong Lin LICONGLIN@BERKELEY.EDU
Department of Statistics
University of California, Berkeley

Peter L. Bartlett PETER@BERKELEY.EDU
Departments of Statistics and Computer Sciences
University of California, Berkeley and Google DeepMind

Editor: Sebastian Stich

Abstract

We study *gradient descent* (GD) for logistic regression on linearly separable data with stepsizes that adapt to the current risk, scaled by a constant hyperparameter η. We show that after at most $1/\gamma^2$ burn-in steps, GD achieves a risk upper bounded by $\exp(-\Theta(\eta))$, where γ is the margin of the dataset. As η can be arbitrarily large, GD attains an arbitrarily small risk *immediately after the burn-in steps*, though the risk evolution may be *non-monotonic*.

We further construct hard datasets with margin γ, where any batch (or online) first-order method requires $\Omega(1/\gamma^2)$ steps to find a linear separator. Thus, GD with large, adaptive stepsizes matches the worst-case $1/\gamma^2$ dependence when the sample size is unrestricted. Notably, the classical *Perceptron* (Novikoff, 1962), a first-order online method, also achieves a step complexity of $1/\gamma^2$, matching GD even in constants.

Finally, our GD analysis extends to a broad class of loss functions and certain two-layer networks.

Keywords: logistic regression, gradient descent, edge of stability, adaptive stepsizes, minimax optimality

1. Introduction

A large class of optimization methods in machine learning are variants of *gradient descent* (GD). In this paper, we study the convergence of GD with *adaptive* stepsizes given by

$$\mathbf{w}_{t+1} := \mathbf{w}_t - \eta_t \nabla \mathcal{L}(\mathbf{w}_t), \quad t \geq 0, \tag{GD}$$

where $\mathcal{L}(\cdot)$ is the objective to be minimized, $\mathbf{w}_t \in \mathbb{R}^d$ is the trainable parameters, and $(\eta_t)_{t\geq 0}$ are the stepsizes. We allow the stepsize η_t to adapt to the current risk $\mathcal{L}(\mathbf{w}_t)$.

Classical analyses of GD require the stepsizes to be sufficiently small so that the risk decreases monotonically (Nesterov, 2018). This is often referred to as the *descent lemma.* For example, the descent lemma is satisfied for stepsizes such that $\eta_t < 2/\sup_{\mathcal{I}} \lambda_{\max}(\nabla^2 \mathcal{L}(\cdot))$, where the supremum is taken over the interval $\mathcal{I}$ connecting $\mathbf{w}_t$ and $\mathbf{w}_{t+1}$ and $\lambda_{\max}(\cdot)$ is the maximum eigenvalue of a symmetric matrix. In this regime, a large volume of theory has been developed to show the convergence of GD in a variety of settings (see Lan, 2020, for example).

However, practical deep learning models trained by GD often converge in the long run while suffering from a locally oscillatory risk (Wu et al., 2018; Lewkowycz et al., 2020; Cohen et al., 2020). This oscillation occurs when the stepsizes are too large and violate the descent lemma. This unstable convergence phenomenon is referred to by Cohen et al. (2020) as the *edge of stability* (EoS). To obtain a reasonable optimization and generalization performance in practice, it is often necessary to use large stepsizes so that GD enters the EoS regime (Wu et al., 2018; Cohen et al., 2020), violating the seemingly theoretically desirable descent lemma.

Surprisingly, a recent line of works showed that GD provably converges faster by violating the descent lemma in various settings (see Altschuler and Parrilo, 2024; Wu et al., 2024, for examples; other related works are discussed later in Section 1.2). Specifically, Altschuler and Parrilo (2024) proposed a stepsize scheduler in which GD violates the descent lemma but (occasionally) achieves an improved convergence rate for smooth convex optimization. The work by Wu et al. (2024) focused on GD with a constant stepsize for logistic regression with linearly separable data. They showed that GD with a large stepsize can achieve an accelerated convergence rate, while this is impossible if the stepsize is small and enables the descent lemma. Note that the stepsizes considered in (Altschuler and Parrilo, 2024; Wu et al., 2024) are *oblivious*, which are determined before the GD run and do not adapt to the evolving risk.

1.1 Our Results

This work complements the prior theory by considering the convergence of GD with large and *adaptive* stepsizes. We focus on logistic regression with linearly separable data. Specifically, the risk to be minimized is given by

$$\mathcal{L}(\mathbf{w}) := \frac{1}{n}\sum_{i=1}^{n} \ell(y_i \mathbf{x}_i^\top \mathbf{w}), \quad \ell \in \big\{\ell_{\mathsf{exp}} : z \mapsto \exp(-z),\ \ell_{\mathsf{log}} : z \mapsto \ln(1+\exp(-z))\big\}, \tag{1}$$

where the loss function can be exponential or logistic losses and the dataset $(\mathbf{x}_i, y_i)_{i=1}^n$ satisfies the following standard conditions (Novikoff, 1962):

Assumption 1 (Linear Separability) *Assume the dataset $(\mathbf{x}_i, y_i)_{i=1}^n$ satisfies*

A. for every $i = 1, \ldots, n$, $\|\mathbf{x}_i\| \le 1$ and $y_i \in \{\pm 1\}$;

B. there is a margin $\gamma > 0$ and a unit vector $\mathbf{w}^*$ such that $y_i \mathbf{x}_i^\top \mathbf{w}^* \ge \gamma$ for every $i = 1, \ldots, n$.

Under Assumption 1, the minimizer of the risk in (1) is at infinity; moreover, the landscape becomes flatter as the risk decreases. To compensate for this effect, we consider GD with the following *adaptive* stepsizes (proposed by Nacson et al., 2019; Ji and Telgarsky, 2021),

$$\eta_t := \eta \cdot (-\ell^{-1})' \circ \mathcal{L}(\mathbf{w}_t) \approx \eta / \mathcal{L}(\mathbf{w}_t),$$

where $\eta > 0$ is a constant hyperparameter.

We make the following significant contributions.

Benefits of Large and Adaptive Stepsizes. We show that GD with adaptive stepsizes achieves improved convergence by entering the EoS regime. Specifically, we show that after at most $1/\gamma^2$ burn-in steps, GD attains a risk upper bounded by $\exp(-\Theta(\eta))$, which is arbitrarily small by setting η large enough. By doing so, however, the risk evolution could be non-monotonic. On the other hand, if the hyperparameter η is set such that GD does not enter the EoS regime, we provide examples in which GD needs at least $\Omega(\ln(1/\varepsilon))$ steps to achieve an ε-risk. These together justify the benefits of operating in the EoS regime.

In comparison, prior works on the same problem focused on GD with a large but constant stepsize (Wu et al., 2024) or adaptive but small stepsizes (Ji and Telgarsky, 2021). They established $\widetilde{\mathcal{O}}\big(1/(\gamma^2\sqrt{\varepsilon})\big)$ and $\mathcal{O}(\ln(1/\varepsilon)/\gamma^2)$ step complexity for GD to attain an ε-risk respectively (see Sections 2.2 and 3 for comparisons with other related works). In stark contrast, we show that, using large and adaptive steps together, $1/\gamma^2$ steps are sufficient.

Minimax Lower Bounds. Furthermore, we construct hard datasets with margin γ, in which *any* batch first-order methods must take $\Omega(\min\{\ln n, 1/\gamma^2\})$ steps to identify a linear separator of the dataset, while any online first-order method needs at least $\Omega(\min\{n, 1/\gamma^2\})$ updates. Thus, GD with large and adaptive stepsize is minimax optimal (ignoring constant factors) among all first-order batch methods when the sample size n is unrestricted. It is worth noting that the seminal *Perceptron* algorithm (Novikoff, 1962), a first-order online method, also takes $1/\gamma^2$ steps to find a linear separator. This matches GD even with constants.

General Losses and Two-Layer Networks. Finally, we extend our results beyond logistic regression. We establish conditions on the loss functions that enable our analysis. We also obtain similar results for a class of two-layer networks for fitting linearly separable data.

1.2 Related Works

In this part, we review and discuss additional related papers.

Edge of Stability. Our research is motivated by the empirically observed *edge of stability* (EoS) phenomenon (see Cohen et al., 2020, and references therein). Specifically, in practice, GD often induces a convergent yet oscillatory risk, implying the stepsizes are large such that the descent lemma is violated. A growing body of papers investigates the theoretical mechanism of EoS under various scenarios (Kong and Tao, 2020; Wang et al., 2022b, 2023; Lyu et al., 2022; Wang et al., 2022c; Ma et al., 2022; Zhu et al., 2023; Damian et al., 2022; Ahn et al., 2022, 2023; Chen and Bruna, 2023; Kreisler et al., 2023; Even et al., 2023; Andriushchenko et al., 2023; Lu et al., 2023; Chen et al., 2024). We are less interested in characterizing the mechanism of EoS; rather, we focus on understanding the benefits of operating in the EoS regime for improving optimization efficiency.

Aggressive Stepsize Schedulers. For smooth and (strongly) convex optimization, a recent line of research showed that GD with certain stepsize schedulers converges faster than GD with a constant stepsize (Kelner et al., 2022; Altschuler and Parrilo, 2025, 2024; Grimmer, 2024; Zhang and Jiang, 2024; Zhang et al., 2025; Grimmer et al., 2025). Similar to our results, they obtained the acceleration because their stepsize schedulers violate the descent lemma (occasionally). However, there are several notable differences. First, the considered problem classes are not directly comparable. They considered smooth and (strongly) convex optimization problems where the minimizers are finite. In contrast, we study the problem of finding a linear separator of a linearly separable dataset by minimizing a convex loss; this includes logistic regression as a special case. Although our logistic regression problem is smooth and convex, it does not admit a finite minimizer and, thus, does not belong to their problem class. Moreover, their stepsize schedulers are *oblivious*, determined before the GD run, while our stepsizes are *adaptive* to the current risk.

Logistic Regression. Logistic regression with linearly separable data is a standard testbed for studying the implicit bias of GD (see Soudry et al., 2018; Ji and Telgarsky, 2018, and references thereafter). Specifically, this means that GD with a small constant stepsize (satisfying the descent lemma) diverges to infinity in norm but converges to the direction that maximizes the margin (Soudry et al., 2018; Ji and Telgarsky, 2018). The same result is later extended to GD with an arbitrarily large constant stepsize by Wu et al. (2023). Their results imply a risk convergence rate of $\widetilde{\mathcal{O}}(1/t)$ for GD with a small constant stepsize, where t is the number of steps (Soudry et al., 2018; Ji and Telgarsky, 2018). Remarkably, Wu et al. (2024) showed that with a large constant stepsize, GD enters the EoS regime and attains an accelerated $\widetilde{\mathcal{O}}(1/t^2)$ rate. Our work is directly motivated by Wu et al. (2024), although we consider GD with large and adaptive stepsizes while they focused on GD with a large constant stepsize.

Compared to the constant stepsize, it is known that adaptive stepsizes improve the risk (and margin) convergence rate of GD for logistic regression with separable data (Nacson et al., 2019; Ji and Telgarsky, 2021). Specifically, Nacson et al. (2019) and Ji and Telgarsky (2021) showed risk convergence rates of $\exp(-\Omega(\sqrt{t}))$ and $\exp(-\Omega(t))$, respectively. But they both considered GD with adaptive but small stepsizes that satisfy the descent lemma. In comparison, for the same algorithm, we show that simply letting the stepsizes be large can greatly accelerate the risk convergence.

Additionally, the risk (and margin) convergence rate can be further accelerated by using momentum techniques with GD (Ji et al., 2021; Wang et al., 2022a). The work by Ji et al. (2021) obtained $\exp(-\Omega(t^2 - \ln(t)))$ risk convergence rate (see Proposition 6), where the $\ln(t)$ term was later removed by Wang et al. (2022a). Their results are again limited to small stepsizes such that a certain potential decreases monotonically (see the discussions after Proposition 6). In contrast, we show that not using momentum but just using large stepsizes can lead to faster convergence.

Finally, we highlight that our lower bounds suggest that GD with large and adaptive stepsizes is minimax optimal. In contrast, GD with a large but constant stepsize (Wu et al., 2024) or adaptive but small stepsizes (Ji and Telgarsky, 2021) or momentum (Ji et al., 2021; Wang et al., 2022a) are all suboptimal.

Perceptron. The seminal paper by Novikoff (1962) showed that the *Perceptron* algorithm takes $1/\gamma^2$ steps to find a linear separator of a dataset with margin γ. Our lower bound matches this rate up to constants in the online first-order setting. Note that Perceptron is an *online* method while GD for logistic regression considered in this paper is a *batch* method. In our analysis, entering the EoS regime is a key factor enabling GD to be minimax optimal. Interestingly, Perceptron can be viewed as one-pass stochastic gradient descent with stepsize 1 under the hinge loss, $z \mapsto \max\{0, -z\}$. Since the hinge loss is nonsmooth, stepsize 1 violates the descent lemma, so Perceptron effectively operates in the EoS regime, too. It seems that operating in the EoS regime might be necessary for achieving first-order minimax optimality in this problem.

A recent paper by Tyurin (2025) proposed a batch version of the Perceptron algorithm, attaining the same step complexity of $1/\gamma^2$ when translated to our notation. This is also minimax optimal according to our lower bound. Their method can be viewed as GD with adaptive stepsizes. Their stepsize scheme is motivated by Perceptron, while our stepsize scheme is motivated by the loss landscape (Nacson et al., 2019; Ji and Telgarsky, 2021). We also point out that Tyurin (2025) did not provide minimax lower bounds.

A recent paper by Kornowski and Shamir (2025) considered lower bounds for finding linear separators of a linearly separable dataset from a game-theoretic perspective. Our lower bounds are connected to theirs but with several differences. First, our lower bound for batch methods (Theorem 8) is comparable with their Theorem 4.2: their Theorem 4.2 covers more algorithms, but only showing an $\Omega(\gamma^{-2/3})$ lower bound (ignoring dependence on sample size), while our Theorem 8 shows an $\Omega(\gamma^{-2})$ lower bound but only covers first-order batch methods. Second, their Theorem 4.1 matches our lower bound in Theorem 10 for online methods while covering more algorithms. They both can be viewed as solutions to Shalev-Shwartz and Ben-David (2014, Section 9.6, Exercise 3). We choose to keep our version as it is easier to use in our context.

2. Logistic Regression

In this section, we present an improved analysis for GD with large and adaptive stepsizes for logistic regression on linearly separable data.

2.1 Convergence of GD with Large and Adaptive Stepsizes

Recall the definitions of (GD) and the objective (1). Recall the adaptive stepsizes are defined as

$$\eta_t := \eta \cdot \left(-\ell^{-1}\right)' \circ \mathcal{L}(\mathbf{w}_t) = \begin{cases} \dfrac{\eta}{\mathcal{L}(\mathbf{w}_t)} & \ell = \ell_{\mathsf{exp}}, \\ \dfrac{\eta \exp(\mathcal{L}(\mathbf{w}_t))}{\exp\left(\mathcal{L}(\mathbf{w}_t)\right) - 1} & \ell = \ell_{\mathsf{log}}. \end{cases} \tag{2}$$

It is easy to check that $\|\nabla^2 \mathcal{L}(\mathbf{w})\| \le \mathcal{L}(\mathbf{w})$ under Assumption 1. Thus, the landscape becomes flatter as the risk $\mathcal{L}(\mathbf{w})$ decreases. The adaptive stepsizes (2) are designed to compensate for the flattened curvature (Nacson et al., 2019; Ji and Telgarsky, 2021). Moreover, we point out that GD with adaptive stepsizes (2) for logistic regression (1) is equivalent to (Ji and Telgarsky, 2021)

$$\mathbf{w}_{t+1} := \mathbf{w}_t - \eta \nabla \phi(\mathbf{w}_t), \quad \phi(\mathbf{w}) := -\ell^{-1}\left(\mathcal{L}(\mathbf{w})\right). \tag{3}$$

This is GD with a constant stepsize η under a transformed objective $\phi(\cdot)$. Equivalently, $\phi(\cdot)$ can be viewed as a smooth surrogate of the unnormalized margin (Hardy et al., 1952). This has been exploited by Ji and Telgarsky (2021), where they established a primal-dual analysis of GD and obtained an improved margin convergence rate. Unlike Ji and Telgarsky (2021), we focus on the risk convergence and obtain the following improved results.

Theorem 2 (GD with Large and Adaptive Stepsizes) *Consider* (GD) *with adaptive stepsizes* (2) *for logistic regression* (1) *under Assumption 1. Assume without loss of generality that* $\mathbf{w}_0 = \mathbf{0}$*. Then for every* $t \geq 1$ *and* $\eta > 0$*, we have*

$$\mathcal{L}\left(\overline{\mathbf{w}}_t\right) \leq \exp\bigg(-\frac{(\gamma^2(t+1))^2 - 1}{4\gamma^2(t+1)}\eta\bigg), \quad \textit{where} \;\; \overline{\mathbf{w}}_t := \frac{1}{t+1}\sum_{k=0}^{t}\mathbf{w}_k.$$

In particular, after $1/\gamma^2$ *burn-in steps, for every* $\eta > 0$*, we have*

$$\mathcal{L}(\overline{\mathbf{w}}_t) \leq \exp\left(-\frac{\gamma^2\eta}{4}\right) = \exp\left(-\Theta\left(\eta\right)\right), \quad t \geq \frac{1}{\gamma^2}.$$

The proof of Theorem 2 is deferred to Section 2.3. Theorem 2 provides a sharp risk convergence bound for GD with large and adaptive stepsizes. Note that the hyperparameter η can be chosen arbitrarily large. Therefore, with a sufficiently large η, GD achieves an arbitrarily small risk right after $1/\gamma^2$ burn-in steps. In this case, however, the evolution of the risk might not be monotonic.

In other words, to attain an ε-risk, GD with adaptive and large stepsize only needs $1/\gamma^2$ steps, where the step complexity is independent of targeted risk ε (but the smallest base stepsize depends on ε). This is in stark contrast to the step complexity of GD with a large but constant stepsize (Wu et al., 2024) or adaptive but small stepsizes (Ji and Telgarsky, 2021) (see Section 2.2 and a detailed discussion later in Section 2.2). Moreover, we will show that this $1/\gamma^2$ step complexity is minimax optimal up to constant factors in Section 3. We note that Theorem 2 is stated for the averaged iterate $\overline{\mathbf{w}}_t$. The same proof also yields a comparable best-iterate guarantee for $\min_{0\leq k\leq t}\mathcal{L}(\mathbf{w}_k)$, but it does not provide a comparable last-iterate guarantee for $\mathcal{L}(\mathbf{w}_t)$ during the non-monotone phase induced by large adaptive stepsizes. We treat this as one important future research direction.

Benefits of EoS. In Theorem 2, GD achieves the $1/\gamma^2$ step complexity by using large stepsizes and entering the EoS regime. Our next theorem suggests this is necessary by providing a lower bound on the convergence rate for adaptive stepsize GD that avoids the EoS phase.

Theorem 3 (A Lower Bound for GD in the Stable Regime) *Consider* (GD) *with adaptive stepsizes* (2) *for logistic regression* (1) *with the following dataset*

$$\mathbf{x}_1 = (\gamma,\, \sqrt{1-\gamma^2}), \quad \mathbf{x}_2 = (\gamma,\, -\sqrt{1-\gamma^2}), \quad y_1 = y_2 = 1,$$

where $0 < \gamma < 0.1$*. This dataset satisfies Assumption 1. Let* $\mathbf{w}_0 = \mathbf{0}$*. For all hyperparameter* η *such that* $(\mathcal{L}(\mathbf{w}_t))_{t\geq 0}$ *is nonincreasing, we have*

$$\mathcal{L}(\overline{\mathbf{w}}_t),\, \mathcal{L}(\mathbf{w}_t) \geq \exp\left(-ct\right), \quad t \geq 1,$$

where $c > 0$ *is a parameter that depends on* γ *but is independent of* t *and* η*.*

stepsize/momentum design	step complexity
small, constant (Ji and Telgarsky, 2018, Theorem 3.1)	$\widetilde{\mathcal{O}}\left(1/\left(\gamma^2\varepsilon\right)\right)$
large, constant (Wu et al., 2024, Corollary 2)	$\widetilde{\mathcal{O}}\left(1/\left(\gamma^2\sqrt{\varepsilon}\right)\right)$ for $\varepsilon < \Theta(1/n)$
small, adaptive (Ji and Telgarsky, 2021, Theorem 2.2)	$\mathcal{O}\left(\ln\left(1/\varepsilon\right)/\gamma^2\right)$
small, adaptive, and momentum (Ji et al., 2021, Theorem 3.1)	$\mathcal{O}\big(\sqrt{\ln(1/\varepsilon)+\ln(n)\ln\ln(n)}/\gamma\big)$
large, adaptive (Theorem 2)	$\leq 1/\gamma^2$
minimax lower bound (Theorem 8)	$\Omega(1/\gamma^2)$

Table 1: Step complexities for GD with various designs to achieve an ε-risk for logistic regression.

The proof of Theorem 3 is deferred to Appendix B. Theorem 3 is motivated by Theorem 3 in (Wu et al., 2024), which provides a risk lower bound for GD with a constant stepsize that satisfies the descent lemma. Theorem 3 shows that if GD with adaptive stepsizes satisfies the descent lemma, then the step complexity is at least $\Omega(\ln(1/\varepsilon))$ in the worst case. By contrast, Theorem 2 shows that GD achieves an ε-independent step complexity by violating the descent lemma. Theorems 2 and 3 together justify the optimization benefits of adaptive stepsize GD to operate in the EoS regime.

2.2 Comparisons with Prior Results

In this part, we review representative existing results on variants of GD for logistic regression with linearly separable data (Ji and Telgarsky, 2018, 2021; Ji et al., 2021; Wu et al., 2024) and compare their results with ours. Section 2.2 provides an overview of the comparisons. We discuss each result in detail below.

A Constant Stepsize. The work by Ji and Telgarsky (2018) considered GD with a small constant stepsize satisfying the descent lemma and obtained a $\widetilde{\mathcal{O}}(1/(\gamma^2 t))$ convergence rate (see their Theorem 3.1). This translates to a $\widetilde{\mathcal{O}}(1/(\gamma^2\varepsilon))$ step complexity. Later, the work by Wu et al. (2024) obtained a faster rate by considering GD with a large constant stepsize that violates the descent lemma. Specifically, they showed the following.

Proposition 4 (Corollary 2 in (Wu et al., 2024)) *Consider* (GD) *with constant stepsize* $\eta_t = \eta > 0$ *for logistic regression* (1) *with logistic loss* ℓ_{log} *under Assumption 1. Let* $\mathbf{w}_0 = 0$*. For a given step budget* $T \geq \max\{e, n\}/\gamma^2$*, there exists* $\eta = \Theta(T)$ *such that*

$$\mathcal{L}(\mathbf{w}_T) \leq C\frac{\ln^2(T)}{\gamma^4 T^2},$$

where $C > 1$ *is a numerical constant.*

Proposition 4 leads to an improved step complexity, $\widetilde{\mathcal{O}}(1/(\gamma^2\sqrt{\varepsilon}))$, for GD with a large constant stepsize. There are three caveats. First, this improvement only happens for $\varepsilon < \Theta(1/n)$ (hence, it does not help with finding a linear separator; see Section 3). Second, this improvement works under the logistic loss but not under the exponential loss; in fact, Wu et al. (2023) constructed a separable dataset where GD with a large constant stepsize under the exponential loss does not converge (see their Theorem 4.2). This gap stems from the logistic loss being Lipschitz while the exponential loss is not. Finally, the stepsize is a function of the step budget, meaning that the algorithm needs to be rerun from the beginning when the step budget is changed.

Compared to their results for GD with a constant stepsize (Ji and Telgarsky, 2018; Wu et al., 2024), we show that GD with large and adaptive stepsizes (2) achieves a strictly better step complexity of $1/\gamma^2$. Interestingly, our analysis allows for both logistic and exponential losses. This is not contradictory to the counter-example offered by Wu et al. (2023). Recall that GD with adaptive stepsizes (2) can be viewed as GD with a constant stepsize under a transformed objective $\phi(\cdot)$ defined in (3). While the exponential loss is not Lipschitz, the transformed objective $\phi(\cdot)$ is Lipschitz (see Lemma 14 in Appendix A), enabling large stepsizes. Finally, unlike Proposition 4, the adaptive stepsize scheduler (2) used in Theorem 2 is independent of the step budget (as long as setting η sufficiently large).

Small Adaptive Stepsizes. Before our paper, the best analysis for GD with adaptive stepsizes (2) is by Ji and Telgarsky (2021). Their results only allow small stepsizes, summarized as follows.

Proposition 5 (Consequences of Theorem 2.2 in (Ji and Telgarsky, 2021)) *Consider* (GD) *with adaptive stepsizes* (2) *for logistic regression* (1) *with exponential loss* ℓ_{exp} *under Assumption 1. Then the transformed loss* ϕ *(defined in* (3)*) is* 1*-smooth with respect to* ℓ_∞*-norm. Let* $\mathbf{w}_0 = 0$*. Then for every* $\eta \le 1$*, we have* $\mathcal{L}(\mathbf{w}_t)$ *decreases monotonically and*

$$\mathcal{L}(\mathbf{w}_t) \le C \exp\left(-\gamma^2 \eta t\right),$$

where $C > 1$ *is a numerical constant.*

We note that the main focus of Ji and Telgarsky (2021) is to prove a fast margin convergence rate, and Proposition 5 is merely a side product of their results. Proposition 5 leads to an $\mathcal{O}(\ln(1/\varepsilon)/\gamma^2)$ step complexity for GD with small adaptive stepsizes. In comparison, we analyze the same algorithm, but our Theorem 2 applies to both large and small adaptive stepsizes. Our results suggest that the step complexity reduces to $1/\gamma^2$ when the stepsizes are sufficiently large.

Momentum. The work by Ji et al. (2021) considered a version of momentum GD for logistic regression, achieving a faster margin convergence rate compared to GD. Their results also imply a risk convergence rate, summarized as follows.

Proposition 6 (Consequences of Lemmas C.7 and C.12 in (Ji et al., 2021)) *Consider logistic regression* (1) *with exponential loss* ℓ_{exp} *under Assumption 1. For a version of momentum GD (see Algorithm 1 in Ji et al. (2021)) with* $\mathbf{w}_0 = 0$ *and suitably chosen stepsizes, we have*

$$\mathcal{L}(\mathbf{w}_t) \le \exp\left(-C\left(\gamma^2 t^2 - \ln(t+1)\ln(n)\right)\right), \quad t \ge 1,$$

where $C > 0$ is a numerical constant.

Proposition 6 implies an $\mathcal{O}\big(\sqrt{\ln(1/\varepsilon) + \ln(n)\ln\ln(n)}/\gamma\big)$ step complexity for momentum GD. We remark that with a more careful momentum design, Wang et al. (2022a) improved the $\ln(t+1)\ln(n)$ term in the above proposition to $\ln(n)$, which leads to a slightly improved step complexity of $\mathcal{O}\big(\sqrt{\ln(1/\varepsilon) + \ln(n)}/\gamma\big)$. As the improvement is small, we choose to mainly compare with the earlier results by Ji et al. (2021).

Recall that our Theorem 2 suggests a $1/\gamma^2$ step complexity for GD with large and adaptive stepsizes. In comparison, the step complexity of their momentum GD has a better dependence on γ but has a worse dependence on ε and n. As shown later in Section 3, our step complexity is minimax optimal if there are no restrictions on the sample size n. Note that the analysis by Ji et al. (2021) still relies on the monotonic decreasing of a potential (see Ji et al., 2021, Equation (B.9) in the proof of Lemma B.3), which needs the stepsize to be small. As their momentum GD is designed to minimize a dual objective with a finite minimizer, it remains unclear whether their momentum GD can be used with large stepsizes. We leave this as future work.

2.3 Proof of Theorem 2

Proof (of Theorem 2) As explained in (3), it is equivalent to considering GD with a constant stepsize under a transformed objective $\phi(\cdot)$. We can check that $\phi(\cdot)$ is convex (see Lemma 5.2 in (Ji and Telgarsky, 2021) or Lemma 16 in Appendix A) and 1-Lipschitz (see Lemma 14 in Appendix A). We then use the split optimization technique developed by Wu et al. (2024). Specifically, for a comparator $\mathbf{u} := \mathbf{u}_1 + \mathbf{u}_2$, we have

$$\begin{aligned}
\|\mathbf{w}_{t+1} - \mathbf{u}\|^2 &= \|\mathbf{w}_t - \mathbf{u}\|^2 + 2\eta\langle\nabla\phi(\mathbf{w}_t), \mathbf{u} - \mathbf{w}_t\rangle + \eta^2\|\nabla\phi(\mathbf{w}_t)\|^2 \\
&= \|\mathbf{w}_t - \mathbf{u}\|^2 + 2\eta\langle\nabla\phi(\mathbf{w}_t), \mathbf{u}_1 - \mathbf{w}_t\rangle + \eta\left[2\langle\nabla\phi(\mathbf{w}_t), \mathbf{u}_2\rangle + \eta\|\nabla\phi(\mathbf{w}_t)\|^2\right] \\
&\le \|\mathbf{w}_t - \mathbf{u}\|^2 + 2\eta\langle\nabla\phi(\mathbf{w}_t), \mathbf{u}_1 - \mathbf{w}_t\rangle \qquad (4) \\
&\le \|\mathbf{w}_t - \mathbf{u}\|^2 + 2\eta\left(\phi(\mathbf{u}_1) - \phi(\mathbf{w}_t)\right), \qquad (5)
\end{aligned}$$

where (4) is by the following inequality (see Lemma 15 in Appendix A)

$$2\langle\nabla\phi(\mathbf{w}), \mathbf{u}_2\rangle + \eta\|\nabla\phi(\mathbf{w})\|^2 \le 0 \quad \text{for} \quad \mathbf{u}_2 := \frac{\eta}{2\gamma}\mathbf{w}^*, \tag{6}$$

and (5) is by the convexity of $\phi(\cdot)$. Rearranging (5) and telescoping the sum, we obtain

$$\frac{\|\mathbf{w}_{t+1} - \mathbf{u}\|^2}{2\eta(t+1)} + \frac{1}{t+1}\sum_{k=0}^{t}\phi(\mathbf{w}_k) \le \phi(\mathbf{u}_1) + \frac{\|\mathbf{u}\|^2}{2\eta(t+1)}. \tag{7}$$

For $\mathbf{u}_1 \propto \mathbf{w}^*$, we have $\phi(\mathbf{u}_1) \le -\gamma\|\mathbf{u}_1\|$ by Assumption 1. Further setting $\|\mathbf{u}_1\| = \gamma\eta(t+1)/2$, we get

$$\frac{1}{t+1}\sum_{k=0}^{t}\phi(\mathbf{w}_k) \le -\gamma\|\mathbf{u}_1\| + \frac{\|\mathbf{u}_1 + \mathbf{u}_2\|^2}{2\eta(t+1)} \le -\frac{(\gamma^2(t+1))^2 - 1}{4\gamma^2(t+1)}\eta. \tag{8}$$

We complete the proof by applying the convexity of $\phi(\cdot)$ and the fact that $\mathcal{L}(\cdot) = \ell(-\phi(\cdot))$. ■

stepsize/momentum/loss design	type	step complexity
constant (Ji and Telgarsky, 2018, Theorem 3.1)	batch	$\widetilde{\mathcal{O}}\left(n/\gamma^2\right)$
small, adaptive (Ji and Telgarsky, 2021, Theorem 2.2)	batch	$\mathcal{O}\left(\ln(n)/\gamma^2\right)$
small, adaptive, and momentum (Ji et al., 2021, Theorem 3.1)	batch	$\mathcal{O}\left(\sqrt{\ln(n)\ln\ln(n)/\gamma^2}\right)$
normalized batch Perceptron (Tyurin, 2025, Theorem 5.3)	batch	$\leq 1/\gamma^2$
large and adaptive (Theorem 2)	batch	$\leq 1/\gamma^2$
minimax lower bound (Theorem 8)	batch	$\Omega(\min\{1/\gamma^2, \ln(n)\})$
constant (Wu et al., 2024, Theorem 4)	online	$\mathcal{O}\left(1/\gamma^2\right)$
Perceptron (Novikoff, 1962)	online	$\leq 1/\gamma^2$
minimax lower bound (Theorem 10)	online	$\Omega\big(\min\{1/\gamma^2, n\}\big)$

Table 2: Step complexities for first-order methods to find a linear separator.

3. Minimax Lower Bounds for First-Order Methods

In this section, we consider the task of finding a linear separator for a linearly separable dataset. Specifically, this means to find a parameter $\hat{\mathbf{w}}$ such that

$$\min_{i\in[n]} y_i \mathbf{x}_i^\top \hat{\mathbf{w}} > 0.$$

We point out the fact that an optimization method can find a linear separator by solving the logistic regression problem (1) sufficiently well.

Fact 1 *In logistic regression* (1), *if* $\mathcal{L}(\hat{\mathbf{w}}) < \ell(0)/n$, *then* $\min_{i\in[n]} y_i \mathbf{x}_i^\top \hat{\mathbf{w}} > 0$.

In this section, we establish minimax lower bounds on the step complexity needed by any first-order methods to solve this task. We then compare the performance for GD with various designs in this task, as summarized in Section 3 and will be detailed later in this section. We start with first-order batch methods and then discuss first-order online methods.

3.1 A Lower Bound for First-Order Batch Methods

We formally define first-order batch methods for our problem as follows.

Definition 7 (First-Order Batch Methods) *Let $\ell(\cdot)$ be a locally Lipschitz function. For each z, let $\ell'(z)$ be a unique element from the Clarke subdifferential of $\ell(\cdot)$ at z (Clarke, 1990). For a given dataset $(\mathbf{x}_i, y_i)_{i=1}^n$, define the batch gradient as*

$$\nabla\mathcal{L}(\mathbf{w}) := \frac{1}{n}\sum_{i=1}^{n} \ell'\big(y_i \mathbf{x}_i^\top \mathbf{w}\big) y_i \mathbf{x}_i,$$

We say $\mathbf{w}_t$ *is the output of a first-order batch method in* t *steps with initialization* $\mathbf{w}_0$ *on dataset* $(\mathbf{x}_i, y_i)_{i=1}^n$*, if it can be generated by*

$$\mathbf{w}_k \in \mathbf{w}_0 + \mathsf{Lin}\left\{\nabla\mathcal{L}(\mathbf{w}_0), \dots, \nabla\mathcal{L}(\mathbf{w}_{k-1})\right\}, \quad k = 1, \dots, t,$$

where "Lin*" is the linear span of a vector set and "+" is the Minkowski addition.*

Definition 7 characterizes a class of first-order methods for finding linear separators. Compared to the class of first-order methods for smooth convex optimization (Nesterov, 2018, Assumption 2.1.4), our Definition 7 requires the predictor to be linear—since the goal is to find a linear separator—but does not require the objective function to be smooth or convex. We establish the following lower bounds.

Theorem 8 (A Lower Bound for First-Order Batch Methods) *For every* $0 < \gamma < 1/6$*,* $n > 16$*, and* $\mathbf{w}_0$*, there exists a dataset* $(\mathbf{x}_i, y_i)_{i=1}^n$ *satisfying Assumption 1 such that the following holds. For any* $\mathbf{w}_t$ *output by a first-order batch method in* t*-steps with initialization* $\mathbf{w}_0$ *on this dataset, we have*

$$\min_{i\in[n]} y_i \mathbf{x}_i^\top \mathbf{w}_t > 0 \quad \textit{implies that} \quad t \ge \min\left\{\frac{\ln n}{8\ln 2}, \frac{1}{30\gamma^2}\right\}.$$

The proof of Theorem 8 is deferred to Appendix C.1. The proof is based on a dimension argument, motivated by the classical lower bounds for first-order methods in smooth convex optimization (Nesterov, 2018, Theorem 2.1.7). We emphasize that Theorem 8 should be interpreted as a worst-case lower bound for hard high-dimensional instances as the $1/\gamma^2$ branch is active only when $\ln n \gtrsim 1/\gamma^2$, that is, when n can be exponentially large in $1/\gamma^2$.

Minimax Optimality. Due to Fact 1 and Theorem 2, GD with large and adaptive stepsizes takes $1/\gamma^2$ steps to find a linear separator. Our lower bound in Theorem 8 suggests that this is minimax optimal ignoring constant factors, in the sense that there is no restriction on the sample size n (so n is allowed to be exponential in $1/\gamma$ in the worst case).

Prior Results. Applying Fact 1 to the prior convergence results summarized in Section 2.2, we can conclude the step complexities of the variants of GD considered in previous works (Ji and Telgarsky, 2018, 2021; Ji et al., 2021; Wu et al., 2024) for finding a linear separator. Additionally, the work by Tyurin (2025) provided a first-order batch method called normalized batch Perceptron, which achieves $1/\gamma^2$ step complexity for finding the linear separator. These results are summarized in Section 3.

We make three remarks. First, since the acceleration effect of a large constant stepsize only appears for $\varepsilon < \Theta(1/n)$ (see Section 2.2), a large constant stepsize considered by Wu et al. (2024) does not help GD to find a linear separator (but also does not hurt) compared to the results in (Ji and Telgarsky, 2018). Second, when γ is fixed and the sample size n is allowed to be arbitrary, the methods considered in (Ji and Telgarsky, 2018, 2021; Ji et al., 2021) are all suboptimal in the worst case, while GD with large and adaptive stepsizes and the normalized batch Perceptron by Tyurin (2025) are minimax optimal. Finally, note that Ji et al. (2021) obtained an $\mathcal{O}(\sqrt{\ln(n)\ln\ln(n)/\gamma^2})$ step complexity by using momentum techniques (note that this can be improved to $\mathcal{O}(\sqrt{\ln(n)/\gamma^2})$ by Wang et al. (2022a) as discussed after Proposition 6). Their results do not violate the lower bound in Theorem 8, but suggest that the $1/\gamma^2$ term in our lower bound might be improvable in the regime where $n = \mathrm{poly}(1/\gamma)$.

It turns out that identifying the correct trade-off between n, $1/\gamma$, and the ambient dimension d is challenging. In addition to Theorem 8, we give an alternative dataset construction that leads to a lower bound of $\Omega(\min\{\gamma^{-2/3}, n\})$ (see Theorem 18 in Appendix C.3). We leave it as future work to prove the first-order minimax step complexity that is tight for all choices of $1/\gamma$, n, and the ambient dimension d.

3.2 A Lower Bound for First-Order Online Methods

As a side product, we also establish a step complexity lower bound for first-order online methods for finding a linear separator. We formally define a first-order online method as follows.

Definition 9 (First-Order Online Methods) *Let $\ell(\cdot)$ be a locally Lipschitz function. For each z, let $\ell'(z)$ be a unique element from the Clarke subdifferential of $\ell(\cdot)$ at z (Clarke, 1990). We say a sequence $(\mathbf{w}_k)_{k=0}^t$ is generated by a first-order online method with initialization $\mathbf{w}_0$ on dataset $(\mathbf{x}_i, y_i)_{i=1}^t$, if it satisfies*

$$\mathbf{w}_k \in \mathbf{w}_0 + \mathsf{Lin}\left\{\ell'\big(y_i \mathbf{x}_i^\top \mathbf{w}_{i-1}\big) y_i \mathbf{x}_i,\ i = 1, \dots, k\right\}, \quad k = 1, \dots, t,$$

where "Lin" is the linear span of a vector set and "$+$" is the Minkowski addition.

The following theorem presents our lower bound for the first-order online method. A version of this theorem (that covers more algorithms) also appears in (Kornowski and Shamir, 2025, Theorem 4.1), both of which can be viewed as solutions to Exercise 3 in Section 9.6 of Shalev-Shwartz and Ben-David (2014). Our statement of the lower bound is easier to use in our context.

Theorem 10 (Lower Bounds for Online First-Order Methods) *For every $0 < \gamma < 1/2, n \geq 2$, and $\mathbf{w}_0$, there exists a dataset $(\mathbf{x}_i, y_i)_{i=1}^n$ satisfying Assumption 1 such that the following holds.*

- *For any sequence $(\mathbf{w}_k)_{k=0}^t$ generated by a first-order online method with initialization $\mathbf{w}_0$ and dataset $\big(\mathbf{x}_{\pi(k)}, y_{\pi(k)}\big)_{k=1}^t$, where $\pi(k) \in [n]$ but is otherwise arbitrary, we have*

$$\min_{i \in [n]} y_i \mathbf{x}_i^\top \mathbf{w}_t > 0 \quad \textit{implies that} \quad t \geq \min\left\{\frac{1}{2\gamma^2}, n\right\}.$$

- *For any sequence $(\mathbf{w}_k)_{k=0}^n$ generated by a first-order online method with initialization $\mathbf{w}_0$ and dataset $(\mathbf{x}_i, y_i)_{i=1}^n$, we have*

$$\sum_{k=1}^t \mathbb{1}\{y_k \mathbf{x}_k^\top \mathbf{w}_{k-1} \leq 0\} \geq \min\left\{\frac{1}{2\gamma^2}, t\right\} \quad \textit{for every} \quad 1 \leq t \leq n.$$

We include the proof of Theorem 10 in Appendix C.2 for completeness. We remark that the dataset construction in Theorem 10 is different from the one in Theorem 8. From an offline perspective, Theorem 10 suggests that any first-order online methods need to make $\Omega(1/\gamma^2)$ updates to find a linear separator of a given dataset (assuming that n is large). From an online perspective, it makes $\Omega(1/\gamma^2)$ mistake steps (or regret in zero-one loss) for a large t in the worst case.

Perceptron. Perceptron is a classical method for finding a linear separator of a linearly separable dataset. It takes the following updates:

$$\mathbf{w}_0 = \mathbf{0}, \quad \mathbf{w}_t := \mathbf{w}_{t-1} + \mathbb{1}\{y_t \mathbf{x}_t^\top \mathbf{w}_{t-1} \le 0\} y_t \mathbf{x}_t, \quad t \ge 1.$$

Under Assumption 1, a seminal analysis by Novikoff (1962) suggests that Perceptron makes at most $1/\gamma^2$ mistake steps when run over a dataset of arbitrary size (that is, the total regret in zero-one loss is at most $1/\gamma^2$ for any t).

Note that Perceptron can be viewed as online stochastic gradient descent under the hinge loss $\ell_{\mathrm{hinge}}(z) := \max\{0, -z\}$ with stepsize 1 and subderivative choice $\ell'_{\mathrm{hinge}}(0) := 1$. Thus, Theorem 10 applies to Perceptron, suggesting that Perceptron is minimax optimal when the sample size is unrestricted. This is also known from Kornowski and Shamir (2025, Theorem 4.1) and can be viewed as a solution to Exercise 3 in Section 9.6 of Shalev-Shwartz and Ben-David (2014).

We also point out that the hinge loss is non-smooth. So, the stepsize 1 used in Perceptron also violates the descent lemma (although Perceptron is online), similar to our large stepsize GD. We conjecture that violating the descent lemma might be a fundamental property for attaining first-order optimality in this task.

SGD for Logistic Regression. Similarly to Perceptron, online stochastic gradient descent with a (large) constant stepsize for logistic regression also makes $\mathcal{O}(1/\gamma^2)$ mistake steps (see the proof of Theorem 4 in Wu et al., 2024, for example). Since the logistic loss is close to the hinge loss when zooming away from zero, we should expect SGD with large stepsizes for logistic regression to behave like Perceptron.

4. Extensions

This section extends results in Section 2 to a two-layer network and a generic class of loss functions.

4.1 Two-Layer Networks

We consider a two-layer network defined as (Brutzkus et al., 2018)

$$f(\mathbf{w}; \mathbf{x}) := \frac{1}{m} \sum_{j=1}^{m} a_j \sigma\big(\mathbf{x}^\top \mathbf{w}^{(j)}\big), \quad \mathbf{w} := \big(\mathbf{w}^{(1)}, \mathbf{w}^{(2)}, \dots, \mathbf{w}^{(m)}\big), \tag{9}$$

where m is the number of neurons, $a_j \in \{\pm 1\}$ for $j = 1, \dots, m$ are fixed parameters, $\mathbf{w}$ are the trainable parameters, and $\sigma(z) := \max\{z, \alpha z\}$ for $0 < \alpha < 1$ is the leaky ReLU activation. We fix a choice of subderivative $\sigma'(0) := 1$ for clarity. But our results extend to any choice of subderivative $\sigma'(0) \in [\alpha, 1]$. The objective is then given by

$$\mathcal{L}(\mathbf{w}) := \frac{1}{n} \sum_{i=1}^{n} \ell\left(y_i f\left(\mathbf{w}; \mathbf{x}_i\right)\right), \tag{10}$$

where $\ell(\cdot)$ is exponential loss or logistic loss and the dataset satisfies Assumption 1. Similarly, we consider GD with adaptive stepsizes (2). The next theorem provides an improved convergence analysis for GD with large and adaptive stepsizes.

Theorem 11 (A Two-Layer Network) *Consider* (GD) *with adaptive stepsizes* (2) *for minimizing objective* (10) *under Assumption 1. Assume without loss of generality that* $\mathbf{w}_0 = \mathbf{0}$. *Then for every* $t \geq 1$ *and* $\eta > 0$, *we have*

$$\min_{k \leq t} \mathcal{L}(\mathbf{w}_k) \leq \exp\left(-\frac{(\alpha\gamma^2(t+1))^2 - 1}{4\gamma^2(t+1)}\eta\right).$$

In particular, after $1/(\alpha\gamma^2)$ *burn-in steps, for every* $\eta > 0$, *we have*

$$\min_{k \leq t} \mathcal{L}(\mathbf{w}_k) \leq \exp\left(-\frac{\alpha^2\gamma^2\eta}{4}\right) = \exp\left(-\Theta\left(\eta\right)\right), \quad t \geq \frac{1}{\alpha\gamma^2}.$$

The proof of Theorem 11 is deferred to Appendix D, which combines techniques from the proof of Theorem 2 and the analysis of GD with a large constant stepsize for two-layer networks by Cai et al. (2024). Theorem 11 suggests GD with large and adaptive stepsizes attains an arbitrarily small risk right after $1/(\alpha\gamma^2)$ burn-in steps. In comparison, the work by Cai et al. (2024) only obtained an $\widetilde{\mathcal{O}}(1/t^2)$ rate. Other discussions for Theorem 2 also apply here in a similar manner.

We remark that the leaky ReLU activation in Theorem 11 can be extended to other near-homogeneous activation functions, such as leaky GeLU, leaky Softplus, and leaky SiLU (see Cai et al., 2024). This is done in Appendix D.

4.2 General Loss Functions

In this part, we extend our results in Section 2 from logistic and exponential losses to a generic class of loss functions.

Assumption 12 (Loss Function Conditions) *Let* $\ell : \mathbb{R} \to (0, \infty)$ *be a positive and continuously differentiable function. Assume that*

A. *The loss function* ℓ *is positive, strictly decreasing, and convex.*

B. *For* $z > 0$, *the inverse* $\ell^{-1}(z)$ *exists, and is differentiable and decreasing.*

C. *The transformed objective* $\phi(\cdot)$ *in* (3) *is convex and* C_ℓ*-Lipschitz for a constant* $C_\ell \geq 1$.

Our next theorem shows that Theorem 2 applies to loss functions beyond the logistic and exponential losses, as long as they satisfy Assumption 12.

Theorem 13 (General Loss Functions) *Consider* (GD) *with adaptive stepsizes* (2) *for objective function* $\mathcal{L}(\mathbf{w}) := (1/n)\sum_{i=1}^{n} \ell\left(y_i \mathbf{x}_i^\top \mathbf{w}\right)$ *under Assumptions 1 and 12. Assume without loss of generality that* $\mathbf{w}_0 = \mathbf{0}$*. Then for every* $t \geq 1$ *and* $\eta > 0$*, we have*

$$\mathcal{L}\left(\overline{\mathbf{w}}_t\right) \leq \ell\bigg(-\frac{\left(\gamma^2(t+1)\right)^2 - C_\ell}{4\gamma^2(t+1)}\eta\bigg), \quad \textit{where}\ \ \overline{\mathbf{w}}_t := \frac{1}{t+1}\sum_{k=0}^{t}\mathbf{w}_k.$$

In particular, after C_ℓ/γ^2 *burn-in steps, for every* $\eta > 0$*, we have*

$$\mathcal{L}(\overline{\mathbf{w}}_t) \leq \ell\left(-\frac{\gamma^2\eta}{4}\right), \quad t \geq \frac{C_\ell}{\gamma^2}.$$

The proof of Theorem 13 is deferred to Appendix E. We conclude this section by providing several examples of loss functions that satisfy Assumption 12. The proof is also deferred to Appendix E.

Example 1 *The following loss functions satisfy Assumption 12.*

1. *The logistic loss* $\ell_{\log} : z \mapsto \ln(1+\exp(-z))$ *and the exponential loss* $\ell_{\exp} : z \mapsto \exp(-z)$ *satisfy Assumption 12 with* $C_\ell = 1$.

2. *The polynomial loss of degree* $k > 0$ *(Ji and Telgarsky, 2021),*

$$\ell_{\text{poly}}(z) := \begin{cases} \dfrac{1}{(1+z)^k} & z \geq 0, \\ -2kz + \dfrac{1}{(1-z)^k} & z \leq 0, \end{cases}$$

   satisfies Assumption 12 with $C_\ell = n^{1/k}$.

3. *The semi-circle loss (Shen, 2005),*

$$\ell_{\text{semi}}(z) := \frac{-z + \sqrt{z^2+4}}{2},$$

   satisfies Assumption 12 with $C_\ell = n+1$.

5. Conclusion

We consider logistic regression on linearly separable data with margin γ. We show that GD with large and adaptive stepsizes achieves an arbitrarily small risk within $1/\gamma^2$ steps, although the risk evolution might not be monotonic. This is impossible if GD with adaptive stepsize induces a monotonically decreasing risk. Additionally, we establish batch and online first-order lower bounds of $\Omega(\min\{\ln n, 1/\gamma^2\})$ and $\Omega(\min\{n, 1/\gamma^2\})$, respectively, for finding a linear separator. This shows that GD with large and adaptive stepsizes matches the worst-case dependence and is minimax optimal when the sample size is unrestricted. Finally, our results extend to a broad class of loss functions and certain two-layer networks.

We conclude by discussing the following open questions.

Convergence of the Last Iterate. Note that Theorem 2 only controls the averaged iterate of GD with large and adaptive stepsizes. Although this implies the same guarantee for the best iterate, it is unclear whether or not a similar guarantee applies to the last iterate. This is an interesting question left for future research.

Optimal Tradeoff between n and $1/\gamma$. Consider the first-order batch methods for identifying a linear separator for a separable dataset (see Definition 7). The best-known upper bound on the step complexity is given by $\min\{1/\gamma^2, \mathcal{O}(\sqrt{\ln n}/\gamma)\}$, whereas the two branches are achieved by GD with large and adaptive stepsizes (Theorem 2) and momentum GD with adaptive stepsizes (Ji et al., 2021; Wang et al., 2022a), respectively. In comparison, we constructed hard examples suggesting the step complexity of first-order methods is at least $\Omega\big(\max\big\{\min\{\ln n, 1/\gamma^2\}, \min\{n, 1/\gamma^{2/3}\}\big\}\big)$, whereas the two branches are given by Theorem 8 and Theorem 18, respectively. We leave it as a future direction to close this gap for all n and $1/\gamma$.

Other Oracle Models. In the exact online first-order setting of Definition 9, the worst-case mistake complexity is characterized up to constants by Perceptron and Theorem 10. It would be interesting to understand how this picture changes under other oracle models, such as transductive online algorithms that know the unlabeled sequence in advance (Qian et al., 2026), relaxed-benchmark or covering-based online algorithms (Montasser et al., 2025), or stronger two-sided-oracle access for linear separability (Kornowski and Shamir, 2025), where qualitatively different dependence on n, $1/\gamma$, and the ambient dimension d may be possible.

Acknowledgments

We thank Yuhang Cai, Hossein Mobahi, and Matus Telgarsky for their helpful comments. We thank Qiuyu Ren for his significant help in improving the minimax lower bound. We thank Guy Kornowski and Ohad Shamir for pointing out missing references on the optimality of the Perceptron. We thank Nikita Zhivotovskiy for inspiring comments on other ideas for logistic regression with a linearly separable dataset and pointing to other related literature. We gratefully acknowledge the NSF's support of FODSI through grant DMS-2023505 and the NSF and the Simons Foundation for the Collaboration on the Theoretical Foundations of Deep Learning through awards DMS-2031883 and #814639 and the ONR's support through MURI award N000142112431.

Appendix A. Missing Parts in the Proof of Theorem 2 in Section 2.3

We first prove the 1-Lipschitzness of the transformed loss function $\phi(\cdot)$.

Lemma 14 *For the exponential loss and logistic loss, the transformed loss $\phi(\cdot)$ defined in* (3) *is 1-Lipschitz with respect to $\|\cdot\|$.*

Proof (of Lemma 14) Recall that $\phi(\mathbf{w}) := -\ell^{-1}(\mathcal{L}(\mathbf{w}))$ and $\|\mathbf{x}_i\| \le 1$ for $1 \le i \le n$. We have

$$\begin{aligned}\|\nabla\phi(\mathbf{w})\| &= \left\|\left(-\ell^{-1}\right)'(\mathcal{L}(\mathbf{w}))\cdot\frac{1}{n}\sum_{i=1}^{n}\ell'(y_i\mathbf{x}_i^\top\mathbf{w})y_i\mathbf{x}_i\right\| \\ &\le \frac{1}{n}\sum_{i=1}^{n}\left|\ell'(y_i\mathbf{x}_i^\top\mathbf{w})\right|\cdot\left|\left(\ell^{-1}\right)'(\mathcal{L}(\mathbf{w}))\right| := \star.\end{aligned}$$

Let $\ell_i = \ell\left(y_i\mathbf{x}_i^\top\mathbf{w}\right)$. We rearrange $\star$ as

$$\star = \frac{\frac{1}{n}\sum_{i=1}^{n}(-\ell')\left(\ell^{-1}(\ell_i)\right)}{(-\ell')\left(\ell^{-1}\left(\frac{1}{n}\sum_{i=1}^{n}\ell_i\right)\right)} = \frac{\frac{1}{n}\sum_{i=1}^{n}h(\ell_i)}{h\left(\frac{1}{n}\sum_{i=1}^{n}\ell_i\right)},$$

where $h(\cdot)$ is defined as

$$h(z) := \left(-\ell'\right)\left(\ell^{-1}(z)\right) = \begin{cases} z & \ell = \ell_{\mathsf{exp}}, \\ 1-\exp(-z) & \ell = \ell_{\mathsf{log}}.\end{cases}$$

For both losses, $h(\cdot)$ is concave on $z > 0$. Therefore, $\star \le 1$. ∎

Lemma 15 *Let $\phi(\cdot)$ be the transformed loss function defined in* (3)*. Under Assumption 1, if $\phi(\cdot)$ is C_ℓ-Lipschitz, we have*

$$2\left\langle\nabla\phi(\mathbf{w}),\mathbf{u}_2\right\rangle + \eta\left\|\nabla\phi(\mathbf{w})\right\|^2 \le 0, \quad \mathbf{u}_2 := (C_\ell\eta/(2\gamma))\mathbf{w}^*.$$

Proof (of Lemma 15) Recall that $\phi(\mathbf{w}) := -\ell^{-1}(\mathcal{L}(\mathbf{w}))$ and $\mathbf{u}_2 = (C_\ell\eta/(2\gamma))\mathbf{w}^*$. We also use the fact that $\langle\mathbf{w}^*, y_i\mathbf{x}_i\rangle \ge \gamma > 0$ and $\|\mathbf{x}_i\| \le 1$ for $1 \le i \le n$ from Assumption 1. Applying Lemma 14, we have

$$\begin{aligned}&2\left\langle\nabla\phi(\mathbf{w}),\mathbf{u}_2\right\rangle + \eta\left\|\nabla\phi(\mathbf{w})\right\|^2 \\ &\le \frac{2}{n}\left(-\ell^{-1}\right)'(\mathcal{L}(\mathbf{w}))\sum_{i=1}^{n}\ell'(y_i\mathbf{x}_i^\top\mathbf{w})\left\langle\mathbf{u}_2, y_i\mathbf{x}_i\right\rangle + C_\ell\eta\left\|\frac{\left(-\ell^{-1}\right)'(\mathcal{L}(\mathbf{w}))}{n}\sum_{i=1}^{n}\ell'(y_i\mathbf{x}_i^\top\mathbf{w})y_i\mathbf{x}_i\right\| \\ &\hspace{28em}(\phi(\cdot)\text{ is } C_\ell\text{-Lipschitz}) \\ &\le -\frac{2\gamma\left\|\mathbf{u}_2\right\|}{n}\left|\left(\ell^{-1}\right)'(\mathcal{L}(\mathbf{w}))\right|\sum_{i=1}^{n}\left|\ell'(y_i\mathbf{x}_i^\top\mathbf{w})\right| + \frac{C_\ell\eta}{n}\left(\left|\left(\ell^{-1}\right)'(\mathcal{L}(\mathbf{w}))\right|\sum_{i=1}^{n}\left|\ell'(y_i\mathbf{x}_i^\top\mathbf{w})\right|\right) \\ &= \frac{1}{n}\sum_{i=1}^{n}\left|\ell'(y_i\mathbf{x}_i^\top\mathbf{w})\right|\cdot\left|\left(\ell^{-1}\right)'(\mathcal{L}(\mathbf{w}))\right|\cdot\left[-2\gamma\left\|\mathbf{u}_2\right\| + C_\ell\eta\right].\end{aligned}$$

Invoking the definition of $\mathbf{u}_2$ completes the proof. ∎

The following lemma is a restatement of Lemma 5.2 in (Ji and Telgarsky, 2021).

Lemma 16 (Lemma 5.2 in (Ji and Telgarsky, 2021)) *Let $\ell(\cdot)$ be twice continuously differentiable, positive, decreasing, and convex. If $\frac{\ell'(t)^2}{\ell(t)\cdot\ell''(t)}$ is decreasing on $\mathbb{R}$, then $\psi(\mathbf{z}) := -\ell^{-1}(\frac{1}{n}\sum_{i=1}^{n}\ell(z_i))$ is convex. Moreover, $\phi(\cdot)$ is also convex since $\phi(\mathbf{w})$ is a composition between a linear mapping and $\psi(\cdot)$.*

Appendix B. Proof of Theorem 3

Before proving Theorem 3, let us first delve into a technical lemma. This lemma shows that for some special datasets, the stepsize must be small if the loss is monotonically decreasing.

Lemma 17 (A Variant of Lemma 14 in (Wu et al., 2024)) *Let* $\mathbf{w}_0 = \mathbf{0}, y_i = 1$ *for* $i = 1, 2, ..., n,$ *and* $\bar{\mathbf{x}} := (1/n) \cdot \sum_{i=1}^n \mathbf{x}_i$. *Assume there exist constants* $r > 0$ *and* $q \in (0, 1)$ *such that*

$$\frac{\left|\left\{i \in [n] : \mathbf{x}_i^\top \bar{\mathbf{x}} < -r\right\}\right|}{n} \geq q. \tag{11}$$

Suppose we run gradient descent according to (GD). *If* $\mathcal{L}(\mathbf{w}_1) \leq \mathcal{L}(\mathbf{w}_0)$, *then* $\eta \leq \ell(0)/(qr)$.

Proof (of Lemma 17) Starting from $\mathbf{w}_0 = \mathbf{0}$, we have $\mathcal{L}(\mathbf{w}_0) = \ell(0)$, $\nabla\mathcal{L}(\mathbf{w}_0) = \ell'(0) \cdot \bar{\mathbf{x}}$, $\nabla\phi(\mathbf{w}_0) = (-\ell^{-1})'(\ell(0)) \cdot \ell'(0) \cdot \bar{\mathbf{x}}$. For both the exponential loss and logistic loss, one can verify that $-(-\ell^{-1})'(\ell(0)) \cdot \ell'(0) = 1$, which implies $\mathbf{w}_1 = \eta\bar{\mathbf{x}}$. Suppose $\eta > qr/\ell(0)$. Then,

$$\mathcal{L}(\mathbf{w}_1) \geq \frac{1}{n}\sum_{i=1}^n \ell(\eta \mathbf{x}_i^\top \bar{\mathbf{x}})\mathbb{I}\left(\mathbf{x}_i^\top \bar{\mathbf{x}} < -r\right) \geq \ell(-\eta r) \cdot \frac{1}{n}\sum_{i=1}^n \mathbb{I}\left(\mathbf{x}_i^\top \bar{\mathbf{x}} < -r\right) \geq \ell(-\eta r) \cdot q \quad \text{(From (11))}$$

$$\geq q \cdot \ln\left(1 + \exp(\eta r)\right) \geq \eta q r \geq \ell(0) = \mathcal{L}(\mathbf{w}_0),$$

a contradiction to $\mathcal{L}(\mathbf{w}_1) \leq \mathcal{L}(\mathbf{w}_0)$. Thus $\eta \leq \ell(0)/(qr)$. ■

Now we prove the Theorem 3.

Proof (of Theorem 3) Consider the specific dataset from Theorem 3, which satisfies the conditions of Lemma 17 with $r = 0.1$ and $q = 0.5$. So Lemma 17 implies $\eta \leq c_1$, where c_1 is a constant that depends on the loss (but *not* on t). Next, from the GD iterate on the transformed loss (3) and the 1-Lipschitzness of $\phi(\cdot)$ (Lemma 14), we have

$$\|\mathbf{w}_t\| = \left\|\eta\sum_{k=0}^{t-1}\nabla\phi(\mathbf{w}_k)\right\| \leq \eta \cdot \sum_{k=0}^{t-1}\|\nabla\phi(\mathbf{w}_k)\| \leq \eta t.$$

Since $\|\mathbf{x}_i\| \leq 1$, it follows

$$\mathcal{L}(\mathbf{w}_t) = \frac{1}{n}\sum_{i=1}^n \ell(\mathbf{w}_t^\top \mathbf{x}_i) \geq \ell\left(c_1 \cdot t\right) \simeq \exp\left(-c_1 \cdot t\right)$$

The same argument applies to $\mathcal{L}(\overline{\mathbf{w}}_t)$. This completes the proof. ■

Appendix C. Missing Proofs for Theorems in Section 3

C.1 Proof of Theorem 8

Proof (of Theorem 8) For $0 < \gamma < 1/6$ and $n > 16$, we define $d := \lfloor 1/5\gamma^2 \rfloor \geq 6$, where $\lfloor\cdot\rfloor$ is the floor function. Let $(\mathbf{e}_i)_{i=1}^d$ be a set of standard basis vectors for $\mathbb{R}^d$. Note that all the first-order methods defined in Definition 7 are rotation invariant. Moreover, the criterion for all data classified correctly is also rotation invariant. Therefore, without loss of generality, we can assume $\mathbf{w}_0$ is proportional to $\mathbf{e}_1$, that is, $\mathbf{w}_0 \in \mathsf{Lin}\{\mathbf{e}_1\}$.

Let $k := \min\{\lfloor \log_2 n \rfloor, d-2\} \geq 4$. We construct a hard dataset $(\mathbf{x}_i, y_i)_{i=1}^n$ as follows. Let $y_i = 1$ for all $i \in [n]$. For $j = 1, \ldots, k$, let

$$\mathbf{x}_i := \frac{2}{\sqrt{5}}\mathbf{e}_{j+1} - \frac{1}{\sqrt{5}}\mathbf{e}_{j+2} \quad \text{for} \;\; 2^k - 2^{k-j+1} + 1 \leq i \leq 2^k - 2^{k-j}.$$

Note that $j \leq k \leq d-2$, thus such $\mathbf{x}_i$'s are well defined. Let the remaining $\mathbf{x}_i$'s be

$$\mathbf{x}_i := \frac{1}{\sqrt{5}}\mathbf{e}_{k+2} \quad \text{for} \;\; 2^k \leq i \leq n.$$

Note that $\|\mathbf{x}_i\| \leq 1$ for $i = 1, \ldots, n$. Moreover, for the unit vector

$$\mathbf{w}^* := \frac{1}{\sqrt{d}}(1, 1, \ldots, 1)^\top,$$

we have

$$y_i \mathbf{x}_i^\top \mathbf{w}^* = \frac{1}{\sqrt{5d}} \geq \gamma, \quad \text{for } i = 1 \ldots, n.$$

Thus $(\mathbf{x}_i, y_i)_{i=1}^n$ satisfies Assumption 1.

For a vector $\mathbf{w} \in \mathbb{R}^d$, we also write it as $\mathbf{w} := (w^{(1)}, w^{(2)}, \ldots, w^{(d)})^\top$. Then, the objective function can be written as

$$\begin{aligned}
\mathcal{L}(\mathbf{w}) &= \frac{1}{n}\sum_{i=1}^n \ell\big(\mathbf{w}^\top \mathbf{x}_i\big) \\
&= \frac{1}{n}\left[\sum_{j=1}^k 2^{k-j}\ell\Big(\frac{2}{\sqrt{5}}w^{(j+1)} - \frac{1}{\sqrt{5}}w^{(j+2)}\Big) + \big(n - 2^k + 1\big)\ell\Big(\frac{1}{\sqrt{5}}w^{(k+2)}\Big)\right].
\end{aligned}$$

Thus the j-th coordinate of $\nabla\mathcal{L}(\mathbf{w})$ is given by

$$\big[\nabla\mathcal{L}(\mathbf{w})\big]_j = \begin{cases}
0 & j = 1, \\
\dfrac{2^k}{\sqrt{5}n}\ell'\Big(\dfrac{2w^{(2)}}{\sqrt{5}} - \dfrac{w^{(3)}}{\sqrt{5}}\Big) & j = 2, \\
\dfrac{2^{k-j+2}}{\sqrt{5}n}\left[\ell'\Big(\dfrac{2w^{(j)}}{\sqrt{5}} - \dfrac{w^{(j+1)}}{\sqrt{5}}\Big) - \ell'\Big(\dfrac{2w^{(j-1)}}{\sqrt{5}} - \dfrac{w^{(j)}}{\sqrt{5}}\Big)\right] & 3 \leq j \leq k+1, \\
\dfrac{1}{\sqrt{5}n}\left[\big(n - 2^k + 1\big)\ell'\Big(\dfrac{w^{(k+2)}}{\sqrt{5}}\Big) - \ell'\Big(\dfrac{2w^{(k+1)}}{\sqrt{5}} - \dfrac{w^{(k+2)}}{\sqrt{5}}\Big)\right] & j = k+2, \\
0 & \text{otherwise.}
\end{cases}$$

Consider a sequence $(\mathbf{w}_s)_{s=0}^t$ generated by a first-order method according to Definition 7. Since $\mathbf{w}_0 \in \mathsf{Lin}\{\mathbf{e}_1\}$, the gradient at $\mathbf{w}_0$ vanishes in all coordinates except the second and the $(k+2)$-th coordinates. Therefore we have

$$\mathbf{w}_1 \in \mathsf{Lin}\{\mathbf{e}_1, \, \mathbf{e}_2, \, \mathbf{e}_{k+2}\}.$$

Then the gradient at $\mathbf{w}_1$ vanishes in all coordinates except the second, third, $(k+1)$-th, and $(k+2)$-th coordinates. Therefore we have

$$\mathbf{w}_2 \in \mathsf{Lin}\{\mathbf{e}_1,\, \mathbf{e}_2,\, \mathbf{e}_3,\, \mathbf{e}_{k+1},\, \mathbf{e}_{k+2}\}.$$

By induction, we conclude that for $t \le t_0 - 2$ for $t_0 := \lfloor (k+1)/2 \rfloor$, it holds that

$$\mathbf{w}_t \in \mathsf{Lin}\{\mathbf{e}_1, \dots, \mathbf{e}_{t+1}, \mathbf{e}_{k+3-t}, \dots, \mathbf{e}_{k+2}\}.$$

So for all $(\mathbf{w}_s)_{s=0}^t$, their t_0-th and (t_0+1)-th coordinates must be zero. By our dataset construction, there exists $i \le 2^k - 1$ such that

$$y_i \mathbf{x}_i^\top \mathbf{w}_k = 0, \quad \text{for all } k = 0, \dots, t.$$

This means that the dataset cannot be separated by any of $(\mathbf{w}_k)_{k=0}^t$. Thus, for the first-order method to output a linear separator, we must have

$$\begin{aligned} t \ge t_0 - 1 = \left\lfloor \frac{k-1}{2} \right\rfloor \ge \frac{k}{2} - 1 \ge \min\left\{ \frac{\log_2 n - 3}{2}, \frac{d}{2} - 2 \right\} \ge \min\left\{ \frac{\log_2 n}{8}, \frac{d}{3} \right\} \\ \ge \min\left\{ \frac{\log_2 n}{8}, \frac{1}{15\gamma^2} - \frac{1}{3} \right\} \ge \min\left\{ \frac{\ln n}{8 \ln 2}, \frac{1}{30\gamma^2} \right\}. \end{aligned}$$

This completes the proof. ■

C.2 Proof of Theorem 10

Proof (of Theorem 10) For $0 < \gamma < 1/2$ and $n \ge 2$, we define $d := \lfloor 1/\gamma^2 \rfloor \ge 4$, where $\lfloor \cdot \rfloor$ is the floor function. Let $(\mathbf{e}_i)_{i=1}^d$ be a set of standard basis vectors for $\mathbb{R}^d$. Note that all the first-order online methods defined in Definition 9 are rotation invariant. Moreover, the criterion for all data classified correctly is also rotation invariant. Therefore, without loss of generality, we can assume $\mathbf{w}_0$ is proportional to $\mathbf{e}_1$, that is, $\mathbf{w}_0 \in \mathsf{Lin}\{\mathbf{e}_1\}$.

Let $k := \min\{n, d-1\} \ge 2$. We construct a hard dataset $(\mathbf{x}_i, y_i)_{i=1}^n$ as follows. Let $y_i = 1$ for all $i \in [n]$. For i = 1,2,...,k, let

$$\mathbf{x}_i = \mathbf{e}_{i+1}.$$

Note that $k \le d-1$, thus such $\mathbf{x}_i$'s are well defined. Let the remaining $\mathbf{x}_i$'s be

$$\mathbf{x}_i = \mathbf{e}_{k+1} \;\text{ for }\; k+1 \le i \le n \;\text{ if }\; n \ge k+1.$$

Moreover, for the unit vector

$$\mathbf{w}^* := \frac{1}{\sqrt{d}} (1, 1, \dots, 1)^\top,$$

we have

$$y_i \mathbf{x}_i^\top \mathbf{w}^* = \frac{1}{\sqrt{d}} \ge \gamma, \quad \text{for } i = 1 \dots, n.$$

Thus $(\mathbf{x}_i, y_i)_{i=1}^n$ satisfies Assumption 1.

Consider a sequence $(\mathbf{w}_s)_{s=0}^{t}$ generated by a first-order online method with initialization $\mathbf{w}_0$ and dataset $\big(\mathbf{x}_{\pi(s)}, y_{\pi(s)}\big)_{s=1}^{t}$, where $\pi(s) \in [n]$ but is otherwise arbitrary. Since $\mathbf{w}_0 \in \mathsf{Lin}\{\mathbf{e}_1\}$, the gradient at $\mathbf{w}_0$ vanishes in all coordinates except the direction of $\ell'(y_{\pi(1)}\mathbf{x}_{\pi(1)}^\top \mathbf{w}_0) y_{\pi(1)}\mathbf{x}_{\pi(1)}$. Therefore we have

$$\mathbf{w}_1 \in \mathsf{Lin}\{\mathbf{e}_1,\, \mathbf{x}_{\pi(1)}\}.$$

Then the gradient at $\mathbf{w}_1$ vanishes in all coordinates except the span of

$$\ell'(y_{\pi(1)}\mathbf{x}_{\pi(1)}^\top \mathbf{w}_0) y_{\pi(1)}\mathbf{x}_{\pi(1)} \quad \text{and} \quad \ell'(y_{\pi(2)}\mathbf{x}_{\pi(2)}^\top \mathbf{w}_0) y_{\pi(2)}\mathbf{x}_{\pi(2)}.$$

Therefore we have

$$\mathbf{w}_2 \in \mathsf{Lin}\{\mathbf{e}_1,\, \mathbf{x}_{\pi(1)}, \mathbf{x}_{\pi(2)}\}.$$

By induction, we conclude that for all $t \le k-1$, it holds that

$$\mathbf{w}_t \in \mathsf{Lin}\{\mathbf{e}_1, \mathbf{x}_{\pi(1)}, \dots, \mathbf{x}_{\pi(t)}\}.$$

From our dataset construction, there exists $j \in [d]$, such that for all $(\mathbf{w}_s)_{s=0}^{t}$, their j-th coordinates are zero. Therefore,

$$y_j \mathbf{x}_j^\top \mathbf{w}_s = 0, \quad \text{for all } s = 0, \dots, t.$$

This means that the dataset cannot be separated by any of $(\mathbf{w}_s)_{s=0}^{t}$. Thus, for the first-order online method to output a linear separator, we must have

$$t \ge k = \min\{n, d-1\} \ge \min\left\{\frac{1}{2\gamma^2}, n\right\}.$$

This proves the first claim.

For the second claim, consider a sequence $(\mathbf{w}_s)_{s=0}^{t}$ generated by a first-order online method with initialization $\mathbf{w}_0$ and dataset $(\mathbf{x}_i, y_i)_{i=1}^{n}$ as defined previously. By the same induction, we conclude that for all $0 \le t \le k-1$, it holds that

$$\mathbf{w}_t \in \mathsf{Lin}\{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_{t+1}\}.$$

Since $\mathbf{x}_{t+1} = \mathbf{e}_{t+2}$ for $0 \le t \le k-1$, this implies

$$y_{t+1}\mathbf{x}_{t+1}^\top \mathbf{w}_t = 0,$$

which suggests that the algorithm makes a mistake step. Therefore, the total mistake steps up to the t-th step is

$$\sum_{j=0}^{t-1} \mathbb{1}\{y_{j+1}\mathbf{x}_{j+1}^\top \mathbf{w}_j \le 0\} \ge \min\{t, k\} = \min\{t, d-1\} \ge \min\left\{\frac{1}{2\gamma^2}, t\right\}.$$

This completes the proof. ■

C.3 An Alternative Lower Bound for First-Order Batch Algorithms

Now we present another lower bound for first-order batch algorithms with a different trade-off than Theorem 8. We leave it as an open problem to figure out the optimal trade-off between the sample size n and margin γ in finding linear separators.

Theorem 18 (An Alternate Lower Bound for Batch First-Order Methods) *For every* $0 < \gamma < 1/8$*,* $n \ge 4$*, and* $\mathbf{w}_0$*, there exists a dataset* $(\mathbf{x}_i, y_i)_{i=1}^n$ *satisfying Assumption 1 such that the following holds. For any* $\mathbf{w}_t$ *output by a first-order batch method in t-steps with initialization* $\mathbf{w}_0$ *on this dataset, we have*

$$\min_{i\in[n]} y_i \mathbf{x}_i^\top \mathbf{w}_t > 0 \quad \textit{implies that} \quad t \ge \min\left\{\frac{n}{4}, \frac{1}{8\gamma^{2/3}}\right\}.$$

Proof (of Theorem 18) For $0 < \gamma < 1/8$ and $n \ge 2$, we define $d := \lfloor \gamma^{-2/3} \rfloor \ge 4$, where $\lfloor \cdot \rfloor$ is the floor function. Let $(\mathbf{e}_i)_{i=1}^d$ be a set of standard basis vectors for $\mathbb{R}^d$. Note that all the first-order methods defined in Definition 7 are rotation invariant. Moreover, the criterion for all data classified correctly is also rotation invariant. Therefore, without loss of generality, we can assume $\mathbf{w}_0$ is proportional to $\mathbf{e}_1$, that is, $\mathbf{w}_0 \in \mathsf{Lin}\{\mathbf{e}_1\}$.

Let $k := \min\{n, d-2\}$. We construct a hard dataset $(\mathbf{x}_i, y_i)_{i=1}^n$ as follows. Let $y_i = 1$ for all $i \in [n]$. For $j = 1, \dots, k$, let

$$\mathbf{x}_i := -\frac{1}{\sqrt{2}}\mathbf{e}_{j+1} + \frac{1}{\sqrt{2}}\mathbf{e}_{j+2}.$$

Let the remaining $\mathbf{x}_i$'s be

$$\mathbf{x}_i := \frac{1}{\sqrt{2}}\mathbf{e}_{k+2} \quad \text{for} \quad k+1 \le i \le n \quad \text{if} \quad n \ge k+1.$$

Note that $j \le k \le d-2$, thus such $\mathbf{x}_i$'s are well defined. Also note that $\|\mathbf{x}_i\| \le 1$ for $i = 1, \dots, n$. Moreover, for the unit vector

$$\mathbf{w}^* := \sqrt{\frac{6}{d(d+1)(2d+1)}}\,(1, 2, \dots, d)^\top,$$

we have

$$y_i \mathbf{x}_i^\top \mathbf{w}^* = \sqrt{\frac{3}{d(d+1)(2d+1)}} \ge \sqrt{\frac{1}{d^3}} \ge \gamma, \quad \text{for} \quad i = 1 \dots, n.$$

Thus $(\mathbf{x}_i, y_i)_{i=1}^n$ satisfies Assumption 1.

For a vector $\mathbf{w} \in \mathbb{R}^d$, we also write it as $\mathbf{w} := (w^{(1)}, w^{(2)}, \dots, w^{(d)})^\top$. Then, the objective function can be written as

$$\mathcal{L}(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^n \ell\big(\mathbf{w}^\top \mathbf{x}_i\big) = \frac{1}{n}\left[\sum_{j=1}^k \ell\bigg(-\frac{1}{\sqrt{2}}w^{(j+1)} + \frac{1}{\sqrt{2}}w^{(j+2)}\bigg) + (n-k)\ell\bigg(\frac{1}{\sqrt{2}}w^{(k+2)}\bigg)\right].$$

Thus the j-th coordinate of $\nabla\mathcal{L}(\mathbf{w})$ is given by

$$\big[\nabla\mathcal{L}(\mathbf{w})\big]_j = \begin{cases} 0 & j=1, \\ -\dfrac{1}{\sqrt{2}n}\ell'\Big(-\dfrac{w^{(2)}}{\sqrt{2}}+\dfrac{w^{(3)}}{\sqrt{2}}\Big) & j=2, \\ \dfrac{1}{\sqrt{2}n}\Big[\ell'\Big(-\dfrac{w^{(j-1)}}{\sqrt{2}}+\dfrac{w^{(j)}}{\sqrt{2}}\Big)-\ell'\Big(-\dfrac{w^{(j)}}{\sqrt{2}}+\dfrac{w^{(j+1)}}{\sqrt{2}}\Big)\Big] & 3\le j\le k+1, \\ \dfrac{1}{\sqrt{2}n}\Big[(n-k)\ell'\Big(\dfrac{w^{(k+2)}}{\sqrt{2}}\Big)-\ell'\Big(-\dfrac{w^{(k+1)}}{\sqrt{2}}+\dfrac{w^{(k+2)}}{\sqrt{2}}\Big)\Big] & j=k+2, \\ 0 & \text{otherwise.} \end{cases}$$

Consider a sequence $(\mathbf{w}_s)_{s=0}^t$ generated by a first-order method according to Definition 7. Since $\mathbf{w}_0 \in \mathsf{Lin}\{\mathbf{e}_1\}$, the gradient at $\mathbf{w}_0$ vanishes in all coordinates except the second and the $(k+2)$-th coordinates. Therefore we have

$$\mathbf{w}_1 \in \mathsf{Lin}\{\mathbf{e}_1,\, \mathbf{e}_2,\, \mathbf{e}_{k+2}\}.$$

Then the gradient at $\mathbf{w}_1$ vanishes in all coordinates except the second, third, $(k+1)$-th, and $(k+2)$-th coordinates. Therefore we have

$$\mathbf{w}_2 \in \mathsf{Lin}\{\mathbf{e}_1,\, \mathbf{e}_2,\, \mathbf{e}_3,\, \mathbf{e}_{k-1},\, \mathbf{e}_{k+2}\}.$$

By induction, we conclude that for $t \le t_0 - 2$ with $t_0 := \lfloor (k+1)/2 \rfloor$, it holds that

$$\mathbf{w}_t \in \mathsf{Lin}\{\mathbf{e}_1, \dots, \mathbf{e}_{t+1}, \mathbf{e}_{k+3-t}, \dots, \mathbf{e}_{k+2}\}.$$

So for all $(\mathbf{w}_s)_{s=0}^t$, their t_0-th and (t_0+1)-th coordinates must be zero. By our dataset construction, there exists $i \in [k]$ such that

$$y_i \mathbf{x}_i^\top \mathbf{w}_k = 0, \quad \text{for all } k = 0, \dots, t.$$

This means that the dataset cannot be separated by any of $(\mathbf{w}_k)_{k=0}^t$. Thus, for the first-order method to output a linear separator, we must have

$$\begin{aligned} t \ge t_0 - 1 = \left\lfloor \frac{k-1}{2} \right\rfloor \ge \frac{k}{2} - 1 \ge \min\left\{\frac{n}{2}-1, \frac{d}{2}-2\right\} \ge \min\left\{\frac{n}{4}, \frac{d}{4}\right\} \\ \ge \min\left\{\frac{n}{4}, \frac{1}{4}\big(\gamma^{-2/3}-1\big)\right\} \ge \min\left\{\frac{n}{4}, \frac{1}{8\gamma^{2/3}}\right\} \end{aligned}$$

This completes the proof. ■

Appendix D. A Generalization of Theorem 11 and Its Proof

In this section, we generalize Theorem 11 to more general activation functions and present its proof. First, we provide general assumptions on the activation functions.

Assumption 19 (Activation Function) *In two-layer neural networks* (9)*, let the activation function* $\sigma(z) : \mathbb{R} \to \mathbb{R}$ *be a locally Lipschitz function. For each* z*, let* $\sigma'(z)$ *be a unique element from the Clarke subdifferential of* $\sigma(\cdot)$ *at* z *(Clarke, 1990). Moreover, we assume*

- *There exists* $0 < \alpha < 1$ *such that* $\sigma'(z) \in [\alpha, 1]$.
- *For any* $z \in \mathbb{R}$*, it holds that* $|\sigma(z) - \sigma'(z)z| \le \kappa$ *for some* $\kappa > 0$.

Examples of Activation Functions. Assumption 19 holds for a broad class of leaky activations. For instance, let $\sigma_*(\cdot)$ be one of the following: GeLU $\sigma_{\mathsf{GeLU}}(x) := x \cdot F(x)$, where $F(x)$ is the cumulative density function of the standard Gaussian distribution; Softplus $\sigma_{\mathsf{Softplus}}(x) := \ln(1 + \exp(x))$; SiLU $\sigma_{\mathsf{SiLU}}(x) := x/(1 + \exp(-x))$; ReLU $\sigma_{\mathsf{ReLU}}(x) := \max\{x, 0\}$. Then, we fix some constant $0.5 < c < 1$ and define $\sigma(x) = c \cdot x + \frac{1-c}{4} \cdot \sigma_*(x)$, where σ_* can be any activation function above. It is straightforward to check that such a "leaky" variant satisfies Assumption 19 (see Example 2.1 in Cai et al. (2024)). Moreover, the leaky ReLU activation defined as $\sigma(z) = \max\{z, \alpha z\}$ satisfies Assumption 19 with α and $\kappa = 0$.

Now, we present the improved convergence rate for two-layer neural networks. Setting $\kappa = 0$ for leaky ReLU function recovers Theorem 11.

Theorem 20 (General Two-Layer Neural Networks) *Consider* (GD) *on* (10) *with adaptive stepsizes* (2) *for two-layer neural networks* (9) *under Assumption 1 and Assumption 19. Assume without loss of generality that* $\mathbf{w}_0 = \mathbf{0}$. *Then for* every $t \ge 1$ *and* $\eta > 0$*, we have*

$$\min_{k \le t} \mathcal{L}(\mathbf{w}_k) \le \exp\left(\kappa - \frac{(\alpha\gamma^2(t+1))^2 - 1}{4\gamma^2(t+1)}\eta\right).$$

In particular, after $1/(\alpha\gamma^2)$ *burn-in steps, for every* $\eta > 0$*, we have*

$$\min_{k \le t} \mathcal{L}(\mathbf{w}_k) \le \exp\left(\kappa - \frac{\alpha^2\gamma^2\eta}{4}\right) = \exp\left(-\Theta\left(\eta\right)\right), \quad t \ge \frac{1}{\alpha\gamma^2}. \tag{12}$$

Before we prove Theorem 20, we first state and prove two lemmas.

Lemma 21 *Under Assumption 1, for every* $\mathbf{w} \in \mathbb{R}^{md}$*, it holds that*

$$I_2(\mathbf{w}) := 2\left\langle m \cdot \nabla\phi(\mathbf{w}), \mathbf{u}_2\right\rangle + \eta\left\|m \cdot \nabla\phi(\mathbf{w})\right\|^2 \le 0 \;\; \textit{for} \;\; \mathbf{u}_2^{(j)} := \frac{a_j\eta}{2\gamma}\mathbf{w}^*, \; j = 1, 2, ...m.$$

Proof (of Lemma 21) Define $g_{i,j} := \ell'(y_i f(\mathbf{w}, \mathbf{x}_i)) \cdot \sigma'(\mathbf{x}_i^\top \mathbf{w}^{(j)}) \le 0$ for $i \in [n]$ and $j \in [m]$. We know $|g_{i,j}| \le |\ell'(y_i f(\mathbf{w}, \mathbf{x}_i))|$ since $\sigma'(\cdot) \le 1$. Then, for $j \in [m]$, we have

$$\begin{aligned}\|m \cdot \nabla_{\mathbf{w}^{(j)}}\phi(\mathbf{w})\| &= \left\|\left(-\ell^{-1}\right)'(\mathcal{L}(\mathbf{w})) \cdot \frac{1}{n}\sum_{i=1}^n a_j g_{i,j} y_i \mathbf{x}_i\right\| \\ &\le \left|\left(-\ell^{-1}\right)'(\mathcal{L}(\mathbf{w}))\right| \cdot \frac{1}{n}\sum_{i=1}^n |\ell'(y_i f(\mathbf{w}, \mathbf{x}_i))|.\end{aligned}$$

Let $\ell_i := \ell(y_i f(\mathbf{w}, \mathbf{x}_i))$. We arrange the term above as

$$\left|\left(-\ell^{-1}\right)'(\mathcal{L}(\mathbf{w}))\right| \cdot \frac{1}{n}\sum_{i=1}^{n}|\ell'(y_i f(\mathbf{w}, \mathbf{x}_i))| = \frac{\frac{1}{n}\sum_{i=1}^{n}(-\ell')\left(\ell^{-1}(\ell_i)\right)}{(-\ell')\left(\ell^{-1}\left(\frac{1}{n}\sum_{i=1}^{n}\ell_i\right)\right)} = \frac{\frac{1}{n}\sum_{i=1}^{n}h(\ell_i)}{h\left(\frac{1}{n}\sum_{i=1}^{n}\ell_i\right)},$$

where $h(\cdot)$ is defined as $h(z) := (-\ell')\left(\ell^{-1}(z)\right)$. For both losses, $h(\cdot)$ is concave on $z > 0$ (see Lemma 14 in Appendix A). Therefore, we have $\|m\nabla_{\mathbf{w}^{(j)}}\phi(\mathbf{w})\|^2 \le 1$ for $j \in [m]$. Apply this upper bound of $\|\nabla\phi(\mathbf{w})\|$, and recall Assumption 1, we have

$$\begin{aligned}
I_2(\mathbf{w}) &= \frac{2}{n}\left(-\ell^{-1}\right)'(\mathcal{L}(\mathbf{w})) \cdot \sum_{j=1}^{m}\sum_{i=1}^{n} g_{i,j} \cdot \left\|\mathbf{u}_2^{(j)}\right\| \cdot y_i \mathbf{x}_i^\top \mathbf{w}^* + \eta \cdot \sum_{j=1}^{m}\|m\nabla_{\mathbf{w}^{(j)}}\phi(\mathbf{w})\|^2 \\
&\le \frac{2}{n}\left(-\ell^{-1}\right)'(\mathcal{L}(\mathbf{w})) \cdot \sum_{j=1}^{m}\sum_{i=1}^{n} g_{i,j} \cdot \left\|\mathbf{u}_2^{(j)}\right\| \cdot y_i \mathbf{x}_i^\top \mathbf{w}^* + \eta \cdot \sum_{j=1}^{m}\|m\nabla_{\mathbf{w}^{(j)}}\phi(\mathbf{w})\| \\
&\le \frac{-2\gamma}{n}\left(-\ell^{-1}\right)'(\mathcal{L}(\mathbf{w})) \cdot \sum_{j=1}^{m}\sum_{i=1}^{n}|g_{i,j}| \cdot \left\|\mathbf{u}_2^{(j)}\right\| + \eta\left(-\ell^{-1}\right)'(\mathcal{L}(\mathbf{w})) \cdot \frac{1}{n}\sum_{j=1}^{m}\sum_{i=1}^{n}|g_{i,j}| \\
&= \left(-\ell^{-1}\right)'(\mathcal{L}(\mathbf{w})) \cdot \frac{1}{n}\sum_{j=1}^{m}\sum_{i=1}^{n}|g_{i,j}| \cdot \left(-2\gamma\left\|\mathbf{u}_2^{(j)}\right\| + \eta\right).
\end{aligned}$$

Invoking the definition of $\mathbf{u}_2$, we complete the proof. ■

Lemma 22 *Under Assumption 1, if* $\mathbf{u}_1^{(j)} \propto a_j \cdot \mathbf{w}^*$ *for* $j = 1, 2, ..., m$, *then for any* $\mathbf{w} \in \mathbb{R}^{md}$,

$$I_1(\mathbf{w}) := \langle\nabla\phi(\mathbf{w}), \mathbf{u}_1 - \mathbf{w}\rangle \le \kappa - \frac{\alpha\gamma}{m}\sum_{j=1}^{m}\left\|\mathbf{u}_1^{(j)}\right\| - \phi(\mathbf{w}).$$

Proof (of Lemma 22) From the definition of the transformed loss function $\phi(\cdot)$, we have

$$\begin{aligned}
I_1(\mathbf{w}) &= \left(-\ell^{-1}\right)'(\mathcal{L}(\mathbf{w})) \cdot \frac{1}{n}\sum_{i=1}^{n}\ell'(y_i f(\mathbf{w}; \mathbf{x}_i)) \cdot \frac{1}{m}\sum_{j=1}^{m} y_i a_j \sigma'\left(\mathbf{x}_i^\top \mathbf{w}^{(j)}\right) \cdot \mathbf{x}_i^\top\left(\mathbf{u}_1^{(j)} - \mathbf{w}^{(j)}\right) \\
&= \left(-\ell^{-1}\right)'(\mathcal{L}(\mathbf{w})) \cdot \frac{1}{n}\sum_{i=1}^{n}\ell'(y_i f(\mathbf{w}; \mathbf{x}_i)) \cdot \left[J_i - y_i f(\mathbf{w}, \mathbf{x}_i)\right],
\end{aligned}$$

where

$$\begin{aligned}
J_i &:= \frac{1}{m}\sum_{j=1}^{m} a_j\left[\sigma'\left(\mathbf{x}_i^\top\mathbf{w}^{(j)}\right) y_i \mathbf{x}_i^\top \mathbf{u}_1^{(j)}\right] + \frac{1}{m}\sum_{j=1}^{m} y_i a_j \underbrace{\left[\sigma\left(\mathbf{x}_i^\top\mathbf{w}^{(j)}\right) - \sigma'\left(\mathbf{x}_i^\top\mathbf{w}^{(j)}\right)\mathbf{x}_i^\top\mathbf{w}^{(j)}\right]}_{|\cdot|\le\kappa} \\
&\ge \frac{1}{m}\sum_{j=1}^{m} a_j\left[\sigma'\left(\mathbf{x}_i^\top\mathbf{w}^{(j)}\right) y_i \mathbf{x}_i^\top \mathbf{u}_1^{(j)}\right] - \kappa \ge \frac{\alpha\gamma}{m}\sum_{j=1}^{m}\left\|\mathbf{u}_1^{(j)}\right\| - \kappa,
\end{aligned}$$

where the last inequality utilizes $\sigma'(\cdot) \ge \alpha$, $\mathbf{u}_1^{(j)} \propto a_j \mathbf{w}^*$, and Assumption 1. Notice $\ell'(\cdot) \le 0$, we define $\mathbf{z} := (y_1 f(\mathbf{w}; \mathbf{x}_1), y_2 f(\mathbf{w}, \mathbf{x}_2), ..., y_n f(\mathbf{w}, \mathbf{x}_n))^\top$ and $\mathbf{1}_n = (1, 1, ..., 1)^\top \in \mathbb{R}^n$. We also define $\psi(\mathbf{z}) = (-\ell^{-1})(1/n \cdot \sum_{i=1}^{n} \ell(\mathbf{z}_i))$. From the definition, we know $\phi(\mathbf{w}) = \psi(\mathbf{z})$ and $\psi(\cdot)$ is convex for both exponential and logistic loss (Theorem 5.1 in (Ji and Telgarsky, 2021), also see Lemma 16). Therefore,

$$\begin{aligned} I_1(\mathbf{w}) &\leq \left(-\ell^{-1}\right)' (\mathcal{L}(\mathbf{w})) \cdot \frac{1}{n} \sum_{i=1}^{n} \ell' \left(y_i f\left(\mathbf{w}; \mathbf{x}_i\right)\right) \cdot \left(\frac{\alpha\gamma}{m} \sum_{j=1}^{m} \left\| \mathbf{u}_1^{(j)} \right\| - \kappa - y_i f\left(\mathbf{w}, \mathbf{x}_i\right) \right) \\ &= \left\langle \nabla \psi\left(\mathbf{z}\right), \left(\frac{\alpha\gamma}{m} \sum_{j=1}^{m} \left\| \mathbf{u}_1^{(j)} \right\| - \kappa \right) \cdot \mathbf{1}_n - \mathbf{z} \right\rangle \\ &\leq \psi\left(\left(\frac{\alpha\gamma}{m} \sum_{j=1}^{m} \left\| \mathbf{u}_1^{(j)} \right\| - \kappa \right) \cdot \mathbf{1}_n \right) - \psi\left(\mathbf{z}\right) \\ &= \kappa - \frac{\alpha\gamma}{m} \sum_{j=1}^{m} \left\| \mathbf{u}_1^{(j)} \right\| - \phi\left(\mathbf{w}\right). \end{aligned}$$

This completes the proof. ■

Now we present the proof of Theorem 20

Proof (of Theorem 20) As explained in (3), it is equivalent to considering GD with a constant stepsize under a transformed objective $\phi(\cdot)$. We then use the split optimization technique developed by Wu et al. (2024) and Cai et al. (2024). Specifically, consider a comparator $\mathbf{u} = \mathbf{u}_1 + \mathbf{u}_2 \in \mathbb{R}^{dm}$:

$$\mathbf{u}_1 = \begin{pmatrix} \mathbf{u}_1^{(1)} \\ \mathbf{u}_1^{(2)} \\ \cdots \\ \mathbf{u}_1^{(m)} \end{pmatrix}, \quad \mathbf{u}_2 = \begin{pmatrix} \mathbf{u}_2^{(1)} \\ \mathbf{u}_2^{(2)} \\ \cdots \\ \mathbf{u}_2^{(m)} \end{pmatrix}.$$

From GD iterate in (3), we have

$$\begin{aligned} & \left\| \mathbf{w}_{t+1} - \mathbf{u} \right\|^2 \\ = \ & \left\| \mathbf{w}_t - \mathbf{u} \right\|^2 + 2\eta m \left\langle \nabla\phi(\mathbf{w}_t), \mathbf{u}_1 - \mathbf{w}_t \right\rangle + \eta \Big(2 \left\langle m \cdot \nabla\phi(\mathbf{w}_t), \mathbf{u}_2 \right\rangle + \eta \left\| m \cdot \nabla\phi(\mathbf{w}_t) \right\|^2 \Big) \\ \leq \ & \left\| \mathbf{w}_t - \mathbf{u} \right\|^2 + 2\eta m \bigg(\kappa - \frac{\alpha\gamma}{m} \sum_{j=1}^{m} \left\| \mathbf{u}_1^{(j)} \right\| - \phi\left(\mathbf{w}_t\right) \bigg), \end{aligned} \tag{13}$$

where (13) is by the following two lemmas. Rearranging (13) and telescoping the sum to obtain

$$\frac{\left\| \mathbf{w}_t - \mathbf{u} \right\|^2}{2\eta m(t+1)} + \frac{1}{t+1} \sum_{k=0}^{t} \phi(\mathbf{w}_k) \leq \kappa - \frac{\alpha\gamma}{m} \sum_{j=1}^{m} \left\| \mathbf{u}_1^{(j)} \right\| + \frac{\left\| \mathbf{u} \right\|^2}{2\eta m(t+1)}. \tag{14}$$

Further setting $\|\mathbf{u}_1^{(j)}\| = \alpha\gamma\eta(t+1)/2$, we get

$$\frac{1}{t+1} \sum_{k=0}^{t} \phi(\mathbf{w}_k) \leq \kappa - \frac{\alpha\gamma}{m} \sum_{j=1}^{m} \left\| \mathbf{u}_1^{(j)} \right\| + \frac{\left\| \mathbf{u}_1 + \mathbf{u}_2 \right\|^2}{2\eta m(t+1)} \leq \kappa - \frac{(\alpha\gamma^2(t+1))^2 - 1}{4\gamma^2(t+1)} \eta. \tag{15}$$

We complete the proof by applying the fact that $\mathcal{L}(\cdot) = \ell\left(-\phi(\cdot)\right).$ ■

Appendix E. Proofs for Theorem 13 and Example 1

Proof (of Theorem 13) The entire proof is analogous to the proof of Theorem 2 in Section 2.3. Define $\mathbf{u} = \mathbf{u}_1 + \mathbf{u}_2$, where $\mathbf{u}_2 = (C_\ell \eta/(2\gamma)) \cdot \mathbf{w}^* \in \mathbb{R}^d$, where C_ℓ is the loss-dependent constant in Assumption 12. Recall the gradient descent iterate (GD), the learning rate scheduler (2), and the definition for $\phi(\cdot)$. Then,

$$\begin{aligned}
\left\|\mathbf{w}_{t+1} - \mathbf{u}\right\|^2 &= \left\|\mathbf{w}_t - \mathbf{u}\right\|^2 + 2\eta \left\langle \nabla\phi(\mathbf{w}_t), \mathbf{u} - \mathbf{w}_t \right\rangle + \eta^2 \left\|\nabla\phi(\mathbf{w}_t)\right\|^2 \\
&= \left\|\mathbf{w}_t - \mathbf{u}\right\|^2 + 2\eta \left\langle \nabla\phi(\mathbf{w}_t), \mathbf{u}_1 - \mathbf{w}_t \right\rangle + \eta \left[2\left\langle \nabla\phi(\mathbf{w}_t), \mathbf{u}_2 \right\rangle + \eta \left\|\nabla\phi(\mathbf{w}_t)\right\|^2 \right] \\
&\overset{(a)}{\leq} \left\|\mathbf{w}_t - \mathbf{u}\right\|^2 + 2\eta \left\langle \nabla\phi(\mathbf{w}_t), \mathbf{u}_1 - \mathbf{w}_t \right\rangle \overset{(b)}{\leq} \left\|\mathbf{w}_t - \mathbf{u}\right\|^2 + 2\eta \left(\phi(\mathbf{u}_1) - \phi(\mathbf{w}_t)\right).
\end{aligned} \tag{16}$$

Here, (a) is from Lemma 15 and (b) is from the convexity of $\phi(\cdot)$ following Assumption 12C. Rearranging the inequality above and telescoping the sum, we have

$$\frac{\left\|\mathbf{w}_t - \mathbf{u}\right\|^2}{2\eta(t+1)} + \frac{1}{t+1}\sum_{k=0}^{t} \phi(\mathbf{w}_k) \leq \phi(\mathbf{u}_1) + \frac{\left\|\mathbf{u}\right\|^2}{2\eta(t+1)}. \tag{17}$$

Recall $\langle \mathbf{w}^*, y_i \mathbf{x}_i \rangle \geq \gamma$, we have $\phi(\mathbf{u}_1) \leq -\gamma \left\|\mathbf{u}_1\right\|.$ Then, setting $\mathbf{u}_1 = (\gamma\eta(t+1)/2) \cdot \mathbf{w}^*$ and invoking $\mathbf{w}_0 = \mathbf{0}$ gives

$$\frac{1}{t+1}\sum_{k=0}^{t} \phi(\mathbf{w}_k) \leq -\gamma \left\|\mathbf{u}_1\right\| + \frac{\left\|\mathbf{u}_1 + \mathbf{u}_2\right\|^2}{2\eta(t+1)} \leq -\frac{\left(\gamma^2(t+1)\right)^2 - C_\ell}{4\gamma^2(t+1)}\eta.$$

Finally, we complete the proof by recalling the convexity of $\phi(\cdot)$ and $\mathcal{L}(\cdot) = \ell(-\phi(\cdot))$. ■

Proof (of Lemma 1) For exponential loss and logistic loss, Assumption 12A and Assumption 12B are trivial, and Assumption 12C is proven by Lemma 14 and Lemma 16.

For the polynomial loss ℓ_{poly}(Ji and Telgarsky, 2021), it is straightforward to verify Assumption 12A and Assumption 12B by taking first and second order derivatives. We know $\ell(\cdot)$ is twice continuously differentiable, and its inverse function $\ell^{-1}(z) = z^{-1/k} - 1$ $(z > 0)$ is continuously differentiable. To verify the convexity in Assumption 12C, we can use Lemma 16 and show that $\frac{\ell'(t)^2}{\ell(t)\ell''(t)}$ is decreasing on $\mathbb{R}$ (Ji and Telgarsky, 2021, Theorem 5.1). To further verify the Lipschitzness of $\phi(\cdot)$, let's define $h(z) := \left|\ell'(z)\right| / (k \cdot \ell(z)^{\frac{k+1}{k}})$. Observe that $h(\cdot)$ is continuously differentiable, $h(z) = 1$ for $z \geq 0$, $\lim_{z\to 0^-} h(z) = 1$ and $\lim_{z\to-\infty} h(z) = 0$. In addition, we can show that $h(z)$ is increasing for $z \leq 0$ by taking derivatives. This implies $h(z) \leq 1$, and hence, $\left|\ell'(z)\right| \leq k \cdot \ell(z)^{\frac{k+1}{k}}$, for every $z \in \mathbb{R}$. Denote $z_i = y_i \mathbf{x}_i^\top \mathbf{w}$. We have

$$\begin{aligned}
\left\|\nabla\phi(\mathbf{w})\right\| &\leq \frac{1}{n}\sum_{i=1}^{n} \left|\ell'(z_i)\right| \cdot \left| \left(\ell^{-1}\right)' \left(\frac{1}{n}\sum_{i=1}^{n} \ell(z_i) \right) \right| \leq \frac{k \cdot \frac{1}{n}\sum_{i=1}^{n} \ell(z_i)^{\frac{k+1}{k}}}{k \cdot \left(\frac{1}{n}\sum_{i=1}^{n} \ell(z_i)\right)^{\frac{k+1}{k}}} \\
&\leq \left(\frac{\max_{1\leq i\leq n} \ell(z_i)}{\frac{1}{n}\sum_{i=1}^{n} \ell(z_i)} \right)^{\frac{1}{k}} \leq n^{1/k}.
\end{aligned}$$

This proves $\phi(\cdot)$ satisfies Assumption 12C with $C_\ell = n^{1/k}$.

For the semi-circle loss (Shen, 2005), one can also verify Assumption 12A and Assumption 12B simply by taking derivatives and computing the inverse function $\ell^{-1}(z) = 1/z - z$ for $z > 0$. To verify the convexity of $\phi(\cdot)$, we use Lemma 16 and we can show the following quantity is decreasing along $\mathbb{R}$:

$$\frac{(\ell'(z))^2}{\ell(z)\ell''(z)} = \frac{\sqrt{z^2+4}}{\sqrt{z^2+4}+z}.$$

Finally, to verify the Lipschitzness in Assumption 12C, we define $z_i = y_i \mathbf{x}_i^\top \mathbf{w}$. Note that $|\ell'(z)| \le 1$ for all z, and $(\ell^{-1})'(z) = 1 + 1/z^2$ for $z > 0$. We then have

$$\|\nabla \phi(\mathbf{w})\| \le \frac{1}{n}\sum_{i=1}^n \big|\ell'(z_i)\big| \cdot \left|(\ell^{-1})'\left(\frac{1}{n}\sum_{i=1}^n \ell(z_i)\right)\right| \le \underbrace{\frac{1}{n}\sum_{i=1}^n \big|\ell'(z_i)\big|}_{\le 1} + \underbrace{\frac{\frac{1}{n}\sum_{i=1}^n |\ell'(z_i)|}{\left(\frac{1}{n}\sum_{i=1}^n \ell(z_i)\right)^2}}_{\le n} \le 1+n.$$

The upper bound for the second addition comes from the fact that $|\ell'(z)| \le \ell^2(z)$ for any z. Therefore, the semi-circle loss satisfies Assumption 12C with $C_\ell = n+1$. ■

References

Kwangjun Ahn, Jingzhao Zhang, and Suvrit Sra. Understanding the unstable convergence of gradient descent. In *International Conference on Machine Learning*, pages 247–257. PMLR, 2022.

Kwangjun Ahn, Sébastien Bubeck, Sinho Chewi, Yin Tat Lee, Felipe Suarez, and Yi Zhang. Learning threshold neurons via edge of stability. *Advances in Neural Information Processing Systems*, 36:19540–19569, 2023.

Jason M Altschuler and Pablo A Parrilo. Acceleration by stepsize hedging: Silver stepsize schedule for smooth convex optimization. *Mathematical Programming*, pages 1–14, 2024.

Jason M Altschuler and Pablo A Parrilo. Acceleration by stepsize hedging: Multi-step descent and the silver stepsize schedule. *Journal of the ACM*, 72(2):1–38, 2025.

Maksym Andriushchenko, Aditya Vardhan Varre, Loucas Pillaud-Vivien, and Nicolas Flammarion. Sgd with large step sizes learns sparse features. In *International Conference on Machine Learning*, pages 903–925. PMLR, 2023.

Alon Brutzkus, Amir Globerson, Eran Malach, and Shai Shalev-Shwartz. Sgd learns over-parameterized networks that provably generalize on linearly separable data. In *International Conference on Learning Representations*, 2018.

Yuhang Cai, Jingfeng Wu, Song Mei, Michael Lindsey, and Peter Bartlett. Large stepsize gradient descent for non-homogeneous two-layer networks: Margin improvement and fast optimization. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Lei Chen and Joan Bruna. Beyond the edge of stability via two-step gradient updates. In *International Conference on Machine Learning*, pages 4330–4391. PMLR, 2023.

Xuxing Chen, Krishnakumar Balasubramanian, Promit Ghosal, and Bhavya Agrawalla. From stability to chaos: Analyzing gradient descent dynamics in quadratic regression. *Transactions on machine learning research*, 2024.

Frank H Clarke. *Optimization and nonsmooth analysis.* SIAM, 1990.

Jeremy Cohen, Simran Kaur, Yuanzhi Li, J Zico Kolter, and Ameet Talwalkar. Gradient descent on neural networks typically occurs at the edge of stability. In *International Conference on Learning Representations*, 2020.

Alex Damian, Eshaan Nichani, and Jason D Lee. Self-stabilization: The implicit bias of gradient descent at the edge of stability. In *The Eleventh International Conference on Learning Representations*, 2022.

Mathieu Even, Scott Pesme, Suriya Gunasekar, and Nicolas Flammarion. (S)GD over diagonal linear networks: Implicit bias, large stepsizes and edge of stability. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Benjamin Grimmer. Provably faster gradient descent via long steps. *SIAM Journal on Optimization*, 34(3):2588–2608, 2024.

Benjamin Grimmer, Kevin Shu, and Alex L Wang. Accelerated objective gap and gradient norm convergence for gradient descent via long steps. *INFORMS Journal on Optimization*, 2025.

Godfrey Harold Hardy, John Edensor Littlewood, and George Pólya. *Inequalities.* Cambridge university press, 1952.

Ziwei Ji and Matus Telgarsky. Risk and parameter convergence of logistic regression. *arXiv preprint arXiv:1803.07300*, 2018.

Ziwei Ji and Matus Telgarsky. Characterizing the implicit bias via a primal-dual analysis. In *Algorithmic Learning Theory*, pages 772–804. PMLR, 2021.

Ziwei Ji, Nathan Srebro, and Matus Telgarsky. Fast margin maximization via dual acceleration. In *International Conference on Machine Learning*, pages 4860–4869. PMLR, 2021.

Jonathan Kelner, Annie Marsden, Vatsal Sharan, Aaron Sidford, Gregory Valiant, and Honglin Yuan. Big-step-little-step: Efficient gradient methods for objectives with multiple scales. In *Conference on Learning Theory*, pages 2431–2540. PMLR, 2022.

Lingkai Kong and Molei Tao. Stochasticity of deterministic gradient descent: Large learning rate for multiscale objective function. *Advances in neural information processing systems*, 33:2625–2638, 2020.

Guy Kornowski and Ohad Shamir. The oracle complexity of simplex-based matrix games: Linear separability and nash equilibria. In *The Thirty Eighth Annual Conference on Learning Theory*, pages 3327–3353. PMLR, 2025.

Itai Kreisler, Mor Shpigel Nacson, Daniel Soudry, and Yair Carmon. Gradient descent monotonically decreases the sharpness of gradient flow solutions in scalar networks and beyond. In *International Conference on Machine Learning*, pages 17684–17744. PMLR, 2023.

Guanghui Lan. *First-order and stochastic optimization methods for machine learning*, volume 1. Springer, 2020.

Aitor Lewkowycz, Yasaman Bahri, Ethan Dyer, Jascha Sohl-Dickstein, and Guy Gur-Ari. The large learning rate phase of deep learning: the catapult mechanism. *arXiv preprint arXiv:2003.02218*, 2020.

Miao Lu, Beining Wu, Xiaodong Yang, and Difan Zou. Benign oscillation of stochastic gradient descent with large learning rate. In *The Twelfth International Conference on Learning Representations*, 2023.

Kaifeng Lyu, Zhiyuan Li, and Sanjeev Arora. Understanding the generalization benefit of normalization layers: Sharpness reduction. *Advances in Neural Information Processing Systems*, 35:34689–34708, 2022.

Chao Ma, Daniel Kunin, Lei Wu, and Lexing Ying. Beyond the quadratic approximation: the multiscale structure of neural network loss landscapes. *Journal of Machine Learning 1 (2022), pp. 247-267*, 2022.

Omar Montasser, Abhishek Shetty, and Nikita Zhivotovskiy. Beyond worst-case online classification: Vc-based regret bounds for relaxed benchmarks. In *The Thirty Eighth Annual Conference on Learning Theory*, pages 4168–4202. PMLR, 2025.

Mor Shpigel Nacson, Jason Lee, Suriya Gunasekar, Pedro Henrique Pamplona Savarese, Nathan Srebro, and Daniel Soudry. Convergence of gradient descent on separable data. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3420–3428. PMLR, 2019.

Yurii Nesterov. *Lectures on Convex Optimization.* Springer Publishing Company, Incorporated, 2nd edition, 2018. ISBN 3319915770.

Albert BJ Novikoff. On convergence proofs for perceptrons. In *Proceedings of the Symposium on the Mathematical Theory of Automata.* New York, NY, 1962.

Jian Qian, Alexander Rakhlin, and Nikita Zhivotovskiy. Refined risk bounds for unbounded losses via transductive priors. *Journal of Machine Learning Research*, 27(26):1–64, 2026.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms.* Cambridge university press, 2014.

Yi Shen. *Loss functions for binary classification and class probability estimation*. PhD thesis, University of Pennsylvania, 2005.

Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *Journal of Machine Learning Research*, 19(70):1–57, 2018.

Alexander Tyurin. From logistic regression to the perceptron algorithm: Exploring gradient descent with large step sizes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 20938–20946, 2025.

Guanghui Wang, Rafael Hanashiro, Etash Kumar Guha, and Jacob Abernethy. On accelerated perceptrons and beyond. In *The Eleventh International Conference on Learning Representations*, 2022a.

Yuqing Wang, Minshuo Chen, Tuo Zhao, and Molei Tao. Large learning rate tames homogeneity: Convergence and balancing effect. In *International Conference on Learning Representations*, 2022b.

Yuqing Wang, Zhenghao Xu, Tuo Zhao, and Molei Tao. Good regularity creates large learning rate implicit biases: edge of stability, balancing, and catapult. In *NeurIPS 2023 Workshop on Mathematics of Modern Machine Learning*, 2023.

Zixuan Wang, Zhouzi Li, and Jian Li. Analyzing sharpness along gd trajectory: Progressive sharpening and edge of stability. *Advances in Neural Information Processing Systems*, 35: 9983–9994, 2022c.

Jingfeng Wu, Vladimir Braverman, and Jason D. Lee. Implicit bias of gradient descent for logistic regression at the edge of stability. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Jingfeng Wu, Peter L Bartlett, Matus Telgarsky, and Bin Yu. Large stepsize gradient descent for logistic loss: Non-monotonicity of the loss improves optimization efficiency. *Conference on Learning Theory*, 2024.

Lei Wu, Chao Ma, et al. How sgd selects the global minima in over-parameterized learning: A dynamical stability perspective. *Advances in Neural Information Processing Systems*, 31, 2018.

Zehao Zhang and Rujun Jiang. Accelerated gradient descent by concatenation of stepsize schedules. *arXiv preprint arXiv:2410.12395*, 2024.

Zihan Zhang, Jason Lee, Simon Du, and Yuxin Chen. Anytime acceleration of gradient descent. In *The Thirty Eighth Annual Conference on Learning Theory*, pages 5991–6013. PMLR, 2025.

Xingyu Zhu, Zixuan Wang, Xiang Wang, Mo Zhou, and Rong Ge. Understanding edge-of-stability training dynamics with a minimalist example. In *The Eleventh International Conference on Learning Representations*, 2023.