ADAM IS NO BETTER THAN NORMALIZED SGD: DISSECTING HOW ADAPTIVE METHODS IMPROVE GANS PERFORMANCE

Anonymous authors
Paper under double-blind review

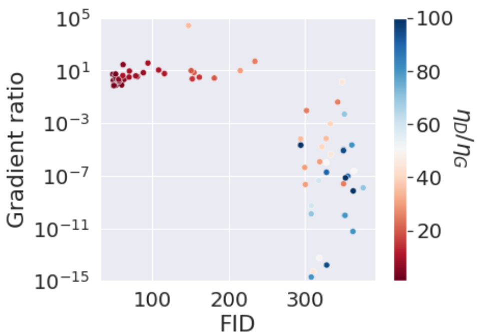
ABSTRACT

Adaptive methods are widely used for training generative adversarial networks (GAN). While there has been some work to pinpoint the marginal value of adaptive methods in minimization problems, it remains unclear why it is still the method of choice for GAN training. This paper formally studies how adaptive methods help performance in GANs. First, we dissect Adam—the most popular adaptive method for GAN training—by comparing with SGDA the direction and the norm of its update vector. We empirically show that SGDA with the same vector norm as Adam reaches similar or even better performance than the latter. This empirical study encourages us to consider normalized stochastic gradient descent ascent (nSGDA) as a simpler alternative to Adam. We then propose a synthetic theoretical framework to understand why nSGDA yields better performance than SGDA for GANs. In that situation, we prove that a GAN trained with nSGDA provably recovers all the modes of the true distribution. In contrast, the same networks trained with SGDA (and any learning rate configuration) suffers from mode collapsing. The critical insight in our analysis is that normalizing the gradients forces the discriminator and generator to update at the *same pace*. We empirically show the competitive performance of nSGDA on real-world datasets.

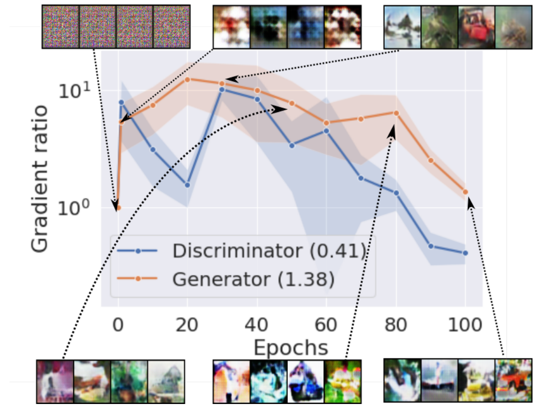
1 INTRODUCTION

It is commonly accepted that adaptive algorithms are required to train modern neural network architectures in various deep learning tasks. This includes minimization problems that arise in natural language processing (Vaswani et al., 2017) and fMRI (Zbontar et al., 2018) or min-max problems such as generative adversarial network (GAN) training (Goodfellow et al., 2014). Indeed, it has been empirically observed that Adam (Kingma & Ba, 2014) yields a solution with better generalization than stochastic gradient descent (SGD) in these problems (Choi et al., 2019). Several works have attempted to explain this phenomenon in the minimization case. Common explanations are that adaptive methods train faster (Zhou et al., 2018), escape faster very flat saddle-point like plateaus (Orvieto et al., 2021) or deal better with heavy-tailed stochastic gradients (Zhang et al., 2019b). However, much less is known regarding min-max problems such as GANs. In this paper, we investigate why GANs trained with adaptive methods outperform those trained using stochastic gradient descent ascent with momentum (SGDA).

Some prior works attribute this outperformance to the superior convergence speed of adaptive methods. For instance, Liu et al. (2019) show that a variant of Optimistic Gradient Descent (Daskalakis et al., 2017) converges faster than SGDA for a class of non-convex non-concave min-max problems. However, contrary to the minimization setting, convergence to a stationary point is not guaranteed and not even a requirement to ensure a satisfactory GAN performance. Indeed, Mescheder et al. (2018) empirically shows that popular architectures such as Wasserstein GANs (WGANs) (Arjovsky et al., 2017) do not always converge, and yet produce realistic images. We support this observation through the following experiment. We train a DCGAN (Radford et al., 2015) using Adam –the most popular adaptive method– and set up the generator (G) and discriminator (D) step-sizes respectively as η_D, η_G. Note that D is usually trained faster than G i.e. $\eta_D \geq \eta_G$. Figure 1(a) displays the GAN convergence measured by the ratio of gradient norms, and the GAN's performance measured in FID score (Heusel et al., 2017). We observe that when η_D/η_G is close to 1, the algorithm does not converge and yet, the model produces high-quality solutions. On the other hand, when $\eta_D/\eta_G \gg 1$, the model converges to an equilibrium –a similar statement has been proved by Jin et al. (2020) and Fiez & Ratliff (2020) in the case of SGDA. However, the GAN produces low-quality solutions at



(a) Each circle corresponds to a specific step-size configuration η_D/η_G. We observe that the best-performing models have step-size configurations close to 1 and do not converge. As η_D/η_G increases, the models perform worse and most of them manage to converge to an equilibrium.



(b) shows that during training, the gradient ratio of a well-performing GAN approximately stays constant to 1. We also display the images produced by the model during training.

Figure 1: Gradient ratio against FID score (a) and number of epochs (b) obtained with DCGAN on CIFAR-10. This ratio is equal to $\|\text{grad}_G^{(t)}\|_2/\|\text{grad}_G^{(0)}\|_2 + \|\text{grad}_D^{(t)}\|_2/\|\text{grad}_D^{(0)}\|_2$, where $\text{grad}_G^{(t)}$ (resp. $\text{grad}_D^{(t)}$) and $\text{grad}_G^{(0)}$ (resp. $\text{grad}_D^{(0)}$) are the current and initial gradients of G (resp. D). Note that $\|\cdot\|_2$ refers to the sum of all the parameters norm in a network. For all the plots, the models are trained for 100 epochs using a batch-size 64. For (b), the results are averaged over 5 seeds.

this equilibrium. Thus, simply comparing the convergence speed of adaptive methods and SGDA *cannot* explain the GAN's performance obtained with adaptive methods. This observation motivates the central question in this paper: *What factors explain that Adam produces better quality solutions than SGDA when training GANs?*

To address this question, we dissect Adam following the approach by Agarwal et al. (2020). They frame a generic optimizer's update as $W^{(t+1)} = W^{(t)} - \eta a^{(t)}\mathcal{G}^{(t)}$, where $W^{(t)} \in \mathbb{R}^d$ is the iterate, $\mathcal{G}^{(t)} \in \mathbb{R}^d$ such that $\|\mathcal{G}^{(t)}\|_2 = 1$ is the optimizer's *direction* and $a^{(t)} \geq 0$ is the optimizer's *magnitude*. Therefore, a first step in our paper is to understand whether Adam outperforms SGDA maintly due to its direction or to its magnitude. As detailed in Section 2, we train a GAN using i) AdaLR, an algorithm that updates in the direction of SGDA but with the magnitude of Adam ii) AdaDir which uses the direction of Adam but the magnitude of SGDA. We empirically show that not only, AdaLR significantly outperforms AdaDir, SGDA, and Adam itself. This observation encourages us to conclude that:

Adam produces higher quality solutions relative to SGDA in GANs mainly due to its adaptive magnitude and not to its adaptive direction.

In Section 2, we empirically analyze the adaptive magnitude of AdaLR and observe that it stays approximately constant throughout training. This observation eventually encourages the study of AdaLR with a constant step-size. Such algorithm actually corresponds to *normalized* SGDA (nSGDA).Compared to SGDA, nSGDA has the same direction but differs in magnitude since we divide the gradient by its norm. Intuitively, this normalization forces D and G to be updated by vectors with constant magnitudes *no matter how different* the norms of D's and G's gradients are.

Motivated by the aforementioned observations, this paper studies the performance of GANs trained with nSGDA. We believe that this is a first step to formally understand the role of adaptive methods in GANs. Our contributions are divided as follows:

- In Section 3, we experimentally confirm that nSDGA consistently competes with Adam and outperforms SGDA when using different GAN architectures on a wide range of datasets.
- In Section 4, we provide a theoretical explanation on why GANs trained with nSGDA outperform those trained with SGDA. More precisely, we devise a data generation problem where the target distribution $\mathcal{D}$ is made of multiple modes. The model trained with nSGDA provably recovers all the modes in the target distribution while the SGDA's one fails to do it under *any step-size configuration*: We prove that even when SGDA converges to a locally optimal min-max equilibrium, the model still *suffers from mode collapsing* and fails to learn recover the modes separately.

The key insight of our theoretical analysis is that no matter how the step-sizes are prescribed, D and G necessarily update at very different speeds when we use SGDA. Therefore, i) either D updates its weights too fast, thus learns a weighted average of the modes of $\mathcal{D}$. This makes G learn this weighted average of modes ii) or D does not update its weights fast enough and thus, G aligns its weights with those of D. This forces D to converge to a locally optimal min-max equilibrium that classifies any instance as "fake". On the other hand, by normalizing the gradients as done in SGDA, we *force* D and G to *update at the same speed throughout training*. Thus, whenever D learns a mode of the distribution, G learns it right after, which makes both of them learn all the modes of the distribution separately.

Our paper advocates for the use of *balanced updates* in GAN training i.e. the ratio of D vs G updates should remain close to constant. To our knowledge, we are the first to *theoretically* show the importance of these balanced updates. This insight contrasts with the related work that analyzes GANs, and more generally zero-sum differentiable games, in the regime where D is updated much faster than G i.e. $\eta_D/\eta_G \gg 1$ (Fiez & Ratliff, 2020; Jin et al., 2020; Fiez et al., 2020).

RELATED WORK

Adaptive methods in games optimization. Several works designed adaptive algorithms and analyzed their convergence to show their benefits relative to SGDA. For variational inequality problems, Gasnikov et al. (2019); Antonakopoulos et al. (2019); Bach & Levy (2019); Antonakopoulos et al. (2020) propose adaptive algorithms that reach optimal convergence rates under regularity assumptions. For a class of non-convex non-concave min-max problems, Liu et al. (2019); Barazandeh et al. (2021) design algorithms that converge faster than SGDA. Our work differs from these papers as we analyze Adam and do not focus on the convergence properties but rather on the fit of the trained model on the *true* (and not empirical) data distribution.

Statistical results in GANs. Early works investigate whether GANs memorize the training data or actually learn the distribution (Arora et al., 2017; 2018; Dumoulin et al., 2016). Zhang et al. (2017); Bai et al. (2018) then show that for specific GANs, the model learn some distributions with non-exponential sample complexity (Liang, 2017; Feizi et al., 2017). Recently, Li & Dou (2020); Allen-Zhu & Li (2021) further characterized the distributions learned by the generator. On the other hand, some works attempted to explain GAN performance through the optimization lens. Lei et al. (2020); Balaji et al. (2021) show that GAN models trained with SGDA converge to a global saddle point when the generator is one-layer neural network and the discriminator is a specific quadratic/linear function. Our contribution significantly differs from these two works as i) we construct a setting where SGDA converges to a locally optimal min-max equilibrium and yet suffer from mode collapse. Conversely, nSGDA does not necessarily converge and yet, recovers the true distribution ii) our setting is more challenging since we need at least a degree-3 discriminator to learn the distribution – see Section 4 for a justification.

Normalized gradient descent. Introduced by Nesterov (1984), normalized gradient descent has been widely used in the minimization setting. Indeed, it has been observed that normalizing out the gradient improves the 'slow crawling' problem of gradient descent and avoids the iterates to be stuck in flat regions – such as spurious local minima or saddle points – (Hazan et al., 2015; Levy, 2016; Murray et al., 2019). Normalized gradient descent or its variants outperform the non-normalized counterparts in multi-agent coordination (Cortés, 2006) and deep learning tasks (You et al., 2017; 2019; Cutkosky & Mehta, 2020; Liu et al., 2021). Our work rather considers the min-max setting and shows that nSGDA performs better than SGDA as it forces the discriminator and generator to update at the same rate.

2 FROM ADAM TO nSGDA

Generative adversarial networks. Given a training set sampled from some target distribution $\mathcal{D}$, a GAN learns to generate new data from this distribution. The architecture is constituted of two networks: the generator maps points in the latent space $\mathcal{D}_z$ to sample candidates of the desired distribution; the discriminator evaluates these samples by comparing them to samples from $\mathcal{D}$.

More formally, the generator is a mapping $G_{\mathcal{V}}\colon \mathbb{R}^k \to \mathbb{R}^d$ where $\mathcal{V}$ is some parameter set. Generally, the latent variables are sampled from the normal distribution. On the other hand, the discriminator is a mapping $D_{\mathcal{W}}\colon \mathbb{R}^d \to \mathbb{R}$ where $\mathcal{W}$ is some parameter set. In this section, one can think of these parameter sets as made of matrices and vectors. To train the model, we consider the WGAN-GP problem formulation (Gulrajani et al., 2017) (where $\widehat{\mathcal{D}} = \epsilon\mathcal{D} + (1-\epsilon)\mathcal{D}_z$ for $\epsilon > 0$),

$$\min_{\mathcal{V}} \max_{\mathcal{W}} \mathbb{E}_{z\sim\mathcal{D}_z}[D_{\mathcal{W}}(G_{\mathcal{V}}(z))] - \mathbb{E}_{x\sim\mathcal{D}}[D_{\mathcal{W}}(x)] + \lambda\mathbb{E}_{y\sim\widehat{\mathcal{D}}}[(\|\nabla_{\mathcal{W}} D_{\mathcal{W}}\|_2 - 1)^2] := f(\mathcal{V},\mathcal{W}). \quad (1)$$

Algorithm 1 Generic second-moment adaptive optimizer

Input: initial points $\mathcal{W}^{(0)}, \mathcal{V}^{(0)}$, step-size schedules $\{(\eta_G^{(t)}, \eta_D^{(t)})\}$, hyperparameters $\{\beta_1, \beta_2, \varepsilon\}$.
Initialize $\mathbf{M}_{\mathcal{W},1}^{(0)}$, $\mathbf{M}_{\mathcal{W},2}^{(0)}$, $\mathbf{M}_{\mathcal{V},1}^{(0)}$ and $\mathbf{M}_{\mathcal{V},2}^{(0)}$ to zero.
for $t = 0 \ldots T-1$ **do**
  Receive stochastic gradients $\mathbf{g}_{\mathcal{W}}^{(t)}, \mathbf{g}_{\mathcal{V}}^{(t)}$ evaluated at $\mathcal{W}^{(t)}$ and $\mathcal{V}^{(t)}$.
  Update accumulators for $\mathcal{Y} \in \{\mathcal{W}, \mathcal{V}\}$, $\ell \in [2]$: $\mathbf{M}_{\mathcal{Y},\ell}^{(t+1)} = \beta_\ell \mathbf{M}_{\mathcal{Y},\ell}^{(t)} + \mathbf{g}_{\mathcal{Y}}^{(t)}$.
  Compute gradient oracles for $Y \in \{V, W\}$: $\mathbf{A}_{\mathcal{Y}}^{(t+1)} = \frac{\mathbf{M}_{\mathcal{Y},1}^{(t+1)}}{\sqrt{\mathbf{M}_{\mathcal{Y},2}^{(t+1)} + \varepsilon}}$.
  Update:
$$\mathcal{W}^{(t+1)} = \mathcal{W}^{(t)} + \eta_D^{(t)} \mathbf{A}_{\mathcal{W}}^{(t+1)}, \qquad \mathcal{V}^{(t+1)} = \mathcal{V}^{(t)} - \eta_G^{(t)} \mathbf{A}_{\mathcal{V}}^{(t+1)}.$$
return $\mathcal{W}^{(T)}, \mathcal{V}^{(T)}$.

Adaptive methods. In this paper, we particularly focus on second-moment-based adaptive optimizers to solve (1). Algorithm 1 captures the usual formulations of Adam (Kingma & Ba, 2014), Adagrad (Duchi et al., 2011) and RMSprop (Tieleman et al., 2012) up to some scaling conventions that can be absorbed into $\{(\eta_G^{(k)}, \eta_D^{(k)})\}$. Note that all operations on vectors in Algorithm 1 are entry-wise and $\mathbf{g}^2$ denotes the entry-wise power on vector $\mathbf{g}$.

The exponential window parameters $\beta_1, \beta_2 \in [0, 1)$ respectively denote the first- and second-order momentum parameters. A rule of thumb is to set them as $\beta_1 = 0.5$ and $\beta_2 = 0.9$. As we aim to understand the role of adaptivity, an important baseline is stochastic gradient descent ascent with momentum (SGDA) which consists in the update of Algorithm 1when the denominator $\sqrt{M_{\mathcal{Y},2} + \epsilon}$ is omitted. *In what follows, we refer to SGDA as the SGDA with momentum update*.

Optimizers as autorefalg:adaptive are *adaptive* in that they keep updating step-sizes while training the model. Therefore, two types of schedules are involved: the *explicit step size schedules* $\{(\eta_G^{(t)}, \eta_D^{(t)})\}$ that the practitioner manually sets up and the *implicit step size schedules* induced by the optimizers. The contributions of these two schedules overlap and it remains unclear how they contribute to the superior performance of adaptive methods relative to SGDA in GANs.

Method dissection using grafting. To decouple the explicit and implicit schedules, we adopt the step size grafting approach proposed by Agarwal et al. (2020) and described as follows. At each iteration, we compute stochastic gradients, pass them to two optimizers $\mathcal{A}_1, \mathcal{A}_2$ and make a grafted step which combines the *magnitude* of $\mathcal{A}_1$'s step and *direction* of $\mathcal{A}_2$'s step. Since our goal is to understand the benefits of adaptivity, it is natural to consider the grafting approach with Adam and SGDA. We define *AdaLR*, an optimizer that updates in the SGDA direction with Adam magnitude:

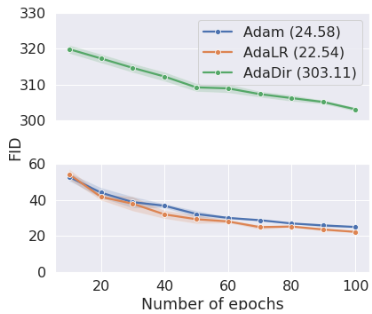
$$\mathcal{W}^{(t+1)} = \mathcal{W}^{(t)} + \eta_D^{(t)} \|\mathbf{A}_{\mathcal{W}}^{(t)}\|_2 \frac{\mathbf{M}_{\mathcal{W},1}^{(t)}}{\|\mathbf{M}_{\mathcal{W},1}^{(t)}\|_2 + \varepsilon}, \; \mathcal{V}^{(t+1)} = \mathcal{V}^{(t)} - \eta_G^{(t)} \|\mathbf{A}_{\mathcal{V}}^{(t)}\|_2 \frac{\mathbf{M}_{\mathcal{V},1}^{(t)}}{\|\mathbf{M}_{\mathcal{V},1}^{(t)}\|_2 + \varepsilon}, \quad (2)$$

and *AdaDir* which updates in the Adam direction with SGDA magnitude

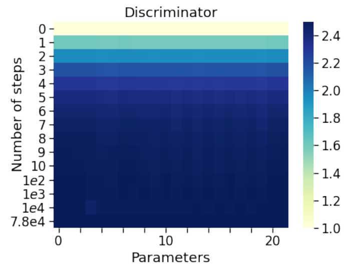
$$\mathcal{W}^{(t+1)} = \mathcal{W}^{(t)} + \eta_D^{(t)} \|\mathbf{M}_{\mathcal{W},1}^{(t)}\|_2 \frac{\mathbf{A}_{\mathcal{W}}^{(t)}}{\|\mathbf{A}_{\mathcal{W}}^{(t)}\|_2 + \varepsilon}, \; \mathcal{V}^{(t+1)} = \mathcal{V}^{(t)} - \eta_G^{(t)} \|\mathbf{M}_{\mathcal{V},1}^{(t)}\|_2 \frac{\mathbf{A}_{\mathcal{V}}^{(t)}}{\|\mathbf{A}_{\mathcal{V}}^{(t)}\|_2 + \varepsilon}. \quad (3)$$

Note that two implementations are possible for AdaLR and AdaDir. In the *layer-wise* version, $\mathcal{Y}^{(t)}$ is a single parameter group (typically a layer in a neural network) and therefore, the update is applied to each group. In the *global* version, $\mathcal{Y}^{(t)}$ contains all of the model's weights. Note that in Figure 2, we set up AdaLR and AdaDir with the layer-wise version –the global AdaLR and AdaDir perform approximately as well as their layer-wise counterparts.

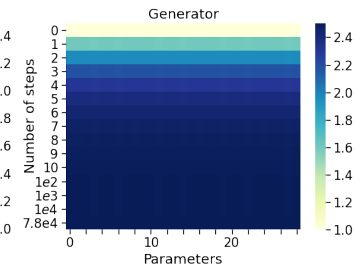
We train a WGAN-GP (Arjovsky et al., 2017) on CIFAR-10 with AdaLR, AdaDir and Adam. Figure 2(a) shows the GAN performance measured in FID score (Heusel et al., 2017) obtained by the three trained models. We observe that AdaDir does not generate samples from the desired distribution as its FID score is high. On the other hand, AdaLR performs slightly better than Adam.



(a)



(b)



(c)

Figure 2: (a) FID score against the number of epochs of a Resnet WGAN-GP trained on CIFAR-10 with Adam, AdaLR, and AdaDir. AdaLR performs slightly better than Adam while AdaDir performs very poorly. (b)-(c) displays the fluctuations of AdaLR's adaptive magnitude. We plot the ratio $\|\mathbf{A}_{\mathcal{Y}}^{(t)}\|_2/\|\mathbf{A}_{\mathcal{Y}}^{(0)}\|_2$ for each discriminator's (b) and generator's (c) parameters. At early stages, this ratio barely increases and remains constant after 10 steps. We train for 100 epochs using a batch-size 64 and results are averaged over 5 seeds.

Therefore, we deduce that the success of adaptive methods in GANs may be explained by the *implicit step-size schedule* induced by the algorithm.

From AdaLR to normalized SGDA. The previous experiment hints that AdaLR which combines Adam's magnitude and SGDA direction performs better than Adam. We here take a closer look at this algorithm. In particular, we investigate the fluctuations of Adam's magnitude during training. Figures 2(b),2(c) show that this magnitude barely varies for both generator and discriminator.

Therefore, an update with SGDA direction and constant step-size seems to be a valid approximation of Adam in GAN training. Such an algorithm corresponds to normalized SGDA (nSGDA),

$$\mathcal{W}^{(t+1)} = \mathcal{W}^{(t)} + \eta_D^{(t)} \frac{\mathbf{M}_{\mathcal{W},1}^{(t)}}{\|\mathbf{M}_{\mathcal{W},1}^{(t)}\|_2+\varepsilon}, \qquad \mathcal{V}^{(t+1)} = \mathcal{V}^{(t)} - \eta_G^{(t)} \frac{\mathbf{M}_{\mathcal{V},1}^{(t)}}{\|\mathbf{M}_{\mathcal{V},1}^{(t)}\|_2+\varepsilon}. \tag{4}$$

Similarly to AdaLR and AdaDir, we also define two versions of nSGDA: layer-wise nSGDA (l-nSGDA) and global nSGDA (g-nSGDA). We use both of these in the numerical experiments. As a comparison to (4), the SGDA (with momentum) update is:

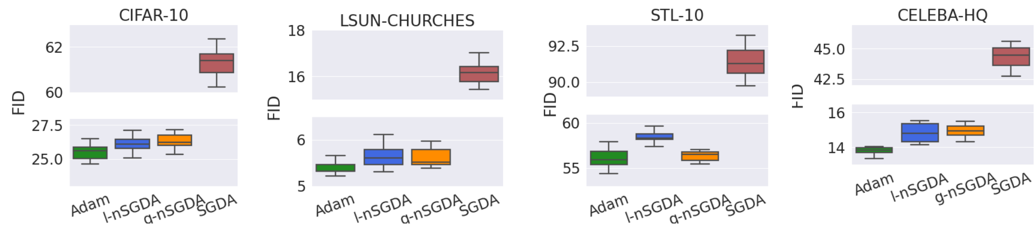
$$\mathcal{W}^{(t+1)} = \mathcal{W}^{(t)} + \eta_D^{(t)} \mathbf{M}_{\mathcal{W},1}^{(t)}, \qquad \mathcal{V}^{(t+1)} = \mathcal{V}^{(t)} - \eta_G^{(t)} \mathbf{M}_{\mathcal{V},1}^{(t)}. \tag{5}$$

3 Numerical performance of nSGDA

Section 2 indicates that nSGDA may work as well as Adam in GAN training. In this section, we empirically verify this hypothesis through an extensive numerical study. To evaluate the proposed algorithm, we conducted extensive experiments on CIFAR-10 (Krizhevsky et al., 2009), LSUN Churches (Yu et al., 2016), STL-10 (Coates et al., 2011) and Celeba-HQ (Liu et al., 2015). Similarly to above, we choose the Fréchet Inception distance (FID) (Heusel et al., 2017) to quantitatively assess the performance of the model. In all our experiments, 50k samples are randomly generated for each model to compute the FID. As for the architectures, we choose Resnets (He et al., 2016) from Gidel et al. (2018) and set up the WGAN-GP loss (Gulrajani et al., 2017). Note that for each optimizer, we grid-search over stepsizes to find the best one in terms of FID. Due to limited computational resources, we trained the models for 100 epochs in the case of CIFAR-10 and Celeba-HQ and for 50 epochs for LSUN and STL-10. We apply a linear decay learning rate scheduling during training. All the results are averaged over 10 seeds.

nSGDA competes with Adam. Figure 3 shows the performance of l-nSGDA, g-nSGDA, Adam, and SGD on different datasets. We first observe that l-nSGDA and g-nSGDA compete with Adam in all these datasets. On the other hand, SGD struggles to produce good quality images compared to the other methods as expected. It is worth reminding that the nSGDA methods and SGD have the *same direction* but *differ in their magnitude*. Therefore, this experiment confirms our hypothesis that adaptive methods outperform SGDA thanks to their implicit step-size schedule. Lastly, we highlight that l-nSGDA and g-nSGDA perform almost as well. This suggests that a global step-size adaptivity is enough to perform well in GANs. Similar observations hold for DCGAN (see Appendix A).

Influence of batch size. While the nSGDA methods compete with Adam in GAN, it remains unclear whether such performance is proper to the algorithm or tied to the stochastic noise of the gradients.



(a) CIFAR-10 (b) LSUN Churches (c) STL-10 (d) Celeba-HQ

Figure 3: FID scores obtained when training a Resnet WGAN-GP using Adam, l-nSGDA, g-nSGDA, and SGD on different datasets. In all these datasets, l-nSGDA, g-nSGDA and Adam perform approximately as well. SGDA performs much worse. The models are trained with batch-size 64.

Figure 4: FID score of a Resnet WGAN-GP model trained with l-nSGDA, g-nSGDA, Adam, and SGDA against the batch size in the CIFAR-10 dataset. At small batch sizes, the best performing models are those trained with nSGDA methods. As the batch size increases, the performance of nSGDA methods worsens and Adam performs better. Lastly, models trained with nSGDA consistently outperform those trained with SGDA.

Figure 4 shows the performance of the model trained with l-nSGDA, g-nSGDA, Adam, and SGD and using a wide range of batch sizes. We remark that the nSGDA methods perform better than Adam when the batch size is small. However, as the batch size increases, their performance seriously deteriorates. On the other hand, the performance of Adam seems to be less sensitive to the change of batch size. As expected, SGD overall performs poorly. nSGDA performs well in a relevant batch-size regime, as many GAN architectures such as DCGAN, WGAN-GP, SNGAN (Miyato et al., 2018), SAGAN (Zhang et al., 2019a) requires small batch sizes for optimal performances.

4 nSGDA CAN PROVABLY WORK BETTER THAN SGDA

In Section 2, we numerically observed that Adam outperforms SGDA mainly thanks to its adaptive magnitude (implicit step-size schedule). In Section 3, we observed that nSGDA methods – which differs from SGDA only by adaptive magnitude – compete with Adam and significantly surpass SGDA. To the best of our knowledge, there is no *theoretical* result that demonstrates the importance of adaptive magnitude in GANs performance. Thus, we set the following question:

From a theoretical perspective, for what distribution learning problem using GANs does nSGDA perform better comparing to SGDA?

We devise a simple data generation problem where the target distribution consists of two modes, described as vectors u_1, u_2, that are slightly correlated (See Assumption 1). We show that using standard GANs' training objective, with high probability, SGDA – with *any reasonable* [1] *step-size configuration* – only learn distributions that suffer from *mode collapse* i.e. u_1, u_2 always show up in the learned distribution simultaneously. Conversely, nSGDA learns the two modes *separately*.

Notations. We set the GAN 1-sample loss $L^{(t)}_{\mathcal{V},\mathcal{W}}(X,z) = \log(D^{(t)}_{\mathcal{W}}(X)) + \log(1 - D^{(t)}_{\mathcal{W}}(G^{(t)}_{\mathcal{V}}(z)))$. $\mathrm{g}^{(t)}_{\mathcal{Y}} = \nabla_{\mathcal{Y}} L^{(t)}_{\mathcal{V},\mathcal{W}}(X,z)$ is the 1-sample stochastic gradient (without momentum).

4.1 SETTINGS

In this section, we consider the classical GAN problem formulation.

$$\min_{\mathcal{V}} \max_{\mathcal{W}} \quad \mathbb{E}_{X\sim\mathcal{D}}[\log(D_{\mathcal{W}}(X))] + \mathbb{E}_{z\sim\mathcal{D}_z}[\log(1 - D_{\mathcal{W}}(G_{\mathcal{V}}(z)))]. \tag{6}$$

In the following paragraphs, we describe each element of our formulation.

Data distribution $\mathcal{D}$. The data generation problem consists in having a training dataset with points sampled from a target distribution $\mathcal{D}$. Using this dataset, our goal is to train a model that generates samples with the same statistics as $\mathcal{D}$. The latter is defined as follows.

[1] Here reasonable simply means that the learning rates are bounded to prevent the training to blow up.

Let u_1 and u_2 two vectors in $\mathbb{R}^d$ such that $\|u_i\|_2 = 1$, for $i \in \{1, 2\}$.
Each sample from $\mathcal{D}$ consists of an input data X generated as:

1. Sample $s = (s_1, s_2)$ from the distribution $\mathcal{S}$ defined as $s_i \in [0, 1]$, $\mathbb{P}[s_i = 1] \geq 1/2$ for $i \in \{1, 2\}$, such that $\|s\|_0 \geq 1$. Note that s_1 and s_2 are not necessarily independent.
2. Define data-point $X = s_1 u_1 + s_2 u_2 \in \mathbb{R}^d$.

The distribution $\mathcal{D}$ generates points that are a linear combination of two modes u_1 and u_2. In what follows, we make the following assumptions on the coefficients (s_i) and modes (u_i).

Assumption 1 (Data structure)**.** *Let $\gamma = \frac{1}{\mathrm{polylog}(d)}$. The coefficients s_1, s_2 and modes u_1, u_2 of the distribution $\mathcal{D}$ respect* one *of the following conditions:*

1. Correlated Modes: $\langle u_1, u_2 \rangle = \gamma$ and the generated data point is either $X = u_1$ or $X = u_2$.

2. Correlated Coefficients: $\mathbb{P}[s_1 = s_2 = 1] = \gamma$ and the modes are orthogonal, i.e., $\langle u_1, u_2 \rangle = 0$.

Assumption 1 captures some of the realistic structure of images. Case 1 models the setting where we have two pure modes (e.g., two different types of cats) that are correlated. Case 2 corresponds to two (roughly) orthogonal modes (e.g., vertical and horizontal edges), that may sometimes be mixed together (images containing object corners in the example above).

Learner models. To learn the true distribution $\mathcal{D}$, we use a linear generator $G_{\mathcal{V}}$ defined as

$$G_{\mathcal{V}}(z) = Vz = \sum_{i=1}^{m_G} v_i z_i, \tag{7}$$

where $V = [v_1^\top, v_2^\top, \cdots, v_{m_G}^\top] \in \mathbb{R}^{m_G \times d}$ is the weight matrix and $z \in \{0, 1\}^{m_G}$. We set $\mathcal{V} = \{V\}$. Intuitively, $G_{\mathcal{V}}$ outputs linear combinations of modes (v_i). We assume that $G_{\mathcal{V}}$ samples from the latent distribution $\mathcal{D}_z$ defined for $z \in \{0, 1\}^{m_G}$, $\|z\|_0 \geq 1$ as:

$$\forall i, j \in [m_G],\ \Pr[z_i = 1] = \Theta\left(\frac{1}{m_G}\right), \quad \Pr[z_i = z_j = 1] = \frac{1}{m_G^2 \mathrm{polylog}(d)} \tag{8}$$

First, remark that vectors sampled from $\mathcal{D}_z$ are binary-valued. Although usual latent distributions in GANs are either Gaussian or uniform, the distribution $\mathcal{D}_z$ should be rather seen as modelling the weights' distribution of a hidden layer of a deep generator. Indeed, Allen-Zhu & Li (2021) argue that the distributions of these hidden layers are sparse, non-negative, and non-positively correlated. Besides, in (8), $\Pr[z_i = z_j = 1] = \frac{1}{m_G^2 \mathrm{polylog}(d)}$ ensures that that the output of the generator is only made of one mode with probability $1 - o(1)$ and thus avoid creating weighted averages of the two or more v_i's (which might cause mode collapsing). To assess the distribution learned by $G_{\mathcal{V}}$, we also train a non-linear neural network as discriminator $D_{\mathcal{W}}$

$$D_{\mathcal{W}}(X) = \mathrm{sigmoid}\left(a \sum_{i \in [m_D]} \sigma(\langle w_i, X \rangle) + \lambda b\right), \quad \sigma(z) = \begin{cases} z^3 & \text{if } |z| \leq \Lambda \\ 3\Lambda^2 z - 2\Lambda^3 & \text{if } z > \Lambda \\ 3\Lambda^2 z + 2\Lambda^3 & \text{otherwise} \end{cases}, \tag{9}$$

where $W = [w_1^\top, \ldots, w_{m_D}^\top] \in \mathbb{R}^{m_D \times d}$ and $a, b \in \mathbb{R}$ are the trainable parameters of $D_{\mathcal{W}}$, $\lambda > 0$ is a fixed scaling factor (specified below) and $\Lambda = d^{0.2}$. For simplicity, we set $\mathcal{W} = \{W, a, b\}$.

Besides, $\sigma(\cdot)$ is the truncated degree-3 activation function—it is thus made Lipschitz, which is only needed in the proof to *deal* with the case where the generator is trained much faster than the discriminator. Note that this latter case is uncommon in practice.

Lastly, we choose a cubic activation as it is the smallest polynomial degree for the discriminator that is sufficient: as pointed out in Li & Dou (2020), with linear or quadratic activations, the generator can fool the discriminator by only matching the first and second moments of $\mathcal{D}$. By doing so, the generator *cannot* recover the modes u_1, u_2 but only weighted averages of them even at the global optimal solution.

Algorithms. We solve the training problem (6) using SGDA and nSGDA *with momentum* $= 0$. For simplicity, we also set the batch-size to 1. At each step t, we sample $X \sim \mathcal{D}$ and $z \sim \mathcal{D}_z$ and define SGDA's update as

$$\mathcal{W}^{(t+1)} = \mathcal{W}^{(t)} + \eta_D \mathbf{g}_{\mathcal{W}}^{(t)}, \qquad \mathcal{V}^{(t+1)} = \mathcal{V}^{(t)} - \eta_G \mathbf{g}_{\mathcal{V}}^{(t)}, \tag{SGDA}$$

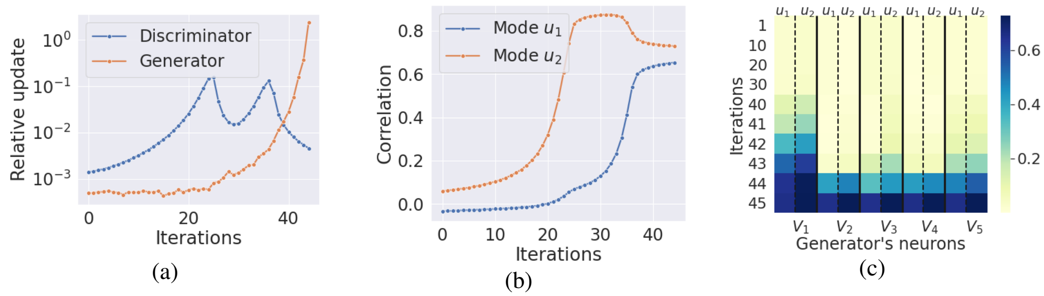


Figure 5: Overview of the learning process of SGDA with $m_D = m_G = 5, d = 1000$.

where $\eta_D, \eta_G > 0$ are constant step-sizes. On the other hand, nSGDA is defined by the update rule

$$\mathcal{W}^{(t+1)} = \mathcal{W}^{(t)} + \eta_D \frac{\mathbf{g}_{\mathcal{W}}^{(t)}}{\|\mathbf{g}_{\mathcal{W}}^{(t)}\|_2}, \qquad \mathcal{V}^{(t+1)} = \mathcal{V}^{(t)} - \eta_G \frac{\mathbf{g}_{\mathcal{V}}^{(t)}}{\|\mathbf{g}_{\mathcal{V}}^{(t)}\|_2}. \tag{nSGDA}$$

Compared to (4), (nSGDA) is a global nSGDA update (without momentum) (i.e. we do not consider *layer-wise* normalization). Indeed, $\|\mathbf{g}_{\mathcal{Y}}^{(t)}\|_2$ in the update refers to the sum of norms of the gradients with respect to a, b, W. We now detail how to set parameters in (SGDA) and (nSGDA).

Parametrization 4.1. *When running SGDA and nSGDA on (1), we set the parameters as*

*– **Initialization**:* $b^{(0)} = 0,\ a^{(0)} \sim \mathcal{N}\left(0, \frac{1}{m_D \mathrm{polylog}(d)}\right),\ w_i^{(0)} \sim \mathcal{N}\left(0, \frac{1}{d}\mathbf{I}\right),\ v_j^{(0)} \sim \mathcal{N}\left(0, \frac{1}{d^2}\mathbf{I}\right)$ *for* $i \in [m_D]$, $j \in [m_G]$.

*– **Number of iterations**: we run SGDA for* $t \leq T_0$ *iterations where* T_0 *is the first iteration such that* $\|\nabla \mathbb{E}[L_{\mathcal{V}^{(T_0)}, \mathcal{W}^{(T_0)}}(X, z)]\|_2 \leq 1/\mathrm{poly}(d)$. *. For nSGDA, we run for* $T_1 = \tilde{\Theta}\left(\frac{1}{\eta_D}\right)$ *iterations.*

*– **Step-sizes**: For SGDA,* $\eta_D, \eta_G \in (0, \frac{1}{\mathrm{poly}(d)})$. *For nSGDA,* $\eta_D \in (0, \frac{1}{\mathrm{poly}(d)}]$, $\eta_G = \frac{\eta_D}{\mathrm{polylog}(d)}$.

*– **Over-parametrization**: For SGDA,* $m_D, m_G = \mathrm{polylog}(d)$ *are arbitrarily chosen i.e.* m_D *may be larger than* m_G *or the opposite. For nSGDA, we set* $m_D = \log(d)$ *and* $m_G = \log(d) \log\log d$.

Parametrization 4.1 corresponds to usual initialization and optimization hyper-parameters. Regarding initialization, the discriminator's weights are sampled from a standard normal and its bias is set to zero. The weights of the generator are initialized from a normal with variance $1/d^2$ (instead of the $1/d$ in standard normal). Such a choice is explained as follows. In practice, the target $X \sim \mathcal{D}$ is a 1D image, thus has entries in $[0, 1]^d$ and norm $O(\sqrt{d})$. Yet, we sample the initial generator's weights from $\mathcal{N}(0, \mathbf{I}_d/d)$ in this case. In our case, since $\|u_i\|_2 = 1$, the target $X = s_1 u_1 + s_2 u_2$ has norm $O(1)$. Therefore, we scale down the variance in the normal distribution by a factor of $1/d$ to match the configuration encountered in practice. Therefore, we also set $\lambda = \frac{1}{\sqrt{d}\mathrm{polylog}(d)}$ in (9) to ensure that the weights and the bias in the discriminator learn at the same speed.

Regarding the number of iterations, our theorem holds when running SGDA for any (polynomially) possible number of iterations –after T_0 steps, the gradient becomes inverse polynomially small and SGDA essentially stops updating the parameters. Lastly, we allow any step-size configuration for SGDA i.e. larger, smaller, or equal step-size for D compared to G. To our knowledge, this setting has never been studied in GAN performance. Note that our choice of step-sizes for nSGDA matches with the one used in practice i.e. η_D slightly larger than η_G.

4.2 Main results

We state our main results on the performance of models trained using (SGDA) and (nSGDA). We show that nSGDA is able to learn the two modes of the distribution $\mathcal{D}$ while SGDA is not.

Theorem 4.1 (SGDA suffers from mode collapse)**.** *Let* T_0, η_G, η_D *and the initialization as defined in Parametrization 4.1. Let t be such that* $t \leq T_0$. *Run SGDA for t iterations with step-sizes* η_G, η_D. *Then, with probability at least* $1 - o(1)$, *for all* $z \in \{0, 1\}^{m_G}$, *we have:*

$$G_{\mathcal{V}}^{(t)}(z) = \alpha^{(t)}(z)(u_1 + u_2) + \xi^{(t)}(z), \quad \textit{where} \ \ \alpha^{(t)}(z) \in \mathbb{R} \ \ \textit{and} \ \ \xi^{(t)}(z) \in \mathbb{R}^d,$$

such that for all $\ell \in [2]$, $|\langle \xi^{(t)}(z), u_\ell \rangle| = o(1)\|\xi^{(t)}(z)\|_2$ *for every* $z \in \{0, 1\}^{m_G}$.

In the specific case where $\eta_G = \frac{\sqrt{d}\eta_D}{\mathrm{polylog}(d)}$, *the model mode collapses i.e.* $\|\xi^{(T_0)}(z)\|_2 = o(\alpha^{(T_0)}(z))$.

Theorem 4.1 indicates that when using SGDA with any step-size configuration, the generator either does not learn the modes at all – when $\alpha^{(t)}(z) = 0,\ G_{\mathcal{V}}^{(t)}(z) = \xi^{(t)}(z)$ – or learns an average of the modes – when $\alpha^{(t)}(z) \neq 0,\ G_{\mathcal{V}}^{(t)}(z) \approx \alpha^{(t)}(z)(u_1 + u_2)$. We emphasize that the theorem holds *for any* time $t \leq T_0$ which is the iteration where SGDA converges to an approximate first-order locally optimal min-max equilibrium. Conversely, nSGDA succeeds to learn the two modes separately.

Theorem 4.2 (nSGDA recovers modes separately)**.** *Let T_1, η_G, η_D and the initialization as defined in Parametrization 4.1. Run nSGDA for T_1 iterations with step-sizes η_G, η_D. Then, the generator learns both modes u_1, u_2 i.e.,*

$$\Pr{}_{z\sim\mathcal{D}_z}\left(\Big\|\frac{G_{\mathcal{V}}^{(T_1)}(z)}{\|G_{\mathcal{V}}^{(T_1)}(z)\|_2} - u_\ell\Big\|_2 = o(1)\right) = \tilde{\Omega}(1)\,, \quad \textit{for} \quad \ell = 1, 2\,. \tag{10}$$

Theorem 4.2 indicates that when we train a GAN with nSGDA in the regime where the discriminator updates slightly faster than the generator (as done in practice), the generator successfully learns the distribution containing the direction of both modes.

4.3 Why does SGDA suffer from mode collapse?

We now sketch the reason why SGDA suffers from mode collapse using Figure 5. Figure 5(a) displays the relative update speed $\frac{\eta\|\mathbf{g}_{\mathcal{Y}}^{(t)}\|_2}{\|\mathcal{Y}^{(t)}\|_2}$, Figure 5(b) the correlation $\frac{\langle w_i^{(t)}, u_\ell\rangle}{\|w_i^{(t)}\|_2}$ between D's neuron and mode u_ℓ and Figure 5(c) the correlation $\frac{\langle v_j^{(t)}, u_\ell\rangle}{\|v_j^{(t)}\|_2}$ between G's neuron and mode u_ℓ.

We focus on the case where the discriminator's step-size is set so it updates slightly faster than the generator at the beginning – as displayed in iterations 1–5 in Figure 5(a). The key observation is that in the early stages, the discriminator's update is approximately

$$w_i^{(t+1)} \approx w_i^{(t)} + \eta_D\langle w_i^{(t)}, X\rangle^2 X, \qquad \text{for } i \in [m_D]. \tag{11}$$

With high probability, there exists at least a neuron i such that $\langle w_i^{(t)}, X\rangle > 0$. Thus, (11) implies that $\langle w_i^{(t)}, X\rangle$ is an increasing sequence. As t increases, $w_i^{(t)}$ gradually grows its correlation with one of the modes u_ℓ (iterations 1-20 in Figure 5(b)) and D's gradient norm thus increases. Therefore, after a first phase where D updates slightly faster than G, D's update speed becomes significantly larger than G's one (iterations 5–20 in Figure 5(a)). Thus, D learns the first mode after 20 iterations in Figure 5(b).

However, one of the goal of D (from the training objective in (6)) is to maximize its *average* correlation with the target distribution $\mathcal{D}$. Since G does not catch up and $\mathcal{D}$ is made of two modes showing up with equal probability, the optimal solution for D is to have each of its $w_i = \alpha_i(u_1+u_2)$ to maximize the average correlation. Therefore, after iteration 20, D learns the second mode and eventually gets this optimal solution (which is a mode collapse) at iteration 40. Then, D's update speed starts dropping as we see in Figure 5(a) which helps G to catch up and grow its update speed. However, since D already learnt a weighted average of the modes, it can only teach G to learn this average and thus mode collapses as we see in Figure 5(c).

On the other hand, nSGDA ensures that G and D *always learn at the same speed*, so that G can learn one mode immediately when D learns (such as at iteration 25 in (b) in Figure (5)), which avoids mode collapse.

5 Conclusion

Our work offers a complementary view to several works in the min-max optimization literature where the discriminator is much faster than the generator to converge to an equilibrium. Here, instead, we advocate the use of balanced updates to ensure that the GAN performs well.

Our work is a first step towards understanding how adaptive methods improve the GAN performance. Our empirical observations and theorems heavily rely on the fact that the batch size is small. However, nSGDA methods seem to not work well for large batch sizes and may not be suitable to some large-scale GANs such as BigGAN (Brock et al., 2018). It would be interesting to understand why adaptive methods are crucial in this case. Another interesting direction would be to improve the training of nSGDA in the batch setting.

REFERENCES

Naman Agarwal, Rohan Anil, Elad Hazan, Tomer Koren, and Cyril Zhang. Disentangling adaptive gradient methods from learning rates. *arXiv preprint arXiv:2002.11803*, 2020.

Zeyuan Allen-Zhu and Yuanzhi Li. Towards understanding ensemble, knowledge distillation and self-distillation in deep learning. *CoRR*, abs/2012.09816, 2020. URL https://arxiv.org/abs/2012.09816.

Zeyuan Allen-Zhu and Yuanzhi Li. Forward super-resolution: How can gans learn hierarchical generative models for real-world distributions. *arXiv preprint arXiv:2106.02619*, 2021.

Kimon Antonakopoulos, Veronica Belmega, and Panayotis Mertikopoulos. An adaptive mirror-prox method for variational inequalities with singular operators. *Advances in Neural Information Processing Systems*, 32:8455–8465, 2019.

Kimon Antonakopoulos, E Veronica Belmega, and Panayotis Mertikopoulos. Adaptive extra-gradient methods for min-max optimization and games. *arXiv preprint arXiv:2010.12100*, 2020.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pp. 214–223. PMLR, 2017.

Sanjeev Arora, Rong Ge, Yingyu Liang, Tengyu Ma, and Yi Zhang. Generalization and equilibrium in generative adversarial nets (gans). In *International Conference on Machine Learning*, pp. 224–232. PMLR, 2017.

Sanjeev Arora, Andrej Risteski, and Yi Zhang. Do gans learn the distribution? some theory and empirics. In *International Conference on Learning Representations*, 2018.

Francis Bach and Kfir Y Levy. A universal algorithm for variational inequalities adaptive to smoothness and noise. In *Conference on Learning Theory*, pp. 164–194. PMLR, 2019.

Yu Bai, Tengyu Ma, and Andrej Risteski. Approximability of discriminators implies diversity in gans. *arXiv preprint arXiv:1806.10586*, 2018.

Yogesh Balaji, Mohammadmahdi Sajedi, Neha Mukund Kalibhat, Mucong Ding, Dominik Stöger, Mahdi Soltanolkotabi, and Soheil Feizi. Understanding overparameterization in generative adversarial networks. *arXiv preprint arXiv:2104.05605*, 2021.

Babak Barazandeh, Davoud Ataee Tarzanagh, and George Michailidis. Solving a class of non-convex min-max games using adaptive momentum methods. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3625–3629. IEEE, 2021.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

Dami Choi, Christopher J Shallue, Zachary Nado, Jaehoon Lee, Chris J Maddison, and George E Dahl. On empirical comparisons of optimizers for deep learning. *arXiv preprint arXiv:1910.05446*, 2019.

Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 215–223. JMLR Workshop and Conference Proceedings, 2011.

Jorge Cortés. Finite-time convergent gradient flows with applications to network consensus. *Automatica*, 42(11):1993–2000, 2006.

Ashok Cutkosky and Harsh Mehta. Momentum improves normalized sgd. In *International Conference on Machine Learning*, pp. 2260–2268. PMLR, 2020.

Constantinos Daskalakis, Andrew Ilyas, Vasilis Syrgkanis, and Haoyang Zeng. Training gans with optimism. *arXiv preprint arXiv:1711.00141*, 2017.

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

Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, and Aaron Courville. Adversarially learned inference. *arXiv preprint arXiv:1606.00704*, 2016.

Soheil Feizi, Farzan Farnia, Tony Ginart, and David Tse. Understanding gans: the lqg setting. *arXiv preprint arXiv:1710.10793*, 2017.

Tanner Fiez and Lillian Ratliff. Gradient descent-ascent provably converges to strict local minmax equilibria with a finite timescale separation. *arXiv preprint arXiv:2009.14820*, 2020.

Tanner Fiez, Benjamin Chasnov, and Lillian Ratliff. Implicit learning dynamics in stackelberg games: Equilibria characterization, convergence analysis, and empirical study. In *International Conference on Machine Learning*, pp. 3133–3144. PMLR, 2020.

AV Gasnikov, PE Dvurechensky, FS Stonyakin, and AA Titov. An adaptive proximal method for variational inequalities. *Computational Mathematics and Mathematical Physics*, 59(5):836–841, 2019.

Gauthier Gidel, Hugo Berard, Gaëtan Vignoud, Pascal Vincent, and Simon Lacoste-Julien. A variational inequality perspective on generative adversarial networks. *arXiv preprint arXiv:1802.10551*, 2018.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans. *arXiv preprint arXiv:1704.00028*, 2017.

Elad Hazan, Kfir Y Levy, and Shai Shalev-Shwartz. Beyond convexity: Stochastic quasi-convex optimization. *arXiv preprint arXiv:1507.02030*, 2015.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pp. 630–645. Springer, 2016.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Chi Jin, Praneeth Netrapalli, and Michael Jordan. What is local optimality in nonconvex-nonconcave minimax optimization? In *International Conference on Machine Learning*, pp. 4880–4889. PMLR, 2020.

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

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Qi Lei, Jason Lee, Alex Dimakis, and Constantinos Daskalakis. Sgd learns one-layer networks in wgans. In *International Conference on Machine Learning*, pp. 5799–5808. PMLR, 2020.

Kfir Y Levy. The power of normalization: Faster evasion of saddle points. *arXiv preprint arXiv:1611.04831*, 2016.

Yuanzhi Li and Zehao Dou. Making method of moments great again?–how can gans learn distributions. *arXiv preprint arXiv:2003.04033*, 2020.

Tengyuan Liang. How well can generative adversarial networks learn densities: A nonparametric view. *arXiv preprint arXiv:1712.08244*, 2017.

Mingrui Liu, Youssef Mroueh, Jerret Ross, Wei Zhang, Xiaodong Cui, Payel Das, and Tianbao Yang. Towards better understanding of adaptive gradient algorithms in generative adversarial nets. *arXiv preprint arXiv:1912.11940*, 2019.

Yang Liu, Jeremy Bernstein, Markus Meister, and Yisong Yue. Learning by turning: Neural architecture aware optimisation. *arXiv preprint arXiv:2102.07227*, 2021.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *International conference on machine learning*, pp. 3481–3490. PMLR, 2018.

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

Ryan Murray, Brian Swenson, and Soummya Kar. Revisiting normalized gradient descent: Fast evasion of saddle points. *IEEE Transactions on Automatic Control*, 64(11):4818–4824, 2019.

Y.E. Nesterov. Minimization methods for nonsmooth convex and quasiconvex functions. *Econ. Mat. Met.*, 20:519–531, 01 1984.

Antonio Orvieto, Jonas Kohler, Dario Pavllo, Thomas Hofmann, and Aurelien Lucchi. Vanishing curvature and the power of adaptive methods in randomly initialized deep networks, 2021.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

Tijmen Tieleman, Geoffrey Hinton, et al. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Yang You, Igor Gitman, and Boris Ginsburg. Large batch training of convolutional networks. *arXiv preprint arXiv:1708.03888*, 2017.

Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. Large batch optimization for deep learning: Training bert in 76 minutes. *arXiv preprint arXiv:1904.00962*, 2019.

Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop, 2016.

Jure Zbontar, Florian Knoll, Anuroop Sriram, Tullie Murrell, Zhengnan Huang, Matthew J Muckley, Aaron Defazio, Ruben Stern, Patricia Johnson, Mary Bruno, et al. fastmri: An open dataset and benchmarks for accelerated mri. *arXiv preprint arXiv:1811.08839*, 2018.

Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *International conference on machine learning*, pp. 7354–7363. PMLR, 2019a.

Jingzhao Zhang, Sai Praneeth Karimireddy, Andreas Veit, Seungyeon Kim, Sashank J Reddi, Sanjiv Kumar, and Suvrit Sra. Why are adaptive methods good for attention models? *arXiv preprint arXiv:1912.03194*, 2019b.

Pengchuan Zhang, Qiang Liu, Dengyong Zhou, Tao Xu, and Xiaodong He. On the discrimination-generalization tradeoff in gans. *arXiv preprint arXiv:1711.02771*, 2017.

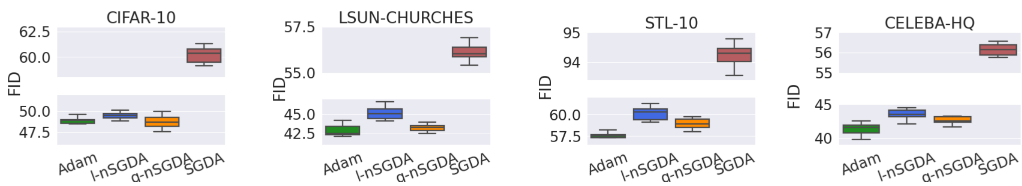
Dongruo Zhou, Jinghui Chen, Yuan Cao, Yiqi Tang, Ziyan Yang, and Quanquan Gu. On the convergence of adaptive gradient methods for nonconvex optimization. *arXiv preprint arXiv:1808.05671*, 2018.

A ADDITIONAL EXPERIMENTS

A.1 EXPERIMENTS WITH DCGAN

This section shows that experimental results obtained in Section 3 are also valid for other architectures such as DCGAN. Indeed, we observe that nSGDA methods compete with Adam and nSGDA work when the batch size is small.

A.1.1 NSGDA COMPETES WITH ADAM



(a) CIFAR-10 (b) LSUN Churches (c) STL-10 (d) Celeba-HQ

Figure 6: FID scores obtained when training a DCGAN using Adam, lnSGDA, gnSGDA and SGD on different datasets. In all these datasets, lnSGDA, gnSGDA and Adam perform approximately as well. As expected, SGDA performs much worse than the other optimizers. The models are trained with batch-size 64 –which is the usual batch-size used for DCGAN.

A.1.2 INFLUENCE OF THE BATCH SIZE

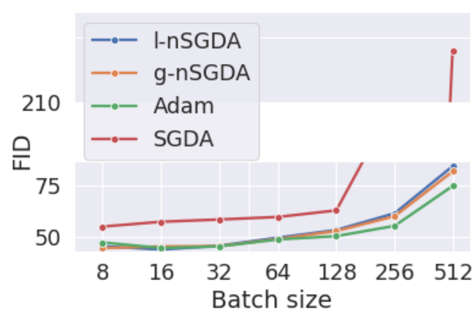
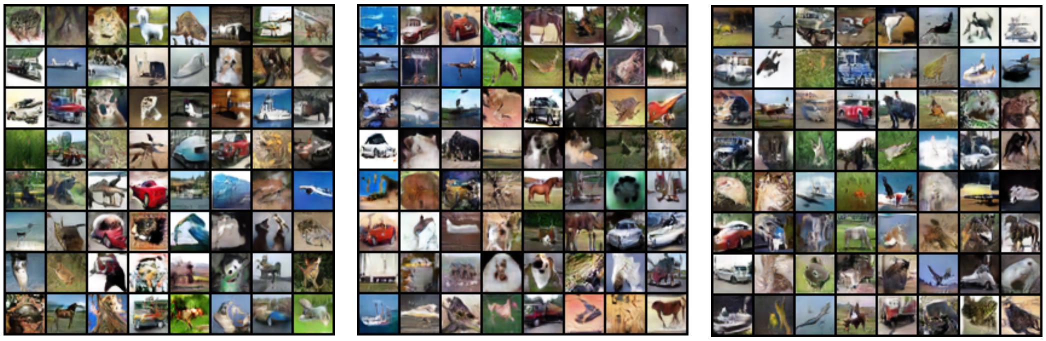


Figure 7: FID score of a DCGAN model trained with lnSGDA, gnSGDA, Adam and SGDA against the batch size in the CIFAR-10 dataset. The same observations as the plot for WGAN-GP holds. We however remark that the gap with SGDA is much smaller for DCGAN.

A.2 IMAGES GENERATED WITH WGAN-GP

In this section, we display the images obtained when training the Resnet WGAN-GP model from Section 3.

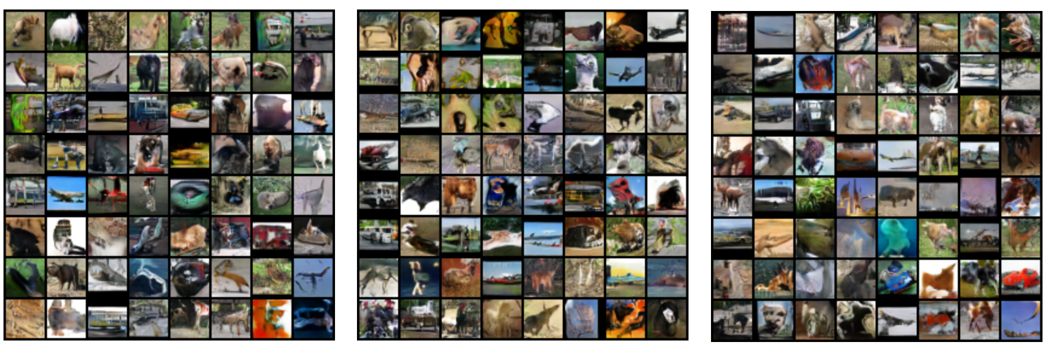


(a) Adam (b) lnSGDA (c) gnSGDA

Figure 8: CIFAR-10 images generated by a Resnet WGAN-GP model

(a) Adam (b) lnSGDA (c) gnSGDA

Figure 9: LSUN-Churches images generated by a Resnet WGAN-GP model



(a) Adam (b) lnSGDA (c) gnSGDA

Figure 10: STL-10 images generated by a Resnet WGAN-GP model

(a) Adam (b) lnSGDA (c) gnSGDA

Figure 11: Celeba-HQ images generated by a Resnet WGAN-GP model

B Notations

Let us also write $\tau_b = \lambda$ as the scaling factor of the bias. We can easily observe that at every step, all of $w_i^{(t)}$ and $v_i^{(t)}$ lies in the span of $\{w_j^{(0)}, v_j^{(0)}, u_1, u_2\}$. Therefore, let us denote

$$w_i^{(t)} = \sum_{j\in[m_D]} \alpha(w_i, w_j, t)\frac{w_j^{(0)}}{\|w_j^{(0)}\|_2} + \sum_{j\in[m_G]} \alpha(w_i, v_j, t)\frac{v_j^{(0)}}{\|v_j^{(0)}\|_2} + \sum_{j\in[2]} \alpha(w_i, u_j, t)\frac{u_j}{\|u_j\|}$$

and $v_i^{(t)}$ as $\alpha(v_i, *, t)$, where $\alpha(*, *, *) \in \mathbb{R}$.

Let us denote

$$f(X) = a\left(\sum_{i\in[m_D]} \sigma(\langle w_i, X\rangle)\right) + \tau_b b$$

as the function in discriminator without going through sigmoid, and define $h(X) = \sum_{i\in[m_D]} \sigma(\langle w_i, X\rangle)$.

Gradient The gradient of $L(X, z)$ is given as:

$$\nabla_a L(X, z) = -\operatorname{Sigmoid}(-f(X))h(X) + \operatorname{Sigmoid}(f(G(z)))h(G(z))$$
$$\nabla_b L(X, z) = -\operatorname{Sigmoid}(-f(X)) + \operatorname{Sigmoid}(f(G(z)))$$
$$\nabla_{w_i} L(X, z) = -\operatorname{Sigmoid}(-f(X))a\sigma'(\langle w_i, X\rangle)X + \operatorname{Sigmoid}(f(G(z)))a\sigma'(\langle w_i, G(z)\rangle)G(z)$$
$$\nabla_{v_i} L(X, z) = -1_{z_i=1}\operatorname{Sigmoid}(f(G(z)))a\sum_{j\in[m_D]} \sigma'(\langle w_i, G(z)\rangle)w_i$$

We use $a^{(t)}, b^{(t)}, w_i^{(t)}, v_i^{(t)}$ to denote the value of those weights at iteration t.

We use $a = b \pm c$ for $c \in \mathbb{R}^*$ to denote: (1). $a \in [b - c, b + c]$ if $a, b \in \mathbb{R}$, (2). $\|a - b\|_2 \le c$ if a, b are vectors.

For simplicity, we focus on the case when all $\Pr[z_i = 1]$ are equal. The other cases can be proved similarly (by replacing the $1/m_G$ factor in the generators update by the exact value of $\Pr[z_i = 1]$).

C Initialization conditions and three regime of learning

We first show the following Lemma regarding initialization:

Let

$$A_{i,\ell} = \frac{1}{2}\sigma'(\langle w_i^{(0)}, u_\ell\rangle)\operatorname{sign}(\langle w_i^{(0)}, u_\ell\rangle)$$

and

$$B_{i,j} = \frac{1}{m_G}\sigma'(\langle w_i^{(0)}, v_j^{(0)}\rangle)\,\mathrm{sign}(\langle w_i^{(0)}, v_j^{(0)}\rangle)$$

and

$$C_{i,\ell} = \sigma'(\langle v_i^{(0)}, u_\ell\rangle)$$

Let $A = \max_{i\in[m_D],\ell\in[2]} A_{i,\ell}$, $B = \max_{i\in[m_D],j\in[m_G]} B_{i,j}$, $C = \max_{i\in[m_G],\ell\in[2]} C_{i,\ell}$, we have: Using a corollary of Proposition $G.1$ in Allen-Zhu & Li (2020):

Lemma C.1. *For every $\eta_D, \eta_G > 0$, we have that: with probability at least $1 - o(1)$, we have that: $A = \frac{\mathrm{polyloglog}(d)}{\sqrt{d}}$, $B = \frac{\mathrm{polyloglog}(d)}{dm_G}$. Moreover, with probability at least $1 - o(1)$, one and only one of the following holds:*

1. *(Discriminator trains too fast): $\eta_G B < \frac{1}{\mathrm{polylog}(d)}\eta_D A$;*
2. *(Balanced discriminator and generator): $\eta_G B > \frac{1}{\mathrm{polylog}(d)}\eta_D A$, $\eta_D A > \eta_G B(1 + \frac{1}{\mathrm{polyloglog}(d)})$;*
3. *(Generator trains too fast): $\eta_D A < \eta_G B(1 - \frac{1}{\mathrm{polyloglog}(d)})$.*

This Lemma implies that in case 2, $\eta_G = \tilde{\Theta}(\sqrt{d})\eta_B$.

We will show the following induction hypothesis for each case. Intuitively, in case one we have the following learning process: (too powerful D).

1. At first D starts to learn, then because of the learning rate of G is too small, so D just saturate the loss to make the gradient to zero.

In case two we have: ("balanced" D and G but still not enough).

1. At first D starts to learn one u_j in each of the neuron.
2. However, the generator still could not catch up immediate after D learns one u_j, so D starts to a mixture of u_1, u_2 in its neurons since u_1, u_2 are positively correlated.
3. After that G starts to learn, however since D already stuck at the mixtures of u_1, u_2, so G is only able to learn mixtures of u_1, u_2 as well.

In case three we have: (Too powerful G)

1. G starts to learn without D learning any meaningful signal yet, so G aligns its outputs with the (close to random) weights of D and just pushes the discriminator to zero. In this case, G simply learns something random to fool D instead of learning the signals.

Moreover, similar to Lemma (C.1), we also have the following condition regarding the gap between the top one and the second largest one in terms of correlation:

Lemma C.2. *Let*

$$i_D, \ell_D = \arg\max_{i\in[m_D],\ell\in[2]} A_{i,\ell}$$

Let

$$i_G, j_G = \arg\max_{i\in[m_D],j\in[m_G]} B_{i,j}$$

Then with probability at least $1 - o(1)$ over the random initialization, the following holds:

$$\forall i, \ell \neq i_D, \ell_D : A_{i_D,\ell_D} \geq A_{i,\ell}\left(1 + \frac{1}{\mathrm{polyloglog}(d)}\right)$$

$$\forall i, j \neq i_G, j_G : B_{i_G,j_G} \geq B_{i,j}\left(1 + \frac{1}{\mathrm{polyloglog}(d)}\right)$$

and

$$A = \frac{\Theta(\sqrt{\log\log(d)})}{\sqrt{d}}, \quad B, C = \frac{\Theta(\sqrt{\log\log(d)})}{d}$$

For simplicity, we also define $i^* = i_D$.

D CRITICAL LEMMA

The proof heavily relies on the following Lemma about tensor power method, which is a corollary of Lemma C.19 in Allen-Zhu & Li (2020).

Lemma D.1. *For every $\delta \in (0, 0.1)$, every $C > 10$, for every sequence of $x_t, y_t > 0$ such that $x_0 > (1 + 10\delta)y_0$, suppose there is a sequence of $S_t \in [0, C]$ such that for $\eta \in \left(0, \frac{1}{\mathrm{poly}(C/\delta)}\right)$:*

$$x_{t+1} \geq x_t + \eta S_t x_t^2$$

$$y_t \leq y_t + \eta S_t (1 + \delta) y_t^2$$

For every $\tau > 0$, let T_0 be the first iteration where $x_t > \tau$, then we must have:

$$y_{T_0} \leq \frac{y_0}{\mathrm{poly}(\delta)}$$

Moreover, if all $S_t \geq H$ for some $H > 0$, then $T_0 \leq O\left(\frac{1}{\eta H x_0}\right)$.

Similar to the Lemma above, one can easily show the following auxiliary Lemma:

Lemma D.2. *Suppose there are sequences $a_t, b_t \in \mathbb{R}^d$ such that $a_0, b_0 > 0$ with $a_0 < 0.82b_0$. Suppose there exists a sequence of $C_t \in (0, d)$ such that*

$$a_{t+1} \leq a_t - \eta_D C_t b_t$$

$$b_{t+1} \geq b_t - 1.0000001\eta_D C_t a_t$$

Then we must have that for every $t \leq T$ where T is the first iteration such that $a_T \leq 0$, then the following holds:

$$a_t = a_0 - \Theta\left(\eta \sum_{s \leq t-1} C_s\right)$$

$$\sum_{s \leq t} |a_t C_t \eta_D| \leq 0.49b_0$$

Moreover, if in addition that $a_0 < \frac{1}{C} b_0$ for any $C > 100$, then we must have:

$$\sum_{s \leq t} |a_t C_t \eta_D| \leq \frac{10}{C} b_0$$

In the end, we have the following comparison Lemma, whose proof is obvious:

Lemma D.3. *Suppose $a_t, b_t > 0$ satisfies that $a_0, b_0 \leq 1$, and the update of a_t, b_t is given as: For some values $C > 0$ and $C_t \in [0, \mathrm{poly}(d)]$:*

$$a_{t+1} = a_t + \eta_D C_t \tag{12}$$

$$b_{t+1} = b_t + \eta_D \left[\frac{1}{C}, 1\right] \times C_t \tag{13}$$

Then let T be the first iteration where $a_T \geq 2C$, we must have:

$$b_T \in [1, 2C + 1]$$

Using this Lemma, we can directly prove the following Lemma:

Lemma D.4. *For every $\eta_D, \eta_G \in \left(0, \frac{1}{\mathrm{poly}(d)}\right]$ such that $\eta_G = \eta_D \Gamma$ for $\Gamma = \tilde{\Theta}(\sqrt{d})$, suppose there are vectors $p_t, q_{i,t} \in \mathbb{R}^d$ ($i \in [m_G]$) and a value $a_t \in \mathbb{R}, H > 0$ satisfies that for a sequence of $H_{i,t} \in [H, 1]$ for $i \in [m_G]$, $G_t = \tilde{\Theta}(\sum_{i \in [m_G]} H_{i,t})$, a value $\tau = \tilde{O}(d^{-0.5})$, and a vector $\beta_t \in$ span$\{u_1, u_2\}$ with $\|\beta_t\|_2 = O(1)$: For all $i \in [m_G]$ and $t \geq 0$:*

$$\|q_{i,0}\|_2 = \tilde{\Theta}(d^{-0.49}), \|p_0\|_2 = \log^{\Theta(1)}(d), 0 < a_0 \leq 0.819\|p_0\|_2 \quad \frac{\langle q_{i,0}, p_0\rangle}{\|q_{i,0}\|_2, \|p_{i,0}\|_2} \geq 1 - o(1)$$

$$p_t = p_t - \eta_D \sum_{i \in [m_G]} G_{i,t} a_t \sigma'(\langle p_t, q_{i,t}\rangle) q_{i,t} + \tilde{O}(\eta_D a_t \gamma_t)\beta_t$$

$$a_t = a_t - \eta_D \sum_{i \in [m_G]} G_{i,t} \sigma(\langle p_t, q_{i,t}\rangle) \pm \tilde{O}(\eta_D \gamma_t)$$

$$q_{i,t} = \left(q_{i,t} + \eta_G H_{i,t} a_t \left(\sigma'(\langle p_t, q_{i,t}\rangle) + \sum_{j \in [m_G]} \gamma_{i,j,t} \sigma'(\langle p_t, q_{j,t}\rangle)\right) p_t \pm \eta_G |a_t| \tilde{O}\left(\tau \|q_{i,t}\|_2\right)^2\right)$$

In addition, we have: $\gamma_{i,j,t} = \tilde{O}(1)$, and

$$\max_{i \in [m_G]} \|q_{i,t}\|_2 \in \left(0, \frac{1}{\mathrm{polylog}(d)}\right] \cup [\mathrm{polylog}(d), +\infty) \implies \forall i, j \in [m_G], H_{i,t} = \tilde{\Theta}(G_t), \gamma_{i,j,t} = \tilde{\Theta}(1)$$

Then we must have that: let T be the first iteration where $a_T \leq 0$, we have: for every $t \leq T$: there is a scaling factor $\ell_t = \Theta(1)$ such that

$$\|p_t - \ell_t p_0\|_2 \leq o(1)\|p_0\|_2, \quad \|\Pi_{\mathit{span}\{u_1, u_2, p_0\}^\perp}(p_t - p_0)\|_2 \leq d^{-0.6}\|p_0\|_2$$

Moreover, for every $i, j \in [m_G]$, $\|q_{i,t}\|_2 = \tilde{\Theta}(\|q_{j,t}\|_2)$ and $\|q_{i,T}\|_2 \geq \tilde{\Theta}(\sqrt{\Gamma})$, and as long as $\max_{i \in [m_G]} \|q_{i,t}\|_2 \geq \mathrm{polylog}(d)$, we have that $a_t\|q_{i,t}\|_2 \geq \mathrm{polylog}(d)$.

Moreover,

$$\|\Pi_{\mathit{span}\{u_1, u_2, p_0\}^\perp}(q_{i,t} - q_{i,0})\|_2 \leq d^{-0.6}\|q_{i,t}\|_2$$

proof of Lemma (D.4). For simplicity we consider the case when $H = \tilde{\Omega}(1)$, the other cases follow similarly.

To proof this result, we maintain the following decomposition of p_t and $q_{i,t}$ as:

$$p_t = \alpha(t)p_0 + \beta(t) + \gamma(t)$$

Where $\beta(t) \in$ span$\{u_1, u_2\}$ and $\gamma(t) \perp$span$\{u_1, u_2, p_0\}$. Note that $\alpha(0) = 1, \beta(0) = \gamma(0) = 0$.

$$q_{i,t} = \alpha(i,t)p_0 + \beta(i,t) + \gamma(i,t)$$

Where $\beta(i,t) \in$ spanu_1, u_2 and $\gamma(i,t) \perp$span$\{p_0, u_1, u_2\}$.

We maintain the following induction hypothesis (which we will prove at the end): For some $\mu = 0.00001$ and $C_1 = d^{-0.1}, C_2 = d^{-0.6}$, we have:

1. Through out the iterations, $\alpha(t) \geq 0.5$ and $\|\beta(t)\|_2 \leq 0.5(1-\mu)C_1, \|\gamma(t)\|_2 \leq 0.5(1-\mu)C_2$.

2. $\alpha(i,t) \in (0, \tilde{O}\sqrt{\Gamma})$ and $\|\beta(i,t)\|_2 \leq C_1\alpha(i,t) + \|\beta(i,0)\|_2, \quad \|\gamma(i,t)\|_2 \leq C_2\alpha(i,t) + \|\gamma(i,0)\|_2$

The induction hypothesis implies that through out the iterations, $\langle q_{i,t}, p_t \rangle = \tilde{\Omega}(\|q_{i,t}\|_2)$.

We can now write down the update of $a_t, \alpha's, \beta's$ and $\gamma's$ as:

$$a_{t+1} = a_t - \eta_D \left(\sum_{i \in [m_G]} G_{i,t} \sigma(\langle p_t, q_{i,t} \rangle) \pm \tilde{O}(1) \right) \tag{14}$$

$$\alpha(t+1) = \alpha(t) - \eta_D a_t \sum_{i \in [m_G]} G_{i,t} \sigma'(\langle p_t, q_{i,t} \rangle) \alpha(i,t) \tag{15}$$

$$\beta(t+1) = \beta(t) - \eta_D a_t \sum_{i \in [m_G]} G_{i,t} \sigma'(\langle p_t, q_{i,t} \rangle) \beta(i,t) \pm \tilde{O}(\eta_D |a_t|) \tag{16}$$

$$\gamma(t+1) = \gamma(t) - \eta_D a_t \sum_{i \in [m_G]} G_{i,t} \sigma'(\langle p_t, q_{i,t} \rangle) \gamma(i,t) \tag{17}$$

By the induction hypothesis, we know that

$$\sigma'(\langle p_t, q_{j,t} \rangle) \geq \tilde{\Omega}\left(\|q_{j,t}\|_2^2 \times \frac{\Lambda^2}{\Gamma} \right)$$

Moreover, we have that let $h_{i,t} := \left(\sigma'(\langle p_t, q_{i,t} \rangle) + \sum_{j \in [m_G]} \tilde{\Theta}(\sigma'(\langle p_t, q_{j,t} \rangle)) \right)$

$$\alpha(i,t+1) = \left(\alpha(i,t) + \eta_G H_{i,t} a_t h_{i,t} (1 \pm \tilde{O}(\tau^2 \Lambda^2 / \Gamma) \alpha(t) \right) \tag{18}$$

$$\beta(i,t+1) = \left(\beta(i,t) + \eta_G H_{i,t} a_t h_{i,t} \left(\beta(t) \pm \tilde{O}(\tau^2 \Lambda^2 / \Gamma) \right) \right) \tag{19}$$

$$\gamma(i,t+1) = \left(\alpha(i,t) + \eta_G H_{i,t} a_t h_{i,t} \left(\gamma(t) \pm \tilde{O}(\tau^2 \Lambda^2 / \Gamma) \right) \right) \tag{20}$$

From these formula, we can easily that as long as (1). $\alpha(t) \geq 0.5$ and $\|\beta(t)\|_2 \leq 0.5(1-\mu)C_1, \|\gamma(t)\|_2 \leq 0.5(1-\mu)C_2$, (2). $C_1, C_2 = \tilde{\Omega}(\tau^2\Lambda^2/\Gamma)$, we must have that $\alpha(i,t) > 0$ and $\|\beta(i,t)\|_2 \leq C_1 \alpha(i,t) + \|\beta(i,0)\|_2, \quad \|\gamma(i,t)\|_2 \leq C_2 \alpha(i,t) + \|\gamma(i,0)\|_2$. Therefore, it remains to only prove (1) in the induction hypothesis. Moreover, it is easy to observe that $\alpha(i,t) = \tilde{\Theta}(\alpha(j,t))$ for all $i, j \in [m_G]$ and all t.

Now, we divide the update process into two stages:

Before all $\|q_{i,t}\|_2 = \Omega(\Lambda)$. Let's call these iterations $[T_1]$ Let us consider $T_{i,1}$ such that for all $t \in [T_{i,1}]$ when $q_{i,t} = O(\Lambda)$ and $a_t = \Omega(1)$. In these iterations, by the update rule, we have

$$q_{i,t} = q_{i,t} + \tilde{\Omega}(\eta_G) \sigma'(\langle p_t, q_{i,t} \rangle) p_t \pm \tilde{O}(\eta_G \tau^2 \|q_{i,t}\|_2^2)$$

By the induction hypothesis, we can simplify the update as:

$$\langle q_{i,t}, p_0 \rangle \geq \langle q_{i,t}, p_0 \rangle + \tilde{\Omega}\left(\eta_G \sigma'(\langle q_{i,t}, p_0 \rangle) \right)$$

Therefore, we know that $T_{i,1} \leq \tilde{O}\left(\frac{d^{0.49}}{\eta_G} \right)$ and

$$\sum_{t \leq T_{i,1}} \sigma'(\langle q_{i,t}, p_0 \rangle), \sum_{t \leq T_{i,1}} \sigma'(\langle q_{i,t}, p_t \rangle) \leq \tilde{O}\left(\frac{\Lambda}{\eta_G} \right) \tag{21}$$

Together with the induction hypothesis, the fact that $\alpha(i,t) = \tilde{\Theta}(\alpha(j,t))$, the fact that $\sigma(\langle p_t, q_{i,t}\rangle) = \tilde{\Theta}(\sigma'(\langle p_t, q_{i,t}\rangle)\|q_{i,t}\|_2)$ and update formula Eq equation (14) equation (29) equation (16) equation (17), we know that for all $t \le \max\{T_{i,1}\}$:

$$a_t = a_0 \pm \tilde{O}\left(\frac{\eta_D \Lambda^2}{\eta_G}\right) = a_0 \pm \tilde{O}(d^{-0.01}) \tag{22}$$

$$\alpha(t) = \alpha(0) \pm \tilde{O}\left(\frac{\eta_D \Lambda^2}{\eta_G}\right) = \alpha(0) \pm \tilde{O}(d^{-0.01}) \tag{23}$$

$$\|\beta(t)\|_2 \le \tilde{O}\left(\frac{\eta_D \Lambda^2}{\eta_G}\right) C_1 + \tilde{O}\left(\frac{\eta_D \|\beta(i,0)\|_2 \Lambda}{\eta_G}\right) \le \tilde{O}(d^{-0.01}) C_1 \tag{24}$$

$$\|\gamma(t)\|_2 \le \tilde{O}\left(\frac{\eta_D \Lambda^2}{\eta_G}\right) C_2 + \tilde{O}\left(\frac{\eta_D \|\gamma(i,0)\|_2 \Lambda}{\eta_G}\right) \le \tilde{O}(d^{-0.01}) C_2 \tag{25}$$

When all $\|q_{i,t}\|_2 = \Omega(\Lambda)$: In this case, since $\|p_0\|_2 = \omega(1)$, we know that $\langle p_t, q_{i,t}\rangle = \omega(\Lambda)$, so $\sigma(\langle p_t, q_{i,t}\rangle)$ acts on the linear regime, which means that:

$$\sigma(\langle p_t, q_{i,t}\rangle) = (1 \pm o(1))3\Lambda^2 \langle p_t, q_{i,t}\rangle, \quad \sigma'(\langle p_t, q_{i,t}\rangle) = (1 \pm o(1))3\Lambda^2$$

Therefore, we know that:

$$a_{t+1} \le a_t - (1 - o(1))\eta_D \left(\sum_{i \in [m_G]} G_{i,t} 3\Lambda^2 \|q_{i,t}\|_2\right) \alpha(t)\|p_0\|_2 \tag{26}$$

$$\alpha(t+1)\|p_0\|_2 \ge \alpha(t)\|p_0\|_2 - (1 + o(1))\eta_D \left(\sum_{i \in [m_G]} G_{i,t} 3\Lambda^2 \|q_{i,t}\|_2\right) a_t \tag{27}$$

Now, using the fact that $a_0 \le 0.819\alpha(0)$ and with Eq equation (22) and Eq equation (23), apply Lemma (D.2) we have that until $a_t \le 0$,

$$\sum_t \eta_D \left(\sum_{i \in [m_G]} G_{i,t} 3\Lambda^2 \|q_{i,t}\|_2\right) a_t \le 0.49\|p_0\|_2 \tag{28}$$

Plug in to the update rule:

$$\alpha(t+1) = \alpha(t) \pm (1 + o(1))\eta_D a_t \sum_{i \in [m_G]} G_{i,t} 3\Lambda^2 \alpha(i,t) \tag{29}$$

$$= \alpha(t) \pm (1 + o(1))\eta_D a_t \sum_{i \in [m_G]} G_{i,t} 3\Lambda^2 \frac{\|q_{i,t}\|_2}{\|p_0\|} \tag{30}$$

$$\|\beta(t+1)\|_2 \le \|\beta(t)\|_2 + (1 + o(1))\eta_D a_t \left(\sum_{i \in [m_G]} G_{i,t} 3\Lambda^2 \beta(i,t) + \tilde{O}(1)\right) \tag{31}$$

$$\le \|\beta(t)\|_2 + \eta_D(1 + o(1)) a_t \left(\sum_{i \in [m_G]} G_{i,t} 3\Lambda^2 \frac{\|q_{i,t}\|_2 C_1}{\|p_0\|_2} + \tilde{O}(1)\right) \tag{32}$$

$$\le \|\beta(t)\|_2 + \eta_D(1 + o(1)) a_t \left(\sum_{i \in [m_G]} G_{i,t} 3\Lambda^2 \frac{\|q_{i,t}\|_2 C_1}{\|p_0\|_2}\right) \tag{33}$$

$$\|\gamma(t+1)\|_2 \leq \|\gamma(t)\|_2 + \eta_D(1+o(1))a_t \sum_{i\in[m_G]} G_{i,t}3\Lambda^2\gamma(i,t) \tag{34}$$

$$\leq \|\gamma(t)\|_2 + \eta_D(1+o(1))a_t \sum_{i\in[m_G]} G_{i,t}3\Lambda^2\frac{\|q_{i,t}\|_2 C_2}{\|p_0\|_2} \tag{35}$$

We directly complete the proof of the induction hypothesis using Eq equation (28).

Now it remains to prove that $\|q_{i,T}\|_2 = \Omega(\sqrt{\Gamma})$. Compare the update rule of $q_{i,t}$ and a_t we have:

$$a_{t+1} = a_t - \tilde{\Theta}(\eta_D)G_t\left(\sum_{i\in[m_G]}\Lambda^2\|q_{i,t}\|_2\right) \tag{36}$$

and

$$\sum_{i\in[m_G]}\|q_{i,t+1}\|_2 = \|q_{i,t}\|_2 + \tilde{\Theta}(\eta_G)G_t\Lambda^2 a_t \tag{37}$$

We can directly conclude that $\|q_{i,t}\|_2 \leq \tilde{O}\left(\sqrt{\Gamma}\right)$ and $\|q_{i,T}\|_2 = \tilde{\Omega}(\sqrt{\Gamma})$.

□

Lemma D.5. *For every $\eta_D, \eta_G \in \left(0, \frac{1}{\mathrm{poly}(d)}\right]$ such that $\eta_G = \eta_D\Gamma$ for $\Gamma \geq \tilde{\Omega}(\sqrt{d})$, suppose for sufficiently large $C = \mathrm{poly}(\log(d)m_D)$ there are vectors $\{q_{i,t}\}_{i\in[m_G]}$, $\{p_i\}_{i\in[m_D]}$ in $\mathbb{R}^d$ such that $\|p_i\|_2 = 1, \langle p_i, p_{i'}\rangle \leq \tilde{O}(1/\sqrt{d})$ for i, i', values $H_{i,j,t}, G_{i,t} \in \left[\frac{1}{C^2}, C^2\right]$ and a value $a_0 \geq 0$ satisfies that:*

$$a_0 = \frac{1}{\mathrm{polylog}(d)}, \|q_{j,0}\|_2 = \tilde{\Theta}(\Lambda); \quad q_{j,0} = \sum_{i\in[m_D]} a_i p_i + \xi_j, a_i \geq 0, \|\xi_i\|_2 \leq \frac{1}{C}\|q_{j,0}\|_2 \tag{38}$$

$$a_{t+1} = a_t - \eta_D\left(H_{i,j,t}\sum_{i\in[m_D],j\in[m_G]}\sigma(\langle p_i, q_{j,t}\rangle)\right) \tag{39}$$

$$q_{i,t+1} = q_{i,t} + \eta_G a_t G_{i,t}\sum_{j\in[m_D]}\left(\sigma'(\langle p_j, q_{i,t}\rangle)\left(p_j \pm \frac{1}{C}\right)\right) \tag{40}$$

Then we must have: within $T = \tilde{O}\left(\frac{\sqrt{\Gamma}}{\eta_G}\right)$ many iterations, we must have that $a_t \leq 0$ and $\max_{j\in[m_G]}\|q_{j,T}\|_2 = \tilde{\Theta}(\sqrt{\Gamma})$. Moreover, for every $t \leq T$, we have: for every $j \in [m_G]$,

$$\sum_{i\in[m_D]}\sigma(\langle p_i, q_{j,t}\rangle) = \Omega\left(\max_{i\in[m_D]}\sigma'(\langle p_i, q_{j,t}\rangle)\|q_{j,t}\|_2\right)$$

and

$$\max_{i\in[m_D]}\langle p_i, q_{j,t}\rangle \geq \left(1 - \frac{1}{C^{0.2}}\right)\|q_{j,t}\|_2$$

Proof of Lemma (D.5). Let us denote $r_{i,t} = \max_{j\in[m_D]}\{\langle p_j, q_{i,t}\rangle\}$.

By the update rule, we can easily conclude that:

$$r_{i,t+1} = r_{i,t} + \eta_G G_{i,t}\left(1 - \frac{1}{C^{0.5}}\right)\sigma'(r_{i,t})$$

On the other hand, let us write $q_{i,t} = \sum_{j \in [m_D]} \alpha_{i,j,t} q_j + \xi_{i,t}$, where $\alpha_{i,j,t} \geq 0$. We know that:

$$\|\xi_{i,t+1}\|_2 \leq \|\xi_{i,t}\|_2 + \eta_G G_{i,t} \frac{m_D}{C} \sigma'(r_{i,t}) \tag{41}$$

By the comparison Lemma (D.3) we can easily conclude that for every t,

$$\|\xi_{i,t}\|_2 \leq \frac{1}{C^{0.5}} r_{i,t}$$

This implies that: there exists values $u_t \in [1/C^2, C^2]$ such that

$$a_{t+1} = a_t - \eta_D u_t \sum_{i \in [m_G]} \sigma(r_{i,t}) \tag{42}$$

Comparing this with the update rule of $r_{i,t}$, we know that for every t with $a_t \geq 0$, we must have:

$$r_{i,t} = \tilde{O}\left(\sqrt{\Gamma}\right), \quad r_{i,T} = \tilde{\Theta}(\sqrt{\Gamma})$$

□

Lemma D.6 (Auxiliary Lemma)**.**

For every $g > 0$ we must have: $\mathrm{Sigmoid}(-gx - b)x$ *is a decreasing function of x as long as $gx > 1$ and $gx + b > 0$.*

Lemma D.7. *For $a_t, b_t, c_t, d_t \in \mathbb{R}^d$ be defined as: $a_0, c_0, d_0 = \frac{1}{\mathrm{polylog}(d)}$, $|b_t| \leq O(\log d)$ and $|b_t| \leq \min\{a_t c_t^3, a_t d_t^3\}$.*

$$a_{t+1} = a_t + \eta_D \frac{1}{2} \left(\left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \mathrm{Sigmoid}(-a_t c_t^3 - b_t) c_t^3 + \left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \mathrm{Sigmoid}(-a_t d_t^3 - b_t) d_t^3 \right) \tag{43}$$

$$c_{t+1} = c_t + \eta_D \frac{3}{2} \left(\left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \mathrm{Sigmoid}(-a_t c_t^3 - b_t) c_t^2 a_t \right) \tag{44}$$

$$d_{t+1} = d_t + \eta_D \frac{3}{2} \left(\left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \mathrm{Sigmoid}(-a_t d_t^3 - b_t) d_t^2 a_t \right) \tag{45}$$

Then we have: for every $t \in \left(\frac{\mathrm{polylog}(d)}{\eta_D}, \frac{\mathrm{poly}(d)}{\eta_D}\right]$, we must have:

$$a_t = \sqrt{\frac{2}{3}} \left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) c_t \tag{46}$$

$$c_t = \left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) d_t \tag{47}$$

Proof of Lemma (D.7). By the update formula, we can easily conclude that for $t \leq \frac{\mathrm{poly}(d)}{\eta_D}$, we have that $a_t, c_t, d_t \in \left[\frac{1}{\mathrm{polylog}(d)}, \mathrm{polylog}(d)\right]$. This implies that for every $t \in \left[\frac{\mathrm{polylog}(d)}{\eta_D}, \frac{\mathrm{poly}(d)}{\eta_D}\right]$, we have that

$$a_t c_t^3, a_t d_t^3 \in [1, O(\log d)]$$

Apply Lemma (D.6) we have that: As long as $a_t > 3(c_t + d_t)$, we must have that

$$\mathrm{Sigmoid}(-a_t c_t^3 - b_t) c_t^3 + \mathrm{Sigmoid}(-a_t d_t^3 - b_t) d_t^3 < \mathrm{Sigmoid}(-a_t c_t^3 - b_t) c_t^2 a_t + \mathrm{Sigmoid}(-a_t d_t^3 - b_t) d_t^2 a_t$$

This implies that

$$\frac{a_{t+1}}{3} - \frac{a_t}{3} < c_{t+1} + d_{t+1} - c_t - d_t$$

Note that initially, $a_0, c_0, d_0 = \frac{1}{\mathrm{polylog}(d)}$. This implies that when $t \geq \frac{\mathrm{polylog}(d)}{\eta_D}$, we must have that $a_t \leq 4(c_t + d_t)$, therefore $c_t + d_t = \Omega(1)$. Similarly, we can prove that $a_t \geq 0.1 \min\{c_t, d_t\}$.

as long as $c_t > d_t$, we must have:

$$\text{Sigmoid}(-a_t c_t^3 - b_t) c_t^2 a_t < \text{Sigmoid}(-a_t d_t^3 - b_t) d_t^2 a_t$$

Which implies that:

$$\frac{c_{t+1}}{1 + 1/\operatorname{polylog}(d)} - \frac{c_t}{1 + 1/\operatorname{polylog}(d)} < d_{t+1} - d_t \tag{48}$$

Note that initially, $c_0, d_0 = \frac{1}{\operatorname{polylog}(d)}$ and when $t \geq \frac{\operatorname{polylog}(d)}{\eta_D}$, $c_t + d_t = \Omega(1)$. This implies that for every $t \in \left[\frac{\operatorname{polylog}(d)}{\eta_D}, \frac{\operatorname{poly}(d)}{\eta_D}\right]$, we have: $c_t = \left(1 \pm \frac{1}{\operatorname{polylog}(d)}\right) d_t$. Which also implies that $c_t, d_t \leq O(\log d)$.

Similarly, we can prove the bound for a_t.

□

E Induction hypothesis

E.1 Case 1: Balanced generator and discriminator

In this section we consider the case 2 in Lemma (C.1). Here we give the induction hypothesis to prove the case of balanced generator and discriminator, this is the most difficult case and other cases are just simple modification of this one. Without loss of generality (by symmetry), let us assume that $a^{(0)} > 0$ and $a^{(0)} = \frac{1}{\operatorname{polylog}(d)}$ (this happens with probability $1 - o(1)$).

We divide the training into five stages: For a sufficiently large $C = \operatorname{polylog}(d)$

1. Stage 1: Before one of the $\alpha(w_i, u_j, t) \geq 1/C$. Call this exact iteration $T_{B,1}$.
2. Stage 2: After $T_{B,1}$, before $T_{B,2} = T_{B,1} + \frac{1}{\eta_D 2^{\sqrt{\log(d)}}}$.
3. Stage 3: After $T_{B,2}$, before one of the $\alpha(v_i, u_j, t) \geq d^{-0.49}$. Call this exact iteration $T_{B,3}$.
4. Stage 4: After $T_{B,3}$, before $a^{(t)} \leq \tilde{O}\left(\frac{1}{\Lambda^2 d^{1/4}}\right)$. Call this exact iteration $T_{B,4}$.
5. Stage 5: After $T_{B,4}$, until convergence.

We maintain the following things about α and a, b during each stage:

Stage 1 : We maintain: For every $t \leq T_{B,1}$:

1. (B.1.0). For all but the $i^* \in [m_D]$, and for all $j \in [m_G]$ (Below $*$ can be $w_{i'}, v_{j'}, u_\ell$ for every $i' \in [m_D], j' \in [m_G]$ and $\ell \in [2]$).

$$\forall * \neq u_1, u_2 : |\alpha(w_i, *, t) - \alpha(w_i, *, 0)| \leq \frac{1}{d^{0.9}}, \quad |\alpha(w_i, u_\ell, t) - \alpha(w_i, u_\ell, 0)| \leq \frac{C}{\sqrt{d}}$$

2. (B.1.1). For all $j \in [m_G]$:

$$|\alpha(v_j, *, t) - \alpha(v_j, *, 0)| \leq \frac{\operatorname{polyloglog}(d)}{d}$$

$$|\alpha(v_j, u_\ell, t)| \leq \frac{1}{d}$$

3. (B.1.2). For $i^* = i$, we have that: for all $* \neq u_1, u_2$:

$$|\alpha(w_i, *, t) - \alpha(w_i, *, 0)| \leq \frac{1}{d^{0.9}}$$

4. (B.1.3). a and b remains nice:

$$a^{(t)} \in (1 - 1/C, 1 + 1/C) a_0, |b^{(t)}| \leq \frac{1}{C}$$

Stage 2 : For every $t \in [T_{B,1}, T_{B,2}]$.

1. (B.1.0), (B.1.1), (B.1.2) still holds.
2. (B.2.1): For $i = i^*$, we have:

$$a^{(t)}, \alpha(w_i, u_\ell, t) = \tilde{\Theta}(1)$$

Stage 3 : For every $t \in [T_{B,2}, T_{B.3}]$.

1. (B.1.0), (B.1.2) still holds.
2. (B.3.2): For every $j \in [m_G]$: For $* \neq w_{i^*}, u_1, u_2$, we have:

$$|\alpha(v_j, *, t) - \alpha(v_j, *, 0)| \leq \frac{C^3}{\sqrt{d}} \|v_j^{(t)}\|_2$$

and

$$\alpha(v_i, u_\ell, t) \geq -O\left(\frac{1}{d}\right)$$

Moreover, let $\alpha(t) := \max_{j \in [m_G], \ell \in [2]} \langle v_j^{(t)}, u_\ell \rangle$, we have that:

$$|\alpha(v_j, w_{i^*}, t)| \leq \text{polyloglog}(d)\alpha(t), \quad |\langle v_j^{(t)}, u_\ell \rangle| \leq O(\alpha(t))$$

3. (B.3.3): Balanced update: for every X,

$$\text{Sigmoid}\left(-a^{(t)} \langle w_{i^*}^{(t)}, X \rangle^3 - b^{(t)}\right) \in \left[\frac{1}{\sqrt{d}\,\text{polylog}(d)}, \frac{1}{\text{polylog}(d)}\right] \text{Sigmoid}\left(b^{(t)}\right)$$

and

$$\text{Sigmoid}\left(-a^{(t)} \langle w_{i^*}^{(t)}, u_1 \rangle^3 - b^{(t)}\right) = \left(1 \pm \frac{1}{\text{polylog}(d)}\right) \text{Sigmoid}\left(-a^{(t)} \langle w_{i^*}^{(t)}, u_2 \rangle^3 - b^{(t)}\right)$$

Stage 4 : For every $t \in [T_{B,3}, T_{B.4}]$.

1. (B.3.1), (B.3.2) still holds.
2. (B.4.1) for $i = i^*$, we have that for all $* \neq u_1, u_2, w_i$:

$$|\alpha(w_i, *, t) - \alpha(w_i, *, 0)| \leq \frac{C}{\sqrt{d}}$$

For all $* \in [u_1, u_2, w_i]$:

$$\alpha(w_i, *, t) = \Theta(\alpha(w_i, *, T_{B,3}))$$

3. For every $i, j \in [m_G]$: $\|v_i^{(t)}\|_2 = \tilde{\Theta}(\|v_j^{(t)}\|_2)$ and after $t = T_{B,4}$, we have that for every $i \in [m_G]$, $\|v_i^{(t)}\|_2 = \tilde{\Theta}(d^{1/4})$.
4. $|a^{(t)}|, |b^{(t)}| = O(\log(d))$.

Stage 5 : For every $t \in [T_{B,4}, T_0]$.

1. For every $i \in [m_D]$,

$$|\alpha(w_i, *, T_{B,4}) - \alpha(w_i, *, t)| \leq d^{-0.1}$$

2. For every $i \in [m_G]$,

$$|\alpha(v_i, *, T_{B,4}) - \alpha(v_i, *, t)| \leq d^{0.2}$$

3. $|a^{(t)}| \leq \tilde{O}\left(\frac{1}{\Lambda^2 d^{1/4}}\right)$, and for every z:

$$\langle w_{i^*}^{(t)}, G^{(t)}(z) \rangle = \tilde{\Omega}(d^{1/4})$$

E.2 CASE 2: GENERATOR IS DOMINATING

We now consider another case where the generator's learning rate dominates that of the discriminator. This corresponds to case 3 in Lemma (C.1). In this case, we divide the learning into four stages: For a sufficiently large $C = 2^{\sqrt{\log d}}$:

1. Before $\alpha(v_{j_G}, w_{i_G}, t) \geq d^{-0.49}$. Call this iteration $T_{G,1}$.
2. After $T_{G,1}$, before $\alpha(v_{j_G}, w_{i_G}, t) \geq \Lambda$. Call this iteration $T_{G,2}$.
3. After iteration $T_{G,2}$, before $a_t \leq 0$. Call this iteration $T_{G,3}$.
4. After $T_{G,3}$.

We maintain the following induction hypothesis:

Stage 1 : In this stage, we maintain the following induction hypothesis: Let $\alpha(t) := \alpha(v_{j_G}, w_{i_G}, t)$, for every $t \leq T_{G,1}$:

1. (G.1.1). For all $i \in [m_D]$, and for all $j \in [m_G]$:

$$|\alpha(w_i, *, t) - \alpha(w_i, *, 0)| \leq \frac{C}{\sqrt{d}}$$

2. (G.1.2). For all $j \in [m_G]$, for all $* \neq w_{i_G}$:

$$|\alpha(v_j, *, t) - \alpha(v_j, *, 0)| \leq \frac{C}{\sqrt{d}}\alpha(t)$$

Stage 2 : In this stage, we maintain: for every $t \in [T_{G,1}, T_{G,2}]$

1. (G.2.1). For every $i \in [m_D]$, we have:

$$|\alpha(w_i, *, t) - \alpha(w_i, *, 0)| \leq \frac{1}{C}$$

2. (G.2.2). For every $j \in [m_G]$, $\alpha(v_i, w_{i_G}, t) \geq d^{-0.49}$.
3. For every $i \in [m_G]$, we have: for every $* \neq w_{i_G}$:

$$|\alpha(v_i, *, t) - \alpha(v_i, *, 0)| \leq \frac{2}{C}|\alpha(v_i, w_{i_G}, t)|$$

Stage 3 : In this stage, we maintain: For every $t \in [T_{G,2}, T_{G,3}]$:

1. (G.2.1), (G.2.2) still holds.
2. For every $i \in [m_G]$, we have: for every $* = v_r$ or $* = u_\ell$:

$$|\alpha(v_i, *, t) - \alpha(v_i, *, 0)| \leq \frac{2}{C}\|v_i^{(t)}\|_2$$

Stage 4 : In this stage, we maintain: For every $t \in [T_{G,3}, T_1]$:

1. (G.2.1) still holds.
2. For every $i \in [m_G]$, we have:

$$|\alpha(v_i, *, t) - \alpha(v_i, *, T_{G,3})| \leq \frac{1}{C}\|v_i^{(T_{G,3})}\|_2$$

3. $|\alpha_t| = \tilde{O}\left(\frac{1}{\Lambda^2\sqrt{\eta_G/\eta_D}}\right)$, $\|v_i^{(t)}\|_2 = \tilde{\Theta}(\sqrt{\eta_G/\eta_D})$, and for all $z \neq 0$, $\sum_{i\in[m_D]} h(G^{(t)}(z)) = \tilde{\Theta}(\Lambda^2\sqrt{\eta_G/\eta_D})$.

F PROOF OF THE LEARNING PROCESS IN BALANCED CASE

For simplicity, we are only going to prove the case when $u_1 \perp u_2$ and $\Pr[s_1 = s_2 = 1] = \gamma$. The other case can be proved identically.

F.1 STAGE 1

In this stage, by the induction hypothesis we know that $\|v_i^{(t)}\|_2 \le \tilde{O}(1/\sqrt{d})$. Therefore, the update of $w_i^{(t)}$ can be approximate as:

Lemma F.1. *For every $t \le T_{B,1}$, we know that: when the random samples are (X, z):*

$$w_i^{(t+1)} = w_i^{(t)} + \eta_D a^{(0)} \left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \frac{3}{2} \langle w_i^{(t)}, X\rangle^2 X \pm \eta_D \tilde{O}\left(\frac{1}{d^{1.5}}\right) \tag{49}$$

Moreover, we have that if $z_i = 1$:

$$v_i^{(t+1)} = v_i^{(t)} + \eta_G a^{(0)} \frac{3}{2}\left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \sum_{j\in[m_D]} \left(\langle w_j^{(0)}, G^{(t)}(z)\rangle \pm \frac{1}{C^{0.5} d}\right)^2 w_j^{(t)} \tag{50}$$

$$= v_i^{(t)} + \eta_G a^{(0)} \frac{3}{2}\left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \sum_{j\in[m_D]} \left(\langle w_j^{(0)}, G^{(t)}(z)\rangle \pm \frac{1}{C^{0.5} d}\right)^2 w_j^{(0)} \pm \eta_G O\left(\frac{1}{C^{0.5} d^2}\right) \tag{51}$$

Taking expectation of the above Lemma, we can easily conclude that:

$$\mathbb{E}[w_i^{(t+1)}] = \mathbb{E}[w_i^{(t)}] + \eta_D a^{(0)} \left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \frac{3}{4} \left(\langle w_i^{(t)}, u_1\rangle^2 u_1 + \langle w_i^{(t)}, u_2\rangle^2 u_2 + \Theta(\gamma)\langle w_i^{(t)}, u_1 + u_2\rangle^2 (u_1 + u_2)\right) \tag{52}$$

$$\pm \eta_D \tilde{O}\left(\frac{1}{d^{1.5}}\right) \tag{53}$$

and

$$\mathbb{E}[\langle w_j^{(0)}, G^{(t)}(z)\rangle^2 \mid z_i = 1] = \langle w_j^{(0)}, v_i^{(t)}\rangle^2 \pm O\left(\frac{1}{m_G\, \mathrm{polylog}(d)} \sum_{i\in[m_G]} |\langle w_j^{(0)}, v_i^{(t)}\rangle|\right)^2 \tag{54}$$

Therefore, let $\zeta_t = \max_{i\in[m_G], j\in[m_D]} \langle v_i^{(t)}, w_j^{(0)}\rangle$, $\Upsilon_t = \max_{j\in[m_D], \ell\in[2]} \langle w_j^{(t)}, u_\ell\rangle$, we have that:

$$\mathbb{E}[\Upsilon_{t+1}] = \Upsilon + \eta_D a^{(0)} \frac{3}{4}\left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \Upsilon_t^2 \tag{55}$$

$$\mathbb{E}[\zeta_{t+1}] = \zeta_t + \eta_G a^{(0)} \frac{3}{2m_G}\left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \zeta_t^2 \tag{56}$$

Proof of Lemma (F.1). By the gradient formula, we have:

$$\nabla_{w_i} L(X, z) = -\operatorname{Sigmoid}(-f(X)) a\sigma'(\langle w_i, X\rangle) X + \operatorname{Sigmoid}(f(G(z))) a\sigma'(\langle w_i, G(z)\rangle) G(z)$$

$$\nabla_{v_i} L(X, z) = -1_{z_i=1} \operatorname{Sigmoid}(f(G(z))) a \sum_{j\in[m_D]} \sigma'(\langle w_i, G(z)\rangle) w_i$$

At iteration t, by induction hypothesis, we have that $a^{(t)} = a^{(0)}(1 \pm 1/C)$.

Moreover, by the induction hypothesis again, we have hat $|f(X)| = \tilde{O}(d^{-1.5})$ and $\|G(z)\|_2 \le O(d^{-0.5})$. Together with $\|w_i^{(t)}\|_2 = \tilde{O}(1)$, this implies that

$$\|\operatorname{Sigmoid}(f(G(z))) a\sigma'(\langle w_i, G(z)\rangle) G(z)\|_2 = \tilde{O}(d^{-1.5})$$

This proves the update formula for $w_i^{(t)}$. As for v_i, we observe that by the induction hypothesis and notice that w.h.p. over the randomness of initialization, $|\langle v_i^{(0)}, u_\ell\rangle| \leq \frac{\log d}{d}$, therefore, we can conclude that

$$\langle w_j^{(t)}, G^{(t)}(z)\rangle = \langle w_j^{(0)}, G^{(t)}(z)\rangle \pm \tilde{O}\left(\frac{1}{d^{1.35}}\right) \pm O\left(\frac{\log d}{Cd}\right) = \langle w_j^{(0)}, G^{(t)}(z)\rangle \pm \frac{1}{C^{0.5}d} \tag{57}$$

Note that by induction hypothesis, $\|w_j^{(t)} - w_j^{(0)}\|_2 \leq \frac{1}{C}$ and $\langle w_j^{(0)}, G^{(t)}(z)\rangle \leq \frac{m_G(C^{0.1}+\log d)}{d}$. This implies that

$$\langle w_j^{(t)}, G^{(t)}(z)\rangle^2 w_j^{(t)} = \left(\langle w_j^{(0)}, G^{(t)}(z)\rangle \pm \frac{1}{C^{0.5}d}\right)^2 w_j^{(t)} \tag{58}$$

$$= \left(\langle w_j^{(0)}, G^{(t)}(z)\rangle \pm \frac{1}{C^{0.5}d}\right)^2 w_j^{(0)} + O\left(\frac{1}{C^{0.5}d^2}\right) \tag{59}$$

□

Now, apply Lemma (D.4) and the fact that w.p. $1 - o(1)$, $\zeta_0 = \frac{\text{polyloglog}(d)}{d}$, we have that:

Lemma F.2.

$$\sum_{t \leq T_1} \eta_G \zeta_t^2 \leq O\left(\frac{m_G \,\text{polyloglog}(d)}{a^{(0)}d}\right) \tag{60}$$

In the end, we can show the following Lemma:

Lemma F.3. *When $t = T_{B,1}$, we have that: for both $\ell \in [2]$,*

$$\alpha(w_{i^*}, u_\ell, t) = \frac{1}{\text{polylog}(d)}$$

Proof of Lemma (F.3). By the update formula in Eq equation (52), and the fact that $\Pr[X = u_1 + u_2] \geq \frac{1}{\text{polylog}(d)}$ and the induction hypothesis, we know that for $i = i^*$, for $t \leq T_{B,1}$ we have that:

$$\alpha(w_i, u_\ell, t+1) \geq \alpha(w_i, u_\ell, t) + \tilde{\Omega}(\eta_D) \times \left(\alpha(w_i, u_{3-\ell}, t) - \frac{1}{d}\right)^2$$

This implies that at the end of Stage 1, when $\alpha(w_i, u_{3-\ell}, t) \geq \frac{1}{C}$, we must have $\alpha(w_i, u_\ell, t) \geq \tilde{\Omega}(1)$ as well. □

F.2 Stage 2 and Stage 3

At this stage, by the induction hypothesis, we can approximate the function value as:

$$\sum_{i \in [m_D]} \sigma\left(\langle w_i^{(t)}, X\rangle\right) = \langle w_{i^*}^{(t)}, X\rangle^3 \pm \tilde{O}\left(\frac{1}{d^{1.5}}\right) \tag{61}$$

$$\left|\sum_{i \in [m_D]} \sigma\left(\langle w_i^{(t)}, G^{(t)}(z)\rangle\right)\right| \leq \tilde{O}\left(\|G^{(t)}(z)\|_2\right)^3 \leq \tilde{O}\left(\frac{1}{d^{1.45}}\right) \tag{62}$$

Therefore, at this stage, we can easily approximate the update of $W_D^{(t)}$ as:

Lemma F.4. *When the sample is (X, z), we have: for every $t \in (T_{B,1}, T_{B,3}]$, the following holds:*

$$a^{(t+1)} = a^{(t)} + \eta_D \left(1 \pm \tilde{O}\left(\frac{1}{d}\right)\right) \operatorname{Sigmoid}\left(-a^{(t)} \langle w_{i^*}^{(t)}, X\rangle^3 - b^{(t)}\right) \langle w_{i^*}^{(t)}, X\rangle^3 \tag{63}$$

$$\pm \eta_D \tilde{O}\left(\frac{1}{d^{1.45}}\right) \operatorname{Sigmoid}\left(b^{(t)}\right) \tag{64}$$

$$w_i^{(t+1)} = w_i^{(t)} + 3\eta_D \left(1 \pm \tilde{O}\left(\frac{1}{d}\right)\right) \operatorname{Sigmoid}\left(-a^{(t)} \langle w_i^{(t)}, X\rangle^3 - b^{(t)}\right) a^{(t)} \langle w_i^{(t)}, X\rangle^2 X \tag{65}$$

$$\pm \eta_D \tilde{O}\left(\frac{1}{d^{1.45}}\right) \operatorname{Sigmoid}\left(b^{(t)}\right) \tag{66}$$

$$b^{(t)} = b^{(t)} + \eta_D \tau_b \left(1 \pm \tilde{O}\left(\frac{1}{d}\right)\right) \operatorname{Sigmoid}\left(-a^{(t)} \langle w_{i^*}^{(t)}, X\rangle^3 - b^{(t)}\right) \tag{67}$$

$$- \eta_D \tau_b \left(1 \pm \tilde{O}\left(\frac{1}{d^{1.45}}\right)\right) \operatorname{Sigmoid}\left(b^{(t)}\right) \tag{68}$$

Moreover, the update formula also let us bound $a^{(t)}, \alpha(w_{i^*}, u_1, t)$ as:

Lemma F.5. *Let α_t, a_t be updated as: for $t = T_{B,2}$, $\alpha_t = \alpha(w_{i^*}, u_1, t)$ and $a_t = a^{(t)}$, such that*

$$a_{t+1} = a_t + \eta_G \operatorname{Sigmoid}(-a_t \alpha_t^3 - b_t)\alpha_t^3$$

$$\alpha_{t+1} = \alpha_t + \frac{3}{2}\eta_G a_t \operatorname{Sigmoid}(-a_t \alpha_t^3 - b_t)\alpha_t^2$$

Where b_t be updated as: for $t = T_{B,2}$, $b_t = b^{(t)}$ and update as:

$$b_{t+1} = b_t - \eta_D \tau_b \operatorname{Sigmoid}(b_t)$$

Then we have: for every $t \in [T_{B,2}, T_{B,3}]$

$$a_t = \left(1 \pm \frac{1}{\operatorname{polylog}(d)}\right) a^{(t)}, \quad \alpha_t = \left(1 \pm \frac{1}{\operatorname{polylog}(d)}\right) \alpha(w_{i^*}, u_1, t)$$

$$\operatorname{Sigmoid}(b^{(t)}) = \left(1 \pm \frac{1}{\operatorname{polylog}(d)}\right) \operatorname{Sigmoid}(b_t)$$

Moreover, when $t = T_{B,3}$, we have: $a_t \leq 0.819\alpha_t$.

Proof of Lemma (F.5). By the update formula in Lemma (F.4) and the bound in induction hypothesis (B.3.3), we can simplify the update of $a^{(t)}, b^{(t)}$ and $\alpha(w_i, u_\ell, t)$ as: for $i = i^*$, when $X = u_\ell$:

$$a^{(t+1)} = a^{(t)} + \eta_D \left(1 \pm \frac{1}{\operatorname{polylog}(d)}\right) \operatorname{Sigmoid}\left(-a^{(t)} \alpha(w_i, u_\ell, t)^3 - b^{(t)}\right) \alpha(w_i, u_\ell, t)^3 \tag{69}$$

$$\alpha(w_i, u_\ell, t+1) = \alpha(w_i, u_\ell, t) \tag{70}$$

$$+ 3\eta_D \left(1 \pm \frac{1}{\operatorname{polylog}(d)}\right) \operatorname{Sigmoid}\left(-a^{(t)} \alpha(w_i, u_\ell, t)^3 - b^{(t)}\right) a^{(t)} \alpha(w_i, u_\ell, t)^2 \tag{71}$$

$$b^{(t)} = b^{(t)} - \eta_D \tau_b \left(1 \pm \frac{1}{\operatorname{polylog}(d)}\right) \operatorname{Sigmoid}\left(b^{(t)}\right) \tag{72}$$

By the last inequality, we know that when $\operatorname{Sigmoid}(b^{(t)}) > \left(1 + \frac{1}{\operatorname{polylog}(d)}\right) \operatorname{Sigmoid}(b_t)$, then $b^{(t)}$ must be decreasing faster than b_t, otherwise if $\operatorname{Sigmoid}(b^{(t)}) > \left(1 - \frac{1}{\operatorname{polylog}(d)}\right) \operatorname{Sigmoid}(b_t)$, then b_t must be decreasing faster than $b^{(t)}$, which proves the bound of $b^{(t)}$. Moreover, the update formula of b_t also gives us that for every $t \leq \mathrm{poly}(d)$, we have: $|b_t| = O(\log d)$. This implies that for every Z:

$$\mathrm{Sigmoid}(Z + b^{(t)}) = \left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \mathrm{Sigmoid}(Z + b_t)$$

To obtain the bound of $a^{(t)}$ and $\alpha(w_i^*, u_1, t)$, notice that when $X = u_1 + u_2$, we have that:

$$\mathrm{Sigmoid}\left(-a^{(t)}(\alpha(w_i, u_1, t) + \alpha(w_i, u_2, t))^3 - b^{(t)}\right) \leq \min_{\ell \in [2]} \mathrm{Sigmoid}\left(-a^{(t)}\alpha(w_i, u_\ell, t)^3 - b^{(t)}\right)$$

Therefore, we can conclude:

$$\mathbb{E}[a^{(t+1)}] = a^{(t)} \tag{73}$$

$$+ \eta_D \frac{1}{2}\left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right)\left(\sum_{\ell \in [2]} \mathrm{Sigmoid}\left(-a^{(t)}\alpha(w_i, u_\ell, t)^3 - b_t\right)\alpha(w_i, u_\ell, t)^3\right) \tag{74}$$

$$\mathbb{E}[\alpha(w_i, u_\ell, t+1)] = \alpha(w_i, u_\ell, t) \tag{75}$$

$$+ \frac{3}{2}\eta_D\left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \mathrm{Sigmoid}\left(-a^{(t)}\alpha(w_i, u_\ell, t)^3 - b_t\right) a^{(t)}\alpha(w_i, u_\ell, t)^2 \tag{76}$$

Using Lemma (F.3) we can conclude that

$$\alpha(w_{i^*}, u_\ell, T_{B,1}) = \frac{1}{\mathrm{polylog}(d)}$$

and now apply Lemma (D.7), we have: $a^{(t)} = \Theta(\alpha(w_{i^*}, u_1, t))$ and

$$\alpha(w_{i^*}, u_1, t) = [\alpha(w_{i^*}, u_1, t)]\left(1 \pm \frac{1}{\mathrm{polyloglog}(d)}\right)$$

This implies that:

$$\mathbb{E}[a^{(t+1)}] = a^{(t)} + \eta_D\left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right)\left(\mathrm{Sigmoid}\left(-a^{(t)}\alpha(w_i, u_1, t)^3 - b_t\right)\alpha(w_i, u_\ell, t)^3\right) \tag{77}$$

$$\mathbb{E}[\alpha(w_i, u_1, t+1)] = \alpha(w_i, u_1, t) \tag{78}$$

$$+ \frac{3}{2}\eta_D\left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \mathrm{Sigmoid}\left(-a^{(t)}\alpha(w_i, u_1, t)^3 - b_t\right) a^{(t)}\alpha(w_i, u_1, t)^2 \tag{79}$$

Apply Lemma (D.7) again, we know that when

$$a^{(t)}\alpha(w_i, u_1, t)^3 > \left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) a_t \alpha_t^3$$

We must have that $a^{(t)} \geq a_t$ and $\alpha(w_i, u_1, t) > \alpha_t$. Therefore, apply Lemma (D.6) we know that in this case:

$$\mathrm{Sigmoid}\left(-a^{(t)}\alpha(w_i, u_1, t)^3 - b_t\right)\alpha(w_i, u_\ell, t)^3 \leq \mathrm{Sigmoid}(-a_t\alpha_t^3 - b_t)\alpha_t^3$$

and

$$\mathrm{Sigmoid}\left(-a^{(t)}\alpha(w_i, u_1, t)^3 - b_t\right) a^{(t)}\alpha(w_i, u_1, t)^2 \leq \mathrm{Sigmoid}(-a_t\alpha_t^3 - b_t)a_t\alpha_t^2$$

Combine this with the update rule we can directly complete the proof.

$\square$

The Lemma (F.5) immediately implies that the $\alpha(w_{i^*}, u_\ell, t)$ will be balanced after a while:

Lemma F.6. *We have that for every $t \in [T_{B,2}, T_{B,3}]$, the following holds:*

$$\alpha(w_{i^*}, u_1, t) = [\alpha(w_{i^*}, u_1, t)] \left(1 \pm \frac{1}{\mathrm{polyloglog}(d)}\right)$$

and

$$\alpha(w_{i^*}, u_1, t) \geq \log^{0.1}(d)$$

Using Lemma (F.6), we also have the Lemma that approximate the update of $v_i^{(t)}$ as:

Lemma F.7. *Let us define $\alpha(t) := \max_{j\in[m_G],\ell\in[2]}\langle v_j^{(t)}, u_\ell\rangle$. For every $t \in [T_{B,2}, T_{B,3}]$, we have: for $j \neq i^*$:*

$$\langle w_j^{(t)}, G^{(t)}(z)\rangle^2 w_j^{(t)} = \langle w_j^{(t)}, G^{(t)}(z)\rangle^2 w_j^{(0)} \pm \frac{C^2}{d^{0.5}}\langle w_j^{(t)}, G^{(t)}(z)\rangle^2 \tag{80}$$

For $j = i^*$:

$$\langle w_j^{(t)}, G^{(t)}(z)\rangle^2 w_j^{(t)} = \langle w_j^{(t)}, G^{(t)}(z)\rangle^2 \left(w_j^{(0)} + \alpha(w_j, u_1, t)u_1 + \alpha(w_j, u_2, t)u_2\right) \pm \frac{C}{d^{0.9}}\langle w_j^{(t)}, G^{(t)}(z)\rangle^2 \tag{81}$$

Now, for $\langle w_j^{(t)}, G^{(t)}(z)\rangle$ we have: For $j \neq i^*$:

$$\mathbb{E}_z[\langle w_j^{(t)}, G^{(t)}(z)\rangle^2 \mid z_i = 1] = \left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right)\left(\langle w_j^{(0)}, v_i^{(t)}\rangle \pm \frac{C^2}{\sqrt{d}}\alpha(t)\right)^2 \pm \frac{\alpha(t)^2}{\mathrm{polylog}(d)} \tag{82}$$

For $j = i^*$:

$$\mathbb{E}_z[\langle w_j^{(t)}, G^{(t)}(z)\rangle^2 \mid z_i = 1] = \left(1 \pm \frac{1}{\mathrm{polyloglog}(d)}\right)\alpha(w_{i^*}, u_1, t)^2 \left\langle (u_1 + u_2), v_i^{(t)}\right\rangle^2 \tag{83}$$

$$\pm \frac{\alpha(t)^2}{\mathrm{polyloglog}(d)}\alpha(w_{i^*}, u_1, t)^2 \tag{84}$$

Moreover, the update of Sigmoid can be approximate as:

$$\mathrm{Sigmoid}\left(f^{(t)}\left(G^{(t)}(z)\right)\right) = \left(1 \pm \tilde{O}\left(\frac{1}{d^{1.45}}\right)\right)\mathrm{Sigmoid}\left(b^{(t)}\right)$$

Proof of Lemma (F.7). The first half of the lemma regarding $w_j^{(t)}$ follows trivially from the induction hypothesis, we only need to look at $\langle w_j^{(t)}, G^{(t)}(z)\rangle$.

We know that for $j \neq i^*$, we have that by the induction hypothesis,

$$\langle w_j^{(t)}, G^{(t)}(z)\rangle = \langle w_j^{(0)}, G^{(t)}(z)\rangle \pm \tilde{O}\left(\frac{1}{d^{1.35}}\right) \pm O\left(\frac{Cm_G\alpha(t)}{\sqrt{d}}\right) \tag{85}$$

For $j = i^*$, we have that

$$\langle w_j^{(t)}, G^{(t)}(z)\rangle = \langle w_j^{(0)}, G^{(t)}(z)\rangle + \alpha(w_j, u_1, t)\langle u_1, G^{(t)}(z)\rangle + \alpha(w_j, u_2, t)\langle u_2, G^{(t)}(z)\rangle \pm \tilde{O}\left(\frac{1}{d^{1.35}}\right) \tag{86}$$

$$= \alpha(w_j, u_1, t)\langle u_1 + u_2, G^{(t)}(z)\rangle \pm \frac{1}{\mathrm{polyloglog}(d)}\alpha(w_j, u_1, t)\alpha(t)\|z\|_1 \pm \tilde{O}\left(\frac{1}{d^{1.35}}\right) \tag{87}$$

$$= \alpha(w_j, u_1, t)\langle u_1 + u_2, v_i^{(t)}\rangle \pm \tilde{O}\left(\frac{1}{d^{1.35}}\right) \pm O(\alpha(w_j, u_1, t)\alpha(t))(\|z\|_1 - 1) \tag{88}$$

Taking expectation we can complete the proof.

□

With Eq equation (80) and Eq equation (81) in lemma (F.7), together with the induction hypothesis, we immediately obtain

Lemma F.8. *For every $t \in [T_{B,2}, T_{B,3}]$, we have that: for every $i \in [m_G]$:*

$$v_i^{(t)} = v_i^{(T_{B,2})} + \sum_{\ell \in [2]} \alpha_{i,\ell}^{(t)} u_\ell + \sum_{j \in [m_D]} \beta_{i,j}^{(t)} w_j^{(0)} \pm \xi_{i,t} \tag{89}$$

Where $\alpha_{i,\ell}^{(t)}, \beta_{i,j}^{(t)} > 0$ and $\alpha_{i,\ell}^{(t)} = (1 \pm o(1))\alpha_{i,3-\ell}^{(t)}$; $\|\xi_{i,t}\|_2^2 \leq \tilde{O}(1/d)\left(\sum_{\ell,j}(\alpha_{i,\ell}^{(t)})^2 + (\beta_{i,j}^{(t)})^2\right)$

We now can immediately control the update of $v_i^{(t)}$ using the following sequence:

Lemma F.9. *Let v_t be defined as: for $t = T_{B,2}$*

$$v_t = \max_{i \in [m_G]} \langle v_i^{(t)}, u_1 + u_2 \rangle \left(1 + \frac{1}{\text{polyloglog}(d)}\right)$$

and the update of v_t is given as: for α_t defined as in Lemma (F.5)

$$v_{t+1} = v_t + \frac{3}{m_G}\text{Sigmoid}(b_t)\alpha_t^2 v_t^2$$

Then we must have: for every $t \in [T_{B,2}, T_{B,3}]$:

$$\max_{i \in [m_G]} \langle v_i^{(t)}, u_1 + u_2 \rangle \leq v_t \tag{90}$$

On the other hand, if for $t = T_{B,2}$,

$$v_t = \max_{i \in [m_G]} \langle v_i^{(t)}, u_1 + u_2 \rangle \left(1 - \frac{1}{\text{polyloglog}(d)}\right)$$

Then we must have: for every $t \in [T_{B,2}, T_{B,3}]$:

$$\max_{i \in [m_G]} \langle v_i^{(t)}, u_1 + u_2 \rangle \geq v_t \tag{91}$$

Proof of Lemma (F.9). In the setting of Lemma (F.7), let us define $beta(t) := \max_{j \in [m_G]} \langle v_j^{(t)}, u_1 + u_2 \rangle$. We have that:

$$\beta(t+1) = \beta(t) + \eta_G \alpha_t \left(1 \pm \frac{1}{\text{polylog}(d)}\right) \frac{3}{m_G}\beta(t)^2 \pm \frac{1}{\text{polyloglog}(d)}\alpha(t)^2 \tag{92}$$

By the induction hypothesis we know that for all $j \in [m_G], \ell \in [2]$:

$$\langle v_j^{(t)}, u_\ell \rangle \geq \langle v_j^{(0)}, u_\ell \rangle - O\left(\frac{1}{d}\right) \geq -\frac{\log\log^2 d}{d} \tag{93}$$

This implies that $\beta(t) \geq \alpha(t) - \frac{\log\log^2(d)}{\sqrt{d}}$ and $\beta(T_{B,2}) \geq \frac{1}{d}$, $\alpha(T_{B,2}) \leq \frac{\text{polyloglog}(d)}{\sqrt{d}}$. This implies that:

$$\beta(t+1) = \beta(t) + \eta_G \alpha_t \left(1 \pm \frac{1}{\text{polyloglog}(d)}\right) \frac{3}{m_G}\beta(t)^2 \tag{94}$$

This completes the proof by applying Lemma (D.1). □

Now, by the comparison Lemma (D.1), we know that one of the following event would happen (depending on the initial value of v_t at iteration $T_{B,2}$):

Lemma F.10. *With probability $1 - o(1)$, one of the following would happen:*

1. $T_{B,3} \geq T_0$.
2. $T_{B,3} < T_0$, *moreover, at iteration $T_{B,3}$, we have that* $\mathrm{Sigmoid}(b_t) \geq \frac{1}{\mathrm{polylog}(d)}$.

In the end, we can easily derive an upper bound on the sum of Sigmoid as below, which will be used to prove the induction hypothesis.

Lemma F.11. *For every $t \in (T_{B,1}, T_{B,3}]$, we have that: for every X, z:*

$$\sum_{t \in (T_{B,1}, T_{B,3}]} \eta_D \,\mathrm{Sigmoid}\left(a^{(t)} \langle w_{i^*}^{(t)}, X\rangle^3 + b^{(t)}\right) = \tilde{O}(1) \tag{95}$$

$$\sum_{t \in (T_{B,1}, T_{B,3}]} \eta_D \tau_b \left(b^{(t)}\right) \leq \tilde{O}(1) \tag{96}$$

We will also show the following Lemma regarding all the $v_i^{(t)}$ at iteration $T_{B,3}$:

Lemma F.12. *For all $i \in [m_G]$, if we are in case 2 in Lemma (F.10), we have that:*

$$\langle v_i^{(t)}, u_1\rangle, \langle v_i^{(t)}, u_2\rangle = \tilde{\Omega}(d^{-0.49}) \tag{97}$$

Proof of Lemma (F.12). Since with probability at least $1/\mathrm{poly}(d)$, $z_i = z_j = 1$, so we have: By the update Lemma (F.7) of v, we know that for all $j \in [m_G]$: Let $\alpha(t)$ be defined as in Lemma (F.7):

$$\mathbb{E}[\alpha(v_j, u_\ell, t+1)] \geq \alpha(v_j, u_\ell, t) + \eta_G \frac{1}{\mathrm{polylog}(d)} \left(\alpha(t) - \tilde{O}\left(\frac{1}{d}\right)\right)^2 \pm \tilde{O}\left(\frac{1}{d^{0.8}}\right)\alpha(t)^2 \tag{98}$$

$$\mathbb{E}[\alpha(v_j, u_\ell, t+1)] \leq \alpha(v_j, u_\ell, t) + \eta_G \,\mathrm{polylog}(d)\alpha(t)^2 \tag{99}$$

By Lemma (F.10) we know that $\alpha(t) = \tilde{\Theta}(d^{-0.49})$ at iteration $t = T_{B,3}$, which implies what we want to prove. ☐

F.3 Stage 4 and 5

In Stage 4 we can easily calculate that by induction hypothesis, for every $i \in [m_G]$ and for every $j \in [m_D]$, $j \neq i^*$:

$$|\langle v_i^{(t)}, w_j^{(t)}\rangle| \leq \tilde{O}\left(\frac{\|v_i^{(t)}\|_2}{\sqrt{d}}\right)$$

Let

$$S_{i,t} = \mathbb{E}_z\left[\mathrm{Sigmoid}\left(a^{(t)}\sigma\left(\langle w_{i^*}^{(t)}, G^{(t)}(z)\rangle\right) + b^{(t)}\right) \mid z_i = 1\right]$$

Note that by induction hypothesis, $|a^{(t)}|, b^{(t)} = O(\log(d))$. Which implies that as long as $\max_{i \in [m_G]} \|v_i^{(t)}\|_2 \leq \frac{1}{\mathrm{polylog}(d)}$ or for all $i \in [m_G]$, $a^{(t)}\sigma'(\langle v_i^{(t)}, w_{i^*}\rangle) \geq \tilde{\Omega}(\log(d))$, we have that: for all z, z' we have that:

$$\mathrm{Sigmoid}\left(a^{(t)}\sigma\left(\langle w_{i^*}^{(t)}, G^{(t)}(z)\rangle\right) + b^{(t)}\right) = \Theta(1) \times \mathrm{Sigmoid}\left(a^{(t)}\sigma\left(\langle w_{i^*}^{(t)}, G^{(t)}(z')\rangle\right) + b^{(t)}\right)$$

We can immediately obtain the following Lemma:

Lemma F.13. *The update of $v_i^{(t)}$ is given as:*

$$\mathbb{E}[v_i^{(t+1)}] = v_i^{(t)} + \tilde{\Theta}(\eta_G)a^{(t)}S_{i,t}\left(\left(\sigma'(\langle w_{i^*}^{(t)}, v_i^{(t)}\rangle + \sum_{j \in [m_G]} \gamma_{i,j,t}\sigma'(\langle w_{i^*}^{(t)}, v_j^{(t)}\rangle)\right) w_i^{(t)} \pm \tilde{O}\left(\frac{\|v_i^{(t)}\|_2}{\sqrt{d}}\right)^2\right) \tag{100}$$

Where $\gamma_{i,j,t} > 0$*;* $\gamma_{i,j,t} = \tilde{\Theta}(1)$ *if* $\max_{i\in[m_G]} \|v_i^{(t)}\|_2 \leq \frac{1}{\mathrm{polylog}(d)}$ *or for all* $i \in [m_G]$, $a^{(t)}\sigma'(\langle v_i^{(t)}, w_{i^*}\rangle) \geq \tilde{\Omega}(\log(d))$*, and* $\gamma_{i,j,t} = \tilde{O}(1)$ *otherwise.*

Here the additional $\sigma'(\langle w_{i^*}^{(t)}, v_j^{(t)}\rangle)$ part comes from $\Pr[z_i, z_j = 1] = \frac{1}{\mathrm{polylog}(d)}$. The remaining part of this stage follows from simply apply Lemma (D.4).

In stage 5, we bound the update of $a^{(t)}, b^{(t)}$ as:

Let

$$S_t = \mathbb{E}_z\left[\mathrm{Sigmoid}\left(a^{(t)}\sigma\left(\langle w_{i^*}^{(t)}, G^{(t)}(z)\rangle\right) + b^{(t)}\right)\right]$$

In this stage, with the induction hypothesis, we can easily approximate the sigmoid as:

Lemma F.14. *For* $t \geq T_{B,4}$*, the sigmoid can be approximate as: For every* X, z

$$\mathrm{Sigmoid}(-f^{(t)}(X)) = \left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right)\mathrm{Sigmoid}(-b^{(t)})$$

$$\mathrm{Sigmoid}(f^{(t)}(G^{(t)}(z))) = \left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right)\mathrm{Sigmoid}\left(a^{(t)}\sigma\left(\langle w_{i^*}^{(t)}, G^{(t)}(z)\rangle\right) + b^{(t)}\right)$$

Then by the update rule, we can easily conclude that:

Lemma F.15. *For* $t \geq T_{B,4}$*, the update of* $a^{(t)}, b^{(t)}$ *is given as:*

$$a^{(t+1)} = a^{(t)} + \tilde{O}(\eta_D)\,\mathrm{Sigmoid}\left(-b^{(t)}\right) - \tilde{\Omega}(\eta_D)S_t\Lambda^2 d^{1/4}$$

$$\mathbb{E}[b^{(t+1)}] = b^{(t)} + \eta_D\tau_b\left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right)\mathrm{Sigmoid}\left(-b^{(t)}\right) - \eta_D\tau_b\left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right)S_t$$

This Lemma, together with the induction hypothesis, implies that:

Lemma F.16. *We have:*

$$\sum_{t\geq T_{B,4}} S_t \leq \tilde{O}\left(\frac{1}{\eta_D\tau_b\Lambda^2 d^{1/4}}\right) \tag{101}$$

$$\sum_{t\geq T_{B,4}} \mathrm{Sigmoid}(b^{(t)}) \leq \tilde{O}\left(\frac{1}{\eta_D\tau_b}\right) \tag{102}$$

Proof of Lemma (F.16). Let us denote $R = \sum_{t=T_{B,4}}^{T_0}\mathrm{Sigmoid}\left(-b^{(t)}\right)$ and $Q = \sum_{t=T_{B,4}}^{T_0} S_t$

Sum the update up for $t = T_{B,4}$ to T_0, we have that:

$$a^{(T_0)} - a^{(T_{B,4})} = \tilde{O}(\eta_D)R - \tilde{\Omega}(\eta_D)Q\Lambda^2 d^{1/4} \tag{103}$$

$$\mathbb{E}[b^{(T_0)}] - b^{(T_{B,4})} = \Theta(\eta_D\tau_b)R - \Theta(\eta_D\tau_b)Q \tag{104}$$

By the induction hypothesis that $|a^{(t)}| \leq \frac{\tilde{O}(1)}{\Lambda^2 d^{1/4}}$ and $|b^t| = \tilde{O}(1)$, we have that:

$$|\tilde{O}(\eta_D)R - \tilde{\Omega}(\eta_D)Q\Lambda^2 d^{1/4}| \leq \frac{\tilde{O}(1)}{\Lambda^2 d^{1/4}} \tag{105}$$

$$|\Theta(\eta_D\tau_b)R - \Theta(\eta_D\tau_b)Q| \leq \tilde{O}(1) \tag{106}$$

Thus, we have:

$$\tilde{\Omega}(\eta_D)Q\Lambda^2 d^{1/4} \leq \tilde{O}(\eta_D)R + \frac{\tilde{O}(1)}{\Lambda^2 d^{1/4}} \leq \tilde{O}(\eta_D)\left(\frac{1}{\eta_D\tau_b} + Q\right) + \frac{\tilde{O}(1)}{\Lambda^2 d^{1/4}} \tag{107}$$

Therefore we have that $\tilde{\Omega}(\eta_D)Q\Lambda^2 d^{1/4} \leq \tilde{O}\left(\frac{1}{\eta_D \tau_b} + \frac{1}{\eta_D \Lambda^2 d^{1/4}}\right)$, which implies that

$$Q \leq \tilde{O}\left(\frac{1}{\eta_D \tau_b \Lambda^2 d^{1/4}}\right)$$

Similarly, we can show that

$$R \leq \tilde{O}\left(\frac{1}{\eta_D \tau_b} + Q\right) \leq \tilde{O}\left(\frac{1}{\eta_D \tau_b} + \frac{1}{(\Lambda^2 d^{1/4})^2} + \frac{1}{\Lambda^2 d^{1/4}} R\right) \tag{108}$$

This implies that $R \leq \tilde{O}\left(\frac{1}{\eta_D \tau_b}\right)$.

□

F.4 Proof of the induction hypothesis and the final theorem

The final theorem follows immediately from the induction hypothesis (v part) together with Lemma (F.8).

Now it remains to prove the induction hypothesis. We will assume that all the hypothesises are true until iteration t, then we will prove that they are true at iteration $t+1$.

Stage 1 .

To prove the induction hypothesis at Stage 1, for w, we have that by Lemma (F.1), we know that: for $* \neq u_1, u_2$,

$$|\alpha(w_i, *, t+1) - \alpha(w_i, *, t)| \leq \eta_D \tilde{O}\left(\frac{1}{d^{1.5}}\right) \tag{109}$$

By $T_1 \leq O\left(\frac{\sqrt{d}}{\eta_D a^{(0)}}\right)$ we can conclude that

$$|\alpha(w_i, *, t+1) - \alpha(w_i, *, 0)| \leq \eta_D \tilde{O}\left(\frac{1}{d^{1.5}}\right) \times O\left(\frac{\sqrt{d}}{\eta_D a^{(0)}}\right) \leq \frac{1}{d^{0.9}} \tag{110}$$

On the v part, again by Lemma (F.1), we know that for $* \notin \{w_j\}_{j \in [m_D]}$:

$$|\alpha(v_i, *, t) - \alpha(v_i, *, 0)| \leq \eta_G \left(\frac{1}{C^{0.5} d^2}\right) \times O\left(\frac{\sqrt{d}}{\eta_D a^{(0)}}\right) \leq \frac{1}{d} \tag{111}$$

On the other hand, we know that for w_j:

$$|\alpha(v_i, w_j, t+1) - \alpha(v_i, w_j, t)| \leq \eta_G \left(1 + \frac{1}{\mathrm{polylog}(d)}\right) \frac{3}{2m_G} \zeta_t^2 \tag{112}$$

Apply Lemma (F.2) we complete the proof using Lemma (D.3).

As for the $a^{(t)}, b^{(t)}$ part, we know that:

$$|a^{(t+1)} - a^{(t)}| \leq O\left(\eta_D m_D \Upsilon_t^3\right), |b^{(t)}| \leq \tau_b \eta_D T_1 \tag{113}$$

Combine with the update rule of Υ in Eq equation (55), we complete the proof.

Stage 2 and 3 For the w part, we know that by Lemma (F.4), we have that for every $* \neq u_1, u_2$

$$|\alpha(w_i, *, t+1) - \alpha(w_i, *, t)| \leq \eta_D \tilde{O}\left(\frac{1}{d^{1.45}}\right) \mathrm{Sigmoid}(b^{(t)}) \tag{114}$$

Now, by Lemma (F.11) we have that:

$$\sum_{t \in (T_{B,1}, T_{B,3}]} \eta_D \tau_b \left(b^{(t)}\right) \leq \tilde{O}(1)$$

This implies that

$$|\alpha(w_i, *, t+1) - \alpha(w_i, *, T_{B,1})| \leq \eta_D \tilde{O}\left(\frac{1}{d^{1.45}}\right) \times \frac{1}{\eta_b \eta_D} \leq \frac{1}{d^{0.9}}$$

For the v part for $t \leq T_{B,2}$, since $T_{B,2} - T_{B,1} = \tilde{O}(d^{o(1)}/\eta_D)$, we can easily prove it for $t \leq T_{B,2}$ as in stage 1. On the other hand, for $t \in (T_{B,2}, T_{B,3}]$: By Lemma (F.7) and Lemma (F.6), we have that define

$$\alpha(t) := \max_{j \in [m_G], \ell \in [2]} \langle v_j^{(t)}, u_\ell \rangle, \quad \beta(t) := \max_{j \in [m_G], j' \in [m_G], j \neq j'; i \in [m_D], i \neq i^*} |\langle v_j^{(t)}, w_i^{(0)} \rangle| + |\langle v_j^{(t)}, v_{j'}^{(0)} \rangle|$$

We have that:

$$\mathbb{E}[\alpha(t+1)] \geq \alpha(t) + \eta_G \Omega\left(\frac{1}{m_G}\right) \mathrm{Sigmoid}(b^{(t)}) \alpha(t)^2 \log^{0.1}(d) \tag{115}$$

and

$$\mathbb{E}[\beta(t+1)] \leq \beta(t) + \eta_G O\left(\frac{1}{m_G}\right) \mathrm{Sigmoid}(b^{(t)}) \left(\beta(t)^2 + \frac{C^2}{\sqrt{d}} \alpha(t)^2\right) \tag{116}$$

By Lemma (C.2) and Lemma (D.1) we can show that $\beta(t) = O\left(\beta(0) + \frac{C^2}{\sqrt{d}}\alpha(t)\right)$, which complete the proof that for all $* \neq w_{i^*}, u_1, u_2$:

$$|\alpha(v_j, *, t)| \leq \frac{C^3}{\sqrt{d}} \|v_j^{(t)}\|_2$$

.

Stage 4 and 5 At stage 4 we simply use Lemma (D.4), the only remaining part is to show that $|b^{(t)}| = O(\log(d))$. To see this, we know that by the update formula:

$$\nabla_b L(X, z) = -\mathrm{Sigmoid}(-f(X)) + \mathrm{Sigmoid}(f(G(z)))$$

By our induction hypothesis, we know that $a^{(t)} \left(\sum_{i \in [m_D]} \sigma(\langle w_i^{(t)}, X \rangle)\right) > 0$

and $a^{(t)} \left(\sum_{i \in [m_D]} \sigma(\langle w_i^{(t)}, G(z) \rangle)\right) > 0$. Therefore, $b < O(\log(d))$ is immediate. Now it remains to show that $b > -O(\log d)$: By the update formula, we have:

$$-b^{(t+1)} \leq -b^{(t)} + \eta_D \tau_b \sum_{i \in [m_G]} S_{i,t}$$

and by Lemma (F.13) and the proof in Lemma (D.4), we have that:

$$\sum_{i \in [m_G]} \langle v_i^{(t+1)}, w_{i^*}^{(0)} \rangle \geq \sum_{i \in [m_G]} \langle v_i^{(t)}, w_{i^*}^{(0)} \rangle + a^{(t)} \tilde{\Omega}(\eta_G) \left(\sum_{i \in [m_G]} S_{i,t}\right) \left(\sum_{i \in [m_G]} \sigma'(\langle v_i^{(t)}, w_{i^*}^{(0)} \rangle)\right) \tag{117}$$

Compare this two updates we can easily obtain that $|b^{(t)}| = O(\log(d))$.

At stage 5, we have that since $|a^{(t)}| = \tilde{O}\left(\frac{1}{d^{1/4}\Lambda^2}\right)$: For every $j \in [m_G], i \in [m_D]$

$$\|v_j^{(t+1)} - v_j^{(t)}\|_2 \leq \tilde{O}(\eta_G) S_t \Lambda^2 \times \frac{1}{\Lambda^2 d^{1/4}} \tag{118}$$

$$\|w_i^{(t+1)} - w_i^{(t)}\|_2 \leq \tilde{O}(\eta_D) \operatorname{Sigmoid}(-b^{(t)}) \frac{1}{d^{1/4}\Lambda^2} \tag{119}$$

Apply Lemma (F.16) we have that:

$$\|v_j^{(t+1)} - v_j^{(T_{B,4})}\|_2 \leq \tilde{O}(\eta_G) \times \frac{1}{d^{1/4}} \times \frac{1}{\eta_D \tau_b \Lambda^2 d^{1/4}} \leq d^{0.15} \tag{120}$$

$$\|w_i^{(t+1)} - w_i^{(T_{B,4})}\|_2 \leq \tilde{O}(\eta_D) \frac{1}{d^{1/4}\Lambda^2} \times \frac{1}{\eta_D \tau_b} \leq \frac{1}{d^{0.1}} \tag{121}$$

Which proves the induction hypothesis.

G Proof of the learning process in other cases

.

We now consider other cases, in case 1 of Lemma (C.1), the proof is identical to case 2, the only difference is at Stage 3, we have that $T_{B,3} > T_0$.

In case 2, the Stage 1 is identical to the Stage 1, 2, 3 in the balanced case. For Stage 3, its identical to Stage 4 in the balanced case (the only difference is to apply Lemma (D.5) and the case 2 of Lemma (D.2) instead of Lemma (D.4)). For Stage 4, its identical to Stage 5 in the balanced case.

At Stage 2, by the induction hypothesis, we know that for $j \neq i_G$, we have that $|\langle v_j^{(t)}, w_j^{(t)} \rangle| \leq \tilde{O}\left(\frac{1}{C}\|v_j^{(t)}\|_2\right)$. Thus, we can approximate the update of w, v as:

$$w_i^{(t+1)} = w_i^{(t)} \pm \tilde{O}(\eta_D \sum_{j \in [m_G]} \|v_j^{(t)}\|_2^2)\Lambda \tag{122}$$

$$v_j^{(t+1)} = v_j^{(t)} + \tilde{\Theta}(\eta_G \sum_{j \in [m_G]} \|v_j^{(t)}\|_2^2) w_{i_G}^{(0)} \pm \frac{1}{C^2}\tilde{\Theta}(\eta_G \sum_{j \in [m_G]} \|v_j^{(t)}\|_2^2) \tag{123}$$

Using the fact that $\eta_G \geq \tilde{\Omega}(\sqrt{d})\eta_D$ in case 3 we immediately proves the induction hypothesis.

The proof of the theorem follows immediately from the induction hypothesis on v in this case v only learns noises (linear combinations of $w_i^{(0)}$).

H Normalized SGD

In this section we look at the update of normalized SGD.

Let us define:

$$i_1^* = \arg\max_{i \in [m_D]} \{\langle w_i^{(0)}, u_1 \rangle\}$$

$$i_2^* = \arg\max_{i \in [m_D]} \{\langle w_i^{(0)}, u_2 \rangle\}$$

Let us define:

$$g_j^* = \arg\max_{i \in [m_D]} \{\langle v_j^{(0)}, w_i^{(0)} \rangle\}$$

Then we first show the following Lemma about initialization:

Lemma H.1. *With probability at least $1 - o(1)$ over the randomness of the initialization, the following holds:*

1. *For all $\ell \in [2]$, for all $i \in [m_D]$ such that $i \neq i^*_\ell$, we have:*

$$\langle w^{(0)}_{i^*_\ell}, u_\ell \rangle \geq \left(1 - \frac{1}{\mathrm{polyloglog}(d)}\right) \langle w^{(0)}_i, u_\ell \rangle$$

2. *For all $j \in [m_G]$, we have that for all $i \in [m_D]$ such that $i \neq g^*_j$,*

$$\langle v^{(0)}_j, w^{(0)}_{g^*_j} \rangle \geq \left(1 - \frac{1}{\log^4(d)}\right) \langle v^{(0)}_j, w^{(0)}_i \rangle$$

3. $\{g^*_j\}_{j \in [m_G]} = [m_D]$.

We now divide the training stage into two: For a sufficiently large $C = \mathrm{polylog}(d)$, consider the case when $\eta_G = \eta_D * C^{-0.6}$.

1. Stage 1: When both $\alpha(w_{i*_1}, u_1, t), \alpha(w_{i*_2}, u_2, t) \leq \frac{1}{C^{0.95}}$. Call this iteration $T_{N,1}$.
2. Stage 2: After $T_{N,1}$, before T_1

H.1 Induction Hypothesis

We will use the following induction hypothesis: for a

Stage 1: for every $t \leq T_{N,1}$: Let $\alpha(t) := \max_{\ell \in [2]} \alpha(w_{i^*_\ell}, u_\ell, t)$, $\beta(t) := \max_{i \in [m_G]} \alpha(v_i, w_{g^*_i}, t)$.

1. Domination: For every $i \in [m_G]$, we have:

$$|\alpha(v_i, *, t) - \alpha(v_i, *, 0)| \leq \min\left\{\frac{1}{C}\alpha(t), \beta(t)\right\}$$

   For every $i \in [m_D]$, $i \neq i^*_1, i^*_2$, we have that for $* \neq u_1, u_2$:

$$|\alpha(w_i, *, t) - \alpha(w_i, *, 0)| \leq \frac{1}{C}\alpha(t)$$

   and

$$|\alpha(w_i, u_1, t)|, |\alpha(w_i, u_2, t)| \leq \alpha(t)$$

   For i^*_1, we have that for every $* \neq u_1$,

$$|\alpha(w_{i^*_1}, *, t) - \alpha(w_{i^*_1}, *, 0)| \leq \frac{1}{C}\alpha(t)$$

   For i^*_2, we have that for every $* \neq u_2$,

$$|\alpha(w_{i^*_2}, *, t) - \alpha(w_{i^*_2}, *, 0)| \leq \frac{1}{C}\alpha(t)$$

2. (N.1.2): Growth rate: we have that for every $i \in [m_D]$

$$\alpha(w_{i^*_\ell}, u_\ell, t) \in \left(\Omega\left(\frac{1}{m_D}\right), 1\right) \eta_D t$$

   and for every $i \in [m_G]$:

$$\alpha(v_i, w_{g^*_i}, t) \in \left(\Omega\left(\frac{1}{m_G^2}\right), 1\right) \eta_G t$$

   Therefore by our choice of η_D, η_G we have that $\beta(t) \in C^{-0.6}\left[\frac{1}{\log^5 d}, \log^5 d\right] \times \alpha(t)$.

3. $a^{(t)} = [0.5, 1]a^{(0)}, |b^{(t)}| \leq \frac{1}{d^{0.1}}$.

Stage 2: We maintain: For every $t \in [T_{N,1}, T_1]$:

1. (N.1.2) still holds.
2. $\alpha(w_{i_\ell^*}, u_\ell, t) \in \left[\frac{1}{C}, \text{polylogloglog}(d)\right]$, $\beta(t) \in C^{-0.6}\left[\frac{1}{\log^5 d}, \log^5 d\right] \times \alpha(t)$, $a^{(t)} = \Omega(\alpha(w_{i_\ell^*}, u_\ell, t))$.
3. w_i's are good: For every $i \neq i_1^*, i_2^*$, for every $*$:
$$|\alpha(w_i, *, t) - \alpha(w_i, *, 0)| \leq \frac{1}{C}\alpha(t)$$
and for $\ell \in [2]$: for every $* \neq u_\ell$, we have:
$$|\alpha(w_i, *, t) - \alpha(w_i, *, 0)| \leq \frac{1}{C}\alpha(t)$$
4. v_i's are good: For every $i \in [m_G]$ and every $j \in [m_D]$, $j \neq g_i^*$:
$$\langle v_i^{(t)}, w_{g_i^*}^{(t)}\rangle \geq C^{0.9}|\langle v_i^{(t)}, w_j^{(t)}\rangle|$$
and for $g_i^* \neq i_\ell^*$, we have that:
$$\langle v_i^{(t)}, w_{g_i^*}^{(t)}\rangle \geq C^{0.9}|\langle v_i^{(t)}, u_\ell\rangle|$$
For $g_i^* = i_\ell$, we have that $\langle v_i^{(t)}, u_\ell\rangle \geq -\frac{1}{C^{0.5}}\beta(t)$

H.2 Stage 1 training

With the induction hypothesis, we can show the following Lemma:

Lemma H.2. *For $t \leq T_{N,1}$, for $\varepsilon_t := \frac{(\alpha(t)+d^{-0.5})^2}{C^{1.5}} + C^{0.5}(\alpha(t) + d^{-0.5})^3$, when the sample is $X \in \{u_1, u_2\}$, the update of $w_i^{(t)}$ can be approximate as:*

$$w_i^{(t+1)} = w_i^{(t)} + \eta_D\left(1 \pm \frac{1}{\text{polylog}(d)}\right)\frac{\sigma'(\langle w_i^{(t)}, X\rangle)X \pm \varepsilon_t}{\sqrt{\sum_{j\in[m_D]}\sigma'(\langle w_j^{(t)}, X\rangle)^2\|X\|_2^2}} \tag{124}$$

Which can be further simplified as:

$$\mathbb{E}[\langle w_i^{(t+1)}, u_\ell\rangle] = \langle w_i^{(t)}, u_\ell\rangle + \eta_D\frac{1}{2}\left(1 \pm \frac{1}{\text{polylog}(d)}\right)\frac{\sigma'(\langle w_i^{(t)}, u_\ell\rangle) \pm \varepsilon_t}{\sqrt{\sum_{j\in[m_D]}\sigma'(\langle w_j^{(t)}, u_\ell\rangle)^2}} \pm \eta_D\gamma \tag{125}$$

When $z = e_i$, the update of v can be approximate as: For $\delta_t := O\left(\frac{1}{C^{0.94}}(\beta(t) + \frac{1}{d})^2\right)$:

$$v_i^{(t+1)} = v_i^{(t)} + \eta_G\left(1 \pm \frac{1}{\text{polylog}(d)}\right)\frac{\sum_{j\in[m_D]}\sigma'(\langle v_i^{(t)}, w_j^{(0)}\rangle)w_j^{(0)} \pm \delta_t}{\|\sum_{j\in[m_D]}\sigma'(\langle v_i^{(t)}, w_j^{(0)}\rangle)w_j^{(0)}\|_2} \tag{126}$$

Where we have:

$$\langle v_i^{(t)}, w_j^{(t)}\rangle = \langle v_i^{(t)}, w_j^{(0)}\rangle \pm \frac{1}{C^{0.9}}\beta(t)$$

Proof of the update Lemma (H.2). By the induction hypothesis, We have that:

$$\begin{aligned}
\langle v_i^{(t)}, w_j^{(t)}\rangle &= \langle v_i^{(t)}, w_j^{(0)}\rangle + \langle v_i^{(t)}, w_j^{(t)} - w_j^{(0)}\rangle &(127)\\
&= \langle v_i^{(t)}, w_j^{(0)}\rangle + \langle v_i^{(0)}, w_j^{(t)} - w_j^{(0)}\rangle + \langle v_i^{(t)} - v_i^{(0)}, w_j^{(t)} - w_j^{(0)}\rangle &(128)\\
&= \langle v_i^{(t)}, w_j^{(0)}\rangle \pm \tilde{O}\left(\frac{1}{\sqrt{d}}\alpha(t)\right) \pm O\left(\frac{1}{C^{0.94}}\beta(t)\right) &(129)\\
&= \langle v_i^{(t)}, w_j^{(0)}\rangle \pm \frac{1}{C^{0.9}}\beta(t) &(130)
\end{aligned}$$

Here we use the fact that $\|w_j^{(0)} - w_j^{(t)}\|_2 \le O\left(\frac{1}{C^{0.95}}\right)$ from the induction hypothesis.

Consider the update of w_i, we have that: at stage 1, we must have $|f(X)|, |f(G(z))| \le \frac{1}{\mathrm{polylog}(d)}$. Therefore,

$$\nabla_{w_i} L(X, z) = \left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) a^{(t)} \sigma'(\langle w_i^{(t)}, X\rangle) X - a^{(t)} \sigma'(\langle w_i^{(t)}, G^{(t)}(z)\rangle) G^{(t)}(z) \quad (131)$$

By the induction hypothesis, we have that by $\beta(t) \le \alpha(t)$, it holds that:

$$\|\sigma'(\langle w_i^{(t)}, G^{(t)}(z)\rangle) G^{(t)}(z)\|_2 \le O\left(\frac{\alpha(t)}{C} + \frac{1}{C\sqrt{d}}\right)^2 m_G^2 \times m_G \left(\frac{1}{\sqrt{d}} + \frac{\alpha(t)}{C}\right) \le \epsilon_t$$

On the other hand, we must have that when $X = u_\ell$, we have

$$\left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \sigma'(\langle w_{i_\ell^*}^{(t)}, X\rangle) \ge \left(\frac{1}{m_D}\alpha(t) + \frac{1}{\sqrt{d}}\right)^2 \ge \mathrm{polylog}(d)\epsilon_t \quad (132)$$

This completes the proof of the w_i part. For v_i part the proof is the same using the fact that $\|w_j^{(0)} - w_j^{(t)}\|_2 \le O\left(\frac{1}{C^{0.95}}\right)$ from the induction hypothesis.

□

H.3 Stage 2 training

In this stage, we can maintain the following simple update rule: For w_i:

Lemma H.3. *For every $t \in (T_{N,1}, T_1]$, we have that: for every $i \in [m_D]$, for $i = i_\ell^*$:*

$$\mathbb{E}[w_i^{(t+1)}] = w_i^{(t)} + \Theta(\eta_D) u_\ell \pm \eta_D \frac{1}{C^{1.501}} \pm \eta_D \gamma$$

and for $i \neq i_1^*, i_2^*$,

$$\mathbb{E}[w_i^{(t+1)}] = w_i^{(t)} \pm \eta_D \frac{1}{C^{1.501}} \pm \eta_D \gamma$$

For v_i:

$$\mathbb{E}[v_i^{(t+1)}] = v_i^{(t)} + \left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \frac{1}{m_G} \eta_G \frac{w_{g_i^*}^{(t)}}{\|w_{g_i^*}^{(t)}\|_2} \pm \eta_G \frac{1}{C^{1.5}} \quad (133)$$

Proof of Lemma (H.3). This Lemma can be proved identically to Lemma (H.2): By the induction hypothesis, we have

$$|\langle w_i^{(t)}, v_j^{(t)}\rangle| \le \log^5 \beta(t) \quad (134)$$

Therefore,

$$\|\sigma'(\langle w_i^{(t)}, G^{(t)}(z)\rangle) v_j^{(t)}\|_2 \le C^{0.01}\beta(t)^3 \le \frac{1}{C^{1.51}}\alpha(t)^2$$

Which implies that:

$$w_i^{(t+1)} = w_i^{(t)} + \eta_D \frac{\sigma'(\langle w_i^{(t)}, X\rangle) X \pm \frac{1}{C^{1.51}\alpha(t)^2}}{\sqrt{\sum_{j\in[m_D]} (\sigma'(\langle w_j^{(t)}, X\rangle)^2 \|X\|_2^2 + \sum_{j\in[m_D]} \left(\sigma'(\langle w_j^{(t)}, X\rangle)^2\right)^3}} \quad (135)$$

Where $\sum_{j\in[m_D]} \left(\sigma'(\langle w_j^{(t)}, X\rangle)^2\right)^3$ comes from the gradient of $a^{(t)}$. By the induction hypothesis we have that $a^{(t)} = \Omega(\alpha(w_{i_\ell^*}, u_\ell, t))$, so we have

$$w_i^{(t+1)} = w_i^{(t)} + \Theta(\eta_D) \frac{\sigma'(\langle w_i^{(t)}, X\rangle) X \pm \frac{1}{C^{1.51}\alpha(t)^2}}{\sqrt{\sum_{j\in[m_D]} (\sigma'(\langle w_j^{(t)}, X\rangle)^2 \|X\|_2^2}} \quad (136)$$

On the other hand, by the induction hypothesis, for $\ell \in 2[$: For $i = i_\ell^*$: $\langle w_i^*, u_\ell \rangle \geq \frac{1}{m_D}\alpha(t)$, and for $i \neq i_1^*, i_2^*$: $|\langle w_i^*, X\rangle| \leq O\left(\frac{1}{C}\alpha(t)\right)$.

This implies that: for $i = i_\ell^*$:

$$\mathbb{E}[w_i^{(t+1)}] = w_i^{(t)} + \Theta(\eta_D)u_\ell \pm \eta_D \frac{1}{C^{1.5}} \pm \eta_D \gamma$$

and for $i \neq i_1^*, i_2^*$,

$$\mathbb{E}[w_i^{(t+1)}] = w_i^{(t)} \pm \eta_D \frac{1}{C^{1.5}} \eta_D \gamma$$

Where the additional γ factor comes from the case when $X = u_1 + u_2$ or $X = 0$.

On the other hand, we also know that:

$$\sum_{i \in [m_D]} \sigma'(\langle w_i^{(t)}, v_j^{(t)} \rangle) w_i^{(t)} \tag{137}$$

$$= \sigma'(\langle w_{g_j^*}^{(t)}, v_j^{(t)} \rangle) w_{g_j^*}^{(t)} \pm m_D \left(\frac{1}{C^{0.9}}\right)^2 \langle w_{g_j^*}^{(t)}, v_j^{(t)} \rangle^2 \operatorname{polylogloglog}(d) \tag{138}$$

$$= \sigma'(\langle w_{g_j^*}^{(t)}, v_j^{(t)} \rangle) w_{g_j^*}^{(t)} \pm \sigma'(\langle w_{g_j^*}^{(t)}, v_j^{(t)} \rangle) \frac{1}{C^{1.6}} \tag{139}$$

Notice that $\|w_i^{(t)}\|_2 = \Omega(1)$ so we complete the proof. □

H.4 Proof of the induction hypothesis

Now it remains to prove the induction hypothesis:

Stage 1: In this stage, we will use the update Lemma (H.2). By the induction hypothesis we know that for $X = u_\ell$,

$$\langle w_j^{(t)}, X \rangle = \alpha(w_j, u_\ell, t) + \alpha(w_j, w_j, 0) \left\langle \frac{w_j^{(0)}}{\|w_j^{(0)}\|_2}, u_\ell \right\rangle \pm O\left(\frac{1}{C^{0.5}\sqrt{d}}\right) \tag{140}$$

This implies that

$$\sum_j \sigma'(\langle w_j^{(t)}, X \rangle)^2 \|X\|_2^2 \geq \left(\frac{1}{m_D}\alpha(t) + \frac{1}{\sqrt{d}}\right)^2$$

Now, apply Lemma (H.2) we know that:

$$\alpha(t+1) \geq \alpha(t) + \Omega\left(\frac{1}{m_D}\right)\eta_D \tag{141}$$

$$\forall * \neq u_1, u_2 : |\alpha(w_i, *, t+1)| \leq |\alpha(w_i, *, t)| + \eta_D \frac{\epsilon_t}{\left(\frac{1}{m_D}\alpha(t) + \frac{1}{\sqrt{d}}\right)} \leq |\alpha(w_i, *, t)| + \eta_D \frac{1}{C^{1.4}} \tag{142}$$

Compare these two updates we can prove the bounds on w_j for $* \neq u_1, u_2$. For $* = u_1, u_2$, we can see that: By Lemma (H.2), there exists $S_{t,\ell} \in (0, \operatorname{poly}(d)]$ such that for $\ell \in [2]$ such that for every $i \in [m_D]$:

$$\sum_{i \in [m_D]} \langle w_i^{(t)}, u_\ell \rangle^4 = \frac{1}{S_{t,\ell}^2} \tag{143}$$

$$\mathbb{E}[\langle w_i^{(t+1)}, u_\ell \rangle] = \langle w_i^{(t)}, u_\ell \rangle + \eta_D \left(1 \pm \frac{1}{\operatorname{polylog}(d)}\right) S_{t,\ell} \langle w_i^{(t)}, u_\ell \rangle^2 \pm \eta_D \frac{1}{\operatorname{polylog}(d)} \tag{144}$$

Apply Lemma (D.4) and Lemma (H.1) we can complete the proof that

$$|\alpha(w_i, u_1, t)|, |\alpha(w_i, u_2, t)| \leq \alpha(t), \quad \alpha(w_{i_\ell}, u_\ell, t) \geq \frac{\eta_D}{3m_D}$$

and at iteration $t = T_{N,1}$, we have that: for all $i \neq i_1^*, i_2^*$, for all $\ell \in [2]$

$$|\alpha(w_i, u_\ell, t)| \leq \frac{1}{C}\alpha(t)$$

Moreover, when $i = i_{\ell^*}$, $|\alpha(w_i, u_{3-\ell}, t)| \leq \frac{1}{C}\alpha(t)$

The v part can be proved similarly: We have that there exists $S_{t,i} \in (0, \mathrm{poly}(d)]$ where $i \in [m_G]$ such that:

$$\sum_{j \in [m_D]} \langle v_i^{(t)}, w_j^{(0)} \rangle^4 = \frac{1}{S_{t,i}^2} \tag{145}$$

$$\mathbb{E}[\langle v_i^{(t+1)}, w_j^{(0)} \rangle] = \langle v_i^{(t)}, w_j^{(0)} \rangle + \eta_G \frac{1}{m_G}\left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) S_{t,i} \langle v_i^{(t)}, w_j^{(0)} \rangle^2 \pm \eta_G \frac{\log^5(d)}{C} \tag{146}$$

Apply Lemma (D.4) and Lemma (H.1), we have that

$$|\alpha(v_i, w_j, t)| \leq \beta(t), \quad \alpha(v_i, w_{g_i^*}, t) \geq \frac{\eta_G}{3m_G^2}$$

. Moreover, at iteration $t = T_{N,1}$, for all $i \in [m_G], j \in [m_D], j \neq g_i^*$:

$$|\langle v_i^{(t)}, w_j^{(0)} \rangle| \leq C^{-0.95} \langle v_i^{(t)}, w_{g_i^*}^{(0)} \rangle$$

Similarly, we can show that for all $* \neq w_j$, $|\alpha(v_i, *, t)| \leq \beta(t)$ and at iteration $t = T_{N,1}$:

$$|\alpha(v_i, *, t)| \leq \frac{1}{C^{0.95}}\beta(t)$$

Using the fact that $\|w_j^{(t)} - w_j^{(0)}\|_2 \leq \frac{1}{C^{0.94}}$

$$\langle v_i^{(t)}, w_j^{(t)} \rangle = \langle v_i^{(t)}, w_j^{(0)} \rangle + \langle v_i^{(t)}, w_j^{(t)} - w_j^{(0)} \rangle = \langle v_i^{(t)}, w_j^{(0)} \rangle \pm \frac{\beta(t)}{C^{0.93}} \tag{147}$$

Notice that $\langle v_i^{(t)}, w_j^{(t)} \rangle \geq \beta(t) C^{-0.01}$ so we show that at iteration $t = T_{N,1}$:

$$|\langle v_i^{(t)}, w_j^{(t)} \rangle| \leq C^{-0.91} \langle v_i^{(t)}, w_{g_i^*}^{(t)} \rangle$$

Similarly, we can show that for every $\ell \in [2]$,

$$|\langle v_i^{(t)}, w_j^{(t)} \rangle| \leq C^{-0.91} \langle v_i^{(t)}, u_\ell \rangle$$

Stage 2: It remains to prove that for all $t \in [T_{N,1}, T_{N,2}]$, we have that

$$|\langle v_i^{(t)}, w_j^{(t)} \rangle| \leq C^{-0.9} \langle v_i^{(t)}, w_{g_i^*}^{(t)} \rangle$$

The rest of the induction hypothesis follows trivially from Lemma (H.3). (for the relationship between $a^{(t)}$ and $\alpha(w_{i_\ell^*}, u_\ell, t)$ we can use Lemma (D.3)).

To prove this, we know that by the update formula:

$$\langle v_i^{(t+1)}, w_j^{(t+1)} \rangle = \langle v_i^{(t+1)}, w_j^{(t)} \rangle + \langle v_i^{(t+1)}, w_j^{(t+1)} - w_j^{(t)} \rangle \tag{148}$$

$$= \langle v_i^{(t)}, w_j^{(t)} \rangle + \langle v_i^{(t+1)} - v_i^{(t)}, w_j^{(t)} \rangle + \langle v_i^{(t+1)}, w_j^{(t+1)} - w_j^{(t)} \rangle \tag{149}$$

Taking expectation, we have that

$$\mathbb{E}[\langle v_i^{(t+1)}, w_j^{(t+1)}\rangle] = \langle v_i^{(t)}, w_j^{(t)}\rangle + \eta_G \left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \frac{1}{m_G} \frac{\langle w_{g_i^*}^{(t)}, w_j^{(t)}\rangle}{\|w_{g_i^*}^{(t)}\|_2} \pm \eta_G \frac{1}{C^{1.409}} \quad (150)$$

$$+ \sum_{\ell \in [2]} \frac{\eta_D}{2} \langle v_i^{(t+1)}, u_\ell \rangle 1_{j = i_\ell^*} \pm \eta_D \frac{1}{C^{1.501}} \quad (151)$$

and

$$\mathbb{E}[\langle v_i^{(t+1)}, u_\ell\rangle] = \langle v_i^{(t)}, u_\ell\rangle + \eta_G \left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \frac{1}{m_G} \frac{\langle w_{g_i^*}^{(t)}, u_\ell\rangle}{\|w_{g_i^*}^{(t)}\|_2} \pm \eta_G \frac{1}{C^{1.5}} \quad (152)$$

by the induction hypothesis we know that for every $t \le T_{N,2}$, we have that $\langle w_j^{(t)}, w_{j-1}^{(t)}\rangle \le \frac{1}{C^{0.95}}$ and $\|w_j^{(t)}\|_2 = [\Omega(1), \mathrm{polyloglog}\log(d)]$, we know that: when $j = g_i^*$

$$\mathbb{E}[\langle v_i^{(t+1)}, w_j^{(t+1)}\rangle] \ge \langle v_i^{(t)}, w_j^{(t)}\rangle + \eta_G \frac{1}{2m_G\, \mathrm{polyloglog}\log(d)} \quad (153)$$

When $j \ne g_i^*$: using the fact that $\eta_G = \eta_D C^{-0.6}$, we have:

$$\mathbb{E}[|\langle v_i^{(t+1)}, w_j^{(t+1)}\rangle|] \le |\langle v_i^{(t)}, w_j^{(t)}\rangle| + \eta_G \frac{1}{C^{0.9001}} \quad (154)$$

When $i_\ell^* \ne g_i^*$, we have that:

$$\mathbb{E}[|\langle v_i^{(t+1)}, u_\ell\rangle|] \le |\langle v_i^{(t)}, u_\ell\rangle| + \eta_G \frac{1}{C^{0.95}} \quad (155)$$

Thus we complete the proof.

H.5 Proof of the final theorem

To prove the final theorem, notice that by Lemma (H.3), we have that for every $t \in (T_{N,1}, T_1]$, for $i = i_\ell^*$:

$$\mathbb{E}[w_i^{(t+1)}] = w_i^{(t)} + \Theta(\eta_D) u_\ell \pm \eta_D \frac{1}{C^{1.501}} \pm \eta_D \gamma$$

Together with the induction hypothesis, this implies that when $\|w_i^{(t)}\|_2 \ge \log\log\log(d)$, we have that $\langle w_i^{(t)}, u_\ell\rangle \ge (1 - o(1))\|w_i^{(t)}\|_2$. Together with the update formal of $v_j^{(t)}$ we know that when $g_j^* = i_\ell^*$, we have that

$$\mathbb{E}[v_j^{(t+1)}] = v_j^{(t)} + \left(1 \pm \frac{1}{\mathrm{polylog}(d)}\right) \frac{1}{m_G} \eta_G \frac{w_{g_j^*}^{(t)}}{\|w_{g_j^*}^{(t)}\|_2} \pm \eta_G \frac{1}{C^{1.5}} \quad (156)$$

Together with the induction hypothesis, we know that when $\|w_i^{(t)}\|_2 = \mathrm{polyloglog}(d)$, we have that: $\langle v_j^{(t)}, u_\ell\rangle \ge (1 - o(1))\|v_j^{(t)}\|_2$. This proves the theorem.