On the Training Instability of Shuffling SGD with Batch Normalization

David X. Wu [1] **Chulhee Yun** [2] **Suvrit Sra** [3]

Abstract

We uncover how SGD interacts with batch normalization and can exhibit undesirable training dynamics such as divergence. More precisely, we study how Single Shuffle (SS) and Random Reshuffle (RR)—two widely used variants of SGD—interact surprisingly differently in the presence of batch normalization: *RR leads to much more stable evolution of training loss than SS*. As a concrete example, for regression using a linear network with batch normalized inputs, we prove that SS and RR converge to distinct global optima that are "distorted" away from gradient descent. Thereafter, for classification we characterize conditions under which training divergence for SS and RR can, and cannot occur. We present explicit constructions to show how SS leads to distorted optima in regression and divergence for classification, whereas RR avoids both distortion and divergence. We validate our results empirically in realistic settings, and conclude that the separation between SS and RR used with batch normalization is relevant in practice.

1. Introduction

Recent work in deep learning theory attempts to uncover how the choice of optimization algorithm and architecture influence training stability and efficiency. On the optimization front, stochastic gradient descent (SGD) is the *de facto* workhorse, and its importance has correspondingly led to the development of many different variants that aim to increase the ease and speed of training, such as AdaGrad (Duchi et al., 2011) and Adam (Kingma & Ba, 2014).

In reality, practitioners often do not use with-replacement sampling of gradients as required by SGD. Instead they use *without-replacement* sampling, leading to two main variants of SGD: single-shuffle (SS) and random-reshuffle. SS randomly samples and fixes a permutation at the beginning of training, while RR randomly resamples permutations at each epoch. These shuffling algorithms are often more practical and can have improved convergence rates (Haochen & Sra, 2019; Safran & Shamir, 2020; Yun et al., 2021b; 2022; Cho & Yun, 2023; Cha et al., 2023).

Architecture design offers another avenue for practitioners to train networks more efficiently and encode salient inductive biases. Normalizing layers such as BatchNorm (BN) (Ioffe & Szegedy, 2015), LayerNorm (Ba et al., 2016), or InstanceNorm (Ulyanov et al., 2016) are often used with SGD to accelerate convergence and stabilize training. Recent work studies how these scale-invariant layers affect training through the effective learning rate (Li & Arora, 2019; Li et al., 2020; Wan et al., 2021; Lyu et al., 2022).

Motivated by these practical choices, we study how SS and RR interact with batch normalization at *training time*. Our experiments (Fig. 1) suggest that combining SS and BN can lead to surprising and undesirable training phenomena:

(i) The training risk often diverges when using SS+BN to train linear networks (i.e. without nonlinear activations) on real datasets (see Figure 1a), while using SS without BN does not cause divergence (see Figure 10).

(ii) Divergence persists after tuning the learning rate and other hyperparameters (Section 4.3) and also manifests more quickly in deeper linear networks (Figure 1a).

(iii) SS+BN usually converges slower than RR+BN in nonlinear architectures such as ResNet18 (see Figure 1b).

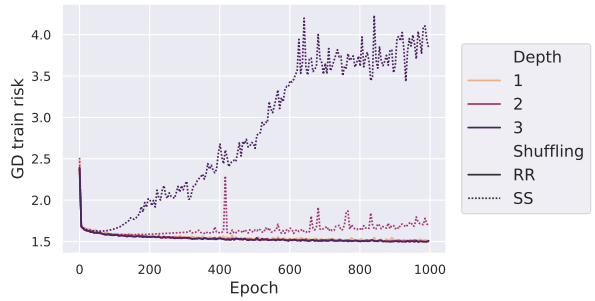
1.1. Summary of our contributions

In light of these experimental findings, we seek to develop a theoretical and experimental understanding of how shuffling SGD and BN collude to create divergence and other undesirable training behavior. Since these phenomena manifest themselves on the training risk, our results are not strictly coupled with generalization.

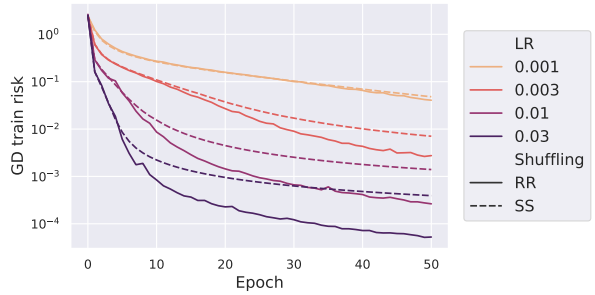
Put simply, the aberrant training dynamics stem from BN *not* being permutation invariant across epochs. This simple property interacts with SS undesirably, although *a priori* it is not obvious whether it should. More concretely, one expects SGD+BN to optimize the gradient descent (GD)

[1]Department of EECS, UC Berkeley, Berkeley, CA, USA [2]Kim Jaechul Graduate School of AI, KAIST, Seoul, Korea [3]Department of EECS, LIDS, MIT, Cambridge, MA, USA. Correspondence to: David X. Wu <david_wu@berkeley.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.



(a) Depths 1, 2, and 3 linear+BN networks.



(b) Finetuned ResNet18.

Figure 1. Surprising training phenomena using SS/RR+BN.

risk in expectation. However, due to BN's sensitivity to permutations, both SS+BN and RR+BN implicitly train induced risks different from GD, and also from each other.

- In Section 3.2, we prove that the network $f(\boldsymbol{X};\Theta) = \boldsymbol{W}\boldsymbol{\Gamma}\mathsf{BN}(\boldsymbol{X})$ which batch normalizes the input features converges to the optimum for the distorted risk induced by SS and RR (Theorems 3.2.2 and 3.2.3); the diagonal matrix $\boldsymbol{\Gamma}$ denotes the *trainable* scale parameters in the BN layer. Our proof requires a delicate analysis of the evolution of gradients, the noise arising from SS, and the two-layer architecture. Due to the presence of $\boldsymbol{\Gamma}$, our results do not assume a fully-connected linear network, which distinguishes them from prior convergence results. In Section 3.3, we present a toy dataset for which SS is distorted away from GD with constant probability while RR averages out the distortion to align with GD. We validate our theoretical findings on synthetic data in Section 3.4.
- In Section 4.1, we connect properties of the distorted risks to divergence of training risk, gaining insights into which regimes can lead to divergence (Theorems 4.1.3 and 4.1.4). We show that in certain regimes, SS+BN can diverge, whereas RR+BN provably avoids divergence. These results motivate us to construct a toy dataset where SS diverges with constant probability, while RR avoids divergence (Section 4.2). In Section 4.3, we empirically validate our results on deeper linear+BN networks on a variety of datasets and hyperparameters. Our experiments also demonstrate that SS trains more slowly than RR in more realistic nonlinear settings, including ReLU+BN networks and ResNet18. In doing so, we extend the relevance of our theoretical results to more complex and realistic settings.

It is worth noting that to obtain our results, our analysis had to overcome complications due to the non-i.i.d. stochastic gradients, the non-i.i.d. data $\boldsymbol{X}$ (Assumption 2), and the intricacies introduced by BN's permutation sensitivity.

1.2. Related work

Theoretical understanding of BN. Since the introduction of BN by Ioffe & Szegedy (2015), there has been a long line of work investigating the theoretical properties of BN; see e.g. (Bjorck et al., 2018; Kohler et al., 2018; Arora et al., 2018; Li & Arora, 2019; Kohler et al., 2019; Daneshmand et al., 2020; Li et al., 2020; Lobacheva et al., 2021). Much attention has been devoted to studying how BN can benefit optimization (Arora et al., 2018; Santurkar et al., 2018; Kohler et al., 2018), for example by implicitly tuning the learning rate or smoothing the loss function. The effect of BN on the intermediate representations of random networks, such as orthogonality or rank collapse, has also been studied (Daneshmand et al., 2020; 2021). We study the general setting with nonrandom linear activations. The scale invariance induced by BN also interacts with other optimization choices such as weight decay, which can lead to instability phenomena (Lobacheva et al., 2021; Wan et al., 2021; Lyu et al., 2022). However, these phenomena have a different origin than the distorted risks studied in this paper.

Interplay between BN and SGD. Prior theoretical work primarily studied how BN interacts with GD or with-replacement SGD (Arora et al., 2018; Santurkar et al., 2018; Li & Arora, 2019; Cai et al., 2019; Wan et al., 2021; Lyu et al., 2022). Arora et al. (2018); Wan et al. (2021) assumed global bounds on the smoothness with respect to network parameters and the SGD noise to analyze convergence to stationary points. We instead prove convergence to the global minimum of the SS distorted risk $\mathcal{L}_\pi$ with *no* such assumptions (Theorem 3.2.2). Li & Arora (2019) assumed the batch size is large enough to ignore SGD noise, whereas we explicitly exhibit and study the separation between shuffling SGD and GD. For fully scale-invariant networks trained with GD, Lyu et al. (2022) identified an oscillatory edge of stability behavior around a manifold of minimizers. Our BN network has trainable scale-variant parameters $\boldsymbol{W}$ and $\boldsymbol{\Gamma}$, and we train with shuffling SGD instead of GD. Hence, the noise that leads to distorted risks is fundamentally different.

BN's effect on risk function. Previous work identified the distortion of risk function due to noisy batch statistics in BN. Yong et al. (2020) studied the asymptotic regularization effect of noisy batch statistics *in expectation* for with-replacement SGD. In contrast, we characterize this noise

nonasymptotically w.h.p. over π for SS and a.s. with respect to the data for RR. Wu & Johnson (2021) studied the difficulty of precisely estimating the population statistics at train time, especially when using an exponential moving average. We avoid these issues altogether by evaluating directly on the GD risk. Moreover, we prove concentration inequalities for without-replacement batch statistics (Proposition C.2.4).

Ghost batch normalization. In the presence of BN, it is common practice to use *ghost batch normalization*, a scheme which break up large batches into virtual "ghost" batches, as this tends to improve the generalization of the network (Hoffer et al., 2017; Shallue et al., 2019; Summers & Dinneen, 2020). Minibatch statistics are calculated with respect to the ghost batches, and each gradient step is computed by summing the gradient contributions from the ghost batches. This algorithm is closely related to our method of analysis for SS+BN/RR+BN. Indeed, in our setup we also break up the full batch into mini-batches, and our analysis reduces to showing that SS+BN and RR+BN trajectories track those obtained by following the aggregate gradient signal from summing over mini-batches. We comment more on the similarities between ghost BN and our setup in Section 3.1.

Shuffling and optimization. Outside SGD, the effect of random shuffling has also been studied for classical nonlinear optimization schemes such as coordinate gradient descent (CGD) and ADMM (see Sun et al. (2020); Gürbüzbalaban et al. (2020) and references therein). On convex quadratic optimization problems, they demonstrate separations in convergence rates between SS, RR, and with-replacement sampling. Our main focus is the optimum that the algorithms converge to rather than their convergence rates.

Implicit bias. Our work is also motivated by a burgeoning line of work which studies the *implicit bias* of different optimization algorithms (Soudry et al., 2018; Gunasekar et al., 2018; Ji & Telgarsky, 2018; 2019; 2020; Yun et al., 2021a; Jagadeesan et al., 2022). These results establish how optimization algorithms such as gradient flow (GF), gradient descent (GD) or even with-replacement SGD are biased towards certain optima. For example, in the interpolating regime, GD converges to the min-norm solution (Gunasekar et al., 2018; Woodworth et al., 2020) for linear regression and the max-margin classifier for classification (Soudry et al., 2018; Nacson et al., 2019b;a). While our work does not focus on generalization, it is connected in spirit to implicit bias. Indeed, our analysis centers the study of how the risk functions and optima are affected by choices of the optimizer (SS/RR) and the architecture (BN).

2. Problem setup

For $n \in \mathbb{Z}^+$ we use the notation $[n] \triangleq \{1, \dots, n\}$. We write π to denote a permutation of $[n]$, and $\mathbb{S}_n$ is the symmetric group of all such π. For any matrix $\boldsymbol{A} \in \mathbb{R}^{d\times n}$, $\pi \circ \boldsymbol{A} \in \mathbb{R}^{d\times n}$ is result of shuffling the columns of $\boldsymbol{A}$ according to π. Also, $\|\boldsymbol{A}\|_2$ and $\|\boldsymbol{A}\|_F$ refer to the spectral norm and Frobenius norm, respectively. We write $\sigma_{\min}(\boldsymbol{A}) \triangleq \inf_{\|\boldsymbol{v}\|=1} \|\boldsymbol{A}\boldsymbol{v}\|$ to denote minimum singular value of $\boldsymbol{A}$. We use $\mathrm{Span}(\boldsymbol{A})$ to denote the span of $\boldsymbol{A}$'s columns. The (coordinatewise) sign function $\mathrm{sgn}(\cdot) : \mathbb{R} \to \{-1, 0, 1\}$ is defined as $\mathrm{sgn}(x) = x/|x|$ for $x \neq 0$ and $\mathrm{sgn}(0) = 0$.

Data. Let $\boldsymbol{Z} = (\boldsymbol{X}, \boldsymbol{Y})$ be the given dataset, with $\boldsymbol{X} = \begin{bmatrix}\boldsymbol{x}_1 & \cdots & \boldsymbol{x}_n\end{bmatrix} \in \mathbb{R}^{d\times n}$ representing the feature matrix and corresponding labels $\boldsymbol{Y} = \begin{bmatrix}\boldsymbol{y}_1 & \cdots & \boldsymbol{y}_n\end{bmatrix} \in \mathbb{R}^{p\times n}$. In the classification setting we will assume $\boldsymbol{Y} \in \{\pm 1\}^{1\times n}$.

Prediction model. A batch normalization (BN) layer can be separated into a normalizing component BN and a scaling component $\boldsymbol{\Gamma}$; we ignore the bias parameters for analysis. Given any matrix $\boldsymbol{B} = \begin{bmatrix}\boldsymbol{x}_1 & \cdots & \boldsymbol{x}_q\end{bmatrix} \in \mathbb{R}^{d\times q}$ (here, $q \geq 2$ is arbitrary), the normalizing transform $\mathsf{BN}(\cdot)$ maps it to $\mathsf{BN}(\boldsymbol{B}) \in \mathbb{R}^{d\times q}$ by operating coordinatewise on each $\boldsymbol{x}_i$ in $\boldsymbol{B}$. In particular, for the kth coordinate of $\boldsymbol{x}_i$, denoted as $x_{i,k}$, the transform BN sends $x_{i,k} \mapsto \frac{x_{i,k}-\mu_k}{\sqrt{\sigma_k^2+\epsilon}}$ where μ_k and σ_k^2 are the batch empirical mean and variance of the kth coordinate, respectively, and ϵ is an arbitrary positive constant used to avoid numerical instability. For technical reasons, we omit ϵ in our analysis. The scaling matrix $\boldsymbol{\Gamma} \in \mathbb{R}^{d\times d}$ is a diagonal matrix which models the tunable coordinatewise scale parameters inside the BN layer.

Throughout the paper, we consider neural networks of the form $f(\cdot; \Theta) = \boldsymbol{W}\boldsymbol{\Gamma}\mathsf{BN}(\cdot)$[1]. We use $\Theta = (\boldsymbol{W}, \boldsymbol{\Gamma})$ to denote the collection of all parameters in the network. With the presence of batch normalization layers, the output of f is a function of the input datapoint *as well as* the batch it belongs to. Even changing one point of a batch $\boldsymbol{B}$ can affect the batch statistics (i.e., μ_k's and σ_k^2's) and in turn change the outputs of f for the entire batch.

Loss functions. We study regression with squared loss $\ell(\hat{\boldsymbol{y}}, \boldsymbol{y}) \triangleq \|\hat{\boldsymbol{y}} - \boldsymbol{y}\|^2$ and binary classification with logistic loss $\ell(\hat{y}, y) \triangleq -\log(\rho(y\hat{y}))$, where $\rho(t) = 1/(1 + e^{-t})$. Let $\hat{\boldsymbol{Y}}, \boldsymbol{Y} \in \mathbb{R}^{p\times q}$ denote network outputs and true labels for a mini-batch of q datapoints, respectively. Define the mini-batch risk as the columnwise sum $\mathcal{L}(\hat{\boldsymbol{Y}}, \boldsymbol{Y}) \triangleq \sum_{i=1}^q \ell(\hat{\boldsymbol{Y}}_{:,i}, \boldsymbol{Y}_{:,i})$, where $\boldsymbol{Y}_{:,i}$ denotes the ith column of $\boldsymbol{Y}$.

Optimization methods. We consider shuffling-based variants of SGD, namely single-shuffle (SS) and random-reshuffle (RR). These algorithms proceed in *epochs*, i.e., full passes through shuffled dataset. As the names suggest, SS randomly samples a permutation $\pi \in \mathbb{S}_n$ at the beginning of the first epoch and adheres to this permutation. RR

[1] We can readily generalize to arbitrary learned (but frozen) feature mappings under suitable changes to the assumptions.

randomly resamples permutations $\pi_k \in \mathbb{S}_n$ at each epoch k.

Throughout, the (mini-)batch size will be denoted as B. For simplicity, we assume that the n datapoints can be divided into m batches of size B. With a permutation $\pi \in \mathbb{S}_n$, the dataset $\boldsymbol{Z} = (\boldsymbol{X}, \boldsymbol{Y})$ is thus perfectly partitioned into m batches $(\boldsymbol{X}_\pi^1, \boldsymbol{Y}_\pi^1), \ldots, (\boldsymbol{X}_\pi^m, \boldsymbol{Y}_\pi^m)$, where $\boldsymbol{X}_\pi^j \in \mathbb{R}^{d\times B}$ and $\boldsymbol{Y}_\pi^j \in \mathbb{R}^{p\times B}$ consist of the $(j(B-1)+1, \ldots, jB)$th columns of the shuffled $\pi \circ \boldsymbol{X}$ and $\pi \circ \boldsymbol{Y}$, respectively.

For a parameter Θ optimized with SS or RR, we denote the jth iterate on the kth epoch by Θ_j^k. The starting iterate of the kth epoch is Θ_0^k which is equal to the last iterate of the previous epoch Θ_m^{k-1}. For each $j \in [m]$, SS and RR perform a mini-batch SGD update with stepsize $\eta_k > 0$:

$$\Theta_j^k \leftarrow \Theta_{j-1}^k - \eta_k \nabla_\Theta \mathcal{L}(f(\boldsymbol{X}_{\pi_k}^j; \Theta_{j-1}^k), \boldsymbol{Y}_{\pi_k}^j).$$

3. Main regression results: convergence to optima of distorted risks

In this section, we introduce the framework of distorted risks to elucidate the distinction between SS+BN and RR+BN. For ease of theoretical analysis, we consider the simplified setup where BN is applied only to the input features, although we note that the framework can be readily generalized to any learned but frozen features. This framework also applies to classification; we continue to study it in Section 4. We then present our global convergence results (Theorems 3.2.2 and 3.2.3) for the distorted risks induced by SS and RR for squared loss regression. In the one-dimensional case, we uncover an averaging relationship between the SS and RR optima (Proposition 3.3.1) which can help RR reduce distortion. We exemplify this averaging relationship with a simple example and extend it to higher dimensions with experiments on synthetic data.

3.1. Framework: the idea of distorted risks

We now formally introduce the notion of a *distorted risk*. Distorted risks are crucial to our analysis, as they encode the interaction between shuffling SGD and BN. We show that these distorted risks $\mathcal{L}_\pi$ and $\mathcal{L}_{\mathrm{RR}}$ are respectively induced by certain batch normalized datasets $\overline{\boldsymbol{X}}_\pi$ and $\overline{\boldsymbol{X}}_{\mathrm{RR}}$ obtained by batch normalizing the input features.

The *undistorted* risk we actually want to minimize is the risk that corresponds to full-batch GD. Define the GD features $\overline{\boldsymbol{X}}_{\mathrm{GD}} \triangleq \mathsf{BN}(X)$, which induces the *GD risk*:

$$\mathcal{L}_{\mathrm{GD}}(\Theta) \triangleq \mathcal{L}(f(\boldsymbol{X}; \Theta), \boldsymbol{Y}) = \mathcal{L}(\boldsymbol{W}\boldsymbol{\Gamma}\overline{\boldsymbol{X}}_{\mathrm{GD}}, \boldsymbol{Y}).$$

However, during epoch k, SS or RR optimize a distorted risk dependent on π_k. To see why, define the SS dataset

$$\overline{\boldsymbol{X}}_\pi \triangleq \mathsf{BN}_\pi(\boldsymbol{X}) \triangleq \begin{bmatrix}\mathsf{BN}(\boldsymbol{X}_\pi^1) & \cdots & \mathsf{BN}(\boldsymbol{X}_\pi^m)\end{bmatrix}$$
$$\boldsymbol{Y}_\pi \triangleq \begin{bmatrix}\boldsymbol{Y}_\pi^1 & \cdots & \boldsymbol{Y}_\pi^m\end{bmatrix},$$

for every permutation $\pi \in \mathbb{S}_n$. Similarly, form the RR dataset $(\overline{\boldsymbol{X}}_{\mathrm{RR}}, \boldsymbol{Y}_{\mathrm{RR}}) \in \mathbb{R}^{d\times(n\cdot n!)} \times \mathbb{R}^{p\times(n\cdot n!)}$ by concatenating the SS datasets $(\overline{\boldsymbol{X}}_\pi, \boldsymbol{Y}_\pi)$ across all π.

Crucially, the SS data $\overline{\boldsymbol{X}}_\pi$ encodes the distortion due to the interaction between SS with permutation π and BN; the RR data $\overline{\boldsymbol{X}}_{\mathrm{RR}}$ does the same for RR and BN. Indeed, since SS uses a fixed π, it implicitly optimizes the SS distorted risk

$$\mathcal{L}_\pi(\Theta) \triangleq \sum_{j=1}^m \mathcal{L}(f(\boldsymbol{X}_\pi^j; \Theta), \boldsymbol{Y}_\pi^j) = \mathcal{L}(\boldsymbol{W}\boldsymbol{\Gamma}\overline{\boldsymbol{X}}_\pi, \boldsymbol{Y}_\pi).$$

Likewise, by collapsing the epoch update into a noisy "SGD" update, we observe that RR over epochs implicitly optimizes the RR distorted risk

$$\mathcal{L}_{\mathrm{RR}}(\Theta) \triangleq \frac{1}{n!}\sum_{\pi\in\mathbb{S}_n} \mathcal{L}_\pi(\Theta) = \frac{1}{n!}\mathcal{L}(\boldsymbol{W}\boldsymbol{\Gamma}\overline{\boldsymbol{X}}_{\mathrm{RR}}, \boldsymbol{Y}_{\mathrm{RR}}).$$

We reiterate that SS and RR distortions originate from using *both* shuffling and batch normalization: shuffling alters the batch-dependent affine transforms that BN applies. With this notation, the connection between SS+BN/RR+BN and ghost BN becomes more evident: one can view the full batch as the batch in ghost BN and the mini-batches as the virtual ghost batches. Moreover, the proofs of Theorems 3.2.2 and 3.2.3 demonstrate that ghost BN would witness the same type of distortion as SS+BN/RR+BN.

To aid clarity, we adopt the convention that overlines connote batch normalization with *some* batching, and vice versa. For example, the SS dataset $\overline{\boldsymbol{X}}_\pi \triangleq \mathsf{BN}_\pi(\boldsymbol{X})$ is normalized, while the shuffled dataset $\boldsymbol{X}_\pi = \pi \circ \boldsymbol{X}$ is not.

3.2. Convergence results for regression

We now present our main regression results: SS+BN and RR+BN converge to the global optima of their respective distorted risks encoded by the SS dataset $\overline{\boldsymbol{X}}_\pi$ and the RR dataset $\overline{\boldsymbol{X}}_{\mathrm{RR}}$. We require the following rank assumptions.

Assumption 1 (Full rank assumption)**.**

(a) $\overline{\boldsymbol{X}}_\pi \in \mathbb{R}^{d\times n}$ *satisfies* $\mathrm{rank}(\overline{\boldsymbol{X}}_\pi) \geq d$. *In particular,* $\sigma_{\min}(\overline{\boldsymbol{X}}_\pi \overline{\boldsymbol{X}}_\pi^\top) > 0$.

(b) $\overline{\boldsymbol{X}}_{\mathrm{RR}} \in \mathbb{R}^{d\times(n\cdot n!)}$ *satisfies* $\mathrm{rank}(\overline{\boldsymbol{X}}_{\mathrm{RR}}) \geq d$. *In particular,* $\sigma_{\min}(\overline{\boldsymbol{X}}_{\mathrm{RR}} \overline{\boldsymbol{X}}_{\mathrm{RR}}^\top) > 0$.

It is natural to ask when Assumption 1 holds. We demonstrate that the following mild assumption implies it; the assumption states that the feature matrix $\boldsymbol{X}$ is drawn from a joint density on matrices in a *potentially non-i.i.d. fashion*.

Assumption 2. $\boldsymbol{X}$ *is drawn from a density with respect to the Lebesgue measure on* $\mathbb{R}^{d\times n}$.

Since BN centers the mini-batch features, we have $\mathrm{rank}(\overline{\boldsymbol{X}}_\pi) \leq \min\left\{d, (B-1)\frac{n}{B}\right\}$ and $\mathrm{rank}(\overline{\boldsymbol{X}}_{\mathrm{RR}}) \leq$

$\min\{d, (B-1)\binom{n}{B}\}$.[2] We now show that if $B > 2$ these upper bounds are achieved almost surely. Thus, we identify reasonable conditions under which Assumption 1 holds almost surely over the draw of data, irrespective of shuffling.

Proposition 3.2.1. *Assume Assumption 2 and $B > 2$. Then we have $\operatorname{rank}(\overline{\boldsymbol{X}}_\pi) = \min\{d, (B-1)\frac{n}{B}\}$ and $\operatorname{rank}(\overline{\boldsymbol{X}}_{\mathrm{RR}}) = \min\{d, (B-1)\binom{n}{B}\}$ a.s.. Consequently, if $(B-1)\frac{n}{B} \geq d$, Assumption 1(a) holds a.s. for $\overline{\boldsymbol{X}}_\pi$, and if $(B-1)\binom{n}{B} \geq d$, Assumption 1(b) holds a.s. for $\overline{\boldsymbol{X}}_{\mathrm{RR}}$.*

Although we could have just assumed Assumption 1, the nonlinearity introduced by BN makes it nontrivial to identify mild sufficient conditions on the original features to control the rank of SS and RR datasets. Furthermore, controlling the rank of these datasets is crucial to our analysis of GD risk divergence in the classification setting (see Section 4).

Next, we present our main SS convergence result: SS converges for appropriate stepsizes. We defer the proof and explicit convergence rates to Appendix A.1.

Theorem 3.2.2 (Convergence of SS)**.** *Let $f(\cdot;\Theta) = \boldsymbol{W}\boldsymbol{\Gamma}\mathsf{BN}(\cdot)$ be a linear+BN network initialized at $\Theta_0^1 = (\boldsymbol{W}_0^1, \boldsymbol{\Gamma}_0^1) = (\boldsymbol{0}, \boldsymbol{I})$. We train f using SS with permutation π and suppose that Assumption 1(a) holds for this π. SS uses the following decreasing stepsize, which is well-defined:*

$$\eta_k = \frac{1}{k^\beta} \cdot \min\left\{O\left(\frac{1}{\sigma_{\min}(\overline{\boldsymbol{X}}_\pi \overline{\boldsymbol{X}}_\pi^\top)}\right), \frac{\sqrt{2\beta-1}\,\mathrm{poly}(\sigma_{\min}(\overline{\boldsymbol{X}}_\pi^\top))}{\mathrm{poly}(n,d,\|\boldsymbol{Y}\|_F)}\right\},$$

where $1/2 < \beta < 1$. Then the risk $\mathcal{L}_\pi(\Theta_0^k)$ converges to the global minimum $\mathcal{L}_\pi^*$ as $k \to \infty$.

Theorem 3.2.2 shows that using both SS and BN induces the network to converge to the global optimum of the SS distorted risk instead of the usual GD risk. The proof proceeds by aggregating the epoch-wise gradient updates on the collapsed matrix $\boldsymbol{W}\boldsymbol{\Gamma}$. The main difficulty lies in carefully bounding the accumulation of various types of noise.

We now turn to RR convergence. For the sake of analysis, we make the following compact iterates assumption which is common in the RR literature (Haochen & Sra, 2019; Nagaraj et al., 2019; Ahn et al., 2020; Rajput et al., 2020).

Assumption 3. *For all (i,k), the iterates $\Theta_i^k = (\boldsymbol{W}_i^k, \boldsymbol{\Gamma}_i^k)$ satisfy $\left\|\boldsymbol{W}_i^k \boldsymbol{\Gamma}_i^k\right\|_2 \leq A_{\mathrm{RR}}$ for some absolute constant A_{RR}.*

Finally, we can show that RR converges in expectation to the global optimum of the RR distorted risk $\mathcal{L}_{\mathrm{RR}}$. We defer the proof and explicit convergence rates to Appendix A.2.

Theorem 3.2.3 (Convergence of RR)**.** *Assume Assumption 1(b) and Assumption 3. Using the same f and initialization as in Theorem 3.2.2. we train training f using RR with the following decreasing stepsize, which is well-defined:*

$$\eta_k = \frac{1}{k^\beta} \cdot \min\left\{O\left(\frac{1}{\sigma_{\min}(\overline{\boldsymbol{X}}_{\mathrm{RR}} \overline{\boldsymbol{X}}_{\mathrm{RR}}^\top)}\right), \frac{\sqrt{2\beta-1}}{\mathrm{poly}(n,d,\|\boldsymbol{Y}\|_F, A_{\mathrm{RR}})}\right\},$$

where $1/2 < \beta < 1$. Then the risk $\mathcal{L}_{\mathrm{RR}}(\Theta_0^k)$ converges in expectation to the global minimum $\mathcal{L}_{\mathrm{RR}}^*$ as $k \to \infty$.

The proof of Theorem 3.2.3 is similar to the SS case; the main subtlety is using Assumption 3 to handle expectations.

The main takeaway of Theorems 3.2.2 and 3.2.3 is that SS+BN and RR+BN converge to the optima of the SS and RR distorted risks, respectively. These distorted optima may differ from optimum of the GD risk. Moreover, the required stepsize for convergence is usually smaller for SS (where the requirement depends on π) compared to RR.

3.3. RR averages out SS distortion

Having shown that the two different algorithms drive the network parameters to global optima of two different distorted risks, it behooves us to study these optima. By collapsing the final layers $\boldsymbol{W}$ and $\boldsymbol{\Gamma}$ into a single matrix $\boldsymbol{M} = \boldsymbol{W}\boldsymbol{\Gamma} \in \mathbb{R}^{p\times d}$, we can study the global optima $\boldsymbol{M}_\pi^*$ and $\boldsymbol{M}_{\mathrm{RR}}^*$ on the normalized datasets $\overline{\boldsymbol{X}}_\pi$ and $\overline{\boldsymbol{X}}_{\mathrm{RR}}$. These global optima naturally correspond to the global optima of $\mathcal{L}_\pi$ and $\mathcal{L}_{\mathrm{RR}}$. In this section we illustrate how RR can average out SS distortion in the one-dimensional case.

We first relate the SS optima $\boldsymbol{M}_\pi^*$ to the RR optimum $\boldsymbol{M}_{\mathrm{RR}}^*$. A simple gradient calculation reveals $\boldsymbol{M}_{\mathrm{RR}}^* = \sum_\pi \boldsymbol{Y}_\pi \overline{\boldsymbol{X}}_\pi^\top (\sum_\pi \overline{\boldsymbol{X}}_\pi \overline{\boldsymbol{X}}_\pi^\top)^{-1}$. Since BN enforces the unit variance constraint, $\overline{\boldsymbol{X}}_\pi \overline{\boldsymbol{X}}_\pi^\top = n$ if $d = 1$. Simple algebraic manipulation then implies the following proposition.

Proposition 3.3.1. *If $d = 1$, $\boldsymbol{M}_{\mathrm{RR}}^* = \frac{1}{n!}\sum_{\pi\in\mathbb{S}_n} \boldsymbol{M}_\pi^*$.*

Proposition 3.3.1 identifies an explicit averaging relationship between RR and SS in the one-dimensional case. This motivates the following simple construction where RR's averaging behavior removes SS distortion.

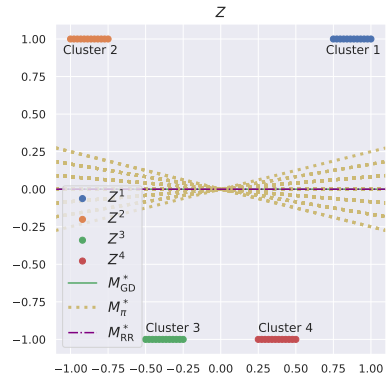
Dataset: SS distorted with constant probability, RR averages out distortion. We visualize our toy dataset with $16n$ datapoints where $d = p = 1$, $B = 2$, and $n = 3$ in Figure 2a, along with the possible SS optima $\boldsymbol{M}_\pi^*$. The dataset is comprised of four clusters of $4n$ points in the square $[-1,1]^2$. By vertical symmetry of the clusters and Proposition 3.3.1, the *RR and GD optima coincide at zero*. However, SS is distorted away from GD. An anticoncentration calculation shows $\boldsymbol{M}_\pi^* \neq 0$ with probability $1 - O(\frac{1}{\sqrt{n}})$ and $|\boldsymbol{M}_\pi^*| = \Omega(\frac{1}{\sqrt{n}})$ with constant probability. The key insight is linking SS distortion to breaking symmetry in the SS dataset (see Proposition E.1.1 for details).

[2] Note that $\overline{\boldsymbol{X}}_{\mathrm{RR}}$ contains many duplicate batches; only $\binom{n}{B}$ of them are unique, up to permutations of B columns inside a batch.

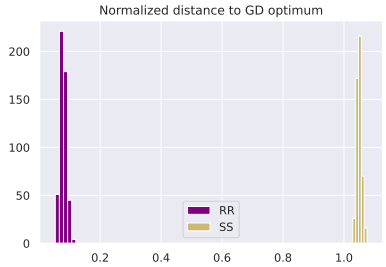
3.4. Regression experiments

For our regression experiments, we used synthetic data with $n = 100$, $B = 10$, and $d = 10$. For $i \in [n]$, we sampled $\boldsymbol{x}_i \sim N(\boldsymbol{0}, \boldsymbol{I}_d)$ and generated $y_i = \boldsymbol{M}_{\text{true}}\boldsymbol{x}_i + \epsilon_i \in \mathbb{R}$ with $\boldsymbol{M}_{\text{true}} \sim U[-1, 1]^d$ and $\epsilon_i \sim N(0, 1)$. We trained the network $\boldsymbol{W}\boldsymbol{\Gamma}\mathsf{BN}(\boldsymbol{X})$ using SS and RR with an inverse learning rate schedule initialized at $\eta = 0.01$. We observed convergence to near optimal values on the SS and RR risks (Figure 7), which supports the convergence results (Theorems 3.2.2 and 3.2.3).

We also extended the toy dataset to the synthetic setup described above. As Figure 2b makes apparent, SS is consistently distorted away from the GD optimum, whereas RR averages out this distortion effect. We generated 500 datasets and evaluated the distortion for each one with the normalized distance $d(\boldsymbol{M}) \triangleq \frac{\|\boldsymbol{M}-\boldsymbol{M}^*_{\text{GD}}\|}{\|\boldsymbol{M}_{\text{GD}*}\|}$. For SS, we computed the mean $d(\boldsymbol{M}^*_\pi)$ for 1000 random draws of π. For RR, we approximated $d(\boldsymbol{M}^*_{\text{RR}})$ as follows. We sampled 1000 fresh random permutations to approximate the RR dataset $\overline{\boldsymbol{X}}_{\text{RR}}$, which we then used to approximate $\boldsymbol{M}^*_{\text{RR}}$ (since it is intractable to average over all $n!$ permutations). We see that $d(\boldsymbol{M}^*_\pi) > 1$ for all of the SS experiments while $d(\boldsymbol{M}^*_\pi) \approx 0.1$ for all of the RR experiments.



(a) Dataset with 48 datapoints demonstrating distortion of SS optima $\boldsymbol{M}^*_\pi$.



(b) Normalized distance to GD optimum $d(\boldsymbol{M}) = \|\boldsymbol{M} - \boldsymbol{M}^*_{\text{GD}}\|/\|\boldsymbol{M}^*_{\text{GD}}\|$.

Figure 2. Top: toy dataset for regression, showing how RR can average out the distortion of SS. Bottom: histogram of distortion of SS and RR optima on synthetic data for $d = 10$. The SS optima significantly deviate from the GD optima, whereas the RR optima are relatively close. This supports the intuition that RR can nontrivially smooth out the bias of SS in higher dimensions.

4. Main classification results: divergence regimes based on distorted risks

We now turn to analyzing linear+BN binary classifiers $f(\boldsymbol{X}; \Theta) = \text{sgn}(\boldsymbol{W}\boldsymbol{\Gamma}\mathsf{BN}(\boldsymbol{X}))$ trained with the logistic risk. To characterize divergence, we identify salient properties of the distorted risks first introduced in Section 3.1. These properties identify regimes where the SS+BN classifier can diverge on the GD risk (Theorem 4.1.3) yet the RR+BN classifier does not diverge (Theorem 4.1.4). This motivates the construction of a toy dataset (Section 4.2) where the optimal SS classifier diverges on the GD risk with constant probability. In Section 4.3 we extend our results to more realistic networks and datasets, demonstrating that these phenomena are not an artifact of our theoretical setup. Our theoretical results offer some justification for the empirical phenomenon of divergence when SS SGD is combined with BN for classification.

We briefly remark on why we analyze divergence conditions instead of directional convergence. The main difficulty lies in analyzing SGD instead of GD. One could hope to extend the techniques for directional convergence for homogeneous networks in Lyu & Li (2019) to the stochastic setting, but this is outside the scope of our paper. Furthermore, the analyses for deep linear networks such as Ji & Telgarsky (2020) rely on invariants which do not hold for us due to the diagonal $\boldsymbol{\Gamma}$ and the BN layers for deeper networks.

Throughout, we use $\boldsymbol{v} = (\boldsymbol{W}\boldsymbol{\Gamma})^\top \in \mathbb{R}^d$ to refer to the vector that determines the decision boundary of our classifier f. We remind the reader of the datasets which induce the different distorted risks (Section 3.1). Given dataset $\boldsymbol{Z} = (\boldsymbol{X}, \boldsymbol{Y})$, the GD dataset is $\overline{\boldsymbol{Z}}_{\text{GD}} \triangleq (\overline{\boldsymbol{X}}_{\text{GD}}, \boldsymbol{Y}_{\text{GD}}) = (\mathsf{BN}(\boldsymbol{X}), \boldsymbol{Y})$. Similarly define the SS dataset $\overline{\boldsymbol{Z}}_\pi \triangleq (\overline{\boldsymbol{X}}_\pi, \boldsymbol{Y}_\pi) = (\mathsf{BN}_\pi(\boldsymbol{X}), \pi \circ \boldsymbol{Y})$ and the RR dataset $\overline{\boldsymbol{Z}}_{\text{RR}} \triangleq (\overline{\boldsymbol{X}}_{\text{RR}}, \boldsymbol{Y}_{\text{RR}})$ by concatenating $\overline{\boldsymbol{Z}}_\pi$ over all permutations π. If the labels are clear from context, we occasionally abuse terminology and refer to the features as the dataset.

4.1. Analysis of problem structure for classification

To analyze the optima of the distorted risks, we introduce relevant concepts from Ji & Telgarsky (2019). Given a dataset $\boldsymbol{Z} = (\boldsymbol{X}, \boldsymbol{Y}) = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$, with labels $y_i \in \{\pm 1\}$, greedily define a *maximal linearly separable subset* $\boldsymbol{S}^{\text{LS}} \triangleq (\boldsymbol{X}^{\text{LS}}, \boldsymbol{Y}^{\text{LS}})$ as follows. Include $(\boldsymbol{x}_i, y_i)$ in $\boldsymbol{S}^{\text{LS}}$ if there exists a classifier $\boldsymbol{u}_i \in \mathbb{R}^d$ with $y_i \boldsymbol{u}_i^\top \boldsymbol{x}_i > 0$ and $y_j \boldsymbol{u}_i^\top \boldsymbol{x}_j \geq 0$ for all j. For reasons that will be clear shortly, denote the complement of $\boldsymbol{S}^{\text{LS}}$ in $\boldsymbol{Z}$ by $\boldsymbol{S}^{\text{SC}} \triangleq (\boldsymbol{X}^{\text{SC}}, \boldsymbol{Y}^{\text{SC}})$.

In particular, there exists a classifier $\boldsymbol{u}$ such that: (1) $\boldsymbol{S}^{\mathrm{LS}}$ is perfectly separated by $\boldsymbol{u}$ (2) the datapoints $\boldsymbol{X}^{\mathrm{SC}}$ in $\boldsymbol{S}^{\mathrm{SC}}$ are orthogonal to $\boldsymbol{u}$, so they are on the decision boundary. We can choose $\boldsymbol{u}$ to be the max-margin classifier $\boldsymbol{u}^{\mathrm{MM}}$ on $\boldsymbol{S}^{\mathrm{LS}}$. The notation $\boldsymbol{S}^{\mathrm{SC}}$ is chosen because the logistic risk is strongly convex when restricted to bounded subsets of $\mathrm{Span}(\boldsymbol{X}^{\mathrm{SC}})$, meaning there is a unique finite minimizer $\boldsymbol{v}^{\mathrm{SC}}$ in this subspace. Ji & Telgarsky (2019) show that linear classifiers trained on the logistic risk with GD are implicitly biased towards the ray $\boldsymbol{v}^{\mathrm{SC}} + t \cdot \boldsymbol{u}^{\mathrm{MM}}$ for $t > 0$.

We now identify a salient property of the distorted risks.

Definition 1 (Separability decomposition)**.** *The* separability decomposition *of dataset $\boldsymbol{Z}$ refers to $\boldsymbol{Z} = \boldsymbol{S}^{\mathrm{LS}} \sqcup \boldsymbol{S}^{\mathrm{SC}}$.*

If $\boldsymbol{S}^{\mathrm{LS}} = \boldsymbol{Z}$, we say $\boldsymbol{Z}$ is linearly separable (LS). If both $\boldsymbol{S}^{\mathrm{LS}}$ and $\boldsymbol{S}^{\mathrm{SC}}$ are nonempty, we say $\boldsymbol{Z}$ is *partially linearly separable* (PLS). Finally, if $\boldsymbol{S}^{\mathrm{SC}} = \boldsymbol{Z}$, we slightly abuse terminology and say $\boldsymbol{Z}$ is strongly convex (SC).[3]

Because the logistic loss does not always have finite infima, we now introduce the notion of an optimal direction.

Definition 2 (Optimal direction)**.** *Given dataset $\boldsymbol{Z} = (\boldsymbol{X}, \boldsymbol{Y})$, we say a sequence of iterates $\boldsymbol{v}(t)$* infimizes *$\mathcal{L}$ if $\mathcal{L}(\boldsymbol{v}(t)^\top \boldsymbol{X}, \boldsymbol{Y}) \to \inf_{\boldsymbol{w} \in \mathbb{R}^d} \mathcal{L}(\boldsymbol{w}^\top \boldsymbol{X}, \boldsymbol{Y})$. We call $\boldsymbol{v} \in \mathbb{R}^d$ an* optimal direction *if there exists $\boldsymbol{u} \in \mathbb{R}^d$ such that $\{\boldsymbol{u} + t\boldsymbol{v}\}_{t \geq 1}$ infimizes $\mathcal{L}$.*[4]

Definition 1 is motivated by the following results which identify how the separability decomposition affects optimal directions. Their proofs are deferred to Appendix B.4.

Lemma 4.1.1. *Let $\boldsymbol{Z} = \boldsymbol{S}^{\mathrm{LS}} \sqcup \boldsymbol{S}^{\mathrm{SC}}$. If $\boldsymbol{v}$ is an optimal direction for $\mathcal{L}$, then $\boldsymbol{v}^\top \boldsymbol{x} = 0$ for all $\boldsymbol{x} \in \mathrm{Span}(\boldsymbol{X}^{\mathrm{SC}})$ and $y_i \boldsymbol{v}^\top \boldsymbol{x}_i > 0$ for every $(\boldsymbol{x}_i, y_i) \in \boldsymbol{S}^{\mathrm{LS}}$.*

Combining the above lemma and the definitions yields the following proposition, which characterizes SS and RR divergence using the separability decomposition.

Proposition 4.1.2. *Suppose Assumption 1(a) holds, the iterates $\boldsymbol{v}_\pi(t)$ infimize $\mathcal{L}_\pi$, and their projections onto $\mathrm{Span}(\overline{\boldsymbol{X}}_\pi^{\mathrm{SC}})^\perp$ converge in direction to some optimal direction $\boldsymbol{v}_\pi^*$ for $\mathcal{L}_\pi$. Then the GD risk $\mathcal{L}_{\mathrm{GD}}$ diverges if and only if $\overline{\boldsymbol{Z}}_\pi$ is PLS or LS and there exists some $(\boldsymbol{x}_i, y_i) \in \overline{\boldsymbol{Z}}_{\mathrm{GD}}$ such that $y_i \boldsymbol{v}_\pi^{*\top} \boldsymbol{x}_i < 0$. The analogous statement holds true for $\overline{\boldsymbol{Z}}_{\mathrm{RR}}$ under Assumption 1(b). Furthermore, the "if" part holds true for SS and RR without Assumption 1.*

In particular, Proposition 4.1.2 implies that if the RR dataset is SC and rank d, the GD risk does not diverge. Moreover, it naturally leads to the question of understanding ranks and separability decompositions of the SS and RR datasets; the former question is already answered by Proposition 3.2.1.

To analyze the separability decomposition with high probability or almost surely, we assume the labels are balanced.

Assumption 4 (Balanced classes)**.** *The data $\boldsymbol{Z}$ either has*

(a) *an equal number of positive and negative examples; or*

(b) *at least B positive and B negative examples.*

Finally, we informally state our main classification result: SS+BN can diverge in some regimes (see Theorem B.2.1 for details).

Theorem 4.1.3 (SS+BN can diverge (informal))**.** *Assume Assumption 2, Assumption 4(a), and $B > 2$. If $d \leq (B-1)\frac{n}{B}$, SS can diverge if $B = \Omega(\log n)$ and $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$'s separability decomposition can change with small perturbations. Otherwise, SS can diverge regardless of the batch size and the separability decomposition of $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$.*

Whereas Theorem 4.1.3 establishes regimes where SS+BN can diverge, we can show that RR+BN prevents divergence in a much larger regime (see Theorem B.3.1 for details).

Theorem 4.1.4 (RR+BN does not diverge (informal))**.** *Assume Assumption 2, Assumption 4(b), and $B > 2$. If $d \leq (B-1)\binom{n}{B}$, RR does not diverge almost surely.*

Theorem 4.1.3 implies that one cannot prevent SS divergence by simply increasing the batch size B; it is also necessary for the GD dataset to be "robustly" LS or SC. Moreover, as soon as $d > (B-1)\frac{n}{B}$, SS can diverge. In stark contrast, Theorem 4.1.4 establishes that even for small B, RR is *almost surely* robust to divergence as long as $d \leq (B-1)\binom{n}{B}$. Although our theorems do not prove that SS+BN *necessarily* diverges, they offer some theoretical explanation for why SS+BN appears to be less stable than RR+BN for classification.

4.2. RR prevents divergence while SS diverges

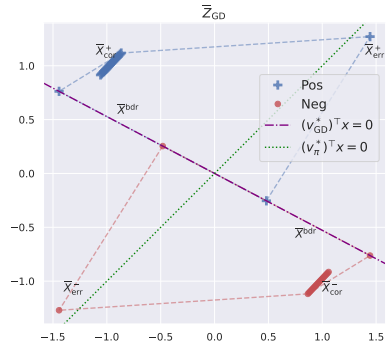
We present a toy dataset where SS drastically distorts the optimal direction, leading to divergence with constant probability. Meanwhile, RR does not diverge on this dataset. We use $d = B = 2$ to simplify the construction.[5]

Dataset: SS often diverges; RR does not. We describe our construction (Figure 3a) at a high level; see Proposition E.2.1 for details. The GD dataset is PLS with unique optimal direction $\boldsymbol{v}_{\mathrm{GD}}^*$ (its decision boundary is the purple dash-dotted line). Moreover, with constant probability the SS dataset is PLS with unique optimal direction $\boldsymbol{v}_\pi^*$ (green dotted line) distorted away from $\boldsymbol{v}_{\mathrm{GD}}^*$. Also, $\boldsymbol{v}_\pi^*$ misclassifies points in the GD dataset ($\overline{\boldsymbol{X}}_{\mathrm{err}}^+$ and $\overline{\boldsymbol{X}}_{\mathrm{err}}^-$). Under the

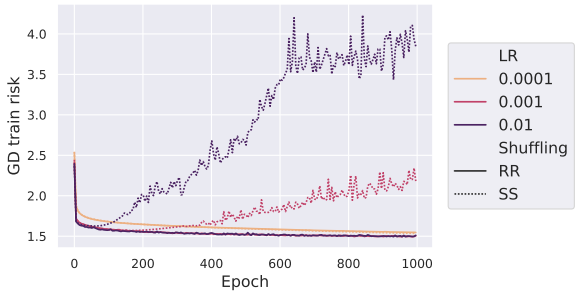
[3] Here, PLS refers to the "general case" discussed in Ji & Telgarsky (2019), but we chose to use this alternative terminology because we found the term "general" can lead to confusion.

[4] This definition is catered towards the SC+full rank $\boldsymbol{X}$ or PLS/LS case. However, since Proposition 3.2.1 provides sufficient conditions for full-rank data, this subtlety is unimportant.

[5] Since $B = 2$, there is no contradiction with Theorem 4.1.3.



(a) Toy classification dataset showing divergence of SS with constant probability.



(b) 3 layer linear+BN networks trained with varying stepsizes.

Figure 3. Top: Toy dataset demonstrating divergence of GD risk with constant probability. The dashed lines trace out the convex hulls of the positive and negative points. Bottom: divergence of GD risk for a variety of stepsizes on CIFAR10. Note that there was eventually a separation for $\eta = 10^{-4}$ (see Figure 8).

additional assumptions in Proposition 4.1.2, the GD risk diverges. Finally, since the RR dataset is SC and rank d, RR does not diverge on the GD dataset.

4.3. Experiments on linear and nonlinear networks

We now verify our theoretical classification results on linear+BN and extend them to nonlinear networks on a variety of real-world datasets. This demonstrates that the separation between SS, RR, and GD is relevant in realistic settings and not merely an artifact of the linear setting. We refer to the linear+BN network $\boldsymbol{W}\boldsymbol{\Gamma}\mathsf{BN}(\boldsymbol{X})$ as 1-layer linear network, and also consider deeper linear networks with tunable parameters inside BN layers. We observe strikingly different training behaviors in the shallow and deep linear networks. The networks are formally defined in Appendix D; see `https://github.com/davidxwu/sgd-batchnorm-icml` for the experiment code.

As a motivating example, we ran an experiment on synthetic data (Figure 4) with the 2-layer linear network $f(\boldsymbol{X}) = \boldsymbol{W}\boldsymbol{\Gamma}\mathsf{BN}(\boldsymbol{A}\boldsymbol{X})$. Note that the tunable matrix $\boldsymbol{A}$ acts before BN. Intriguingly, we observe that the SS dataset with features $\overline{\boldsymbol{X}}_\pi = \mathsf{BN}_\pi(\boldsymbol{A}\boldsymbol{X})$ is SC *at initialization*, but updating $\boldsymbol{A}$ with SS makes it LS *after training*. Moreover, the batch size is large relative to n, so this dataset satisfies the necessary conditions for divergence in Proposition 4.1.2 and Theorem 4.1.3.

More specifically, Figures 4a and 4c plot the 2-dimensional GD and SS datasets, respectively, which are SC at initialization. However, after training with SS, we can see from Figures 4b and 4d that SS updates $\boldsymbol{A}$ to make the SS dataset LS, whereas the GD dataset stays SC. Hence, by Proposition 4.1.2, the GD risk diverges. This example partially explains the discrepancy in training behavior between the 1-layer and deeper networks. Indeed, whereas the 1-layer architecture has static $\overline{\boldsymbol{Z}}_\pi$, the deeper networks have evolving weights inside BN which can push $\overline{\boldsymbol{Z}}_\pi$ to be LS/PLS.

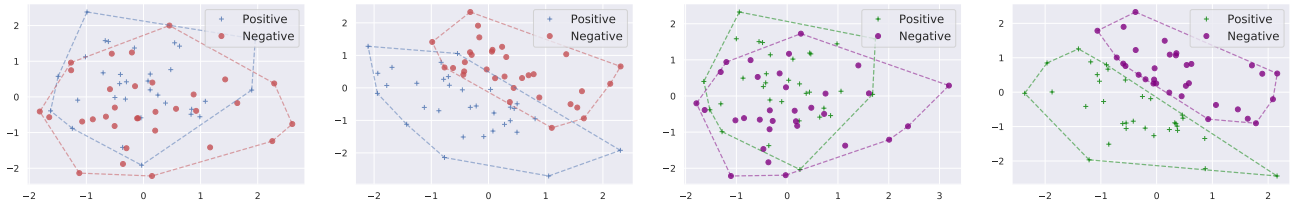
To exhibit the above divergence on real data, we conducted experiments on the CIFAR10. Using SS and RR, we trained linear+BN networks of depths up to 3 for $T = 10^3$ epochs using stepsize $\eta = 10^{-2}$, batch size $B = 128$, and 512 hidden units per layer (see Appendix D for precise details).

As depicted in Figure 1a, we consistently observed SS divergence for the deeper networks (see Figure 9 for more evidence of divergence). As predicted by Theorem 4.1.4, RR did not exhibit divergence behavior. These phenomena persisted despite ablating the learning rate in $\{0.01, 0.001, 0.0001\}$, momentum in $\{0, 0.9, 0.99\}$, and batch size in $\{32, 64, 128\}$. The learning rate ablation is shown in Figure 3b; see Appendix D for the rest.

For the nonlinear experiments, we extended to the CIFAR10, MNIST, and CIFAR100 datasets. We used SS and RR to train 3-layer 512 hidden unit MLPs with BN and ReLU activation for $T = 10^3$ epochs, and also to finetune pretrained ResNet18 for $T = 50$ epochs. We consistently observed that in the final stages of training (i.e., relatively small training risk), SS trained slower than RR across all of the datasets, even after tuning the learning rate (see Figures 5 and 6).

5. Conclusion

This paper established that training BN networks with SS can lead to undesirable training behavior, including slower convergence or even divergence of the GD risk. However, RR provably mitigates this divergence behavior, and experimental evidence suggests that using RR usually converges faster than SS. This separation in training behavior between SS, RR, and GD is because data shuffling directly affects how BN operates on mini-batches. Our theoretical results establish a separation for the special case where BN is applied to the input features. The more general and realistic case where BN is applied to dynamically evolving layers is left as an important direction for future work. We also observed in preliminary experiments that a similar separation manifested for generalization, and we hope that adopting



(a) $\overline{\boldsymbol{Z}}_{\mathrm{GD}} = (\mathsf{BN}(\boldsymbol{A}_0^0\boldsymbol{X}), \boldsymbol{Y})$ at initialization. (b) $\overline{\boldsymbol{Z}}_{\mathrm{GD}} = (\mathsf{BN}(\boldsymbol{A}_0^T\boldsymbol{X}), \boldsymbol{Y})$ after training. (c) $\overline{\boldsymbol{Z}}_{\pi} = (\mathsf{BN}_{\pi}(\boldsymbol{A}_0^0\boldsymbol{X}), \boldsymbol{Y}_{\pi})$ at initialization. (d) $\overline{\boldsymbol{Z}}_{\pi} = (\mathsf{BN}_{\pi}(\boldsymbol{A}_0^T\boldsymbol{X}), \boldsymbol{Y}_{\pi})$ after training.

Figure 4. Snapshots of GD dataset $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ and SS dataset $\overline{\boldsymbol{Z}}_{\pi}$ before and after running SS for $T = 10^4$ epochs with 32 positive and negative synthetic examples. While the GD dataset remains SC, the SS dataset become LS. Here $B = 16$, $\eta = 10^{-2}$, and $\epsilon = 10^{-5}$ for BN.

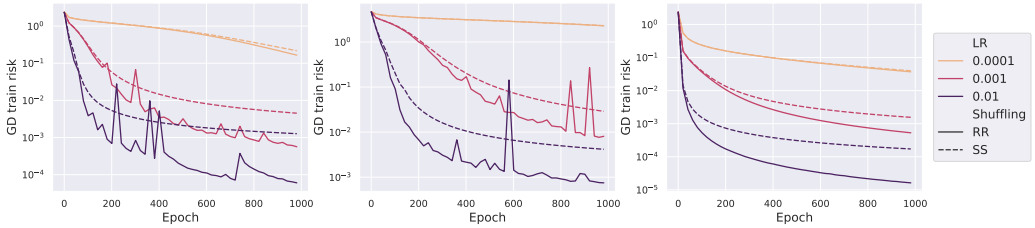


Figure 5. 3 layer ReLU+BN MLP on (left to right): CIFAR10, CIFAR100, and MNIST. Note the slower convergence for SS versus RR in the final stages of training for CIFAR10 and MNIST, and the early stages of training for CIFAR100.

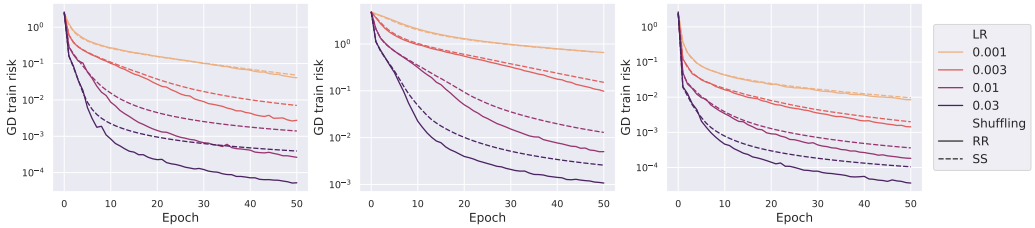


Figure 6. ResNet18 finetuned on (left to right): CIFAR10, CIFAR100, and MNIST. Note the slower convergence for SS versus RR across datasets in the final stages of training for CIFAR10 and MNIST, and the early stages of training for CIFAR100. For the smallest learning rate $\eta = 10^{-3}$, we observed a separation after 200 epochs.

a similar perspective will prove fruitful in pursuing this direction. We remark that similar surprising phenomena may arise when using other design choices that are implemented in a mini-batch fashion such as mixup (Zhang et al., 2017) and Sharpness-Aware Minimization (SAM) (Foret et al., 2020). For these reasons, we generally recommend that practitioners use RR instead of SS. Further future directions include establishing directional convergence for homogeneous classifiers trained with shuffling SGD and theoretically understanding conditions under which deeper networks diverge faster.

Acknowledgements

Part of the work was done while DW was an undergraduate at MIT. DW acknowledges support from NSF Graduate Research Fellowship DGE-2146752. CY acknowledges support by Institute of Information & communications Technology Planning & evaluation (IITP) grant (No. 2019-0-00075, Artificial Intelligence Graduate School Program (KAIST)) funded by the Korea government (MSIT). CY is also supported by the National Research Foundation of Korea (NRF) grants (No. NRF-2019R1A5A1028324, RS-2023-00211352) funded by the Korea government (MSIT). CY acknowledges support from a grant funded by Samsung Electronics Co., Ltd. SS acknowledge support from an NSF CAREER grant (1846088), and NSF CCF-2112665 (TILOS AI Research Institute). DW appreciates helpful discussions with Xiang Cheng, Sidhanth Mohanty, Erik Jenner, Louis Golowich, Sam Gunn, and Thiago Bergamaschi.

References

Agarwal, P. K., Guibas, L. J., Har-Peled, S., Rabinovitch, A., and Sharir, M. Penetration depth of two convex polytopes in 3d. *Nord. J. Comput.*, 7(3):227–240, 2000.

Ahn, K., Yun, C., and Sra, S. Sgd with shuffling: optimal rates without component convexity and large epoch requirements. *arXiv preprint arXiv:2006.06946*, 2020.

Arora, S., Li, Z., and Lyu, K. Theoretical analysis of auto rate-tuning by batch normalization. *arXiv preprint arXiv:1812.03981*, 2018.

Ba, J. L., Kiros, J. R., and Hinton, G. E. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Bardenet, R. and Maillard, O.-A. Concentration inequalities for sampling without replacement. *Bernoulli*, 21:1361–1385, 2015.

Bjorck, N., Gomes, C. P., Selman, B., and Weinberger, K. Q. Understanding batch normalization. *Advances in neural information processing systems*, 31, 2018.

Boyd, S., Boyd, S. P., and Vandenberghe, L. *Convex optimization*. Cambridge university press, 2004.

Cai, Y., Li, Q., and Shen, Z. A quantitative analysis of the effect of batch normalization on gradient descent. In *International Conference on Machine Learning*, pp. 882–890. PMLR, 2019.

Cha, J., Lee, J., and Yun, C. Tighter lower bounds for shuffling SGD: Random permutations and beyond. *arXiv preprint arXiv:2303.07160*, 2023.

Cho, H. and Yun, C. SGDA with shuffling: faster convergence for nonconvex-PŁ minimax optimization. In *International Conference on Learning Representations*, 2023.

Daneshmand, H., Kohler, J., Bach, F., Hofmann, T., and Lucchi, A. Batch normalization provably avoids ranks collapse for randomly initialised deep networks. *Advances in Neural Information Processing Systems*, 33: 18387–18398, 2020.

Daneshmand, H., Joudaki, A., and Bach, F. Batch normalization orthogonalizes representations in deep random networks. *Advances in Neural Information Processing Systems*, 34:4896–4906, 2021.

Duchi, J., Hazan, E., and Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

Foret, P., Kleiner, A., Mobahi, H., and Neyshabur, B. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020.

Gunasekar, S., Lee, J., Soudry, D., and Srebro, N. Characterizing implicit bias in terms of optimization geometry. In *International Conference on Machine Learning*, pp. 1832–1841. PMLR, 2018.

Gürbüzbalaban, M., Ozdaglar, A., Vanli, N. D., and Wright, S. J. Randomness and permutations in coordinate descent methods. *Mathematical Programming*, 181:349–376, 2020.

Haochen, J. and Sra, S. Random shuffling beats sgd after finite epochs. In *International Conference on Machine Learning*, pp. 2624–2633. PMLR, 2019.

Hoffer, E., Hubara, I., and Soudry, D. Train longer, generalize better: closing the generalization gap in large batch training of neural networks. *Advances in Neural Information Processing Systems*, 30, 2017.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pp. 448–456. PMLR, 2015.

Jagadeesan, M., Razenshteyn, I., and Gunasekar, S. Inductive bias of multi-channel linear convolutional networks with bounded weight norm. In *Conference on Learning Theory*, pp. 2276–2325. PMLR, 2022.

Ji, Z. and Telgarsky, M. Gradient descent aligns the layers of deep linear networks. *arXiv preprint arXiv:1810.02032*, 2018.

Ji, Z. and Telgarsky, M. The implicit bias of gradient descent on nonseparable data. In *Conference on Learning Theory*, pp. 1772–1798. PMLR, 2019.

Ji, Z. and Telgarsky, M. Directional convergence and alignment in deep learning. *Advances in Neural Information Processing Systems*, 33:17176–17186, 2020.

Johnson, C. R. *Matrix theory and applications*, volume 40. American Mathematical Soc., 1990.

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

Kohler, J., Daneshmand, H., Lucchi, A., Zhou, M., Neymeyr, K., and Hofmann, T. Towards a theoretical understanding of batch normalization. *stat*, 1050:27, 2018.

Kohler, J., Daneshmand, H., Lucchi, A., Hofmann, T., Zhou, M., and Neymeyr, K. Exponential convergence rates for batch normalization: The power of length-direction decoupling in non-convex optimization. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 806–815. PMLR, 2019.

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

Li, Z., Lyu, K., and Arora, S. Reconciling modern deep learning with traditional optimization analyses: The intrinsic learning rate. *Advances in Neural Information Processing Systems*, 33:14544–14555, 2020.

Lobacheva, E., Kodryan, M., Chirkova, N., Malinin, A., and Vetrov, D. P. On the periodic behavior of neural network training with batch normalization and weight decay. *Advances in Neural Information Processing Systems*, 34: 21545–21556, 2021.

Lyu, K. and Li, J. Gradient descent maximizes the margin of homogeneous neural networks. *arXiv preprint arXiv:1906.05890*, 2019.

Lyu, K., Li, Z., and Arora, S. Understanding the generalization benefit of normalization layers: Sharpness reduction. *Advances in Neural Information Processing Systems*, 36, 2022.

Maurer, A. Concentration inequalities for functions of independent variables. *Random Structures & Algorithms*, 29 (2):121–138, 2006.

Mityagin, B. The zero set of a real analytic function. *arXiv preprint arXiv:1512.07276*, 2015.

Nacson, M. S., Lee, J., Gunasekar, S., Savarese, P. H. P., Srebro, N., and Soudry, D. Convergence of gradient descent on separable data. In Chaudhuri, K. and Sugiyama, M. (eds.), *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 3420–3428. PMLR, 16–18 Apr 2019a. URL https://proceedings.mlr.press/v89/nacson19b.html.

Nacson, M. S., Srebro, N., and Soudry, D. Stochastic gradient descent on separable data: Exact convergence with a fixed learning rate. In Chaudhuri, K. and Sugiyama, M. (eds.), *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 3051–3059. PMLR, 16–18 Apr 2019b. URL https://proceedings.mlr.press/v89/nacson19a.html.

Nagaraj, D., Jain, P., and Netrapalli, P. Sgd without replacement: Sharper rates for general smooth convex functions. In *International Conference on Machine Learning*, pp. 4703–4711. PMLR, 2019.

Nguyen, L. M., Tran-Dinh, Q., Phan, D. T., Nguyen, P. H., and van Dijk, M. A unified convergence analysis for shuffling-type gradient methods. *Journal of Machine Learning Research*, 22(207):1–44, 2021.

Rajput, S., Gupta, A., and Papailiopoulos, D. Closing the convergence gap of sgd without replacement. In *International Conference on Machine Learning*, pp. 7964–7973. PMLR, 2020.

Safran, I. and Shamir, O. How good is sgd with random shuffling? In *Conference on Learning Theory*, pp. 3250–3284. PMLR, 2020.

Santurkar, S., Tsipras, D., Ilyas, A., and Madry, A. How does batch normalization help optimization? *Advances in neural information processing systems*, 31, 2018.

Shallue, C. J., Lee, J., Antognini, J., Sohl-Dickstein, J., Frostig, R., and Dahl, G. E. Measuring the effects of data parallelism on neural network training. *Journal of Machine Learning Research*, 20(112):1–49, 2019.

Soudry, D., Hoffer, E., Nacson, M. S., Gunasekar, S., and Srebro, N. The implicit bias of gradient descent on separable data. *The Journal of Machine Learning Research*, 19(1):2822–2878, 2018.

Summers, C. and Dinneen, M. J. Four things everyone should know to improve batch normalization. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=HJx8HANFDH.

Sun, R., Luo, Z.-Q., and Ye, Y. On the efficiency of random permutation for admm and coordinate descent. *Mathematics of Operations Research*, 45(1):233–271, 2020.

Thomas, M. and Joy, A. T. *Elements of information theory*. Wiley-Interscience, 2006.

Ulyanov, D., Vedaldi, A., and Lempitsky, V. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

Wan, R., Zhu, Z., Zhang, X., and Sun, J. Spherical motion dynamics: Learning dynamics of normalized neural network using SGD and weight decay. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=RcbphT7qjTq.

Woodworth, B., Gunasekar, S., Lee, J. D., Moroshko, E., Savarese, P., Golan, I., Soudry, D., and Srebro, N. Kernel and rich regimes in overparametrized models. In *Conference on Learning Theory*, pp. 3635–3673. PMLR, 2020.

Wu, L., Wang, Q., and Ma, C. Global convergence of gradient descent for deep linear residual networks. *arXiv preprint arXiv:1911.00645*, 2019.

Wu, Y. and Johnson, J. Rethinking" batch" in batchnorm. *arXiv preprint arXiv:2105.07576*, 2021.

Yong, H., Huang, J., Meng, D., Hua, X., and Zhang, L. Momentum batch normalization for deep learning with small batch size. In *European Conference on Computer Vision*, pp. 224–240. Springer, 2020.

Yun, C., Krishnan, S., and Mobahi, H. A unifying view on implicit bias in training linear neural networks. In *International Conference on Learning Representations*, 2021a.

Yun, C., Sra, S., and Jadbabaie, A. Open problem: Can single-shuffle SGD be better than reshuffling SGD and GD? In *Conference on Learning Theory*, pp. 4653–4658. PMLR, 2021b.

Yun, C., Rajput, S., and Sra, S. Minibatch vs local sgd with shuffling: Tight convergence bounds and beyond. In *International Conference on Learning Representations*, 2022.

Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Zhou, Y. and Liang, Y. Characterization of gradient dominance and regularity conditions for neural networks. *arXiv preprint arXiv:1710.06910*, 2017.

A. Proofs for regression results

In this appendix, we provide the full details for the proof of convergence for SS and RR in the regression case.

Additional notation. We introduce some additional notation which we will use throughout the proof of Theorems 3.2.2 and 3.2.3. For a matrix $\boldsymbol{A}$, we use $\boldsymbol{A}_{i,:}$ and $\boldsymbol{A}_{:,j}$ to denote the ith row and jth column of $\boldsymbol{A}$, respectively. We also use $A_{i,j}$ to denote the (i,j)th entry of $\boldsymbol{A}$. The Hadamard product of two matrices $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{m\times n}$ is denoted by $\boldsymbol{A} \odot \boldsymbol{B}$, with $(\boldsymbol{A} \odot \boldsymbol{B})_{i,j} = A_{i,j}B_{i,j}$. The diagonal operator $\mathrm{diag} : \mathbb{R}^{m\times m} \to \mathbb{R}^{m\times m}$ is defined by $\mathrm{diag}(\boldsymbol{A}) = \boldsymbol{I} \odot \boldsymbol{A}$. We denote the Frobenius inner product $\langle \boldsymbol{A}, \boldsymbol{B}\rangle_F = \sum_{i,j} A_{i,j}B_{i,j}$ and its induced norm by $\|\boldsymbol{A}\|_F$.

Also recall from Section 2 that when Θ is optimized with SS or RR, the ith iterate on the kth epoch is denoted by Θ_i^k. For simplicity, we will often say the (i,k)th iterate to refer to Θ_i^k. Denote the collapsed parameter matrix defined in Section 3 by $\boldsymbol{M} \triangleq \boldsymbol{W}\boldsymbol{\Gamma}$. We will abuse notation and sometimes denote the (i,k)th iterate by $\boldsymbol{M}_i^k \triangleq \boldsymbol{W}_i^k\boldsymbol{\Gamma}_i^k$.

Recall that the mini-batch risk used for updating the (i,k)th iterate of SS or RR is given by $\mathcal{L}(f(\boldsymbol{X}_\pi^{i+1}; \Theta_i^k), \boldsymbol{Y}_\pi^{i+1}) = \left\|\boldsymbol{Y}_\pi^{i+1} - \boldsymbol{W}_i^k\boldsymbol{\Gamma}_i^k\mathsf{BN}(\boldsymbol{X}_\pi^{i+1})\right\|_F^2$ where π denotes the permutation chosen for the kth epoch and $\boldsymbol{X}_\pi^j \in \mathbb{R}^{d\times B}$ and $\boldsymbol{Y}_\pi^j \in \mathbb{R}^{p\times B}$ consist of the $(jB - B + 1, \dots, jB)$th columns of $\pi \circ \boldsymbol{X}$ and $\pi \circ \boldsymbol{Y}$, respectively. Since this notation is a bit lengthy, we simplify it to $\mathcal{L}(\boldsymbol{X}_\pi^j; \Theta) \triangleq \mathcal{L}(f(\boldsymbol{X}_\pi^j; \Theta), \boldsymbol{Y}_\pi^j)$ for any $j \in [m]$. Here, we can also view the mini-batch risk as a function of $\boldsymbol{M} = \boldsymbol{W}\boldsymbol{\Gamma}$, so we will sometimes abuse notation and write

$$\mathcal{L}(\boldsymbol{X}_\pi^j; \boldsymbol{M}) \triangleq \left\|\boldsymbol{Y}_\pi^j - \boldsymbol{M}\mathsf{BN}(\boldsymbol{X}_\pi^j)\right\|_F^2,$$
$$\boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}(\boldsymbol{X}_\pi^j; \boldsymbol{M}) \triangleq -(\boldsymbol{Y}_\pi^j - \boldsymbol{M}\mathsf{BN}(\boldsymbol{X}_\pi^j))\mathsf{BN}(\boldsymbol{X}_\pi^j)^\top.$$

For SS, we work with a fixed permutation $\pi \in \mathbb{S}_n$ and input dataset $(\boldsymbol{X}, \boldsymbol{Y})$. Recall that we defined $\overline{\boldsymbol{X}}_\pi \triangleq \mathsf{BN}_\pi(\boldsymbol{X})$ from Section 3, i.e., the column-wise concatenation of all batches after batch normalization: $\overline{\boldsymbol{X}}_\pi = [\mathsf{BN}(\boldsymbol{X}_\pi^1) \ \cdots \ \mathsf{BN}(\boldsymbol{X}_\pi^m)]$. When the context of parameters $\Theta = (\boldsymbol{W}, \boldsymbol{\Gamma})$ and permutation $\pi \in \mathbb{S}_n$ chosen by SS are clear, we denote the collection of outputs over the dataset by $\hat{\boldsymbol{Y}}_\pi \triangleq \boldsymbol{W}\boldsymbol{\Gamma}\overline{\boldsymbol{X}}_\pi$. Also recall that the distorted SS risk $\mathcal{L}_\pi(\Theta)$ we set out to optimize is defined to be $\mathcal{L}_\pi(\Theta) = \mathcal{L}_\pi(\boldsymbol{W}, \boldsymbol{\Gamma}) = \left\|\boldsymbol{Y}_\pi - \boldsymbol{W}\boldsymbol{\Gamma}\overline{\boldsymbol{X}}_\pi\right\|_F^2$. With $\boldsymbol{M} \triangleq \boldsymbol{W}\boldsymbol{\Gamma}$, we also abuse notation and write

$$\mathcal{L}_\pi(\boldsymbol{M}) \triangleq \left\|\boldsymbol{Y}_\pi - \boldsymbol{M}\overline{\boldsymbol{X}}_\pi\right\|_F^2,$$
$$\boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}) \triangleq -(\boldsymbol{Y}_\pi - \boldsymbol{M}\overline{\boldsymbol{X}}_\pi)\overline{\boldsymbol{X}}_\pi^\top.$$

We will use big–O notation throughout to simplify the presentation of the proofs. When we write $O(\eta_k^t)$ for some exponent $t \geq 1$, we hide constants that depend on m, $\left\|\overline{\boldsymbol{X}}_\pi\right\|_F$, and various absolute constants defined explicitly below. These constants have at most polynomial dependence on these parameters and absolute constants.

A.1. Proof of convergence for SS

Let us first prove Theorem 3.2.2. First, we draw the reader's attention to some standard properties in optimization theory that allow us to prove global convergence. We then sketch out the proof in Appendix A.1.2 and flesh out the details in subsequent sections.

A.1.1. Optimization properties

It is profitable to keep in mind the general idea behind proving global convergence of SGD for a function $\mathcal{L}(\Theta)$, which has been exploited in Ahn, Yun, and Sra (2020); Zhou and Liang (2017); Nguyen, Tran-Dinh, Phan, Nguyen, and van Dijk (2021). The following two properties of the optimization problem are critical in such approaches:

Property 1 (Smoothness). *G-smoothness of $\mathcal{L}$, i.e., the gradients of $\mathcal{L}$ are G-Lipschitz. In particular, it implies the following two standard properties:*

(i) $\mathcal{L}(\Theta) \leq \mathcal{L}(\Theta') + \langle \boldsymbol{\nabla}_\Theta\mathcal{L}(\Theta'), \Theta - \Theta'\rangle + \frac{G}{2}\|\Theta' - \Theta\|^2$ *for all* Θ, Θ' *in the domain of* $\mathcal{L}$.

(ii) The Hessian $\boldsymbol{H} = \boldsymbol{\nabla}_\Theta^2\mathcal{L}(\Theta)$ *satisfies* $\|\boldsymbol{H}\|_2 \leq G$ *for all* Θ *in the domain of* $\mathcal{L}$.

Property 2 (PŁ condition)**.** *The loss function $\mathcal{L}$ satisfies the α-Polyak-Łojasiewicz condition, i.e., $\|\nabla\mathcal{L}(\Theta)\|^2 \geq 2\alpha(\mathcal{L}(\Theta) - \mathcal{L}^*)$ for all Θ in the domain of $\mathcal{L}$.*

In our case, we can use global smoothness and strong convexity (which implies the PŁ condition) of $\mathcal{L}_\pi$ with respect to $\boldsymbol{M} = \boldsymbol{W}\boldsymbol{\Gamma}$, but these global properties do not hold with respec to our optimization variables $\Theta = (\boldsymbol{W}, \boldsymbol{\Gamma})$. Importantly, unlike the analyses of Ahn, Yun, and Sra (2020); Nguyen, Tran-Dinh, Phan, Nguyen, and van Dijk (2021), we cannot directly leverage the global smoothness and strong convexity as is, because we do not directly perform gradient updates on $\boldsymbol{M}$. Instead, we effectively use a "dynamic" PŁ condition which depends on $\boldsymbol{\Gamma}$. The subtlety in the analysis is to show that such behavior can be controlled to ensure convergence in the end.

Finally, a third property — which is often exploited to prove convergence results for linear neural networks — is the notion of an (approximate) invariance property satisfied by the layers of the neural network. Indeed, in the continuous time case, i.e., when we minimize $\mathcal{L}_\pi(\Theta(t))$ with gradient flow $\dot{\Theta}(t) = -\nabla_\Theta\mathcal{L}_\pi(\Theta(t))$, such an invariance can be directly shown by the differential equations, see Wu, Wang, and Ma (2019) for instance. To that end, define the following quantity

$$\boldsymbol{D} \triangleq \boldsymbol{I} + \operatorname{diag}(\boldsymbol{W}^\top\boldsymbol{W} - \boldsymbol{\Gamma}^2), \tag{1}$$

which we refer to as the invariance matrix. For each iterate Θ_i^k of SS, the corresponding $\boldsymbol{D}_i^k$ can also be naturally defined. In gradient flow, $\boldsymbol{D}(t)$ actually remains invariant with time $t \in [0, \infty)$. We quickly prove this property here, and later prove that an approximate version holds in the discrete and stochastic case, although the bounds are messier.

Fact A.1.1. *In the gradient flow formulation, we have $\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{D}(t) = \boldsymbol{0}$. Moreover, in both the gradient flow and discrete time formulation, we have*

$$\operatorname{diag}(\boldsymbol{W}^\top\nabla_{\boldsymbol{W}}\mathcal{L}_\pi) = (\nabla_{\boldsymbol{\Gamma}}\mathcal{L}_\pi)\boldsymbol{\Gamma}. \tag{2}$$

Proof. For the proof, we write out the (full) gradients of $\mathcal{L}_\pi$ with respect to $\boldsymbol{W}$ and $\boldsymbol{\Gamma}$ for reference:

$$\nabla_{\boldsymbol{W}}\mathcal{L}_\pi = -(\boldsymbol{Y}_\pi - \hat{\boldsymbol{Y}}_\pi)\overline{\boldsymbol{X}}_\pi^\top\boldsymbol{\Gamma}, \tag{3}$$

$$\nabla_{\boldsymbol{\Gamma}}\mathcal{L}_\pi = -\operatorname{diag}(\boldsymbol{W}^\top(\boldsymbol{Y}_\pi - \hat{\boldsymbol{Y}}_\pi)\overline{\boldsymbol{X}}_\pi^\top). \tag{4}$$

A direct calculation shows that $\operatorname{diag}(\boldsymbol{W}^\top\nabla_{\boldsymbol{W}}\mathcal{L}_\pi) = (\nabla_{\boldsymbol{\Gamma}}\mathcal{L}_\pi)\boldsymbol{\Gamma}$. Due to the gradient flow formulation $\dot{\Theta}(t) = -\nabla_\Theta\mathcal{L}_\pi(\Theta(t))$ we have $\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{W}(t) = -\nabla_{\boldsymbol{W}}\mathcal{L}_\pi$ and $\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\Gamma}(t) = -\nabla_{\boldsymbol{\Gamma}}\mathcal{L}_\pi$, so it follows from Equation (2) that $\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{D}(t) = 0$. □

We now formally state the smoothness and PŁ guarantees for our setup.

Lemma A.1.2 (Smoothness with respect to $\boldsymbol{M}$)**.** *The SS risk $\mathcal{L}_\pi$ is G_π–smooth with respect to $\boldsymbol{M} = \boldsymbol{W}\boldsymbol{\Gamma}$, where*

$$G_\pi \triangleq \left\|\overline{\boldsymbol{X}}_\pi\right\|_2^2.$$

Proof. We directly check the Lipschitz gradient condition. Indeed, we have

$$\begin{aligned}
&\|\nabla_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}) - \nabla_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}')\|_2 \\
&= \left\|(\boldsymbol{Y}_\pi - \boldsymbol{M}\overline{\boldsymbol{X}}_\pi)\overline{\boldsymbol{X}}_\pi^\top - (\boldsymbol{Y}_\pi - \boldsymbol{M}'\overline{\boldsymbol{X}}_\pi)\overline{\boldsymbol{X}}_\pi^\top\right\|_2 \\
&= \left\|(\boldsymbol{M} - \boldsymbol{M}')\overline{\boldsymbol{X}}_\pi\overline{\boldsymbol{X}}_\pi^\top\right\|_2 \leq \left\|\overline{\boldsymbol{X}}_\pi\right\|_2^2\|\boldsymbol{M} - \boldsymbol{M}'\|_2,
\end{aligned}$$

Note that the same inequality holds (with the same value of G_π) if we instead used the Frobenius norm, due to the fact that $\|\boldsymbol{A}\boldsymbol{B}\|_F \leq \|\boldsymbol{B}\|_2\|\boldsymbol{A}\|_F$ in the last line. □

Lemma A.1.3 (Strong convexity with respect to $\boldsymbol{M}$)**.** *Under Assumption 1(a), SS risk $\mathcal{L}_\pi$ is α_π–strongly convex with respect to $\boldsymbol{M} = \boldsymbol{W}\boldsymbol{\Gamma}$, where*

$$\alpha_\pi \triangleq \sigma_{\min}(\overline{\boldsymbol{X}}_\pi\overline{\boldsymbol{X}}_\pi^\top).$$

Hence, $\mathcal{L}_\pi$ is also α_π–PŁ with respect to $\boldsymbol{M}$.

Proof. Take the Hessian of $\mathcal{L}_\pi(\boldsymbol{M})$ with respect to the vectorized version $\operatorname{vec}(\boldsymbol{M})$ of $\boldsymbol{M}$ to obtain $\nabla^2_{\operatorname{vec}(\boldsymbol{M})}\mathcal{L}_\pi(\boldsymbol{M}) = \overline{\boldsymbol{X}}_\pi\overline{\boldsymbol{X}}_\pi^\top \otimes \boldsymbol{I}_p$, where $\otimes$ denotes the Kronecker product. Then evidently $\nabla^2_{\operatorname{vec}(\boldsymbol{M})}\mathcal{L}_\pi(\boldsymbol{M}) \succeq \sigma_{\min}(\overline{\boldsymbol{X}}_\pi\overline{\boldsymbol{X}}_\pi^\top)\boldsymbol{I}_p$. Owing to Assumption 1(a), this proves the claim. □

A.1.2. PROOF SKETCH OF CONVERGENCE

Proof sketch of Theorem 3.2.2. The high level idea is this: we want to prove that $\mathcal{L}_\pi(\boldsymbol{M}_0^k) \to \mathcal{L}_\pi^*$ as $k \to \infty$. However, we will instead show the much stronger statement that $\mathcal{L}_\pi(\boldsymbol{M}_i^k) \to \mathcal{L}_\pi^*$ for all $i \in [m]$. Our high level approach is heavily inspired by the proof strategies in Wu et al. (2019); Ahn et al. (2020). Indeed, many of the technical lemmas in Appendix A.1.4 are analogous to ones proved in Wu et al. (2019), and the motivation for unrolling shuffling mini-batch updates to an epoch update with additional noise comes from Ahn et al. (2020).

As a necessary ingredient of the proof, we will demonstrate that for sufficiently small chosen η_k, we have an update equation that roughly looks like (modulo constants and noise terms)

$$\mathcal{L}_\pi(\boldsymbol{M}_i^{k+1}) - \mathcal{L}_\pi^* \lesssim (1-\eta_k)(\mathcal{L}_\pi(\boldsymbol{M}_i^k) - \mathcal{L}_\pi^*) + O(\eta_k^2) \qquad \text{for all } 0 \le i \le m-1. \tag{5}$$

Remark A.1.4. *Note that it is* not *necessarily the case that*

$$\mathcal{L}_\pi(\boldsymbol{M}_{i+1}^k) - \mathcal{L}_\pi^* \lesssim (1-\eta_k)(\mathcal{L}_\pi(\boldsymbol{M}_i^k) - \mathcal{L}_\pi^*) + O(\eta_k^2)$$

That is, the SS excess risk $\mathcal{L}_\pi$ does not necessarily "decrease" from one iterate to the next; however, we can instead guarantee that the per-epoch progress bound (Equation (5)) holds for any fixed iteration index $i \in [m]$ after every epoch.

We impose an ordering relation on pairs (a, b) in the natural way: we say $(a, b) \le (i, k)$ if $k = b$ and $a \le i$, or if $b < k$. This is just tracking whether the iteration index (a, b) (the ath iterate of the bth epoch) is seen before the iterate (i, k). To complete the induction on an iterate $(i, k+1)$ we need three inductive hypotheses $L[a, b]$, $D[a, b]$, and $R[a, b]$ to hold for all $(a, b) < (i, k+1)$. We define them formally below.

Hypothesis 1 (Loss stays bounded by an absolute constant)**.** *For all a, b satisfying $0 \le a \le m-1$ and $b \ge 1$, the inductive property $L[a, b]$ states $\mathcal{L}_\pi(\Theta_a^b) \le C_L$, for some appropriately chosen absolute constant C_L.*

In particular, we can set $C_L \triangleq \max\left\{\mathcal{L}_\pi(\Theta_t^1) : 0 \le t \le m-1\right\}$. Since we only look at the loss values for the first epoch, C_L is indeed an absolute constant depending on π.

Hypothesis 2 (Loss satisfies one-epoch inequality)**.** *For all a, b satisfying $0 \le a \le m-1$ and $b > 1$, the inductive property $R[a, b]$ states that*

$$\mathcal{L}_\pi(\boldsymbol{M}_a^b) - \mathcal{L}_\pi^* \le \left(1 - \frac{\alpha_\pi \eta_k}{2}\right)(\mathcal{L}_\pi(\boldsymbol{M}_a^{b-1}) - \mathcal{L}_\pi^*) + O(\eta_k^2),$$

where the constant hidden in the $O(\eta_k^2)$ does not depend on k.

Hypothesis 3 (Approximate invariances hold)**.** *For all a, b satisfying $0 \le a \le m-1$ and $b \ge 1$, the inductive property $D[a, b]$ states that*

$$\left\|\boldsymbol{D}_a^b\right\|_2 \le \begin{cases} C_D \sum_{t=1}^{b-1} \eta_t^2 \le \frac{1}{2} & \text{if } a = 0, \\ C_D \sum_{t=1}^{b} \eta_t^2 \le \frac{1}{2} & \text{otherwise,} \end{cases}$$

where C_D is an appropriately chosen absolute constant which does not depend on a or b.

Since the first iterate of the kth epoch Θ_0^k is the same as the last iterate of the $(k-1)$th epoch Θ_m^k, the same convention applies to inductive hypotheses; for example, by $L[m, k-1]$ we mean $L[0, k]$.

In particular, the inductive hypotheses imply the following claims.

(i) By Corollary A.1.8, $L[a, b]$ implies that $\left\|\boldsymbol{M}_a^b\right\|_2 \le \frac{C_L^{1/2} + \|\boldsymbol{Y}_\pi\|_F}{\sigma_{\min}(\overline{\boldsymbol{X}}_\pi^\top)} \triangleq \xi$.

(ii) Also by Corollary A.1.8, $D[a, b]$ and $L[a, b]$ together imply that we have $\left\|\boldsymbol{W}_a^b\right\|_2^2 \le d^2(\frac{1}{2} + \xi)$ and $\left\|\boldsymbol{\Gamma}_a^b\right\|_2^2 \le \frac{3}{2} + d^2(\frac{1}{2} + \xi)$. For the sake of notational convenience we will write $C_{\mathrm{w}} \triangleq \sqrt{\frac{3}{2} + d^2(\frac{1}{2} + \xi)}$, so that $\max\left\{\left\|\boldsymbol{W}_a^b\right\|_2, \left\|\boldsymbol{\Gamma}_a^b\right\|_2\right\} \le C_{\mathrm{w}}$.

(iii) By Corollary A.1.13, $D[a, b]$ implies that $\sigma_{\min}(\boldsymbol{\Gamma}_a^b)^2 \ge 1/2$.

(iv) By Proposition A.1.23, if $R[a,b]$ holds for all (a,b), then for appropriately chosen η_k, the risk $\mathcal{L}_\pi(\boldsymbol{M}_a^b)$ converges to $\mathcal{L}_\pi^*$ at a sublinear rate.

We will explain at a high level how these statements together allow us to conclude that $L[i,k+1]$, $D[i,k+1]$, and $R[i,k+1]$ hold. The idea, as in Ahn, Yun, and Sra (2020), is to accumulate the gradient updates in each epoch and isolate the signal and noise components of each gradient update. For clarity of exposition, we assume for now that $i=0$. Here are a couple subtleties which we spell out explicitly, including how to generalize to $i>0$.

- We are not directly performing gradient updates on $\boldsymbol{M}$; we instead perform gradient updates on $\boldsymbol{W}$ and $\boldsymbol{\Gamma}$. Nevertheless, the *effective* gradient signal for $\boldsymbol{M}$ can still be extracted, and we term the remaining noise the *mismatched gradient noise*. For every iterate (j,k), this will formally be denoted by $\boldsymbol{q}_j^k$.
- We are not taking a full batch gradient step from $\boldsymbol{M}_0^k$ to $\boldsymbol{M}_0^{k+1}$. Rather, we are taking mini-batch updates which induce path dependency. Nevertheless, as previous works have shown, even at iterate (j,k), we can still extract the full-batch gradient signal evaluated at $\boldsymbol{M}_0^k$, and we term the remaining noise the *path dependent noise*. For every iterate (j,k), this will formally be denoted by $\boldsymbol{e}_j^k$.
- If $i>0$, then the stepsize changes from η_k to η_{k+1} in the middle of our pass through the entire dataset. Nevertheless, it's not hard to see that this noise should be relatively small, of order $\eta_{k+1}-\eta_k$ — which is $O(\eta_k^2)$, as $\eta_k=\Omega(1/k)$. We will call this the *stepsize noise*, the accumulation of which for an epoch update starting from iterate (i,k) to $(i,k+1)$ will be denoted by $\boldsymbol{s}_{(i,k+1)}^{(i,k)}$.

We can accumulate these noise terms across the update across epoch k to form a composite noise term $\boldsymbol{r}^k$. The *full-batch update signal for* $\boldsymbol{M}$ starting from $\boldsymbol{M}_0^k$ will be denoted by $\tilde{\boldsymbol{g}}^k$. We emphasize that $\tilde{\boldsymbol{g}}^k \neq \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k)$ because we only perform direct gradient updates on the component layers $\boldsymbol{W}$ and $\boldsymbol{\Gamma}$. Then as we will show in Appendix A.1.3, we can write

$$\boldsymbol{M}_0^{k+1} = \boldsymbol{M}_0^k - \eta_k\tilde{\boldsymbol{g}}^k + \eta_k^2\boldsymbol{r}^k. \tag{6}$$

Next, as seen in Lemma A.1.2, $\mathcal{L}_\pi$ is globally G_π–smooth with respect to $\boldsymbol{M}$ for some absolute constant G_π which *depends* on π. Thus, using the smoothness inequality as in Property 1, we obtain

$$\mathcal{L}_\pi(\boldsymbol{M}_0^{k+1}) - \mathcal{L}_\pi(\boldsymbol{M}_0^k) \le \left\langle \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k), \boldsymbol{M}_0^{k+1}-\boldsymbol{M}_0^k\right\rangle_F + \frac{G_\pi}{2}\left\|\boldsymbol{M}_0^{k+1}-\boldsymbol{M}_0^k\right\|_F^2.$$

The main idea is that we have the following inequality (proved in Lemma A.1.14) that shows that even though $\tilde{\boldsymbol{g}}^k \neq \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k)$, it is nonetheless correlated to the "correct" gradient update $\boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k)$:

$$\left\langle \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k), \tilde{\boldsymbol{g}}^k\right\rangle_F \ge \sigma_{\min}(\boldsymbol{\Gamma}_0^k)^2\left\|\boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k)\right\|_F^2 \ge \frac{1}{2}\left\|\boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k)\right\|_F^2,$$

due to the inductive hypothesis $D[0,k]$.

For the stated stepsizes η_k, one can then plug in the gradient update Equation (6) and massage the inequalities a bit to obtain that

$$\mathcal{L}_\pi(\boldsymbol{M}_0^{k+1}) - \mathcal{L}_\pi(\boldsymbol{M}_0^k) \le -\frac{\eta_k}{4}\left\|\boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k)\right\|_F^2 + O(\eta_k^2), \tag{7}$$

where the constant hidden by the big–O notation is $\mathrm{poly}(m, C_{\mathrm{w}}, C_L, \left\|\overline{\boldsymbol{X}}_\pi\right\|_F)$.

We now use α_π-strong convexity of $\mathcal{L}_\pi$ with respect to $\boldsymbol{M}$ (and hence α_π-PŁ) shown in Lemma A.1.3 to obtain

$$\mathcal{L}_\pi(\boldsymbol{M}_0^{k+1}) - \mathcal{L}_\pi^* \le \left(1-\frac{\alpha_\pi\eta_k}{2}\right)(\mathcal{L}_\pi(\boldsymbol{M}_0^k)-\mathcal{L}_\pi^*) + O(\eta_k^2). \tag{8}$$

Note that this is precisely the statement of $R[0,k+1]$.

Provided that we can appropriately bound the noise terms $\boldsymbol{r}^k$ to get the asserted $O(\eta_k^2)$ term above, this will imply $R[0,k+1]$. For sufficiently small stepsizes η_k, we can also use Equation (8) to prove $L[0,k+1]$.

On the other hand, to prove $D[0, k+1]$, we can directly bound the update $\left\|\boldsymbol{D}_0^{k+1} - \boldsymbol{D}_{m-1}^k\right\|_2 \leq O(\eta_k^2)$ and combine this with the inductive hypothesis $D[m-1, k]$ using the triangle inequality. If the stepsize $\eta_k = O(1/k^\beta)$ for $1/2 < \beta < 1$, then $\sum_{k\geq 1} \eta_k^2 < \infty$, so the absolute constant C_D can be picked such that $\left\|\boldsymbol{D}_0^{k+1}\right\|_2 \leq \frac{1}{2}$.

Hence, $R[0, k]$, as stated in Equation (8), holds for all k by induction. We can thus unroll the inequality and conclude that $\mathcal{L}_\pi(\boldsymbol{M}_0^k)$ converges to $\mathcal{L}_\pi^*$ under the stated stepsize assumptions, as desired. □

We now outline the structure of the proceeding sections, which fill in the details of the above proof sketch. In Appendix A.1.3, we explicitly write out the accumulation of gradient updates across an entire epoch, decomposing into signal and noise components. In Appendix A.1.4, we prove some technical lemmas controlling the singular values and norms of various weight matrices and gradients via the approximate invariance matrix $\boldsymbol{D}$ and the inductive hypotheses. In Appendix A.1.5 we leverage the norm bounds developed in Appendix A.1.4 to demonstrate that the accumulated noise terms defined in Appendix A.1.3 are negligible. Using these results, we are able to establish the $R[i, k+1]$ and $L[i, k+1]$ in Appendix A.1.6. We then turn to bounding the approximate invariances to establish $D[i, k+1]$ in Appendix A.1.7. The stray details of the induction are spelled out in Appendix A.1.8.

A.1.3. Rewriting SS epoch gradient updates

To show that $L[0, k+1]$ holds, we need to accumulate gradients from $\boldsymbol{M}_0^k$ to $\boldsymbol{M}_0^{k+1}$.

First, we look at a single iterate update. For every $j < m$ we have

$$\begin{aligned}\boldsymbol{M}_{j+1}^k &= (\boldsymbol{W}_j^k - \eta_k \boldsymbol{\nabla}_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k))(\boldsymbol{\Gamma}_j^k - \eta_k \boldsymbol{\nabla}_{\boldsymbol{\Gamma}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k)) &(9)\\ &= \boldsymbol{M}_j^k - \eta_k \boldsymbol{g}_j^k + \eta_k^2 \boldsymbol{q}_j^k, &(10)\end{aligned}$$

where we have defined

$$\boldsymbol{g}_j^k \triangleq \boldsymbol{\nabla}_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k)\boldsymbol{\Gamma}_j^k + \boldsymbol{W}_j^k \boldsymbol{\nabla}_{\boldsymbol{\Gamma}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k), \tag{11}$$

which is the gradient of the $(j+1)$th batch of $\overline{\boldsymbol{X}}_\pi$ evaluated on the jth iterate on epoch k, and

$$\boldsymbol{q}_j^k \triangleq \boldsymbol{\nabla}_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k)\boldsymbol{\nabla}_{\boldsymbol{\Gamma}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k), \tag{12}$$

which is the mismatched gradient noise term associated with the fact that we performed gradient updates on $\boldsymbol{W}$ and $\boldsymbol{\Gamma}$ rather than $\boldsymbol{M}$ directly.

The key observation here is that

$$\boldsymbol{g}_j^k = \boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \boldsymbol{M}_j^k)(\boldsymbol{\Gamma}_j^k)^2 + \boldsymbol{W}_j^k \operatorname{diag}((\boldsymbol{W}_j^k)^\top \boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \boldsymbol{M}_j^k)).$$

In other words, $\boldsymbol{g}_j^k$ is correlated to the "true" mini-batch gradient $\boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \boldsymbol{M}_j^k)$ with respect to $\boldsymbol{M}$ through the "interaction terms" $\boldsymbol{\Gamma}_j^k$ and $\boldsymbol{W}_j^k$.

We show in Lemma A.1.16 that we can control the size of the noise terms $\boldsymbol{q}_j^k$ which arise from the fact that we are not truly taking gradient updates with respect to $\boldsymbol{M}$. More specifically, Lemma A.1.16 implies that $\left\|\boldsymbol{q}_j^k\right\|_F = O(1)$.

Next, we actually accumulate gradients. The main obstacle we have to deal with is that the mini-batch updates prevent the gradient accumulation from being exactly equal to the full-batch update starting at $\boldsymbol{M}_0^k$. Inspired by the approach in Ahn et al. (2020, Theorem 1), we separate out the gradient update $\boldsymbol{g}_j^k$ into a signal term $\tilde{\boldsymbol{g}}_j^k$ and noise term $\boldsymbol{e}_j^k$. Specifically, we write

$$\boldsymbol{M}_{j+1}^k = \boldsymbol{M}_j^k - \eta_k \tilde{\boldsymbol{g}}_j^k + \eta_k^2 \boldsymbol{e}_j^k + \eta_k^2 \boldsymbol{q}_j^k, \tag{13}$$

where

$$\tilde{\boldsymbol{g}}_j^k \triangleq \boldsymbol{\nabla}_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_0^k)\boldsymbol{\Gamma}_0^k + \boldsymbol{W}_0^k \boldsymbol{\nabla}_{\boldsymbol{\Gamma}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_0^k), \tag{14}$$

is the signal of the gradient update of the $(j+1)$th batch evaluated with parameter values Θ_0^k (instead of Θ_j^k) and

$$\boldsymbol{e}_j^k \triangleq \frac{\tilde{\boldsymbol{g}}_j^k - \boldsymbol{g}_j^k}{\eta_k}. \tag{15}$$

In particular, in Lemma A.1.18 below we show that $\left\|\boldsymbol{e}_j^k\right\|_F = O(1)$, so that indeed the noise term is negligible with respect to the true gradient signal.

Taking this as given for now, when we accumulate the gradient updates across epoch k, we see that we can define

$$\tilde{\boldsymbol{g}}^k \triangleq \sum_{j=0}^{m-1} \tilde{\boldsymbol{g}}_j^k = \boldsymbol{\nabla}_{\boldsymbol{W}} \mathcal{L}_\pi(\Theta_0^k)\boldsymbol{\Gamma}_0^k + \boldsymbol{W}_0^k \boldsymbol{\nabla}_{\boldsymbol{\Gamma}} \mathcal{L}_\pi(\Theta_0^k), \tag{16}$$

so that the accumulation reads

$$\boldsymbol{M}_0^{k+1} = \boldsymbol{M}_0^k - \eta_k \tilde{\boldsymbol{g}}^k + \eta_k^2 \sum_{j=0}^{m-1} (\boldsymbol{e}_j^k + \boldsymbol{q}_j^k) \tag{17}$$

$$= \boldsymbol{M}_0^k - \eta_k \tilde{\boldsymbol{g}}^k + \eta_k^2 \boldsymbol{r}^k, \tag{18}$$

where we have additionally defined the composite noise term:

$$\boldsymbol{r}^k \triangleq \sum_{j=0}^{m-1} (\boldsymbol{e}_j^k + \boldsymbol{q}_j^k), \tag{19}$$

Note that if we instead start from $i > 0$, then the composite noise term $\boldsymbol{r}^k$ will have an additional noise term $\boldsymbol{s}_{(i,k+1)}^{(i,k)}$, which we will address in Appendix A.1.5. In particular, we show there that the norm of $\boldsymbol{s}_{(i,k+1)}^{(i,k)}$ is $O(1)$. Combining this with Lemmas A.1.16 and A.1.18, we can conclude that $\left\|\boldsymbol{r}^k\right\|_F = O(1)$.

A.1.4. Norm and singular value bounds based on approximate invariances

In this section, we prove several helper lemmas which help us bound noise terms in Appendix A.1.5 and the approximate invariances in Appendix A.1.7.

Upper bounds on the norms of $\boldsymbol{W}$ and $\boldsymbol{\Gamma}$. Much of Wu, Wang, and Ma (2019) is dedicated towards showing that the approximate invariances control the weight norms. The trouble with directly extending their strategy lies in the fact that in our setting the invariance $\boldsymbol{D}$ is diagonal, which complicates the process of bounding various matrix norms. We first state the following technical lemma which involves the operator norm of Hadamard products.

Lemma A.1.5 (3.1f in Johnson (1990))**.** *Let $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{d\times d}$ be matrices such that $\boldsymbol{A}$ is positive definite. Then $\|\boldsymbol{A} \odot \boldsymbol{B}\|_2 \leq \|\boldsymbol{A}\|_2 \|\boldsymbol{B}\|_2$*

We leverage Lemma A.1.5 to prove the following useful helper lemma that relates bounds on $\left\|\boldsymbol{I} \odot \boldsymbol{W}^\top \boldsymbol{W}\right\|_2$ to $\|\boldsymbol{W}\|_2$.

Lemma A.1.6. *Suppose $\left\|\boldsymbol{I} \odot \boldsymbol{W}^\top \boldsymbol{W}\right\|_2 \leq \beta$, where $\boldsymbol{W} \in \mathbb{R}^{p\times d}$. Then $\|\boldsymbol{W}\|_2 \leq \sqrt{d\beta}$. Conversely, if $\|\boldsymbol{W}\|_2 \leq \beta$, then $\left\|\boldsymbol{I} \odot \boldsymbol{W}^\top \boldsymbol{W}\right\|_2 \leq \beta^2$.*

Proof. Note that $\boldsymbol{I} \odot \boldsymbol{W}^\top \boldsymbol{W}$ is a diagonal matrix with diagonal entries $\boldsymbol{W}_{:,1}^\top \boldsymbol{W}_{:,1}, \boldsymbol{W}_{:,2}^\top \boldsymbol{W}_{:,2}, \ldots, \boldsymbol{W}_{:,d}^\top \boldsymbol{W}_{:,d}$, where $\boldsymbol{W}_{:,i}$ denotes the ith column of $\boldsymbol{W}$. Hence $\mathrm{Tr}\big(\boldsymbol{I} \odot \boldsymbol{W}^\top \boldsymbol{W}\big) = \|\boldsymbol{W}\|_F^2$. Hence $\|\boldsymbol{W}\|_F^2 \leq d\beta$ (or tighter by replacing d with the rank of $\boldsymbol{W}$), from which it follows that $\|\boldsymbol{W}\|_2 \leq \sqrt{d\beta}$. For the other direction, we set $\boldsymbol{A} = \boldsymbol{I}$ and $\boldsymbol{B} = \boldsymbol{W}^\top \boldsymbol{W}$ in Lemma A.1.5, so $\left\|\boldsymbol{I} \odot \boldsymbol{W}^\top \boldsymbol{W}\right\|_2 \leq \|\boldsymbol{W}\|_2^2 \leq \beta^2$, as desired. □

With Lemma A.1.6 in hand, we prove the following technical lemma which gives a uniform bound on the norms of $\boldsymbol{\Gamma}$ and $\boldsymbol{W}$ based on $\xi = \|\boldsymbol{W}\boldsymbol{\Gamma}\|_2$.

Lemma A.1.7. *If $\|\boldsymbol{D}\|_2 \leq \epsilon < 1$ and $\|\boldsymbol{W}\boldsymbol{\Gamma}\|_2 \leq \xi$, we have*

$$\|\boldsymbol{W}\|_2 \leq d\sqrt{1 - \epsilon + \xi},$$

and

$$\left\|\boldsymbol{\Gamma}^2\right\|_2 \leq 1 + \epsilon + d^2(1 - \epsilon + \xi).$$

Proof. We have from $\|\boldsymbol{W}\boldsymbol{\Gamma}\|_2 \leq \xi$ that

$$\left\|\boldsymbol{W}\boldsymbol{\Gamma}^2\boldsymbol{W}^\top\right\|_2 \leq \xi^2.$$

Next, our hypothesis that $\|\boldsymbol{D}\|_2 = \left\|\boldsymbol{I} + \operatorname{diag}(\boldsymbol{W}^\top\boldsymbol{W}) - \boldsymbol{\Gamma}^2\right\|_2 \leq \epsilon$ implies that

$$\boldsymbol{W}\boldsymbol{\Gamma}^2\boldsymbol{W}^\top \succeq \boldsymbol{W}((1-\epsilon)\boldsymbol{I} + \operatorname{diag}(\boldsymbol{W}^\top\boldsymbol{W}))\boldsymbol{W}^\top.$$

Taking norms of both sides and applying the reverse triangle inequality, we obtain that

$$\xi^2 \geq \left\|\boldsymbol{W}\operatorname{diag}(\boldsymbol{W}^\top\boldsymbol{W})\boldsymbol{W}^\top\right\|_2 - (1-\epsilon)\|\boldsymbol{W}\|_2^2.$$

We now lower bound $\left\|\boldsymbol{W}\operatorname{diag}(\boldsymbol{W}^\top\boldsymbol{W})\boldsymbol{W}^\top\right\|_2$. In particular, we expand out the matrix product. Note here that $\operatorname{diag}(\boldsymbol{W}^\top\boldsymbol{W})_{i,i} = \|\boldsymbol{W}_{:,i}\|_2^2$. Thus we can write $\boldsymbol{W}\operatorname{diag}(\boldsymbol{W}^\top\boldsymbol{W})\boldsymbol{W}^\top$ as

$$\begin{bmatrix} \boldsymbol{W}_{:,1} & \boldsymbol{W}_{:,2} & \cdots & \boldsymbol{W}_{:,d} \end{bmatrix} \begin{bmatrix} \|\boldsymbol{W}_{:,1}\|_2^2 & & & \\ & \|\boldsymbol{W}_{:,2}\|_2^2 & & \\ & & \ddots & \\ & & & \|\boldsymbol{W}_{:,d}\|_2^2 \end{bmatrix} \begin{bmatrix} \boldsymbol{W}_{:,1}^\top \\ \boldsymbol{W}_{:,2}^\top \\ \vdots \\ \boldsymbol{W}_{:,d}^\top \end{bmatrix},$$

from which we observe that the ith diagonal entry of $\boldsymbol{W}\operatorname{diag}(\boldsymbol{W}^\top\boldsymbol{W})\boldsymbol{W}^\top$ is

$$(\boldsymbol{W}\operatorname{diag}(\boldsymbol{W}^\top\boldsymbol{W})\boldsymbol{W}^\top)_{i,i} = \sum_{j=1}^d \|\boldsymbol{W}_{:,j}\|_2^2 W_{i,j}^2.$$

It follows that $\operatorname{Tr}\big(\boldsymbol{W}\operatorname{diag}(\boldsymbol{W}^\top\boldsymbol{W})\boldsymbol{W}^\top\big) = \sum_{j=1}^d \|\boldsymbol{W}_{:,j}\|_2^4$. Note that $\|\boldsymbol{A}\|_2 \geq \max_{i,j}|A_{i,j}|$ (the RHS is also known as the *max norm*). For our case we set $\boldsymbol{A} = \boldsymbol{W}\operatorname{diag}(\boldsymbol{W}^\top\boldsymbol{W})\boldsymbol{W}^\top$ and note that the diagonal is nonnegative. So in fact in our case we obtain

$$\left\|\boldsymbol{W}\operatorname{diag}(\boldsymbol{W}^\top\boldsymbol{W})\boldsymbol{W}^\top\right\|_2 \geq \frac{1}{d}\sum_{j=1}^d \|\boldsymbol{W}_{:,j}\|_2^4.$$

Now notice that $\sum_j \|\boldsymbol{W}_{:,j}\|_2^4 = \sum_j(\sum_i W_{i,j}^2)^2$. Applying Cauchy-Schwarz to the outer sum we find that

$$\sum_j \|\boldsymbol{W}_{:,j}\|_2^4 \geq \frac{(\sum_j\sum_i W_{i,j}^2)^2}{d},$$

but the RHS is equal to $\|\boldsymbol{W}\|_F^4$. Since $\|\boldsymbol{W}\|_F \geq \|\boldsymbol{W}\|_2$, we conclude that

$$\left\|\boldsymbol{W}\operatorname{diag}(\boldsymbol{W}^\top\boldsymbol{W})\boldsymbol{W}^\top\right\|_2 \geq \frac{\|\boldsymbol{W}\|_2^4}{d^2}.$$

In summary, we have

$$\frac{\|\boldsymbol{W}\|_2^4}{d^2} - (1-\epsilon)\|\boldsymbol{W}\|_2^2 - \xi^2 \leq 0.$$

Applying the quadratic formula, we find that

$$\|\boldsymbol{W}\|_2 \leq d\sqrt{1-\epsilon+\xi}.$$

For the bound on $\|\boldsymbol{\Gamma}\|_2$, we start from the definition of $\boldsymbol{D}$ and apply the reverse triangle inequality to obtain

$$\left|1 + \left\|\operatorname{diag}(\boldsymbol{W}^\top\boldsymbol{W})\right\|_2 - \left\|\boldsymbol{\Gamma}^2\right\|_2\right| \leq \epsilon,$$

so we obtain

$$\left\|\boldsymbol{\Gamma}^2\right\|_2 \leq 1 + \epsilon + \|\boldsymbol{W}\|_2^2,$$

where we used $\left\|\boldsymbol{I}\odot\boldsymbol{W}^\top\boldsymbol{W}\right\|_2 \leq \|\boldsymbol{W}\|_2^2$ from Lemma A.1.6. From this, the conclusion directly follows. □

Under the inductive hypotheses, Lemma A.1.7 implies that we can uniformly bound $\max\left\{\left\|\boldsymbol{W}_j^k\right\|_2, \left\|\boldsymbol{\Gamma}_j^k\right\|_2\right\}$. This is spelled out in the following corollary.

Corollary A.1.8 (Norms stay bounded)**.** *Suppose that $L[j,k]$ and $D[j,k]$ hold. Define*

$$C_{\mathrm{w}} \triangleq \sqrt{\frac{3}{2} + d^2\left(\frac{1}{2} + \xi\right)},$$

with

$$\xi \triangleq \frac{C_L^{1/2} + \|\boldsymbol{Y}_\pi\|_F}{\sigma_{\min}(\overline{\boldsymbol{X}}_\pi^\top)}.$$

Here C_L was defined in Hypothesis 1. Then

$$\left\|\boldsymbol{M}_j^k\right\| \leq \xi,$$

and

$$\max\left\{\left\|\boldsymbol{W}_j^k\right\|_2, \left\|\boldsymbol{\Gamma}_j^k\right\|_2\right\} \leq C_{\mathrm{w}}.$$

Proof. We have by triangle inequality that

$$\left\|\boldsymbol{M}_j^k\overline{\boldsymbol{X}}_\pi\right\|_2 \leq \left\|\boldsymbol{M}_j^k\overline{\boldsymbol{X}}_\pi\right\|_F \leq \left\|\boldsymbol{Y}_\pi - \boldsymbol{M}_j^k\overline{\boldsymbol{X}}_\pi\right\|_F + \|\boldsymbol{Y}_\pi\|_F \leq \mathcal{L}_\pi(\boldsymbol{M}_j^k)^{1/2} + \|\boldsymbol{Y}_\pi\|_F.$$

Since $L[j,k]$ holds, we have have $\left\|\boldsymbol{Y}_\pi - \boldsymbol{M}_j^k\overline{\boldsymbol{X}}_\pi\right\|_F^2 \leq C_L$. Furthermore, as $n \geq d$, we know that $\left\|\boldsymbol{M}_j^k\overline{\boldsymbol{X}}_\pi\right\|_2 \geq \sigma_{\min}(\overline{\boldsymbol{X}}_\pi^\top)\left\|\boldsymbol{M}_j^k\right\|_2$ and by Item Assumption 1(a) we have $\sigma_{\min}(\overline{\boldsymbol{X}}_\pi^\top) > 0$. Hence we obtain

$$\left\|\boldsymbol{M}_j^k\right\|_2 \leq \frac{C_L^{1/2} + \|\boldsymbol{Y}_\pi\|_F}{\sigma_{\min}(\overline{\boldsymbol{X}}_\pi^\top)} = \xi.$$

It follows that ξ works as a bound on $\left\|\boldsymbol{M}_j^k\right\|_2$ for the application of Lemma A.1.7. Since $D[j,k]$ holds by assumption, this means that the hypothesis on $\boldsymbol{D}_j^k$ is satisfied with $\epsilon = 1/2$. In summary, all the hypotheses of Lemma A.1.7 are satisfied. We can thus conclude that

$$\max\left\{\left\|\boldsymbol{W}_j^k\right\|_2, \left\|\boldsymbol{\Gamma}_j^k\right\|_2\right\} \leq C_{\mathrm{w}},$$

as desired. □

The importance of these upper bounds on weight norms is that they allow us to upper bound the norms of gradients of $\mathcal{L}$ with respect to various parameters.

Upper bounding the norms of gradients. The following lemma gives an upper bound on the norms of various gradients.

Lemma A.1.9. *For any $a \in [m]$ and $\Theta = (\boldsymbol{W}, \boldsymbol{\Gamma})$ we have*

$$\begin{aligned}
\|\nabla_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{X}_\pi^a;\Theta)\|_F^2 &\leq \|\boldsymbol{\Gamma}\|_2^2\|\mathsf{BN}(\boldsymbol{X}_\pi^a)\|_2^2\mathcal{L}(\boldsymbol{X}_\pi^a;\Theta)\\
\|\nabla_{\boldsymbol{\Gamma}}\mathcal{L}(\boldsymbol{X}_\pi^a;\Theta)\|_F^2 &\leq \|\boldsymbol{W}\|_2^2\|\mathsf{BN}(\boldsymbol{X}_\pi^a)\|_2^2\mathcal{L}(\boldsymbol{X}_\pi^a;\Theta)\\
\|\nabla_{M}\mathcal{L}(\boldsymbol{X}_\pi^a;\boldsymbol{M})\|_F^2 &\leq \|\mathsf{BN}(\boldsymbol{X}_\pi^a)\|_2^2\mathcal{L}(\boldsymbol{X}_\pi^a;\boldsymbol{M})
\end{aligned}$$

Proof. First, we have by definition

$$\mathcal{L}(\boldsymbol{X}_\pi^a;\Theta) = \|\boldsymbol{W}\boldsymbol{\Gamma}\mathsf{BN}(\boldsymbol{X}_\pi^a) - \boldsymbol{Y}_\pi^a\|_F^2.$$

Hence, the mini-batch gradients can be computed explicitly as

$$\nabla_{\boldsymbol{M}}\mathcal{L}(\boldsymbol{X}_\pi^a;\boldsymbol{M}) = -(\boldsymbol{Y}_\pi^a - \boldsymbol{M}\mathsf{BN}(\boldsymbol{X}_\pi^a))\mathsf{BN}(\boldsymbol{X}_\pi^a)^\top, \tag{20}$$

$$\nabla_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{X}_\pi^a;\Theta) = \nabla_{\boldsymbol{M}}\mathcal{L}(\boldsymbol{X}_\pi^a;\boldsymbol{M})\boldsymbol{\Gamma}, \tag{21}$$

$$\nabla_{\boldsymbol{\Gamma}}\mathcal{L}(\boldsymbol{X}_\pi^a;\Theta) = \mathrm{diag}(\boldsymbol{W}^\top\nabla_{\boldsymbol{M}}\mathcal{L}(\boldsymbol{X}_\pi^a;\boldsymbol{M})). \tag{22}$$

Since $\mathcal{L}(\boldsymbol{X}_\pi^a; \boldsymbol{M}) = \|\boldsymbol{Y}_\pi^a - \boldsymbol{M}\mathsf{BN}(\boldsymbol{X}_\pi^a)\|_F^2$ and $\|\boldsymbol{AB}\|_F \leq \|\boldsymbol{A}\|_2 \|\boldsymbol{B}\|_F$, Equation (20) gives

$$\|\boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}(\boldsymbol{X}_\pi^a; \boldsymbol{M})\|_F^2 \leq \|\mathsf{BN}(\boldsymbol{X}_\pi^a)\|_2^2 \mathcal{L}(\boldsymbol{X}_\pi^a; \boldsymbol{M}).$$

It thus follows from Equation (21) that

$$\|\boldsymbol{\nabla}_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^a; \Theta)\|_F^2 \leq \|\boldsymbol{\Gamma}\|_2^2 \|\boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}(\boldsymbol{X}_\pi^a; \Theta)\|_F^2 \leq \|\boldsymbol{\Gamma}\|_2^2 \|\mathsf{BN}(\boldsymbol{X}_\pi^a)\|_2^2 \mathcal{L}(\boldsymbol{X}_\pi^a; \Theta).$$

Similarly, inspecting Equation (22), since $\|\mathrm{diag}(\boldsymbol{A})\|_F^2 \leq \|\boldsymbol{A}\|_F^2$, we have

$$\|\boldsymbol{\nabla}_{\boldsymbol{\Gamma}} \mathcal{L}(\boldsymbol{X}_\pi^a; \Theta)\|_F^2 \leq \left\|\boldsymbol{W}^\top \boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}(\boldsymbol{X}_\pi^a; \Theta)\right\|_F^2 \leq \|\boldsymbol{W}\|_2^2 \|\mathsf{BN}(\boldsymbol{X}_\pi^a)\|_2^2 \mathcal{L}(\boldsymbol{X}_\pi^a; \Theta).$$

□

As a consequence of Corollary A.1.8, under the inductive hypotheses we can also bound the gradient norms by absolute constants.

Corollary A.1.10. *Assume $D[j,k]$ and $L[j,k]$ hold. Then, for any $a \in [m]$, we have*

$$\begin{aligned}
\left\|\boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}(\boldsymbol{X}_\pi^a; \boldsymbol{M}_j^k)\right\|_F^2 &\leq C_L \|\mathsf{BN}(\boldsymbol{X}_\pi^a)\|_2^2, \\
\left\|\boldsymbol{\nabla}_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^a; \Theta_j^k)\right\|_F^2 &\leq C_{\mathrm{w}}^2 C_L \|\mathsf{BN}(\boldsymbol{X}_\pi^a)\|_2^2, \\
\left\|\boldsymbol{\nabla}_{\boldsymbol{\Gamma}} \mathcal{L}(\boldsymbol{X}_\pi^a; \Theta_j^k)\right\|_F^2 &\leq C_{\mathrm{w}}^2 C_L \|\mathsf{BN}(\boldsymbol{X}_\pi^a)\|_2^2,
\end{aligned}$$

where C_{w} was previously defined in Corollary A.1.8.

We now turn from upper bounds to lower bounds. The crux here is to start with bounding the minimum singular value of $\boldsymbol{\Gamma}$ away from zero. This in turns allows us to lower bound the correlation between $\tilde{\boldsymbol{g}}^k$ and $\boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}_\pi(\boldsymbol{M}_0^k)$ away from zero. As we will see, we can also show similar correlation lower bounds for the cases $i > 0$.

Bounding the minimum singular value of $\boldsymbol{\Gamma}^2$. In order to bound $\sigma_{\min}(\boldsymbol{\Gamma}_i^k)$ away from zero, we need to show that the approximate invariances prevent $\boldsymbol{\Gamma}$ from vanishing on any coordinate. To do so, we appeal to an alternate formulation of the Courant-Fisher theorem for singular values, which we restate below for completeness.

Theorem A.1.11 (Courant-Fisher)**.** *Let $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{m \times n}$. Then $|\sigma_k(\boldsymbol{A}) - \sigma_k(\boldsymbol{B})| \leq \|\boldsymbol{A} - \boldsymbol{B}\|_2$ for $k \in [\min\{m,n\}]$.*

With this in mind, we formally prove that the minimum singular value of $\boldsymbol{\Gamma}^2$ is bounded away from zero.

Lemma A.1.12. *Suppose that $\|\boldsymbol{D}\|_2 = \left\|\boldsymbol{I} + \mathrm{diag}(\boldsymbol{W}^\top \boldsymbol{W} - \boldsymbol{\Gamma}^2)\right\|_2 \leq \epsilon$. Then we have*

$$\sigma_{\min}(\boldsymbol{\Gamma}^2) \geq 1 - \epsilon.$$

Proof. Setting $\boldsymbol{A} \triangleq \boldsymbol{I} + \mathrm{diag}(\boldsymbol{W}^\top \boldsymbol{W})$ and $\boldsymbol{B} \triangleq \boldsymbol{\Gamma}^2$ in Courant-Fisher yields

$$\left|\sigma_d(\boldsymbol{I} + \mathrm{diag}(\boldsymbol{W}^\top \boldsymbol{W})) - \sigma_d(\boldsymbol{\Gamma}^2)\right| \leq \left\|I + \mathrm{diag}(\boldsymbol{W}^\top \boldsymbol{W}) - \boldsymbol{\Gamma}^2\right\|_2.$$

Since the RHS is just $\boldsymbol{D}$, we obtain that

$$\sigma_{\min}(\boldsymbol{\Gamma}^2) \geq 1 + \sigma_{\min}(\mathrm{diag}(\boldsymbol{W}^\top \boldsymbol{W})) - \|\boldsymbol{D}\|_2.$$

The conclusion easily follows. □

Under the inductive hypothesis $D[i,k]$, i.e. $\left\|\boldsymbol{D}_i^k\right\|_2 \leq \frac{1}{2}$, this immediately implies the following corollary. We will see in the following section (in Corollary A.1.15) that this minimum singular value bound for $\boldsymbol{\Gamma}_i^k$ can be interpreted in the following manner. Although the effective PŁ condition evolves dynamically, the associated PŁ constant always stays bounded away from zero.

Corollary A.1.13 (PŁ bounded away from zero)**.** *Assume $D[i,k]$ holds. Then we have*

$$\sigma_{\min}(\boldsymbol{\Gamma}_i^k)^2 \geq \frac{1}{2}.$$

The accumulated gradient signal is correlated with the full-batch gradient signal.

Lemma A.1.14 (Correlation of $\tilde{\boldsymbol{g}}^k$ and $\boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k)$)**.** *For all k, we have*

$$\left\langle \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k), \tilde{\boldsymbol{g}}^k \right\rangle_F \geq \sigma_{\min}(\boldsymbol{\Gamma}_0^k)^2 \left\| \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k) \right\|_F^2.$$

Proof. Recall that we previously defined

$$\tilde{\boldsymbol{g}}^k \triangleq \boldsymbol{\nabla}_{\boldsymbol{W}}\mathcal{L}_\pi(\Theta_0^k)\boldsymbol{\Gamma}_0^k + \boldsymbol{W}_0^k \boldsymbol{\nabla}_{\boldsymbol{\Gamma}}\mathcal{L}_\pi(\Theta_0^k).$$

Note that if we have $\boldsymbol{A}, \boldsymbol{\Lambda} \in \mathbb{R}^{n\times n}$, with $\boldsymbol{\Lambda} = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$ a diagonal matrix with nonnegative entries, then

$$\langle \boldsymbol{A}, \boldsymbol{A}\boldsymbol{\Lambda} \rangle_F = \left\langle \boldsymbol{A}\boldsymbol{\Lambda}^{1/2}, \boldsymbol{A}\boldsymbol{\Lambda}^{1/2} \right\rangle_F = \left\| \boldsymbol{A}\boldsymbol{\Lambda}^{1/2} \right\|_F^2 \geq \min_i \lambda_i \|\boldsymbol{A}\|_F^2.$$

Also, we have

$$\langle \boldsymbol{A}, \operatorname{diag}(\boldsymbol{A}) \rangle_F = \langle \operatorname{diag}(\boldsymbol{A}), \operatorname{diag}(\boldsymbol{A}) \rangle_F = \|\operatorname{diag}(\boldsymbol{A})\|_F^2 \geq 0.$$

Hence combining Equations (20) and (22) and the above inequalities, we have

$$\begin{aligned}
\left\langle \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k), \tilde{\boldsymbol{g}}^k \right\rangle_F &= \left\langle \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k), \boldsymbol{\nabla}_{\boldsymbol{W}}\mathcal{L}_\pi(\Theta_0^k)\boldsymbol{\Gamma}_0^k \right\rangle_F \\
&\quad + \left\langle \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k), \boldsymbol{W}_0^k \boldsymbol{\nabla}_{\boldsymbol{\Gamma}}\mathcal{L}_\pi(\Theta_0^k) \right\rangle_F \\
&= \left\langle \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k), \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k)(\boldsymbol{\Gamma}_0^k)^2 \right\rangle_F \\
&\quad + \left\langle (\boldsymbol{W}_0^k)^\top \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k), \operatorname{diag}((\boldsymbol{W}_0^k)^\top \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k)) \right\rangle_F \\
&\geq \sigma_{\min}(\boldsymbol{\Gamma}_0^k)^2 \left\| \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k) \right\|_F^2.
\end{aligned}$$

□

We obtain the following corollary of the above lemma and Corollary A.1.13.

Corollary A.1.15. *Assume $D[0,k]$ holds. We have*

$$\left\langle \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k), \tilde{\boldsymbol{g}}^k \right\rangle_F \geq \frac{1}{2} \left\| \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_0^k) \right\|_F^2. \tag{23}$$

More generally, assume $D[i,k]$ holds. We have

$$\left\langle \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_i^k), \tilde{\boldsymbol{g}}^{(i,k)} \right\rangle_F \geq \frac{1}{2} \left\| \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M}_i^k) \right\|_F^2, \tag{24}$$

where $\tilde{\boldsymbol{g}}^{(i,k)}$ is the analogous quantity to $\tilde{\boldsymbol{g}}^k$ for accumulating gradients starting at iterate (i,k) rather than $(0,k)$. It is defined more formally in Equation (26).

A.1.5. Bounding noise terms

We now turn to bounding the composite noise term $\boldsymbol{r}^k$. This is crucial to ensure the global convergence via Equation (8) and also to control the approximate invariances.

Mismatched gradient noise is negligible. As promised, we show that the mismatched gradient noise terms $\boldsymbol{q}_j^k$ are negligible when we accumulate gradients from $\boldsymbol{M}_0^k$ to $\boldsymbol{M}_0^{k+1}$.

Lemma A.1.16. *Assume that $L[j,k]$ and $D[j,k]$ hold for $j < m$. Then we have*

$$\left\| \boldsymbol{q}_j^k \right\|_F \leq C_{\mathrm{w}}^2 C_L \left\| \mathsf{BN}(\boldsymbol{X}_\pi^{j+1}) \right\|_2^2.$$

Furthermore, for any $t < m$ we have

$$\sum_{j=0}^{t} \left\| \boldsymbol{q}_j^k \right\|_F \leq C_{\mathrm{w}}^2 C_L \left\| \overline{\boldsymbol{X}}_\pi \right\|_F^2.$$

Proof. Recall the definition of $\boldsymbol{q}_j^k$, reproduced here for reference:

$$\boldsymbol{q}_j^k \triangleq \boldsymbol{\nabla}_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k) \boldsymbol{\nabla}_{\boldsymbol{\Gamma}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k).$$

Since $L[j,k]$ and $D[j,k]$ hold for (j,k), we can apply Corollary A.1.10 to conclude that

$$\left\|\boldsymbol{q}_j^k\right\|_F \leq C_{\mathrm{w}}^2 C_L \left\|\mathsf{BN}(\boldsymbol{X}_\pi^{j+1})\right\|_2^2.$$

Since the inductive hypotheses hold for every $j < m$, when we accumulate the noise terms from $(0,k)$ to (t,k), we can apply the above bound to conclude that

$$\begin{aligned}
\sum_{j=0}^{t} \left\|\boldsymbol{q}_j^k\right\|_F &\leq \sum_{j=0}^{t} \left\|\mathsf{BN}(\boldsymbol{X}_\pi^{j+1})\right\|_2^2 C_{\mathrm{w}}^2 C_L \\
&\leq C_{\mathrm{w}}^2 C_L \sum_{j=0}^{m-1} \left\|\mathsf{BN}(\boldsymbol{X}_\pi^{j+1})\right\|_F^2 \\
&= C_{\mathrm{w}}^2 C_L \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^2,
\end{aligned}$$

where the last equality used the definition of $\overline{\boldsymbol{X}}_\pi$. □

Path dependent noise arising from mini-batch updates is negligible. In order to bound the noise term coming from mini-batch updates, we first prove the following auxiliary lemma that shows that the iterates don't move far within an epoch.

Lemma A.1.17. *Fix $t \leq m$ and assume $D[j,k]$ and $L[j,k]$ hold for all $j < t$. Then we have*

$$\left\|\boldsymbol{W}_t^k - \boldsymbol{W}_0^k\right\|_2 \leq \sqrt{t}\eta_k C_{\mathrm{w}} C_L^{1/2} \left\|\overline{\boldsymbol{X}}_\pi\right\|_F.$$

The same inequality holds true if we replace $\boldsymbol{W}$ with $\boldsymbol{\Gamma}$.

We also have

$$\left\|\boldsymbol{M}_t^k - \boldsymbol{M}_0^k\right\|_2 \leq 2\sqrt{t}\eta_k C_{\mathrm{w}}^2 C_L^{1/2} \left\|\overline{\boldsymbol{X}}_\pi\right\|_F + \eta_k^2 C_{\mathrm{w}}^2 C_L \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^2.$$

Proof. We have by definition that

$$\boldsymbol{W}_t^k = \boldsymbol{W}_0^k - \eta_k \sum_{j=0}^{t-1} \boldsymbol{\nabla}_{\boldsymbol{W}} \ell(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k).$$

Now, we have

$$\begin{aligned}
\left\|\boldsymbol{W}_t^k - \boldsymbol{W}_0^k\right\|_2 &\leq \eta_k \sum_{j=0}^{t-1} \left\|\boldsymbol{\nabla}_{\boldsymbol{W}} \ell(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k)\right\|_2 \\
&\leq \eta_k C_{\mathrm{w}} C_L^{1/2} \sum_{j=0}^{t} \left\|\mathsf{BN}(\boldsymbol{X}_\pi^{j+1})\right\|_2 \\
&\leq \sqrt{t}\eta_k C_{\mathrm{w}} C_L^{1/2} \left\|\overline{\boldsymbol{X}}_\pi\right\|_F
\end{aligned}$$

where in the first line we have applied the triangle inequality, in the second line we have applied Corollary A.1.10, and in the last line we have applied Cauchy-Schwarz.

The same proof holds for $\boldsymbol{\Gamma}$.

For $\boldsymbol{M}$, Equation (10) gives

$$\boldsymbol{M}_t^k = \boldsymbol{M}_0^k - \eta_k \sum_{j=0}^{t-1} \boldsymbol{g}_j^k + \eta_k^2 \sum_{j=0}^{t-1} \boldsymbol{q}_j^k.$$

Combining Equation (11) and Corollaries A.1.8 and A.1.10 yields

$$\begin{aligned}\left\|\boldsymbol{g}_j^k\right\|_2 &\leq \left\|\boldsymbol{\Gamma}_j^k\right\|_2 \left\|\boldsymbol{\nabla}_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\Theta_j^k)\right\|_2 + \left\|\boldsymbol{W}_j^k\right\|_2 \left\|\boldsymbol{\nabla}_{\boldsymbol{\Gamma}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\Theta_j^k)\right\|_2 \\ &\leq 2C_{\mathrm{w}}^2 C_L^{1/2} \left\|\mathsf{BN}(\boldsymbol{X}_\pi^{j+1})\right\|_2.\end{aligned}$$

Hence, summing up over j, using Cauchy-Schwarz, and applying the noise bound Lemma A.1.16, it follows that

$$\left\|\boldsymbol{M}_t^k - \boldsymbol{M}_0^k\right\|_2 \leq 2\sqrt{t}\eta_k C_{\mathrm{w}}^2 C_L^{1/2} \left\|\overline{\boldsymbol{X}}_\pi\right\|_F + \eta_k^2 C_{\mathrm{w}}^2 C_L \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^2.$$

□

Now we show that the noise term $\left\|\boldsymbol{e}_j^k\right\|_2$ is $O(1)$.

Lemma A.1.18. *If $L[j,k]$ and $D[j,k]$ both hold for all $j < m$ then we have for each j that*

$$\left\|\boldsymbol{e}_j^k\right\|_2 \leq 4\sqrt{j} C_{\mathrm{w}}^2 C_L^{1/2} \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^2 (C_L^{1/2} + C_{\mathrm{w}}^2 \left\|\overline{\boldsymbol{X}}_\pi\right\|_F) + O(\eta_k).$$

Hence, we also have

$$\sum_{i=0}^{m-1} \left\|\boldsymbol{e}_j^k\right\|_2 \leq 4m^{3/2} C_{\mathrm{w}}^2 C_L^{1/2} \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^2 (C_L^{1/2} + C_{\mathrm{w}}^2 \left\|\overline{\boldsymbol{X}}_\pi\right\|_F) + O(\eta_k).$$

Proof. Inspecting the definition of $\boldsymbol{e}_j^k$ (Equation (15)), let us bound the quantity

$$\begin{aligned}\eta_k \boldsymbol{e}_j^k = &\underbrace{\boldsymbol{\nabla}_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\Theta_0^k)\boldsymbol{\Gamma}_0^k - \boldsymbol{\nabla}_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\Theta_j^k)\boldsymbol{\Gamma}_j^k}_{\text{(I)}} \\ &+ \underbrace{\boldsymbol{W}_0^k \boldsymbol{\nabla}_{\boldsymbol{\Gamma}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\Theta_0^k) - \boldsymbol{W}_j^k \boldsymbol{\nabla}_{\boldsymbol{\Gamma}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\Theta_j^k)}_{\text{(II)}}.\end{aligned}$$

First, we have by triangle inequality and the identity Equation (21) that the norm of (I) is at most

$$\begin{aligned}&\left\|\boldsymbol{\nabla}_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\Theta_0^k)\boldsymbol{\Gamma}_0^k - \boldsymbol{\nabla}_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\Theta_0^k)\boldsymbol{\Gamma}_j^k\right\|_2 \\ &\quad + \left\|\boldsymbol{\nabla}_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\Theta_0^k)\boldsymbol{\Gamma}_j^k - \boldsymbol{\nabla}_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\Theta_j^k)\boldsymbol{\Gamma}_j^k\right\|_2 \\ &\leq \left\|\boldsymbol{\Gamma}_0^k(\boldsymbol{\Gamma}_j^k - \boldsymbol{\Gamma}_0^k)\right\|_2 \left\|\boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\boldsymbol{M}_0^k)\right\|_2 \\ &\quad + \left\|\boldsymbol{\Gamma}_j^k\right\|_2 \left\|\boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\boldsymbol{M}_0^k)\boldsymbol{\Gamma}_0^k - \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\boldsymbol{M}_j^k)\boldsymbol{\Gamma}_j^k\right\|_2 \\ &\leq \left\|\boldsymbol{\Gamma}_0^k(\boldsymbol{\Gamma}_j^k - \boldsymbol{\Gamma}_0^k)\right\|_2 \left\|\boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\boldsymbol{M}_0^k)\right\|_2 \\ &\quad + \left\|\boldsymbol{\Gamma}_j^k\right\|_2 \Bigg(\left\|\boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\boldsymbol{M}_0^k)\boldsymbol{\Gamma}_0^k - \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\boldsymbol{M}_0^k)\boldsymbol{\Gamma}_j^k\right\|_2 \\ &\qquad\qquad + \left\|\boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\boldsymbol{M}_0^k)\boldsymbol{\Gamma}_j^k - \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\boldsymbol{M}_j^k)\boldsymbol{\Gamma}_j^k\right\|_2 \Bigg) \\ &\leq (\left\|\boldsymbol{\Gamma}_0^k\right\|_2 + \left\|\boldsymbol{\Gamma}_j^k\right\|_2) \left\|\boldsymbol{\Gamma}_j^k - \boldsymbol{\Gamma}_0^k\right\|_2 \left\|\boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\boldsymbol{M}_0^k)\right\|_2 \\ &\quad + \left\|\boldsymbol{\Gamma}_j^k\right\|_2^2 \left\|\boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\boldsymbol{M}_0^k) - \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}(\boldsymbol{X}_\pi^{j+1};\boldsymbol{M}_j^k)\right\|_2.\end{aligned}$$

Applying the weight bounds in Corollaries A.1.8 and A.1.10 and Lemma A.1.17 to the first term yields an upper bound of

$$2C_{\mathrm{w}} \cdot (\sqrt{j}\eta_k C_{\mathrm{w}} C_L^{1/2} \left\|\overline{\boldsymbol{X}}_\pi\right\|_F) \cdot (\left\|\mathsf{BN}(\boldsymbol{X}_\pi^{j+1})\right\|_2 C_L^{1/2}) \leq 2\sqrt{j}\eta_k C_{\mathrm{w}}^2 C_L \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^2.$$

Turning to the second term, we can apply the smoothness bound in Lemma A.1.2 and the inductive bounds in Corollaries A.1.8 and A.1.10 and Lemma A.1.17 to obtain an upper bound of

$$C_{\mathrm{w}}^2 \left\|\mathsf{BN}(\boldsymbol{X}_\pi^{j+1})\right\|_2^2 \left\|\boldsymbol{M}_j^k - \boldsymbol{M}_0^k\right\|_2 \leq 2\sqrt{j}\eta_k C_{\mathrm{w}}^4 C_L^{1/2} \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^3 + \eta_k^2 C_{\mathrm{w}}^4 C_L \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^4.$$

Putting it together, we have

$$\text{(I)} \leq 2\sqrt{j}\eta_k C_{\mathrm{w}}^2 C_L^{1/2} \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^2 (C_L^{1/2} + C_{\mathrm{w}}^2 \left\|\overline{\boldsymbol{X}}_\pi\right\|_F) + \eta_k^2 C_{\mathrm{w}}^4 C_L \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^4.$$

Similarly, for (II) we have the exact same bound since we can apply Lemma A.1.5 to remove the diagonal operator and uniformly bound $\left\|\boldsymbol{W}_j^k\right\|_2$ and $\left\|\boldsymbol{\Gamma}_j^k\right\|_2$ by C_{w}.

Finally, combining (I) and (II) and dividing through by η_k, we can conclude that

$$\left\|\boldsymbol{e}_j^k\right\|_2 \leq 4\sqrt{j} C_{\mathrm{w}}^2 C_L^{1/2} \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^2 (C_L^{1/2} + C_{\mathrm{w}}^2 \left\|\overline{\boldsymbol{X}}_\pi\right\|_F) + O(\eta_k).$$

Summing up $\left\|\boldsymbol{e}_j^k\right\|_2$ over all j and crudely bounding $\sum_{j=0}^{m-1} \sqrt{j} \leq m^{3/2}$, we see that

$$\sum_{i=0}^{m-1} \left\|\boldsymbol{e}_j^k\right\|_2 \leq 4m^{3/2} C_{\mathrm{w}}^2 C_L^{1/2} \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^2 (C_L^{1/2} + C_{\mathrm{w}}^2 \left\|\overline{\boldsymbol{X}}_\pi\right\|_F) + O(\eta_k).$$

□

Stepsize noise is negligible for $i > 0$. We now quickly show that the effect of generalizing the induction to $i > 0$ is negligible. In particular, we can carry out the same proof, except we will have to redefine the per-epoch update so it can account for gradient updates starting at an arbitrary iterate (i, k) rather than $(0, k)$. We explicitly redefine these terms below by quickly revisiting the signal-noise decomposition in Appendix A.1.3. Recall that a single-iterate update at iteration (a, b) can be written as (Equation (10))

$$\boldsymbol{M}_{a+1}^b = \boldsymbol{M}_a^b - \eta_b \boldsymbol{g}_a^b + \eta_b^2 \boldsymbol{q}_a^b,$$

where we defined

$$\begin{aligned}
\boldsymbol{g}_a^b &\triangleq \boldsymbol{\nabla}_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^{a+1}; \Theta_a^b) \boldsymbol{\Gamma}_a^b + \boldsymbol{W}_a^b \boldsymbol{\nabla}_{\boldsymbol{\Gamma}} \mathcal{L}(\boldsymbol{X}_\pi^{a+1}; \Theta_a^b), \\
\boldsymbol{q}_a^b &\triangleq \boldsymbol{\nabla}_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^{a+1}; \Theta_a^b) \boldsymbol{\nabla}_{\boldsymbol{\Gamma}} \mathcal{L}(\boldsymbol{X}_\pi^{a+1}; \Theta_a^b).
\end{aligned}$$

Consider carrying out the same accumulation as in Appendix A.1.3, but this time choosing (i, k) instead $(0, k)$ as the "pivot." For this purpose, we will change our notational convention a little bit and use superscripts to denote the pivot or the starting point (i, k). As the redefinitions of the "signal" $\tilde{\boldsymbol{g}}_a^b$ (Equation (14)) and path dependent noise $\boldsymbol{e}_a^b$ (Equation (15)), we define

$$\begin{aligned}
\tilde{\boldsymbol{g}}_a^{(i,k)} &\triangleq \boldsymbol{\nabla}_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^{a+1}; \Theta_i^k) \boldsymbol{\Gamma}_i^k + \boldsymbol{W}_i^k \boldsymbol{\nabla}_{\boldsymbol{\Gamma}} \mathcal{L}(\boldsymbol{X}_\pi^{a+1}; \Theta_i^k), \\
\boldsymbol{e}_{(a,b)}^{(i,k)} &\triangleq \frac{\tilde{\boldsymbol{g}}_a^{(i,k)} - \boldsymbol{g}_a^b}{\eta_k},
\end{aligned}$$

for indices (a, b) satisfying $(i, k) \leq (a, b) \leq (i - 1, k + 1)$.

This way, the accumulation of updates on $\boldsymbol{M}$ from iteration (i, k) to $(i - 1, k + 1)$ can be represented as

$$\begin{aligned}
\boldsymbol{M}_i^{k+1} &= \boldsymbol{M}_i^k - \eta_k \sum_{j=i}^{m-1} \boldsymbol{g}_j^k + \eta_k^2 \sum_{j=i}^{m-1} \boldsymbol{q}_j^k - \eta_{k+1} \sum_{j=0}^{i-1} \boldsymbol{g}_j^{k+1} + \eta_{k+1}^2 \sum_{j=0}^{i-1} \boldsymbol{q}_j^{k+1} \\
&= \boldsymbol{M}_i^k - \eta_k \sum_{(a,b)=(i,k)}^{(i-1,k+1)} \boldsymbol{g}_a^b + \eta_k^2 \sum_{(a,b)=(i,k)}^{(i-1,k+1)} \boldsymbol{q}_a^b \\
&\quad - (\eta_{k+1} - \eta_k) \sum_{j=0}^{i-1} \boldsymbol{g}_j^{k+1} + (\eta_{k+1}^2 - \eta_k^2) \sum_{j=0}^{i-1} \boldsymbol{q}_j^{k+1} \\
&= \boldsymbol{M}_i^k - \eta_k \tilde{\boldsymbol{g}}^{(i,k)} + \eta_k^2 \sum_{(a,b)=(i,k)}^{(i-1,k+1)} \left(\boldsymbol{e}_{(a,b)}^{(i,k)} + \boldsymbol{q}_a^b \right) + \eta_k^2 \boldsymbol{s}_{(i,k+1)}^{(i,k)}
\end{aligned} \tag{25}$$

where in the last equality we used $\boldsymbol{g}_a^b = \tilde{\boldsymbol{g}}_a^{(i,k)} - \eta_k \boldsymbol{e}_{(a,b)}^{(i,k)}$ and also defined the accumulated signal $\tilde{\boldsymbol{g}}^{(i,k)}$ (a redefinition of $\tilde{\boldsymbol{g}}^k$ from Equation (16)) and the stepsize noise $\boldsymbol{s}_{(i,k+1)}^{(i,k)}$ as

$$\tilde{\boldsymbol{g}}^{(i,k)} \triangleq \sum_{(a,b)=(i,k)}^{(i-1,k+1)} \tilde{\boldsymbol{g}}_a^{(i,k)} = \boldsymbol{\nabla}_{\boldsymbol{W}} \mathcal{L}_\pi(\Theta_i^k) \boldsymbol{\Gamma}_i^k + \boldsymbol{W}_i^k \boldsymbol{\nabla}_{\boldsymbol{\Gamma}} \mathcal{L}_\pi(\Theta_i^k), \tag{26}$$

$$\boldsymbol{s}_{(i,k+1)}^{(i,k)} \triangleq -\frac{\eta_{k+1} - \eta_k}{\eta_k^2} \sum_{j=0}^{i-1} \boldsymbol{g}_j^{k+1} + \frac{\eta_{k+1}^2 - \eta_k^2}{\eta_k^2} \sum_{j=0}^{i-1} \boldsymbol{q}_j^{k+1}. \tag{27}$$

As a sanity check, we can quickly see that the stepsize noise $\boldsymbol{s}_{(i,k+1)}^{(i,k)}$ is zero if $i = 0$.

Now notice that Equation (25) can be thought of as a generalization of the per-epoch update (Equation (17)) originally obtained for $i = 0$. For now, suppose we ignore the last term of Equation (25) involving the stepsize noise $\boldsymbol{s}_{(i,k+1)}^{(i,k)}$. Then, if we carry out the above analysis for bounding the remaining terms in Equation (25), there is no difference in the argument up to reindexing; we can consider this as using η_k for the stepsize *even for the iterates* (a, b) *where* $b = k + 1$. In particular, the lemmas of the previous sections all hold up to reindexing notation.

Therefore, it now suffices to show that the stepsize noise $\boldsymbol{s}_{(i,k+1)}^{(i,k)}$ is of the same order as the other noise terms; in particular, $\left\|\boldsymbol{s}_{(i,k+1)}^{(i,k)}\right\|_F = O(1)$.

Lemma A.1.19. *Assume* $L[j, k+1]$ *and* $D[j, k+1]$ *hold for all* $j \le i - 1$. *Suppose that*

$$\eta_k = O\left(\frac{1}{k^\beta}\right),$$

for some $1/2 < \beta < 1$. *Then the stepsize noise*

$$\left\|\boldsymbol{s}_{(i,k+1)}^{(i,k)}\right\|_F = O(1).$$

Proof. Since $D[j, k+1]$ holds, Lemma A.1.16 demonstrates that $\left\|\boldsymbol{q}_j^{k+1}\right\|_2 = O(1)$. On the other hand, Corollaries A.1.8 and A.1.10 show that $\left\|\boldsymbol{g}_j^{k+1}\right\|_2 = O(1)$. Since $\eta_k = O(1/k^\beta)$ and $\beta \le 1$, we have $\eta_{k+1} - \eta_k = O(1/k^{\beta+1}) = O(\eta_k^2)$. Similarly $\eta_{k+1}^2 - \eta_k^2 = O(1/k^{2\beta+1}) = O(\eta_k^3)$. Plugging these into Equation (27), we conclude that $\left\|\boldsymbol{s}_{(i,k+1)}^{(i,k)}\right\|_F = O(1)$, as desired. □

Composite noise term is negligible. Now that we have formally defined the stepsize noise that arise for $i > 0$, we also redefine the composite noise term $\boldsymbol{r}^k$ (Equation (19)) originally defined for $i = 0$. The updated definition is simply

$$\boldsymbol{r}^{(i,k)} \triangleq \sum_{(a,b)=(i,k)}^{(i-1,k+1)} \left(\boldsymbol{e}_{(a,b)}^{(i,k)} + \boldsymbol{q}_a^b\right) + \boldsymbol{s}_{(i,k+1)}^{(i,k)}, \tag{28}$$

which allows us to rewrite the epoch update spelled out in Equation (25) as

$$\boldsymbol{M}_i^{k+1} = \boldsymbol{M}_i^k - \eta_k \tilde{\boldsymbol{g}}^{(i,k)} + \eta_k^2 \boldsymbol{r}^{(i,k)}., \tag{29}$$

which is a generalization of Equation (18).

It is left to show formally that the composite noise term $\boldsymbol{r}^{(i,k)}$ defined in Equation (28), obtained from combining the mismatched gradient noise terms $\boldsymbol{q}_a^b$, the path dependent noise $\boldsymbol{e}_{(a,b)}^{(i,k)}$ for $(i, k) \le (a, b) \le (i-1, k+1)$, and the stepsize noise $\boldsymbol{s}_{(i,k+1)}^{(i,k)}$, is indeed $O(1)$.

Proposition A.1.20 (Composite noise term)**.** *Suppose* $L[a, b]$ *and* $D[a, b]$ *hold for* $(i, k) \le (a, b) \le (i-1, k+1)$, *and* $\eta_k = O(1/k^\beta)$ *for some* $1/2 < \beta < 1$. *Then the composite noise term* $\boldsymbol{r}^{(i,k)}$ *satisfies*

$$\left\|\boldsymbol{r}^{(i,k)}\right\|_F \le \mathrm{poly}(m, C_{\mathrm{w}}, C_L, \left\|\overline{\boldsymbol{X}}_\pi\right\|_F) + O(\eta_k).$$

Proof. Combining Lemmas A.1.16, A.1.18 and A.1.19, taking care to analyze the constants (which are all $\mathrm{poly}(m, C_{\mathrm{w}}, C_L, \|\overline{\boldsymbol{X}}_\pi\|_F)$) hidden by the big–$O$ notation, yields the desired result. □

A.1.6. ACCUMULATED LOSS UPDATE

In this section, we formally account for the noise terms and prove that an accumulated loss inequality holds. More precisely, we can use the gradient update spelled out in Equation (29) and the noise bounds in Appendix A.1.5 to obtain a single epoch loss update. In other words, this section prove the inductive step that the hypotheses $R[i, k+1]$ and $L[i, k+1]$ holds. For the sake of simplicity, in this section we focus on the case $i = 0$. However, as discussed above, the case $i > 0$ only adds a negligible stepsize error and the same arguments go through.

We start with the following proposition, which proves the hypothesis $R[0, k+1]$.

Proposition A.1.21. *Assume that $L[j, k]$ and $D[j, k]$ hold for all $j < m$. Consider optimizing the linear+BN network with stepsize satisfying*

$$\eta_k \leq \frac{1}{2k^\beta},$$

for $1/2 < \beta < 1$.

Then

$$\mathcal{L}_\pi(\boldsymbol{M}_0^{k+1}) - \mathcal{L}_\pi^* \leq \Big(1 - \frac{\alpha_\pi \eta_k}{2}\Big)(\mathcal{L}_\pi(\boldsymbol{M}_0^k) - \mathcal{L}_\pi^*) + \mathrm{poly}(m, C_{\mathrm{w}}, C_L, \|\overline{\boldsymbol{X}}_\pi\|_F)\eta_k^2, \tag{30}$$

where the $\mathrm{poly}(m, C_{\mathrm{w}}, C_L, \|\overline{\boldsymbol{X}}_\pi\|_F)$ term is independent of k and has constant degree.

Proof. First, we use the G_π-smoothness of $\mathcal{L}_\pi$ with respect to $\boldsymbol{M}$ guaranteed by Lemma A.1.2 to obtain

$$\mathcal{L}_\pi(\boldsymbol{M}_0^{k+1}) - \mathcal{L}_\pi(\boldsymbol{M}_0^k) \leq \langle \boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}_\pi(\boldsymbol{M}_0^k), \boldsymbol{M}_0^{k+1} - \boldsymbol{M}_0^k \rangle_F + \frac{G_\pi}{2}\|\boldsymbol{M}_0^{k+1} - \boldsymbol{M}_0^k\|_F^2.$$

Using the gradient update Equation (18) and Cauchy-Schwarz, we can upper bound the RHS by

$$\begin{aligned}
&\langle \boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}_\pi(\boldsymbol{M}_0^k), -\eta_k \tilde{\boldsymbol{g}}^k + \eta_k^2 \boldsymbol{r}^k \rangle_F + \frac{G_\pi}{2}\|\boldsymbol{M}_0^{k+1} - \boldsymbol{M}_0^k\|_F^2 \\
&\leq -\eta_k \langle \boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}_\pi(\boldsymbol{M}_0^k), \tilde{\boldsymbol{g}}^k \rangle_F + \eta_k^2 \|\boldsymbol{r}^k\|_F \|\boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}_\pi(\boldsymbol{M}_0^k)\|_F + \frac{G_\pi}{2}\|\boldsymbol{M}_0^{k+1} - \boldsymbol{M}_0^k\|_F^2.
\end{aligned}$$

Next, we can use Lemma A.1.17, with $t = m$, together with the inequality $(a+b)^2 \leq 2a^2 + 2b^2$, to obtain an upper bound of

$$\begin{aligned}
&\leq -\eta_k \langle \boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}_\pi(\boldsymbol{M}_0^k), \tilde{\boldsymbol{g}}^k \rangle_F + \eta_k^2 \|\boldsymbol{r}^k\|_F \|\boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}_\pi(\boldsymbol{M}_0^k)\|_F \\
&\quad + \frac{G_\pi}{2}(4m\eta_k^2 C_{\mathrm{w}}^4 C_L \|\overline{\boldsymbol{X}}_\pi\|_F^2 + \eta_k^4 C_{\mathrm{w}}^4 C_L^2 \|\overline{\boldsymbol{X}}_\pi\|_F^4).
\end{aligned}$$

Then, because the inductive hypotheses apply we can apply Corollary A.1.10 to bound gradients, Corollary A.1.15 to bound the inner product $\langle \boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}_\pi(\boldsymbol{M}_0^k), \tilde{\boldsymbol{g}}^k \rangle_F$. Moreover, since $\eta_k = O(1/k^\beta)$, we can use Proposition A.1.20 to bound $\|\boldsymbol{r}^k\|_F$. This yields an upper bound of

$$\begin{aligned}
&-\frac{\eta_k}{2}\|\boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}_\pi(\boldsymbol{M}_0^k)\|_F^2 + \eta_k^2 \|\boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}_\pi(\boldsymbol{M}_0^k)\|_F \|\boldsymbol{r}^k\|_F + \mathrm{poly}(m, C_{\mathrm{w}}, C_L, \|\overline{\boldsymbol{X}}_\pi\|_F)\eta_k^2 \\
&\quad \leq \Big(-\frac{\eta_k}{2} + \frac{\eta_k^2}{2}\Big)\|\boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}_\pi(\boldsymbol{M}_0^k)\|_F^2 + \mathrm{poly}(m, C_{\mathrm{w}}, C_L, \|\overline{\boldsymbol{X}}_\pi\|_F)\eta_k^2 \\
&\quad \leq -\frac{\eta_k}{4}\|\boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}_\pi(k)\|_F^2 + \mathrm{poly}(m, C_{\mathrm{w}}, C_L, \|\overline{\boldsymbol{X}}_\pi\|_F)\eta_k^2
\end{aligned}$$

In the second line, we have used $ab \leq \frac{1}{2}(a^2 + b^2)$, and throughout, we have used the assumption $\eta_k \leq \frac{1}{2}$ to reduce higher order terms of η_k.

Putting it together, we find that

$$\mathcal{L}_\pi(\boldsymbol{M}_0^{k+1}) - \mathcal{L}_\pi(\boldsymbol{M}_0^k) \leq -\frac{\eta_k}{4}\|\boldsymbol{\nabla}_{\boldsymbol{M}} \mathcal{L}_\pi(\boldsymbol{M}_0^k)\|_F^2 + \mathrm{poly}(m, C_{\mathrm{w}}, C_L, \|\overline{\boldsymbol{X}}_\pi\|_F)\eta_k^2$$

We now use α_π–strong convexity with respect to $\boldsymbol{M}$ (and hence α_π-PŁ) guaranteed by Lemma A.1.3 to obtain

$$\mathcal{L}_\pi(\boldsymbol{M}_0^{k+1}) - \mathcal{L}_\pi^* \leq \Big(1 - \frac{\alpha_\pi \eta_k}{2}\Big)(\mathcal{L}_\pi(\boldsymbol{M}_0^k) - \mathcal{L}_\pi^*) + \mathrm{poly}(m, C_{\mathrm{w}}, C_L, \left\|\overline{\boldsymbol{X}}_\pi\right\|_F)\eta_k^2.$$

□

One consequence of Proposition A.1.21 is that if the stepsize η_k is small enough, we can guarantee that the loss decreases from $\mathcal{L}_\pi(\boldsymbol{M}_i^k)$ to $\mathcal{L}_\pi(\boldsymbol{M}_i^{k+1})$.

Next, from Proposition A.1.21, we can prove the other inductive hypothesis, namely $L[0, k+1]$.

Corollary A.1.22. *Suppose $L[j,k]$ and $D[j,k]$ hold for all $j < m$ and the stepsize satisfies*

$$\eta_k \leq \frac{1}{k^\beta} \min\left\{\frac{1}{2}, \frac{\alpha_\pi C_L}{\mathrm{poly}(m, C_{\mathrm{w}}, C_L, \left\|\overline{\boldsymbol{X}}_\pi\right\|_F)}\right\},$$

for some $1/2 < \beta < 1$.

Then $L[0, k+1]$ holds, i.e.

$$\mathcal{L}_\pi(\boldsymbol{M}_0^{k+1}) \leq C_L.$$

Proof. Since $\eta_k \leq \frac{1}{2k^\beta}$, we can apply Proposition A.1.21 to conclude that Equation (30) holds. Then, for the bound $\mathcal{L}_\pi(\boldsymbol{M}_0^{k+1}) \leq C_L$ to hold, it suffices to show that

$$\Big(1 - \frac{\alpha_\pi \eta_k}{2}\Big)C_L + \mathrm{poly}(m, C_{\mathrm{w}}, C_L, \left\|\overline{\boldsymbol{X}}_\pi\right\|_F)\eta_k^2 \leq C_L.$$

Equivalently,

$$\mathrm{poly}(m, C_{\mathrm{w}}, C_L, \left\|\overline{\boldsymbol{X}}_\pi\right\|_F)\eta_k \leq \frac{\alpha_\pi}{2} C_L.$$

Clearly this holds for the stated assumption on η_k. □

Finally, we show that by inductively unrolling the inequality in Proposition A.1.21, we can show that $\mathcal{L}_\pi(\boldsymbol{M}_i^k)$ converges to $\mathcal{L}_\pi^*$ at a sublinear rate.

Proposition A.1.23. *Assume we are in the same setup as Proposition A.1.21. Suppose that the stepsize satisfies*

$$\eta_k = \frac{c}{k^\beta},$$

for some absolute constant c such that $c \leq \min\left\{\frac{1}{2}, \frac{2}{\alpha_\pi}\right\}$ and $1/2 < \beta < 1$. Further suppose that $R[0,b]$ holds for every $b \in [k+1]$. Then if $\beta < 1$ we have

$$\mathcal{L}_\pi(\Theta_0^{k+1}) - \mathcal{L}_\pi^* \leq (\mathcal{L}_\pi(\Theta_0^1) - \mathcal{L}_\pi^*)\exp\left(\frac{c\alpha_\pi}{2(1-\beta)}(2 - k^{1-\beta})\right) + \frac{c^2\,\mathrm{poly}(m, C_{\mathrm{w}}, C_L, \left\|\overline{\boldsymbol{X}}_\pi\right\|_F)\log k}{k^\beta},$$

Proof. Note that by inspecting the proof of Proposition A.1.21 and Proposition A.1.20, the term $\mathrm{poly}(m, C_{\mathrm{w}}, C_L, \left\|\overline{\boldsymbol{X}}_\pi\right\|_F)$ has no dependence on k. So for simplicity we will assume that this term is bounded by some absolute constant A. Since $\eta_k \leq \frac{1}{2k^\beta}$, Proposition A.1.21 implies that

$$\mathcal{L}_\pi(\boldsymbol{M}_0^{k+1}) - \mathcal{L}_\pi^* \leq \Big(1 - \frac{\alpha_\pi \eta_k}{2}\Big)(\mathcal{L}_\pi(\boldsymbol{M}_0^k) - \mathcal{L}_\pi^*) + A\eta_k^2 \tag{31}$$

We can unroll the recurrence to obtain

$$\mathcal{L}_\pi(\boldsymbol{M}_0^{k+1}) - \mathcal{L}_\pi^* \leq (\mathcal{L}_\pi(\boldsymbol{M}_0^1) - \mathcal{L}_\pi^*)\prod_{t=1}^k \Big(1 - \frac{\alpha_\pi \eta_t}{2}\Big) + A\sum_{t=1}^k \eta_t^2 \left(\prod_{j=t+1}^k \Big(1 - \frac{\alpha_\pi \eta_j}{2}\Big)\right). \tag{32}$$

We have for any $c_t \leq 1$ that

$$\prod_{t=a}^{b}(1-c_t) \leq \exp\left(\sum_{t=a}^{b}\log(1-c_t)\right) \leq \exp\left(-\sum_{t=a}^{b}c_t\right),$$

where we have used $\log(1-x) \leq -x$ for $x \leq 1$. For $1/2 < \beta < 1$ we have

$$\sum_{t=a}^{b}\frac{1}{t^\beta} \geq \int_a^b \frac{1}{t^\beta}dt = \frac{b^{1-\beta}-a^{1-\beta}}{1-\beta}.$$

Hence, since we assumed $\eta_k = \frac{c}{k^\beta}$, and $\frac{\alpha_\pi c}{2} \leq 1$, we have

$$\prod_{t=a}^{b}\left(1-\frac{\alpha_\pi\eta_t}{2}\right) \leq \exp\left(-\frac{c\alpha_\pi}{2(1-\beta)}(b^{1-\beta}-a^{1-\beta})\right)$$

Now we can bound

$$A\sum_{t=1}^{k}\eta_t^2\left(\prod_{j=t+1}^{k}\left(1-\frac{\alpha_\pi\eta_j}{2}\right)\right) \leq \frac{c^2A}{k^{2\beta}} + \sum_{t=1}^{k-1}\frac{c^2A}{t^{2\beta}}\exp\left(-\frac{c\alpha_\pi}{2(1-\beta)}(k^{1-\beta}-(t+1)^{1-\beta})\right)$$

Define $T \triangleq k - Ck^\beta\log k$, where $C > 0$ is an absolute constant to be picked later. We can split up the sum into $t < T$ and $t \geq T$. For the terms $t < T$ we can use concavity to deduce that

$$k^{1-\beta}-(t+1)^{1-\beta} \geq (1-\beta)(k-t-1)k^{-\beta} \geq \Theta(\log k),$$

where the constant hidden in $\Theta(\log k)$ increases with C. Hence we pick C so that for $t < T$ we have

$$\frac{c\alpha_\pi}{2(1-\beta)}(k^{1-\beta}-(t+1)^{1-\beta}) \geq \beta\log k.$$

Then,

$$\sum_{t<T}\frac{c^2A}{t^{2\beta}}\exp\left(-\frac{c\alpha_\pi}{2(1-\beta)}(k^{1-\beta}-(t+1)^{1-\beta})\right) \leq \exp(-\beta\log k)\sum_{t<T}\frac{c^2A}{t^{2\beta}} \leq O\left(\frac{c^2A}{k^\beta}\right).$$

On the other hand, for the terms $t \geq T$ we can naively bound the exponential term by 1 and obtain

$$\sum_{t\geq T}\frac{c^2A}{t^{2\beta}}\exp\left(-\frac{c\alpha_\pi}{2(1-\beta)}(k^{1-\beta}-(t+1)^{1-\beta})\right) \leq \sum_{t\geq T}\frac{c^2A}{t^{2\beta}} \leq \Theta(k^\beta\log k)\frac{c^2A}{T^{2\beta}} \leq \Theta\left(\frac{c^2A\log k}{k^\beta}\right).$$

Hence we have that

$$\mathcal{L}_\pi(\boldsymbol{M}_0^{k+1}) - \mathcal{L}_\pi^* \leq (\mathcal{L}_\pi(\boldsymbol{M}_0^1) - \mathcal{L}_\pi^*)\exp\left(\frac{c\alpha_\pi}{2(1-\beta)}(2-k^{1-\beta})\right) + \Theta\left(\frac{c^2A\log k}{k^\beta}\right), \tag{33}$$

and the inequality in the proposition statement holds by recalling that $A = \mathrm{poly}(m, C_{\mathrm{w}}, C_L, \left\|\overline{\boldsymbol{X}}_\pi\right\|_F)$. □

A.1.7. BOUNDING APPROXIMATE INVARIANCES

Armed with Corollaries A.1.8 and A.1.10, we slog through the arduous task of inductively bounding the approximate invariance. As a reminder, these corollaries tell us that assuming the inductive hypotheses $L[j,k]$ and $D[j,k]$ hold, all weight norms and losses for iterate (j,k) can be bounded by uniform constants.

Lemma A.1.24. *Suppose $L[j,k]$ and $D[j,k]$ hold for some $j < m$. We have*

$$\left\|\boldsymbol{D}_{j+1}^k - \boldsymbol{D}_j^k\right\|_2 \le 2C_{\mathrm{w}}^2 C_L \left\|\mathsf{BN}(\boldsymbol{X}_\pi^{j+1})\right\|_2^2 \eta_k^2.$$

Hence, if $D[t,k]$ holds for all $t \le j$, we have

$$\left\|\boldsymbol{D}_{j+1}^k\right\|_2 \le 2C_{\mathrm{w}}^2 C_L \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^2 \sum_{t=1}^k \eta_t^2.$$

Proof. We have

$$\begin{aligned}(\boldsymbol{W}_{j+1}^k)^\top \boldsymbol{W}_{j+1}^k &= (\boldsymbol{W}_j^k - \eta_k \nabla_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k))^\top (\boldsymbol{W}_j^k - \eta_k \nabla_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k)) \\ &= (\boldsymbol{W}_j^k)^\top \boldsymbol{W}_j^k - \eta_k \left[\nabla_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k)^\top \boldsymbol{W}_j^k + (\boldsymbol{W}_j^k)^\top \nabla_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k)\right] \\ &\quad + \eta_k^2 [\nabla_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k)^\top \nabla_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k)]\end{aligned}$$

Similarly, we have

$$\begin{aligned}(\boldsymbol{\Gamma}_{j+1}^k)^2 &= (\boldsymbol{\Gamma}_j^k - \eta_k \nabla_{\boldsymbol{\Gamma}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k))(\boldsymbol{\Gamma}_j^k - \eta_k \nabla_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k)) \\ &= (\boldsymbol{\Gamma}_j^k)^2 - \eta_k \left[\nabla_{\boldsymbol{\Gamma}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k) \boldsymbol{\Gamma}_j^k + \boldsymbol{\Gamma}_j^k \nabla_{\boldsymbol{\Gamma}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k)\right] \\ &\quad + \eta_k^2 [\nabla_{\boldsymbol{\Gamma}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k) \nabla_{\boldsymbol{\Gamma}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k)]\end{aligned}$$

The gradient invariance in Fact A.1.1 cancels out the η_k term in $\boldsymbol{D}_{j+1}^k = \boldsymbol{I} + \operatorname{diag}((\boldsymbol{W}_{j+1}^k)^\top \boldsymbol{W}_{j+1}^k - (\boldsymbol{\Gamma}_{j+1}^k)^2$. Hence, if we take the operator norm of $\boldsymbol{D}_{j+1}^k - \boldsymbol{D}_j^k$ and use Lemma A.1.5, we can ignore the diagonal operator. Then, since the inductive hypotheses hold, we can apply the inductive gradient bound (Corollary A.1.10) to obtain

$$\begin{aligned}\left\|\boldsymbol{D}_{j+1}^k - \boldsymbol{D}_j^k\right\|_2 &\le \eta_k^2 \left[\left\|\nabla_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k)\right\|_2^2 + \left\|\nabla_{\boldsymbol{\Gamma}} \mathcal{L}(\boldsymbol{X}_\pi^{j+1}; \Theta_j^k)\right\|_2^2\right] \\ &\le 2C_{\mathrm{w}}^2 C_L \left\|\mathsf{BN}(\boldsymbol{X}_\pi^{j+1})\right\|_2^2 \eta_k^2.\end{aligned}$$

To conclude, we apply triangle inequality and the inductive hypothesis on $D[t,k]$, yielding

$$\begin{aligned}\left\|\boldsymbol{D}_{j+1}^k\right\|_2 &\le \left\|\boldsymbol{D}_0^k\right\|_2 + 2C_{\mathrm{w}}^2 C_L \eta_k^2 \sum_{t=0}^j \left\|\mathsf{BN}(\boldsymbol{X}_\pi^{t+1})\right\|_2^2 \\ &\le \left\|\boldsymbol{D}_0^k\right\|_2 + 2C_{\mathrm{w}}^2 C_L \eta_k^2 \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^2 \\ &\le 2C_{\mathrm{w}}^2 C_L \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^2 \sum_{t=1}^k \eta_t^2.\end{aligned}$$

□

As a corollary, we see that for stepsizes of the form $\eta_k = c/k^\beta$ for $1/2 < \beta < 1$, we can select c to guarantee that $\left\|\boldsymbol{D}_j^k\right\|_2 \le 1/2$ for all (j,k).

Corollary A.1.25. *Assume that $D[t,k]$ holds for all $t \le j < m$ and $\eta_k = \frac{c}{k^\beta}$ for $1/2 < \beta < 1$. If*

$$c^2 \le \frac{1}{4(1 + \frac{1}{2\beta - 1}) C_{\mathrm{w}}^2 C_L \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^2},$$

then

$$\left\|\boldsymbol{D}_{j+1}^{k}\right\|_{2} \leq 2C_{\mathrm{w}}^{2}C_{L}\left\|\overline{\boldsymbol{X}}_{\pi}\right\|_{F}^{2}\sum_{t=1}^{k}\eta_{t}^{2} \leq \frac{1}{2}.$$

In other words, $D[j+1,k]$ holds.

Proof. Lemma A.1.24 implies that

$$\left\|\boldsymbol{D}_{j+1}^{k}\right\|_{2} \leq 2C_{\mathrm{w}}^{2}C_{L}\left\|\overline{\boldsymbol{X}}_{\pi}\right\|_{F}^{2}\sum_{t=1}^{k}\eta_{t}^{2} \leq 2C_{\mathrm{w}}^{2}C_{L}\left\|\overline{\boldsymbol{X}}_{\pi}\right\|_{F}^{2}\sum_{t=1}^{\infty}\eta_{t}^{2}.$$

For $1/2 < \beta < 1$, we have

$$\sum_{k=1}^{\infty}\frac{1}{k^{2\beta}} \leq 1+\int_{1}^{\infty}\frac{1}{t^{2\beta}}dt = 1+\frac{1}{1-2\beta}\,t^{1-2\beta}\Big|_{1}^{\infty} = 1+\frac{1}{2\beta-1}.$$

Hence, if

$$c^{2} \leq \frac{1}{4(1+\frac{1}{2\beta-1})C_{\mathrm{w}}^{2}C_{L}\left\|\overline{\boldsymbol{X}}_{\pi}\right\|_{F}^{2}},$$

then evidently $\left\|\boldsymbol{D}_{j+1}^{k}\right\|_{2} \leq \frac{1}{2}$, as desired. □

A.1.8. Completing the induction

With all the pieces in place, we formally state the theorem for SS convergence.

Theorem A.1.26 (Formal statement of convergence for SS)**.** *Let $\pi \in \mathbb{S}_n$ be such that Assumption 1(a) holds. Let $f(\cdot;\Theta) = \boldsymbol{W}\boldsymbol{\Gamma}\mathsf{BN}(\cdot)$ be a 2-layer linear+BN network initialized at $\Theta_0^1 = (\boldsymbol{W}_0^1, \boldsymbol{\Gamma}_0^1) = (\boldsymbol{0}, \boldsymbol{I})$. Consider optimizing f using SS with permutation π and decreasing stepsize*

$$\eta_k = \frac{1}{k^{\beta}}\cdot\min\left\{\frac{1}{2}, \frac{2}{\alpha_\pi}, \frac{\alpha_\pi C_L}{\mathrm{poly}(m, C_{\mathrm{w}}, C_L, \left\|\overline{\boldsymbol{X}}_{\pi}\right\|_F)}, \sqrt{\frac{1}{4(1+\frac{1}{2\beta-1})C_{\mathrm{w}}^{2}C_{L}\left\|\overline{\boldsymbol{X}}_{\pi}\right\|_{F}^{2}}}\right\}$$

for any $1/2 < \beta < 1$. Then the SS risk satisfies

$$\mathcal{L}_\pi(\Theta_0^{k+1}) - \mathcal{L}_\pi^* \leq \frac{\mathrm{poly}(m, d, C_L, \left\|\overline{\boldsymbol{X}}_{\pi}\right\|_F, \frac{1}{\sigma_{\min}(\overline{\boldsymbol{X}}_{\pi}^{\top})})\log k}{k^{\beta}}.$$

In particular, the SS risk converges to the global optimal risk $\mathcal{L}_\pi^*$.

Proof of Theorem A.1.26. We proceed by induction on the epoch. We restate the key inductive statements to prove:(with appropriate selection of η_k):

- $L[j,k]$: $\mathcal{L}_\pi(\Theta_j^k)$ stays bounded above by some uniform constant $C_L \geq \|\boldsymbol{Y}_\pi\|_F^2$ — this is the content of Corollary A.1.22.
- $R[j,k]$: $\mathcal{L}_\pi(\Theta_j^k)$ satisfies the per-epoch loss inequality — this is the content of Proposition A.1.21.
- $D[j,k]$: The approximate invariances stay bounded in norm away from 1. More precisely, $\left\|\boldsymbol{D}_{j}^{k}\right\|_{2} \leq 2C_{\mathrm{w}}^{2}C_{L}\left\|\overline{\boldsymbol{X}}_{\pi}\right\|_{F}^{2}\sum_{t=1}^{\infty}\eta_{t}^{2} \leq \frac{1}{2}$ — this is the content of Corollary A.1.25.

Notice that the assumptions on η_k exactly satisfy the hypotheses of Corollaries A.1.22 and A.1.25 and Proposition A.1.23.

The base cases follows from the initialization. Recall that we set

$$C_L = \max\left\{\mathcal{L}_\pi(\Theta_t^1) : 0 \leq t \leq m-1\right\}.$$

Since we only look at the loss values for the first epoch, C_L is indeed an absolute constant depending on π. So $L[j,1]$, as defined in Hypothesis 1, holds for all $j < m$. Next, we have from the initialization that $\boldsymbol{D}_0^1 = \mathbf{0}$. Then, applying Lemma A.1.24, we can conclude $D[j,1]$ also holds for all $j < m$, so Hypothesis 3 holds. Finally, there is no need to check Hypothesis 2 because it is only defined for $k > 1$.

Now our inductive hypotheses are that $L[j,k]$, $R[j,k]$, $D[j,k]$ hold for all $j < m$. For the inductive step, we want to prove $L[0,k+1]$, $R[0,k+1]$, $D[0,k+1]$. By construction of η_k, the hypotheses of Corollary A.1.22 is satisfied, so $L[0,k+1]$ holds. Moreover, the hypotheses of Proposition A.1.21 are satisfied, so $R[0,k+1]$ holds. Finally, the hypotheses of Lemma A.1.24 are satisfied, so that $D[0,k+1]$ holds.

As asserted earlier, the above argument is robust up to reindexing if we want to prove the statement for $i > 0$. In particular, all of the results of the previous section go through, as we showed in Proposition A.1.20 that the stepsize noise is negligible. Hence the induction is completed for all (i,k) and so the unrolled update equation in Proposition A.1.23 holds for all k. This gives the formal rate of convergence for the stated stepsize. In particular, we see that the SS risk $\mathcal{L}_\pi$ converges to its global minimum. □

Proof of Theorem 3.2.2. All we need to do is convert the stepsizes requirements. Examining the stepsize requirements in Theorem A.1.26, they depend on C_L, C_{w}, and $\left\|\overline{\boldsymbol{X}}_\pi\right\|_F$.

Now recall the definition of C_{w} in Corollary A.1.8:

$$C_{\mathrm{w}}^2 \le \frac{3}{2} + d^2\left(\frac{1}{2} + \frac{C_L^{1/2} + \|\boldsymbol{Y}_\pi\|_F}{\sigma_{\min}(\overline{\boldsymbol{X}}_\pi^\top)}\right).$$

Hence $C_{\mathrm{w}} = \mathrm{poly}(d, C_L, \|\boldsymbol{Y}_\pi\|_F, 1/\sigma_{\min}(\overline{\boldsymbol{X}}_\pi^\top))$. Finally, since $\|\boldsymbol{X}_\pi\|_F^2 = dn$, $\sigma_{\min}(\overline{\boldsymbol{X}}_\pi \overline{\boldsymbol{X}}_\pi^\top) = \sigma_{\min}(\overline{\boldsymbol{X}}_\pi^\top)^2$, and C_L is an absolute constant, we can convert the stepsize requirements into

$$\eta_k = \frac{1}{k^\beta} \cdot \min\left\{\frac{1}{2}, \frac{2}{\sigma_{\min}(\overline{\boldsymbol{X}}_\pi \overline{\boldsymbol{X}}_\pi^\top)}, \frac{\sqrt{2\beta-1}\,\mathrm{poly}(\sigma_{\min}(\overline{\boldsymbol{X}}_\pi^\top))}{\mathrm{poly}(n, d, \|\boldsymbol{Y}_\pi\|_F)}\right\},$$

which directly implies the stepsize requirements in Theorem 3.2.2. □

A.2. Proof of convergence for RR

In this section, we prove Theorem 3.2.3. With RR, we randomly resample permutation $\pi_k \in \mathbb{S}_n$ on epoch k. Hence, it is natural to seek a convergence bound in expectation. We briefly comment on the complications that arise in this setting.

Since we want to prove convergence an expectation, an inductive approach that controls the approximate invariances and loss evolution is complicated by the necessity for bounds on these quantities that are stronger than merely being in expectation. This is precisely why we need Assumption 3.

Additional notation. As introduced in Section 2, we can view RR as optimizing the risk

$$\mathcal{L}_{\mathrm{RR}}(\Theta) \triangleq \frac{1}{n!}\sum_{\pi \in \mathbb{S}_n} \mathcal{L}_\pi(\Theta)$$

via with-replacement SGD on an *epoch level* (i.e., $\mathcal{L}_\pi$ is sampled uniformly with replacement at every epoch), albeit with noise terms due to the shuffling algorithm. Motivated by the setup in Section 2, we can also write $\mathcal{L}_{\mathrm{RR}}$ in an equivalent form to $\mathcal{L}_{\mathrm{GD}}$ as follows:

$$\mathcal{L}_{\mathrm{RR}}(\Theta) \triangleq \frac{1}{n!}\sum_{\pi \in \mathbb{S}_n} \mathcal{L}(f(\boldsymbol{X}_\pi;\Theta), \boldsymbol{Y}_\pi) = \frac{1}{n!}\mathcal{L}(\boldsymbol{W}\boldsymbol{\Gamma}\mathsf{BN}_{\mathrm{RR}}(\boldsymbol{X}), \boldsymbol{Y}_{\mathrm{RR}}),$$

$$\text{where } \mathsf{BN}_{\mathrm{RR}}(\boldsymbol{X}) \triangleq \begin{bmatrix}\overline{\boldsymbol{X}}_{\pi_1} & \dots & \cdots & \overline{\boldsymbol{X}}_{\pi_{n!}}\end{bmatrix} \in \mathbb{R}^{d\times(n\cdot n!)},$$

$$\boldsymbol{Y}_{\mathrm{RR}} \triangleq \begin{bmatrix}\boldsymbol{Y}_{\pi_1} & \cdots & \boldsymbol{Y}_{\pi_{n!}}\end{bmatrix} \in \mathbb{R}^{p\times(n\cdot n!)}.$$

For notational convenience, we also write $\overline{\boldsymbol{X}}_{\mathrm{RR}} \triangleq \mathsf{BN}_{\mathrm{RR}}(\boldsymbol{X})$, and reiterate that overlines indicate the presence of batch normalization.

Just as in the SS case, we will abuse notation and refer to the RR risk function as a function of $\boldsymbol{M} = \boldsymbol{W}\boldsymbol{\Gamma}$ by writing $\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}) \triangleq \frac{1}{n!}\sum_{\pi\in\mathbb{S}_n}\mathcal{L}_\pi(\boldsymbol{M})$. Similarly, we will often refer to the gradient of the RR risk with respect to $\boldsymbol{M}$ as $\boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}) \triangleq \frac{1}{n!}\sum_{\pi\in\mathbb{S}_n}\boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_\pi(\boldsymbol{M})$. We will find it helpful to use the notation $\overline{\boldsymbol{X}}_{\max,2} \triangleq \arg\max_{\pi\in\mathbb{S}_n}\left\|\overline{\boldsymbol{X}}_\pi\right\|_2$. Similarly we will denote the maximum Frobenius norm batch normalized dataset by $\overline{\boldsymbol{X}}_{\max,F} \triangleq \arg\max_{\pi\in\mathbb{S}_n}\left\|\overline{\boldsymbol{X}}_\pi\right\|_F$. Furthermore, it follows from the unit variance constraint in the definition of BN that $\left\|\overline{\boldsymbol{X}}_{\max,2}\right\|_2 \le \left\|\overline{\boldsymbol{X}}_{\max,F}\right\|_F \le \sqrt{dn}$.

At a high level, the RR proof of convergence closely follows the SS proof of convergence. Indeed, most of the technical legwork has already been fleshed out in the SS case — most of the results port over immediately, taking care to replace π with π_k. However, we will be careful in accounting for where we need to deviate from the SS logic.

A.2.1. Checking optimization properties

We first check the smoothness and strong convexity property with respected to the merged matrix $\boldsymbol{M}$ that we heavily relied on for the proof of convergence for SS.

Fact A.2.1 (Smoothness of RR)**.** *Define*

$$G_{\mathrm{RR}} \triangleq \frac{1}{n!}\sum_{\pi\in\mathbb{S}_n} G_\pi.$$

Then $\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M})$ is G_{RR}-smooth with respect to $\boldsymbol{M}$.

Proof. The statement follows from combining Lemma A.1.2 with the fact that if f_i are G_i-smooth, then $\sum_{i=1}^n f_i$ is $\sum_{i=1}^n G_i$-smooth. □

As before, we cannot directly use a PŁ inequality on $\boldsymbol{W}$ or $\boldsymbol{\Gamma}$; we must instead bootstrap this from the strong convexity of the risk with respect to $\boldsymbol{M} = \boldsymbol{W}\boldsymbol{\Gamma}$.

Fact A.2.2 (Strong convexity)**.** *Suppose that Assumption 1(b) holds for some $\pi \in \mathbb{S}_n$. Then the loss function $\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}) = \frac{1}{n!}\sum_{\pi\in\mathbb{S}_n}\mathcal{L}_\pi(\boldsymbol{M})$ is α_{RR}-strongly convex with respect to $\boldsymbol{M}$ with $\alpha_{\mathrm{RR}} \triangleq \frac{1}{n!}\sum_\pi \sigma_{\min}(\overline{\boldsymbol{X}}_\pi\overline{\boldsymbol{X}}_\pi^\top) = \frac{1}{n!}\sum_\pi \alpha_\pi$.*

Proof. Take the Hessian of $\mathcal{L}_{\mathrm{RR}}$ with respect to $\mathrm{vec}(\boldsymbol{M})$ to obtain $\boldsymbol{\nabla}^2_{\mathrm{vec}(\boldsymbol{M})}\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}) = \frac{1}{n!}\sum_{\pi\in\mathbb{S}_n}\overline{\boldsymbol{X}}_\pi\overline{\boldsymbol{X}}_\pi^\top \otimes \boldsymbol{I}$. Hence we have $\boldsymbol{\nabla}^2_{\mathrm{vec}(\boldsymbol{M})}\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}) \succeq \frac{1}{n!}\sum_\pi \sigma_{\min}(\overline{\boldsymbol{X}}_\pi\overline{\boldsymbol{X}}_\pi^\top)$. Since we assumed Assumption 1(b) holds, the sum is strictly positive, so $\mathcal{L}_{\mathrm{RR}}$ is indeed strongly convex. □

A.2.2. Proof sketch of RR convergence

We first state the modified inductive hypothesis for the one-epoch risk update, which replaces $R[j,k]$.

Hypothesis 4. *For $k > 1$, the inductive hypothesis $S[k]$ states that*

$$\mathbb{E}_{\pi_{k-1}}[\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^k)|\mathcal{F}_{k-1}] - \mathcal{L}^*_{\mathrm{RR}} \le \left(1 - \frac{\alpha_{\mathrm{RR}}\eta_{k-1}}{2}\right)(\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^{k-1}) - \mathcal{L}^*_{\mathrm{RR}}) + \eta_{k-1}^2\,\mathrm{poly}(m, A_{\mathrm{w}}, A_L, \left\|\overline{\boldsymbol{X}}_{\max,F}\right\|_F),$$

where $\mathbb{E}_{\pi_{k-1}}$ denotes the expectation over random draws of the permutation π_{k-1}.

Proof sketch of Theorem 3.2.3. As before, we start by writing out the smoothness inequality with respect to $\boldsymbol{M}$:

$$\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^{k+1}) \le \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^k) + \left\langle \boldsymbol{\nabla}_{\boldsymbol{M}}\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^k), \boldsymbol{M}_0^{k+1} - \boldsymbol{M}_0^k\right\rangle + \frac{G_{\mathrm{RR}}}{2}\left\|\boldsymbol{M}_0^{k+1} - \boldsymbol{M}_0^k\right\|^2. \tag{34}$$

Next, we have the same gradient update

$$\boldsymbol{M}_0^{k+1} = \boldsymbol{M}_0^k - \eta_k\tilde{\boldsymbol{g}}^k + \eta_k^2\boldsymbol{r}^k,$$

but now all quantities involving π turn into π_k. For example, we redefine

$$\tilde{\boldsymbol{g}}^k \triangleq \sum_{t=0}^{m-1}\tilde{\boldsymbol{g}}_t^k = \boldsymbol{\nabla}_{\boldsymbol{W}}\mathcal{L}_{\pi_k}(\boldsymbol{M}_0^k)\boldsymbol{\Gamma}_0^k + \boldsymbol{W}_0^k\boldsymbol{\nabla}_{\boldsymbol{\Gamma}}\mathcal{L}_{\pi_k}(\boldsymbol{M}_0^k).$$

Since we want to prove convergence in expectation, it is standard to consider the natural filtration $\mathcal{F}_k$ of which permutations we have picked up to (but not including) epoch k. Formally, $\mathcal{F}_k = \sigma(\pi_1, \ldots, \pi_{k-1})$, where $\sigma(Z)$ denotes the σ-algebra generated by the random variable Z.

Noting the identity

$$\mathbb{E}_\pi[\nabla_{\boldsymbol{M}} \mathcal{L}_\pi(\boldsymbol{M}_0^k)] = \nabla_{\boldsymbol{M}} \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^k),$$

it follows that if we can apply Corollary A.1.15, then

$$\mathbb{E}_{\pi_k}\left[\left\langle \nabla_{\boldsymbol{M}} \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^k), \tilde{\boldsymbol{g}}^k \right\rangle_F\right] = \left\langle \nabla_{\boldsymbol{M}} \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^k), \mathbb{E}_{\pi_k}[\tilde{\boldsymbol{g}}^k] \right\rangle_F \geq \frac{1}{2}\left\| \nabla_{\boldsymbol{M}} \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^k)\right\|_F^2.$$

Hence, we can follow the same argument for upper bounding the smoothness inequality for SS and take a conditional expectation over π_k conditioned on $\mathcal{F}_k$. Assuming for now that the weight norms are bounded by some absolute constant A_{w} and the relevant losses are bounded by an absolute constant A_L, this yields

$$\mathbb{E}_{\pi_k}[\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^{k+1})|\mathcal{F}_k] \leq \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^k) - \frac{\eta_k}{4}\left\| \nabla_{\boldsymbol{M}} \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^k)\right\|_F^2 + \eta_k^2\, \mathbb{E}_{\pi_k}[\mathrm{poly}(m, A_{\mathrm{w}}, A_L, \|\boldsymbol{X}_{\pi_k}\|_F)].$$

Noting that we can upper bound $\|\boldsymbol{X}_{\pi_k}\|_F$ uniformly by $\left\|\overline{\boldsymbol{X}}_{\max,F}\right\|_F$, which does not depend on k, this shows that we have

$$\mathbb{E}_{\pi_k}[\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^{k+1})|\mathcal{F}_k] \leq \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^k) - \frac{\eta_k}{4}\left\| \nabla_{\boldsymbol{M}} \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^k)\right\|_F^2 + \eta_k^2\, \mathrm{poly}(m, A_{\mathrm{w}}, A_L, \left\|\overline{\boldsymbol{X}}_{\max,F}\right\|_F).$$

Next, α_{RR}–strong convexity yields

$$\mathbb{E}_{\pi_k}[\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^{k+1})|\mathcal{F}_k] - \mathcal{L}_{\mathrm{RR}}^* \leq \left(1 - \frac{\alpha_{\mathrm{RR}}\eta_k}{2}\right)(\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^k) - \mathcal{L}_{\mathrm{RR}}^*) + \eta_k^2\, \mathrm{poly}(m, A_{\mathrm{w}}, A_L, \left\|\overline{\boldsymbol{X}}_{\max,F}\right\|_F).$$

This is exactly the statement of $S[k+1]$. To proceed, we must fill in the following details. First, we must show that the relevant losses are bounded by A_L — see Corollary A.2.6. Then, we show that the weight norms are bounded by A_{w} — this is shown in Corollary A.2.7. Finally, we must show that $D[j,k]$ holds, i.e., bound the approximate invariances — this is the content of Corollary A.2.9. Once we address these technicalities, an inductive argument similar to the SS version proves the theorem. Note that the SS inductive hypotheses $L[j,k]$ and $R[j,k]$ are not active in this proof. □

A.2.3. Weight bounds

In this section we elucidate the connection between weight norms and the loss evolution. In particular, we show that bounds on the weight norms confer a bound on the loss function value. First, we state the following inequality, which follows from a quick application of Cauchy-Schwarz.

Fact A.2.3. *We have* $\frac{1}{n!}\sum_{\pi \in \mathbb{S}_n} (\mathcal{L}_\pi(\boldsymbol{M}))^{1/2} \leq \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M})^{1/2}$.

With this identity in hand, we derive the following corollaries about weight and gradient bounds.

Proposition A.2.4. *We have*

$$\|\nabla_{\boldsymbol{M}} \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M})\| \leq \left\|\overline{\boldsymbol{X}}_{\max,2}\right\|_2 \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M})^{1/2}.$$

Proof. We have $\|\nabla_{\boldsymbol{M}} \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M})\|_2 \leq \frac{1}{n!}\sum_\pi \|\nabla_{\boldsymbol{M}} \mathcal{L}_\pi(\boldsymbol{M})\|_2$ by the triangle inequality. Applying the individual gradient bounds in Lemma A.1.9 and uniformly bounding $\left\|\overline{\boldsymbol{X}}_\pi\right\|$ by $\left\|\overline{\boldsymbol{X}}_{\max,2}\right\|$, we see that

$$\|\nabla_{\boldsymbol{M}} \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M})\|_2 \leq \left\|\overline{\boldsymbol{X}}_{\max,2}\right\|_2 \cdot \frac{1}{n!}\sum_\pi \mathcal{L}_\pi^{1/2}(\boldsymbol{M}).$$

Applying Fact A.2.3 yields the desired result. □

As promised, we quantify the relationship between $\mathcal{L}_{\mathrm{RR}}$, $\mathcal{L}_\pi$, and $\|\boldsymbol{M}\|_2$ with the following proposition.

Proposition A.2.5. *Let* $\sigma_0 \triangleq \frac{1}{n!}\sum_\pi \sigma_{\min}(\overline{\boldsymbol{X}}_\pi^\top)$. *We have*

$$\frac{\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}) - 2\|\boldsymbol{Y}\|_F^2}{2\left\|\overline{\boldsymbol{X}}_{\max,F}\right\|_F^2} \le \|\boldsymbol{M}\|_2^2 \le \left(\frac{\|\boldsymbol{Y}\|_F + \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M})^{1/2}}{\sigma_0}\right)^2.$$

Similarly, for any $\pi \in \mathbb{S}_n$, *we have*

$$\frac{\mathcal{L}_{\pi}(\boldsymbol{M}) - 2\|\boldsymbol{Y}\|_F^2}{2\left\|\overline{\boldsymbol{X}}_{\pi}\right\|_F^2} \le \|\boldsymbol{M}\|_2^2 \le \left(\frac{\|\boldsymbol{Y}\|_F + \mathcal{L}_{\pi}(\boldsymbol{M})^{1/2}}{\sigma_{\min}(\overline{\boldsymbol{X}}_\pi^\top)}\right)^2.$$

Proof. First, we have

$$\begin{aligned}\frac{1}{n!}\sum_{\pi\in\mathbb{S}_n}\left\|\boldsymbol{M}\overline{\boldsymbol{X}}_\pi\right\|_2 &\le \|\boldsymbol{Y}\|_F + \frac{1}{n!}\sum_\pi \left\|\boldsymbol{Y}_\pi - \boldsymbol{M}\overline{\boldsymbol{X}}_\pi\right\|_F \\ &= \|\boldsymbol{Y}\|_F + \frac{1}{n!}\sum_\pi \mathcal{L}_\pi(\boldsymbol{M})^{1/2},\end{aligned}$$

where we have used in the first line the fact that $\|\boldsymbol{Y}_\pi\|_F = \|\boldsymbol{Y}\|_F$ for all π. Therefore by Fact A.2.3 and using $\left\|\boldsymbol{M}\overline{\boldsymbol{X}}_\pi\right\|_2 \ge \sigma_{\min}(\overline{\boldsymbol{X}}_\pi^\top)\|\boldsymbol{M}\|_2$, we find that

$$\frac{1}{n!}\sum_\pi \sigma_{\min}(\overline{\boldsymbol{X}}_\pi^\top)\|\boldsymbol{M}\|_2 \le \|\boldsymbol{Y}\|_F + \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M})^{1/2}.$$

It follows that

$$\|\boldsymbol{M}\|_2 \le \frac{\|\boldsymbol{Y}\|_F + \mathcal{L}_{\mathrm{RR}}(\boldsymbol{M})^{1/2}}{\sigma_0}.$$

For the other direction, note that

$$\mathcal{L}_\pi(\boldsymbol{M}) = \left\|\boldsymbol{Y}_\pi - \boldsymbol{M}\overline{\boldsymbol{X}}_\pi\right\|_F^2 \le 2\|\boldsymbol{Y}\|_F^2 + 2\|\boldsymbol{M}\|_2^2\left\|\overline{\boldsymbol{X}}_\pi\right\|_F^2.$$

Averaging over all $\pi \in \mathbb{S}_n$ gives us

$$\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}) \le 2\|\boldsymbol{Y}\|_F^2 + \frac{2\|\boldsymbol{M}\|_2^2}{n!}\sum_\pi \left\|\overline{\boldsymbol{X}}_\pi\right\|_F^2.$$

Uniformly bounding $\left\|\overline{\boldsymbol{X}}_\pi\right\|_F^2$ by $\left\|\overline{\boldsymbol{X}}_{\max,F}\right\|_F^2$ and rearranging yields the desired result.

The set of inequalities for π also follow by a similar argument. □

As a corollary of Proposition A.2.5, it follows from Assumption 3 that each of the losses $\mathcal{L}_\pi(\boldsymbol{M}_i^k)$ stay bounded by an absolute constant throughout training.

Corollary A.2.6 (Uniform bound on SS losses)**.** *Under Assumption 3, for every* $\pi \in \mathbb{S}_n$ *we have*

$$\mathcal{L}_\pi(\boldsymbol{M}_i^k) \le A_L,$$

where

$$A_L \triangleq 2\|\boldsymbol{Y}\|_F^2 + 2A_{\mathrm{RR}}^2\left\|\overline{\boldsymbol{X}}_{\max,F}\right\|_F^2.$$

Here, A_{RR} *was previously defined in Assumption 3.*

Finally, Assumption 3 implies the following inductive statement about the weight norms.

Corollary A.2.7 (Uniform bound on weight norms)**.** *Assume Assumption 3 and* $D[j,k]$ *holds. Then for RR, we have*

$$\max\left\{\left\|\boldsymbol{W}_i^k\right\|_2, \left\|\boldsymbol{\Gamma}_i^k\right\|_2\right\} \le A_{\mathrm{w}},$$

where

$$A_{\mathrm{w}}^2 \triangleq \frac{3}{2} + d^2\left(\frac{1}{2} + A_{\mathrm{RR}}\right).$$

Here, A_{RR} *was previously defined in Assumption 3.*

A.2.4. Bounding approximate invariances

In this section we formally bound the approximate invariances throughout RR training under Assumption 3. In particular, as a consequence of Corollaries A.2.6 and A.2.7, the following two claims follow almost directly upon inspection of the proofs of Lemma A.1.24 and Corollary A.1.25.

Lemma A.2.8. *Suppose $D[j,k]$ holds for some $j < m$. We have*

$$\left\|\boldsymbol{D}_{j+1}^k - \boldsymbol{D}_j^k\right\|_2 \le 2A_{\mathrm{w}}^2 A_L \left\|\mathsf{BN}(\boldsymbol{X}_{\pi_k}^{j+1})\right\|_2^2 \eta_k^2.$$

Hence, if $D[t,k]$ holds for all $t \le j$, we also have

$$\left\|\boldsymbol{D}_{j+1}^k\right\|_2 \le 2A_{\mathrm{w}}^2 A_L \left\|\overline{\boldsymbol{X}}_{\max,F}\right\|_F^2 \sum_{t=1}^k \eta_t^2.$$

Corollary A.2.9. *Assume that $D[t,k]$ holds for all $t \le j < m$ and $\eta_k = \frac{c}{k^\beta}$ for $1/2 < \beta < 1$. If*

$$c^2 \le \frac{1}{4(1+\frac{1}{2\beta-1})A_{\mathrm{w}}^2 A_L \left\|\overline{\boldsymbol{X}}_{\max,F}\right\|_F^2},$$

then

$$\left\|\boldsymbol{D}_{j+1}^k\right\|_2 \le 2A_{\mathrm{w}}^2 A_L \left\|\overline{\boldsymbol{X}}_{\max,F}\right\|_F^2 \sum_{t=1}^k \eta_t^2 \le \frac{1}{2}.$$

In other words, $D[j+1,k]$ holds.

A.2.5. Completing the proof

With the connection between the RR loss function and weight bounds in hand, we can complete the proof of convergence for RR. Finally, we formally state the RR convergence result.

Theorem A.2.10 (Formal statement of convergence for RR)**.** *Suppose Assumption 1(b) and Assumption 3 hold. Let $f(\cdot;\Theta) = \boldsymbol{W}\boldsymbol{\Gamma}\mathsf{BN}(\cdot)$ be a 2-layer linear+BN network initialized at $\Theta_0^1 = (\boldsymbol{W}_0^1, \boldsymbol{\Gamma}_0^1) = (\boldsymbol{0}, \boldsymbol{I})$. Consider optimizing f using RR with decreasing stepsize*

$$\eta_k = \frac{1}{k^\beta} \cdot \min\left\{\frac{1}{2}, \frac{2}{\alpha_{\mathrm{RR}}}, \sqrt{\frac{1}{4(1+\frac{1}{2\beta-1})A_{\mathrm{w}}^2 A_L \left\|\overline{\boldsymbol{X}}_{\max,F}\right\|_F^2}}\right\}$$

for any $1/2 < \beta < 1$. Then the RR risk satisfies

$$\mathbb{E}[\mathcal{L}_{\mathrm{RR}}(\Theta_0^{k+1})] - \mathcal{L}_{\mathrm{RR}}^* \le \frac{\mathrm{poly}(n, d, A_{\mathrm{RR}}, \|\boldsymbol{Y}\|_F)\log k}{k^\beta}.$$

In other words, the RR risk converges to the global optimal risk $\mathcal{L}_{\mathrm{RR}}^*$.

Proof of Theorem A.2.10. We inductively prove that $D[j,k]$ and $S[k]$ hold. There is no need to check $S[1]$ because Hypothesis 4 is defined for $k > 1$. The proof that the base cases $D[j,0]$ all hold follows the same proof as that in the SS case.

Now suppose for the sake of induction that $D[j,k]$ and $S[k]$ hold. We will show that $D[j+1,k]$ holds. Once we show that $D[j,k]$ holds for all $j < m$, we can then show that $S[k+1]$ holds.

In particular, by the assumption on η_k, Corollary A.2.9 implies that $D[j+1,k]$ holds. Hence by induction $D[j,k]$ holds for all $j < m$. We see that assuming Assumption 3 simplified the proof strategy significantly, as we did not have to go through the trouble of proving $L[j,k]$.

Next, let's understand what happens to the per-epoch loss bound in $S[k+1]$. Explicitly, we can follow the same steps as in the proof sketch — which only required $\eta_k \le \frac{c}{k^\beta}$ where $c \le \min\left\{\frac{1}{2}, \frac{2}{\alpha_{\mathrm{RR}}}\right\}$ — to see that

$$\mathbb{E}_{\pi_k}[\mathcal{L}_{\pi_k}(\boldsymbol{M}_0^{k+1})|\mathcal{F}_k] - \mathcal{L}_\pi^* \le \left(1 - \frac{\alpha_\pi \eta_k}{2}\right)(\mathcal{L}_{\mathrm{RR}}(\boldsymbol{M}_0^k) - \mathcal{L}_{\mathrm{RR}}^*) + \mathrm{poly}(m, A_{\mathrm{w}}, A_L, \left\|\overline{\boldsymbol{X}}_{\max,F}\right\|_F)\eta_k^2.$$

Indeed, under Assumption 3 and $D[j,k]$ for $j < m$, we can apply Corollaries A.2.6 and A.2.7 to rigorously bound all of the analogous noise terms a.s.. We can then follow the same argument as in Proposition A.1.23 to unroll the recurrence, using iterated expectation to obtain a total expectation in the end. We can thus conclude that with $\eta_k = \frac{c}{k^\beta}$ for $1/2 < \beta < 1$ and the constant c chosen as the theorem statement that

$$\begin{aligned}\mathbb{E}[\mathcal{L}_{\mathrm{RR}}(\Theta_0^{k+1})] - \mathcal{L}_{\mathrm{RR}}^* &\leq (\mathcal{L}_{\mathrm{RR}}(\Theta_0^1) - \mathcal{L}_\pi^*)\exp\left(\frac{c\alpha_{\mathrm{RR}}}{2(1-\beta)}(2-k^{1-\beta})\right) \\ &\quad + \frac{\mathrm{poly}(m, A_{\mathrm{w}}, A_L, \left\|\overline{\boldsymbol{X}}_{\max,F}\right\|_F)\log k}{k^\beta} \\ &\leq (\mathcal{L}_{\mathrm{RR}}(\Theta_0^1) - \mathcal{L}_\pi^*)\exp\left(\frac{c\alpha_{\mathrm{RR}}}{2(1-\beta)}(2-k^{1-\beta})\right) \\ &\quad + \frac{\mathrm{poly}(n, d, A_{\mathrm{RR}}, \|\boldsymbol{Y}\|_F)\log k}{k^\beta}.\end{aligned}$$

In the last line we used the fact that $A_L = \mathrm{poly}(A_{\mathrm{RR}}, \|\boldsymbol{Y}\|_F, \left\|\overline{\boldsymbol{X}}_{\max,F}\right\|_F)$ from Corollary A.2.6, $A_{\mathrm{w}} = \mathrm{poly}(d, A_{\mathrm{RR}})$ from Corollary A.2.7, and $\left\|\overline{\boldsymbol{X}}_{\max,F}\right\|_F^2 = dn$. The desired claim immediately follows.

We can also immediately see how the stepsize requirements match that of Theorem 3.2.3. □

B. Proofs for classification results

In this section we lay out the groundwork for formally proving our main results Theorems B.2.1 and B.3.1 about the separability decomposition of SS+BN and RR+BN (cf. Theorems 4.1.3 and 4.1.4). At a high level, we show that the separability decomposition is closely linked to the presence of monochromatic batches (Lemma B.1.1) and the dimensionality of the batch normalized dataset (Lemma B.1.2). In Appendix B.4, we formally characterize the optimal directions of linear+BN classifiers. We defer the proofs of the more technical lemmas to Appendix C.

Additional notation and setup

We lay out some additional notation that will aid in our discussion of classification. Division of two vectors should be interpreted in a coordinatewise fashion, so $\frac{\boldsymbol{\mu}}{\boldsymbol{\sigma}} \in \mathbb{R}^d$ with kth coordinate μ_k/σ_k. For a matrix $\boldsymbol{A} \in \mathbb{R}^{d\times n}$, we define $\|\boldsymbol{A}\|_{2,\infty} = \max_{i\in[n]} \|\boldsymbol{A}_{:,i}\|_2$, i.e., the maximum Euclidean norm of the columns.

We remind the reader of some notation introduced previously, with an important redefinition for $\overline{\boldsymbol{X}}_{\mathrm{RR}}$. For a dataset $\boldsymbol{Z} = (\boldsymbol{X}, \boldsymbol{Y})$, we write $\boldsymbol{Z}^+ = (\boldsymbol{X}^+, \boldsymbol{Y}^+)$ and $\boldsymbol{Z}^- = (\boldsymbol{X}^-, \boldsymbol{Y}^-)$ to denote the positive and negative examples, respectively. Recall that we write the dataset batch normalized under a permutation π as $\overline{\boldsymbol{Z}}_\pi \triangleq (\overline{\boldsymbol{X}}_\pi, \boldsymbol{Y}_\pi)$, where $\overline{\boldsymbol{X}}_\pi \triangleq \mathsf{BN}_\pi(\boldsymbol{X}) \in \mathbb{R}^{d\times n}$ and $\boldsymbol{Y}_\pi = \pi \circ \boldsymbol{Y}$. We also use $\overline{\boldsymbol{X}}_\pi^+$ and $\overline{\boldsymbol{X}}_\pi^-$ to denote the submatrices of $\overline{\boldsymbol{X}}_\pi$ containing its columns corresponding to the positive and negative examples, respectively.

For the sake of analyzing the rank of $\overline{\boldsymbol{X}}_{\mathrm{RR}}$, we will redefine it as follows by throwing out redundant batches. Let $\binom{[n]}{B}$ denote the set of all $\binom{n}{B}$ unique (up to permutation) batches of size B that can be created from choosing the columns of $\boldsymbol{X} \in \mathbb{R}^{d\times n}$. Fix an arbitrary labelling of these $\binom{n}{B}$ batches, and let $\boldsymbol{B}^j \in \mathbb{R}^{d\times B}$ refer to the jth such batch. Then

$$\overline{\boldsymbol{X}}_{\mathrm{RR}} \triangleq \mathsf{BN}_{\mathrm{RR}}(\boldsymbol{X}) \triangleq \begin{bmatrix}\mathsf{BN}(\boldsymbol{B}^1) & \dots & \mathsf{BN}(\boldsymbol{B}^{\binom{n}{B}})\end{bmatrix} \in \mathbb{R}^{d\times B\binom{n}{B}}$$

Note that the rank of $\overline{\boldsymbol{X}}_{\mathrm{RR}}$ is the same as the rank of the original definition, since all we did was throw out redundant batches for the purposes of analyzing the rank.

We now turn to laying down some of the background necessary to introduce our technical results. As a motivating step, recall Proposition 4.1.2. It states that SS with permutation π can cause divergence of the GD risk if $\overline{\boldsymbol{Z}}_\pi$ is PLS or LS, but not if it is SC. Hence, determining sufficient conditions for when $\overline{\boldsymbol{Z}}_\pi$ is SC is of primary interest. Intuitively, $\overline{\boldsymbol{Z}}_\pi$ being SC should be related to some notion of genericity — the convex hulls of positive and negative features should be full dimensional. To formalize this intuition, we take a quick detour and recall several standard notions in convex analysis, defined for example in Boyd et al. (2004).

For $S \subseteq \mathbb{R}^d$, its interior $\mathrm{int}(S)$ denotes the largest open set contained in S. We say that S is *affine* if for any $\boldsymbol{x}_1, \boldsymbol{x}_2 \in S$, the

line $\lambda \boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2 \subseteq S$. An *affine combination* of k points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \in \mathbb{R}^d$ is given by $\sum_{i=1}^k \lambda_i \boldsymbol{x}_i$ where $\sum_{i=1}^k \lambda_i = 1$.[6] The affine hull of S is the set of all affine combinations of S, and is denoted by $\mathrm{aff}(S)$, and clearly $\mathrm{aff}(S)$ is an affine set. Similarly, the relative interior $\mathrm{relint}(S)$ denotes the largest open subset of $\mathrm{aff}(S)$ contained in S. For a matrix $\boldsymbol{A}$, we slightly abuse notation and write $\mathrm{aff}(\boldsymbol{A})$ to denote the affine hull of its columns. Similarly, we use $\mathrm{conv}(\boldsymbol{A})$ to denote the convex hull of its columns.

Let $\boldsymbol{x}_0 \in \mathrm{aff}(S)$ be any element of the affine hull of S. It is not hard to see that $\mathrm{aff}(S) = \boldsymbol{x}_0 + V$, where V is a linear subspace of $\mathbb{R}^d$. Furthermore, this V is uniquely determined by S. One can think of $\boldsymbol{x}_0$ as an offset and V as the space of valid directions to move in to stay in $\mathrm{aff}(S)$. We define $\dim(S) \triangleq \dim(V)$.

Definition 3. *A set $S \subseteq \mathbb{R}^d$ is called* full dimensional *if $\dim(S) = d$. This definition is equivalent to saying that $\mathrm{int}(\mathrm{conv}(S))$ is nonempty. Similarly, for any matrix $\boldsymbol{A} \in \mathbb{R}^{d \times n}$, we say $\boldsymbol{A}$ is full dimensional if the set of its columns is full dimensional.*

This formal definition of full dimensional allows us to identify sufficient conditions for a dataset $\boldsymbol{Z} = (\boldsymbol{X}, \boldsymbol{Y})$ to be SC.

B.1. Preliminary results on separability decomposition

In this section, we introduce the technical results that help us analyze the separation between shuffling SGD and GD. A unifying theme is to understand the effect of monochromatic batches and rank on the separability decomposition — and thus, divergence. In particular, we show that having monochromatic batches and being full-rank prevents divergence in the underparameterized regime. In a later section (Appendix B.4), we also show that these properties also significantly influence the optimal directions under the logistic loss.

The following lemma formalizes how monochromatic batches affect the separability decomposition.

Lemma B.1.1. *Given a permutation π, suppose there are two batches $\boldsymbol{Z}_\pi^1$ and $\boldsymbol{Z}_\pi^2$ such that $\boldsymbol{Z}_\pi^1$ consists entirely of positive examples and $\boldsymbol{Z}_\pi^2$ consists entirely of negative examples. Then, if we consider the resulting $\overline{\boldsymbol{Z}}_\pi = (\overline{\boldsymbol{X}}_\pi, \boldsymbol{Y}_\pi)$, the submatrices $\overline{\boldsymbol{X}}_\pi^+$ and $\overline{\boldsymbol{X}}_\pi^-$ of $\overline{\boldsymbol{X}}_\pi$ satisfy*

$$\mathrm{relint}(\mathrm{conv}(\overline{\boldsymbol{X}}_\pi^+)) \cap \mathrm{relint}(\mathrm{conv}(\overline{\boldsymbol{X}}_\pi^-)) \neq \emptyset.$$

Consequently, $\overline{\boldsymbol{Z}}_\pi$ is not LS.

Proof. Batch normalization ensures that the batch normalized features $\overline{\boldsymbol{X}}_\pi^1$ and $\overline{\boldsymbol{X}}_\pi^2$ are mean-zero. But this implies that $\boldsymbol{0}$ is in the convex hulls of each batch, which implies that $\mathrm{conv}(\overline{\boldsymbol{X}}_\pi^1)$ intersects $\mathrm{conv}(\overline{\boldsymbol{X}}_\pi^2)$. In fact, $\boldsymbol{0}$ is in the intersection of their relative interiors as well. To see this, we prove that the mean $\boldsymbol{\mu}$ of a batch $\boldsymbol{B} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_B\}$ lies in the relative interior of $\mathrm{conv}(\boldsymbol{B})$.

This can be shown via an inductive argument on the batch size. If $B = 2$, then $\boldsymbol{\mu}$ is the midpoint between $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, which is in the relative interior of $\mathrm{conv}(\boldsymbol{B})$. Now assume it's true for all possible batches of size B. When we add $\boldsymbol{x}_{B+1}$, we get a new batch $\boldsymbol{B}'$, with mean $\boldsymbol{\mu}' = \frac{B}{B+1}\boldsymbol{\mu} + \frac{1}{B+1}\boldsymbol{x}_{B+1}$. Hence if $\boldsymbol{x}_{B+1} \in \mathrm{conv}(\boldsymbol{B})$, clearly $\boldsymbol{\mu}' \in \mathrm{relint}(\mathrm{conv}(\boldsymbol{B}))$ by convexity. If $\boldsymbol{x}_{B+1} \notin \mathrm{conv}(\boldsymbol{B})$, then $\boldsymbol{x}_{B+1}$ is one of the vertices of $\mathrm{conv}(\boldsymbol{B}')$. Since $\boldsymbol{\mu} \in \mathrm{relint}(\mathrm{conv}(\boldsymbol{B}))$ and convexity, the segment between $\boldsymbol{\mu}$ and $\boldsymbol{x}_{B+1}$ must stay in the relative interior of $\mathrm{conv}(\boldsymbol{B}')$ except at the endpoints. Since $\boldsymbol{\mu}'$ is in the relative interior of this segment, the conclusion follows.

Hence $\mathrm{relint}(\mathrm{conv}(\overline{\boldsymbol{X}}_\pi^1))$ intersects $\mathrm{relint}(\mathrm{conv}(\overline{\boldsymbol{X}}_\pi^2))$. Since we have $\overline{\boldsymbol{X}}_\pi^+ \supseteq \overline{\boldsymbol{X}}_\pi^1$ and $\overline{\boldsymbol{X}}_\pi^- \supseteq \overline{\boldsymbol{X}}_\pi^2$, the relative interiors of the larger convex hulls intersect as well.

Finally, suppose $\overline{\boldsymbol{Z}}_\pi$ was LS. By definition there must exist a strict separating hyperplane for the two hulls. But the hulls intersect, so this is a contradiction. We conclude that $\overline{\boldsymbol{Z}}_\pi$ is not LS. □

As Lemma B.1.1 establishes, monochromatic batches lead to $\overline{\boldsymbol{Z}}_\pi$ being PLS or SC. The following lemma synthesizes nicely with the above result; it demonstrates that if $\boldsymbol{X}$ is full dimensional, and the relative interiors of the convex hulls of $\boldsymbol{X}^+$ and $\boldsymbol{X}^-$ intersect, then $\boldsymbol{Z}$ is SC.

Lemma B.1.2. *Let $\boldsymbol{Z} = (\boldsymbol{X}, \boldsymbol{Y})$ be such that*

[6] Here, unlike the definition of a convex combination, the λ_i's are allowed to be negative.

(1) $\boldsymbol{X}$ *is full dimensional.*

(2) $\operatorname{relint}(\operatorname{conv}(\boldsymbol{X}^+))$ *intersects* $\operatorname{relint}(\operatorname{conv}(\boldsymbol{X}^-))$.

Then $\boldsymbol{Z}$ *is SC.*

Proof. Consider any halfspace in $\mathbb{R}^d$. First, since the relative interiors of the hull of positive points and negative points intersect, then so too do the hulls themselves. So there is no hyperplane that separates $\boldsymbol{X}^+$ from $\boldsymbol{X}^-$, i.e. $\boldsymbol{Z}$ is either PLS or SC.

Suppose that $\boldsymbol{Z}$ is PLS, i.e., there exists a hyperplane $\boldsymbol{v}$ such that $y_i \boldsymbol{v}^\top \boldsymbol{x}_i \geq 0$ for every $(\boldsymbol{x}_i, y_i) \in \boldsymbol{Z}$. Since $\boldsymbol{X}$ is full dimensional, the hyperplane orthogonal to $\boldsymbol{v}$ cannot pass through through every point of $\boldsymbol{X}$ — hyperplanes are affine subspaces of dimension at most $d-1$. So $y_i \boldsymbol{v}^\top \boldsymbol{x}_i > 0$ for some i and also $y_j \boldsymbol{v}^\top \boldsymbol{x}_j = 0$ for some $i \neq j$; otherwise, $\boldsymbol{Z}$ is LS, which is a contradiction. Hence $\operatorname{conv}(\boldsymbol{X}^+)$ and $\operatorname{conv}(\boldsymbol{X}^-)$ touch only at the hyperplane defined by $\boldsymbol{v}$, which contradicts the assumption that the relative interiors intersect. Hence $\boldsymbol{Z}$ is SC. □

Lemmas B.1.1 and B.1.2 taken together show that to identify sufficient conditions for $\overline{\boldsymbol{Z}}_\pi$ to be SC, one should look for conditions under which $\overline{\boldsymbol{X}}_\pi$ is full dimensional and monochromatic batches are present. We already answered the former question in the main text with Proposition 3.2.1, which we restate for reference. Its proof is deferred to Appendix C.3.

Proposition B.1.3. *Assume that the original features* $\boldsymbol{X} \in \mathbb{R}^{d \times n}$ *satisfies Assumption 2 and* $B > 2$. *Then if we batch normalize and remove one datapoint from each normalized batch, to form a* $d \times (B-1)\frac{n}{B}$ *matrix in the SS case and a* $d \times (B-1)\binom{n}{B}$ *matrix in the RR case, the dataset is full-rank almost surely, regardless of which datapoints we remove. In particular, we have* $\operatorname{rank}(\overline{\boldsymbol{X}}_\pi) = \min\left\{d, (B-1)\frac{n}{B}\right\}$ *and* $\operatorname{rank}(\overline{\boldsymbol{X}}_{\mathrm{RR}}) = \min\left\{d, (B-1)\binom{n}{B}\right\}$ *almost surely.*

Let us now consider the other question about the presence of monochromatic batches. Intuitively, under Assumption 4(a), there should be many monochromatic batches w.h.p. as long as B is small. The following lemma formalizes this intuition; its proof is contained in Appendix C.1.

Lemma B.1.4. *Assume Assumption 4(a). If* $B = o(\log n)$ *then there are* $\Omega(n)$ *monochromatic batches w.h.p.. If* $B = \Omega(\log n)$, *then there are no monochromatic batches w.h.p..*

The upshot of Lemmas B.1.1, B.1.2 and B.1.4 and Proposition B.1.3 is that small batch sizes naturally prevent divergence. However, there is a natural tradeoff here: small batch sizes also entail significant variance in the batch statistics, so they can lead to large (but non-diverging) values of the GD risk anyway when we train the network with SS.

Remark B.1.5 (Multiclass classification)**.** *In the multiclass case with* $K > 2$ *different classes, one can directly generalize the above analysis to look at all* $\binom{K}{2}$ *pairwise combinations of classes. Lemma B.1.1 generalizes by requiring the existence of a monochromatic batch* for each class. *Hence as* K *increases the batch size must shrink to ensure that* $\overline{\boldsymbol{Z}}_\pi$ *is SC w.h.p., opening up a wider range of batch sizes for SS divergence.*

Finally, we formally define a robust notion of the separability decomposition that will prove helpful for quantifying the effects of increasing the batch size. To do so, we rely on the notion of the margin of a linearly separably dataset and the so-called penetration depth of overlapping convex hulls.

Definition 4 (Margin and penetration depth)**.** *Let* $\boldsymbol{Z} = (\boldsymbol{X}, \boldsymbol{Y})$ *be a dataset, and let* $\boldsymbol{X}^+, \boldsymbol{X}^-$ *denote the positive and negative features, respectively. If* $\boldsymbol{Z}$ *is linearly separable, then the* margin *of* $\boldsymbol{Z}$ *is defined to be the* ℓ_2 *margin corresponding to the maximum margin classifier for* $\boldsymbol{Z}$.

If $\boldsymbol{Z}$ *is not linearly separable, then by definition* $\operatorname{conv}(\boldsymbol{X}^+)$ *intersects* $\operatorname{conv}(\boldsymbol{X}^-)$. *The* penetration depth *(Agarwal et al., 2000) of* $\boldsymbol{Z}$ *is defined as the smallest Euclidean distance of translation of* $\operatorname{conv}(\boldsymbol{X}^+)$ *such that the resulting convex body still intersects* $\operatorname{conv}(\boldsymbol{X}^-)$. *In words, this quantifies the smallest perturbation we need to make* $\boldsymbol{Z}$ *linearly separable.*

Definition 5 (γ-robust separability decomposition)**.** *Let* $\boldsymbol{Z} = (\boldsymbol{X}, \boldsymbol{Y})$ *be a dataset and* $\boldsymbol{S}_{\mathrm{GD}}^{\mathrm{LS}} \sqcup \boldsymbol{S}_{\mathrm{GD}}^{\mathrm{SC}}$ *be the separability decomposition of* $\overline{\boldsymbol{Z}}_{\mathrm{GD}} \triangleq (\overline{\boldsymbol{X}}_{\mathrm{GD}}, \boldsymbol{Y}) \triangleq (\mathsf{BN}(\boldsymbol{X}), \boldsymbol{Y})$.

For $\gamma > 0$, *we say that* $\boldsymbol{Z}$ *is* γ*-robust if the following conditions hold.*

(1) *One of the following:*

(a) $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ *is LS and* $\boldsymbol{S}_{\mathrm{GD}}^{\mathrm{LS}}$ *has margin at least* γ*; or,*
(b) $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ *is SC and* $\boldsymbol{S}_{\mathrm{GD}}^{\mathrm{SC}}$ *has penetration depth at least* γ*.*

(2) *Let* σ_k*,* a_k*, and* b_k *denote the standard deviation, min, and max of the* k*th feature of* $\boldsymbol{X}$*. Then* $\min_{k\in[d]} \frac{\sigma_k}{b_k - a_k} = \Omega(1)$*.*

(3) $\left\|\overline{\boldsymbol{X}}_{\mathrm{GD}}\right\|_{2,\infty} = O(\sqrt{d})$*, i.e., the maximum Euclidean norm of datapoints in* $\mathsf{BN}(\boldsymbol{X})$ *is* $O(\sqrt{d})$*.*

Definition 5 formalizes the informal statement in Theorem 4.1.3 that "$\overline{\boldsymbol{Z}}_{\mathrm{GD}}$'s separability decomposition can change with small perturbations." If a dataset is robust, its separability decomposition cannot change easily by small perturbations (e.g., due to batch normalization) on datapoints. Notice here that PLS datasets can never be robust; to see why, note that $\boldsymbol{Z}$ is PLS whenever $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ is PLS. It follows that if $\boldsymbol{Z}$ is PLS then it is not γ-robust for any $\gamma > 0$.

This definition of robustness is also natural from the perspective of concentration, since it provides an immediate link between concentration of batch statistics and the separability decomposition of $\overline{\boldsymbol{Z}}_{\pi}$. In order to estimate $\overline{\boldsymbol{X}}_{\mathrm{GD}}$ well, we need to estimate $\boldsymbol{\sigma}$ to within a multiplicative factor, which explains (2). Moreover, the degree to which the SS datapoints concentrate around the GD datapoints also depend multiplicatively on the size of GD datapoints, which explains (3).

The following proposition provides a sufficient condition for the datapoints concentrating within a distance γ of the GD datapoints (cf. the definition of γ-robustness); its proof is deferred to Appendix C.2.

Proposition B.1.6. *Suppose* $\boldsymbol{Z}$ *is* γ*-robust and* $B = \Omega(d\log(nd)/\gamma^2)$*. Then with probability at least* $1 - 1/\operatorname{poly}(n, d)$ *over the choice of* π*, we have* $\left\|\overline{\boldsymbol{X}}_{\pi} - \pi \circ \overline{\boldsymbol{X}}_{\mathrm{GD}}\right\|_{2,\infty} = O(\gamma)$*.*

With this setup in hand, we are ready to present our main theorems characterizing the separability decompositions of SS+BN and RR+BN.

B.2. Separability decomposition for SS

Theorem B.2.1 (Separability decomposition for SS, formal)**.** *Throughout this theorem, assume that* $B > 2$ *and Assumption 4(a) and Assumption 2 hold.*

Suppose that $d \le (B-1)\frac{n}{B}$*. Then the following hold.*

(1) $\overline{\boldsymbol{Z}}_{\pi}$ *is SC w.h.p. for* $B = o(\log n)$*. If we relax Assumption 2, then* $\overline{\boldsymbol{Z}}_{\pi}$ *can be PLS as well.*

(2) *Suppose further that* $\boldsymbol{Z}$ *is* γ*-robust. Then* $\overline{\boldsymbol{Z}}_{\pi}$ *has the same separability decomposition as* $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ *w.h.p. for* $B = \Omega(d\log(nd)/\gamma^2)$*.*

Now suppose $d > (B-1)\frac{n}{B}$*. Then the following hold.*

(3) $\overline{\boldsymbol{Z}}_{\pi}$ *is PLS w.h.p. for* $B = o(\log n)$*.*

(4) $\overline{\boldsymbol{Z}}_{\pi}$ *is LS w.h.p. for* $B = \Omega(\log n)$*.*

Proof. We consider cases based on whether the batch size is large or small.

Small batch size. By Lemma B.1.4, when $B = o(\log n)$, w.h.p. we get a batch comprised solely of positive examples and a batch comprised solely of negative examples. By Lemma B.1.1, this implies that $\overline{\boldsymbol{Z}}_{\pi}$ is PLS or SC, and the relative interior of the positive features intersects the relative interior of the negative features. By Lemma B.1.2, if $\overline{\boldsymbol{X}}_{\pi}$ is full dimensional, then $\overline{\boldsymbol{Z}}_{\pi}$ is SC. Hence it suffices to analyze the rank of $\overline{\boldsymbol{X}}_{\pi}$.

The maximal rank of $\overline{\boldsymbol{X}}_{\pi}$ is $\min\left\{d, (B-1)\frac{n}{B}\right\}$ because of the mean zero constraint enforced in every batch. Because we assumed that Assumption 2 holds, Proposition 3.2.1 implies that $\overline{\boldsymbol{X}}_{\pi}$ achieves this upper bound almost surely. Putting it all together, we conclude that when $d \le (B-1)\frac{n}{B}$, $\overline{\boldsymbol{X}}_{\pi}$ is full-dimensional. It follows that $\overline{\boldsymbol{Z}}_{\pi}$ is SC w.h.p. over the choice of π, which proves (1).

On the other hand, when $d > (B-1)\frac{n}{B}$, there always exists a hyperplane that passes through all of the monochromatic batches of $\overline{\boldsymbol{Z}}_{\pi}$ and perfectly classifies non-monochromatic batches (see Appendix B.4). It follows that $\overline{\boldsymbol{Z}}_{\pi}$ is PLS, which proves (3).

Large batch size. Now consider when $B = \Omega(\log n)$. In this regime, Lemma B.1.4 implies there are no monochromatic batches with high probability. Moreover, Proposition 3.2.1 implies that the features are full rank. It thus follows that when $d > (B-1)\frac{n}{B}$, $\overline{\boldsymbol{Z}}_\pi$ is LS, which proves (4). Let us now consider the case $d \le (B-1)\frac{n}{B}$.

When $\boldsymbol{Z}$ is γ-robust, we can directly apply Proposition B.1.6 to prove (2). Indeed, for $B = \Omega(d\log(nd)/\gamma^2)$, each SS datapoint is within distance $O(\gamma)$ of the corresponding GD datapoint with probability at least $1 - 1/\operatorname{poly}(n,d)$. By increasing the batch size by at most a constant factor, we can guarantee that each SS datapoint is in fact within distance $\gamma/3$ of the corresponding GD datapoint.

If $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ is LS, γ-robustness implies that the max-margin hyperplane for $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ has margin at least γ. Since each SS datapoint moves at most $\gamma/3$ from the corresponding GD datapoint, this hyperplane still has margin at least $2\gamma/3$, implying that $\overline{\boldsymbol{Z}}_\pi$ is LS. If $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ is SC, then γ-robustness implies that we need to translate the convex hulls of positive and negative points by at least γ to separate them. But we can only translate them by a total of $\gamma/3 + \gamma/3 = 2\gamma/3 < \gamma$, so the hulls stay strictly overlapping. This implies $\overline{\boldsymbol{Z}}_\pi$ is also SC. This concludes the proof of (2). □

Remark B.2.2. *If $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ is not γ-robust for any $\gamma > 0$ (for example if $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ is PLS), then it is not hard to construct examples where the separability decomposition of $\overline{\boldsymbol{Z}}_\pi$ is LS or PLS, and SS diverges. For a good example of this scenario, see Figures 4b and 4d. Even if $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ is γ-robust, if $B = o(d\log(nd)/\gamma^2)$, then concentration can fail to hold in the worst case, and we can find analogous constructions where SS diverges.*

B.3. Separability decomposition for RR

Theorem B.3.1 (Separability decomposition for RR, formal)**.** *Suppose that $B > 2$ and Assumption 4(b) and Assumption 2 hold.*

If $d \le (B-1)\binom{n}{B}$, then $\overline{\boldsymbol{Z}}_{\mathrm{RR}}$ is SC and $\overline{\boldsymbol{X}}_{\mathrm{RR}}$ is full-rank almost surely, regardless of the separability decomposition of $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$.

Otherwise, if $d > (B-1)\binom{n}{B}$, then $\overline{\boldsymbol{Z}}_{\mathrm{RR}}$ is deterministically PLS, regardless of the separability decomposition of $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$.

Proof. When there are at least B positive and B negative examples, there exists a batch of all positive and a batch of all negative examples. Hence by Lemma B.1.1, $\overline{\boldsymbol{Z}}_{\mathrm{RR}}$ is PLS or SC. By Proposition B.1.3, under Assumption 2, $\overline{\boldsymbol{X}}_{\mathrm{RR}}$ attains the maximal rank of $\min\left\{d, (B-1)\binom{n}{B}\right\}$ almost surely. So by Lemma B.1.2, if $d \le (B-1)\binom{n}{B}$, $\overline{\boldsymbol{Z}}_{\mathrm{RR}}$ will be SC and $\overline{\boldsymbol{X}}_{\mathrm{RR}}$ is full-rank almost surely. On the other hand, if $d > (B-1)\binom{n}{B}$, then $\overline{\boldsymbol{Z}}_{\mathrm{RR}}$ is PLS deterministically because there always exists a hyperplane that passes through all of the monochromatic batches of $\overline{\boldsymbol{Z}}_{\mathrm{RR}}$ and perfectly classifies non-monochromatic batches. □

B.4. Characterizing the optimal direction of classifiers

Thus far, we have primarily considered the separability decomposition of $\overline{\boldsymbol{Z}}_\pi$ and $\overline{\boldsymbol{Z}}_{\mathrm{RR}}$. In fact, we can say more about the direction of optimal classifiers under the logistic loss. First, we prove Lemma 4.1.1, which constrains optimal directions via the separability decomposition. Next, we leverage overparameterization and the rank properties shown in Proposition B.1.3 to characterize the optimal direction under the logistic loss for data drawn from a density. Using these insights, we can prove our main result of this section, Proposition B.4.3.

We first restate and prove Lemma 4.1.1.

Lemma B.4.1. *Let $\boldsymbol{Z} = \boldsymbol{S}^{\mathrm{LS}} \sqcup \boldsymbol{S}^{\mathrm{SC}}$ be the separability decomposition of $\boldsymbol{Z}$. If $\boldsymbol{v}$ is an optimal direction for $\mathcal{L}$, then $\boldsymbol{v}^\top \boldsymbol{x} = 0$ for all $\boldsymbol{x} \in \operatorname{Span}(\boldsymbol{X}^{\mathrm{SC}})$ and $y_i \boldsymbol{v}^\top \boldsymbol{x}_i > 0$ for every $(\boldsymbol{x}_i, y_i) \in \boldsymbol{S}^{\mathrm{LS}}$.*

Proof. By definition of $\boldsymbol{S}^{\mathrm{SC}}$, there exists some $(\boldsymbol{x}_i, y_i) \in \boldsymbol{S}^{\mathrm{SC}}$ such that $y_i \langle \boldsymbol{v}, \boldsymbol{x}_i \rangle \le 0$. If $\langle \boldsymbol{v}, \boldsymbol{x}_i \rangle \neq 0$, then $y_i \langle t\boldsymbol{v}, \boldsymbol{x}_i \rangle \to -\infty$ as $t \to \infty$, which contradicts the assumption that $\boldsymbol{v}$ is an optimal direction. Similarly, if $y_i \langle \boldsymbol{v}, \boldsymbol{x}_i \rangle \le 0$ for some $(\boldsymbol{x}_i, y_i) \in \boldsymbol{S}^{\mathrm{LS}}$, then clearly $\boldsymbol{u} + t\boldsymbol{v}$ cannot infimize $\mathcal{L}$, as there exists a hyperplane which strictly separates $\boldsymbol{S}^{\mathrm{LS}}$ and is orthogonal to $\operatorname{Span}(\boldsymbol{S}^{\mathrm{SC}})$. □

Next, we restate and prove Proposition 4.1.2.

Proposition B.4.2. *Suppose Assumption 1(a) holds. Assume that the iterates $\boldsymbol{v}_\pi(t)$ infimize $\mathcal{L}_\pi$, and their projections onto $\mathrm{Span}(\overline{\boldsymbol{X}}_\pi^{\mathrm{SC}})^\perp$ converge in direction to some optimal direction $\boldsymbol{v}_\pi^*$ for $\mathcal{L}_\pi$. Then the GD risk $\mathcal{L}_{\mathrm{GD}}$ diverges if and only if $\overline{\boldsymbol{Z}}_\pi$ is PLS or LS and there exists some $(\boldsymbol{x}_i, y_i) \in \overline{\boldsymbol{Z}}_{\mathrm{GD}}$ such that $y_i \boldsymbol{v}_\pi^{*\top} \boldsymbol{x}_i < 0$. The analogous statement holds true for $\overline{\boldsymbol{Z}}_{\mathrm{RR}}$ under Assumption 1(b). Furthermore, the if part holds true for SS and RR without Assumption 1.*

Proof. First suppose Assumption 1(a) holds. Then if $\overline{\boldsymbol{Z}}_\pi$ is SC, Lemma B.4.1 implies that any optimal direction $\boldsymbol{v}_\pi^*$ must be orthogonal to all of $\mathbb{R}^d$, so $\boldsymbol{v}_\pi^* = \boldsymbol{0}$. Hence the iterates $\boldsymbol{v}_\pi(t)$ converge to a finite optimum, which implies the GD risk cannot diverge. Note that if Assumption 1(a) doesn't hold, then the only difference is that being orthogonal to $\overline{\boldsymbol{X}}_\pi^{\mathrm{SC}}$ does not imply that $\boldsymbol{v}_\pi^* = \boldsymbol{0}$.

Now suppose $\overline{\boldsymbol{Z}}_\pi$ is PLS or LS. Regardless of whether Assumption 1(a) holds, if $\boldsymbol{v}_\pi(t)$ infimizes $\mathcal{L}_\pi$, we necessarily have $\|\boldsymbol{v}_\pi(t)\|_2 \to +\infty$. Hence any mistake on $(\boldsymbol{x}_i, y_i) \in \overline{\boldsymbol{Z}}_{\mathrm{GD}}$ implies divergence. And clearly, if there is no mistake on any $(\boldsymbol{x}_i, y_i)$ then the GD risk does not diverge.

The same proof also goes through for $\overline{\boldsymbol{Z}}_{\mathrm{RR}}$, so this proves the theorem. □

At first glance, one might expect to be able to perfectly classify all the datapoints in the overparameterized regime. However, under the logistic risk, the optimal direction instead puts monochromatic batches *on the decision boundary*; the following proposition formalizes this notion.

Proposition B.4.3. *Suppose Assumption 2, $B > 2$, and $d > (B-1)\frac{n}{B}$. Almost surely, for any $\pi \in \mathbb{S}_n$, there exists $\boldsymbol{v} \in \mathbb{R}^d$ which satisfies* (1) *for any non-monochromatic batch $\overline{\boldsymbol{Z}}_\pi^j$ we have $\mathrm{sgn}(\boldsymbol{v}^\top \overline{\boldsymbol{X}}_\pi^j) = \boldsymbol{Y}_\pi^j$ and* (2) *for any monochromatic batch $\overline{\boldsymbol{Z}}_\pi^k$ we have $\boldsymbol{v}^\top \overline{\boldsymbol{X}}_\pi^k = \boldsymbol{0}^\top$. Furthermore, any optimal direction $\boldsymbol{v}_\pi^*$ for the logistic risk $\mathcal{L}_\pi$ necessarily satisfies both* (1) *and* (2)*. The same conclusion holds for RR as well, with the requirement $d > (B-1)\binom{n}{B}$.*

More precisely, our analysis is motivated by the fact that if $d \geq n$ and $\boldsymbol{X} \in \mathbb{R}^{d \times n}$ is full-rank, then given any $\boldsymbol{c} \in \mathbb{R}^n$ we can find some halfspace $\boldsymbol{v} \in \mathbb{R}^d$ such that $\boldsymbol{v}^\top \boldsymbol{X} = \boldsymbol{c}^\top$. In our binary classification setting, linear separability is implied by $\mathrm{sgn}(\boldsymbol{c}^\top) = \boldsymbol{Y}$. However, we cannot directly apply this fact because BN actually prevents $\overline{\boldsymbol{X}}_\pi$ from being full-rank due to the mean zero constraint. However, it turns out that the following slightly weaker statement is true. We can always find a halfspace $\boldsymbol{v} \in \mathbb{R}^d$ that perfectly separates all the non-monochromatic batches. The following lemma proves this half of the proposition.

Lemma B.4.4. *Suppose Assumption 2, $B > 2$, and $d > (B-1)\frac{n}{B}$. Then almost surely there exists $\boldsymbol{v} \in \mathbb{R}^d$ such that for every non-monochromatic batch $\overline{\boldsymbol{Z}}_\pi^j = (\overline{\boldsymbol{X}}_\pi^j, \boldsymbol{Y}_\pi^j)$, we have $\mathrm{sgn}(\boldsymbol{v}^\top \overline{\boldsymbol{X}}_\pi^j) = \boldsymbol{Y}_\pi^j$. The same conclusion holds for RR as well, with the requirement $d > (B-1)\binom{n}{B}$.*

Proof. We denote $\overline{\boldsymbol{X}}_\pi^j = \mathsf{BN}(\boldsymbol{X}_\pi^j) = \begin{bmatrix} \boldsymbol{x}_1 & \cdots & \boldsymbol{x}_B \end{bmatrix}$ and $\boldsymbol{Y}_\pi^j = \begin{bmatrix} y_1 & \cdots & y_B \end{bmatrix}$. Since Assumption 2 holds, it also follows that Proposition 3.2.1 holds. Hence, within each batch, any $B-1$ of the datapoints are linearly independent almost surely. This implies that we can find $\boldsymbol{v} \in \mathbb{R}^d$ such that for any batch $\overline{\boldsymbol{X}}_\pi^j$ and $\boldsymbol{c} \in \mathbb{R}^{B-1}$, we have $\boldsymbol{v}^\top \boldsymbol{x}_i = c_i$ for all $i \in [B-1]$. In particular, we can pick c_i such that $\mathrm{sgn}(c_i) = y_i$. Next, we show that $\boldsymbol{v}$ can be picked such that $\mathrm{sgn}(\boldsymbol{v}^\top \boldsymbol{x}_B) = y_B$. The mean zero constraint enforces that $\boldsymbol{v}^\top \boldsymbol{x}_B = -\sum_{i=1}^{B-1} c_i$. But if the labels are not monochromatic, we can just increase the absolute value of one of the c_i's so that $\mathrm{sgn}(-\sum_{i=1}^{B-1} c_i) = y_B$, as desired. □

Hence, in the overparameterized regime minimizing the logistic risk will lead to a classifier which separates all the non-monochromatic batches. What happens to the monochromatic batches? It turns out that minimizing the logistic risk will lead to a classifier whose decision boundary *contains* all of the monochromatic batches.

Lemma B.4.5. *Assuming $d > (B-1)\frac{n}{B}$, any optimal direction $\boldsymbol{v}_* \in \mathbb{R}^d$ for the logistic risk simultaneously puts all of the monochromatic batches on the decision boundary. More precisely, for any monochromatic batch $\overline{\boldsymbol{Z}}_\pi^j = (\overline{\boldsymbol{X}}_\pi^j, \boldsymbol{Y}_\pi^j)$ we have $\boldsymbol{v}_*^\top \overline{\boldsymbol{X}}_\pi^j = \boldsymbol{0}^\top$. Similarly, if $d > (B-1)\binom{n}{B}$, the same conclusion holds for the RR dataset.*

Proof. Again, we denote the monochromatic batch by $\overline{\boldsymbol{Z}}_\pi^j$ by $\overline{\boldsymbol{X}}_\pi^j = \begin{bmatrix} \boldsymbol{x}_1 & \cdots & \boldsymbol{x}_B \end{bmatrix}$ and $\boldsymbol{Y}_\pi^j = \begin{bmatrix} y_1 & \cdots & y_B \end{bmatrix}$. The logistic loss on a single input $\boldsymbol{x}_i$ for classifier $\boldsymbol{v} \in \mathbb{R}^d$ is $\ell(\boldsymbol{v}^\top \boldsymbol{x}_i, y_i) = -\log \rho(y_i \boldsymbol{v}^\top \boldsymbol{x}_i)$, where $\rho(t) = 1/(1+\exp(-t))$

is the sigmoid function. WLOG suppose that $y_i = 1$ for all $i \in [B]$. Hence the minibatch risk is

$$-\sum_{i=1}^{B} \log \rho(\boldsymbol{v}^\top \boldsymbol{x}_i) = -\sum_{i=1}^{B-1} \log \rho(\boldsymbol{v}^\top \boldsymbol{x}_i) - \log \rho\left(-\boldsymbol{v}^\top \sum_{i=1}^{B-1} \boldsymbol{x}_i\right).$$

For each i we can look at the first order optimality condition for $s_i = \boldsymbol{v}^\top \boldsymbol{x}_i$. This yields $\rho(s_i) = \rho(-\sum_{i=1}^{B-1} s_i)$.

Note that this is satisfied when $s_i = 0$ for all $i \in [B]$, and by strict convexity this is the unique minimizer. And because of overparameterization, we can find a $\boldsymbol{v}_*$ that will satisfy $s_i = 0$ for each monochromatic batch, i.e., the batch entirely lies on the decision boundary of the classifier defined by $\boldsymbol{v}_*$.

The proof carries over immediately to the RR setting, except in that setting, being overparameterized means $d > (B-1)\binom{n}{B}$. Hence the lemma is proved. □

The geometric interpretation of Lemma B.4.5 is that in the overparameterized regime, if the dataset contains any monochromatic batches, then any optimal direction $\boldsymbol{v}_*$ must be orthogonal to the subspace spanned by the monochromatic batches. Note, however, that the definition of overparameterized here depends on whether we look at $\overline{\boldsymbol{Z}}_\pi$ or $\overline{\boldsymbol{Z}}_{\mathrm{RR}}$. In the former case, overparameterized means $d > (B-1)\frac{n}{B}$, whereas in the latter case, overparameterized means $d > (B-1)\binom{n}{B}$. This insight motivates the construction of the toy datasets in Section 4.2. We conclude our characterization of the optimal direction in the overparameterized regime by proving Proposition B.4.3.

Proof of Proposition B.4.3. Proposition 3.2.1 implies that for $d > (B-1)\frac{n}{B}$ and assuming assumption 2, almost surely we have $\mathrm{rank}(\overline{\boldsymbol{X}}_\pi) = (B-1)\frac{n}{B}$. We can lower bound the infimum of the SS logistic risk by the sum of the infima of the mini-batch SS logistic risks. Combining Lemmas B.4.4 and B.4.5, the claim follows. The same argument holds for $\overline{\boldsymbol{Z}}_{\mathrm{RR}}$ assuming $d > (B-1)\binom{n}{B}$. □

C. Proofs of technical lemmas

In this section we spell out the formal details of our guarantees for how permutations interact with BN. In Appendix C.1, we show that given a batch size $B = o(\log n)$ and a constant number of classes K, then w.h.p. over the choice of π there exists (many) monochromatic batches. In other words, for small batch sizes there are many batches consisting solely of positive or negative examples (in case of $K = 2$). Conversely, we show that above this threshold such monochromatic batches do not appear w.h.p.. In Appendix C.2, we show that there is a commensurate threshold above which the batch statistics themselves concentrate in the without-replacement setting. To do so we will appeal to the recent results of Bardenet & Maillard (2015) on the concentration of without-replacement estimators. Finally, in Appendix C.3, we prove that assuming the original features were drawn from a density, the batch normalized features have maximal rank almost surely. In other words, batch normalization preserves genericity of the original features modulo the mean zero constraint inside each batch.

C.1. Presence of monochromatic batches

In this section we formally prove Lemma B.1.4 via standard concentration arguments. One of the potential pitfalls of any mini-batch based algorithm is that its batches may not be representative of the entire dataset. More precisely, let's suppose we have a dataset $\boldsymbol{Z}$ with labels coming from K classes. Suppose furthermore that the dataset is balanced — each class contains $n = Bm$ examples (here K is a constant). For any batch size B which is not sufficiently large, i.e. $B = o(\log n)$, then w.h.p. over the permutation π we will have $\Theta(n)$ batches which are *monochromatic* — batches which only consist of examples in the same class. This is closely related to the classic coupon collector problem, but we restate the guarantees here for the sake of completeness.

To prove this claim, we appeal to standard martingale concentration inequalities. Consider the batches $\overline{\boldsymbol{Z}}_\pi^i$ for $i \in [Km]$ and define the indicator variables $T_i = \mathbf{1}[\overline{\boldsymbol{Z}}_\pi^i$ is monochromatic], and set $T \triangleq \sum_i T_i$, the total number of monochromatic batches.

Next, note that the sequence

$$\mathbb{E}[T], \mathbb{E}[T|T_1], \mathbb{E}[T|T_1, T_2], \ldots, \mathbb{E}[T|T_1, T_2, \ldots, T_{Km}]$$

is a Doob martingale. Indeed the martingale property follows from the tower property:

$$\mathbb{E}[\mathbb{E}[T|T_{1:k}]|\,\mathbb{E}[T|T_{1:(k-1)}]] = \mathbb{E}[T|T_{1:(k-1)}].$$

Note that $\mathbb{E}[T|T_1,\dots,T_{Km}] = T$ and $\mathbb{E}[T_1] = K\binom{n}{B}/\binom{Kn}{B}$. Hence by linearity $\mathbb{E}[T] = \frac{K^2 n}{B}\frac{\binom{n}{B}}{\binom{Kn}{B}}$.

Let us now show that the martingale increments $\mathbb{E}[T|T_{1:k}] - \mathbb{E}[T|T_{1:(k-1)}]$ are bounded a.s.. In the worst case, the kth batch can only decrease this conditional expectation by at most K, since for any fixed class we can only remove at most one monochromatic batch from it. Hence the total number of potential monochromatic batches left can decrease by at most K. This worst case guarantee still holds under conditional expectation, so $\left|\mathbb{E}[T|T_{1:k}] - \mathbb{E}[T|T_{1:(k-1)}]\right| \le K$ a.s..

Azuma-Hoeffding then tells us that for any $\epsilon > 0$ we have

$$\mathbb{P}[|T - \mathbb{E}[T]| \ge \epsilon] \le 2\exp\left(-\frac{B\epsilon^2}{2nK^3}\right).$$

This gives us the following fact which we use in both the regression and classification setting.

Fact C.1.1. *For any $\delta \in (0,1)$, and a constant number of classes K with n datapoints each, we have with probability at least $1-\delta$ that the total number of monochromatic batches T satisfies*

$$\left|T - \frac{K^2 n}{B}\frac{\binom{n}{B}}{\binom{Kn}{B}}\right| \le \sqrt{\frac{2nK^3\log(2/\delta)}{B}}.$$

For $K = O(1)$, we note that the above inequality guarantees that we can get within $O(\sqrt{n\log n})$ of the true expectation with at most $1/\operatorname{poly}(n)$ failure probability. For the toy regression dataset in Section 3.3, we have $K = 2$ and $B = 2$, so we can use Fact C.1.1 to deduce that there will be $\Theta(n)$ monochromatic batches with high probability. We can also use Fact C.1.1 to prove Lemma B.1.4.

Proof of Lemma B.1.4. In the classification setting, we have $K = 2$ classes (positive and negative). Using the folklore inequalities

$$\left(\frac{n}{k}\right)^k \le \binom{n}{k} \le \left(\frac{en}{k}\right)^k,$$

we can deduce the lower bound on the expectation of T:

$$\mathbb{E}[T] \ge \frac{4n}{B(2e)^B}.$$

The lower bound is $\Omega(n^{1-\epsilon})$ for any $\epsilon > 0$ whenever $B = o(\log n)$, so indeed when when $B = o(\log n)$ we have a positive number of monochromatic batches w.h.p.

Let us now upper bound the probability of obtaining any monochromatic batches. We have

$$\mathbb{P}[T_1 = 1] = 2\frac{\binom{n}{B}}{\binom{2n}{B}} \le 2\frac{\prod_{k=0}^{B-1}(n-k)}{\prod_{k=0}^{B-1}(2n-k)} \le 2\frac{\prod_{k=0}^{B-1}(n-k)}{\prod_{k=0}^{B-1}2(n-k)} \le 2^{-B+1}.$$

Hence by union bound the probability that $T > 0$ is upper bounded by $\frac{4n}{B\cdot 2^B}$. This is $1/\operatorname{poly}(n)$ for some $B = \Omega(\log n)$, so indeed when $B = \Omega(\log n)$ we have no monochromatic batches with probability at least $1 - 1/\operatorname{poly}(n)$. This concludes the proof. □

C.2. Concentration of batch statistics for without-replacement sampling

In the following, we are generating B samples of $X_i \in \mathbb{R}$ without replacement from a population of size n, contained in $[a,b]$ a.s.. We let μ be the population mean and σ^2 be the population variance.

Lemma C.2.1 (Corollary 2.5 in (Bardenet & Maillard, 2015)). *Let $\hat{\mu}_B$ denote the sample mean for a sample of size B drawn without replacement from the overall population. For all $B \leq n$ and $\delta \in (0,1)$ we have with probability at least $1-\delta$ that*

$$|\hat{\mu}_B - \mu| \leq (b-a)\sqrt{\frac{\log(2/\delta)}{B}}.$$

Similarly, they prove the following result about concentration of the empirical variance

Lemma C.2.2 (Lemma 4.1 in (Bardenet & Maillard, 2015)). *Let $\hat{\sigma}_B^2 \triangleq \frac{1}{B}\sum_{i=1}^B (X_i - \hat{\mu}_B)^2$ be the (biased) empirical variance estimator and $\hat{\sigma}_B \triangleq \sqrt{\hat{\sigma}_B^2}$. Then for all $\delta \in (0,1)$ we have with probability at least $1-\delta$ that*

$$\hat{\sigma}_B \geq \sigma - 3(b-a)\sqrt{\frac{\log(3/\delta)}{2B}}.$$

We now prove the other side of this concentration inequality with a quick application of (Maurer, 2006, Theorem 1), following the same notation as in Bardenet & Maillard (2015).

Lemma C.2.3. *For all $\delta \in (0,1)$ we have with probability at least $1-\delta$ that*

$$\hat{\sigma}_B \leq \sigma + (b-a)\sqrt{\frac{\log(1/\delta)}{2B}}.$$

Proof. We take the self bounded random variable $Z = \frac{B}{(b-a)^2}\tilde{V}_B$, where

$$\tilde{V}_B \triangleq \frac{1}{B}\sum_{i=1}^B (X_i - \mu)^2$$

is computed with the samples X_i sampled *with replacement*. On the other hand,

$$V_B \triangleq \frac{1}{B}\sum_{i=1}^B (X_i' - \mu)^2$$

is computed with the samples X_i' sampled *without replacement*. We can relate the concentration of V_B to that of $\tilde{V}_B$; the latter quantity is possible to analyze with the entropy method. Indeed, a routine modification of the proof of Bardenet & Maillard (2015, Lemma 3.3) (which uses essentially the same definition of Z) implies that

$$\mathbb{P}\left[V_B - \sigma^2 \geq \frac{(b-a)^2}{B}\epsilon\right] \leq \exp\left(-\frac{(b-a)^2\epsilon^2}{2B\sigma^2}\right).$$

Solving for ϵ in terms of δ yields $\epsilon = \sqrt{\frac{2B\sigma^2}{(b-a)^2}\log(1/\delta)}$, so we obtain

$$\mathbb{P}\left[V_B - \sigma^2 \geq (b-a)\sigma\sqrt{\frac{2\log(1/\delta)}{B}}\right] \leq \delta.$$

Since $V_B = (\hat{\mu}_B - \mu)^2 + \hat{\sigma}_B^2 \geq \hat{\sigma}_B^2$, we can complete the square to obtain

$$\mathbb{P}\left[\hat{\sigma}_B^2 \geq \left(\sigma + (b-a)\sqrt{\frac{\log(1/\delta)}{2B}}\right)^2\right] \leq \delta.$$

So with probability at least $1-\delta$, we have

$$\hat{\sigma}_B^2 \leq \left(\sigma + (b-a)\sqrt{\frac{\log(1/\delta)}{2B}}\right)^2,$$

and taking square roots implies the desired result. □

Now, let us return to the question of concentration for batch norm with a randomly selected permutation π. The following proposition shows that assuming the batch size is large enough, the features of corresponding SS and GD datapoints are close to each other.

Proposition C.2.4. *If*

$$B = \Omega\left(\frac{\log(nd)}{\min_{k\in[d]}(\frac{\sigma_k}{b_k-a_k})^2\epsilon^2}\right),$$

then with probability at least $1 - 1/\operatorname{poly}(n,d)$ *we have*

$$\left\|\overline{\boldsymbol{X}}_\pi - \pi\circ\overline{\boldsymbol{X}}_{\mathrm{GD}}\right\|_{2,\infty} \le \frac{\epsilon}{1-\epsilon}\left\|\overline{\boldsymbol{X}}_{\mathrm{GD}}\right\|_{2,\infty} + \frac{\epsilon\sqrt{d}}{1-\epsilon}.$$

Here, we remind the reader that for a matrix $\boldsymbol{A}\in\mathbb{R}^{d\times n}$*, we define* $\|\boldsymbol{A}\|_{2,\infty} = \max_{i\in[n]}\|\boldsymbol{A}_{:,i}\|_2$*, i.e. the maximum Euclidean norm of the columns.*

Proof. WLOG consider the unnormalized first batch $\boldsymbol{X}_\pi^1 = \{\boldsymbol{x}_1,\dots,\boldsymbol{x}_B\}$. We initially handle concentration along its first coordinate $\{x_1,\dots,x_B\}$ for its first datapoint x_1. Write $\hat{\boldsymbol{\mu}},\hat{\boldsymbol{\sigma}}\in\mathbb{R}^d$ to denote the mini-batch mean and standard deviation, respectively, and $\hat{\mu}_k,\hat{\sigma}_k\in\mathbb{R}$ for the kth coordinate of these vectors. Similarly let $\boldsymbol{\mu},\boldsymbol{\sigma}\in\mathbb{R}^d$ denote the full-batch mean and standard deviation, and let $\mu_k,\sigma_k\in\mathbb{R}$ to denote the kth coordinate of these vectors, respectively. Finally, let $\boldsymbol{a},\boldsymbol{b}\in\mathbb{R}^d$ denote the coordinate-wise min and max of $\boldsymbol{X}$, with $a_k,b_k\in\mathbb{R}$ denoting the min and max for the kth coordinate of $\boldsymbol{X}$.

Hence, the first feature of the first datapoint in the normalized first batch $\overline{\boldsymbol{X}}_\pi^1$ is $\frac{x_1-\hat{\mu}_1}{\hat{\sigma}_1}$, whereas the corresponding quantity in $\overline{\boldsymbol{X}}_{\mathrm{GD}}$ is $\frac{x_1-\mu_1}{\sigma_1}$.

Pick

$$B = \Omega\left(\frac{(\log(1/\delta)}{(\frac{\sigma_1}{b_1-a_1})^2\epsilon^2}\right).$$

Then from Lemmas C.2.1 to C.2.3, we see that $|\hat{\mu}_1-\mu_1|\le\epsilon\sigma_1$ and $|\hat{\sigma}_1-\sigma_1|\le\epsilon\sigma_1$ with probability at least $1-\delta$ for the first datapoint x_1.

Now, we have

$$\begin{aligned}
\left|\frac{x_1-\hat{\mu}_1}{\hat{\sigma}_1} - \frac{x_1-\mu_1}{\sigma_1}\right| &= \left|\frac{x_1-\mu_1}{\hat{\sigma}_1} - \frac{x_1-\mu_1}{\sigma_1} + \frac{\mu_1-\hat{\mu}_1}{\hat{\sigma}_1}\right| \\
&\le \left|\frac{\sigma_1-\hat{\sigma}_1}{\hat{\sigma}_1}\right|\left|\frac{x_1-\mu_1}{\sigma_1}\right| + \frac{|\mu_1-\hat{\mu}_1|}{\hat{\sigma}_1} \\
&\le \frac{\epsilon\sigma_1}{(1-\epsilon)\sigma_1}\left|\frac{x_1-\mu_1}{\sigma_1}\right| + \frac{\epsilon\sigma_1}{(1-\epsilon)\sigma_1} \\
&\le \frac{\epsilon}{1-\epsilon}\left|\frac{x_1-\mu_1}{\sigma_1}\right| + \frac{\epsilon}{1-\epsilon}.
\end{aligned}$$

To aggregate this bound across features, we need to pick B such that $|\hat{\mu}_k-\mu_k|\le\epsilon\sigma_k$ and $|\hat{\sigma}_k-\sigma_k|\le\epsilon\sigma_k$ for every feature $k\in[d]$. Indeed, this is achieved via the union bound by picking $\delta = 1/\operatorname{poly}(d)$ and

$$B = \Omega\left(\frac{\log(1/\delta)}{\min_{k\in[d]}(\frac{\sigma_k}{b_k-a_k})^2\epsilon^2}\right).$$

For this batch size, we see that

$$\left\|\frac{\boldsymbol{x}_1-\hat{\boldsymbol{\mu}}}{\hat{\boldsymbol{\sigma}}} - \frac{\boldsymbol{x}_1-\boldsymbol{\mu}}{\boldsymbol{\sigma}}\right\|_2 \le \frac{\epsilon}{1-\epsilon}\left\|\frac{\boldsymbol{x}_1-\boldsymbol{\mu}}{\boldsymbol{\sigma}}\right\|_2 + \frac{\epsilon\sqrt{d}}{1-\epsilon}.$$

Note that a similar inequality holds for all of $\boldsymbol{x}_1,\dots,\boldsymbol{x}_B$. Now, recalling the definition of $\|\cdot\|_{2,\infty}$, by applying union bound on all the batches, we have

$$\left\|\overline{\boldsymbol{X}}_\pi - \pi\circ\overline{\boldsymbol{X}}_{\mathrm{GD}}\right\|_{2,\infty} \le \frac{\epsilon}{1-\epsilon}\left\|\overline{\boldsymbol{X}}_{\mathrm{GD}}\right\|_{2,\infty} + \frac{\epsilon\sqrt{d}}{1-\epsilon},$$

which occurs with probability at least $1 - 1/\operatorname{poly}(n, d)$ when we appropriately choose $B = \Omega\left(\frac{\log(nd)}{\min_{k\in[d]}(\frac{\sigma_k}{b_k - a_k})^2\epsilon^2}\right)$. □

We now use the above proposition to prove Proposition B.1.6.

Proof of Proposition B.1.6. Conditions (2) and (3) in the definition of γ-robustness ensures that $\min_{k\in[d]}(\frac{\sigma_k}{b_k - a_k})^2 = \Omega(1)$ and $\left\|\overline{\boldsymbol{X}}_{\mathrm{GD}}\right\|_{2,\infty} = O(\sqrt{d})$. Hence taking $B = \Omega(\frac{\log(nd)}{\epsilon^2})$ as in Proposition C.2.4, we conclude that with probability at least $1 - 1/\operatorname{poly}(n, d)$ we have

$$\left\|\overline{\boldsymbol{X}}_{\pi} - \pi \circ \overline{\boldsymbol{X}}_{\mathrm{GD}}\right\|_{2,\infty} \leq O\left(\frac{\epsilon\sqrt{d}}{1-\epsilon}\right).$$

Hence, by taking $\epsilon = O(\frac{\gamma}{\sqrt{d}})$, we see that

$$\left\|\overline{\boldsymbol{X}}_{\pi} - \pi \circ \overline{\boldsymbol{X}}_{\mathrm{GD}}\right\|_{2,\infty} \leq O(\gamma).$$

Plugging this choice of ϵ back into our definition of B, we conclude that when $B = \Omega(\frac{d\log(nd)}{\gamma^2})$, $\overline{\boldsymbol{X}}_{\pi}$ concentrates around $\pi \circ \overline{\boldsymbol{X}}_{\mathrm{GD}}$ within distance γ. □

C.3. Rank of batch normalized features

In this section we prove Proposition 3.2.1, which states that for batch sizes greater than 2, Assumption 2 implies that the batch normalized dataset will be full-rank (and hence full-dimensional) almost surely.

One shift in perspective that is fruitful for proving linear independence is to view batch normalization as an operation that returns functions of the input dataset. Since BN operates independently on each of the d features, we first handle the case where the input is a batch of scalars. Let $\binom{[n]}{B}$ denote the set of all $\binom{n}{B}$ batches of size B that can be created from n datapoints contained in $\boldsymbol{X}$. Fix an arbitrary labelling of these $\binom{n}{B}$ batches, and let $\boldsymbol{B}^j$ refer to the jth such batch. WLOG suppose that $\boldsymbol{B}^1 = \{x_1, \ldots, x_B\} \in \mathbb{R}^B$.

Formally, let $\mathcal{F}^B \triangleq \left\{f : \mathbb{R}^B \setminus \left\{\boldsymbol{x} \in \mathbb{R}^B \mid x_1 = x_2 = \cdots = x_B\right\} \to \mathbb{R}\right\}$ denote the space of real valued functions on batches of size B where BN is defined. On batch $\boldsymbol{B}^j = \{x_{j_1}, \ldots, x_{j_B}\}$, BN is an operation that maps this batch to the set of B functions $\left\{g_i^j(\boldsymbol{B}^j)\right\}_{i=1}^B$, where

$$g_i^j(\boldsymbol{B}^j) \triangleq \frac{x_{j_i} - \mu^j}{\sigma^j} \in \mathcal{F}^B$$

where μ^j and σ^j are the empirical mean and standard deviation, respectively of $\boldsymbol{B}^j$. If j is clear from context, we may drop the superscript j without chance of confusion. We also sometimes abuse notation and write g_i^j as a function of $\boldsymbol{X}$, since $\boldsymbol{X}$ contains all datapoints in $\boldsymbol{B}^j$. From this perspective, BN_{π} maps a dataset of n datapoints to n functions.

We first show that, within a batch, the functions have rank $B - 1$ over $\mathbb{R}$.

Lemma C.3.1. *Viewed as functions, any subset of $B - 1$ functions of the batch normalized outputs $\left\{g_i^1\right\} = \left\{\frac{x_1 - \mu}{\sigma}, \ldots, \frac{x_B - \mu}{\sigma}\right\}$ have rank $B - 1$ over $\mathbb{R}$.*

Proof. First, we note that the functions x_i are linearly independent over $\mathbb{R}$ for $i \in [n]$. WLOG take the subset of $B - 1$ functions to be $\left\{g_i^1\right\}_{i=1}^{B-1}$. Suppose that there is a dependence relationship

$$\sum_{i=1}^{B-1} c_i \frac{x_i - \mu}{\sigma} = 0.$$

Rearranging, we see that

$$\sum_{i=1}^{B-1} c_i x_i = \frac{1}{B}\left(\sum_{t=1}^{B} x_t\right)\left(\sum_{i=1}^{B-1} c_i\right)$$

and because of linear independence of the x_i's as functions over $\mathbb{R}$ and only the right-hand side contains x_B, the only way this can happen is if $\sum_{i=1}^{B-1} c_i = 0$. But then we have a dependence relationship between the x_i's for $i \leq B - 1$. Linear independence of the x_i's thus implies that $c_i = 0$ for each i. □

With the above lemma in hand, we can show that as functions, any collection of batch normalized outputs are essentially full rank. The caveat is we need to throw out one function in each batch, as each batch is trivially dependent because of the zero mean constraint.

Proposition C.3.2. *Let $B > 2$. Consider the $B\binom{n}{B}$ functions that are the batch normalized outputs of all $\binom{n}{B}$ batches $\boldsymbol{B}^j \in \binom{[n]}{B}$. If we take any subset of $(B-1)\binom{n}{B}$ of these functions obtained by removing one function from each of the $\binom{n}{B}$ batches, then the rank of these functions over $\mathbb{R}$ is $(B-1)\binom{n}{B}$. In particular, the rank of the functions corresponding to any π is $(B-1)\frac{n}{B}$.*

Proof. Consider the batch $\boldsymbol{B}^1 = \{x_1, \dots, x_B\}$. By Lemma C.3.1, we know that the functions $\{g_i^1\}_{i=1}^{B-1} = \{\frac{x_i - \mu}{\sigma}\}_{i=1}^{B-1}$ are linearly independent. Consider a dependence relation amongst any subset of the $(B-1)\binom{n}{B}$ described in the theorem statement. WLOG we can suppose that this was formed by throwing out g_B^j from each batch $\boldsymbol{B}^j$ and consider a dependence relation between the $(B-1)\binom{n}{B}$ remaining functions. The dependence relation reads

$$\sum_{i=1}^{B-1} c_{i,1} \frac{x_i - \mu}{\sigma} = \sum_{i=1}^{B-1} \sum_{j>1} c_{i,j} g_i^j. \tag{35}$$

We show that some setting of the input x_i for $i \in [n]$ yields a contradiction unless $c_{i,1} = 0$ for $i \in [B]$. The main insight is that BN has jump discontinuities at the points where it is undefined, i.e., where the entire batch is equal to the same thing.

More formally, for $i > B$ we set the other datapoints x_i to be arbitrary pairwise distinct positive real numbers. Suppose for the sake of contradiction that $c_{i,1} \neq 0$ for some $i \in [B]$; WLOG we can assume that $i = 1$. Since the functions on the LHS are linearly independent by Lemma C.3.1, the LHS is not identically zero. We show that the LHS of Equation (35) exhibits discontinuous behavior in the punctured neighborhood around $(x_1, \dots, x_B) = \mathbf{0}$, whereas the RHS of Equation (35) is continuous on the same neighborhood; this yields a contradiction.

Indeed, set $(x_1, x_2, x_3, \dots, x_B) = (\epsilon, \epsilon^2, -\epsilon - \epsilon^2, 0, \dots, 0)$, where $\epsilon \neq 0$. We have

$$\frac{x_1 - \mu}{\sigma} = \frac{\sqrt{B}\epsilon}{\sqrt{\epsilon^2 + \epsilon^4 + (\epsilon + \epsilon^2)^2}}.$$

If $\epsilon \to 0^+$, then the the first normalized coordinate approaches $+\sqrt{\frac{B}{2}}$. However, if $\epsilon \to 0^-$, then the first normalized coordinate approaches $-\sqrt{\frac{B}{2}}$. This is a contradiction, since the RHS of Equation (35) is continuous as a function of ϵ. We conclude that $c_{i,1} = 0$ for all $i \in [B-1]$.

The same argument holds if we replace the LHS with any batch $\boldsymbol{B}^j$. Hence, $c_{i,j} = 0$ for all $i \in [B-1]$ and $j \in [\binom{n}{B}]$. We conclude the rank of these $(B-1)\binom{n}{B}$ functions is $(B-1)\binom{n}{B}$, as desired. Notice also that this argument also shows that the functions corresponding to the batches in π also have rank $(B-1)\frac{n}{B}$. □

Note that the assumption that $B > 2$ is critical for the construction in the proof. If $B = 2$, then actually $\frac{x_i - \mu}{\sigma} \in \{\pm 1\}$, and the proof breaks down. The batch normalized dataset will be a Boolean matrix; hence, its rank cannot be analyzed by using density arguments.

The following lemma establishes that the zero set of any nontrivial linear combination of the g_i^j's is a measure zero subset of $\mathbb{R}^n$.

Lemma C.3.3. *Suppose that $c_{i,j}$ are not identically zero. Then for $B > 2$, the zero set of $\sum_{i=1}^{B-1} \sum_{j=1}^{\binom{n}{B}} c_{i,j} g_i^j$ is a measure zero subset of $\mathbb{R}^n$.*

Proof. Since $B > 2$, Proposition C.3.2 implies that $f(\boldsymbol{X}) \triangleq \sum_{i=1}^{B-1} \sum_{j=1}^{\binom{n}{B}} c_{i,j} g_i^j$ is not identically zero. Furthermore, $f(\boldsymbol{X})$ is real analytic on finitely many connected components of $\mathbb{R}^n$, since each of the functions g_i^j are real analytic on finitely many connected components of $\mathbb{R}^n$. The claim follows by applying Proposition 0 in Mityagin (2015). □

Having established these results, we can finally prove Proposition 3.2.1.

Proof of Proposition 3.2.1. Denote

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{X}_1 \\ \vdots \\ \boldsymbol{X}_d \end{bmatrix} \in \mathbb{R}^{d\times n},$$

and the $(B-1)\binom{n}{B}$ functions

$$\boldsymbol{g}_i^j(\boldsymbol{X}) = \begin{bmatrix} g_{i,1}^j(\boldsymbol{X}_1) \\ \vdots \\ g_{i,d}^j(\boldsymbol{X}_d) \end{bmatrix} \in \mathbb{R}^d$$

We can assemble these vector valued functions into a matrix $\boldsymbol{G}(\boldsymbol{X})$ of size $d \times (B-1)\binom{n}{B}$ with the kth row consisting of the scalar valued functions $g_{i,k}^j(\boldsymbol{X})$. Now consider the determinant of any $\min\left\{d, (B-1)\frac{n}{B}\right\} \times \min\left\{d, (B-1)\frac{n}{B}\right\}$ submatrix of these scalar functions, which is itself a function of $\boldsymbol{X}$.

To prove the claim, it suffices to show that this determinant — which is a function of $\boldsymbol{X} \in \mathbb{R}^{d\times n}$ — is analytic almost everywhere and not identically zero; then Lemma C.3.3 implies that it vanishes on a measure zero set of $\mathbb{R}^{d\times n}$.

We prove the above claim by induction on d. For $d = 1$, note that for any i, j, the scalar function $g_{i,1}^j(\boldsymbol{X})$ is not identically zero and is analytic on finitely connected components of $\mathbb{R}^{d\times n}$. Now suppose the claim is true for d, we prove the claim for $d+1$. When we use cofactor expansion along the first row of $\boldsymbol{G}$, which has functions $g_{i,1}^j$ which crucially only depend on the $\boldsymbol{X}_1$, we obtain

$$\sum_{i=1}^{B-1}\sum_{j=1}^{\binom{n}{B}} (-1)^{i+j} g_{i,1}^j(\boldsymbol{X}_1)\det(\boldsymbol{M}_{i,j}(\boldsymbol{X})),$$

where $\boldsymbol{M}_{i,j}$ denotes the minor corresponding to the the ith function of batch j. Note that these minors are not functions of $\boldsymbol{X}_1$, so they can be treated as constants with respect to $\boldsymbol{X}_1$. By induction these constants are nonzero almost surely. Then Proposition C.3.2 implies that the determinant, which is a linear combination of the functions $g_{i,1}^j$ is itself is not zero identically. Also, the determinant is manifestly piecewise analytic on finitely connected components of $\mathbb{R}^{(d+1)\times n}$, being a polynomial of such functions. Hence the induction is completed.

We can now finish off the proof of the proposition. When Assumption 2 holds, the probability that the data falls in measure zero set is a probability zero event. In other words, almost surely any such $d \times (B-1)\binom{n}{B}$ matrix constructed by batch normalizing and throwing out one normalized point in each batch is full rank. The same argument holds for the $(B-1)\frac{n}{B}$ functions that correspond to the batch normalized outputs for a permutation π. Hence the proposition is proved. □

D. Additional experiments on real data

In this section we provide detailed explanations of our experimental setup and present our additional experiments for regression and classification. All experiments were implemented in PyTorch 1.12.

D.1. Experiment details

We first define the architectures used in our real data experiments outlined in Section 4.3.

For the linear+BN networks, the 1-layer network is

$$\boldsymbol{X} \mapsto \boldsymbol{W}_1\boldsymbol{\Gamma}_1\mathsf{BN}(\boldsymbol{X}),$$

On the other hand, the 2-layer network is

$$\boldsymbol{X} \mapsto \boldsymbol{W}_2\boldsymbol{\Gamma}_2\mathsf{BN}(\boldsymbol{W}_1\boldsymbol{X})$$

and the 3-layer network is

$$\boldsymbol{X} \mapsto \boldsymbol{W}_3\boldsymbol{\Gamma}_3\mathsf{BN}(\boldsymbol{W}_2\boldsymbol{\Gamma}_2\mathsf{BN}(\boldsymbol{W}_1\boldsymbol{X})).$$

Hence the difference between the 1-layer network and deeper network is that the deeper networks have tunable parameters inside of BN.

For the MLP experiments, the 3-layer network is

$$\boldsymbol{X} \mapsto \boldsymbol{W}_3 \operatorname{ReLU}(\boldsymbol{\Gamma}_3 \mathsf{BN}(\boldsymbol{W}_2 \operatorname{ReLU}(\boldsymbol{\Gamma}_2 \mathsf{BN}(\boldsymbol{W}_1 \boldsymbol{X})))).$$

For the ResNet18 experiments, we used the ResNet18 architecture available through PyTorch, using `ResNet18_Weights.DEFAULT` pretrained weights.

The linear and BN layers were all initialized using the default PyTorch initialization. For the linear+BN networks, the linear layers were instantiated with a width of 512 and `bias=False`. For the 3 layer MLP, the linear layers were instantiated with a width of 512 and `bias=True`. The BN layers were instantiated with `track_running_stats=False`. As alluded to in Section 1.2, to evaluate the training GD risk $\mathcal{L}_{\mathrm{GD}}$ in the eval loop, we passed in the entire dataset as a single batch, thus avoiding EMA altogether. Except for the ResNet18 experiments, the images in the dataset were flattened into vectors.

Except for in the respective batch size and momentum ablation study (Figures 11 and 12), we used batch size $B = 128$ and no momentum. Note that for all of the datasets, $\log_2 n \approx 16$, which suggests that we are in the asymptotic regime where divergence can happen as stated in Theorem 4.1.3.

We now explain the difference in divergence behavior between the 1-layer and deeper linear+BN networks for SS. As suggested by Theorem 4.1.3, divergence can happen if the separability decomposition of $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ is not robust to perturbation. In the 1-layer case, the data remains far from being linearly separable. Meanwhile, in the deeper case, the network is incentivized to train the parameters inside BN such that the final features (e.g., $\mathsf{BN}_\pi(\boldsymbol{W}_1 \boldsymbol{X})$) are closer to being LS. But the nonlinearity of BN is not enough to make $\mathsf{BN}_\pi(\boldsymbol{W}_1 \boldsymbol{X})$ robustly LS. This also explains why introducing nonlinear activations prevents the divergence phenomenon.

We also note that in reality RR is run for T epochs. Thus, if $T < \frac{\binom{n}{B}}{\frac{n}{B}}$, the optimization routine only sees a proper subset of $\overline{\boldsymbol{Z}}_{\mathrm{RR}}$. However, there are other forces that help ensure that the subset of $\overline{\boldsymbol{Z}}_{\mathrm{RR}}$ actually seen during optimization is SC and satisfies Assumption 1(b). For example, it is likely that the algorithm sees non-monochromatic batches that also cause the hulls to overlap. One way this can happen is if $\mathbf{0}$ is in the relative interiors of the convex hulls of the monochromatic portion of each batch. Moreover, with extremely high probability we never see a repeat batch, so by Proposition B.1.3 the rank of the subset of $\overline{\boldsymbol{Z}}_{\mathrm{RR}}$ is w.h.p. equal to $T \cdot \min\left\{d, (B-1)\frac{n}{B}\right\}$. For us, $d = 10 \cdot 512$, $B = 128$, and $n \geq 50000$, so the rank of the subset of $\overline{\boldsymbol{Z}}_{\mathrm{RR}}$ we see easily outstrips the dimensionality of the final linear layers. This ensures that Assumption 1(b) holds.

In Figure 11, we see that divergence on 3 layer linear+BN networks generally occurs for large batch sizes, which corroborates Theorem 4.1.3. These batch sizes were picked because they are common choices for batch sizes in practice. For the largest batch size ($B = 128$), there does not appear to be divergence within 1000 epochs, which we address below.

In Figure 12, we see that the presence of momentum preserves divergence for SS, and in some cases accelerates it. Note that for each stepsize we used the same permutation. For the $\eta = 10^{-4}$ experiment, although the 0 momentum run did not start to diverge within 1000 epochs, the 0.9 and 0.99 momentum runs started to diverge. This further lends evidence to the claim that the apparent reason for no divergence for $\eta = 10^{-4}$ without momentum is that the small learning rate leads to slower convergence to an optimal direction for $\mathcal{L}_\pi$.

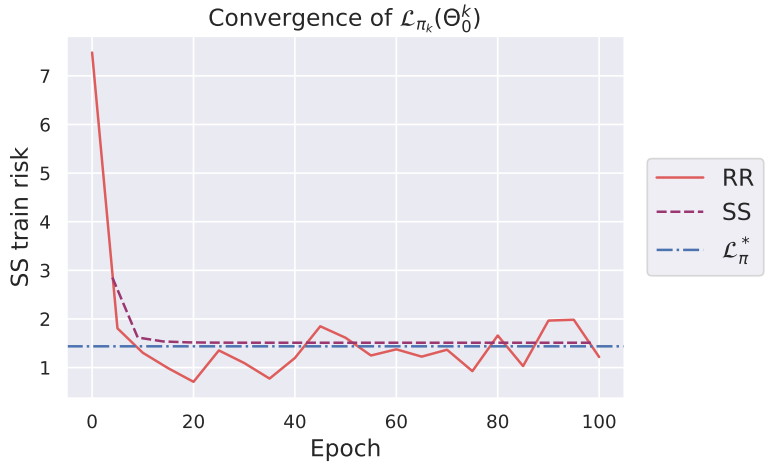


Figure 7. Loss evolution for $\ell_{\pi_k}(\boldsymbol{M}(k))$ for experiment described in Section 3.4. Note how SS converges to $\mathcal{L}_\pi^*$; RR oscillates due to resampling π_k but appears to converge to a value close to $\mathcal{L}_\pi^*$. This corroborates the convergence results Theorems 3.2.2 and 3.2.3 to distorted optima.

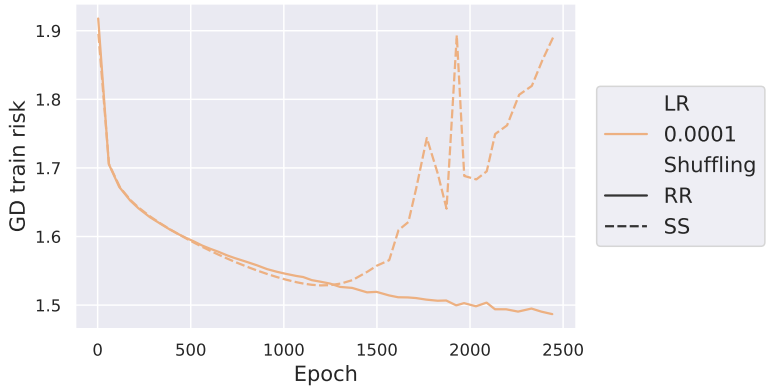


Figure 8. Eventual separation between SS and RR for 3 layer linear+BN network for $\eta = 10^{-4}$ after around epoch 1200 on CIFAR10. The color is consistent with the original Figure 3b.

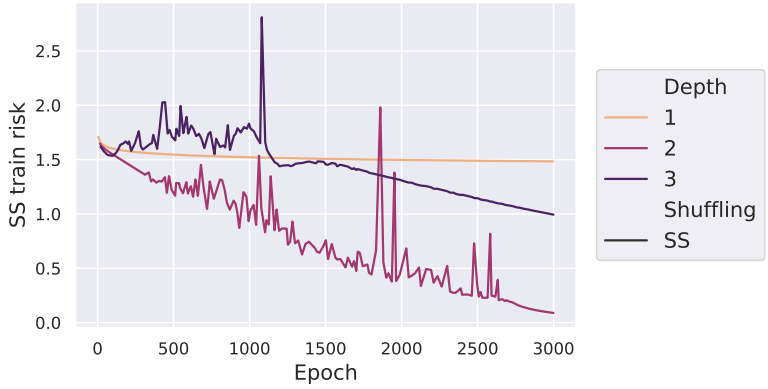


Figure 9. Evidence that the 2 layer and 3 layer linear+BN actually had diverging GD risks when trained with SS. On the y-axis we plot the value of $\mathcal{L}_\pi(\Theta_0^k)$; divergence occurs when the SS features $\overline{\boldsymbol{X}}_\pi$ are LS. As the SS risk for both 2 and 3 layer networks continues to decrease, this supports $\overline{\boldsymbol{X}}_\pi$ being LS. However, the SS risk for the 1 layer network seems to plateau, suggesting $\overline{\boldsymbol{X}}_\pi$ is SC rather than LS.

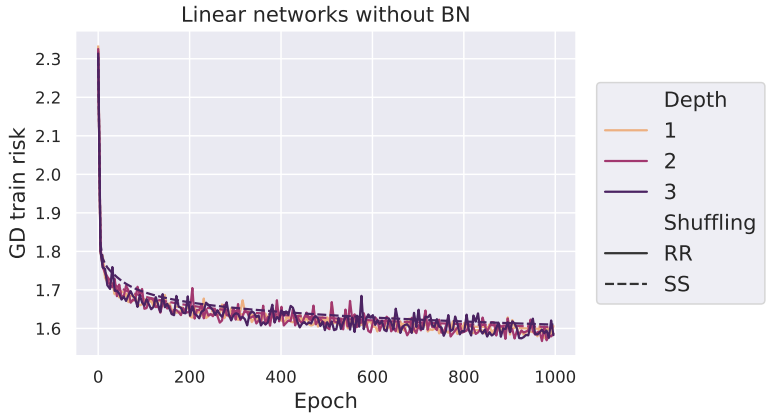


Figure 10. No divergence occurs for 1, 2, and 3 layer linear networks without BN trained with SS. This supports the theory that the strange training behavior occurs when using SS and BN in combination, rather than being an intrinsic failing of SS (cf. Figure 3b). Here we picked $\eta = 10^{-2}$, because we generally observed faster divergence with larger learning rates.

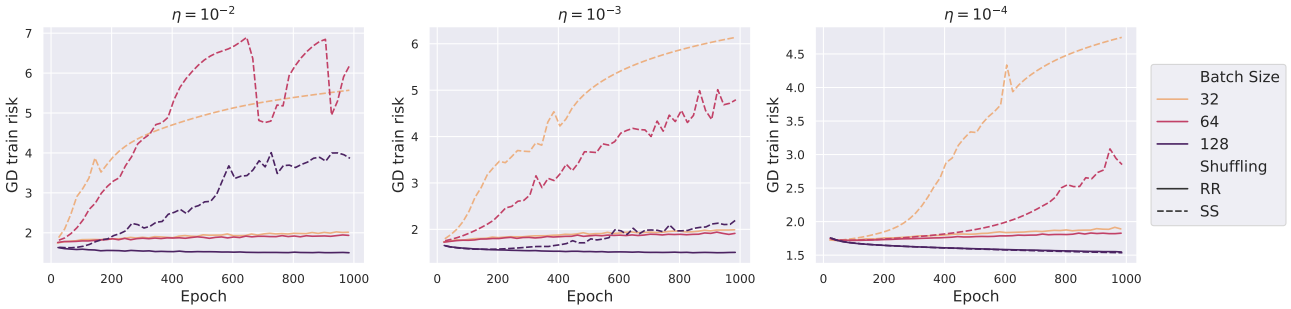


Figure 11. Batch size ablation. These experiments were done on 3 layer linear+BN networks; each subfigure shows the results of training with different stepsizes $\eta \in \{10^{-2}, 10^{-3}, 10^{-4}\}$. All experiments were performed on CIFAR10. Since $\log n = \log 50000 \approx 16$, all batch sizes are in the regime where divergence can happen according to Theorem 4.1.3.

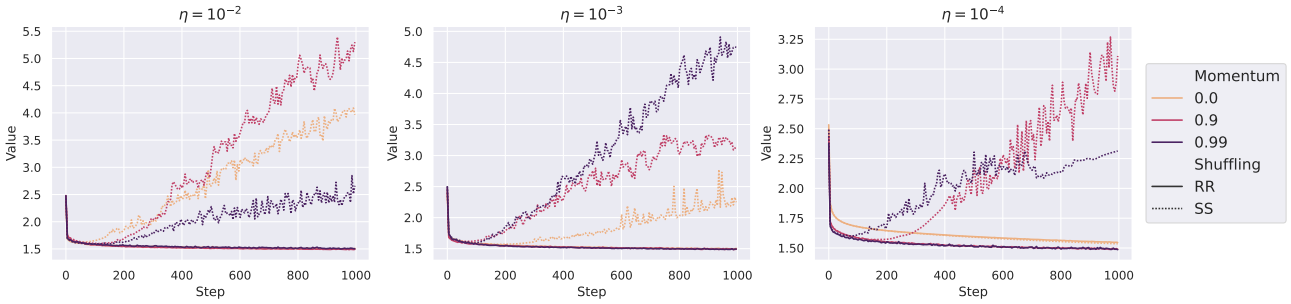


Figure 12. Momentum ablation. These experiments were done on 3 layer linear+BN networks; each subfigure shows the results of training with different stepsizes $\eta \in \{10^{-2}, 10^{-3}, 10^{-4}\}$. All experiments were performed on CIFAR10. For each subfigure, the experiment was run with the same random permutation. Generally, mmomentum doesn't prevent divergence, and in some cases even hastens it.

E. Calculations for toy datasets

In this section we do the detailed calculations and provide additional figures to understand the toy datasets we introduced in Sections 3.3 and 4.2. Since both constructions use $B = 2$, we remind the reader that the batch normalization of any two distinct real numbers is $(-1, 1)$. It follows that if we batch normalize with $B = 2$, we obtain $\overline{\boldsymbol{X}}_\pi \in \{-1, 1\}^{d\times n}$. Recall the distorted risk $\mathcal{L}_\pi$ reflects the single-shuffle batch normalized dataset $\overline{\boldsymbol{Z}}_\pi$. This $\mathcal{L}_\pi$ is a *random* quantity, and it is over this source of randomness (the construction of the batches) that we show that there is a gap between SS, RR, and GD.

E.1. Regression toy dataset

Proposition E.1.1. *There exists a regression dataset $\boldsymbol{Z} = (\boldsymbol{X}, \boldsymbol{Y}) \in [-1, 1]^{1\times 16n} \times [-1, 1]^{1\times 16n}$ such that the following statements hold with batch size $B = 2$:*

(1) $\boldsymbol{M}^*_{\mathrm{GD}} = \boldsymbol{M}^*_{\mathrm{RR}} = 0$.

(2) $\boldsymbol{M}^*_\pi \neq 0$ *with probability at least* $1 - O(\frac{1}{\sqrt{n}})$.

(3) $|\boldsymbol{M}^*_\pi| = \Omega(\frac{1}{\sqrt{n}})$ *with constant probability.*

We first describe the construction and then prove that the dataset satisfies the properties outlined above.

Formal construction of regression dataset: Construct the dataset as follows. Take $\boldsymbol{A} \in \mathbb{R}^{1\times 4n}$ to be $4n$ equally spaced points in the interval $(\frac{3}{4}, 1)$. Define

$$\boldsymbol{X}^1 = \boldsymbol{A}; \qquad \boldsymbol{X}^2 = -\boldsymbol{A}$$
$$\boldsymbol{X}^3 = -\boldsymbol{A} + \frac{1}{2}\mathbf{1}^\top; \qquad \boldsymbol{X}^4 = \boldsymbol{A} - \frac{1}{2}\mathbf{1}^\top$$

and

$$\boldsymbol{Y}^1 = \mathbf{1}^\top; \qquad \boldsymbol{Y}^2 = \mathbf{1}^\top$$
$$\boldsymbol{Y}^3 = -\mathbf{1}^\top; \qquad \boldsymbol{Y}^4 = -\mathbf{1}^\top.$$

Notice that the indices also match which quadrant the cluster of points are in. Visually, these four groups of points $\boldsymbol{Z}^i \triangleq (\boldsymbol{X}^i, \boldsymbol{Y}^i)$ are clusters with the ith cluster in the ith quadrant. For brevity, we also refer to these clusters by their index i only, so cluster 1 refers to the cluster $\boldsymbol{Z}^1 = (\boldsymbol{X}^1, \boldsymbol{Y}^1)$, and so on. These definitions are consistent with the clusters depicted in Figure 13a.

Then take $\boldsymbol{X} = \begin{bmatrix}\boldsymbol{X}^1 & \boldsymbol{X}^2 & \boldsymbol{X}^3 & \boldsymbol{X}^4\end{bmatrix} \in \mathbb{R}^{1\times 16n}$ and $\boldsymbol{Y} = \begin{bmatrix}\boldsymbol{Y}^1 & \boldsymbol{Y}^2 & \boldsymbol{Y}^3 & \boldsymbol{Y}^4\end{bmatrix} \in \mathbb{R}^{1\times 16n}$. After applying BN with permutation π and batch size 2, we obtain a dataset $\overline{\boldsymbol{X}}_\pi$ with every point being located in one of four SS clusters $\overline{\boldsymbol{Z}}^i_\pi \triangleq (\overline{\boldsymbol{X}}^i_\pi, \boldsymbol{Y}^i_\pi)$ for $i \in [4]$ located at $(\pm 1, \pm 1)$ with the same relative labelling: $\overline{\boldsymbol{Z}}^1_\pi$ is located at $(1, 1)$ and then labelling counterclockwise (see Figure 13b).

In Figure 13a, we visualize the construction with $n = 3$ (so the depicted dataset has $16n = 48$ datapoints). We plot the slopes of the $\boldsymbol{M}^*_{\mathrm{GD}}$ (green solid line), $\boldsymbol{M}^*_{\mathrm{RR}}$ (purple dash-dotted line), and typical values for $\boldsymbol{M}^*_\pi$ (yellow dotted line). In Figure 13b, we show what $\overline{\boldsymbol{Z}}_\pi$ looks like for a typical permutation π. The sizes of the points represent the number of points that end up in the corresponding cluster.

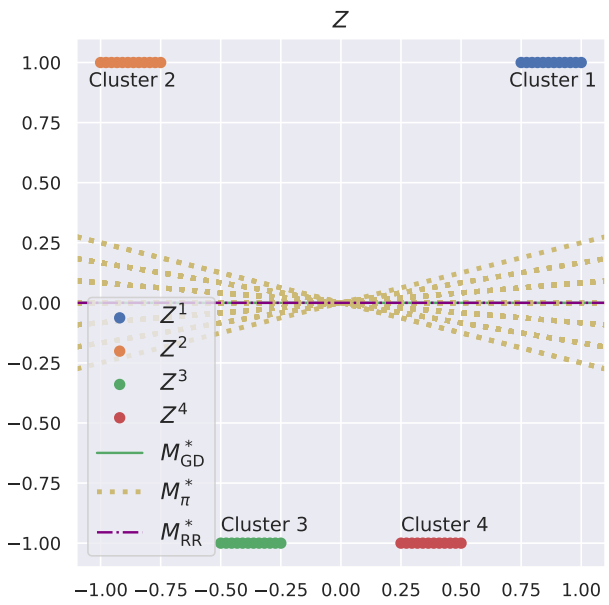
E.1.1. Analyzing the regression toy dataset

In order to prove this proposition, we will need the following standard technical estimate on the asymptotics of binomial coefficients (see e.g. Thomas & Joy (2006))

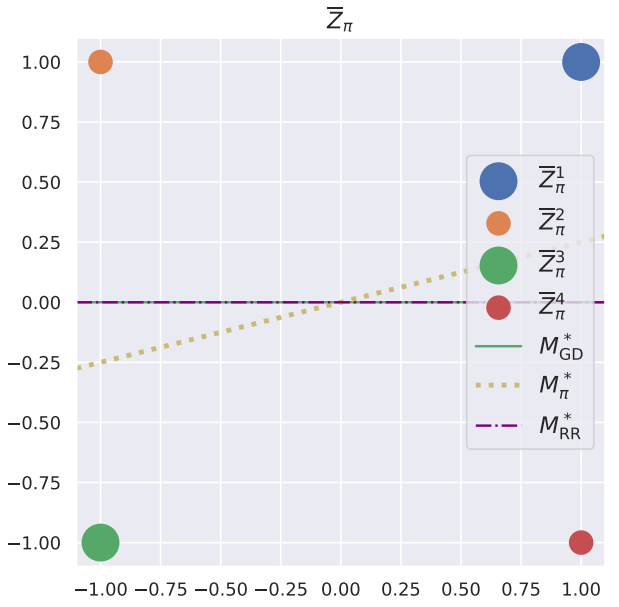
Lemma E.1.2. *For all n and k we have*

$$\sqrt{\frac{n}{8k(n-k)}}2^{H(k/n)n} \leq \binom{n}{k} \leq \sqrt{\frac{n}{\pi k(n-k)}}2^{H(k/n)n},$$

where $H(p) \triangleq -p\log_2 p - (1-p)\log_2(1-p)$ is the binary entropy function.



(a) Unnormalized toy regression dataset $\boldsymbol{Z}$ demonstrating distortion of SS with constant probability. Notice how the GD and RR lines are aligned with slope 0, but the SS lines are distorted away from 0.



(b) Toy regression dataset after BN with permutation π. The size of the point corresponding to $\overline{\boldsymbol{Z}}^i_\pi$ represents the number of normalized points in $\overline{\boldsymbol{Z}}^i_\pi$ (not to scale), and here we picked π such that $\boldsymbol{M}^*_\pi = \frac{1}{4}$.

Now let us prove the prove the proposition.

Proof of Proposition E.1.1. First, it is clear from the symmetry of $\boldsymbol{X}$ that $\boldsymbol{Y}_{\text{GD}}\overline{\boldsymbol{X}}^\top_{\text{GD}} = 0$, so $\boldsymbol{M}^*_{\text{GD}} = 0$. This proves the first half of (1).

Setup: For $i \in [4]$, let k^i_π denote the number of points that end up in cluster $\overline{\boldsymbol{Z}}^i_\pi$ after normalizing with permutation π. Since we have $\boldsymbol{M}^*_\pi = \frac{1}{16n}\boldsymbol{Y}_\pi\overline{\boldsymbol{X}}^\top_\pi$, evidently the k^i_π completely determine $\boldsymbol{M}^*_\pi$. In fact, more is true: we claim that $\boldsymbol{M}^*_\pi = 0$ if and only if $k^1_\pi = 4n$.

To see why this is true, first note that the label $\boldsymbol{Y}$ is unaffected by BN. Hence, there are necessarily $8n$ points that end up with y coordinate 1. It follows that if there are k points in $\overline{\boldsymbol{Z}}^1_\pi$, then there are $8n - k$ points in $\overline{\boldsymbol{Z}}^2_\pi$. Similarly, if there are j in $\overline{\boldsymbol{Z}}^3_\pi$, then there are $8n - j$ in $\overline{\boldsymbol{Z}}^4_\pi$. On the other hand, recall that BN with $B = 2$ and $d = 1$ always sends one point in each batch to $x = +1$ and one point to $x = -1$. Hence, there are $8n$ points that end up with x coordinate 1, which means $k + 8n - j = 8n$, implying that $k = j$.

Referring back to the formula for $\boldsymbol{M}^*_\pi$, we see that

$$\begin{aligned}\boldsymbol{M}^*_\pi &= \frac{1}{16n}\sum_{i=1}^{4}\boldsymbol{Y}^i_\pi(\overline{\boldsymbol{X}}^i_\pi)^\top \\ &= \frac{1}{16n}(k - (8n-k) + k - (8n-k)) \\ &= \frac{k-4n}{4n}.\end{aligned}$$

Hence $\boldsymbol{M}^*_\pi = 0$ if and only if $k^1_\pi = 4n$. Referring back to Figure 13b, we see that the sizes of the clusters represent k^i_π, and the plotted $\boldsymbol{M}^*_\pi$ with slope $\frac{1}{4}$ corresponds to a π where $k^1_\pi = 5n$.

To analyze k^1_π, for $i, j \in [4]$ we introduce the random variables $T^{\{i,j\}}_\pi$ to denote the number of batches formed with one point from cluster i and cluster j with permutation π. Similarly let $U \triangleq \{1, 2\}$ denote the upper clusters and $L \triangleq \{3, 4\}$

denote the lower clusters. Define $T_\pi^{\{i,L\}} \triangleq T_\pi^{\{i,3\}} + T_\pi^{\{i,4\}}$, which represents the total number of batches with one point in cluster i and another point in L when using permutation π. Finally, define $T_\pi^{\{U,L\}} \triangleq T_\pi^{\{1,L\}} + T_\pi^{\{2,L\}}$, which represents the total number of batches with one point in cluster U and another in L with permutation π.

With this notation in hand, let us prove the claim. Evidently, $k_\pi^1 = T_\pi^{\{U,U\}} + T_\pi^{\{1,L\}}$. Similarly we have $k_\pi^2 = T_\pi^{\{U,U\}} + T_\pi^{\{2,L\}}$. Because we established earlier that $k_\pi^1 + k_\pi^2 = 8n$, then $k_\pi^1 = 4n$ if and only if $T_\pi^{\{1,L\}} = T_\pi^{\{2,L\}}$. In words, this means that $\boldsymbol{M}_\pi^* = 0$ if and only if the number of $\{1, L\}$ batches is the same as the number of $\{2, L\}$ batches.

RR averages out distortion: For every π we can find π' such that $\boldsymbol{M}_{\pi'}^* = -\boldsymbol{M}_\pi^*$. This is because we can always find π' that swap $T_\pi^{\{1,L\}}$ and $T_\pi^{\{2,L\}}$, by turning all $\{1, L\}$ batches into $\{2, L\}$ batches and vice versa. Then Proposition 3.3.1 implies that $\boldsymbol{M}_{\mathrm{RR}}^* = 0$, which proves the second half of (1).

SS is distorted: Now, let us show that $\mathbb{P}[k_\pi^1 = 4n] = O(\frac{1}{\sqrt{n}})$. The main idea is that conditioned on $T_\pi^{\{U,L\}} = 2t$, we can compute the probability that $T_\pi^{\{1,L\}} = T_\pi^{\{2,L\}} = t$ exactly. Indeed, of the $4n$ points in cluster 1, we pick t of them to form batches with L, and similarly for cluster 2. This gives $\binom{4n}{t}^2$ ways for $T_\pi^{\{1,L\}} = T_\pi^{\{2,L\}} = t$. In total, there are $8n$ points in U and we picked $2t$ of them to match with L, which gives a denominator of $\binom{8n}{2t}$. Hence

$$\mathbb{P}[k_\pi^1 = 4n | T_\pi^{\{U,L\}} = 2t] = \frac{\binom{4n}{t}^2}{\binom{8n}{2t}}.$$

In order to obtain the $O(\frac{1}{\sqrt{n}})$ bound we desire, we need to use the fact that $T_\pi^{\{U,L\}}$ — the number of batches between U and L — concentrates tightly. In fact, if we color the 4 clusters corresponding to membership in U and L and slightly generalize the analysis leading to Fact C.1.1, we obtain that for some absolute constant C, we have $\left|T_\pi^{\{U,L\}} - 4n\right| \le 2C\sqrt{n \log n}$ with probability at least $1 - 1/n$.

Applying Lemma E.1.2, we obtain for all t such that $|t - 2n| \le 2C\sqrt{n \log n}$, we have

$$\mathbb{P}[k_\pi^1 = 4n | T_\pi^{\{U,L\}} = 2t] = O\left(\frac{\frac{n}{t(4n-t)} 2^{8H(\frac{t}{4n})n}}{\sqrt{\frac{n}{t(4n-t)}} 2^{8H(\frac{t}{4n})n}}\right) \tag{36}$$

$$= O\left(\sqrt{\frac{n}{t(4n-t)}}\right) \tag{37}$$

$$= O\left(\frac{1}{\sqrt{n}}\right). \tag{38}$$

Hence we have

$$\begin{aligned}
\mathbb{P}[k_\pi^1 = 4n] &= \sum_{t=0}^{4n} \mathbb{P}[k_\pi^1 = 4n | T_\pi^{\{U,L\}} = 2t] \, \mathbb{P}[T_\pi^{\{U,L\}} = 2t] \\
&\le \frac{1}{n} + \sum_{|t-2n| \le C\sqrt{n \log n}} \mathbb{P}[k_\pi^1 = 4n | T_\pi^{\{U,L\}} = 2t] \, \mathbb{P}[T_\pi^{\{U,L\}} = 2t] \\
&\le \frac{1}{n} + O\left(\frac{1}{\sqrt{n}}\right) \\
&\le O\left(\frac{1}{\sqrt{n}}\right),
\end{aligned}$$

where in the second line we have used the union bound along with the fact that $T_\pi^{\{U,L\}}$ concentrates, and in the third line we have used Equation (38). This proves (2).

Quantitative SS distortion bounds with constant probability: Finally, we show (3). Suppose that $k_\pi^1 = 4n + d$ for $|d| = O(\sqrt{n})$. The above analysis for the case of $d = 0$ immediately generalizes to show that, if $t - d > 0$,

$$\mathbb{P}[k_\pi^1 = 4n + d | T_\pi^{\{U,L\}} = 2t - d] = \frac{\binom{4n}{t}\binom{4n}{t-d}}{\binom{8n}{2t-d}}.$$

Notice that since $2t - d$ concentrates around $4n$, it suffices to only consider the high probability regime where $2t - d = 4n + O(\sqrt{n \log n})$. In particular, since $|d| = O(\sqrt{n})$, we have $t - d = O(t)$.

Thus, if we plug in Lemma E.1.2, we obtain in the regime where $t - d = O(t)$ that

$$\frac{\binom{4n}{t}\binom{4n}{t-d}}{\binom{8n}{2t-d}} = O\left(\frac{1}{\sqrt{t}}\right) 2^{4n\left[H(\frac{t}{4n}) + H(\frac{t-d}{4n}) - 2H(\frac{2t-d}{8n})\right]}.$$

Concavity of binary entropy implies that $H(\frac{t}{4n}) + H(\frac{t-d}{4n}) - 2H(\frac{2t-d}{8n}) \le 0$. It follows that

$$\mathbb{P}[k_\pi^1 = 4n + d | T_\pi^{\{U,L\}} = 2t - d] = O\left(\frac{1}{\sqrt{t}}\right).$$

Following the same argument as in the $d = 0$ case, we have for $|d| = O(\sqrt{n})$ that

$$\begin{aligned}
\mathbb{P}[k_\pi^1 = 4n + d] &= \sum_{2t-d=0}^{4n} \mathbb{P}[k_\pi^1 = 4n | T_\pi^{\{U,L\}} = 2t - d] \, \mathbb{P}[T_\pi^{\{U,L\}} = 2t - d] \\
&\le \frac{1}{n} + \sum_{\left|t - \frac{d}{2} - 2n\right| \le C\sqrt{n \log n}} \mathbb{P}[k_\pi^1 = 4n + d | T_\pi^{\{U,L\}} = 2t - d] \, \mathbb{P}[T_\pi^{\{U,L\}} = 2t - d] \\
&\le \frac{1}{n} + O\left(\frac{1}{\sqrt{n}}\right) \\
&\le O\left(\frac{1}{\sqrt{n}}\right),
\end{aligned}$$

From here, it follows that there exists some positive constant c such that

$$\mathbb{P}[\left|k_\pi^1 - 4n\right| > c\sqrt{n}] = \Omega(1),$$

which proves (3). □

E.2. Classification toy dataset

In this section, we motivate how we constructed our toy classification dataset parameterized by n. We then give a detailed construction and analysis of the dataset, parameterized by n. We plot the unnormalized $\boldsymbol{Z}$ in Figure 14a and the normalized $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ in Figure 14b for $n = 10$.

The main idea is that since $d = 2$ and the optimal direction is orthogonal to the SC portion of the separability decomposition (Lemma 4.1.1), we can fix the optimal directions of $\overline{\boldsymbol{Z}}_\pi$ and $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ by carefully constraining $\mathrm{Span}(\overline{\boldsymbol{X}}_\pi^{\mathrm{SC}})$ and $\mathrm{Span}(\overline{\boldsymbol{X}}_{\mathrm{GD}}^{\mathrm{SC}})$, respectively. For, $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ we carefully select the boundary points $\boldsymbol{X}_{\mathrm{bdr}}$ which define the boundary of $\mathrm{conv}(\overline{\boldsymbol{X}}_{\mathrm{GD}}^+)$ and $\mathrm{conv}(\overline{\boldsymbol{X}}_{\mathrm{GD}}^-)$ so that $\dim(\mathrm{Span}(\overline{\boldsymbol{X}}_{\mathrm{GD}}^{\mathrm{SC}})) = 1$. For $\overline{\boldsymbol{Z}}_\pi$, we need to ensure that $\mathrm{Span}(\overline{\boldsymbol{X}}_\pi^{\mathrm{SC}})$ is a one dimensional subspace of $\mathbb{R}^2$ which is close to orthogonal with $\mathrm{Span}(\overline{\boldsymbol{X}}_{\mathrm{GD}}^{\mathrm{SC}})$. Although we can guarantee $\dim(\mathrm{Span}(\overline{\boldsymbol{X}}_\pi^{\mathrm{SC}})) \ge 1$ w.h.p., the main subtlety here is ensuring that equality holds with constant probability. Given the above, we are afforded the luxury of adding datapoints which are misclassified by $\boldsymbol{v}_\pi^*$, the optimal direction of $\mathcal{L}_\pi$.

Proposition E.2.1. *There exists a classification dataset $\boldsymbol{Z} = (\boldsymbol{X}, \boldsymbol{Y}) \in [-3, 3]^{2\times(2n+6)} \times \{-1, 1\}^{2n+6}$ such that the following statements hold with batch size $B = 2$:*

(1) *$\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ is PLS and as $n \to \infty$, $\boldsymbol{v}_{\mathrm{GD}}^*$ converges in direction to $\begin{bmatrix}1 & 2\end{bmatrix}^\top$.*

(2) *$\overline{\boldsymbol{Z}}_\pi$ is PLS with constant probability. If so, we have* $\boldsymbol{v}_\pi^* = \begin{bmatrix}1 & -1\end{bmatrix}^\top$.

(3) *There exists points $(\boldsymbol{x}_i, y_i) \in \overline{\boldsymbol{Z}}_{\mathrm{GD}}$ such that $y_i\langle \boldsymbol{v}_\pi^*, \boldsymbol{x}_i\rangle < 0$, i.e., GD points that $\boldsymbol{v}_\pi^*$ misclassifies.*

Hence, the GD risk $\mathcal{L}_{\mathrm{GD}}$ diverges with constant probability if we train with SS.

Proof. In our construction, we separate out $\boldsymbol{Z} = (\boldsymbol{X}, \boldsymbol{Y})$ into several groups of points: $\boldsymbol{X}_{\mathrm{cor}}^+$, $\boldsymbol{X}_{\mathrm{err}}^+$, $\boldsymbol{X}_{\mathrm{bdr}}^+$, and their negative variants. Let the overlined version of these matrices denote the corresponding points after full-batch BN (i.e., after taking $\overline{\boldsymbol{X}}_{\mathrm{GD}} \triangleq \mathsf{BN}(\boldsymbol{X})$), and the overlined version with an extra π subscript denoting the corresponding points after BN with permutation π (i.e., $\overline{\boldsymbol{X}}_\pi \triangleq \mathsf{BN}_\pi(\boldsymbol{X})$). For example, $\overline{\boldsymbol{X}}_{\mathrm{cor}}^+$ denotes the features for the positive examples in $\overline{\boldsymbol{X}}_{\mathrm{GD}}$ that are to be classified correctly by $\boldsymbol{v}_\pi^*$, whereas $\overline{\boldsymbol{X}}_{\mathrm{cor},\pi}^+$ denotes those same datapoints in $\overline{\boldsymbol{X}}_\pi$ after batch normalization with π.

Setup: Let us first explain the semantic meanings of the different groups of points in our construction.

$$
\begin{aligned}
\boldsymbol{X}_{\mathrm{cor}}^+ &: \text{Unnormalized positive examples correctly classified by } \boldsymbol{v}_\pi^* \text{ with positive margin}\\
\boldsymbol{X}_{\mathrm{err}}^+ &: \text{Unnormalized positive example incorrectly classified by } \boldsymbol{v}_\pi^*\\
\boldsymbol{X}_{\mathrm{bdr}}^+ &: \text{Unnormalized positive examples on the decision boundary of } \boldsymbol{v}^*\\
\overline{\boldsymbol{X}}_{\mathrm{cor}}^+ &: \text{full-batch-normalized positive examples correctly classified by } \boldsymbol{v}_\pi^* \text{ with positive margin}\\
\overline{\boldsymbol{X}}_{\mathrm{err}}^+ &: \text{full-batch-normalized positive example incorrectly classified by } \boldsymbol{v}_\pi^*\\
\overline{\boldsymbol{X}}_{\mathrm{bdr}}^+ &: \text{full-batch-normalized positive examples on the decision boundary of } \boldsymbol{v}_{\mathrm{GD}}^*
\end{aligned}
$$

The semantic meanings of the negative versions of these points are completely analogous. We also define $\boldsymbol{X}_{\mathrm{bdr}} = \boldsymbol{X}_{\mathrm{bdr}}^+ \cup \boldsymbol{X}_{\mathrm{bdr}}^-$, and the normalized quantity analogously.

We construct $\boldsymbol{X}_{\mathrm{cor}}^+$, $\boldsymbol{X}_{\mathrm{err}}^+$, and $\boldsymbol{X}_{\mathrm{bdr}}^+$ as follows. Take $\boldsymbol{X}_{\mathrm{cor}}^+$ to be n equally spaced points on the line segment of width $\frac{1}{n}$ centered at $\begin{bmatrix}2 & 2\end{bmatrix}^\top$. For the sake of visual clarity, we increase the spacing of the points in the diagram Figure 14a. Define $\boldsymbol{X}_{\mathrm{err}}^+$ to be $\begin{bmatrix}3 & 2.5\end{bmatrix}^\top$. Next, define $\boldsymbol{X}_{\mathrm{bdr}}^+$ to be $\begin{bmatrix}-3 & 1\\ 1.5 & -0.5\end{bmatrix}$, lying on the line $y = -0.5x$. Finally, define

$$
\begin{aligned}
\boldsymbol{X}_{\mathrm{cor}}^- &= -\boldsymbol{X}_{\mathrm{cor}}^+\\
\boldsymbol{X}_{\mathrm{err}}^- &= -\boldsymbol{X}_{\mathrm{cor}}^+\\
\boldsymbol{X}_{\mathrm{bdr}}^- &= -\boldsymbol{X}_{\mathrm{bdr}}^+.
\end{aligned}
$$

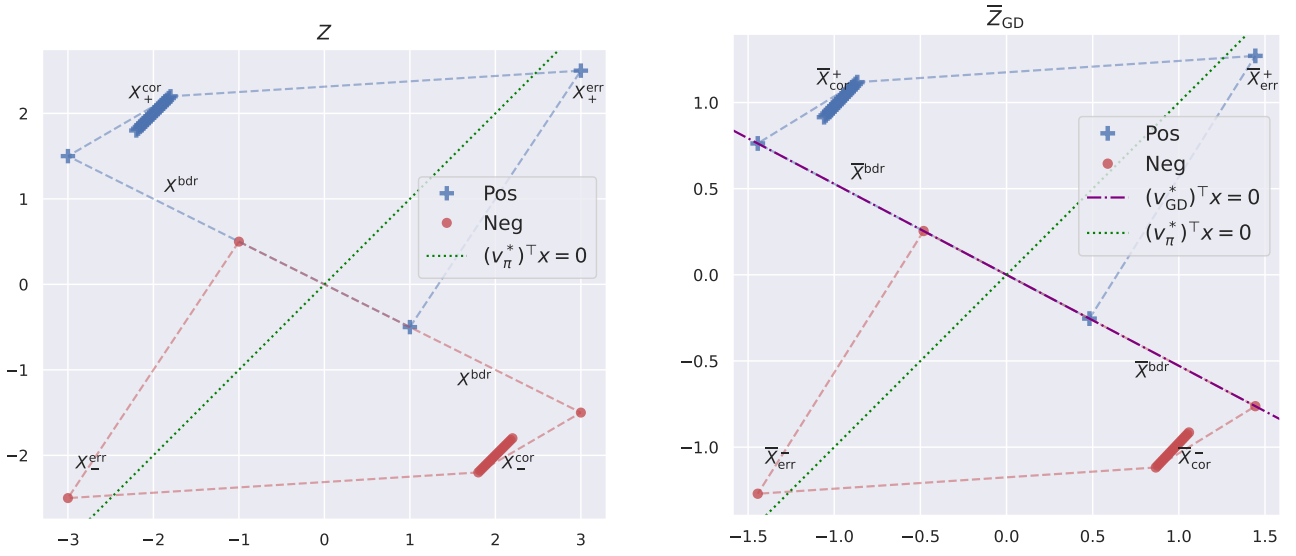
In Figure 14, we visualize this toy dataset. Note the visual similarity between $\boldsymbol{Z}$ in Figure 14a and $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ in Figure 14b; this is a feature of the construction. Indeed, as we'll see shortly, as $n \to \infty$, $\overline{\boldsymbol{X}}_{\mathrm{GD}}$ approaches a uniform rescaling of $\boldsymbol{X}$ in all coordinates. We also plotted the decision boundaries corresponding to $\boldsymbol{v}_{\mathrm{GD}}^*$ and $\boldsymbol{v}_\pi^*$. We highlight the fact that in Figure 14b, $\overline{\boldsymbol{X}}_{\mathrm{err}}^+$ and $\overline{\boldsymbol{X}}_{\mathrm{err}}^-$ are both on the wrong side of the decision boundary for $\boldsymbol{v}_\pi^*$.

GD is PLS: Evidently $\boldsymbol{\mu} = \boldsymbol{0}$ and one can compute that as $n \to \infty$ that $\boldsymbol{\sigma} \to \begin{bmatrix}2 & 2\end{bmatrix}^\top$. Regardless, we see that $\overline{\boldsymbol{Z}}_{\mathrm{GD}}$ is PLS with $\mathrm{Span}(\overline{\boldsymbol{X}}_{\mathrm{GD}}^{\mathrm{SC}})$ corresponding to the one-dimensional subspace spanned by $\overline{\boldsymbol{X}}_{\mathrm{bdr}}$ and $\boldsymbol{v}_{\mathrm{GD}}^*$ correctly classifies all of the other points. In the limit $n \to \infty$, we have that $\mathrm{Span}(\overline{\boldsymbol{X}}_{\mathrm{bdr}}) = \mathrm{Span}(\begin{bmatrix}-2 & 1\end{bmatrix}^\top)$ and $\boldsymbol{v}_{\mathrm{GD}}^*$ is in the direction $\begin{bmatrix}1 & 2\end{bmatrix}^\top$. This proves (1).

SS is PLS with constant probability: Now, let us compute what happens to $\overline{\boldsymbol{Z}}_\pi$. Because BN with $B = 2$ sends features to ± 1, this implies that a normalized batch is either mapped to

(1) $\begin{bmatrix}-1 & +1\\ -1 & +1\end{bmatrix}$, i.e. the normalized batch lies in the direction $\begin{bmatrix}+1 & +1\end{bmatrix}^\top$; or

(2) $\begin{bmatrix}-1 & +1\\ +1 & -1\end{bmatrix}$, i.e the normalized batch lies in the direction $\begin{bmatrix}+1 & -1\end{bmatrix}^\top$.



(a) Unnormalized toy classification dataset $\boldsymbol{Z}$.

(b) Normalized toy classification dataset $\overline{\boldsymbol{Z}}_\pi$ demonstrating divergence of SS with constant probability.

Figure 14. Toy classification dataset (a) before full-batch BN, i.e. $\boldsymbol{Z}$ (b) after full-batch BN, i.e. $\overline{\boldsymbol{Z}}_{\text{GD}}$.

Note that due to Fact C.1.1, with high probability, there will be a batch drawn from $\boldsymbol{X}_{\text{cor}}^+$ and a batch drawn from $\boldsymbol{X}_{\text{cor}}^-$. These batches necessarily land in situation (1), so $\begin{bmatrix}1 & 1\end{bmatrix}^\top \in \text{Span}(\overline{\boldsymbol{X}}_\pi^{\text{SC}})$ with high probability. On the other hand, with high probability there will be a batch with one point from $\boldsymbol{X}_{\text{cor}}^+$ and one from $\boldsymbol{X}_{\text{cor}}^-$, which lands in (2). Hence, as long as $\text{Span}(\overline{\boldsymbol{X}}_\pi^{\text{SC}}) = \text{Span}(\begin{bmatrix}1 & 1\end{bmatrix}^\top)$, this implies that $\overline{\boldsymbol{Z}}_\pi$ is PLS with optimal direction $\begin{bmatrix}-1 & +1\end{bmatrix}^\top$.

By inspection, to ensure that $\text{Span}(\overline{\boldsymbol{X}}_\pi^{\text{SC}}) = \text{Span}(\begin{bmatrix}1 & 1\end{bmatrix}^\top)$, there are two bad events we need to avoid:

(a) We send a positive example to $\begin{bmatrix}+1 & -1\end{bmatrix}^\top$.

(b) We send a negative example to $\begin{bmatrix}-1 & +1\end{bmatrix}^\top$.

We will use

$$\mathbb{P}[\text{avoid (a) and (b)}] = \mathbb{P}[\text{avoid (a)} \mid \text{avoid (b)}]\,\mathbb{P}[\text{avoid (b)}].$$

Note that by symmetry, $\mathbb{P}[\text{avoid (b)}] = \mathbb{P}[\text{avoid (a)}]$.

A little thought reveals that (a) can happen only if the positive boundary example (i.e. in $\boldsymbol{X}_{\text{bdr}}^+$) located originally at $\begin{bmatrix}1 & -0.5\end{bmatrix}^\top$ is batched together with a point in $\boldsymbol{X}_{\text{bdr}} \cup \boldsymbol{X}_{\text{cor}}^+$. In turn, this event occurs with probability at most $\frac{2}{3}$. So $\mathbb{P}[\text{avoid (b)}] \geq \frac{1}{3}$. Also, notice avoiding (a) still happens with probability at least $\frac{1}{3}$ even after conditioning on avoiding (b). Hence, the probability that we avoid both (a) and (b) is at least $\frac{1}{3}^2 = \frac{1}{9}$. So with constant probability, $\text{Span}(\overline{\boldsymbol{X}}_\pi^{\text{SC}}) = \text{Span}(\begin{bmatrix}1 & 1\end{bmatrix}^\top)$. This proves (2).

Putting it all together, we see that with constant probability, $\text{Span}(\overline{\boldsymbol{X}}_\pi^{\text{SC}}) = \text{Span}(\begin{bmatrix}1 & 1\end{bmatrix}^\top)$, and $\overline{\boldsymbol{Z}}_\pi$ is PLS with optimal direction $\boldsymbol{v}_\pi^* = \begin{bmatrix}+1 & -1\end{bmatrix}^\top$. Recall that $\boldsymbol{v}_{\text{GD}}^* \to \begin{bmatrix}1 & 2\end{bmatrix}^\top$ as $n \to \infty$. So asymptotically we have

$$\frac{|\langle \boldsymbol{v}_\pi^*, \boldsymbol{v}_{\text{GD}}^* \rangle|}{\|\boldsymbol{v}_\pi^*\| \|\boldsymbol{v}_{\text{GD}}^*\|} = \frac{1}{\sqrt{10}}.$$

SS misclassifies GD points: On the constant probability event that $\overline{\boldsymbol{Z}}_\pi$ is PLS, we have $\boldsymbol{v}_\pi^*$ is in the direction $\begin{bmatrix}1 & -1\end{bmatrix}^\top$. This misclassifies $\overline{\boldsymbol{X}}_{\text{err}}^+$ and $\overline{\boldsymbol{X}}_{\text{err}}^-$. So this proves (3). □

Remark E.2.2. *Note that at the cost of visual clarity, we can modify the construction to obtain optimal classifiers $\boldsymbol{v}_\pi^*$ and $\boldsymbol{v}_{\mathrm{GD}}^*$ which are asymptotically orthogonal. In this alternate construction, we take $\boldsymbol{X}_{\mathrm{bdr}}^+ = \begin{bmatrix} -1 & 0.5 \\ 1 & -0.5 \end{bmatrix}$, lying on the line $y = -x$, and $\boldsymbol{X}_{\mathrm{cor}}^+$ to be n equally spaced points on the line segment between $\begin{bmatrix} -2 & 2 \end{bmatrix}^\top$ and $\begin{bmatrix} -2 + \frac{1}{n} & 2 + \frac{1}{n} \end{bmatrix}^\top$. Then $\boldsymbol{v}_{\mathrm{GD}}^* \to \begin{bmatrix} 1 & 1 \end{bmatrix}^\top$, and $\boldsymbol{v}_\pi^* = \begin{bmatrix} 1 & -1 \end{bmatrix}^\top$ with constant probability, as desired.*