SGD: The Role of Implicit Regularization, Batch-size and Multiple Epochs

Satyen Kale
Google Research, NY
satyen@google.com

Ayush Sekhari
Cornell University
as3663@cornell.edu

Karthik Sridharan
Cornell University
ks999@cornell.edu

Abstract

Multi-epoch, small-batch, Stochastic Gradient Descent (SGD) has been the method of choice for learning with large over-parameterized models. A popular theory for explaining why SGD works well in practice is that the algorithm has an implicit regularization that biases its output towards a good solution. Perhaps the theoretically most well understood learning setting for SGD is that of Stochastic Convex Optimization (SCO), where it is well known that SGD learns at a rate of $O(1/\sqrt{n})$, where n is the number of samples. In this paper, we consider the problem of SCO and explore the role of implicit regularization, batch size and multiple epochs for SGD. Our main contributions are threefold:

1. We show that for any regularizer, there is an SCO problem for which Regularized Empirical Risk Minimzation fails to learn. This automatically rules out any implicit regularization based explanation for the success of SGD.
2. We provide a separation between SGD and learning via Gradient Descent on empirical loss (GD) in terms of sample complexity. We show that there is an SCO problem such that GD with any step size and number of iterations can only learn at a suboptimal rate: at least $\widetilde{\Omega}(1/n^{5/12})$.
3. We present a multi-epoch variant of SGD commonly used in practice. We prove that this algorithm is at least as good as single pass SGD in the worst case. However, for certain SCO problems, taking multiple passes over the dataset can significantly outperform single pass SGD.

We extend our results to the general learning setting by showing a problem which is learnable for any data distribution, and for this problem, SGD is strictly better than RERM for any regularization function. We conclude by discussing the implications of our results for deep learning, and show a separation between SGD and ERM for two layer diagonal neural networks.

1 Introduction

We consider the problem of stochastic optimization of the form:

$$\text{Minimize} \;\; F(w) \tag{1}$$

where the objective $F : \mathbb{R}^d \mapsto \mathbb{R}$ is given by $F(w) = \mathbb{E}_{z\sim D}[f(w; z)]$. The goal is to perform the minimization based only on samples $S = \{z_1, \dots, z_n\}$ drawn i.i.d. from some distribution $\mathcal{D}$. Standard statistical learning problems can be cast as stochastic optimization problems, with $F(w)$ being the population loss and $f(w, z)$ being the instantaneous loss on sample z for the model w.

Stochastic Convex Optimization (SCO). Perhaps the most well studied stochastic optimization problem is that of SCO. We define a problem to be an instance of a SCO problem if,

$$\textbf{Assumption I}\text{: Population loss } F \text{ is convex.} \tag{2}$$

35th Conference on Neural Information Processing Systems (NeurIPS 2021).

Notice above that we only require the population loss to be convex and do not impose such a condition on the instantaneous loss functions f.

Algorithms like Stochastic Gradient Descent (SGD), Gradient Descent on training loss (GD), and methods like Regularized Empirical Risk Minimization (RERM) that minimize training loss with additional penalty in the form of a regularizer are all popular choices of algorithms used to solve the above problem and have been analyzed theoretically for various settings of Stochastic Optimization problems (convex and non-convex). We discuss below a mix of recent empirical and theoretical insights about SGD algorithms that provide motivation for this work.

SGD and Implicit Regularization. A popular theory for why SGD generalizes so well when used on large over-parameterized models has been that of implicit regularization. It has been oberved that in large models, often there are multiple global minima for the empirical loss. However not all of these empirical minima have low suboptimality at the population level. SGD when used as the training algorithm often seems to find empirical (near) global minima that also generalize well and have low test loss. Hence while a general Empirical Risk Minimization (ERM) algorithm might fail, the implicit bias of SGD seems to yield a well-generalizing ERM. The idea behind implicit regularization is that the solution of SGD is equivalent to the solution of a Regularized Empirical Risk Minimizer (RERM) for an appropriate implicit regularizer.

The idea of implicit regularization of SGD has been extensively studied in recent years. In Gunasekar et al. [2018a], Zou et al. [2021], the classical setting of linear regression (with square loss) is considered and it was shown that when considering over-parameterized setting, the SGD algorithm is equivalent to fitting with a linear predictor with the smallest euclidean norm. In Soudry et al. [2018], Ji and Telgarsky [2018] linear predictors with logistic loss are considered and it was noted that SGD for this setting can be seen as having an implicit regularization of ℓ_2 norm. Gunasekar et al. [2018b] considered multi-layer convolutional networks with linear activation are considered and showned that SGD for this model can be seen as having an implicit regularization of $\ell_{2/L}$ norm (bridge penality for depth L networks) of the Fourier frequencies corresponding to the linear predictor. Gunasekar et al. [2018c] considered matrix factorization and showed that running SGD is equivalent to having a nuclear norm based regularizer. More recent work of Arora et al. [2019], Razin and Cohen [2020] shows that in particular in the deep matrix factorization setting, SGD cannot be seen as having any norm based implicit regularizer but rather a rank based one. However, in all these cases, the behavior of SGD corresponds to regularizers that are independent of the training data (e.g. rank, l_p-norm, etc).

One could surmise that a grand program for this line of research is that for problems where SGD works well, perhaps there is a corresponding implicit regularization explanation. That is, there exists a regularizer R such that SGD can been seen as performing exact or approximate RERM with respect to this regularizer. In fact, one can ask this question specific to SCO problems. That is, for the problem of SCO, is there an implicit regularizer R such that SGD can be seen as performing approximate RERM? In fact, a more basic question one can ask: is it true that SCO problem is always learnable using some regularized ERM algorithms? *We answer both these questions in the negative.*

SGD vs GD: Smaller the Batch-size Better the Generalization. It has been observed that in practice, while SGD with small batch size, and performing gradient descent (GD) with empirical loss as the objective function both minimize the training error equally well, the SGD solution generalizes much better than the full gradient descent one [Keskar et al., 2016, Kleinberg et al., 2018]. However, thus far, most existing literature on theorizing why SGD works well for over-parameterized deep learning models also work for gradient descent on training loss [Allen-Zhu and Li, 2019, Allen-Zhu et al., 2018a,b, Arora et al., 2018]. In this work, we construct a convex learning problem where single pass SGD provably outperforms GD run on empirical loss, which converges to a solution that has large excess risk. We thus provide *a problem instance where SGD works but GD has strictly inferior sample complexity (with or without early stopping).*

Multiple Epochs Help. The final empirical observation we consider is the fact that multiple epochs of SGD tends to further continually decrease not only the training error but also the test error [Zhang et al., 2016, Ma et al., 2018, Bottou and Bousquet, 2011]. In this paper, we construct *an SCO instance for which multiple epochs of single sample mini-batched SGD significantly outperforms single pass SGD*, and for the same problem, *RERM fails to converge to a good solution*, and hence, so does GD when run to convergence.

1.1 Our Contributions

We now summarize our main contributions in the paper:

1. **SGD and RERM Separation.** In Section 3, we demonstrate a SCO problem where a single pass of SGD over n data points obtains a $1/\sqrt{n}$ suboptimality rate. However, for any regularizer R, regularized ERM does not attain a diminishing suboptimality. We show that this is true even if the regularization parameter is chosen in a sample dependent fashion. Our result immediately rules out the explanation that SGD is successful in SCO due to some some implicit regularization.
2. **SGD and GD Separation.** In Section 4, we provide a separation between SGD and GD on training loss in terms of sample complexity for SCO. To the best of our knowledge, this is the first[1] such separation result between SGD and GD in terms of sample complexity. In this work, we show the existence of SCO problems where SGD with n samples achieves a suboptimality of $1/\sqrt{n}$, however, irrespective of what step-size is used and how many iterations we run for, GD cannot obtain a suboptimality better than $1/(n^{5/12}\log^2(n))$.
3. **Single-pass vs Multi-pass SGD.** In Section 5, we provide an adaptive multi-epoch SGD algorithm that is provably at least as good as single pass SGD algorithm. On the other hand, we also show that this algorithm can far outperform single pass SGD on certain problems where SGD only attains a rate of $1/\sqrt{n}$. Also, on these problems RERM fails to learn, indicating that GD run to convergence fails as well.
4. **SGD and RERM Separation in the Distribution Free Agnostic PAC Setting.** The separation result between SGD and RERM introduced earlier was for SCO. However, it turns out that the problem is not agnostically learnable for all distributions but only for distributions that make F convex. In Section 6, we provide a learning problem that is distribution-free learnable, and where SGD provably outperforms any RERM.
5. **Beyond Convexity (Deep Learning).** The convergence guarantee for SGD can be easily extended to stochastic optimization settings where the population loss $F(w)$ is *linearizable*, but may not be convex. We formalize this in Section 7, and show that for two layer diagonal neural networks with ReLU activations, there exists a distribution for which the population loss is *linearizable* and thus SGD works, but ERM algorithm fails to find a good solution. This hints at the possibility that the above listed separations between SGD and GD / RERM for the SCO setting, also extend to the deep learning setting.

1.2 Preliminaries

A standard assumption made by most gradient based algorithms for stochastic optimization problems is the following:

$$\textbf{Assumption II:} \begin{cases} F \text{ is } L\text{-Lipschitz (w.r.t. } \ell_2 \text{ norm)} \\ \exists w^\star \in \operatorname{argmin}_w F(w) \text{ s.t. } \|w_1 - w^\star\| \le B \; . \\ \sup_w \mathbb{E}_{z\sim D} \|\nabla f(w,z) - \nabla F(w)\|^2 \le \sigma^2 \end{cases} \tag{3}$$

In the above and throughout this work, the norm denotes the standard Euclidean norm and w_1 is some initial point known to the algorithm. Next, we describe below more formally what the regularized ERM, GD and SGD algorithms are.

(Regularized) Empirical Risk Minimization. Perhaps the simplest algorithm one could consider is the Empirical Risk Minimization (ERM) algorithm where one returns a minimizer of training loss (empirical risk)

$$F_S(w) := \frac{1}{n}\sum_{t=1}^{n} f(w; z_t) \; .$$

That is, $w_{\mathrm{ERM}} \in \operatorname{argmin}_{w\in\mathcal{W}} F_S(w)$. A more common variant of this method is one where we additionally penalize complex models using a regularizer function $R : \mathbb{R}^d \mapsto \mathbb{R}$. That is, a Regularized Empirical Risk Minimization (RERM) method consists of returning:

$$w_{\mathrm{RERM}} = \operatorname*{argmin}_{w\in\mathcal{W}} F_S(w) + R(w). \tag{4}$$

[1] Despite the way the result is phrased in Amir et al. [2021], their result does not imply separation between SGD and GD in terms of sample complexity but only number of iterations.

Gradient Descent (GD) on Training Loss. Gradient descent on training loss is the algorithm that performs the following update on every iteration:

$$w_{i+1}^{\mathrm{GD}} \leftarrow w_i^{\mathrm{GD}} - \eta \nabla F_S(w_i^{\mathrm{GD}}), \tag{5}$$

where η denotes the step size. After t rounds, we return the point $\widehat{w}_t^{\mathrm{GD}} := \frac{1}{t}\sum_{i=1}^t w_i^{\mathrm{GD}}$.

Stochastic Gradient Descent (SGD). Stochastic gradient descent (SGD) has been the method of choice for training large over-parameterized deep learning models, and other convex and non-convex learning problems. Single pass SGD algorithm runs for n steps and for each step takes a single data point and performs gradient update with respect to that sample. That, is on round t,

$$w_{i+1}^{\mathrm{SGD}} \leftarrow w_i^{\mathrm{SGD}} - \eta \nabla f(w_i^{\mathrm{SGD}}; z_i) \tag{6}$$

Finally, we return $\widehat{w}_n^{\mathrm{SGD}} = \frac{1}{n}\sum_{i=1}^n w_i^{\mathrm{SGD}}$. Multi-epoch (also known as multi-pass) SGD algorithm simply cycles over the dataset multiple times continuing to perform the same update specified above. It is well know that single pass SGD algorithm enjoys the following convergence guarantee:

Theorem 1 (Nemirovski and Yudin [1983]). *On any SCO problem satisfying Assumption I in (2) and Assumption II in (3), running SGD algorithm for n steps with the step size of $\eta = 1/\sqrt{n}$ enjoys the guarantee*

$$\mathbb{E}_S[F(\widehat{w}_n^{\mathrm{SGD}})] - \inf_{w \in \mathbb{R}^d} F(w) \le O\Big(\frac{1}{\sqrt{n}}\Big),$$

where the constant in the order notation only depends on constants B, L and σ in (3) and is independent of the dimension d.

In fact, even weaker assumptions like one-point convexity or star convexity of F w.r.t. an optimum on the path of the SGD suffices to obtain the above guarantee. We use this guarantee of SGD algorithm throughout this work. Up until Section 6 we only consider SCO problems for which SGD automatically works with the above guarantee.

2 Related Work

On the topic of RERM, implicit regularization and SGD, perhaps the work most relevant to this paper is that of Dauber et al. [2020]. Just like this work, they also consider the general setting of stochastic convex optimization (SCO) and show that for this setting, for no so-called *"admissible"* data independent regularizer, SGD can be seen as performing implicit regularized ERM. For the implicit regularization part of our work, while in spirit the work aims at accomplishing some of the similar goals, their work is in the setting where the instantaneous losses are also convex and Lipschitz. This means that, for their setting, while SGD and regularized ERM may not coincide, regularized ERM is indeed still an optimal algorithm as shown in Shalev-Shwartz et al. [2009]. In this work we show a strict separation between SGD and RERM with an example where SGD works but no RERM can possibly provide a non-trivial learning guarantee. Second, qualitatively, their separation result in Theorems 2 and 3 are somewhat unsatisfactory. This is because while they show that for every regularizer there is a distribution for which the SGD solution is larger in value than the regularized training loss of the regularized ERM w.r.t. that regularizer, the amount by which it is larger can depend on the regularizer and can be vanishingly small. Hence it might very well be that SGD is an approximate RERM. In fact, if one relaxes the requirement of their admissible relaxations then it is possible that their result doesn't hold. For instance, for square norm regularizer, the gap on regularized objective between RERM and SGD is only shown to be as large as the regularization parameter which is typically set to be a diminishing function of n.

On the topic of comparison of GD on training loss with single epoch SGD, one can hope for three kinds of separation. First, separation in terms of work (number of gradient computations), second, separation in terms of number of iterations, and finally separation in terms of sample complexity. A classic result from Nemirovski and Yudin [1983] tells us that for the SCO setting, the number of gradient computations required for SGD to obtain a suboptimality guarantee of ε is equal to the optimal sample complexity (for ε) and hence is optimal (in the worst case). On the other hand, a single iteration of GD on training loss requires the same number of gradient computations and hence any more than a constant number of iterations of GD already gives a separation between GD and

SGD in terms of work. This result has been explored in Shalev-Shwartz et al. [2007] for instance. The separation of GD and SGD in terms of number of iterations has been considered in Amir et al. [2021]. Amir et al. [2021] demonstrate a concrete SCO problem on which GD on the training loss requires at least $\Omega(1/\varepsilon^4)$ steps to obtain an ε-suboptimal solution for the test loss. Whereas, SGD only requires $O(1/\varepsilon^2)$ iterations. However, their result does not provide any separation between GD and SGD in terms of sample complexity. Indeed, in their example, using the upper bound for GD using Bassily et al. [2020] one can see that if GD is run on n samples for $T = n^2$ iterations with the appropriate step size, then it does achieve a $1/\sqrt{n}$ suboptimality. In comparison our work provides a much stronger separation between GD and SGD. We show a sample complexity separation, meaning that to obtain a specific suboptimality, GD requires more samples than SGD, irrespective of how many iterations we run it for. Our separation result also yields separation in terms of both number of iterations and number of gradient computations.

3 Regularized ERM, Implicit Regularization and SGD

In Shalev-Shwartz et al. [2009] (see also Feldman [2016]), SCO problems where not just the population loss F but where also for each $z \in \mathcal{Z}$, the instantaneous loss $f(\cdot, z)$ is convex is considered. In this setting, the authors show that the appropriate ℓ_2 norm square regularized RERM always obtains the optimal rate of $1/\sqrt{n}$. However, for SGD to obtain a $1/\sqrt{n}$ guarantee, one only needs convexity at the population level. In this section, based on construction of SCO problems that are convex only at population level and not at the empirical level, we show a strict separation between SGD and RERM. Specifically, while SGD is always successful for any SCO problem we consider here, we show that for any regularizer R, there is an instance of an SCO problem for which RERM w.r.t. this regularizer has suboptimality lower bounded by a constant.

In Shalev-Shwartz et al. [2009] (see also Feldman [2016]), SCO problems where not just the population loss F but where also for each $z \in \mathcal{Z}$, the instantaneous loss $f(\cdot, z)$ is convex is considered. In this setting, the authors show that the appropriate ℓ_2 norm square regularized RERM always obtains the optimal rate of $1/\sqrt{n}$. However, for SGD to obtain a $1/\sqrt{n}$ guarantee, one only needs convexity at the population level. In this section, based on construction of SCO problems that are convex only at population level and not at the empirical level, we show a strict separation between SGD and RERM. Specifically, while SGD is always successful for any SCO problem we consider here, we show that for any regularizer R, there is an instance of an SCO problem for which RERM with respect to this regularizer has suboptimality lower bounded by a constant.

Theorem 2. *For any regularizer R, there exists an instance of a SCO problem that satisfies both Assumptions I and II given in Equations (2) and (3), for which*

$$\mathbb{E}_S[F(w_{\mathrm{RERM}})] - \inf_{w \in \mathbb{R}^d} F(w) \geq \Omega(1),$$

where w_{RERM} is the solution to (4) with respect to the prescribed regularizer.

The regularizer R we consider in the above result is sample independent, that is, it has to be chosen before receiving any samples. In general, if one is allowed an arbitrary sample dependent regularizer, one can encode any learning algorithm as an RERM. This is because, for any algorithm, one can simply contrive the regularizer R to have its minimum at the output model of the algorithm on the given sample, and a very high penalty on other models. Hence, to have a meaningful comparison between SGD (or for that matter any algorithm) and RERM, one can either consider sample independent regularizers or at the very least, consider only some specific restricted family of sample dependent regularizers. One natural variant of considering mildly sample dependent regularizers is to first, in a sample independent way pick some regularization function R and then allow arbitrary sample dependent regularization parameters that multiply the regularizer R. The following corollary shows that even with such mildly data dependent regularizers, one can still find SCO instances for which the RERM solution has no non-trivial convergence guarantees.

Corollary 1. *For any regularizer R, there exists an instance of a SCO problem that satisfies both Assumptions I and II, for which the point $w_{\mathrm{RERM}} = \mathrm{argmin}_{w \in \mathcal{W}} F_S(w) + \lambda R(w)$, where λ is any arbitrary sample dependent regularization parameter, has the lower bound*

$$\mathbb{E}_S[F(w_{\mathrm{RERM}})] - \inf_{w \in \mathbb{R}^d} F(w) \geq \Omega(1).$$

The SCO problem in Theorem 2 and Corollary 1 is based on the function $f_{(A)}$ given by:

$$f_{(A)}(w;z) = y\|(w-\alpha)\odot x\|, \tag{A}$$

where each instance $z \in \mathcal{Z}$ can be written as a triplet $z = (x, y, \alpha)$ and the notation $\odot$ denotes Hadamard product (entry wise product) of the two vectors. We set $x \in \{0,1\}^d$, $y \in \{\pm 1\}$ and $\alpha \in \{0, e_1, \ldots, e_d\}$ where e_1 to e_d denote the standard basis in d dimensions. We also set $d > 2^n$.

In the following, we provide a sketch for why ℓ_2-norm square regularization fails and show no regularization parameter works. In the detailed proof provided in the Appendix, we deal with arbitrary regularizers. The basic idea behind the proof is simple. Consider the distribution: $x \sim \mathrm{Unif}(\{0,1\})^d$, and y is set to be $+1$ with probability 0.6 and -1 with probability 0.4, and set $\alpha = e_1$ deterministically. In this case, note that the population function is $0.2\,\mathbb{E}_{x\sim\mathrm{Unif}\{0,1\}^d}\,\|x \odot (w - e_1)\|_2$ which is indeed convex, Lipchitz and sandwiched between $0.1\|w-e_1\|$ and $0.2\|w-e_1\|$. Hence, any ε sub-optimal solution $\widehat{w}$ must satisfy $\|\widehat{w} - e_1\| \le 10\varepsilon$.

Since $d > 2^n$, with constant probability there is at least one coordinate, say $\widehat{j} \in [d]$, such that for any data sample (x_t, y_t) for $t \in [n]$, we have $x_t[\widehat{j}] = 0$ whenever $y_t = +1$ and $x_t[\widehat{j}] = 1$ whenever $y_t = -1$. Hence, ERM would simply put large weight on the $\widehat{j}$ coordinate and attain a large negative value for training loss (as an example, $w = e_{\widehat{j}}$ has a training loss of roughly -0.4). However, for test loss to be small, we need the algorithm to put little weight on coordinate $\widehat{j}$. Now for any square norm regularizer $R(w) = \lambda\|w\|_2^2$, to prevent RERM from making this $\widehat{j}$ coordinate large, λ has to be at least as large as a constant.

However, we already argued that any ε sub-optimal solution $\widehat{w}$ should be such that $\|\widehat{w} - e_1\| \le 10\varepsilon$. When λ is chosen to be as large as a constant, the regularizer will bias the solution towards 0 and thus the returned solution will never satisfy the inequality $\|\widehat{w} - e_1\| \le 10\varepsilon$. This leads to a contradiction: To find an ε sub-optimal solution, we need a regularization that would avoid picking large value on the spurious coordinate $\widehat{j}$ but on the other hand, any such strong regularization, will not allow RERM to pick a solution that is close enough to e_1. Hence, any such regularized ERM cannot find an ε sub-optimal solution. This shows that no regularization parameter works for norm square regularization. In the Appendix we expand this idea for arbitrary regularizers R.

The construction we use in this section has dimensionality that is exponential in number of samples n. However similarly modifying the construction in Feldman [2016], using the $y = \pm 1$ variable multiplying the construction there, one can extend this result to a case where dimensionality is $\Theta(n)$. Alternatively, noting that for the maximum deviation from mean over d coordinates is of order $\sqrt{\log d/n}$, one can use a simple modification of the exact construction here, and instead of getting the strong separation like the one above where RERM does not learn but SGD does, one can instead have $d = n^p$ for $p \in (0,1]$ and obtain a separation where SGD obtains the $1/\sqrt{n}$ rate but no RERM can beat a rate of order $\sqrt{\log n/n}$.

Implicit Regularization. As mentioned earlier, a proposed theory for why SGD algorithms are so successful is that they are finding some implicitly regularized empirical risk minimizers, and that this implicit bias helps them learn effectively with low generalization error. However, at least for SCO problems, the above strict separation result tells us that SGD cannot be seen as performing implicitly regularized ERM, neither exactly nor approximately.

4 Gradient Descent (Large-batch) vs SGD (Small-batch)

In the previous section, we provided an instance of a SCO problem on which no regularized ERM works as well as SGD. When gradient descent algorithm (specified in (5)) is used for training, one would expect that GD, and in general large batch SGD, will also eventually converge to an ERM and so after enough iterations would also fail to find a good solution. In this section, we formalize this intuition to provide lower bounds on the performance of GD.

Theorem 3. *There exists an instance of a SCO problem such that for any choice of step size η and number of iterations T, the following lower bound on performance of GD holds:*

$$\mathbb{E}_S[F(\widehat{w}_T^{GD})] - \inf_{w\in\mathbb{R}^d} F(w) \ge \Omega\Big(\frac{1}{n^{5/12}}\Big).$$

Theorem 3 suggests that there is an instance of a SCO problem for which the performance of GD is lower bound by $1/n^{0.42}$. On the other hand, SGD with the step size of $\eta = 1/n^{0.5}$ learns at a rate of $1/n^{0.5}$ for this problem. This suggests that GD (large batch size) is a worse learning algorithm than SGD. We defer the proof to Appendix C and give a brief sketch below.

Our lower bound proof builds on the recent works of Amir et al. [2021], which gives an instance of a SCO problem for which the performance guarantee of GD algorithm is $\Omega(\eta\sqrt{T} + 1/\eta T)$. We provide an instance of a SCO problem for which GD has a lower bound of $\Omega(\eta T/n)$. Adding the two instances together gives us an SCO problem for which GD has lower bound of

$$\Omega\Big(\eta\sqrt{T} + \frac{1}{\eta T} + \frac{\eta T}{n}\Big). \tag{7}$$

This lower bound is by itself not sufficient. In particular, for $\eta = 1/n^{3/2}$ and $T = n^2$, the right hand side of (7) evaluates to $O(1/\sqrt{n})$, which matches the performance guarantee of SGD for this SCO problem. Hence, this leaves open the possibility that GD may work as well as SGD. However, note that in order to match the performance guarantee of SGD, GD needs to be run for quadratically more number of steps and with a smaller step size. In fact, we can show that in order to attain a $O(1/\sqrt{n})$ rate in (7), we must set $\eta = O(1/n^{3/2})$ and $T = \omega(n^2)$.

The lower bound in Theorem 3 follows by adding to the SCO instance in (7) another objective that rules out small step-sizes. This additional objective is added by increasing the dimensionality of the problem by one and on this extra coordinate adding a stochastic function that is convex in expectation. The expected loss on this coordinate is a piecewise linear convex function. However, the stochastic component on this coordinate has random kinks at intervals of width $1/n^{5/4}$ that vanish in expectation, but can make the empirical loss point in the opposite direction with probability $1/2$. Since the problem is still an SCO problem, SGD works as earlier. On the other hand, when one considers training loss, there are up to n of these kinks and roughly half of them make the training loss flat. Thus, if GD is used on training loss with step size smaller than $1/n^{5/4}$ then it is very likely that GD hits at least one such kink and will get stuck there. This function, thus, rules out GD with step size η smaller than $1/n^{5/4}$. Restricting the step size $\eta = \Omega(1/n^{5/4})$, the lower bound in (7) evaluates to the $\Omega(1/n^{5/12})$ giving us the result of Theorem 3.

Our lower bound in (7) matches the recently shown performance guarantee for GD algorithm by Bassily et al. [2020]; albeit under slightly different assumptions on the loss function $f(\cdot; z)$. Their work assumes that the loss functions $f(w; z)$ is convex in w for every $z \in \mathcal{Z}$. On the other hand, we do not require convexity of f but only that $F(w)$ is convex in w (see our Assumptions I and II).

5 Single-pass vs Multi-pass SGD

State of the art neural networks are trained by taking multiple passes of SGD over the dataset. However, it is not well understood when and why multiple passes help. In this section, we provide theoretical insights into the benefits of taking multiple passes over the dataset. The multi-pass SGD algorithm that we consider is:

1. Split the dataset S into two equal sized datasets S_1 and S_2.
2. Run k passes of SGD algorithm using a fixed ordering of the samples in S_1 where,
   - The step size for the jth pass is set as $\eta_j = 1/\sqrt{nj}$.
   - At the end of jth pass, compute $\widehat{w}_j = \frac{2}{nj}\sum_{t=1}^{nj/2} w_j^{\mathrm{SGD}}$ as the average of all the iterates generated so far.
3. Output the point $\widehat{w}^{\mathrm{MP}} := \operatorname{argmin}_{w\in\widehat{\mathcal{W}}} F_{S_2}(w)$ where $\widehat{\mathcal{W}} := \{\widehat{w}_1, \dots, \widehat{w}_k\}$.

The complete pseudocode is given in the Appendix. In the following, we show that the above multi-pass SGD algorithm performs at least as well as taking a single pass of SGD.

Proposition 1. *The output $\widehat{w}^{\mathrm{MP}}$ of multipass-SGD algorithm satisfies*

$$\mathbb{E}_S[F(\widehat{w}^{\mathrm{MP}})] \le \mathbb{E}_S[F(\widehat{w}_{n/2}^{\mathrm{SGD}})] + \widetilde{O}\Big(\frac{1}{\sqrt{n}}\Big),$$

where $\widehat{w}_{n/2}^{\mathrm{SGD}}$ denotes the output of running (one pass) SGD algorithm for $n/2$ steps with step size $1/\sqrt{n}$.

This suggests that the output point of the above multi-pass SGD algorithm is not too much worse than that of SGD (single pass). For problems in SCO, SGD has a rate of $O(1/\sqrt{n})$ (see Theorem 1), and in for these problems, the above bound implies that multi-pass SGD also enjoys the rate of $\widetilde{O}(1/\sqrt{n})$.

5.1 Multiple Passes Can Help!

In certain favorable situations the output of multi-pass SGD can be much better:

Theorem 4. *Let k be a positive integer and let $R(\cdot)$ be a regularization function. There exists an instance of a SCO problem such that:*

(a) For any step size η, the output of the SGD algorithm has the lower bound

$$\mathbb{E}_S[F(\widehat{w}_n^{SGD})] - \inf_{w \in \mathbb{R}^d} F(w) \geq \Omega\Big(\frac{1}{\sqrt{n}}\Big).$$

Furthermore, running SGD with $\eta = 1/\sqrt{n}$ achieves the above $1/\sqrt{n}$ rate.

(b) On the other hand, multi-pass SGD algorithm with k passes has the following guarantee:

$$\mathbb{E}_S[F(\widehat{w}^{\mathrm{MP}})] - \inf_{w \in \mathbb{R}^d} F(w) \leq O\Big(\frac{1}{\sqrt{nk}}\Big).$$

(c) RERM algorithm has the lower bound: $\mathbb{E}_S[F(w_{\mathrm{RERM}})] - \inf_{w \in \mathbb{R}^d} F(w) \geq \Omega(1)$.

We defer the proof details to Appendix D and provide the intuition below. Parts (a) and (b) follow easily by taking a standard construction in [Nesterov, 2014, Section 3.2.1]. Here, a convex deterministic function F_N over an optimization variable v is provided such that Assumption II is satisfied with $L, B = O(1)$, and any gradient based scheme (such as SGD, which is equivalent to GD on F_N) has a suboptimality of $\Omega(1/\sqrt{n})$ after n iterations (which corresponds to single pass SGD). On the other hand, because F_N satisfies Assumption II, k passes of SGD, which correspond to nk iterations of GD, will result in suboptimality of $O(1/\sqrt{nk})$ for the given step sizes.

The more challenging part is to prove part (c). It is tempting to simply add the SCO instance from Theorem 2, but that may break the required upper bound of $O(1/\sqrt{nk})$ for multipass SGD. To get around this issue, we construct our SCO instance by making z consist of k components $\{\xi_1, \dots, \xi_k\}$ drawn independently from the distribution considered in Theorem 2. Furthermore, the loss function considered in Theorem 2 also defines the loss corresponding to each ξ_i component for an optimization variable w that is different from the optimization variable v.

The key idea in the construction is that, while each data sample consists of k independently sampled components, at any time step, SGD gets to see only one of these components. Specifically, in the first pass over the dataset, we only observe the ξ_1 component of every sample, and in the second pass we only observe the ξ_2 component for every sample and so on for further passes. This behavior for SGD is induced by using an independent control variable u. The optimization objective for u is such that SGD increases u monotonically with every iteration. The value of u controls which of the ξ_i components is observed during that time step. In particular, when we run our multipass SGD algorithm, during the first pass, the value of u is such that we get to see ξ_1 only. However, on the second pass, u has become large enough to reveal $\xi_s 2$, and so on for further passes. Thus, in k passes, multipass SGD gets to see nk "fresh" samples and hence achieves an suboptimality of $O(1/\sqrt{nk})$, as required in part (b). Finally, part (c) follows from the same reasoning as in Theorem 2 since the same SCO instance is used.

6 Distribution Free Learning

In this section, we consider the general learning setting [Vapnik, 2013] and aim for a distribution free learnability result. That is, we would like to provide problem settings where one has suboptimality that diminishes with n for any distribution over the instance space $\mathcal{Z}$. For the SCO setting we considered earlier, the assumptions in (2) and the assumption in (3) that the population loss is convex, imposes restrictions on the distributions allowed to be considered. Specifically, since $f(\cdot\,; z)$ need not be convex for every z, one can easily construct distributions for which the problem is not SCO and in fact, may not even be learnable. E.g., consider the distribution that puts all its mass on z for which $f(\cdot\,; z)$ is non-convex. In this section we will consider problems that are so called learnable, meaning that there is a learning algorithm with diminishing in n suboptimality for any distribution $\mathcal{D}$ on the instance space $\mathcal{Z}$. Under this setting, we show a separation between SGD and RERM. Specifically, we provide a problem instance that is learnable at a rate of $c_n > \Omega(1/n^{1/4})$ for any distribution over the instance space $\mathcal{Z}$. In particular, we show that the worst case rate of SGD is c_n for this problem. However, on the subset of distributions for which the problem is SCO, SGD as expected obtains a rate of $1/\sqrt{n}$. On the other hand, for the same problem we show that RERM while having worst case rate no better than c_n, has a lower bound of $\Omega(c_n^2)$ on SCO instances.

Our lower bound in this section is based on the following learning setting: For $d > 2^n$, let $\mathcal{Z} = \{0,1\}^d \times \{0, e_1, \dots, e_d\}$ be the instance space, let the instantaneous loss function be given by:

$$f_{(B)}(w; z) = \frac{1}{2}\|(w-\alpha)\odot x\|^2 - \frac{c_n}{2}\|w-\alpha\|^2 + \max\{1, \|w\|^4\}, \tag{B}$$

where $z = (x, \alpha)$ and $c_n := n^{-(\frac{1}{4}-\gamma)}$ for some $\gamma > 0$.

Additionally, let $\mathscr{D}_c$ denote the set of distributions over instances space $\mathcal{Z}$ for which F is convex, i.e.

$$\mathscr{D}_c := \left\{\mathcal{D} \mid F(w) = \mathbb{E}_{z\sim\mathcal{D}}\left[f_{(B)}(w,z)\right] \text{ is convex in } w\right\}. \tag{8}$$

Note that under the distributions from $\mathscr{D}_c$, the problem is an instance of an SCO problem. We show a separation between SGD and RERM over the class $\mathscr{D}_c$. In the theorem below we claim that the above learning problem is learnable with a worst case rate of order c_n (by SGD), and whenever $\mathcal{D} \in \mathscr{D}_c$, SGD learns at a faster rate of $1/\sqrt{n}$. To begin with we first provide the following proposition for the problem described by the function $f_{(B)}$ that shows that no algorithm can have a worst case rate better than c_n for this problem.

Proposition 2. *For any algorithm* ALG *that outputs $\widehat{w}^{\mathrm{ALG}}$, there is a distribution $\mathcal{D}$ on the instance space, such that for the learning problem specified by function $f_{(B)}$:*

$$\mathbb{E}_S[F(\widehat{w}^{\mathrm{ALG}})] - \inf_{w\in\mathcal{W}} F(w) \geq \frac{c_n}{4}$$

Next, we show that SGD obtains this worst case rate, and a much better rate of $1/\sqrt{n}$ for any distribution in $\mathscr{D}_c$. However, we also show that for this problem, no RERM can obtain a rate better than c_n^2 for every distribution in $\mathscr{D}_c$. This shows that while SGD and RERM have the same worst case rate, SGD outperforms any RERM whenever the problem turns out to be convex.

Theorem 5. *For the learning problem specified by function $f_{(B)}$:*

(a) For every regularizer R, there exists a distribution $\mathcal{D} \in \mathscr{D}_c$ such that,

$$\mathbb{E}_S[F(w_{\mathrm{RERM}})] - \inf_{w\in\mathbb{R}^d} F(w) \geq \Omega(c_n^2).$$

(b) For the SGD algorithm we have the following upper bounds:

$$\forall \mathcal{D} \in \mathscr{D}_c,\ \ \mathbb{E}_S[F(\widehat{w}_n^{\mathrm{SGD}})] - \inf_{w\in\mathbb{R}^d} F(w) \leq O\left(\frac{1}{\sqrt{n}}\right),$$

$$\forall \mathcal{D} \notin \mathscr{D}_c,\ \ \mathbb{E}_S[F(\widehat{w}_n^{\mathrm{SGD}})] - \inf_{w\in\mathbb{R}^d} F(w) \leq O\left(c_n\right).$$

As an example, plugging in $c_n = n^{-\frac{1}{8}}$ implies that when $\mathcal{D} \notin \mathscr{D}_c$, the suboptimality of SGD is bounded by $O(1/n^{1/8})$, and when $\mathcal{D} \in \mathscr{D}_c$, the suboptimality of SGD is bounded by $O(1/\sqrt{n})$. However, for any RERM, there exists a distribution $D \in \mathscr{D}_c$, on which the RERM has a suboptimality of $\Omega(1/n^{1/4})$ and the worst case rate of any RERM is also $n^{-1/8}$. This suggests that SGD is a superior algorithm to RERM for any regularizer R, even in the distribution free learning setting.

7 α-Linearizable Functions and Deep Learning

While the classic convergence proof for SGD is shown for SCO setting, a reader familiar with the proof technique will recognize that the same result also holds when we only assume that the population loss F is star-convex or one-point-convex with respect to any optimum w^*, or in fact even if it is star-convex only on the path of SGD. The following definition of Linearizable population loss generalizes star-convexity and one-point-convexity.

Definition 1 (α-Linearizable)**.** *A stochastic optimization problem with population loss $F(w)$ is α-Linearizable if there exists a $w^* \in \operatorname{argmin} F(w)$ such that for every point $w \in \mathbb{R}^d$,*

$$F(w) - F(w^*) \leq \alpha \langle \nabla F(w), w - w^* \rangle.$$

For linearizable function, one can upper bound the suboptimality at any point w by a linear function given by $\nabla F(w)$. The convergence guarantee for SGD now follows by bounding the cumulative sum of this linear function using standard arguments, giving us the following performance guarantee.

Theorem 6. *On any α-Lineariazable stochastic optimization problem satisfying Assumption II in (3), running SGD algorithm for n steps with the step size of $\eta = 1/\sqrt{n}$ enjoys the guarantee:*

$$\mathbb{E}_S[F(\widehat{w}_n^{\mathrm{SGD}})] - \inf_{w \in \mathbb{R}^d} F(w) \leq O\Big(\frac{\alpha}{\sqrt{n}}\Big),$$

where the constant in the order notation only depends on constants B, L and σ in (3) and is independent of the dimension d.

The hypothesis that this phenomenon is what makes SGD successful for deep learning has been proposed and explored in various forms [Zhou et al., 2019, Kleinberg et al., 2018]. On the other hand our lower bound results hold even for the simpler SCO setting. Of course, to claim such a separation of SGD with respect to GD or RERM in the deep learning setting, one would need to show that our lower bound constructions can be represented as deep neural networks with roughly the same dimensionality as the original problem. In fact, all the functions that we considered so far can be easily expressed by restricted deep neural networks (where some weights are fixed) with square activation functions as we show in Appendix F.3. Although, it would be a stretch to claim that practical neural networks would look anything like our restricted neural network constructions; it still opens the possibility that the underlying phenomena we exploit to show these separations hold in practical deep learning setting. In the following, we give an example of a simple two layer neural network with ReLU activation function where SGD enjoys a rate of $1/\sqrt{n}$, but any ERM algorithm fails to find an $O(1)$-suboptimal solution.

Let the input sample (x, y) be such that $x \in \{0,1\}^d$ and $y \in \{-1, 1\}$. Given the weights $w = (w_1, w_2)$, where $w_1 \in \mathbb{R}^d$ and $w_2 \in \mathbb{R}^d$, we define a two layer ReLU neural network that on the input x outputs

$$h(w; x) = \mathrm{ReLU}(w_2^\top \mathrm{ReLU}(w_1 \odot x)).$$

This is a two layer neural network with the input layer having a diagonal structure and output layer being fully connected (hence the name diagonal neural network). Suppose the network is trained using the absolute loss, i.e. on data sample $z = (x, y)$ and for weights $w = (w_1, w_2)$, we use the loss

$$f(w; z) = |y - h(w; z)| = |y - \mathrm{ReLU}(w_2^\top \mathrm{ReLU}(w_1 \odot x))|. \tag{9}$$

Theorem 7 (Two layer diagonal network)**.** *For the loss function given in (9) using a two layer diagonal neural network, there exists a distribution $\mathcal{D}$ over the instance space $\mathcal{Z}$ such that:*

(a) *$F(w)$ is $1/2$-Linearizable, and thus SGD run with step-size $1/\sqrt{n}$ has excess risk $O(1/\sqrt{n})$.*

(b) *For $d \geq 2^n$, with probability 0.9, ERM algorithm fails to find an $O(1)$-suboptimal point.*

Remark 1. *The result of Theorem 7 can be extended to diagonal two layer neural networks trained with linear loss $f(w; z) = yh(w; x)$, or with hinge loss $f(w; x) = \max\{0, 1 - yh(w; z)\}$.*

While the above result shows that for a simple two layer neural network, SGD performs better than ERM, our construction requires the first layer to be diagonal. It is an interesting future research direction to explore whether a similar phenomena can be demonstrated in more practical network architectures, for eg. fully connected networks, convolutional neural networks (CNN), recurrent neural networks (RNN), etc. It would also be interesting to extend our lower bounds for GD algorithm from Section 4 to these network architectures. The key idea is that SGD only requires certain nice properties (eg. convex, Linearizable, etc) at the population level, which might fail to hold at the empirical level in large dimensional models; hence, batch algorithms like GD and RERM might fail.

Acknowledgements

We thank Roi Livni, Robert Kleinberg and Mehryar Mohri for helpful discussions. AS was an intern at Google Research, NY when a part of the work was performed. KS acknowledges support from NSF CAREER Award 1750575.

Funding Transparency Statement

Funding in direct support of this work: NSF CAREER Award 1750575.

References

Suriya Gunasekar, Jason Lee, Daniel Soudry, and Nathan Srebro. Characterizing implicit bias in terms of optimization geometry. In *International Conference on Machine Learning*, pages 1832–1841. PMLR, 2018a.

Difan Zou, Jingfeng Wu, Vladimir Braverman, Quanquan Gu, and Sham M Kakade. Benign overfitting of constant-stepsize sgd for linear regression. *arXiv preprint arXiv:2103.12692*, 2021.

Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *The Journal of Machine Learning Research*, 19(1): 2822–2878, 2018.

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

Suriya Gunasekar, Jason Lee, Daniel Soudry, and Nathan Srebro. Implicit bias of gradient descent on linear convolutional networks. *arXiv preprint arXiv:1806.00468*, 2018b.

Suriya Gunasekar, Blake Woodworth, Srinadh Bhojanapalli, Behnam Neyshabur, and Nathan Srebro. Implicit regularization in matrix factorization. In *2018 Information Theory and Applications Workshop (ITA)*, pages 1–10. IEEE, 2018c.

Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. Implicit regularization in deep matrix factorization. *arXiv preprint arXiv:1905.13655*, 2019.

Noam Razin and Nadav Cohen. Implicit regularization in deep learning may not be explainable by norms. *arXiv preprint arXiv:2005.06398*, 2020.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*, 2016.

Bobby Kleinberg, Yuanzhi Li, and Yang Yuan. An alternative view: When does sgd escape local minima? In *International Conference on Machine Learning*, pages 2698–2707. PMLR, 2018.

Zeyuan Allen-Zhu and Yuanzhi Li. Can sgd learn recurrent neural networks with provable generalization? *arXiv preprint arXiv:1902.01028*, 2019.

Zeyuan Allen-Zhu, Yuanzhi Li, and Yingyu Liang. Learning and generalization in overparameterized neural networks, going beyond two layers. *arXiv preprint arXiv:1811.04918*, 2018a.

Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. On the convergence rate of training recurrent neural networks. *arXiv preprint arXiv:1810.12065*, 2018b.

Sanjeev Arora, Nadav Cohen, and Elad Hazan. On the optimization of deep networks: Implicit acceleration by overparameterization. In *International Conference on Machine Learning*, pages 244–253. PMLR, 2018.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

Siyuan Ma, Raef Bassily, and Mikhail Belkin. The power of interpolation: Understanding the effectiveness of sgd in modern over-parametrized learning. In *International Conference on Machine Learning*, pages 3325–3334. PMLR, 2018.

Léon Bottou and Olivier Bousquet. The tradeoffs of large-scale learning. *Optimization for machine learning*, page 351, 2011.

Idan Amir, Tomer Koren, and Roi Livni. Sgd generalizes better than gd (and regularization doesn't help). *arXiv preprint arXiv:2102.01117*, 2021.

Arkadi Semenovič Nemirovski and David Borisovich Yudin. Problem complexity and method efficiency in optimization. 1983.

Assaf Dauber, Meir Feder, Tomer Koren, and Roi Livni. Can implicit bias explain generalization? stochastic convex optimization as a case study. *arXiv preprint arXiv:2003.06152*, 2020.

Shai Shalev-Shwartz, Ohad Shamir, Nathan Srebro, and Karthik Sridharan. Stochastic convex optimization. In *COLT*, 2009.

S. Shalev-Shwartz, Y. Singer, and N. Srebro. Pegasos: Primal Estimated sub-GrAdient SOlver for SVM. pages 807–814, 2007.

Raef Bassily, Vitaly Feldman, Cristóbal Guzmán, and Kunal Talwar. Stability of stochastic gradient descent on nonsmooth convex losses. *arXiv preprint arXiv:2006.06914*, 2020.

Vitaly Feldman. Generalization of erm in stochastic convex optimization: The dimension strikes back. *arXiv preprint arXiv:1608.04414*, 2016.

Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer Publishing Company, Incorporated, 1 edition, 2014. ISBN 1461346916.

Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.

Yi Zhou, Junjie Yang, Huishuai Zhang, Yingbin Liang, and Vahid Tarokh. Sgd converges to global minimum in deep learning via star-convex path. *arXiv preprint arXiv:1901.00451*, 2019.

Jiří Matoušek and Jan Vondrák. The probabilistic method. *Lecture Notes, Department of Applied Mathematics, Charles University, Prague*, 2001.

Dmitriy Panchenko. Some extensions of an inequality of vapnik and chervonenkis. *Electronic Communications in Probability*, 7:55–65, 2002.

Contents of Appendix

A Preliminaries

A.1 Additional notation

For a vector $w \in \mathbb{R}^d$, fo any $j \in [d]$, $w[j]$ denotes the j-th coordinate of w, $\|w\|$ denotes the Euclidean norm and $\|w\|_\infty$ denotes the ℓ_∞ norm. For any two vectors w_1 and w_2, $\langle w_1, w_2 \rangle$ denotes their inner product, and $w_1 \odot w_2$ denotes the vector generated by taking the Hadamard product of w_1 and w_2, i.e. $(w_1 \odot w_2)[j] = w_1[j] w_2[j]$ for $j \in [d]$. We denote by $\mathbf{1}_d$ a d-dimensional vector of all 1s, and the notation $\mathbb{I}_d$ denotes the identity matrix in d-dimensions. The notation $\mathcal{N}(0, \sigma^2)$ denotes Gaussian distribution with variance σ^2, and $\mathcal{B}(p)$ denotes the Bernoulli distribution with mean p.

For a function $f : \mathbb{R}^d \times \mathbb{R}$, we denote the gradient of f at the point $w \in \mathbb{R}^d$ by $\nabla f(w) \in \mathbb{R}^d$. The function f is said to be L-Lipschitz if $f(w_1) - f(w_2) \leq L \|w_1 - w_2\|$ for all w_1, w_2.

A.2 Basic algorithmic results

The following convergence guarantee for SGD algorithm is well known in stochastic convex optimization literature, and is included here for the sake of completeness.

Theorem 8 (Nemirovski and Yudin [1983]). *Let $w \in \mathbb{R}^d$ and $z \in \mathcal{Z}$. Given an initial point $w_1 \in \mathbb{R}^d$, loss function $f(w; z)$ and a distribution $\mathcal{D}$ such that:*

(a) The population loss $F(w) = \mathbb{E}_{z \sim \mathcal{D}}[f(w; z)]$ is L-Lipschitz and convex in w.

(b) For any w, $\mathbb{E}_{z \sim \mathcal{D}}[\|\nabla f(w; z) - \nabla F(w)\|] \leq \sigma^2$.

(c) The initial point w_1 satisfies $\|w_1 - w^*\| \leq B$ where $w^* \in \operatorname{argmin} F(w)$.

Further, let $S \sim \mathcal{D}^n$. Then, the point $\widehat{w}_n^{\mathrm{SGD}}$ obtained by running SGD algorithm given in (6), with step size $\eta = 1/\sqrt{n}$ for n steps using the dataset S satisfies

$$\mathbb{E}\big[F(\widehat{w}_n^{\mathrm{SGD}}) - F^*\big] \leq \frac{1}{\sqrt{n}}\big(\sigma^2 + L^2 + B^2\big),$$

where $F^* := \min_w F(w)$.

Proof. Let $\{w_t\}_{t \geq 1}$ denote the sequence of iterates generated by the SGD algorithm. We note that for any time $t \geq 1$,

$$\begin{aligned} \|w_{t+1} - w^*\|_2^2 &= \|w_{t+1} - w_t + w_t - w^*\|_2^2 \\ &= \|w_{t+1} - w_t\|_2^2 + \|w_t - w^*\|_2^2 + 2\langle w_{t+1} - w_t, w_t - w^* \rangle \\ &= \|-\eta \nabla f(w_t; z_t)\|_2^2 + \|w_t - w^*\|_2^2 + 2\langle -\eta \nabla f(w_t; z_t), w_t - w^* \rangle, \end{aligned}$$

where the last line follows from plugging in the SGD update rule that $w_{t+1} = w_t - \eta \nabla f(w_t; z_t)$.

Rearranging the terms in the above, we get that

$$\langle \nabla f(w_t; z_t), w_t - w^* \rangle \leq \frac{\eta}{2} \|\nabla f(w_t; z_t)\|_2^2 + \frac{1}{2\eta}\big(\|w_t - w^*\|_2^2 - \|w_{t+1} - w^*\|_2^2\big).$$

Taking expectation on both the sides, while conditioning on the point w_t, implies that

$$\begin{aligned} \langle \nabla F(w_t), w_t - w^* \rangle &\leq \frac{\eta}{2} \mathbb{E}\Big[\|\nabla f(w_t; z_t)\|_2^2\Big] + \frac{1}{2\eta}\big(\|w_t - w^*\|_2^2 - \|w_{t+1} - w^*\|_2^2\big) \\ &\leq \eta\, \mathbb{E}\Big[\|\nabla f(w_t; z_t) - \nabla F(w_t)\|_2^2\Big] + \eta \|\nabla F(w_t)\|_2^2 \\ &\qquad + \frac{1}{2\eta}\big(\|w_t - w^*\|_2^2 - \|w_{t+1} - w^*\|_2^2\big) \\ &\leq \eta(\sigma^2 + L^2) + \frac{1}{2\eta}\big(\|w_t - w^*\|_2^2 - \|w_{t+1} - w^*\|_2^2\big), \end{aligned}$$

where the inequality in the second line is given by the fact that $(a - b)^2 \leq 2a^2 + 2b^2$ and the last line follows from using Assumption II (see (3)) which implies that $F(w)$ is L-Lipschitz in w and that

$\mathbb{E}[\|\nabla f(w; z_t) - \nabla F(w)\|_2^2] \leq \sigma^2$ for any w. Next, using convexity of the function F, we have that $F(w^*) \geq F(w_t) - \langle \nabla F(w_t), w_t - w^* \rangle$. Thus

$$F(w_t) - F^* \leq \eta(\sigma^2 + L^2) + \frac{1}{2\eta}\big(\|w_t - w^*\|_2^2 - \|w_{t+1} - w^*\|_2^2\big).$$

Telescoping the above for t from 1 to n, we get that

$$\begin{aligned}\sum_{t=1}^{n}(F(w_t) - F^*) &\leq \eta n(\sigma^2 + L^2) + \frac{1}{2\eta}\big(\|w_1 - w^*\|_2^2 - \|w_{n+1} - w^*\|_2^2\big) \\ &\leq \eta n(\sigma^2 + L^2) + \frac{1}{2\eta}\|w_1 - w^*\|_2^2.\end{aligned}$$

Dividing both the sides by n, we get that

$$\frac{1}{n}\sum_{t=1}^{n}(F(w_t) - F^*) \leq \eta(\sigma^2 + L^2) + \frac{1}{2\eta n}\|w_1 - w^*\|_2^2.$$

An application of Jensen's inequality on the left hand side, implies that the point $\widehat{w}_n := \frac{1}{n}\sum_{t=1}^{n} w_t$ satisfies

$$\mathbb{E}[F(\widehat{w}_n) - F^*] \leq \eta(\sigma^2 + L^2) + \frac{1}{2\eta n}\|w_1 - w^*\|_2^2.$$

Setting $\eta = \frac{1}{\sqrt{n}}$ and using the fact that $\|w_1 - w^*\|_2 \leq B$ in the above, we get that

$$\mathbb{E}[F(\widehat{w}_n) - F^*] \leq \frac{1}{\sqrt{n}}\big(\sigma^2 + L^2 + B^2\big),$$

which is the desired claim. Dependence on the problem specific constants (σ, L and B) in the above bound can be improved further with a different choice of the step size η; getting the optimal dependence on these constants, however, is not the focus of this work . □

A.3 Basic probability results

Lemma 1 (Hoeffding's inequality)**.** *Let $X_1, \dots, X_n$ be independent random variables with values in the interval $[a, b]$, and the expected value $\mathbb{E}[X] = \mu$. Then, for every $t \geq 0$,*

$$\Pr\Big(\big|\frac{1}{n}\sum_{j=1}^{n} X_j - \mu\big| \geq t\Big) \leq 2\exp\Big(-\frac{2t^2 n}{(b-a)^2}\Big).$$

Lemma 2. *Let $j^* \in [d]$. Let X be a $\{0,1\}^d$ valued random variable such that $X[j]$ is sampled independently from $\mathcal{B}(p)$ for every $j \in [d]$. Let the $\{X_1, \dots, X_n\}$ denote n i.i.d. samples of the random variable X. If*

$$n \leq \log_{\frac{1}{1-p}}\Big(\frac{d-1}{\ln(10)}\Big),$$

then, with probability at least 0.9, there exists a coordinate $\widehat{j} \in [d]$ such that $\widehat{j} \neq j^*$, and $X_i[\widehat{j}] = 0$ for all $i \in [n]$.

Proof. Let E_j denote the event the coordinate $X_i[j] = 0$ for all $i \in [n]$. We note that for any $j \in [d]$,

$$\Pr(E_j) = (1-p)^n,$$

Further, let E denote the event that there exists a coordinate $\widehat{j} \in [d] \setminus \{j^*\}$ such that $X_i[\widehat{j}] = 0$ for all $i \in [n]$. Thus,

$$\Pr(E^c) = \Pr\Big(\bigcap_{j \in [d]\setminus\{j^*\}} E_j^c\Big) \overset{(i)}{=} \prod_{j \in [d]\setminus\{j^*\}} \Pr(E_j^c)$$

where the equality in (i) follows from the fact that the events $\{E_j\}$ are mutually independent to each other. Plugging in the bound for $\Pr(E_j^c)$, we get

$$\Pr(E^c) = (1 - (1-p)^n)^{d-1}.$$

Using the fact that $1 - a \leq e^{-a}$ for $a \geq 0$, we get

$$\Pr(E^c) \leq e^{-(1-p)^n (d-1)} \leq 0.1,$$

where the second inequality above holds for $n \leq \log_{\frac{1}{1-p}} \left(\frac{d-1}{\ln(10)} \right)$. □

Lemma 3 (Proposition 7.3.2, Matoušek and Vondrák [2001]). *For n even, let $X_1, \ldots, X_n$ be independent samples from $\mathcal{B}(1/2)$. Then, for any $t \in [0, n/8]$,*

$$\Pr\Big(\sum_{i=1}^{n} X_1 \geq \frac{n}{2} + t \Big) \geq \frac{1}{15} e^{-16t^2/n}.$$

Lemma 4 (Panchenko [2002]). *Let $\mathcal{W}$ denote a finite class of k points $\{w_1, \ldots, w_K\}$, and let $f(w; z)$ denote a loss function that is L-Lipschitz in the variable w for all $z \in \mathcal{Z}$. Further, let $S = \{z_i\}_{i=1}^n$ denote a dataset of n samples, each drawn independently from some distribution $\mathcal{D}$. Define the point $\widehat{w}_S = \operatorname{argmin}_{w \in \mathcal{W}} \sum_{i=1}^n \frac{1}{n} f(w, z_i)$. Then, with probability at least $1 - \delta$ over the sampling of the set S,*

$$F(\widehat{w}_S) \leq F^* + O\Big(\frac{L \log(K/\delta)}{n} + \sqrt{\frac{F^* L \log(K/\delta)}{n}} \Big),$$

where $F(w) := \mathbb{E}_{z \sim \mathcal{D}}[f(w; z)]$ and $F^* := \min_{w \in \mathcal{W}} F(w)$.

B Missing proofs from Section 3

Throughout this section, we assume that a data sample z consists of (x, y, α), where $x \in \{0,1\}^d$, $y \in \{-1, 1\}$ and $\alpha \in \{0, e_1, \ldots, e_d\}$. The loss function $f_{(A)} : \mathbb{R}^d \times \mathcal{Z}$ is given by:

$$f_{(A)}(w; (x, y, \alpha)) = y \| (w - \alpha) \odot x \|. \tag{10}$$

We also assume that $d \geq \ln(10) 2^n + 1$. Since $f_{(A)}$ is not differentiable when $w = 0$, we define the sub-gradient $\partial f_{(A)}(w) = 0$ at the point $w = 0$. Further, for the rest of the section, whenever clear from the context, we will ignore the subscript (A), and denote the loss function by $f(w; z)$. Additionally we define the following distribution over the samples $z = (x, y, \alpha)$.

Definition 2. *For parameters $\delta \in [0, \frac{1}{2}]$, $p \in [0, 1]$, and $a \in \{0, e_1, \ldots, e_d\}$, define the distribution $\mathcal{D}(\delta, p, a)$ over $z = (x, y, \alpha)$ as follows:*

$$x \sim \mathcal{B}(p)^{\otimes d}, \qquad y = 2r - 1 \quad \textit{for} \quad r \sim \mathcal{B}(\tfrac{1}{2} + \delta), \qquad \alpha = a.$$

The components x, y and α are sampled independently.

B.1 Supporting technical results

The following lemma provides some properties of the population loss under $\mathcal{D}(\delta, p, a)$ given in Definition 2.

Lemma 5. *For any $\delta \in [0, \frac{1}{2}]$, $p \in (0, 1]$, and $a \in \{0, e_1, \ldots, e_d\}$, the population loss under $f_{(A)}$ when the data is sampled i.i.d. from $\mathcal{D}(\delta, p, a)$ is convex in the variable w. Furthermore, the population loss has a unique minimizer at $w = a$ with $F(a) = 0$, and any ε-suboptimal minimizer w of the population loss must satisfy $\|w - a\| \leq \varepsilon / 2\delta p$.*

Proof. Let F be the population loss. Using the definition of F, we get

$$\begin{aligned} F(w) &= \mathbb{E}_{(x,y,\alpha) \sim \mathcal{D}(\delta, p, a)}[y \| (w - \alpha) \odot x \|] \\ &= \Pr(y = 1)\, \mathbb{E}_x[\| (w - a) \odot x \|] - \Pr(y = -1)\, \mathbb{E}_{x,\alpha}[\| (w - a) \odot x \|] \\ &= (\tfrac{1}{2} + \delta)\, \mathbb{E}_x[\| (w - a) \odot x \|] - (\tfrac{1}{2} - \delta)\, \mathbb{E}_x[\| (w - a) \odot x \|] \end{aligned}$$

$$= 2\delta\, \mathbb{E}_x[\|(w - a) \odot x\|].$$

Since, for any x and a, the function $\|(w - a) \odot x\|$ is a convex function of w, the above formula implies that F is also a convex function of w. Furthermore, F is always non-negative and $F(a) = 0$.

Now note that

$$F(w) = 2\delta\, \mathbb{E}_x[\|(w - a) \odot x\|] \geq 2\delta\|(w - a) \odot \mathbb{E}_x[x]\| = 2\delta p\|w - a\|,$$

where the second inequality follows from Jensen's inequality, and the last equality from the fact that $x \sim \mathcal{B}(p)^d$. Now, if w is an ε-suboptimal minimizer of F, then since $F(a) = 0$, the above bound implies that $\|w - a\| \leq \varepsilon/2\delta p$. This also implies, in particular using $\varepsilon = a$, that a is a unique minimizer of F. □

The next lemma establishes empirical properties of a dataset S of size n drawn from the distribution $\mathcal{D}(1/10, 1/2, a)$ given in Definition 2.

Lemma 6. *Let $j^* \in [d]$. Let S denote a dataset of size $n \leq \log_2(d/\ln(10))$ sampled i.i.d. from a distribution $\mathcal{D}(\frac{1}{10}, \frac{1}{2}, a)$ for some vector $a \in \{0, e_1, \dots, e_d\}$. Then, with probability at least 0.9 over the choice of the dataset S, there exists an index $\widehat{j} \in [d]$ such that $\widehat{j} \neq j^*$ and $x[\widehat{j}] = 0$ for all $z \in S$ for which $y = 1$ and $x[\widehat{j}] = 1$ for all $z \in S$ for which $y = -1$.*

Proof. Let S denote a set of n samples drawn i.i.d. from $\mathcal{D}(\frac{1}{10}, \frac{1}{2}, a)$. We define the sets S^+ and S^- as follows:

$$\begin{aligned} S^+ &:= \{z_i \in S \mid y_i = +1\}, \\ S^- &:= \{z_i \in S \mid y_i = -1\}. \end{aligned} \tag{11}$$

Let E_j denote the event that the coordinate $x[j] = 0$ for all $z = (x, y) \in S^+$, and $x[j] = 1$ for all $z = (x, y) \in S^-$. Since, for each sample, $x[j]$ is drawn independently from $\mathcal{B}(1/2)$, we have that

$$\Pr(E_j) = \frac{1}{2^n}.$$

Next, let E denote the event that there exists some $\widehat{j} \in [d] \setminus \{j^*\}$ for which $x[\widehat{j}] = 0$ for all $z = (x, y) \in S^+$, and $x[\widehat{j}] = 1$ for all $z = (x, y) \in S^-$. We thus note that,

$$\Pr(E^c) = \Pr\Big(\bigcap_{j \in [d] \setminus \{j^*\}} E_j^c\Big) \overset{(i)}{=} \prod_{j \in [d] \setminus \{j^*\}} \Pr\big(E_j^c\big)$$

where the equality in (i) follows from the fact that the events $\{E_j\}$ are mutually independent to each other. Plugging in the bound for $\Pr(E_j^c)$, we get

$$\Pr(E^c) = \left(1 - \frac{1}{2^n}\right)^{d-1}.$$

Using the fact that $1 - a \leq e^{-a}$ for $a \geq 0$, we get

$$\Pr(E^c) \leq e^{-(d-1)/2^n} \leq 0.1,$$

where the second inequality above holds for $n \leq \log_2((d-1)/\ln(10))$. □

B.2 Proof of Theorem 2

We now have all the tools required to prove Theorem 2, which states for any regularization function $R(w)$, there exists an instance of a SCO problem for which RERM fails to find $O(1)$-suboptimal solution, in expectation.

Proof of Theorem 2. In this proof, we will assume that $n \geq 300$, $d \geq \log(10)2^n + 1$ and the initial point $w_1 = 0$. Assume, for the sake of contradiction, that there exists a regularizer $R : \mathbb{R}^d \to \mathbb{R}$

such that for any SCO instance over $\mathbb{R}^d$ satisfying Assumption II with[2] $L, B = 1$ the expected suboptimality gap for RERM is at most $\varepsilon = 1/20000$. Then, by Markov's inequality, with probability at least 0.9 over the choice of sample set S, the suboptimality gap is at most 10ε.

Before delving into the proof, we first define some additional notation based on the regularization function $R(\cdot)$. For $j \in [d]$, define the points w_j^* such that

$$w_j^* \in \operatorname*{argmin}_{w \text{ s.t. } \|w - e_j\| \le 100\varepsilon} R(w). \tag{12}$$

and define the index $j^* \in [d]$ such that $j^* \in \operatorname{argmax}_{j \in [d]} R(w_j^*)$.

Now we will construct an instance of SCO in $d = \lceil 2^n \ln(10) + 1 \rceil$ dimensions. The instance will be based on the function $f_{(A)}$ given in (10). The data distribution of interest is $\mathcal{D}_1 := \mathcal{D}(\frac{1}{10}, \frac{1}{2}, e_{j^*})$ (see Definition 2) and suppose that the dataset $S = \{z_i\}_{i=1}^n$ is sampled i.i.d. from $\mathcal{D}_1$. The population loss $F(w)$ corresponding to $\mathcal{D}_1$ is given by

$$F(w) = \mathbb{E}_{z \sim \mathcal{D}_1}\left[f_{(A)}(w; z)\right] = 0.2\, \mathbb{E}_x[\|(w - e_{j^*}) \odot x\|].$$

Clearly, $F(w)$ is convex in w and so minimizing it is an instance of SCO. Furthermore, $f_{(A)}$ is 1-Lipschitz, and the initial point $w_1 = 0$ satisfies $\|w_1 - w^*\| \le 1$. This implies that $f_{(A)}$ satisfies both Assumptions I and II given in (2) and (3) respectively. Additionally, note that e_{j^*} is the unique minimizer of $F(\cdot)$, and as a consequence of Lemma 5, any 10ε-suboptimal minimizer w' for $F(\cdot)$ must satisfy

$$\|w' - e_{j^*}\| \le 100\varepsilon. \tag{13}$$

Using the dataset S, we define some additional sets as follows:

- Define the set S^+ as the set of all the sample points in S for which $y = +1$, i.e

$$S^+ := \{(x, y, \alpha) \in S \mid y = +1\},$$

and define the set $S^- := S \setminus S^+$.

- Define the set U as the set of all the sample points in S^+ for which $x[j^*] = 1$, i.e.

$$U := \big\{(x, y, \alpha) \in S^+ \mid x[j^*] = 1\big\}.$$

- Similarly, define the set V as the set of all the sample points in S^- for which $x[j^*] = 1$, i.e.

$$V := \big\{(x, y, \alpha) \in S^- \mid x[j^*] = 1\big\}.$$

Next, define the event E such that all of the following hold:

(a) $|S^-| \ge 7n/20$.

(b) $|U| \le 39n/100$.

(c) $|V| \ge 7n/50$.

(d) There exists $\widehat{j}$ such that $\widehat{j} \ne j^*$ and $x[\widehat{j}] = 0$ for all $z \in S^+$ and $x[\widehat{j}] = 1$ for all $z \in S^-$.

(e) RERM with regularization $R(\cdot)$ and using the dataset S returns an 10ε-suboptimal solution for the test loss $F(w)$.

Using Hoeffding's inequality (Lemma 1) and the fact that $\mathbb{E}[|S^-|] = 2n/5$, $\mathbb{E}[|U|] = 3n/10$, and $\mathbb{E}[|V|] = n/5$, we get that parts (a), (b) and (c) hold simultaneously with probability at least 0.3 for $n \ge 300$. Furthermore, by Lemma 6, part (d) holds with probability at least 0.9 since $d \ge \ln(10)2^n + 1$, and part (e) holds with probability 0.9 by our assumption. Hence, the event E occurs with probability at least 0.1. In the following, we condition on the occurrence of the event E.

[2] The specific values $L, B = 1$ are used for convenience; the proof immediately yields the required SCO instances for arbitray values of L and B that are $O(1)$ in magnitude as a function of n.

Consider the point $w^*_{\widehat{j}}$ defined in (12) corresponding to the coordinate $\widehat{j}$ (that occurs in event E). By definition, we have that $\|w^*_{\widehat{j}} - e_{\widehat{j}}\| \leq 100\varepsilon$. Thus, we have

$$\|w^*_{\widehat{j}} - e_{j^*}\| \geq \|e_{\widehat{j}} - e_{j^*}\| - \|w^*_{\widehat{j}} - e_{\widehat{j}}\| \geq \sqrt{2} - 100\varepsilon > 100\varepsilon.$$

The first line above follows from Triangle inequality, and the second line holds because $\widehat{j} \neq j^*$ and because $\varepsilon = 1/20000$. As a consequence of the above bound and the condition in (13), we get that the point $w^*_{\widehat{j}}$ is not an 10ε-suboptimal point for the population loss $F(\cdot)$, and thus would not be the solution of the RERM algorithm (as the RERM solution is 10ε-suboptimal w.r.t $F(\cdot)$). Since, any RERM must satisfy condition (13), we have that

$$\begin{aligned}
\widehat{F}(w^*_{\widehat{j}}) + R(w^*_{\widehat{j}}) &> \min_{w\colon \|w - e_{j^*}\| \leq 100\varepsilon} \big(\widehat{F}(w) + R(w)\big) \\
&\geq \min_{w\colon \|w - e_{j^*}\| \leq 100\varepsilon} \widehat{F}(w) + \min_{w\colon \|w - e_{j^*}\| \leq 100\varepsilon} R(w) \\
&\geq -100\varepsilon + R(w^*_{j^*}),
\end{aligned} \tag{14}$$

where $\widehat{F}(w) = \frac{1}{n}\sum_{i=1}^n f(w; z_i)$ denotes the empirical loss on the dataset S, and the inequality in the last line follows from the definition of the point $w^*_{j^*}$ and by observing that $\widehat{F}(w) \geq -100\varepsilon$ for any w for which $\|w - e_{j^*}\| \leq 100\varepsilon$ since $\widehat{F}$ is 1-Lipschitz and $\widehat{F}(e_{j^*}) = 0$. For the left hand side, we note that

$$\begin{aligned}
\widehat{F}(w^*_{\widehat{j}}) &= \frac{1}{n}\sum_{z \in S} y\|(w^*_{\widehat{j}} - e_{j^*}) \odot x\| \\
&\overset{(i)}{\leq} 100\varepsilon + \frac{1}{n}\sum_{z \in S} y\|(e_{\widehat{j}} - e_{j^*}) \odot x\| \\
&\overset{(ii)}{\leq} 100\varepsilon + \frac{1}{n}\Big(\sum_{z \in S^+} \|(e_{\widehat{j}} - e_{j^*}) \odot x\| - \sum_{z \in S^-} \|(e_{\widehat{j}} - e_{j^*}) \odot x\|\Big) \\
&\overset{(iii)}{\leq} 100\varepsilon + \frac{1}{n}\Big(\sum_{z \in S^+} |x[j^*]| - \sum_{z \in S^+} \sqrt{1 + (x[j^*])^2}\Big) \\
&\overset{(iv)}{\leq} 100\varepsilon + \frac{1}{n}\Big(\sum_{z \in U} 1 - \sum_{z \in V} \sqrt{2} - \sum_{z \in S^- \setminus V} 1\Big) \\
&= 100\varepsilon + \frac{1}{n}\Big(|U| - (\sqrt{2} - 1)|V| - |S^-|\Big)
\end{aligned}$$

where the inequality (i) is due to the definition of the point $w^*_{\widehat{j}}$, and because $\widehat{F}$ is 1-Lipschitz, the inequality in (ii) follows from the definition of the sets S^+ and S^-, the inequality (iii) holds due to the fact that $x[\widehat{j}] = 0$ for all $(x, \alpha) \in S^+$, and $x[\widehat{j}] = 1$ for all $x \in S^-$, and finally, the inequality (iv) follows from the definition of the sets U and V. Plugging the bounds on $|U|$, $|V|$ and $|S^-|$ from the event E defined above, we get that:

$$\widehat{F}(w^*_{\widehat{j}}) \leq 100\varepsilon - \frac{3}{200}. \tag{15}$$

Combining the bounds in (14) and (15) and rearranging the terms, we get that

$$200\varepsilon \geq \frac{3}{200} + R(w^*_{j^*}) - R(w^*_{\widehat{j}}) \geq \frac{3}{200}.$$

where the second inequality above holds because $j^* \in \operatorname{argmax}_{j \in [d]} R(w^*_j)$ (by definition). Thus, $\varepsilon \geq 3/40000 > 1/20000$, a contradiction, as desired.[3]

Finally, note that since the function $f_{(A)}$ is $1-$Lipschitz, the initial point $w_1 = 0$ satisfies $\|w_1 - w^*\| = \|w_1 - e_{j^*}\| \leq 1$ and $F(w)$ is convex, due to Theorem 1, SGD run with a step size of $1/\sqrt{n}$ for n steps learns at a rate of $O(1/\sqrt{n})$. □

[3] The constant in the lower bound for ε can be improved further via a tighter analysis for the sizes of the sets $|U|$, $|V|$ and $|S^-|$ in the event E.

B.3 Proof of Corollary 1

The following proof closely follows along the lines of the proof of Theorem 2 above.

Proof. In this proof, we will assume that $n \geq 300$, $d \geq \log(10)2^n + 1$ and the initial point $w_1 = 0$. Assume, for the sake of contradiction, that there exists a regularizer $R : \mathbb{R}^d \to \mathbb{R}$ such that for any SCO instance over $\mathbb{R}^d$ satisfying Assumption II with $L, B = 1$, there exists a regularization parameter λ such that the expected suboptimality gap for the point $w_{\mathrm{RERM}} = \mathrm{argmin}_{w \in \mathcal{W}} F_S(w) + \lambda R(w)$ is at most $\varepsilon = 1/20000$. Then, by Markov's inequality, with probability at least 0.9 over the choice of sample set S, the suboptimality gap is at most 10ε.

We next define the functions $f(w; z)$, the distribution $\mathcal{D}_1$ and the population loss function $F(w)$ identical to the the corresponding quantities in the proof of Theorem 2. Furthermore, we also define the points w_j for $j \in [d]$, the coordinates j^* and the event E identical to the corresponding definitions in the proof of Theorem 2. As we argued in the proof of Theorem 2 above, we note that any 10ε-suboptimal minimizer w' for $F(\cdot)$ must satisfy

$$\|w' - e_{j^*}\| \leq 100\varepsilon. \tag{16}$$

Thus, the point $w^*_{\widehat{j}}$ (where $\widehat{j}$ is defined in the event E) is not an 10ε-suboptimal point for the population loss $F(\cdot)$, and thus for any regularization parameter λ (that can even depend on the dataset S) should not correspond to the RERM solution (as the point w_{RERM} is 10ε-suboptimal with respect to $F(\cdot)$). Since, any RERM must satisfy condition (16), we have that for any regularization parameter λ that can even depend on the dataset S,

$$\begin{aligned}
\widehat{F}(w^*_{\widehat{j}}) + \lambda R(w^*_{\widehat{j}}) &> \min_{w \colon \|w - e_{j^*}\| \leq 100\varepsilon} \big(\widehat{F}(w) + \lambda R(w)\big) \\
&\geq \min_{w \colon \|w - e_{j^*}\| \leq 100\varepsilon} \widehat{F}(w) + \min_{w \colon \|w - e_{j^*}\| \leq 100\varepsilon} \lambda R(w) \\
&\geq -100\varepsilon + \lambda R(w^*_{j^*}), \qquad (17)
\end{aligned}$$

where the inequality in the last line follows from the definition of the point $w^*_{j^*}$ and by observing that $\widehat{F}(w) \geq -100\varepsilon$ for any w for which $\|w - e_{j^*}\| \leq 100\varepsilon$ since $\widehat{F}$ is 1-Lipschitz and $\widehat{F}(e_{j^*}) = 0$. For the left hand side, similar to the proof of Theorem 2, we upper bound

$$\widehat{F}(w^*_{\widehat{j}}) \leq 100\varepsilon - \frac{3}{200}. \tag{18}$$

Combining the bounds in (17) and (18) and rearranging the terms, we get that

$$200\varepsilon \geq \frac{3}{200} + \lambda R(w^*_{j^*}) - \lambda R(w^*_{\widehat{j}}) \geq \frac{3}{200}.$$

where the second inequality above holds because $j^* \in \mathrm{argmax}_{j \in [d]} R(w^*_j)$ (by definition). Thus, $\varepsilon \geq 3/40000 > 1/20000$, a contradiction, as desired.

We remark that in the above proof the regularization parameter λ can be arbitrary and can even depend on the dataset S, but λ should not depend on w as this will change the definition of the points w^*_j. Only the regularization function $R(\cdot)$ is allowed to depend on w. □

C Missing proofs from Section 4

In this section, we first provide a learning problem for which, for any $\eta \in [1/n^2, 1)$ and $T \in [1, n^3)$, the point $\widehat{w}^{\mathrm{GD}}_{\eta,T}$ returned by running GD algorithm with step size η for T time steps has the lower bound

$$\mathbb{E}[F(\widehat{w}^{\mathrm{GD}}_{\eta,T})] - \inf_{w \in \mathbb{R}^d} F(w) = \Omega\Big(\frac{1}{\log^4(n)} \min\Big\{\eta\sqrt{T} + \frac{1}{\eta T} + \frac{\eta T}{n}, 1\Big\}\Big). \tag{19}$$

Then, we will provide a learning setting in which GD algorithm when run with step size $\eta \leq 1/64n^{5/4}$ has the lower bound of $\Omega(1/n^{3/8})$ for all $T \geq 0$. Our final lower bound for GD algorithm, given in Theorem 3 then follows by considering the above two lower bound constructions together.

C.1 Modification of Amir et al. [2021] lower bound

Amir et al. [2021] recently provided the following lower bound on the performance of GD algorithm.

Theorem 9 (Modification of Theorem 3.1, Amir et al. [2021])**.** *Fix any n, $\bar{\eta}$ and $\bar{T}$. There exists a function $f(w; z)$ that is 4-Lipschitz and convex in w, and a distribution $\mathcal{D}$ over the instance space $\mathcal{Z}$, such that for any $\eta \in [\bar{\eta}, \bar{\eta}\sqrt{3/2})$ and $T \in [\bar{T}, 2\bar{T})$, the point $\widehat{w}^{GD}[\eta, T]$ returned by running GD algorithm with a step size of η for T steps has excess risk*

$$\mathbb{E}[F(\widehat{w}^{GD}[\eta, T])] - \inf_{w \in \mathbb{R}^d} F(w) \geq \Omega\Big(\min\Big\{\eta\sqrt{T} + \frac{1}{\eta T}, 1\Big\}\Big), \tag{20}$$

where $F(w) := \mathbb{E}_{z\sim\mathcal{D}}[f(w; z)]$. Additionally, there exists a point $w^* \in \operatorname{argmin}_w F(w)$ such that $\|w^*\| \leq 1$.

Proof. We refer the reader to Amir et al. [2021] for full details about the loss function construction and the lower bound proof, and discuss the modifications below:

- Amir et al. [2021] provide the lower bound for a fixed η and T. In particular, their loss function construction given in eqn-(16) (in their paper) consists of parameters γ_1, γ_2, γ_3, $\varepsilon_1, \dots, \varepsilon_3$ and d which are chosen depending η and T. However, a slight modification of these parameters easily extends the lower bound to hold for all $\eta \in [\bar{\eta}, \bar{\eta}\sqrt{3/2})$ and $T \in [\bar{T}, 2\bar{T})$. The modified parameter values are:
  - Set $d = \frac{2^n}{3\bar{\eta}^2}$.
  - Set γ_1 such that $\gamma_1(1 + 5\bar{\eta}\bar{T})\sqrt{d} \leq \frac{1}{64}\min\{\bar{\eta}\sqrt{\bar{T}}, \frac{1}{3}\}$.
  - Set $\gamma_2 = 2\gamma_1\bar{\eta}\bar{T}$ and $\gamma_3 = 1$.
  - Set $0 < \varepsilon_1 < \cdots < \varepsilon_d < \frac{\gamma_1\bar{\eta}}{2n}$.

  Their proof of the lower bound in Theorem 3.1 follows using Lemma 4.1, Theorem 6.1, Lemma 6.2 and Claim 6.3 (in their paper respectively). We note that the above parameter setting satisfies the premise of Lemma 4.1 and Claim 6.3 for all $\eta \in [\bar{\eta}, \bar{\eta}\sqrt{3/2})$ and $T \in [\bar{T}, 2\bar{T})$. Furthermore, it can be easily verified that the proofs of Theorem 6.1 and Lemma 6.2 also follow through with the above parameter setting for all $\eta \in [\bar{\eta}, \bar{\eta}\sqrt{3/2})$ and $T \in [\bar{T}, 2\bar{T})$. Thus, the desired lower bound holds for all $\eta \in [\bar{\eta}, \bar{\eta}\sqrt{3/2})$ and $T \in [\bar{T}, 2\bar{T})$.
- Amir et al. [2021] consider GD with projection on the unit ball as their training algorithm. However, their lower bound also holds for GD without the projection step (as we consider in our paper; see (5)). In fact, as pointed out in the proof of Lemma 4.1 (used to Prove Theorem 3.1) in their paper, the iterates w_t generated by the GD algorithm never leave the unit ball. Thus, the projection step is never invoked, and GD with projection and GD without projection produce identical iterates.

The upper bound on $\|w^*\|$ also follows from the provided proof in Amir et al. [2021]. As they show in the proof of Lemma 4.1 (page 28, last paragraph in the proof), all the GD iterates as well as the minimizer point w^* lie a ball of unit radius.

Finally, note that the loss function in Theorem 9 is 4-Lipschitz, and bounded over the unit ball which contains all the iterates generated by GD algorithm. Thus, a simple application of the Markov's inequality suggests that the lower bound in (20) also holds with constant probability. However, for our purposes, the in-expectation result suffices. □

The loss function construction in Theorem 9 depends on $\bar{\eta}$ and $\bar{T}$, and thus the lower bound above only holds when the GD algorithm is run with step size $\eta \in [\bar{\eta}, \bar{\eta}\sqrt{3/2})$ and for $T \in [\bar{T}, 2\bar{T})$. In the following, we combine together multiple such lower bound instances to get an anytime and any stepsize guarantee.

Theorem 10. *Fix any $n \geq 200$. There exists a function $f(w; z)$ that is 1-Lipschitz and convex in w, and a distribution $\mathcal{D}$ over $\mathcal{Z}$ such that, for any $\eta' \in [1/n^2, 1)$ and $T' \in [1, n^3)$, the point $\widehat{w}^{GD}[\eta', T']$*

returned by running GD algorithm with a step size of η' for T' steps satisfies the lower bound

$$\mathbb{E}[F(\widehat{w}^{GD}[\eta', T'])] - \inf_{w \in \mathbb{R}^d} F(w) \geq \Omega\Big(\frac{1}{\log^4(n)} \min\Big\{\eta'\sqrt{T'} + \frac{1}{\eta' T'}, 1\Big\}\Big), \tag{21}$$

where $F(w) := \mathbb{E}_{z\sim\mathcal{D}}[f(w;z)]$. Additionally, there exists a minimizer $w^* \in \operatorname{argmin}_w F(w)$ such that $\|w^*\| = O(1)$.

Proof. Set $\gamma := \sqrt{3/2}$. We consider a discretization of the intervals $[1/n^3, 1)$ for the step sizes (we take a slightly larger interval that the domain $[1/n^2, 1)$ of the step size) and $[1, n^3)$ for the time steps respectively. Define the set

$$\mathcal{N} := \Big\{\frac{1}{n^3}, \frac{\gamma}{n^3}, \frac{\gamma^2}{n^3}, \dots, \frac{\gamma^{\lceil 3\log(n)/\log(\gamma)\rceil}}{n^3}\Big\}$$

of step sizes such that for any $\eta' \in [1/n^3, 1)$, there exists an $\eta \in \mathcal{N}$ that satisfies $\eta \leq \eta' < \gamma\eta$. Similarly, define the set

$$\mathcal{T} := \big\{1, 2, 4, \dots, 2^{\lceil 3\log(n)\rceil}\big\}$$

of time steps such that for any $T' \in [1, n^3)$, there exists a $T \in \mathcal{T}$ that satisfies $T \leq T' < 2T$. Further, define $M = |\mathcal{N}||\mathcal{T}|$. Clearly, $M = \lceil 3\log(n)/\log(\gamma)\rceil \cdot \lceil 3\log(n)\rceil$ and for $n \geq 20$ satisfies the bound $40\log^2(n) \leq M \leq 80\log^2(n)$.

In the following, we first define the component function $f_{\eta,T}$ for every $\eta \in \mathcal{N}$ and $T \in \mathcal{T}$. We then define the loss function f and show that it is convex and Lipschitz in the corresponding optimization variable. Finally, we show the lower bound for GD for this loss function f for any step size $\eta \in [1/n^2, 1)$ and time steps $T \in [1, n^3)$.

Component functions. For any $\eta \in \mathcal{N}$ and $T \in \mathcal{T}$, let $\bar{w}_{\eta,T}$, $z_{\eta,T}$, $f_{\eta,T}$ and $\mathcal{D}_{\eta,T}$ denote the optimization variable, data instance, loss function and the corresponding data distribution in the lower bound construction in Theorem 9 where $\bar{\eta}$ and $\bar{T}$ are set as η and T respectively. We note that:

(a) For any $z_{\eta,T}$, the function $f_{\eta,T}(\bar{w}_{\eta,T}; z_{\eta,T})$ is 4-Lipschitz and convex in $\bar{w}_{\eta,T}$.

(b) For any $\eta'' \in [\eta, \gamma\eta)$ and $T'' \in [T, 2T)$, the output point[4] $\widehat{w}^{\text{GD}}_{\eta,T}[\eta'', T'']$ returned by running GD algorithm with a step size of η'' for T'' steps has excess risk

$$\mathbb{E}\big[F_{\eta,T}\big(\widehat{w}^{\text{GD}}_{\eta,T}[\eta'', T'']\big)\big] - \inf_{\bar{w}_{\eta,T}} F_{\eta,T}(\bar{w}_{\eta,T}) \geq \Omega\Big(\min\Big\{\eta''\sqrt{T''} + \frac{1}{\eta'' T''}, 1\Big\}\Big), \tag{22}$$

where the population loss $F_{\eta,T}(\bar{w}_{\eta,T}) := \mathbb{E}_{z_{\eta,T}\sim\mathcal{D}_{\eta,T}}[f_{\eta,T}(\bar{w}_{\eta,T}; z_{\eta,T})]$.

(c) There exists a point $\bar{w}^*_{\eta,T} \in \operatorname{argmin} F_{\eta,T}(\bar{w}_{\eta,T})$ such that $\|\bar{w}^*_{\eta,T}\| \leq 1$.

Lower bound construction. We now present our lower bound construction:

- *Optimization variable*: For any η and T, define $w_{\eta,T} := \bar{w}_{\eta,T}/\log(n)$. The optimization variable w is defined as the concatenation of the variables $(w_{\eta,T})_{\eta\in\mathcal{N}, T\in\mathcal{T}}$.
- *Data instance*: z is defined as the concatenation of the data instances $(z_{\eta,T})_{\eta\in\mathcal{N}, T\in\mathcal{T}}$.
- *Data distribution*: $\mathcal{D}$ is defined as the cross product of the distributions $(\mathcal{D}_{\eta,T})_{\eta\in\mathcal{N}, T\in\mathcal{T}}$. Thus, for any $\eta \in \mathcal{N}$ and $T \in \mathcal{T}$, the component $z_{\eta,T}$ is sampled independent from $\mathcal{D}_{\eta,T}$.
- *Loss function*: is defined as

$$f(w; z) = \frac{1}{M} \sum_{\eta\in\mathcal{N}, T\in\mathcal{T}} f_{\eta,T}(\bar{w}_{\eta,T}; z_{\eta,T}), \tag{23}$$

where recall that $\bar{w}_{\eta,T} = w_{\eta,T}\log(n)$. Additionally, we define the population loss $F(w) := \mathbb{E}_{z\sim\mathcal{D}}[f(w;z)]$.

[4] We use the notation $\widehat{w}^{\text{GD}}_{\eta,T}[\eta'', T'']$ to denote the value of the variable $\bar{w}_{\eta,T}$ computed by running GD algorithm with the step size η'' for T'' time steps. The subscripts η, T are used to denote the fact that the corresponding variables are associated with the loss function construction in Theorem 9 where $\bar{\eta}$ and $\bar{T}$ are set as η and T respectively.

f is convex and 1-Lipschitz. Since, for any $\eta \in \mathcal{N}$ and $T \in \mathcal{T}$, the function $f_{\eta,T}(\bar{w}_{\eta,T}; z_{\eta,T})$ is convex in $\bar{w}_{\eta,T}$ for every $z_{\eta,T}$, and since $w_{\eta,T} = \bar{w}_{\eta,T}/\log(n)$, we immediately get that the function $f(w;z)$ is also convex in w for every z. Furthermore, for any w, w' and z, we have

$$\begin{aligned}
|f(w;z) - f(w';z)| &\overset{(i)}{\le} \frac{1}{M} \sum_{\eta\in\mathcal{N},T\in\mathcal{T}} \left| f_{\eta,T}(\bar{w}_{\eta,T}; z_{\eta,T}) - f_{\eta,T}(\bar{w}'_{\eta,T}; z_{\eta,T}) \right| \\
&\overset{(ii)}{\le} \frac{4\log(n)}{M} \sum_{\eta\in\mathcal{N},T\in\mathcal{T}} \|w_{\eta,T} - w'_{\eta,T}\| \\
&= \frac{4\log(n)}{M} \sum_{\eta\in\mathcal{N},T\in\mathcal{T}} \sqrt{\|w_{\eta,T} - w'_{\eta,T}\|^2} \\
&\overset{(iii)}{\le} 4\log(n) \sqrt{\frac{1}{M} \sum_{\eta\in\mathcal{N},T\in\mathcal{T}} \|w_{\eta,T} - w'_{\eta,T}\|^2} \\
&= \frac{4\log(n)}{\sqrt{M}} \sqrt{\sum_{\eta\in\mathcal{N},T\in\mathcal{T}} \|w_{\eta,T} - w'_{\eta,T}\|^2} \\
&= \frac{4\log(n)}{\sqrt{M}} \sqrt{\sum_{\eta\in\mathcal{N},T\in\mathcal{T}} \|w_{\eta,T} - w'_{\eta,T}\|^2} \\
&\overset{(iv)}{=} \frac{4\log(n)}{\sqrt{M}} \sqrt{\|w - w'\|^2} \\
&\overset{(v)}{\le} \|w - w'\|,
\end{aligned}$$

where the inequality (i) follows from Triangle inequality and the inequality (ii) holds because $f_{\eta,T}$ is 4-Lipschitz in the variable $\bar{w}_{\eta,T}$ for every $\eta \in \mathcal{N}$ and $T \in \mathcal{T}$, and by using the fact that $\bar{w}_{\eta,T} = w_{\eta,T}\log(n)$. The inequality (iii) above follows from an application of Jensen's inequality and using the concavity of square root. Furthermore, the equality (iv) holds by the construction of the variable w as concatenation of the variables $(w_{\eta,T})_{\eta\in\mathcal{N},T\in\mathcal{T}}$. Finally, the inequality (v) follows from the fact that $M \ge 16\log^2(n)$ for $n \ge 4$. Thus, the function $f(w;z)$ is 1-Lipschitz in w for every z.

Bound on the minimizer. Since the components $(w_{\eta,T})_{\eta\in\mathcal{N},T\in\mathcal{T}}$ do not interact with each other in the loss function f, any point $w^* \in \operatorname{argmin} F(w)$ satisfies:

$$\|w^*\| = \sqrt{\sum_{\eta\in\mathcal{N},T\in\mathcal{T}} \|w^*_{\eta,T}\|^2} = \frac{1}{\log(n)} \sqrt{\sum_{\eta\in\mathcal{N},T\in\mathcal{T}} \|\bar{w}^*_{\eta,T}\|^2},$$

where $\bar{w}^*_{\eta,T}$ denote the corresponding minimizers of the population loss $F_{\eta,T}$. Due to Theorem 9, we have that for any η and T, there exists a $\bar{w}^*_{\eta,T}$ that satisfies $\|\bar{w}^*_{\eta,T}\| \le 1$. Plugging these in the above, we get that there exists a w^* for which

$$\|w^*\| \le \frac{1}{\log(n)}\sqrt{M} \le 10,$$

where the second inequality follows by using the fact that $M \le 100\log^2(n)$ for $n \ge 20$.

Lower bound proof. We next provide the desired lower bound for the loss function in (23). First, note that when running GD update on the variable w, with a step size of η' and using the loss function $f(w;z)$, each of the component variables $\bar{w}_{\eta,T}$ are updated as if independently performing GD on the function $f_{\eta,T}$ but with the step size of η'/M.

Now, suppose that we run GD on the loss function f with step size $\eta' \in [1/n^2, 1)$ and for $T' \in [1, n^3)$ steps. For $n \ge 20$, the step size $\widetilde{\eta} := \eta'/M \le \eta/40\log^2(n)$ with which each component is updated clearly satisfies $\widetilde{\eta} \in [1/n^3, 1]$. Thus, by construction of the sets $\mathcal{N}$ and $\mathcal{T}$, there exists some $\bar{\eta} \in \mathcal{N}$ and $\bar{T} \in \mathcal{T}$ such that $\bar{\eta} \le \widetilde{\eta}' < \gamma\bar{\eta}$ and $\bar{T} \le T' < 2\bar{T}$. Thus, due to (23), we have that for the component function corresponding to $(\bar{\eta}, \bar{T})$, the point $\widehat{w}^{\mathrm{GD}}_{\bar{\eta},\bar{T}}[\widetilde{\eta}, T']$ returned after running GD with step size $\widetilde{\eta}$ for T' steps satisfies

$$\begin{aligned}\mathbb{E}\big[F_{\bar{\eta},\bar{T}}\big(\widehat{w}^{\mathrm{GD}}_{\bar{\eta},\bar{T}}[\widetilde{\eta}, T']\big)\big] - \inf_{\bar{w}_{\bar{\eta},\bar{T}}} F_{\bar{\eta},\bar{T}}(\bar{w}_{\bar{\eta},\bar{T}}) &\geq \Omega\Big(\min\Big\{\widetilde{\eta}\sqrt{T'} + \frac{1}{\widetilde{\eta}\, T'}, 1\Big\}\Big) \\ &\geq \Omega\Big(\frac{1}{M}\min\Big\{\eta'\sqrt{T'} + \frac{1}{\eta' T'}, 1\Big\}\Big), \qquad (24)\end{aligned}$$

where the last line holds for $\widetilde{\eta} = \eta'/M$.

Our desired lower bound follows immediately from (24). Let $\widehat{w}^{\mathrm{GD}}[\eta', T']$ be the output of running GD algorithm on the function f with step size η' and for T' steps. We have that

$$\begin{aligned}\mathbb{E}\big[F\big(\widehat{w}^{\mathrm{GD}}[\eta', T']\big)\big] - \min_{w} F(w) &= \frac{1}{M}\sum_{\eta\in\mathcal{N}, T\in\mathcal{T}}\Big(\mathbb{E}\big[F_{\eta,T}\big(\widehat{w}^{\mathrm{GD}}_{\eta,T}[\widetilde{\eta}, T']\big)\big] - \inf_{w_{\eta,T}} F_{\eta,T}(w_{\eta,T})\Big) \\ &\geq \frac{1}{M}\Big(\mathbb{E}\big[F_{\bar{\eta},\bar{T}}\big(\widehat{w}^{\mathrm{GD}}_{\bar{\eta},\bar{T}}[\widetilde{\eta}, T']\big)\big] - \inf_{w_{\bar{\eta},\bar{T}}} F_{\bar{\eta},\bar{T}}(w_{\bar{\eta},\bar{T}})\Big) \\ &= \Omega\Big(\frac{1}{M^2}\min\Big\{\eta'\sqrt{T'} + \frac{1}{\eta' T'}, 1\Big\}\Big),\end{aligned}$$

where the equality in the first line holds because the variables $\{w_{\eta,T}\}$ do not interact with each other, the inequality in the second line follows by ignoring rest of the terms which are all guaranteed to be positive, and finally, the last line follows by plugging in (24). Using the fact that $M \geq 40\log^2(n)$ in the above, we get that for any $\eta' \in [1/n^2, 1)$ and for $T' \in [1, n^3)$, the point $\widehat{w}^{\mathrm{GD}}[\eta', T']$ satisfies

$$\mathbb{E}\big[F(\widehat{w}^{\mathrm{GD}}[\eta', T'])\big] - \min_{w} F(w) = \Omega\Big(\frac{1}{\log^4(n)}\min\Big\{\eta'\sqrt{T'} + \frac{1}{\eta' T'}, 1\Big\}\Big).$$

□

C.2 Lower bound of $\eta T/n$

The lower bound in (21) already matches the first two terms in our desired lower bound in (19). In the following, we provide a function for which GD algorithm has expected suboptimality of $\eta T/n$.

Lemma 7. *Fix any n. Let $w \in \mathbb{R}$ denote the optimization variable and z denote a data sample from the instance space $\mathcal{Z} = \{-1, 1\}$. There exists a 2-Lipschitz function $f(w; z)$ and a distribution $\mathcal{D}$ over $\mathcal{Z}$ such that:*

(a) *The population loss $F(w) := \mathbb{E}_{z\sim\mathcal{D}}[f(w; z)]$ is convex in w. Furthermore, $w^* = 0$ is the unique minimizer of the population loss $F(w)$.*

(b) *For any η and T, the point $\widehat{w}^{GD}[\eta, T]$ returned by running GD with a step size of η for T steps satisfies*

$$\mathbb{E}\big[F(\widehat{w}^{GD}[\eta, T])\big] - \inf_{w\in\mathbb{R}} F(w) \geq \Omega\Big(\frac{\eta T}{n}\Big).$$

Proof. For $z \in \{-1, 1\}$ and $w \in \mathbb{R}$, define the instance loss function f as

$$f(w; z) := \big(\frac{1}{4\sqrt{n}} + z\big)|w|.$$

Define the distribution $\mathcal{D}$ such that $z = +1$ or $z = -1$ with probability $1/2$ each. Clearly, $f(w; z)$ is 2-Lispchitz w.r.t. w for any $z \in \mathcal{Z}$. Furthermore, for any $w \in \mathbb{R}$,

$$F(w) = \mathbb{E}_{z\sim\mathcal{D}}[f(w; z)] = \mathbb{E}_z\Big[\big(\frac{1}{4\sqrt{n}} + z\big)|w|\Big] = \frac{1}{4\sqrt{n}}|w|,$$

where the last equality holds because $\mathbb{E}[z] = 0$. Thus, $F(w)$ is convex in w and $w^* = 0$ is the unique minimizer of the population loss $F(w)$. This proves part-(a).

We now provide a lower bound for GD algorithm. Let $S = \{z_i\}_{i=1}^n$ denote a dataset of size n sampled i.i.d. from $\mathcal{D}$. The update rule for GD algorithm from (5) implies that

$$w_{t+1}^{\text{GD}} \leftarrow w_t^{\text{GD}} - \eta \cdot \text{sign}(w_t^{\text{GD}}) \cdot \big(\frac{1}{4\sqrt{n}} + \frac{1}{n}\sum_{i=1}^n z_i\big), \tag{25}$$

and finally the returned point is given by $\widehat{w}^{\text{GD}}[\eta, T] = \frac{1}{n}\sum_{t=1}^T w_t^{\text{GD}}$.

For $i \in [n]$, define the random variables $y_i = (1 - z_i)/2$. Note that $y_i \sim \mathcal{B}(1/2)$, and thus Lemma 3 implies that

$$\sum_{i=1}^n y_i \geq \frac{n}{2} + \frac{\sqrt{n}}{4},$$

with probability at least $1/15e$. Rearranging the terms, we get that $\sum_{i=1}^n z_i \leq -\sqrt{n}/2$ with probability at least $1/15e$. Plugging this in (25), we have that with probability at least $1/15e$, for all $t \geq 0$,

$$w_{t+1}^{\text{GD}} \geq w_t^{\text{GD}} + \frac{\eta}{4\sqrt{n}}\text{sign}(w_t^{\text{GD}}).$$

Without loss of generality, assume that $w_1 > 0$. In this case, the above update rule implies that

$$w_t \geq w_1 + \frac{t\eta}{4\sqrt{n}}$$

for all $t \geq 0$, which further implies that

$$\widehat{w}^{\text{GD}}[\eta, T] = \frac{1}{T}\sum_{t=1}^T w_t \geq w_1 + \frac{\eta(T-1)}{8\sqrt{n}}.$$

Thus,

$$F(\widehat{w}^{\text{GD}}[\eta, T]) - \inf_w F(w) = F(\widehat{w}^{\text{GD}}[\eta, T]) \geq \frac{w_1}{4\sqrt{n}} + \frac{\eta(T-1)}{32n},$$

giving us the desired lower bound on the performance guarantee of the returned point $\widehat{w}^{\text{GD}}[\eta, T]$. The final in-expectation statement follows by observing that the above holds with probability at least $1/15e$. The proof follows similarly when $w_1 \leq 0$. □

We next prove the lower bound in (19) which follows by combining the lower bound construction from Theorem 9 and Lemma 7.

Theorem 11. *Fix any $n \geq 200$. There exists a 3-Lipschitz function $f(w; z)$ and a distribution $\mathcal{D}$ over z, such that:*

(a) *The population loss $F(w) := \mathbb{E}_{z\sim\mathcal{D}}[f(w; z)]$ is convex in w. Furthermore, there exists a $w^* \in \text{argmin}_w F(w)$ such that $\|w^*\| = O(1)$.*

(b) *For any $\eta \in [1/n^2, 1)$ and $T \geq 1$, the point $\widehat{w}^{GD}[\eta, T]$ returned by running GD with a step size of η for T steps has excess risk*

$$\mathbb{E}\big[F(\widehat{w}^{GD}[\eta, T])\big] - \inf_{w\in\mathbb{R}^d} F(w) = \Omega\Big(\frac{1}{\log^4(n)}\min\Big\{\eta\sqrt{T} + \frac{1}{\eta T} + \frac{\eta T}{n}, 1\Big\}\Big). \tag{26}$$

Proof. We first define some additional notation. Let $w_{(1)}$, $z_{(1)}$, $f_{(1)}$ and $\mathcal{D}_{(1)}$ denote the optimization variable, the data sample, the instance loss and the distribution over $z_{(1)}$ corresponding to the lower bound construction in Theorem 10. Additionally, let $F_{(1)}(w_{(1)})$ denote the corresponding population loss under the distribution $D_{(1)}$. We note that the function $f_{(1)}$ is 1-Lipschitz in $w_{(1)}$ for any $z_{(1)}$. Furthermore,Theorem 10 implies that $F_{(1)}(w_{(1)})$ is convex in $w_{(1)}$, there exists a minimizer $w_{(1)}^* \in \text{argmin}_{w_{(1)}} F_{(1)}(w_{(1)})$ such that $\|w_{(1)}^*\| = O(1)$, and that for any $\eta \in [1/n^2, 1)$

and $T \in [1, n^3)$, the point $\widehat{w}^{\mathrm{GD}}_{(1)}[\eta, T]$ returned by running GD algorithm with step size η for T time steps satisfies

$$\mathbb{E}\big[F_{(1)}(\widehat{w}^{\mathrm{GD}}_{(1)}[\eta, T])\big] - \inf_{w_{(1)}} F(w_{(1)}) = \Omega\Big(\frac{1}{\log^4(n)} \min\Big\{\eta\sqrt{T} + \frac{1}{\eta T}, 1\Big\}\Big). \tag{27}$$

Similarly, let $w_{(2)}$, $z_{(2)}$, $f_{(2)}$ and $\mathcal{D}_{(2)}$ denote the corresponding quantities for the lower bound construction in Lemma 7, and let $F_{(2)}(w_{(2)})$ denote the corresponding population loss under the distribution $\mathcal{D}_{(2)}$. We note that the function $f_{(2)}$ is 2-Lipschitz in $w_{(2)}$ for any $z_{(2)}$. Furthermore, Lemma 7 implies that $F_{(2)}(w_{(2)})$ is convex in $w_{(2)}$ with $w^* = 0$ being the unique minimizer, and that for any η and T the point $\widehat{w}^{\mathrm{GD}}_{(2)}[\eta, T]$ returned by running GD algorithm with step size η for T time steps satisfies

$$\mathbb{E}\big[F_{(2)}(\widehat{w}^{\mathrm{GD}}_{(2)}[\eta, T])\big] - \inf_{w_{(2)}} F(w_{(2)}) = \Omega\Big(\frac{\eta T}{n}\Big). \tag{28}$$

Our desired lower bound follows by combining the lower bound constructions from Theorem 9 and Lemma 7 respectively.

Lower bound construction. Consider the following learning setting:

- *Optimization variable*: w is defined as the concatenation of the variables $(w_{(1)}, w_{(2)})$.
- *Data instance*: z is defined as the concatenation of the data instances $(z_{(1)}, z_{(2)})$.
- *Data distribution*: $\mathcal{D}$ is defined as $\mathcal{D}_{(1)} \times \mathcal{D}_{(2)}$, i.e. $z_{(1)}$ and $z_{(2)}$ are sampled independently from $\mathcal{D}_{(1)}$ and $\mathcal{D}_{(2)}$ respectively.
- *Loss function*: is defined as

  $$f(w; z) := f_{(1)}(w_{(1)}; z_{(1)}) + f_{(2)}(w_{(2)}; z_{(2)}),$$

  Additionally, define the population loss $F(w) := \mathbb{E}_{z\sim\mathcal{D}}[f(w; z)]$.

Since, $f_{(1)}$ is 1-Lipschitz in $w_{(1)}$ and $f_{(2)}$ is 2-Lipschitz in $w_{(2)}$, we have that the function f defined above is 3-Lipschitz in w. Furthermore, the population loss

$$F(w) = F_{(1)}(w_{(1)}) + F_{(2)}(w_{(2)}),$$

is convex in w as both $F_{(1)}(w_{(1)})$ and $F_{(2)}(w_{(2)})$ are convex functions. Furthermore, since the components $(w_{(1)}, w_{(2)})$ do not interact with each other in the function f, we have that there exists a $w^* \in \operatorname{argmin}_w F(w)$ such that:

$$\|w^*\| \le \|w^*_{(1)}\| + \|w^*_{(2)}\| = O(1),$$

where $w^*_{(1)}$ denotes a minimizer of $F_{(1)}$ with $\|w^*_{(1)}\| = O(1)$ and $w^*_{(1)} = 0$ denotes the unique minimizer of $F_{(2)}$.

GD lower bound. From the construction of the function f, we note that the variables $w_{(1)}$ and $w_{(2)}$ are updated independent to each other by GD algorithm. Thus, for any $\eta \in [1/n^2, 1)$ and $T \in [1, n^3)$, the point $\widehat{w}^{\mathrm{GD}}[\eta, T]$ returned by running GD algorithm on the function f with step size η for T time steps satisfies

$$\begin{aligned}
\mathbb{E}\big[F\big(\widehat{w}^{\mathrm{GD}}[\eta, T]\big)\big] - \min_w F(w) &= \mathbb{E}\big[F_{(1)}\big(\widehat{w}^{\mathrm{GD}}_{(1)}[\eta, T]\big) + F\big(\widehat{w}^{\mathrm{GD}}_{(2)}[\eta, T]\big)\big] - \min_{w_{(1)}, w_{(2)}} F_{(1)}(w_{(1)}) + F_{(2)}(w_{(2)}) \\
&= \mathbb{E}\big[F_{(1)}\big(\widehat{w}^{\mathrm{GD}}_{(1)}[\eta, T]\big)\big] - \min_{w_{(1)}} F_{(1)}(w_{(1)}) + \mathbb{E}\big[F\big(\widehat{w}^{\mathrm{GD}}_{(2)}[\eta, T]\big)\big] - \min_{w_{(2)}} F(w_{(2)}) \\
&\overset{(i)}{=} \Omega\Big(\frac{1}{\log^4(n)} \min\Big\{\eta\sqrt{T} + \frac{1}{\eta T}, 1\Big\}\Big) + \Omega\Big(\frac{\eta T}{n}\Big) \\
&= \Omega\Big(\frac{1}{\log^4(n)} \min\Big\{\eta\sqrt{T} + \frac{1}{\eta T} + \frac{\eta T}{n}, 1\Big\}\Big),
\end{aligned}$$

where the lower bound in (i) follows from combining the lower bounds in (27) and (28).

Finally, we note that when $\eta \in [1/n^2, 1)$ and $T \geq n^3$, we have that

$$\begin{aligned}
\mathbb{E}\big[F(\widehat{w}^{\mathrm{GD}}[\eta, T])\big] - \min_{w} F(w) &\geq \mathbb{E}\big[F(\widehat{w}^{\mathrm{GD}}_{(2)}[\eta, T])\big] - \min_{(2)} F_{(2)}(w_{(2)}) \\
&= \Omega\Big(\frac{\eta T}{n}\Big) \\
&= \Omega(1),
\end{aligned}$$

where the second line follows by using the lower bound in (28), and the last line holds for $T > n^3$ because $\eta \geq 1/n^2$. Thus, the desired lower bound holds for all $\eta \in [1/n^2, 1]$ and $T \geq 1$. □

C.3 Lower bound for small step size ($\eta < 1/64n^{5/4}$)

In the following, we provide a learning setting for which GD run with step size $\eta < 1/64n^{5/4}$ has lower bound of $\Omega(1/n^{3/8})$.

Lemma 8. *Let $w \in \mathbb{R}$ denote the optimization variable, and z denote a data sample. There exists a function $f(w; z)$ and a distribution $\mathcal{D}$ over z such that:*

(a) The population loss $F(w) := \mathbb{E}_{z \sim \mathcal{D}}[f(w; z)]$ is 1-Lipschitz and convex in w. Furthermore, there exists a point $w^* \in \operatorname{argmin}_w F(w)$ such that $\|w^*\| = 1$.

(b) The variance of the gradient is bounded, i.e. $\mathbb{E}_{z \sim \mathcal{D}}[\|\nabla f(w; z) - \nabla F(w)\|^2] \leq 1$ for any $w \in \mathbb{R}$.

(c) If $\eta < 1/64n^{5/4}$, then for any $T > 0$, the point $\widehat{w}^{GD}_T$ returned by running GD algorithm with step size η for T steps satisfies

$$\mathbb{E}[F(\widehat{w}^{GD}_T)] - \min_{w \in \mathbb{R}} F(w) = \Omega\Big(\frac{1}{n^{3/8}}\Big). \tag{29}$$

Proof. Before delving into the construction of the function f, we first define some auxiliary functions and notation. Define the kink function $h(w)$ as

$$h(w) := \begin{cases} 0 & \text{if} \quad w < 0 \\ -n^{5/8} w & \text{if} \quad 0 \leq w < \frac{1}{64n^{5/4}} \\ \quad n^{5/8} w - \frac{2}{64n^{5/8}} & \text{if} \quad \frac{1}{64n^{5/4}} \leq w \leq \frac{3}{64n^{5/4}} \\ -n^{5/8} w + \frac{4}{64n^{5/8}} & \text{if} \quad \frac{3}{64n^{5/4}} < w \leq \frac{1}{16n^{5/4}} \\ 0 & \text{if} \quad \frac{1}{16n^{5/4}} \leq w \end{cases}, \tag{30}$$

and the corresponding gradients $\nabla h(w)$ as

$$\nabla h(w) := \begin{cases} 0 & \text{if} \quad w < 0 \\ -n^{5/8} & \text{if} \quad 0 \leq w < \frac{1}{64n^{5/4}} \\ n^{5/8} & \text{if} \quad \frac{1}{64n^{5/4}} \leq w \leq \frac{3}{64n^{5/4}} \\ -n^{5/8} & \text{if} \quad \frac{3}{64n^{5/4}} < w \leq \frac{1}{16n^{5/4}} \\ 0 & \text{if} \quad \frac{1}{16n^{5/4}} \leq w \end{cases}. \tag{31}$$

Additionally, define the set

$$H := \Big\{\frac{1}{4} + \frac{1}{8n^{5/4}}, \frac{1}{4} + \frac{2}{8n^{5/4}}, \ldots, \frac{3}{4}\Big\},$$

where the set H has $4n^{5/4}$ numbers from the interval $[1/4, 3/4]$ spaced at a distance of $1/8n^{5/4}$.

We now present our learning setting:

- *Data sample:* z consists of the tuple (β, y) where $\beta \in H$ and $y \in \{-1, +1\}$.

- *Data Distribution:* $\mathcal{D}$ over the instances $z = (\beta, y)$ is defined such that
$$\beta \sim \text{Uniform}(H) \qquad \text{and} \qquad y \sim \text{Uniform}(\{-1, 1\}), \tag{32}$$
where β and y are sampled independent of each other.
- *Loss function:* is defined as
$$f(w; z) := \frac{1}{n^{3/8}} \max\{-w, -1\} + y \cdot h(w + \beta), \tag{33}$$
Additionally, define the population loss $F(w) := \mathbb{E}_{z \sim \mathcal{D}}[f(w; z)]$.

We next show the desired statements for this learning setting:

(a) For the distribution $\mathcal{D}$ defined in (32), since $\mathbb{E}[y] = 0$ and y is sampled independent of β, we have that the population loss
$$F(w) = \mathbb{E}_{z \sim \mathcal{D}}[f(w; z)] = \frac{1}{n^{3/8}} \max\{-w, -1\}.$$
Clearly, $F(w)$ is 1-Lipschitz and convex is w. Additionally, the point $w^* = 1$ is a minimizer of $F(w)$.

(b) We next bound variance of the stochastic gradient. For any $w \in \mathbb{R}$, we have
$$\begin{aligned} \mathbb{E}_{z \sim \mathcal{D}}[|\nabla f(w; z) - \nabla F(w)|^2] &= \mathbb{E}_{(\beta, y)}\left[|y \nabla h(w + \beta)|^2\right] \\ &= \mathbb{E}_{\beta}\left[|\nabla h(w + \beta)|^2\right] \\ &= \mathbb{E}_{\beta}\left[\mathbb{1}\left\{\beta \in \left[w - \frac{1}{16 n^{5/4}}, w\right]\right\} \cdot n^{5/4}\right] \\ &= n^{5/4} \Pr\left(\beta \in \left[w - \frac{1}{16 n^{5/4}}, w\right]\right) \end{aligned}$$
where the equality in the first line holds because $y \in \{-1, 1\}$, and the second line follows from the construction of the function h which implies that $|\nabla h(w + \beta)| \leq n^{5/8}$ for any w and β (see (31)). Using the fact that $\beta \sim \text{Uniform}(B)$ in the above, we get that
$$\mathbb{E}_{z \sim \mathcal{D}}[|\nabla f(w; z) - \nabla F(w)|^2] \leq 1/4.$$

(c) We next show that GD algorithm when run with the step size of $\eta \leq 1/64 n^{5/4}$ fails to converge to a good solution.

Define $\beta_{(1)}, \dots, \beta_{(n)}$ such that $\beta_{(j)}$ denotes the jth smallest item in the set $\{\beta_1, \dots, \beta_n\}$, and define $y_{(j)}$ as the corresponding y variable for the random variable $\beta_{(j)}$. An application of Lemma 9 implies that with probability at least $1 - 2/n^{1/4}$, there exists a $\widehat{j} \leq \lceil \log(n)/2n^{1/4} \rceil$ such that:

(i) $\beta_{(1)} \neq \beta_{(2)} \neq \beta_{(3)} \neq \dots \neq \beta_{(\widehat{j}+1)}$, and

(ii) $y_{(\widehat{j})} = +1$.

In the following, we condition on the occurrence of the above two events. The key idea of the proof is that at the level of the empirical loss, the first $\widehat{j}$ kinks (that arise as a result of the stochastic function h) would be isolated from each other due to event-(i). Thus, the norm of the gradient of the empirical loss would be bounded by $2/n^{3/8}$ for all points before $\beta_{\widehat{j}}$. At the same time, since $y_{(\widehat{j})} = +1$ from event-(ii), the empirical loss would be flat in a small interval around $\beta_{\widehat{j}}$. As we show in the following, when GD is run on the empirical loss with step size $\eta \leq 1/64 n^{5/4}$, some GD iterate will lie in this small flat region, and after that GD will fail to make progress because the gradient is 0; hence outputting a bad solution. On the other hand, GD / SGD run with a large step size e.g. $1/\sqrt{n}$ will easily jump over these kinks and converge to a good solution. We illustrate this intuition in Figure 1, and provide the proof below.

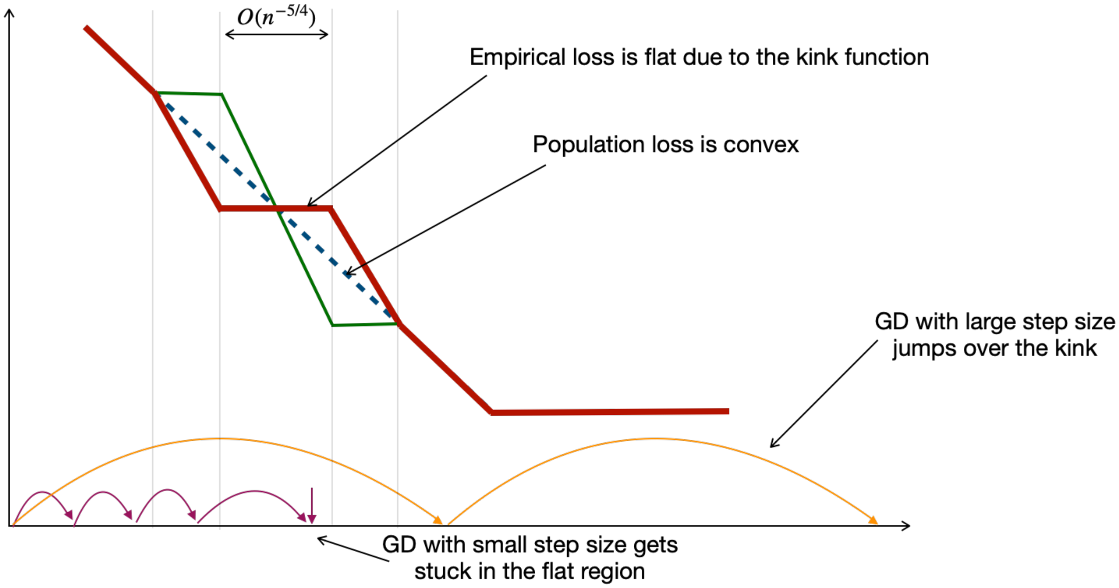
$O(n^{-5/4})$
Empirical loss is flat due to the kink function
Population loss is convex
GD with large step size jumps over the kink
GD with small step size gets stuck in the flat region

Figure 1: (Picture not drawn according to scale) The solid red line shows the empirical loss induced by the kink function when $y = +1$, the solid green line shows the empirical loss when $y = -1$, and the dotted blue line shows the convex population loss. The empirical loss when $y = +1$ has gradient 0 in a region of width $1/32n^{5/4}$. Gradient descent with step size smaller than $1/64n^{5/4}$, shown in the bottom, will get stuck in this flat region and thus fail to find a good solution. On the other hand, gradient descent with large step size will jump over the kink and find a good solution.

Recall that the empirical loss on the dataset S is given by:

$$\widehat{F}(w) = \frac{1}{n^{3/8}} \max\{-w, -1\} + \frac{1}{n} \sum_{j=1}^{n} y_j (h(w + \beta_j))$$

Note that for any $\beta \in H$, the set of w for which $h(w + \beta)$ is non-zeros is given by the interval $[\beta, \beta + 1/16n^{5/4}]$. Furthermore, any two numbers in the set H are at least $1/8n^{5/4}$ apart from each other. Thus, the event-(i) above implies that non-zero parts of the functions $\{h(w + \beta_{(1)}), \dots, h(w + \beta_{(\widehat{j})})\}$, i.e. the first $\widehat{j}$ kinks do not overlap with each other. This implies that for $w < \beta_{(\widehat{j}+1)}$,

$$\begin{aligned} |\nabla \widehat{F}(w)| &\leq \frac{1}{n^{3/8}} + \frac{1}{n} \sum_{j=1}^{n} |\nabla h(w + \beta_j)| \\ &\leq \frac{1}{n^{3/8}} + \frac{n^{5/8}}{n} \leq \frac{2}{n^{3/8}}, \end{aligned} \tag{34}$$

where the first inequality in the above follows from Triangle inequality and the second inequality holds because at most one of the n terms in the stochastic component $\sum_{j=1}^{n} y_j (h(w + \beta_j))$ above is non-zero for $w < \beta_{(\widehat{j}+1)}$. Thus, $\widehat{F}(w)$ is 1-Lipschitz for $w \in [0, \beta_{(\widehat{j}+1)})$.

Next, the event-(ii) above implies that $y_{(\widehat{j})} = 1$. Define the interval $\widetilde{\mathcal{W}} := (\beta_{(\widehat{j})} + 1/64n^{5/4}, \beta_{(\widehat{j})} + 3/64n^{5/4})$ and note that for any $w \in \widetilde{\mathcal{W}}$,

$$\begin{aligned} \nabla \widehat{F}(w) &= -\frac{1}{n^{3/8}} + \frac{1}{n} \nabla h(w + \beta_{(\widehat{j})}) \\ &= -\frac{1}{n^{3/8}} + \frac{1}{n} \cdot n^{5/8} = 0, \end{aligned} \tag{35}$$

where the first equality holds because $y_{(\widehat{j})} = +1$ and using the event-(i) above. This implies that the empirical loss $\widehat{F}(w)$ has 0 gradient for $w \in \widetilde{\mathcal{W}}$, and thus GD algorithm will stop updating if any iterate reaches $\widetilde{\mathcal{W}}$.

In the following, we will show that for $\eta \leq 1/64n^{5/4}$, GD algorithm is bound to get stuck in the interval $\widetilde{\mathcal{W}}$, and will thus fail to find a good solution. Consider the dynamics of the GD algorithm:

$$w_{t+1} \leftarrow w_t - \eta \nabla \widehat{F}(w_t),$$

where the initial point $w_1 = 0$. Suppose there exists some time τ for which $w_\tau > \beta_{(\widehat{j}+1)}$, and let $t_0 > 0$ denote the smallest such time. Thus, for any $t < t_0$ and $\eta \leq 1/64n^{5/4}$, we have that

$$|w_{t+1} - w_t| = \eta |\nabla \widehat{F}(w)| \leq \eta \leq 1/64n^{5/4},$$

where the first inequality holds due to (34). This implies that any two consecutive iterates produced by the GD algorithm for $t < t_0$ are at most $1/64n^{5/4}$ apart from each other. However, note that the interval $\widetilde{\mathcal{W}} \subseteq [0, \beta_{(\widehat{j}+1)})$ and has width of $2/64n^{5/4}$. Thus, there exists some time $t' \leq t_0$ for which $w_{t'}$ will lie in the set $\widetilde{\mathcal{W}}$. However, recall that $\nabla F(w) = 0$ for any $w \in \widetilde{\mathcal{W}}$ as shown in (35). Thus, once $w_{t'} \in \widetilde{\mathcal{W}}$, GD will not update any further implying that for all $t > t'$, $w_t = w_{t'}$. This shows via contradiction that no such time τ exists for which $w_\tau > \beta_{(\widehat{j}+1)}$.

Hence for $\eta < 1/64n^{5/4}$, all iterates $\{w_t\}_{t \geq 0}$ generated by the GD algorithm will lie in the set $[0, \beta_{(\widehat{j}+1)})$. Thus, for any $T > 1$, the returned point $\widehat{w}_T^{\mathrm{GD}}$ satisfies

$$\widehat{w}_T^{\mathrm{GD}} \leq \beta_{(\widehat{j}+1)} \leq \frac{3}{4},$$

where the second inequality holds because $\beta_{(\widehat{j}+1)} \in H$ and is thus smaller than $3/4$. Thus,

$$F(\widehat{w}_T^{\mathrm{GD}}) - \min_{w \in \mathbb{R}} F(w) \geq \frac{1}{4n^{3/8}}. \tag{36}$$

Since, the events (i) and (ii) occur simultaneously with probability at least $1 - 2/n^{1/4}$, we have that for any $T \geq 0$,

$$\mathbb{E}[F(\widehat{w}_T^{\mathrm{GD}})] - \min_{w \in \mathbb{R}} F(w) = \Omega\big(\frac{1}{n^{3/8}}\big).$$

□

Lemma 9. *Suppose $(\beta_1, y_1), \dots, (\beta_n, y_n)$ be n samples drawn independently from Uniform(H) × Uniform$(\{-1, 1\})$, where the set $H := \{\frac{1}{4} + \frac{1}{8n^{5/4}}, \frac{1}{4} + \frac{2}{8n^{5/4}}, \dots, \frac{3}{4}\}$. Further, define $\beta_{(j)}$ to denote the jth smallest number in the set $\{\beta_1, \dots, \beta_n\}$, and define $y_{(j)}$ as the corresponding y variable for $\beta_{(j)}$. Then, with probability at least $1 - 2/n^{1/4}$, there exists a $\widehat{j} \leq \lceil \log(n)/2n^{1/4} \rceil$ such that:*

(i) $\beta_{(1)} \neq \beta_{(2)} \neq \beta_{(3)} \neq \cdots \neq \beta_{(\widehat{j}+1)}$, *and*

(ii) $y_{(\widehat{j})} = +1$.

Proof. Let $k = \lceil \log(n)/2n^{1/4} \rceil$. We give the proof by translating our problem into a “ball and bins" problem.

Let the set H denote $m = 4n^{5/4}$ distinct bins, and let there be n distinct balls. Each of the n balls are tossed independently into one of the m bins drawn uniformly at random. For any i, j, the event where the ball j is tossed into the bin i corresponds to the event that $\beta_j = \frac{1}{4} + \frac{i}{8n^{5/4}}$. In addition to this, the balls are such that they change color after they are tossed into a bin. In particular, after being tossed into a bin, a ball changes color to either red or blue with equal probability. For any i, j, the event where the ball j takes red color corresponds to the event that $y_j = 1$; Similarly, blue color corresponds to $y = -1$. Thus, in the balls and bins model, the position of the n balls in the bins and their corresponding colors after tossing reveals the value of the random variables $(\beta_1, y_1), \ldots, (\beta_n, y_n)$.

In the following, we first show that with probability at least $1 - \frac{2}{n^{1/4}}$, the first k bins will have at most one ball each, and there will be some bin amongst the first k bins that contains a red color ball. Define E_i to denote the event that the bin i gets more than one ball. Thus,

$$\begin{aligned}\Pr(E_i) &= 1 - \Pr(\text{Bin } i \text{ has 0 or 1 ball in it}) \\ &= 1 - \Big(1 - \frac{1}{n^{5/4}}\Big)^n - n\Big(1 - \frac{1}{n^{5/4}}\Big)^{n-1}\frac{1}{n^{5/4}} \\ &\leq 1 - \Big(1 - \frac{1}{n^{1/4}}\Big) - \frac{1}{n^{1/4}}\Big(1 - \frac{n-1}{n^{5/4}}\Big) \\ &= \frac{n-1}{n^{3/2}} \leq \frac{1}{\sqrt{n}} \end{aligned} \tag{37}$$

where the first inequality in the above follows from the fact that $(1-\alpha)^n \leq 1 - \alpha n$ for any $\alpha \geq -1$ and $n \geq 1$. Let A_k denote the event that there exists some bin among the first k bins that has more than one ball. Taking a union bound, we get

$$\Pr(A_k) = \Pr(\cup_{i=1}^k E_i) \leq \sum_{i=1}^k \Pr(E_i) \leq \frac{k}{\sqrt{n}}.$$

Next, let B_k denote the event that there is no red ball in the first k bins after n tosses. When we throw a ball, the probability that it will fall in the first k bins is given by $k/n^{5/4}$, and the probability that it takes the color red is $1/2$. Since the bin chosen and the final color are independent of each other for each ball, the probability that a ball falls into the first k bins and is of red color is given by $k/2n^{5/4}$. Furthermore, since the balls are thrown independent to each other, the probability that if we throw n balls, no ball falls into the first k bins that takes the red color is given by

$$\Pr(B_k) = \Big(1 - \frac{k}{2n^{5/4}}\Big)^n.$$

Finally, let us define C_k to denote the event that there is at most one ball per bin amongst the first k bins, and that there exists a bin amongst the first k bins with a red ball in it. By the definition of the events A_k and B_k, we note that

$$\begin{aligned}\Pr(C_k) &= \Pr(A_k^c \cap B_k^c) \\ &= 1 - \Pr(A_k \cup B_k) \\ &\geq 1 - \Pr(A_k) - \Pr(B_k) \\ &\geq 1 - \frac{k}{\sqrt{n}} - \Big(1 - \frac{k}{2n^{5/4}}\Big)^n \\ &\geq 1 - \frac{k}{\sqrt{n}} - e^{-k/2n^{1/4}}, \end{aligned}$$

where the first inequality in the above follows from the union bound, the second inequality holds by plugging in the corresponding bounds for $P(A_k)$ and $P(B_k)$, and the last line is due to the fact that $(1+\alpha)^n \leq e^{\alpha n}$ for any α. Plugging in the value of $k = \lceil \log(n)/2n^{1/4} \rceil$ in the above, we get that

$$\Pr(C_k) \geq 1 - \frac{2}{n^{1/4}}.$$

Thus, with probability at least $1 - 2/n^{1/4}$, the first k bins have at most one ball each, and there exists some bin $\widehat{j} \in [k]$ that contains a red ball; when this happens, the corresponding random variables $(\beta_1, y_1), \ldots, (\beta_n, y_n)$ satisfy:

(a) $\beta_{(1)} \neq \beta_{(2)} \neq \beta_{(3)} \neq \cdots \neq \beta_{(\widehat{j}+1)}$, and

(b) $y_{(\widehat{j})} = +1$.

□

C.4 Proof of Theorem 3

In this section, we combine the lower bound constructions in Theorem 11 and Lemma 8 to provide an instance of a SCO problem on which, for any step size η and time steps T, GD has the lower bound of $\Omega(1/n^{5/12})$.

Proof of Theorem 3 . Throughout the proof, we assume that $n \geq 200$ and the initial point $w_1 = 0$.

We first define some additional notation: Let $w_{(1)}$, $z_{(1)}$, $f_{(1)}$ and $\mathcal{D}_{(1)}$ denote the optimization variable, the data sample, the instance loss and the distribution over $z_{(1)}$ corresponding to the lower bound construction in Theorem 11. The statement of Theorem 11 implies that:

(a) $f_{(1)}(w_{(1)}; z_{(1)})$ is 3-Lipschitz in the variable $w_{(1)}$ for any $z_{(1)}$.

(b) The population loss $F_{(1)}(w_{(1)}) := \mathbb{E}_{z_{(1)} \sim \mathcal{D}_{(1)}}[f(w_{(1)}; z_{(1)})]$ is 3-Lipschitz and convex in $w_{(1)}$. Additionally, there exists a point $w^*_{(1)} \in \operatorname{argmin}_{w_{(1)}} F_{(1)}(w_{(1)})$ such that $\|w^*_{(1)}\| = O(1)$.

(c) For any $\eta \in [1/n^2, 1)$ and $T \geq 1$, the point $\widehat{w}^{\text{GD}}_{(1)}[\eta, T]$ returned by running GD algorithm with a step size of η for T time steps satisfies:

$$\mathbb{E}\big[F_{(1)}\big(\widehat{w}^{\text{GD}}_{(1)}[\eta, T]\big)\big] - \min_{w_{(1)}} F_{(1)}(w_{(1)}) = \Omega\Big(\frac{1}{\log^4(n)} \min\Big\{\eta\sqrt{T} + \frac{1}{\eta T} + \frac{\eta T}{n}, 1\Big\}\Big). \quad (38)$$

Similarly, let $w_{(2)}$, $z_{(2)}$, $f_{(2)}$ and $\mathcal{D}_{(2)}$ denote the optimization variable, the data sample, the instance loss and the distribution over $z_{(2)}$ corresponding to the lower bound construction in Lemma 8. The statement of Lemma 8 implies that:

(a) The population loss $F_{(2)}(w_{(2)}) := \mathbb{E}_{z_{(2)} \sim \mathcal{D}_{(2)}}\big[f(w_{(2)}; z_{(2)})\big]$ is 1-Lipschitz and convex in $w_{(2)}$. Additionally, there exists a point $w^*_{(2)} \in \operatorname{argmin}_{w_{(2)}} F(w_{(2)})$ such that $\|w^*_{(2)}\| \leq 1$.

(b) The variance of the gradient is bounded, i.e. for any $w_{(2)}$,

$$\mathbb{E}_{z_{(2)} \sim \mathcal{D}_{(2)}}\big[\|\nabla f_{(2)}(w_{(2)}; z_{(2)}) - \nabla F_{(2)}(w_{(2)})\|^2\big] \leq 1. \quad (39)$$

(c) If $\eta < 1/64n^{5/4}$, then for any $T > 0$, the point $\widehat{w}^{\text{GD}}_{(2)}[\eta, T]$ returned by running GD algorithm with step size η for T steps satisfies

$$\mathbb{E}[F_{(2)}\big(\widehat{w}^{\text{GD}}_{(2)}[\eta, T]\big)] - \min_{w_{(2)}} F_{(2)}(w_{(2)}) = \Omega\Big(\frac{1}{n^{3/8}}\Big). \quad (40)$$

Lower bound construction. Consider the following learning setting:

- *Optimization variable*: w is defined as the concatenation of the variables $(w_{(1)}, w_{(2)})$.
- *Data instance*: z is defined as the concatenation of the data instances $(z_{(1)}, z_{(2)})$.
- *Data distribution*: $\mathcal{D}$ is defined as $\mathcal{D}_{(1)} \times \mathcal{D}_{(2)}$, i.e. $z_{(1)}$ and $z_{(2)}$ are sampled independently from $\mathcal{D}_{(1)}$ and $\mathcal{D}_{(2)}$ respectively.
- *Loss function*: is defined as

$$f(w; z) := f_{(1)}(w_{(1)}; z_{(1)}) + f_{(2)}(w_{(2)}; z_{(2)}),$$

  Additionally, define the population loss $F(w) := \mathbb{E}_{z \sim \mathcal{D}}[f(w; z)] = F_{(1)}(w_{(1)}) + F_{(2)}(w_{(2)})$.

Excess risk guarantee for SGD. We first show that the above learning setting is in SCO. Note that the population loss

$$F(w) = F_{(1)}(w_{(1)}) + F_{(2)}(w_{(2)}).$$

Clearly, $F(w)$ is convex in w since the functions $F_{(1)}(w_{(1)})$ and $F_{(2)}(w_{(2)})$ are convex in $w_{(1)}$ and $w_{(2)}$ respectively. Next, note that for any w and w',

$$\begin{aligned}|F(w) - F(w')| &\overset{(i)}{\leq} |F_{(1)}(w_{(1)}) - F_{(1)}(w'_{(1)})| + |F_{(2)}(w_{(2)}) - F_{(2)}(w'_{(2)})| \\ &\overset{(ii)}{\leq} 3\|w_{(1)} - w'_{(1)}\| + \|w_{(2)} - w'_{(2)}\| \\ &\leq 3\big(\sqrt{\|w_{(1)} - w'_{(1)}\|^2} + \sqrt{\|w_{(2)} - w'_{(2)}\|^2}\big) \\ &\overset{(iii)}{\leq} 3\sqrt{2} \cdot \sqrt{\|w_{(1)} - w'_{(1)}\|^2 + \|w_{(2)} - w'_{(2)}\|^2} \\ &\overset{(iv)}{=} 3\sqrt{2} \cdot \sqrt{\|w - w'\|^2} \\ &= 3\sqrt{2}\|w - w'\|,\end{aligned}$$

where the inequality (i) above follows from Triangle inequality and the inequality (ii) is given by the fact that $F_{(1)}$ is 3-Lipschitz in $w_{(1)}$ and that $F_{(2)}$ is 1-Lipschitz in $w_{(2)}$. The inequality in (iii) follows from an application of Jensen's inequality and using concavity of square-root. Finally, the equality in (iv) is by construction of the variable w as concatenation of the variables $(w_{(1)}, w_{(2)})$. Thus, the population loss $F(w)$ is $3\sqrt{2}$-Lipschitz in w.

We next show a bound on the variance of the gradient. Note that for any w,

$$\begin{aligned}&\mathbb{E}_{z\sim\mathcal{D}}\big[\|\nabla f(w;z) - \nabla F(w)\|^2\big] \\ &\qquad = \mathbb{E}_{z\sim\mathcal{D}}\big[\|\nabla f_{(1)}(w_{(1)}; z_{(1)}) + \nabla f_{(2)}(w_{(2)}; z_{(2)}) - \nabla F_{(1)}(w_{(1)}) - \nabla F_{(2)}(w_{(2)})\|^2\big] \\ &\qquad \overset{(i)}{\leq} 2\,\mathbb{E}_{z_{(1)}\sim\mathcal{D}_{(1)}}\big[\|\nabla f_{(1)}(w_{(1)}; z_{(1)}) - \nabla F_{(1)}(w_{(1)})\|^2\big] \\ &\qquad\qquad + 2\,\mathbb{E}_{z_{(2)}\sim\mathcal{D}_{(2)}}\big[\|\nabla f_{(2)}(w_{(2)}; z_{(2)}) - \nabla F_{(2)}(w_{(2)})\|^2\big] \\ &\qquad \overset{(ii)}{\leq} 72 + 2\,\mathbb{E}_{z_{(2)}\sim\mathcal{D}_{(2)}}\big[\|\nabla f_{(2)}(w_{(2)}; z_{(2)}) - \nabla F_{(2)}(w_{(2)})\|^2\big] \overset{(iii)}{\leq} 74,\end{aligned}$$

where the inequality (i) above follows from the fact that $(a+b)^2 \leq 2a^2 + 2b^2$ for any a, b. The inequality (ii) holds because the function $f_{(1)}(w_{(1)}; z_{(1)})$ is 3-Lipchitz in $w_{(1)}$ for any $z_{(1)}$, and because $F_{(1)}(w_{(1)})$ is 3-lipschitz in $w_{(1)}$. Finally, the inequality (iii) is due to the bound in (39).

We next show that there exists a $w^* \in \operatorname{argmin}_w F(w)$ such that $\|w^*\| = O(1)$. Since the components $(w_{(1)}, w_{(2)})$ do not interact with each other in the above construction of the loss function f, we note that the point $w^* = (w^*_{(1)}, w^*_{(2)})$ is a minimizer of $F(w) = F_{(1)}(w_{(1)}) + F_{(1)}(w_{(1)})$. This point w^* satisfies

$$\|w^*\| \leq \|w^*_{(1)}\| + \|w^*_{(2)}\| = O(1),$$

where we used the fact that $\|w^*_{(1)}\| = O(1)$ and $\|w^*_{(2)}\| = O(1)$.

Combining the above derived properties, we get that:

(a) The population loss $F(w)$ is $3\sqrt{2}$-Lipschitz and convex in w.

(b) There exists a point $w^* \in \operatorname{argmin}_w F(w)$ such that $\|w^*\| = O(1)$.

(c) For any w, the gradient variance $\mathbb{E}_{z\sim\mathcal{D}}\big[\|\nabla f(w;z) - \nabla F(w)\|^2\big] \leq 72$.

Thus, as a consequence of Theorem 8, we get that running SGD with step size $\eta = 1/\sqrt{n}$ and initialialization $w_1 = 0$, returns the point $\widehat{w}_n^{\mathrm{SGD}}$ that satisfies

$$\mathbb{E}\big[F(\widehat{w}_n^{\mathrm{SGD}})\big] - \min_w F(w) = O\Big(\frac{1}{\sqrt{n}}\Big). \tag{41}$$

Lower bound for GD. We next show that GD algorithm fails to match the performance guarantee of SGD in (41), for any step size η and time step T.

Let $\widehat{w}^{\mathrm{GD}}[\eta, T]$ denote the point returned by running GD algorithm on the function f with step size η for T steps. Since the components $w_{(1)}$ and $w_{(2)}$ do not interact with each other in the GD update step due to the construction of the function f, we have that

$$\begin{aligned}\mathbb{E}\big[F(\widehat{w}^{\mathrm{GD}}[\eta, T])\big] - \min_{w} F(w) = \mathbb{E}[F_{(1)}(\widehat{w}^{\mathrm{GD}}_{(1)}[\eta, T])] - \min_{w_{(1)}} F_{(1)}(w_{(1)}) \\ + \mathbb{E}[F_{(2)}(\widehat{w}^{\mathrm{GD}}_{(2)}[\eta, T])] - \min_{w_{(2)}} F_{(2)}(w_{(2)})\end{aligned} \tag{42}$$

The key idea behind the lower bound for GD is that first components has a lower bound of $\Omega(1/n^{5/12})$ when η is larger than $1/64n^{5/4}$. Specifically, in order to improve the excess risk bound over the rate of $1/n^{5/12}$ w.r.t. the variable $w_{(1)}$, we need η to be smaller than $1/64n^{5/4}$. However, any choice of $\eta < 1/64n^{5/4}$ fails to find a good solution w.r.t. the component $w_{(2)}$. We formalize this intuition by considering the two cases (a) $\eta < 1/64n^{5/4}$, and (b) $\eta \geq 1/64n^{5/4}$ separately below.

- **Case 1: $\boldsymbol{\eta < 1/64n^{5/4}}$.** Using the fact that $\mathbb{E}[F_{(1)}(\widehat{w}^{\mathrm{GD}}_{(1)}[\eta, T])] - \min_{w_{(1)}} F_{(1)}(w_{(1)}) \geq 0$ in (42), we get that

$$\begin{aligned}\mathbb{E}\big[F(\widehat{w}^{\mathrm{GD}}[\eta, T])\big] - \min_{w} F(w) &\geq \mathbb{E}\big[F_{(2)}\big(\widehat{w}^{\mathrm{GD}}_{(2)}[\eta, T]\big)\big] - \min_{w_{(2)}} F_{(2)}(w_{(2)}) \\ &= \Omega\Big(\frac{1}{n^{3/8}}\Big),\end{aligned} \tag{43}$$

  where the inequality in the second line above is due to the lower bound in (40) which holds for all $\eta < 1/64n^{5/4}$ and $T \geq 1$.

- **Case 2: $\boldsymbol{\eta \geq 1/64n^{5/4}}$.** Using the fact that $\mathbb{E}[F_{(2)}(\widehat{w}^{\mathrm{GD}}_{(2)}[\eta, T])] - \min_{w_{(2)}} F_{(2)}(w_{(2)}) \geq 0$ in (42), we get that

$$\mathbb{E}\big[F(\widehat{w}^{\mathrm{GD}}_{\eta, T})\big] - \min_{w} F(w) \geq \mathbb{E}\big[F_{(1)}\big(\widehat{w}^{\mathrm{GD}}_{(1)}[\eta, T]\big)\big] - \min_{w_{(1)}} F_{(1)}(w_{(1)}). \tag{44}$$

  The lower bound in (38) suggests that for $\eta \in [1/n^2, 1)$ and $T \geq 1$,

$$\begin{aligned}\mathbb{E}\big[F_{(1)}\big(\widehat{w}^{\mathrm{GD}}_{(1)}[\eta, T]\big)\big] - \min_{w_{(1)}} F_{(1)}(w_{(1)}) &= \Omega\Big(\frac{1}{\log^4(n)} \min\Big\{\eta\sqrt{T} + \frac{1}{\eta T} + \frac{\eta T}{n}, 1\Big\}\Big) \\ &= \Omega\Big(\frac{1}{\log^4(n)} \min\Big\{\eta\sqrt{T} + \frac{1}{2\eta T} + \frac{1}{2\eta T} + \frac{\eta T}{n}, 1\Big\}\Big) \\ &\overset{(i)}{=} \Omega\Big(\frac{1}{\log^4(n)} \min\Big\{\eta\sqrt{T} + \frac{1}{2\eta T} + \frac{1}{\sqrt{2n}}, 1\Big\}\Big) \\ &\overset{(ii)}{=} \Omega\Big(\frac{1}{\log^4(n)} \min\Big\{\eta^{1/3} + \frac{1}{\sqrt{n}}, 1\Big\}\Big)\end{aligned} \tag{45}$$

  where (i) follows from an application of the AM-GM inequality for the last two terms, and (ii) holds by setting $T = 1/\eta^{4/3}$ which minimizes the expression in (45). Plugging the above lower bound in (44), we get that

$$\mathbb{E}\big[F\big(\widehat{w}^{\mathrm{GD}}[\eta, T]\big)\big] - \min_{w} F(w) \geq \Omega\Big(\frac{1}{\log^4(n)} \min\Big\{\eta^{1/3} + \frac{1}{\sqrt{n}}, 1\Big\}\Big).$$

  Finally, using the fact that $\eta \geq 1/64n^{5/4}$ in the above bound, we get

$$\mathbb{E}\big[F\big(\widehat{w}^{\mathrm{GD}}[\eta, T]\big)\big] - \min_{w} F(w) \geq \Omega\Big(\frac{1}{\log^4(n)} \min\Big\{\frac{1}{n^{5/12}} + \frac{1}{\sqrt{n}}, 1\Big\}\Big). \tag{46}$$

Combining the lower bound from (43) and (46) for the two cases above, we get that for all $\eta \geq 0$ and $T \geq 1$, the point $\widehat{w}^{\mathrm{GD}}[\eta, T]$ returned by running GD algorithm on the function f with step size η for T steps satisfies:

$$\mathbb{E}\big[F\big(\widehat{w}^{\mathrm{GD}}[\eta, T]\big)\big] - \min_{w} F(w) \geq \Omega\Big(\frac{1}{\log^4(n)} \min\Big\{\frac{1}{n^{5/12}}, 1\Big\}\Big).$$

□

D Missing proofs from Section 5

The pseudocode for multi-pass SGD algorithm given in Algorithm 1 is slightly different from the description of the algorithm given at the beginning of Section 5. In particular, at the start of every epoch, Algorithm 1 uses the following projection operation:

$$\Pi_{w_1, B}(w) = \begin{cases} w & \text{if } \|w - w_1\| \leq B \\ w_1 + \frac{B}{\|w - w_1\|}(w - w_1) & \text{otherwise.} \end{cases}$$

This ensures that the iterate at the start of every epoch has bounded norm. Rest of the algorithm is the same as in the description in the main body.

Algorithm 1 Multi-pass SGD algorithm

```
Input: Dataset $S = \{z_i\}_{i=1}^n$, number of passes $k$, initial point $w_1$.
Define $m := n/2$, $S_1 := \{z_i\}_{i=1}^m$ and $S_2 := S \setminus S_1$
Initialize $\widehat{\mathcal{W}} \leftarrow \emptyset$
for $j = 1, \ldots, k$ do                                          ▷ Multiple passes
    $w_{m(j-1)+1} \leftarrow \Pi_{w_1, B}(w_{m(j-1)+1})$
    $\eta_j \leftarrow \frac{1}{\sqrt{nj}}$
    for $i = 1, 3, \ldots, m$ do
        $w_{m(j-1)+i+1} \leftarrow w_{m(j-1)+i} - \eta_j \nabla f(w_{m(j-1)+i}; z_i)$.
    $\widehat{w}_j \leftarrow \frac{1}{mj} \sum_{t=1}^{mj} w_j$
    $\widehat{\mathcal{W}} \leftarrow \widehat{\mathcal{W}} \cup \{\widehat{w}_j\}$
Return the point $\widehat{w}^{\mathrm{MP}} \in \mathrm{argmin}_{w \in \widehat{\mathcal{W}}} F_{S_2}(W) := \frac{1}{m} \sum_{i=1}^m f(w; z_{m+1})$     ▷ Validation
```

D.1 Proof of Proposition 1

Proof. The following proof uses the fact that the loss function is bounded over the domain of interest. This is automatically ensured for our multi-pass SGD algorithm due to the projection step in Algorithm 1. Assume that f is bounded by M. Since S_2 is independent of S_1, we note S_2 is also independent of the set of points $\widehat{\mathcal{W}}$. Hoeffding's inequality (Lemma 1) thus implies that, with probability at least $1 - \delta$, for all $w \in \widehat{\mathcal{W}}$,

$$|F_{S_2}(w) - F(w)| \leq M\sqrt{\frac{\log(2k/\delta)}{n}}.$$

Thus, the returned point $\widehat{w}^{\mathrm{MP}} \in \mathrm{argmin}_{w \in \widehat{\mathcal{W}}} F_{S_2}(w)$ satisfies:

$$\begin{aligned} F(\widehat{w}^{\mathrm{MP}}) &\leq \min_{w \in \widehat{\mathcal{W}}} F(w) + 2M\sqrt{\frac{\log(2k/\delta)}{n}} \\ &\leq F(\widehat{w}_1) + 2M\sqrt{\frac{\log(2k/\delta)}{n}}. \end{aligned}$$

Observing that the point $\widehat{w}_1$ denotes $\widehat{w}_{n/2}^{\mathrm{SGD}}$, and converting the above high probability statement into an in-expectation result (since f is bounded by M) gives us the desired statement. □

D.2 Proof of Theorem 4

For the upper bound of part (b) in Theorem 4, we need the following slight generalization of Theorem 1.

Lemma 10. *Consider any SCO problem and initial point w_1 satisfying Assumption I in (2) and Assumption II in (3). Suppose starting from the point w_1, we run SGD algorithm with step size η for n steps. Then the average iterate $\widehat{w}_n^{\mathrm{SGD}} := \frac{1}{n}\sum_{i=1}^n w_i$ enjoys the bound*

$$\mathbb{E}_S[F(\widehat{w}_n^{\mathrm{SGD}})] - \inf_w F(w) \leq \eta(\sigma^2 + L^2) + \frac{1}{2\eta n}\|w_1 - w^*\|^2,$$

for any point $w^* \in \operatorname{argmin}_w F(w)$.

Proof. The proof follows exactly along the lines of the proof of Theorem 8 given on page 14. □

Theorem 12. *Consider any SCO problem satisfying Assumption I in (2) and Assumption II in (3). Suppose we run the following variant of SGD: for some integer $k \geq 1$, we run Algorithm 1 for nk steps with the only change being that fresh samples z_t are used in each update step (line 5) instead of reusing samples. Then the average iterate $\widehat{w}_{nk}^{\mathrm{SGD}}$ enjoys the bound*

$$\mathbb{E}_S[F(\widehat{w}_{nk}^{\mathrm{SGD}})] - \inf_{w\colon \|w - w_1\| \leq B} F(w) \leq 2(B^2 + L^2 + \sigma^2)\sqrt{nk}.$$

Proof. The proof is completely standard, we just include it here for completeness. Let $w^* \in \mathbb{R}^d$ be any point such that $\|w_1 - w^*\| \leq B$. We use $\|w_t - w^*\|^2$ as a potential function. Within epoch j, we have

$$\|w_{t+1} - w^*\|^2 = \|w_t - w^*\|^2 - 2\eta_j\langle \nabla f(w_t; z_t), (w_t - w^*)\rangle + \eta_j^2\|\nabla f(w_t; z_t)\|^2.$$

Since $\|\nabla f(w_t; z_t)\|^2 \leq 2L^2 + 2\sigma^2$, by rearranging the above, taking expectation over z_t conditioned on w_t, and using convexity of F we have

$$F(w_t) - F(w^*) \leq \frac{\|w_t - w^*\|^2 - \mathbb{E}[\|w_{t+1} - w^*\|^2 | w_t]}{2\eta_j} + \eta_j(L^2 + \sigma^2).$$

Now taking expectation over the randomness in w_t, and summing up the inequality for all iterations in epoch j, we get

$$\sum_{t=n(j-1)+1}^{nj} F(w_t) - F(w^*) \leq \frac{\mathbb{E}[\|w_{n(j-1)+1} - w^*\|^2 - \mathbb{E}[\|w'_{nj+1} - w^*\|^2]}{2\eta_j} + \eta_j(L^2 + \sigma^2)n,$$

where we use the notation w'_{nj+1} to denote the iterate computed by SGD before the $\Pi_{w_1,B}$ projection to generate w_{nj+1}. Summing over all the k epochs, we have

$$\begin{aligned}
\sum_{t=1}^{nk} F(w_t) - F(w^*) &\leq \frac{\|w_1 - w^*\|^2}{2\eta_1} + \sum_{j=1}^{k-1}\left(\frac{\mathbb{E}[\|w_{nj+1} - w^*\|^2]}{2\eta_{j+1}} - \frac{\mathbb{E}[\|w'_{nj+1} - w^*\|^2]}{2\eta_j}\right) \\
&\quad - \frac{\mathbb{E}[\|w_{nk+1} - w^*\|^2]}{2\eta_k} + \sum_{j=1}^{k}\eta_j(L^2 + \sigma^2)n \\
&\overset{(i)}{\leq} \frac{\|w_1 - w^*\|^2}{2\eta_1} + \sum_{j=1}^{k-1}\left(\frac{1}{2\eta_{j+1}} - \frac{1}{2\eta_j}\right)\mathbb{E}[\|w_{nj+1} - w^*\|^2] + \sum_{j=1}^{k}\eta_j(L^2 + \sigma^2)n \\
&\overset{(ii)}{\leq} \frac{2B^2}{\eta_k} + \sum_{j=1}^{k}\eta_j(L^2 + \sigma^2)n \\
&\overset{(iii)}{\leq} 2(B^2 + L^2 + \sigma^2)\sqrt{nk}.
\end{aligned}$$

Here, (i) follows since $\|w_{nj+1} - w^*\|^2 \leq \|w'_{nj+1} - w^*\|^2$ since the $\Pi_{w_1,B}$ projection can only reduce the distance to w^*, and (ii) follows since $\|w_{nj+1} - w^*\|^2 \leq 4B^2$ for all $j = 0, 1, \dots, k-1$, and telescoping, and finally (iii) follows by plugging in $\eta_j = 1/\sqrt{nj}$.

Finally, the stated bound on $F(\widehat{w}_{nk}^{\mathrm{SGD}})$ follows via an application of Jensen's inequality to the convex function F. The dependence on problem specific constants (σ, L and B) in the above bound can be improved further with a different choice of the step size η; getting the optimal dependence on these constants, however, is not the focus of this work. □

We now prove Theorem 4. To begin, we first define the instance space, the loss function, and the data distribution.

Instance space and loss function. We define the instance space $\mathcal{Z} = \{0,1\}^{kd} \times \{\pm 1\}^k \times \{0, e_1, \dots, e_d\}^k$. That is, each instance $z \in \mathcal{Z}$ can be written as a 3-tuple $z = (x, y, \alpha)$ where $x \in \{0,1\}^{kd}$, $y \in \{\pm 1\}^k$, and $\alpha \in \{0, e_1, \dots, e_d\}^k$. For each $s \in [k]$, we define x_s, y_s, α_s to be the s-th parts of x, y, α respectively when these vectors are split into k contiguous equal sized blocks of sizes $d, 1, 1$ respectively. Define the function $f_{(C)} : \mathbb{R}^{d+n+2} \times \mathcal{Z} \to \mathbb{R}$ on the variables $u \in \mathbb{R}, v \in \mathbb{R}^{n+1}, \tau \in \mathbb{R}^d$ and instance $z = (x, y, \alpha) \in \mathcal{Z}$ as follows. First, define the intervals $I_1 = (-\infty, \frac{1}{k}]$, $I_s = (\frac{(s-1)}{k}, \frac{s}{k}]$ for $s = 2, 3, \dots, k-1$, and $I_k = (\frac{(k-1)}{k}, \infty)$. Then define

$$f_{(C)}((u, v, \tau); z) := f_N(v) + \sum_{s=1}^{k} \mathbb{1}[u \in I_s] f_{(A)}(\tau; (x_s, y_s, \alpha_s)) - \tfrac{2}{\sqrt{kn}} \min\{u, 1\} + c_1, \qquad \text{(C)}$$

where the function $f_N : \mathbb{R}^{n+1} \to \mathbb{R}$ is defined as

$$f_N(v) := \left(\tfrac{\sqrt{n+1}}{1+\sqrt{n+1}}\right) \max_{i \in [n+1]} v_i + \left(\tfrac{1}{2+2\sqrt{n+1}}\right) \|v\|^2,$$

and the constant $c_1 = \frac{2}{\sqrt{kn}} + \frac{1}{2+2\sqrt{n+1}}$. Finally, we define the variable w to denote the tuple (u, v, τ). We also assume that $n \geq 300$ and $d \geq \log(10)2^n + 1$.

For the purpose of defining the SGD update on the function $f_{(C)}(\cdot; z)$ for any given $z \in \mathcal{Z}$, we make the following convention for defining subgradients: when $u \in I_s$ for some $s \in [k]$, we use a subgradient of

$$f_N(v) + f_{(A)}(\tau; (x_s, y_s, \alpha_s)) - \tfrac{2}{\sqrt{kn}} \min\{u, 1\}.$$

It is easy to check that with the above convention, the norm of the subgradient is always $O(1)$.

Input distribution. The samples $z \in \mathcal{Z}$ are drawn by sampling $\langle (x_s, y_s, \alpha_s) \rangle_{s=1}^{k} \sim \mathcal{D}(\frac{1}{10}, \frac{1}{2}, e_{j^*})^{\otimes k}$, for some $j^* \in [d]$ that will be defined in the proof (see Definition 2 for the definition of $\mathcal{D}$).

Proof of Theorem 4. Let $\mathcal{D}'$ denote the distribution specified over the instance space $\mathcal{Z}$. The exact choice of j^* will be specified later in this proof. Note that due to the indicators in (C), the population loss under $f_{(C)}$ when the data are sampled from $\mathcal{D}'$ can be written as

$$F(w) = \mathbb{E}_{z \sim \mathcal{D}'}[f_{(C)}((u, v, \tau); z')] = f_N(v) + \mathbb{E}_{z \sim \mathcal{D}'}[f_{(A)}(\tau; z)] - \tfrac{2}{\sqrt{kn}} \min\{u, 1\}. \qquad (47)$$

Note that by Lemma 5, $\mathbb{E}_{z \sim \mathcal{D}'}[f_{(A)}(\tau; z)]$ is a convex function of τ. Hence, F is convex, 1-Lispchitz and $\tau = e_{j^*}$ denotes its unique minimizer. Also note that F nicely separates out as convex functions of u, v, τ and hence it is minimized by minimizing the component functions separately. In particular, it is easy to check by computing the subgradient that the optima are

$$u = 1, \qquad v = -\tfrac{1}{\sqrt{n+1}} \mathbf{1}, \qquad w = e_{j^*},$$

where $\mathbf{1}$ is the all 1's vector. The corresponding optimal values are $-\frac{2}{\sqrt{kn}}$, $-\frac{1}{2+2\sqrt{n+1}}$, and 0. This implies that $F^* = \min_w F(w) = 0$. More importantly, the suboptimality gap also decomposes as the sum of suboptimality gaps for the three functions, a fact we will use repeatedly in our analysis.

Finally, we will assume that the initial value of the variables u, v, τ is 0.

Proof of part-(a). Since $\|\nabla f(u, v, \tau; z)\| \leq O(1)$ for any $z \in \mathcal{Z}$ and u, v, τ, and the optimal values of u, v, τ are $O(1)$ in magnitude, and $F(\omega)$ is convex in $\omega = (u, v, \tau)$, Theorem 1 implies that single pass SGD run for n steps with a step size of $\frac{1}{\sqrt{n}}$ will obtain $O(\frac{1}{\sqrt{n}})$ suboptimality gap in expectation.

As for the lower bound, note that the function f_N is exactly the same one constructed in [Nesterov, 2014, Section 3.2.1]. There, it is shown (in Theorem 3.2.1) that a certain class of gradient based methods (including GD/SGD) have suboptimality at least $\Omega(1/\sqrt{n})$ after n iterations, which completes the proof of part-(a) of Theorem 4.

Proof of part-(b). In the following, we give convergence guarantee for multi-pass SGD algorithm that does k passes over the dataset S_1 of $n/2$ samples (see Algorithm 1). First, note that the deterministic components of the function $f_{(C)}$, viz. f_N and $-\frac{2}{\sqrt{kn}} \min\{u, 1\}$, are unaffected by the randomized component in the iterates produced by SGD update. Since these deterministic components are 1-Lipschitz, their optimal values are $O(1)$ in magnitude and the corresponding population losses are convex, we conclude via Theorem 12 that the multi-pass SGD, which is equivalent to GD, attains a suboptimality gap of $O(\frac{1}{\sqrt{kn}})$ on these components. Specifically, the points $\widehat{u}_k$ and $\widehat{v}_k$ returned after the k-th pass satisfy

$$f_N(\widehat{v}_k) - \tfrac{2}{\sqrt{kn}} \min\{\widehat{u}_k, 1\} - \min_{v,u} \Big(f_N(v) - \tfrac{2}{\sqrt{kn}} \min\{u, 1\} \Big) \leq O(\frac{1}{\sqrt{nk}}).$$

Coming to the randomized component of $f_{(C)}$, we note that as long as $u < 1$, the gradient of $f_{(C)}$ with respect to u is always $-2/\sqrt{kn}$. Thus, u keeps monotonically increasing at each step of SGD with an increment equal to the step size times $2/\sqrt{kn}$. Suppose we run k pass SGD with step size set to $1/\sqrt{kn}$ and u starting at 0, where in each pass we take $n/2$ steps of SGD using the dataset S_1. It is easy to see that for all $s \in [k]$, the value of u stays in I_s in the s-th pass, and traverses to the next interval I_{s+1} as soon as the $(s+1)$-th pass starts. Thus, within each pass $s \in [k]$ and for all $i \in [n]$, multi-pass SGD encounters a fresh i.i.d. sample $(x_{i,s}, y_{i,s}, \alpha_{i,s}) \sim \mathcal{D}$ for every update. This is thus equivalent to running SGD with $kn/2$ such i.i.d. samples over the first $kn/2$ iterations. An application of Theorem 12 (where we set n to be $n/2$) implies that the suboptimality gap of the iterate $\widehat{\tau}_k$ generated after the kth-pass of SGD algorithm on the randomized component of $f_{(C)}$ is

$$\mathbb{E}_{z\sim\mathcal{D}}[f_{(A)}(\widehat{\tau}_k; z)] - \min_w \mathbb{E}_{z\sim\mathcal{D}}[f_{(A)}(w; z)] \leq O\Big(\frac{1}{\sqrt{nk}}\Big).$$

Taking the above two bounds together, we get that the point $\widehat{w}_k = (\widehat{u}_k, \widehat{v}_k, \widehat{\tau}_k)$ satisfies

$$F(\widehat{w}_k) = F(\widehat{w}_k) - \min_w F(w) \leq O\Big(\frac{1}{\sqrt{nk}}\Big), \tag{48}$$

where the equality in the first line above follows from using the fact that $\min_w F(w) = 0$ by construction.

Finally, note that the returned point $\widehat{w}^{\text{MP}} \in \operatorname{argmin}_{w \in \widehat{\mathcal{W}}} F_{S_2}(W)$ where $\mathcal{W} = \{\widehat{w}_1, \dots, \widehat{w}_k\}$. Lemma 4 thus implies that the point $\widehat{w}^{\text{MP}}$ satisfies

$$\begin{aligned} F(\widehat{w}^{\text{MP}}) &\leq \min_{j\in[k]} F(\widehat{w}_j) + O\Big(\frac{L\log(k/\delta)}{n} + \sqrt{\frac{\min_{j\in[k]} F(\widehat{w}_j)\, L\log(k/\delta)}{n}}\Big) \\ &\leq F(\widehat{w}_k) + O\Big(\frac{L\log(k/\delta)}{n} + \sqrt{\frac{F(\widehat{w}_k)\, L\log(k/\delta)}{n}}\Big) \\ &\leq \frac{1}{\sqrt{nk}} + O\Big(\frac{\log(k/\delta)}{n} + \frac{1}{n}\sqrt{\frac{\log(k/\delta)}{k}}\Big), \end{aligned}$$

where L-denotes the Lipschitz constant for the function f and the inequality in the last line follows by plugging in the bound on $F(\widehat{w}_k)$ from (48) and using the fact that $L = O(1)$. Finally, observing that $\min_w F(w) = 0$, the above bound implies that

$$F(\widehat{w}^{\text{MP}}) - \min_w F(w) \leq O\Big(\frac{1}{\sqrt{nk}}\Big)$$

for $k = o(n)$; proving the desired claim.

Proof of part-(c). The proof follows exactly along the lines of the proof of Theorem 2 in Appendix B.2. Recall that $n \geq 300$ and $d \geq \log(10)2^n + 1$.

Assume, for the sake of contradiction, that there exists a regularizer $R : \mathbb{R} \times \mathbb{R}^{n+1} \times \mathbb{R}^d \to \mathbb{R}$ such that for any distribution $\mathcal{D} = \mathcal{D}(\frac{1}{10}, \frac{1}{2}, e_{j^*})$ (see Definition 2) for generating the components of z, the expected suboptimality gap for RERM is at most $\varepsilon = 1/20000$. Then, by Markov's inequality, with probability at least 0.9 over the choice of sample set S, the suboptimality gap is at most 10ε.

First, since the population loss separates out nicely in terms of losses for u, v, τ in (47), we conclude that if (u', v', τ') is a 10ε-suboptimal minimizer for the population loss when components of z are drawn from $\mathcal{D}$, then u', v', and τ' must be individually 10ε-suboptimal minimizers for the functions $-\frac{1}{\sqrt{kn}} \min\{u, 1\}$, $f_N(v)$ and $\mathbb{E}_{z\sim\mathcal{D}}[f_{(A)}(\tau; z)]$ respectively. Additionally, from Lemma 5, we must have that $\|\tau' - e_{j^*}\| \leq 100\varepsilon$. With this insight, for any $j \in [d]$, define the set

$$G_j := \Big\{(u, v, \tau) : \ -\tfrac{1}{\sqrt{kn}} \min\{u, 1\} \leq -\tfrac{1}{\sqrt{kn}} + 10\varepsilon,\ f_N(v) \leq -\tfrac{1}{2+2\sqrt{n+1}} + 10\varepsilon,\ \|\tau - e_j\| \leq 100\varepsilon\Big\}.$$

This set covers all possible 10ε-suboptimal minimizers of the population loss. Also, for convenience, we use the notation

$$w = (u, v, \tau).$$

As in the proof of Theorem 2, define the points w^*_j for $j \in \{0, 1, \dots, d\}$ to be

$$w^*_j \in \operatorname*{argmin}_{w \in G_j} R(w)$$

Now we are ready to define j^*: let $j^* \in [d]$ be any element of $\operatorname{argmax}_{j\in[d]} R(w^*_j)$. The proof now follows similarly to the proof of Theorem 2. In particular, in the following, we condition on the occurrence of the event E defined in the proof of Theorem 2.

Now define $c := -\frac{1}{\sqrt{kn}} - \frac{1}{2+2\sqrt{n+1}}$. It is easy to see that for any $w = (u, v, \tau) \in G_j$ for any $j \in [d]$, we have

$$c \leq -\tfrac{1}{\sqrt{kn}} \min\{u, 1\} + f_N(v) \leq c + 20\varepsilon. \tag{49}$$

Next, consider the point $w^*_{\widehat{j}}$ defined in (12) for the special coordinate $\widehat{j}$. Reasoning similarly to the proof of Theorem 2, we conclude that that $w^*_{\widehat{j}}$ cannot be an ε-suboptimal minimizer of F, and thus $w^*_{\widehat{j}}$ can not be a minimizer of the regularized empirical risk (as all RERM solutions are 10ε-suboptimal w.r.t. the population loss F). Thus, we must have

$$\begin{aligned} \widehat{F}(w^*_{\widehat{j}}) + R(w^*_{\widehat{j}}) &> \min_{w\in G_{j^*}} \Big(\widehat{F}(w) + R(w)\Big) \\ &\geq \min_{w\in G_{j^*}} \widehat{F}(w) + \min_{w\in G_{j^*}} R(w) \\ &\geq R(w^*_{j^*}) + c - 100\varepsilon, \end{aligned} \tag{50}$$

where $\widehat{F}$ denotes the empirical loss on S,, and the last inequality above follows from (49) and due to the fact that: if $w^* = (u^*, v^*, \tau^*)$ is the minimizer of $\widehat{F}(w)$ over G_{j^*}, and $s \in [k]$ is the index such that $u^* \in I_s$, then the function $\tau \mapsto \frac{1}{n}\sum_{i=1}^n y_{i,s}\|\tau \odot x_{i,s}\|$ is 1-Lipschitz and takes the value 0 at $\tau = e_{j^*}$.

On the other hand, if $w^*_{\widehat{j}} = (\widehat{u}, \widehat{v}, \widehat{\tau})$ and $s \in [k]$ is the index such that $\widehat{u} \in I_s$, then using (49), we have

$$\begin{aligned} \widehat{F}(w^*_{\widehat{j}}) &\leq \tfrac{1}{n}\textstyle\sum_{i=1}^n y_{i,s}\|(\widehat{\tau} - e_{j^*}) \odot x_{i,s}\| - \tfrac{1}{\sqrt{kn}}\min\{\widehat{u}, 1\} + f_N(\widehat{v}) \\ &\leq \tfrac{1}{n}\textstyle\sum_{i=1}^n y_{i,s}\|(\widehat{\tau} - e_{j^*}) \odot x_{i,s}\| + c + 20\varepsilon. \end{aligned} \tag{51}$$

Now, since we are conditioning on the occurrence of the event E, using the same chain of inequalities leading to (15), we conclude that

$$\tfrac{1}{n}\textstyle\sum_{i=1}^n y_{i,s}\|(\widehat{\tau} - e_{j^*}) \odot x_{i,s}\| \leq 100\varepsilon - \tfrac{3}{200}. \tag{52}$$

Combining (50), (51), and (52) and rearranging the terms, we get

$$220\varepsilon \geq R(w^*_{j^*}) - R(w^*_{\widehat{j}}) + \frac{3}{200} \geq \frac{3}{200}. \tag{53}$$

where the second inequality above holds because $j^* \in \operatorname{argmax}_{j\in[d]} R(w^*_j)$ (by definition). Thus, $\varepsilon \geq 3/44000 > 1/20000$, a contradiction, as desired. □

E Missing proofs from Section 6

Throughout this section, we assume that a data sample z consists of (x, α), where $x \in \{0,1\}^d$ and $\alpha \in \{0, e_1, \ldots, e_d\}$. The loss function $f_{(B)} : \mathbb{R}^d \times \mathcal{Z}$ is given by:

$$f_{(B)}(w; z) = \frac{1}{2}\|(w-\alpha) \odot x\|^2 - \frac{c_n}{2}\|w-\alpha\|^2 + \max\{1, \|w\|^4\}, \tag{54}$$

where $c_n := n^{-(\frac{1}{4}-\gamma)}$ for some $\gamma > 0$. We will also assume that $c_n \leq \frac{1}{4}$. Furthermore, since $f_{(B)}(w; z)$ is not differentiable when $\|w\| = 1$, we make the following convention to define the sub-gradient:

$$\partial f(w; z) = (w-\alpha) \odot x - c_n(w-\alpha) + 4\mathbf{1}\{\|w\| > 1\}\|w\|^2 w. \tag{55}$$

Whenever clear from the context in the rest of the section, we will ignore the subscript (B), and denote the loss function by $f(w; z)$. Additionally we define the following distribution over the samples $z = (x, \alpha)$.

Definition 3. *For parameters $\delta \in [0, \frac{1}{2}]$, $c_n \in [0,1]$, and $v \in \{0, e_1, \ldots, e_d\}$, define the distribution $\bar{\mathcal{D}}(\delta, c_n, v)$ over $z = (x, \alpha)$ as follows:*

$$x \sim \mathcal{B}(c_n + \delta)^{\otimes d} \qquad \text{and} \qquad \alpha = v.$$

The components x and α are sampled independently.

E.1 Supporting technical results

Before delving into the proofs, we first establish some technical properties of the empirical loss $f(w; z)$ and the population loss $F(w)$. The following lemma states that for any distribution $\mathcal{D}$, the minimizers of the population loss $F(w) := \mathbb{E}_{z\sim\mathcal{D}}[f(w; z)]$ are bounded in norm.

Lemma 11. *Suppose $c_n \leq \frac{1}{4}$. Then for any data distribution $\mathcal{D}$, the minimizer of $F(w) := \mathbb{E}_{z\sim\mathcal{D}}[f(w; z)]$ has norm at most* 1.

Proof. Note that for any w such that $\|w\| > 1$, we have

$$f(w; z) \geq -c_n(\|w\|^2 + \|\alpha\|^2) + \|w\|^4 \geq -c_n\|w\|^2 - c_n + \|w\|^4 > 1 - 2c_n.$$

Thus, $F(w) > 1 - 2c_n$. On the other hand, $f(0; z) = \frac{1}{2}\|\alpha \odot x\|^2 - \frac{c_n}{2}\|\alpha\|^2 \leq \frac{1-c_n}{2}$. Hence, $F(0) \leq \frac{1-c_n}{2} \leq 1 - 2c_n$ since $c_n \leq \frac{1}{4}$. This implies that such a w is not a minimizer of $F(w)$. □

We show now that the single stochastic gradient descent update keeps the iterates bounded as long as the learning rate η is chosen small enough.

Lemma 12. *Suppose that the learning rate $\eta \leq \frac{1}{100}$ and the point w satisfies $\|w\| \leq 2.5$. Let $w^+ = w - \eta \cdot \partial f(w; z)$ for an arbitrary data point z in the support of the data distribution. Then $\|w^+\| \leq 2.5$.*

Proof. We prove the lemma via a case analysis:

1. **Case 1: $\|w\| < 2$.** Using (55) and the Triangle inequality, we get that

$$\begin{aligned}\|\partial f(w; z)\| &\leq \|(w-\alpha) \odot x\| + \|c_n(w-\alpha)\| + 4\|w\|^3 \\ &\leq (1+c_n)(\|w\| + \|\alpha\|) + 4\|w\|^3 \leq 36,\end{aligned} \tag{56}$$

   where the last inequality follows from the fact that the iterate w satisfies $\|w\| \leq 2$, and that the parameter $c_n \leq \frac{1}{4}$ and $\|\alpha\| \leq 1$.

   Now, for the gradient descent update rule $w_{t+1} = w_t - \eta\partial f(w; z)$, an application of the triangle inequality implies that

$$\|w_{t+1}\| = \|w - \eta\partial f(w; z)\| \leq \|w\| + \eta\|\partial f(w; z)\|. \tag{57}$$

   Plugging in the bounds on $\|\partial f(w; z)\|$ derived above, we get that

$$\|w^+\| \leq 2 + 36\eta \leq 2.5 \tag{58}$$

   since $\eta \leq \frac{1}{100}$.

2. **Case 2: $2 \leq \|w\| \leq 2.5$.** We start by observing that the new iterate w^+ satisfies

$$\begin{aligned}\|w^+\|^2 &= \|w_t - \eta \partial f(w; z)\|^2 \\ &= \|w\|^2 + \eta^2 \|\partial f(w; z)\|^2 - 2\eta \langle w, \partial f(w; z) \rangle. \end{aligned} \tag{59}$$

   Reasoning as in case 1 above, using the fact that $\|w\| \leq 2.5$, we have $\|\partial f(w; z)\| \leq 70$. Furthermore,

$$\begin{aligned}\langle w, \partial f(w; z) \rangle &= \langle w, (w - \alpha) \odot x - c_n (w - \alpha) + 4\|w\|^2 w \rangle \\ &= 4\|w\|^4 + \langle w, w \odot x \rangle - \langle w, \alpha \odot x + c_n (w - \alpha) \rangle \\ &\overset{(i)}{\geq} 4\|w\|^4 - \langle w, \alpha \odot x + c_n (w - \alpha) \rangle \\ &\overset{(ii)}{\geq} 4\|w\|^4 - \|w\| ((1 + c_n)\|\alpha\| + c_n \|w\|) \end{aligned}$$

   where the inequality in (i) follows from the fact that $\langle w, w \odot x \rangle \geq 0$, the inequality in (ii) is given by an application of Cauchy-Schwarz inequality followed by Triangle inequality. Next, using the fact that $c_n \leq \frac{1}{4}$ and $\|\alpha\| \leq 1$, we get

$$\langle w, \partial f(w; z) \rangle \geq 4\|w\|^4 - \|w\| \Big(\frac{5}{4} + \frac{1}{4}\|w\| \Big) \geq 60, \tag{60}$$

   where the last inequality holds as the polynomial $f(a) := 4a^4 - a(1.25 + 0.25a)$ is an increasing function of a over the domain $a \in [2, \infty)$.

   Plugging the bound $\|\partial f(w; z)\| \leq 70$ and and (60) in (59), we have

$$\|w^+\|^2 \leq \|w\|^2 + 4900\eta^2 - 60\eta \leq \|w\|^2,$$

   since $\eta \leq \frac{1}{100}$. Thus, $\|w^+\| \leq \|w\| \leq 2.5$.

Thus, in either case, $\|w^+\| \leq 2.5$, completing the proof. □

Corollary 2. *Suppose that the learning rate $\eta \leq 1/100$ and the initial point w_1 be such that $\|w_1\| \leq 2.5$. Then the iterates obtained when running either full gradient descent on the empirical risk, or by running SGD, have norm bounded by 2.5 at all times $t \geq 0$.*

Proof. The iterates obtained during the running of any of the gradient descent variants described in the statement of the lemma can be seen as convex combinations of single sample gradient descent updates. Thus, the bound on the norm follows immediately from Lemma 12 via the convexity of the norm. □

The following lemma shows that the loss function $f(w; z)$ is Lipschitz in the domain of interest, i.e. the ball of radius 2.5. This is the region where all iterates produced by gradient based algorithms and the global minimizers of the population loss (for any distribution $\mathcal{D}$) are located in.

Lemma 13. *In the ball of radius 2.5 around 0, and for any data point z, the function $w \mapsto f(w; z)$ is 70-Lipschitz.*

Proof. For w such that $\|w\| \leq 2.5$, using (55) and the Triangle inequality, we get that

$$\begin{aligned}\|\partial f(w; z)\| &\leq \|(w - \alpha) \odot x\| + \|c_n (w - \alpha)\| + 4\|w\|^3 \\ &\leq (1 + c_n)(\|w\| + \|\alpha\|) + 4\|w\|^3 \leq 70. \end{aligned}$$

□

E.2 Proof of Proposition 2

Proof. The proof is rather simple. To show this statement, we consider two distributions $\mathcal{D}_1$ and $\mathcal{D}_2$ on the instance space. What we will show is that if a learning algorithm succeeds with a rate any better than c_n on one distribution, then it has to have a worse rate on the other distribution. Thus, we will conclude that any learning algorithm cannot obtain a rate better than c_n.

Without further delay, let us define the two distributions we will use in this proof. The first distribution $\mathcal{D}_1$ is given by:

$$x \sim \mathcal{B}\left(\frac{1}{2}\right)^{\otimes d} \quad \text{and} \quad \alpha = 0$$

and the second distribution we consider is $\mathcal{D}_2$ given as follows, first, we draw $\tilde{j} \sim \mathrm{Unif}[d]$, next we set $x[\tilde{j}] = 0$ deterministically, finally, on all other coordinates $i \neq \tilde{j}$, $x[\tilde{j}] \sim \mathrm{unif}\{0,1\}$. We also set $\alpha = 0$ deterministically.

Now the key observation is the following. Since $d > 2^n$, when we draw n samples from distribution $\mathcal{D}_1$ with constant probability there is a coordinate $\hat{j}$ such that $x_t[\hat{j}] = 0$ for all $t \in [n]$. Further, $\hat{j}$ can be any one of the d coordinates with equal probability. However, on the other hand, if we draw n samples from distribution $\mathcal{D}_2$, then by definition of $\tilde{j}$, we have that $x_t[\tilde{j}] = 0$ for any $t \in [n]$. The main observation is that the learning algorithm is agnostic to the distribution on instances. Now since a draw of n samples from $\mathcal{D}_1$ has a coordinate $\hat{j}$ that to the algorithm is indistinguishable from coordinate $\tilde{j}$ when n samples are drawn from $\mathcal{D}_2$, the algorithm cannot identify which of the two distributions the sample if from.

Hence, with constant probability both samples from $\mathcal{D}_1$ and $\mathcal{D}_2$ will be indistinguishable. However note that for distribution $\mathcal{D}_1$ we have

$$F_1(w) = \mathbb{E}_{z\sim\mathcal{D}_1} f(w,z) = \frac{1}{2}\left(\frac{1}{2} - c_n\right)\|w\|^2 + \max\{1, \|w\|^4\},$$

and for distribution $\mathcal{D}_2$,

$$F_2(w) = \mathbb{E}_{z\sim\mathcal{D}_2} f(w,z) = \frac{1}{2}\left(\frac{1}{2} - c_n\right)\|w_{[d]\setminus\{\tilde{j}\}}\|^2 - \frac{c_n}{2}|w[\tilde{j}]|^2 + \max\{1, \|w\|^4\}.$$

Hence notice that for any w,

$$\begin{aligned} F_1(w) - \inf_w F_1(w) &= \frac{1}{2}\left(\frac{1}{2} - c_n\right)\|w\|^2 + \max\{1, \|w\|^4\} - 1 \\ &\geq \frac{1}{2}\left(\frac{1}{2} - c_n\right)\|w\|^2 \geq \frac{1}{2}\left(\frac{1}{2} - c_n\right)|w[\tilde{j}]|^2 \end{aligned}$$

and

$$\begin{aligned} F_2(w) - \inf_w F_2(w) &= \frac{1}{2}\left(\frac{1}{2} - c_n\right)\|w_{[d]\setminus\{\tilde{j}\}}\|^2 - \frac{c_n}{2}|w[\tilde{j}]|^2 + \max\{1, \|w\|^4\} + \frac{c_n}{2} - 1 \\ &\geq \frac{1}{2}\left(\frac{1}{2} - c_n\right)\|w_{[d]\setminus\{\tilde{j}\}}\|^2 - \frac{c_n}{2}|w[\tilde{j}]|^2 + \frac{c_n}{2} \\ &\geq -\frac{c_n}{2}|w[\tilde{j}]|^2 + \frac{c_n}{2} \end{aligned}$$

Now as mentioned before, they key observation is that with constant probability we get a sample using which we cant distinguish between whether we got samples from $\mathcal{D}_1$ or from $\mathcal{D}_2$. Hence in this case, if we want to obtain a good suboptimality, we need to pick a common solution w for which both $F_1(w) - \inf_w F_1(w)$, and $F_2(w) - \inf_w F_2(w)$ are small. However, note that if we want a w for which $F_2(w) - \inf_w F_2(w) \leq c_n/4$, then it must be the case that

$$\frac{c_n}{4} \geq -\frac{c_n}{2}|w[\tilde{j}]|^2 + \frac{c_n}{2}$$

and hence, it must be the case that, $|w[\tilde{j}]| \geq \frac{1}{\sqrt{2}}$. However, for such a w from the above we clearly have that

$$F_1(w) - \inf_w F_1(w) \geq \frac{1}{2}\left(\frac{1}{2} - c_n\right)|w[\tilde{j}]|^2 \geq \frac{1}{4}\left(\frac{1}{2} - c_n\right) \geq \frac{1}{9}$$

(as long as $c_n = o(1)$). Thus we can conclude that no learning algorithm can attain a rate better than $c_n/4$. □

E.3 Proof of Theorem 5

We prove part-(a) and part-(b) separately below. The following proof of Theorem 5-(a) is similar to the proof of Theorem 2 given in Appendix B.2.

Proof of Theorem 5-(a). In the following proof, we will assume that n is large so that $nc_n^2 \geq 200$ and that $d \geq \ln(10)(1 - c_n)^{-n} + 1$.

Assume, for the sake of contradiction, that there exists a regularizer $R : \mathbb{R}^d \to \mathbb{R}$ such that for any distribution $\mathcal{D} \in \mathscr{D}_c$ the expected suboptimality gap for the RERM solution is at most $\varepsilon/10$. Then, by Markov's inequality, with probability at least 0.9 over the choice of sample set S, the suboptimality gap is at most ε. We will show that ε must be greater than $c_n^2/3200$ for some distribution in the class $\mathscr{D}_c$, hence proving the desired claim.

We first define additional notation. Set $\delta = c_n/10$ and define $\widetilde{\varepsilon} := \varepsilon/\delta$. For the regularization function $R(\cdot)$, define the points w_j^* for $j \in [d]$ such that

$$w_j^* \in \operatorname*{argmin}_{w \text{ s.t. } \|w - e_j\|^2 \leq \widetilde{\varepsilon}} R(w). \tag{61}$$

and define the index $j^* \in [d]$ such that $j^* \in \operatorname{argmax}_{j \in [d]} R(w_j^*)$. We are now ready to prove the desired claim.

Consider the data distribution $\mathcal{D}_1 := \bar{\mathcal{D}}(\delta, c_n, e_{j^*})$ (see Definition 3) and suppose that the dataset $S = \{z_i\}_{i=1}^n$ is sampled i.i.d. from $\mathcal{D}_1$. The population loss $F(w)$ corresponding to $\mathcal{D}_1$ is given by

$$F(w) = \mathbb{E}_{z \sim \mathcal{D}_1}[f(w; z)] = \frac{\delta}{2}\|w - e_{j^*}\|^2 + \max\{1, \|w\|^4\}.$$

Clearly, $F(w)$ is convex in w and thus the distribution $\mathcal{D}_1 \in \mathscr{D}_c$. Furthermore, e_{j^*} is the unique minimizer of $F(\cdot)$, and any ε-suboptimal minimizer w' for $F(\cdot)$ must satisfy

$$\|w' - e_{j^*}\|^2 \leq \frac{2\varepsilon}{\delta} = 2\widetilde{\varepsilon} \leq \frac{1}{5}. \tag{62}$$

To see the above, note that if w' is an ε-suboptimal minimizer of $F(w)$, then $F(w) \leq F(e_{j^*}) + \varepsilon = 1 + \varepsilon$. However, we also have that $F(w') \geq \delta\|w - e_{j^*}\|^2 + 1$. Taking the two inequalities together, and using the fact that $\widetilde{\varepsilon} \leq 1/10$, we get the desired bound on $\|w - e_{j^*}\|$.

Next, define the event E such that the following hold:

(a) For the coordinate j^*, we have $\sum_{z \in S} x[j^*] \leq n(c_n + 2\delta)$.

(b) There exists $\widehat{j}$ such that $\widehat{j} \neq j^*$ and $x[\widehat{j}] = 0$ for all $z \in S$.

(c) RERM (with regularization $R(\cdot)$) using the dataset S returns an ε-suboptimal solution for the test loss $F(w)$.

Since $x[j^*] \sim \mathcal{B}(c_n + \delta)$, Hoeffding's inequality (Lemma 1) implies that the event (a) above occurs with probability at least 0.8 for $n \geq 2/\delta^2 = 200/c_n^2$. Furthermore, Lemma 2 gives us that the event (b) above occurs with probability at least 0.9 for $d \geq \ln(10)(1 - c_n)^{-n}$. Finally, the assumed performance guarantee for RERM with regularization $R(\cdot)$ implies that the event (c) above occurs with probability at least 0.9. Thus, the event E occurs with probability at least 0.6. In the following, we condition on the occurrence of the event E.

Consider the point $w_{\widehat{j}}^*$, defined in (61) corresponding to the coordinate $\widehat{j}$ (that occurs in event E). By definition, we have that $\|w_{\widehat{j}}^* - e_{\widehat{j}}\|^2 \leq \tilde{\varepsilon}$, and thus

$$\begin{aligned}\|w_{\widehat{j}}^* - e_{j^*}\|^2 &\geq \frac{1}{2}\|e_{\widehat{j}} - e_{j^*}\|^2 - \|w_{\widehat{j}}^* - e_{\widehat{j}}\|^2 \\ &\geq 1 - \tilde{\varepsilon} \geq \frac{1}{4},\end{aligned}$$

where the first line above follows from the identity that $(a+b)^2 \leq 2a^2 + 2b^2$ for any $a, b > 0$, and the second line holds because $\widehat{j} \neq j^*$ and because $\widetilde{\varepsilon} \leq 1/10$. As a consequence of the above bound and the condition in (62), we get that the point $w^*_{\widehat{j}}$ is not an ε-suboptimal point for the population loss $F(\cdot)$, and thus would not be the solution of the RERM algorithm (as the RERM solution is ε-suboptimal w.r.t $F(w)$). Since, any RERM must satisfy condition (62), we have that

$$\begin{aligned}
\widehat{F}(w^*_{\widehat{j}}) + R(w^*_{\widehat{j}}) &> \min_{w \colon \left\|w - e_{j^*}\right\|^2 \le \tilde{\varepsilon}} \left(\widehat{F}(w) + R(w)\right) \\
&\ge \min_{w \colon \left\|w - e_{j^*}\right\|^2 \le \tilde{\varepsilon}} \widehat{F}(w) + \min_{w \colon \left\|w - e_{j^*}\right\|^2 \le \tilde{\varepsilon}} R(w) \\
&\overset{(i)}{\ge} \min_{w \colon \left\|w - e_{j^*}\right\|^2 \le \tilde{\varepsilon}} \left(-\frac{c_n}{2}\left\|w - e_{j^*}\right\|^2 + \max\{1, \|w\|^4\}\right) + R(w^*_{j^*}) \\
&\ge \min_{w \colon \left\|w - e_{j^*}\right\|^2 \le \tilde{\varepsilon}} \left(-\frac{c_n}{2}\left\|w - e_{j^*}\right\|^2 + 1\right) + R(w^*_{j^*}) \\
&= -\frac{c_n}{2} \cdot \tilde{\varepsilon} + 1 + R(w^*_{j^*}) \\
&\overset{(ii)}{\ge} -\frac{c_n}{20} + 1 + R(w^*_{j^*})
\end{aligned} \tag{63}$$

where $\widehat{F}(w) := \frac{1}{n}\sum_{i=1}^n f(w; z_i)$ denotes the empirical loss on the dataset S, the inequality (i) follows by ignoring non-negative terms in the empirical loss $\widehat{F}(w)$, and the inequality (ii) is due to the fact that $\tilde{\varepsilon} \le 1/10$. For the left hand side, we note that

$$\begin{aligned}
\widehat{F}(w^*_{\widehat{j}}) &= \frac{1}{2n} \sum_{(x,\alpha) \in S} \|(w^*_{\widehat{j}} - e_{j^*}) \odot x\|^2 - \frac{c_n}{2}\|(w^*_{\widehat{j}} - e_{j^*})\|^2 + \max\{1, \|w^*_{\widehat{j}}\|^4\} \\
&\overset{(i)}{\le} \widetilde{\varepsilon} + \frac{1}{2n} \sum_{(x,\alpha) \in S} \|(e_{\widehat{j}} - e_{j^*}) \odot x\|^2 - \frac{c_n}{2}\|(e_{\widehat{j}} - e_{j^*})\|^2 + \max\{1, (1 + \tilde{\varepsilon})^4\} \\
&\overset{(ii)}{\le} 16\widetilde{\varepsilon} + \frac{1}{2n} \sum_{(x,\alpha) \in S} \|(e_{\widehat{j}} - e_{j^*}) \odot x\|^2 - \frac{c_n}{2}\|(e_{\widehat{j}} - e_{j^*})\|^2 + 1 \\
&\overset{(iii)}{\le} 16\widetilde{\varepsilon} + \frac{1}{2}(c_n + 2\delta) - \frac{\sqrt{2}c_n}{2} + 1,
\end{aligned} \tag{64}$$

where the inequality (i) above holds because $\|w^*_{\widehat{j}} - e_{\widehat{j}}\| \le \tilde{\varepsilon}$ (by definition) and because $\|e_{\widehat{j}}\| = 1$. The inequality in (ii) follows from the fact that $(1 + a)^4 \le 1 + 15a$ for $a < 1$. Finally, the inequality (iii) is due to the fact that $x[\widehat{j}] = 0$ for all $(x, \alpha) \in S$ and because $\sum_{(x,\alpha) \in S} x[j^*] \le n(c_n + 2\delta)$ due to the conditioning on the event E.

Combining the bounds in (63) and (64), plugging in $\delta = c_n/10$, and rearranging the terms, we get that

$$16\widetilde{\varepsilon} \ge \frac{c_n}{20} + R(w^*_{j^*}) - R(w^*_{\widehat{j}}) \ge \frac{c_n}{20},$$

where the second inequality above holds because $j^* \in \operatorname{argmax}_{j \in [d]} R(w^*_j)$ (by definition). Since, $\tilde{\varepsilon} = \varepsilon/\delta$, we conclude that

$$\varepsilon \ge \delta \cdot \frac{c_n}{320} \ge \frac{c_n^2}{3200}.$$

The above suggests that for data distribution $\mathcal{D}_1$, RERM algorithm with the regularization function $R(w)$ will suffer an excess risk of at least $\Omega(c_n^2)$ (with probability at least 0.9). Finally note that $R(w)$ could be any arbitrary function in the above proof, and thus we have the desired claim for any regularization function $R(w)$. □

We now prove Theorem 5-(b). In the following, we give convergence guarantee for SGD algorithm given in (6) when run with step size $\eta = 1/20\sqrt{n} < 1/100$ using the dataset $S = \{z_i\}_{i=1}^n$ drawn i.i.d. from a distribution $\mathcal{D}$. We further assume that assume that the initial point $w_1 = 0$, and thus $\|w_1\| \leq 2.5$. Note that having initial weights to be bounded is typical when learning with non-convex losses, for eg. in deep learning.

Proof of Theorem 5-(b). We consider the two cases, when $D \in \mathscr{D}_c$ and when $D \notin \mathscr{D}_c$, separately below:

Case 1: When $\mathcal{D} \in \mathscr{D}_c$. In this case, we note the population loss $F(w) := \mathbb{E}_{z \sim \mathcal{D}}[f(w; z)]$ is convex in w by the definition of the set $\mathscr{D}_c$ in (8). Further, since $\eta = 1/20\sqrt{n} < 1/100$, Lemma 12 and Lemma 13 imply that f is 70-Lipschitz. Finally, due to Lemma 11, the initial point $w_1 = 0$ satisfies $\|w_1 - w^*\| \leq 2.5$. Thus, we satisfy both Assumption I and Assumption II in (2) and (3) respectively, and an application of Theorem 1 implies that the point $\widehat{w}_n^{\mathrm{SGD}}$ enjoys the performance guarantee

$$\mathbb{E}\big[F(\widehat{w}_n^{\mathrm{SGD}}) - w^* \mid E\big] \leq O\Big(\frac{1}{\sqrt{n}}\Big).$$

Case 2: When $\mathcal{D} \notin \mathscr{D}_c$. In order to prove the performance guarantee of SGD algorithm in this case, we split up the loss function $f(w; z)$ into convex and non-convex parts g and $\widetilde{g}$, defined as:

$$g(w; z) := \frac{1}{2}\|(w - \alpha) \odot x\|^2 + \max\{1, \|w\|^4\} \qquad \text{and} \qquad \widetilde{g}(w; z) := -c_n\|w - \alpha\|^2.$$

Further, we define the functions $G(w)$ and $\widetilde{G}(w)$ to denote their respective population counterparts, i.e. $G(w) = \mathbb{E}_{z \sim \mathcal{D}}[g(w; z)]$ and $\widetilde{G}(w) = \mathbb{E}_{z \sim \mathcal{D}}[\widetilde{g}(w; z)]$. Clearly,

$$F(w) = \mathbb{E}_{z \sim \mathcal{D}}[f(w; z)] = G(w) + \widetilde{G}(w). \tag{65}$$

Let w^* be a minimizer of $F(w; \mathcal{D})$. By Lemma 11, we have $\|w^*\| \leq 1$. The folltowing chain of arguments follows along the lines of proof of SGD in Case 1 above (see the proof of Theorem 8 on page 14).

Let the sequence of iterates generated by SGD algorithm be given by $\{w_t\}_{t=1}^T$. We start by observing that for any $t \geq 0$,

$$\begin{aligned}\|w_{t+1} - w^*\|_2^2 &= \|w_{t+1} - w_t + w_t - w^*\|_2^2 \\ &= \|w_{t+1} - w_t\|_2^2 + \|w_t - w^*\|_2^2 + 2\langle w_{t+1} - w_t, w_t - w^*\rangle \\ &= \|-\eta\nabla f_w(w_t; z_t)\|_2^2 + \|w_t - w^*\|_2^2 + 2\langle -\eta\nabla f_w(w_t; z_t), w_t - w^*\rangle,\end{aligned}$$

where the last line follows from plugging in the update step $w_{t+1} = w_t - \eta\nabla f_w(w_t; z_t)$. Rearranging the terms in the above, we get that

$$\begin{aligned}\langle\nabla f(w_t; z_t), w_t - w^*\rangle &\leq \frac{\eta}{2}\|\nabla f(w_t; z_t)\|_2^2 + \frac{1}{2\eta}\big(\|w_t - w^*\|_2^2 - \|w_{t+1} - w^*\|_2^2\big) \\ &\leq 2450\eta + \frac{1}{2\eta}\big(\|w_t - w^*\|_2^2 - \|w_{t+1} - w^*\|_2^2\big),\end{aligned} \tag{66}$$

where the second inequality in the above follows from the bound on the Lipschitz constant of the function f when the iterates stay bounded in a ball of radius 2.5 around 0 (see Lemma 13). We split the left hand side as:

$$\langle\nabla f(w_t; z_t), w_t - w^*\rangle = \langle\nabla g(w_t; z_t), w_t - w^*\rangle + \langle\nabla\widetilde{g}(w_t; z_t), w_t - w^*\rangle.$$

This implies that

$$\langle\nabla g(w_t; z_t), w_t - w^*\rangle \leq 2450\eta + \frac{1}{2\eta}\big(\|w_t - w^*\|_2^2 - \|w_{t+1} - w^*\|_2^2\big) + \langle\nabla\widetilde{g}(w_t; z_t), w^* - w_t\rangle.$$

Taking expectation on both the sides with respect to the data sample z_t, while conditioning on the point w_t and the occurrence of the event E, we get

$$\langle\nabla G(w_t), w_t - w^*\rangle \leq 2450\eta + \frac{1}{2\eta}\,\mathbb{E}\big[\|w_t - w^*\|_2^2 - \|w_{t+1} - w^*\|_2^2\big] + \langle\nabla\widetilde{G}(w_t), w^* - w_t\rangle.$$

Next, note that the function $G(w)$ is convex (by definition). This implies that $G(w^*) \geq G(w_t) - \langle \nabla G(w_t), w_t - w^* \rangle$, plugging which in the above relation gives us

$$G(w_t) - G(w^*) \leq 2450\eta + \frac{1}{2\eta}\,\mathbb{E}\big[\|w_t - w^*\|_2^2 - \|w_{t+1} - w^*\|_2^2\big] + \langle \nabla \widetilde{G}(w_t), w^* - w_t \rangle.$$

Telescoping the above for t from 1 to n, we get that

$$\begin{aligned}\sum_{t=1}^{n}(G(w_t) - G(w^*)) &\leq 2450\eta n + \frac{1}{2\eta}\big(\|w_1 - w^*\|_2^2 - \|w_{n+1} - w^*\|_2^2\big) + \sum_{t=1}^{n}\langle \nabla \widetilde{G}(w_t), w^* - w_t \rangle \\ &\leq 2450\eta n + \frac{\|w_1 - w^*\|_2^2}{2\eta} + \sum_{t=1}^{n}\|\nabla \widetilde{G}(w_t)\|\|w^* - w_t\|,\end{aligned}$$

where the inequality in the second line follows by ignoring negative terms, and through an application of Cauchy-Schwarz inequality. Since, $\|w_1\| = 0$ and $\|w^*\| \leq 1$ (due to Lemma 11), we have that $\|w_1 - w^*\| \leq 1$. Setting $\eta = 1/20\sqrt{n}$, using the bound $\|w_1 - w^*\| \leq 1$ and by an application of Jensen's inequality in the left hand side, we get that the point $\widehat{w}_n^{\mathrm{SGD}} := \frac{1}{n}\sum_{t=1}^{n} w_t$ satisfies

$$G(\widehat{w}_n^{\mathrm{SGD}}) - G(w^*) \leq \frac{245}{\sqrt{n}} + \frac{1}{n}\sum_{t=1}^{n}\|\nabla \widetilde{G}(w_t)\|\|w^* - w_t\|. \tag{67}$$

Furthermore, by Lemma 12, since $\eta = \frac{1}{20\sqrt{n}} \leq \frac{1}{100}$ and $\|w_1\| \leq 2.5$ (by construction), we have that $\|w_t\| \leq 2.5$ for all t. Thus, $\|w^* - w_t\| \leq 3.5$ for all t. Plugging this bound in (67), we get that

$$G(\widehat{w}_n^{\mathrm{SGD}}) - G(w^*) \leq \frac{245}{\sqrt{n}} + 3.5 \max_{t \leq n}\, \|\nabla \widetilde{G}(w_t)\|.$$

Next, note that from the definition of the function $\widetilde{G}(w_t)$, we have that

$$\|\nabla \widetilde{G}(w_t)\| = c_n\|w_t - \alpha\| \leq 3.5c_n,$$

and thus

$$G(\widehat{w}_n^{\mathrm{SGD}}) - G(w^*) \leq \frac{245}{\sqrt{n}} + 3.5c_n. \tag{68}$$

Finally, using the relation (65) and taking expectations on both the sides, we have that

$$\begin{aligned}\mathbb{E}\big[F(\widehat{w}_n^{\mathrm{SGD}}) - F(w^*) \mid E\big] &= \mathbb{E}\big[G(\widehat{w}_n^{\mathrm{SGD}}) - G(w^*)\big] + \mathbb{E}[\widetilde{G}(\widehat{w}_n^{\mathrm{SGD}}) - \widetilde{G}(w^*)] \\ &\leq \frac{245}{\sqrt{n}} + 3.5c_n + c_n\,\mathbb{E}_z[w^* - \alpha\|^2] \\ &\leq \frac{245}{\sqrt{n}} + 6c_n, \end{aligned} \tag{69}$$

where the inequality in the second line is due to (68) and by using the definition of the function $\widetilde{G}(w)$ and by ignoring negative terms. The last line holds because $\|w^* - \alpha\| \leq \|w^*\| + \|\alpha\| \leq 2$. □

F Missing proofs from Section 7

F.1 α-Linearizable functions

In the following, provide the proof of the performance guarantee for SGD for α-Linearizable functions.

Proof of Theorem 6. The proof follows along the lines of the proof of Theorem 8 on page 14. Let $\{w_t\}_{t\geq 1}$ denote the sequence of iterates generated by the SGD algorithm. We note that for any $w^* \in \operatorname{argmin}_w F(w)$ and time $t \geq 1$,

$$\|w_{t+1} - w^*\|_2^2 = \|w_{t+1} - w_t + w_t - w^*\|_2^2$$

$$\begin{aligned} &= \|w_{t+1} - w_t\|_2^2 + \|w_t - w^*\|_2^2 + 2\langle w_{t+1} - w_t, w_t - w^*\rangle \\ &= \|-\eta \nabla f(w_t; z_t)\|_2^2 + \|w_t - w^*\|_2^2 + 2\langle -\eta \nabla f(w_t; z_t), w_t - w^*\rangle, \end{aligned}$$

where the last line follows from plugging in the SGD update rule that $w_{t+1} = w_t - \eta \nabla f(w_t; z_t)$. Rearranging the terms in the above, we get that

$$\langle \nabla f(w_t; z_t), w_t - w^*\rangle \leq \frac{\eta}{2}\|\nabla f(w_t; z_t)\|_2^2 + \frac{1}{2\eta}\big(\|w_t - w^*\|_2^2 - \|w_{t+1} - w^*\|_2^2\big).$$

Taking expectation on both the sides, while conditioning on the point w_t, implies that

$$\begin{aligned} \langle \nabla F(w_t), w_t - w^*\rangle &\leq \frac{\eta}{2}\,\mathbb{E}\big[\|\nabla f(w_t; z_t)\|_2^2\big] + \frac{1}{2\eta}\big(\|w_t - w^*\|_2^2 - \|w_{t+1} - w^*\|_2^2\big) \\ &\leq \eta\,\mathbb{E}\big[\|\nabla f(w_t; z_t) - \nabla F(w_t)\|_2^2\big] + \eta\|\nabla F(w_t)\|_2^2 \\ &\qquad + \frac{1}{2\eta}\big(\|w_t - w^*\|_2^2 - \|w_{t+1} - w^*\|_2^2\big) \\ &\leq \eta(\sigma^2 + L^2) + \frac{1}{2\eta}\big(\|w_t - w^*\|_2^2 - \|w_{t+1} - w^*\|_2^2\big), \end{aligned} \tag{70}$$

where the inequality in the second line is given by the fact that $(a-b)^2 \leq 2a^2 + 2b^2$ and the last line follows from using Assumption II (see (3)) which implies that $F(w)$ is L-Lipschitz in w and that $\mathbb{E}[\|\nabla f(w; z_t) - \nabla F(w)\|_2^2] \leq \sigma^2$ for any w. Next, using the fact that F is α-Linearizable, we have that there exists a $w^* \in \operatorname{argmin}_w F(w)$ such that for any w,

$$F(w) - F(w^*) \leq \alpha\langle \nabla F(w), w - w^*\rangle. \tag{71}$$

Setting w^* to the one that satisfies (71), and using the above bound for $w = w_t$ with the bound in (70), we get that for any $t \geq 1$,

$$\frac{1}{\alpha}(F(w_t) - F^*) \leq \eta(\sigma^2 + L^2) + \frac{1}{2\eta}\big(\|w_t - w^*\|_2^2 - \|w_{t+1} - w^*\|_2^2\big),$$

where $F^* := F(w^*)$. Telescoping the above for t from 1 to n, we get that

$$\begin{aligned} \sum_{t=1}^{n} \frac{1}{\alpha}(F(w_t) - F^*) &\leq \eta n(\sigma^2 + L^2) + \frac{1}{2\eta}\big(\|w_1 - w^*\|_2^2 - \|w_{n+1} - w^*\|_2^2\big) \\ &\leq \eta n(\sigma^2 + L^2) + \frac{1}{2\eta}\|w_1 - w^*\|_2^2. \end{aligned}$$

Dividing both the sides by n, we get that

$$\frac{1}{\alpha n}\sum_{t=1}^{n}(F(w_t) - F^*) \leq \eta(\sigma^2 + L^2) + \frac{1}{2\eta n}\|w_1 - w^*\|_2^2.$$

An application of Jensen's inequality on the left hand side, implies that for the point $\widehat{w}_n^{\mathrm{SGD}} := \frac{1}{n}\sum_{t=1}^{n} w_t$, we have that

$$\frac{1}{\alpha}\,\mathbb{E}\big[F(\widehat{w}_n^{\mathrm{SGD}}) - F^*\big] \leq \eta(\sigma^2 + L^2) + \frac{1}{2\eta n}\|w_1 - w^*\|_2^2.$$

Setting $\eta = \frac{1}{\sqrt{n}}$ and using the fact that $\|w_1 - w^*\|_2 \leq B$ in the above, we get that

$$\mathbb{E}\big[F(\widehat{w}_n^{\mathrm{SGD}}) - F^*\big] \leq \frac{\alpha}{\sqrt{n}}\big(\sigma^2 + L^2 + B^2\big),$$

which is the desired claim. Dependence on the problem specific constants (σ, L and B) in the above bound can be improved further with a different choice of the step size η; getting the optimal dependence on these constants, however, is not the focus of this work. □

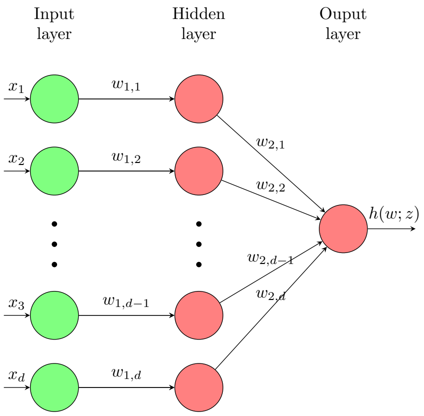

Figure 2: Two layer diagonal neural network. The weights are given by $w_1 \in \mathbb{R}^d$ and $w_2 \in \mathbb{R}^d$ for the first layer and the second layer respectively. The green nodes denote input nodes with linear activation function and red node denote hidden units with ReLU activation function.

F.2 Proof of Theorem 7

Let the input $\mathcal{X} \in \{0,1\}^d$ and the label $\mathcal{Y} \in [-1,1]$. Consider a two layer neural network with ReLU activation and weights given by $w = (w_1, w_2)$ where $w_1 \in \mathbb{R}^d$ denotes the weights of the first layer and $w_2 \in \mathbb{R}^d$ denotes the weights of the second layer, as shown in Figure 2. When given the input x, the output of the network with weights (w_1, w_2) is computed as

$$h(w;x) = \text{ReLU}(w_2^\top \text{ReLU}(w_1 \odot x)).$$

Here, the first layer of the neural network has sparse connections, i.e. each input node connects to only one hidden node. Such networks are denoted as diagonal two layer neural networks. We assume that the neural network is trained using absolute loss function. Specifically, the instantaneous loss on a sample $z = (x, y)$ is given by

$$f(w;z) = |y - h(w;x)| = |y - \text{ReLU}(w_2^\top \text{ReLU}(w_1 \odot x))|. \tag{72}$$

Since $f(w;z)$ is not smooth, for any weights $w = (w_1, w_2)$ and sample $z = (x, y)$, we define the gradient of $f(w;z)$ as

$$\nabla_{w_1} \ell(w;z)[i] = -\operatorname{sign}\{y - h(w;x)\} \cdot \mathbf{1}\{w_2^\top \text{ReLU}(w_1 \odot x) > 0\} \cdot (w_{2,i} \cdot \mathbf{1}\{w_{1,i} \cdot x_i > 0\} \cdot x_i)$$

and

$$\nabla_{w_2} \ell(w;z)[i] = -\operatorname{sign}\{y - h(w;x)\} \cdot \mathbf{1}\{w_2^\top \text{ReLU}(w_1 \odot x) > 0\} \cdot (\text{ReLU}(w_{1,i} \cdot x_i)), \tag{73}$$

for $i \in [d]$. We next show that the population loss $F(w) := \mathbb{E}_{z \sim \mathcal{D}}[f(w;z)]$ is $1/2$-Linearizable.

Lemma 14. *Let $\mathcal{D}$ be defined such that x and y are independent random variables with distributions*

$$x \sim \textit{Uniform}(\{0,1\}^d) \qquad \textit{and} \qquad y \sim \mathcal{B}(1/4).$$

Then, the population loss $F(w) := \mathbb{E}_{z \sim \mathcal{D}}[f(w;z)]$ is $1/2$-Linearizable.

Proof. The population loss $F(w)$ is given by:

$$F(w) = \mathbb{E}_{x,y}[|y - h(w;x)|],$$

where $h(w;x) = \text{ReLU}(w_2^\top \text{ReLU}(w_1 \odot x))$. Using the fact that $y \in \{-1, 1\}$, and $\Pr(y = 1) = 1/4$ and is independent of x, the above can be written as:

$$F(w) = \mathbb{E}_x \left[\frac{3}{4} |h(w;x)| + \frac{1}{4} |1 - h(w;x)| \right].$$

It is easy to verify that $F(w)$ is minimized when $h(w;x) = 0$ for every $x \in \{0,1\}^d$, which occurs at the point $w = 0$. Furthermore, $F(0) = 1/4$. Next, note that for any w, and sample $z = (x, y)$,

$$\begin{aligned}
\langle w, \nabla_w f(w;z)\rangle &= \langle w_1, \nabla_{w_1} f(w;z)\rangle + \langle w_2, \nabla_{w_2} f(w;z)\rangle \\
&\overset{(i)}{=} -\sum_{i=1}^{d} w_{1,i} \cdot \, \mathrm{sign}\{y - h(w;x)\} \cdot \mathbf{1}\{w_2^\top \mathrm{ReLU}(w_1 \odot x) > 0\} \cdot (w_{2,i} \cdot \mathbf{1}\{w_{1,i} \cdot x_i > 0\} \cdot x_i) \\
&\qquad\quad - \sum_{i=1}^{d} w_{2,d} \cdot \, \mathrm{sign}\{y - h(w;x)\} \cdot \mathbf{1}\{w_2^\top \mathrm{ReLU}(w_1 \odot x) > 0\} \cdot (\mathrm{ReLU}(w_{1,i} \cdot x_i)) \\
&= -\,\mathrm{sign}\{y - h(w;x)\} \cdot \mathbf{1}\{w_2^\top \mathrm{ReLU}(w_1 \odot x) > 0\} \cdot (w_2^\top \mathrm{ReLU}(w_1 \odot x)) \\
&\qquad\quad - \,\mathrm{sign}\{y - h(w;x)\} \cdot \mathbf{1}\{w_2^\top \mathrm{ReLU}(w_1 \odot x) > 0\} \cdot (w_2^\top \mathrm{ReLU}(w_1 \odot x)) \\
&= -2\,\mathrm{sign}\{y - h(w;x)\}\, \mathrm{ReLU}(w_2^\top \mathrm{ReLU}(w_1 \odot x)) \\
&= -2\,\mathrm{sign}\{y - h(w;x)\}\, h(w;x) \\
&= 2|y - h(w;x)| - 2\,\mathrm{sign}\{y - h(w;x)\} y
\end{aligned}$$

where the equality (i) follows from using the definition of $\nabla_{w_1} f(w;z)$ and $\nabla_{w_2} f(w;z)$ from (73). Taking expectations on both the sides with respect to z, we get that

$$\begin{aligned}
\langle w, \nabla_w F(w)\rangle &= 2\,\mathbb{E}_{x,y}[|y - h(w;x)|] - 2\,\mathbb{E}_{x,y}[\,\mathrm{sign}\{y - h(w;x)\} y] \\
&= 2F(w) - 2\,\mathbb{E}_{x,y}[\,\mathrm{sign}\{y - h(w;x)\} y] \\
&\geq 2F(w) - 2\,\mathbb{E}_{x,y}[|y|] \\
&= 2(F(w) - F(0)),
\end{aligned}$$

where the last line follows by observing that $\mathbb{E}[|y|] = \mathbb{E}_{x,y}[|y| - h(0;x)] = F(0)$. Defining $w^* := 0$, the above implies that

$$F(w) - F(w^*) \leq \frac{1}{2}\langle w - w^*, \nabla_w F(w)\rangle,$$

thus showing that $F(w)$ is $1/2$-Linearizable. □

Proof of Theorem 7. Consider the distribution $\mathcal{D}$ over the instance space $\mathcal{Z} = \{0,1\}^d \times \{0,1\}$ where

$$x \sim \text{Uniform}(\{0,1\}^d), \qquad \text{and} \qquad y \sim \mathcal{B}(1/4). \tag{74}$$

We now prove the two parts separately below:

(a) In the following, we show that SGD algorithm, run with an additional projection on the unit norm ball after every update, learns at a rate of $O(1/\sqrt{n})$. In particular, we use the following update step for $t \in [n]$,

$$w_{t+1}^{\text{SGD}} \leftarrow \Pi_{w_1}\big(w_t^{\text{SGD}} - \eta \nabla f(w_t^{\text{SGD}}; z_t)\big)$$

where the initial point w_1 is chosen by first sampling $w_1' \sim \mathcal{N}(0, \mathbb{I}_d)$ and then setting $w_1 = w_1'/\|w_1\|$, and the projection operation Π is given by

$$\Pi_{w_1}(w) = \begin{cases} w & \text{if } \|w - w_1\| \leq 1 \\ w_1 + \frac{1}{\|w - w_1\|}(w - w_1) & \text{otherwise.} \end{cases}$$

After taking n steps, the point returned by the SGD algorithm is given by $\widehat{w}_n^{\text{SGD}} := \sum_{t=1}^{n} w_t^{\text{SGD}}/n$.

First note that, for the distribution given in (74), Lemma 14 implies that the population loss $F(w)$ is $1/2$-Liearizable w.r.t. the global minima $w^* = 0$. Next, note that for any point w and data sample z,

$$\|\nabla f(w;z)\|^2 = \|\nabla_{w_1} f(w;z)\|^2 + \|\nabla_{w_2} f(w;z)\|^2$$

$$
\begin{aligned}
&\leq \sum_{j=1}^{d} (w_{2,j} \cdot x_j)^2 + \sum_{j=1}^{d} (\mathrm{ReLU}(w_{1,j} \cdot x_j))^2 \\
&\leq \sum_{j=1}^{d} w_{2,j}^2 + \sum_{j=1}^{d} w_{1,j}^2 \\
&= \|w\|^2
\end{aligned}
$$

where the inequality in the second line follows by plugging in the definition of $\nabla_{w_1} f(w; z)$ and $\nabla_{w_2} f(w; z)$, and by upper bounding the respective indicators by 1. The inequality in the third line above holds because $\mathrm{ReLU}(h) \leq |h|$ for any $h \in \mathbb{R}$, and by using the fact that $x_j \in \{0, 1\}$ for $j \in [d]$. Since, the iterates produced by SGD algorithm satisfy $\|w_t^{\text{SGD}}\| \leq 1$ due to the projection step, the above bound implies that $\|\nabla f(w_t^{\text{SGD}}; z)\| \leq 1$ for any $t \geq 0$, and thus $\|\nabla F(w_t^{\text{SGD}})\| \leq 1$.

The above bounds imply that Assumption II (in (3)) holds on the iterates generated by the SGD algorithm with $\max\{B, \sigma, L\} \leq 2$. Furthermore, the population loss $F(w)$ is $1/2$-Linearizable. Thus, repeating the steps from the proof of Theorem 6 on page 46, and using the fact that $\|\Pi_{w_1}(w) - w^*\| \leq \|w - w^*\|$ for $w^* = 0$ to account for the additional projection step, we get that the point $\widehat{w}_n^{\text{SGD}}$ returned by the SGD algorithm enjoys the performance guarantee

$$
F(\widehat{w}_n^{\text{SGD}}) - F^* \leq O\Big(\frac{1}{\sqrt{n}}\Big).
$$

(b) In the following, we will show that for $d > \log(10)2^n + 1$, with probability at least 0.9 over the choice of $S \sim \mathcal{D}^n$, there exists an ERM solution for which

$$
F(w_{\text{ERM}}) - \inf_{w \in \mathbb{R}^d} F(w) \geq \Omega(1).
$$

Suppose that the dataset $S = \{(x_i, y_i)\}_{i=1}^n$ is sampled i.i.d. from the distribution $\mathcal{D}$ given in (74). A slight modification of Lemma 6 implies that for $n \leq \log_2(d/\log(10))$, with probability at least 0.9, there exists a coordinate $\widehat{j}$ such that $x_i[\widehat{j}] = y_i$ for all $i \in [n]$. In the following, we condition on the occurrence of such a coordinate $\widehat{j}$. Clearly, the empirical loss at the point $\widehat{w} = (e_{\widehat{j}}, e_{\widehat{j}})$ is:

$$
\widehat{F}(\widehat{w}) = \sum_{i=1}^{n} |y_i - \mathrm{ReLU}(e_{\widehat{j}} \mathrm{ReLU}(e_{\widehat{j}} \odot x_i))| = \sum_{i=1}^{n} |y_i - x_i[\widehat{j}]| = 0,
$$

where the last equality follows the fact that $x_i[\widehat{j}] = y_i$ for all $i \in [n]$. Since $\widehat{F}(w) \geq 0$ for any w, we get that the point $\widehat{w}$ is an ERM solution. Next, we note that the population loss at the point $\widehat{w}$ satisfies

$$
\begin{aligned}
F(\widehat{w}) - \min_{w} F(w) &\overset{(i)}{=} F(\widehat{w}) - \frac{1}{4} \\
&\overset{(ii)}{=} \mathbb{E}_{x,y}\Big[\Big|y - \mathrm{ReLU}(e_{\widehat{j}} \mathrm{ReLU}(e_{\widehat{j}} \odot x))\Big|\Big] - \frac{1}{4} \\
&= \mathbb{E}_{x,y}[|y - x[\widehat{j}]|] - \frac{1}{4} \overset{(iii)}{=} \frac{1}{4}
\end{aligned}
$$

where the equality (i) follows by observing that $\min_w F(w) = 1/4$ (see the proof of Lemma 14 for details), the equality (ii) holds for $\widehat{w} = (e_{\widehat{j}}, e_{\widehat{j}})$ and (iii) follows by using the fact that $y \sim \mathcal{B}(1/4)$ and $x[\widehat{j}] \sim \mathcal{B}(1/2)$ and that x and y are sampled independent to each other. Thus, there exists an ERM solution $w_{\text{ERM}} = \widehat{w}$ for which the excess risk:

$$
F(w_{\text{ERM}}) - \min_{w} F(w) = \frac{1}{4}.
$$

The desired claim follows by observing that the coordinate $\widehat{j}$ above occurs with probability at least 0.9 over the choice of $S \sim \mathcal{D}^n$.

□

F.3 Expressing $f_{(A)}$ and $f_{(B)}$ using neural networks

In this section, we show that the loss functions $f_{(A)}$ and $f_{(B)}$ can be represented using restricted neural networks (where some of the weights are fixed to be constant and thus not trained) with poly(d) hidden units. We first provide neural network constructions that in addition to ReLU, use square function $\sigma(a) = a^2$ and square root function $\widetilde{\sigma}(a) = \sqrt{a}$ as activation functions. We then give a general representation result in Lemma 15 which implies that the activations σ and $\widetilde{\sigma}$ can be approximated both in value and in gradients simultaneously using poly(d) number of ReLU units. This suggests that the functions $f_{(A)}$ and $f_{(B)}$ can be represented using restricted RELU networks of poly(d) size.

Note that in our constructions, the NN approximates the corresponding functionst in both value and in terms of the gradient. Thus, running gradient based optimization algorithms on the corresponding NN representation will produce similar solutions as running the same algorithm on the actual function that it is approximating.

Proposition 3. *Function $f_{(A)}$ in Equation (A) can be represented as a restricted diagonal neural network with square and square root activation functions with $O(d)$ units and constant depth.*

Proof. In the following, we will assume that before passing to the neural network, each data sample $z = (x, \alpha, y)$ is preprocessed to get the features $\widetilde{x}$ defined as $\tilde{x} := (x, -\alpha \odot x)^T \in \mathbb{R}^{2d}$. The vector $\widetilde{x}$ is given as the input to the neural network.

We construct a three layer neural network with input $\widetilde{x} \in \mathbb{R}^{2d}$, and weight matrices $W_1 \in \mathbb{R}^{2d \times 2d}$, $W_2 \in \mathbb{R}^{2d \times d}$ and $W_3 \in \mathbb{R}^{d \times 1}$ for the first layer, the second layer and the third layer respectively. The activation function after the ith layer is given by $\sigma_i : \mathbb{R} \mapsto \mathbb{R}$. The neural network consists of d trainable parameters, given by $w \in \mathbb{R}^d$, and is denoted by the function $h(w; \widetilde{x}) : \mathbb{R}^d \times \mathbb{R}^{2d} \mapsto \mathbb{R}$. In the following, we describe the corresponding weight matrices and activation functions for each layer of the network:

- **Layer 1:** Input: $\widetilde{x} \in \mathbb{R}^{2d}$. The weight matrix $W_1 : \mathbb{R}^{2d \times 2d}$ is given by

$$W_1[i,j] := \begin{cases} w[j] & \text{if } 1 \leq i = j \leq d \\ 1 & \text{if } d < i = j \leq 2d \\ 0 & \text{otherwise} \end{cases} .$$

  The activation function σ_1 is given by $\sigma_1(a) = a$. Let $h_1(w; \widetilde{x}) := \sigma_1(\widetilde{x} W_1)$ denote the output of this layer. We thus have that $h_1(w; \widetilde{x}) = (w \odot x, -\alpha \odot x)^T$.

- **Layer 2:** Input: $h_1(w; \widetilde{x}) \in \mathbb{R}^{2d}$. The weight matrix $W_2 : \mathbb{R}^{2d \times d}$ is given by

$$W_1[i,j] := \begin{cases} 1 & \text{if } i - j \in \{0, d\} \\ 0 & \text{otherwise} \end{cases}$$

  The activation function σ_2 is given by $\sigma_2(a) = a^2$. Let $h_2(w; \widetilde{x}) := \sigma_2(h_1(\widetilde{z}, w) W_2)$ denote the output of this layer. We thus have that $h_2(w; \widetilde{x})[j] = (w[j] \odot x[j] - \alpha[j] \odot x[j])^2$ for any $j \in [d]$.

- **Layer 2:** Input: $h_2(w; \widetilde{x}) \in \mathbb{R}^d$. The weight matrix $W_3 : \mathbb{R}^{d \times 1}$ is given by the vector $\mathbf{1}_d = (1, \dots, 1)^T$.

  The activation function σ_2 is given by $\sigma_2(a) = \sqrt{a}$. Let $h_3(w; \widetilde{x}) := \sigma_3(h_2(w; \widetilde{x}) W_3)$ denote the output of this layer. We thus have that $h_3(w; \widetilde{x})[j] = \|w \odot x - \alpha \odot x\|$.

Thus, the output of the neural network is

$$\begin{aligned} h(w; \widetilde{x}) &= h_3(w; \widetilde{x}) \\ &= \sigma_3(\sigma_2(\sigma_1(\widetilde{x} W_1) W_2) W_3) = \|(w - \alpha) \odot x\|. \end{aligned}$$

Except for the first d diagonal elements of W_1, all the other weights of the network are kept fixed during training. Thus, the above construction represents a restricted neural network with trainable parameters given by w. Also note that in this first layer, any input node connects with a single node in the second layer. Such networks are known as diagonal neural networks [Gunasekar et al., 2018b].

We assume that the network is trained using linear loss function, i.e. for the prediction $h(w;\widetilde{x})$ for data point $(\widetilde{x}, y)$, the loss is given by

$$\ell(w;\widetilde{x}) = y \cdot h(w;\widetilde{x}) = y\|(w-\alpha)\odot x\|, \tag{75}$$

Note that the above expression exactly represents $f_{(A)}(w;z)$. This suggests that learning with the loss function $f_{(A)}$ is equivalent to learning with the neural network h (defined above with trainable parameter given by w) with linear loss. Furthermore, the network h has a constant depth, and $O(d)$ units, proving the desired claim. □

We next show how to express the loss function $f_{(B)}$ using a neural network.

Proposition 4. *Function $f_{(B)}$ in Equation (B) can be represented as a restricted diagonal neural network with square and square root activation functions with $O(d)$ units and constant depth.*

Proof of Proposition 4 . In the following, we will assume that before passing to the neural network, each data sample $z = (x, \alpha)$ is preprocessed to get the features $\widetilde{x} := (x, -\alpha\odot x, \mathbf{1}_d, -\alpha, \mathbf{1}_d, \mathbf{1}_d)^T \in \mathbb{R}^{6d}$. The vector $\widetilde{x}$ is given as the input to the neural network.

We construct a four layer neural neural network with input $\widetilde{x} \in \mathbb{R}^{6d}$ and weight matrices $W_1 \in \mathbb{R}^{6d\times 6d}$, $W_2 \in \mathbb{R}^{6d\times 3d}$, $W_3 \in \mathbb{R}^{3d\times 2}$, $W_4 \in \mathbb{R}^{2\times 1}$ for the four layers respectively. The activation function after the ith layer is given by $\sigma_i : \mathbb{R} \mapsto \mathbb{R}$. The neural network consists of d trainable parameters, given by $w \in \mathbb{R}^d$, and is denoted by the function $h(w;\widetilde{x}) : \mathbb{R}^d \times \mathbb{R}^{6d} \mapsto \mathbb{R}$. In the following, we describe the corresponding weight matrices and activation functions for each layer of the network:

- **Layer 1:** Input: $\widetilde{x} \in \mathbb{R}^{4d}$. The weight matrix $W_1 : \mathbb{R}^{6d\times 6d}$ is given by

  $$W_1[i,j] := \begin{cases} w[j] & \text{if } i = j \text{ and } j - \alpha d \leq d \text{ for } \alpha \in \{0,2,4\} \\ 1 & \text{if } i = j \text{ and } j - \alpha d \leq d \text{ for } \alpha \in \{1,3,5\} \\ 0 & \text{otherwise} \end{cases}.$$

  The activation function σ_1 is given by $\sigma_1(a) = a$. Let $h_1(w;\widetilde{x}) := \sigma_1(\widetilde{x}W_1)$ denote the output of this layer. We thus have that $h_1(w;\widetilde{x}) = (w\odot x, -\alpha\odot x, w, -\alpha, w, \mathbf{1}_d)^T$.

- **Layer 2:** Input: $h_1(w;\widetilde{x}) \in \mathbb{R}^{6d}$. The weight matrix $W_2 : \mathbb{R}^{6d\times 3d}$ is given by

  $$W_1[i,j] := \begin{cases} 1 & \text{if } i - j \in \{0, d\} \text{ and } j \leq d \\ 1 & \text{if } i - j \in \{d, 2d\} \text{ and } d < j \leq 2d \\ 1 & \text{if } i = j + 3d \text{ and } 2d < j \\ 0 & \text{otherwise} \end{cases}.$$

  The activation function σ_2 is given by $\sigma_2(a) = a^2$. Let $h_2(w;\widetilde{x}) := \sigma_2(h_1(\widetilde{z}, w)W_2)$ denote the output of this layer. We thus have that $h_2(w;\widetilde{x})[j] = (w[j]\odot x[j] - \alpha[j]\odot x[j])^2$ for any $j \in [d]$, $h_2(w;\widetilde{x})[j] = (w[j] - \alpha[j])^2$ for any $d < j \leq 2d$ and $h_2(w;\widetilde{x})[j] = (w[j])^2$ for $2d < j \leq 3d$.

- **Layer 3:** Input: $h_2(w;\widetilde{x}) \in \mathbb{R}^{3d}$. The weight matrix $W_3 : \mathbb{R}^{3d\times 2}$ is given by

  $$W_3[i,j] = \begin{cases} \frac{1}{2} & \text{if } j = 1 \text{ and } 1 \leq i \leq d \\ -\frac{c_n}{2} & \text{if } j = 1 \text{ and } d + 1 \leq i \leq 2d \\ 1 & \text{if } j = 2 \text{ and } 2d + 1 \leq i \leq 3d \end{cases}.$$

  For the first node in the output of this layer, we use the activation function $\sigma_2(a) = \sqrt{a}$ and for the second node, we use the activation function $\sigma_2(a) = \max\{1, a^2\}$. Let $h_3(w;\widetilde{x}) := \sigma_3(h_2(w;\widetilde{x})W_3)$ denote the output of this layer. We thus have that $h_3(w;\widetilde{x})[j] = (\frac{1}{2}\|(w-\alpha)\odot x\|^2 - \frac{c_n}{2}\|(w-\alpha)\odot x\|^2, \max\{1, \|w\|^4\})$.

- **Layer 4:** Input: $h_3(w; \widetilde{x}) \in \mathbb{R}^2$. The weight matrix $W_4 : \mathbb{R}^{2\times 1}$ is given by $W_4 = (1, 1)$, and the activation function σ_4 is given by $\sigma_4(a) = a$. Let $h_4(w; \widetilde{x}) := \sigma_3(h_2(w; \widetilde{x})W_3)$ denote the output of this layer. We thus have that $h_4(w; \widetilde{x})[j] = \frac{1}{2}\|(w - \alpha) \odot x\|^2 - \frac{c_n}{2}\|(w - \alpha) \odot x\|^2 + \max\{1, \|w\|^4\}$.

Thus, the output of the neural network is given by

$$\begin{aligned}
h(w; \widetilde{x}) &= h_4(w; \widetilde{x}) \\
&= \sigma_4(\sigma_3(\sigma_2(\sigma_1(\widetilde{x}W_1)W_2)W_3)W_4) \\
&= \frac{1}{2}\|(w - \alpha) \odot x\|^2 - \frac{c_n}{2}\|(w - \alpha) \odot x\|^2 + \max\{1, \|w\|^4\}.
\end{aligned}$$

In the above construction, the first layer can be thought of as a convolution with filter weights given by $\mathrm{diag}(w, \mathbf{1}_d)$ and stride $2d$. While training the neural network, we keep all the weights of the network fixed except for the ones that take values from w (in the weight W_1). Thus, the above construction represents a restricted CNN with trainable parameters given by w.

Furthermore, for the prediction $h(w; \widetilde{x})$ for data point $\widetilde{x}$, we treat the output of the neural network as the loss, which is given by

$$\begin{aligned}
\ell(w; \widetilde{x}) &= h(w; \widetilde{x}) + \max\{1, \|w\|^4\} \\
&= \frac{1}{2}\|(w - \alpha) \odot x\|^2 - \frac{c_n}{2}\|(w - \alpha) \odot x\|^2 + \max\{1, \|w\|^4\}.
\end{aligned}$$

Note that the above expression exactly represents $f_{(B)}(w; z)$. This suggests that learning with the loss function $f_{(B)}$ is equivalent to learning with the neural network h (defined above with trainable parameter given by w). Furthermore, the network h has a constant depth, and $O(d)$ units, proving the desired claim. □

We next provide a general representation result which implies that the activation functions $\sigma(a) = a^2$ (square function) and $\widetilde{\sigma}(a) = \sqrt{a}$ (square root function), used in the constructions above, can be approximated both in value and in gradients simultaneously using poly(d) number of ReLU units.

Lemma 15. *Let $f : [a, b] \to \mathbb{R}$ be an L-Lipschitz and α-smooth function. Then for any $\varepsilon > 0$, there is a function $h : [a, b] \to \mathbb{R}$ that can be written as a linear combination h of $\lceil \frac{(b-a)\max\{L,\alpha\}}{\varepsilon} \rceil + 1$ ReLUs with coefficients bounded by $2L$ such that for all $x \in [a, b]$, we have $|f(x) - h(x)| \le \varepsilon$, and if h is differentiable at x, then $|f'(x) - h'(x)| \le \varepsilon$.*

Proof. Consider dividing up the interval $[a, b]$ into equal n equal sized intervals of length $\delta = \frac{b-a}{n}$, for $n = \lceil \frac{(b-a)\max\{L,\alpha\}}{\varepsilon} \rceil$. Define $a_i = a + i\delta$ for $i = 0, 1, \ldots, n$. Let h be a piecewise linear function that interpolates f on the $n+1$ endpoints of the intervals. For any such interval $[a_i, a_{i+1}]$, by the mean value theorem, the slope of h on the interval is equal to $f'(x_i)$ for some $x_i \in (a_i, a_{i+1})$, and hence is bounded by L since f is L-Lipschitz. Furthermore, by the L-Lipschitzness and α-smoothness of f, for any $x \in (a_i, a_{i+1})$, we have

$$|f(x) - h(x)| \le \delta L \le \varepsilon$$

and

$$|f'(x) - h'(x)| = |f'(x) - h'(x_i)| = |f'(x) - f'(x_i)| \le \delta\alpha \le \varepsilon.$$

Now, by Lemma 16 we can represent h as a linear combination of $n + 1$ ReLUs with coefficients bounded by $2L$. □

Lemma 16. *Any piecewise linear function $f : \mathbb{R} \to \mathbb{R}$ with K segments can be written as a linear combination of $K + 2$ ReLUs with coefficients bounded by twice the maximum slope of any segment of f.*

Proof. A piecewise linear function $f : \mathbb{R} \to \mathbb{R}$ is fully determined by the endpoints $a_1 < a_2 < \cdots < a_K$ for some positive integer K which define the segments of f, the value $f(a_1)$, and the slopes

$m_0, m_1, \ldots, m_K \in \mathbb{R}$, such that the slope of f on the segment (a_i, a_{i+1}) is m_i, where we define $a_0 := -\infty$ and $a_{K+1} = +\infty$ for convenience. Specifically, we can write f as the following:

$$f(x) = f(a_1) + \begin{cases} m_0(x - a_1) & \text{if } x < a_1 \\ \sum_{i=1}^{\ell-1} m_i(a_{i+1} - a_i) + m_\ell(x - a_\ell) & \text{if } x \in [a_\ell, a_{\ell+1}) \end{cases}$$

Now define $\sigma(x) = \max\{x, 0\}$ to be the ReLU function, and consider the function h defined as

$$h(x) = f(a_1) - m_0\sigma(a_1 - x) + m_0\sigma(x - a_1) + \sum_{i=1}^{K}(m_i - m_{i-1})\sigma(x - a_i).$$

Since this is a linear combination of ReLUs, it is a piecewise linear function. By direct calculation, one can check that $h(a_1) = f(a_1)$, the endpoints of the segments of h are $a_1, a_2, \ldots, a_K$, and the slopes of the segments are $m_0, m_1, \ldots, m_K$. Hence, the function $f = h$. □