PLUGIn-CS: A simple algorithm for compressive sensing with generative prior

Babhru Joshi **Xiaowei Li** **Yaniv Plan** **Özgür Yılmaz**
Department of Mathematics
The University of British Columbia
{b.joshi, xli, yaniv, oyilmaz}@math.ubc.ca

Abstract

We consider the problem of recovering an unknown latent code vector under a known generative model from compressive measurements. For a d-layer deep generative network $\mathcal{G} : \mathbb{R}^{n_0} \to \mathbb{R}^{n_d}$ with ReLU activation functions and compressive measurement matrix $\Phi \in \mathbb{R}^{m \times n_d}$, let the observation be $\Phi\mathcal{G}(x) + \epsilon$ where ϵ is noise. We introduce a simple novel algorithm, Partially Linearized Update for Generative Inversion in Compressive Sensing (PLUGIn-CS), to estimate x (and thus $\mathcal{G}(x)$). We prove that, when sensing matrix and weights are Gaussian, if layer widths $n_i \gtrsim 5^i n_0$ and number of measurements $m \gtrsim 2^d n_0$ (both up to log factors), then the algorithm converges geometrically to a (small) neighbourhood of x with high probability. Note the inequality on layer widths allows $n_i > n_{i+1}$ when $i \geq 1$ and thus allows the network to have some contractive layers. After a sufficient number of iterations, the estimation errors for both x and $\mathcal{G}(x)$ are at most in the order of $\sqrt{4^d n_0/m}\|\epsilon\|$. Numerical experiments on synthetic data and real data are provided to validate our theoretical results and to illustrate that the algorithm can effectively recover images from compressive measurements.

1 Introduction

We consider the inverse problem of recovering an unknown structured vector $z^* \in \mathbb{R}^N$ from a noisy compressive observation $y \in \mathbb{R}^m$ of the form

$$y = \Phi z^* + \epsilon, \tag{1}$$

where $\epsilon \in \mathbb{R}^m$ is noise, $\Phi \in \mathbb{R}^{m \times N}$ is the compressive measurement matrix. Traditional approaches for solving (1) often use priors on the signal z^*, for example, a sparsity prior with respect to a fixed basis or dictionary [1, 2, 3]. An emerging viewpoint is to use a generative prior that assumes the signal z^* is in the range of a known deep generative model $\mathcal{G} : \mathbb{R}^{n_0} \to \mathbb{R}^N$, i.e., $z^* = \mathcal{G}(x^*)$ for some $x^* \in \mathbb{R}^{n_0}$. We assume $\mathcal{G}$ has the form

$$\mathcal{G}(x) = \sigma(A_d\sigma(A_{d-1} \ldots \sigma(A_1 x) \ldots)), \tag{2}$$

where $\sigma(\cdot) = \max(\cdot, 0)$ is the ReLU activation function and $A_i \in \mathbb{R}^{n_i \times n_{i-1}}$ is the weight matrix in the i-th layer (and $n_d = N$). One can then develop algorithms that can estimate the latent code vector x^* from $\Phi\mathcal{G}(x^*) + \epsilon$, thus recovering $\mathcal{G}(x^*)$.

Recent advancements in training deep neural networks have shown that generative priors can effectively map low dimensional vectors to the space of natural image classes [4, 5, 6]. Learned generative models can then be used as priors to solve various inverse problems including denoising [7, 8], compressive sensing [9, 10, 11, 12, 13, 14], phase retrieval [15], blind deconvolution [16, 17], low-rank matrix recovery [18] and have been shown to perform on par or outperform classical sparsity based approaches for these inverse problems. For example, in [7] the authors empirically showed that

NeurIPS 2021 Workshop on Deep Learning and Inverse Problems, virtual.

an end-to-end approach for denoising using a neural network that maps noisy patches in an image to noise-free ones achieves state-of-the-art performance and is on par with BM3D. Similarly, in [9] the authors empirically showed that for compressive sensing using generative prior, optimization of the empirical risk objective over the latent code space (of the generative prior) can recover a vector that effectively estimates the uncompressed signal with 5-10 times less measurements compared to Lasso in some cases.

Given that y equals $\Phi\mathcal{G}(x^*)$, with possibly some additive noise, a standard way to estimate x^* would be to look for a minimizer of the program

$$\min_{x\in\mathbb{R}^{n_0}} \ \|y-\Phi\mathcal{G}(x)\|^2. \tag{3}$$

Unfortunately, this program is non-convex and to our knowledge there is no known efficient method that can achieve its global minimum in general. On the other hand, in the case with random weight matrices and sensing matrix, a line of papers showed that gradient-based algorithms can provably avoid local minima with high probability [10, 12, 15]. In particular, [12] considers a model with small noise, Gaussian measurement matrix Φ, and Gaussian weight matrices A_i which are highly expansive at each layer. Under these conditions, the authors show that the latent code vector x^* can be accurately estimated if $m \gtrsim dn_0$ (up to log factors) using a gradient-based method that uses the (sub-)gradient updates given by

$$x^{k+1} = x^k - \eta(D_1A_1)^\intercal(D_2A_2)^\intercal\cdots(D_dA_d)^\intercal\Phi^\intercal\left(\Phi\mathcal{G}(x^k)-y\right), \tag{4}$$

where x^k is the k-th estimate, $\eta\in\mathbb{R}$ is step size, and D_j is a diagonal matrix with entries that are either zero or one. Each D_j zeros out the inactive rows of A_j with respect to the estimate x^k and so it is a function of x^k (and A_p for $p<j$). Thus, at each iteration all D_j need to be updated.

In this paper, we show that one can drop all D_j and still recover an accurate estimate of x^*. This result follows from our previous work in the denoising case ($\Phi = I_{n_d}$) [19], where we introduced a novel algorithm called Partially Linearized Update for Generative Inversion (PLUGIn). For the compressive sensing case, we propose the following iterative algorithm to estimate x^*:

$$x^{k+1} = x^k - \eta A_1^\intercal A_2^\intercal\cdots A_d^\intercal\Phi^\intercal\left(\Phi\mathcal{G}(x^k)-y\right) \tag{PLUGIn-CS}$$

This algorithm was inspired by previous work showing that latent vectors for non-linear single-index function can be approximately estimated by treating the function as linear [20, 21]. Similar to [8, 9, 10, 12, 15, 17, 22], for theoretical analysis, we assume the weight matrices are Gaussian. To show that the algorithm works more broadly, we conduct real data simulations. Applying the ideas in [20, 21] one can show that for any fixed x^0, the first iteration of PLUGIn-CS provides an unbiased estimate of x^* with $\eta = 2^d$, which is generally not the case for the gradient descent estimates given by (4). Additionally, each iteration of PLUGIn-CS maps the difference $\Phi\mathcal{G}(x^k)-y$ to the low dimensional latent code space using a *static* matrix $A_1^\intercal A_2^\intercal\ldots A_d^\intercal\Phi^\intercal$, which can be pre-multiplied and reused in subsequent iterations.

Building upon the theory for PLUGIn [19], we show that the estimates provided by PLUGIn-CS converge geometrically to a neighbourhood of x^* (and also $\mathcal{G}(x^k)$ to a neighbourhood of $\mathcal{G}(x^*)$) with high probability. This result holds with the following assumptions:

A1. Each $A_i\in\mathbb{R}^{n_i\times n_{i-1}}$ has i.i.d. $\mathcal{N}(0,1/n_i)$ entries and $\{A_i\}_{i\le d}$ are independent.

A2. Layer widths (number of nodes in each layer) satisfy

$$n_i \ge C_0 5^i n_0 \log\left(\prod_{j=0}^{i-1}\frac{en_j}{n_0}\right), \quad 1\le i\le d \tag{5}$$

for some (sufficiently large) absolute constant C_0.

A3. The measurement matrix $\Phi\in\mathbb{R}^{m\times n_d}$ has i.i.d. $\mathcal{N}(0,1/m)$ entries (independent from weight matrices) with

$$m \ge c_0 2^d n_0 \log\left(\prod_{j=0}^{d}\frac{en_j}{n_0}\right) \tag{6}$$

for some absolute constant c_0.

A4. The noise ϵ does not depend on $\{A_i\}_{i \le d}$ or Φ. (The noise may be deterministic or random.)

Note that A2 allows $n_i > n_{i+1}$ for $i \ge 1$ and thus can provide theoretical guarantees even when $\mathcal{G}$ has some contractive layers. Under these assumptions, PLUGIn-CS algorithm converges to a neighbourhood of x^* for a range of step sizes near 2^d. Precisely, we have the following theorem.

Theorem 1. *Let $\theta \in (0, \frac{4}{3})$ and let $\alpha = |1-\theta| + \frac{1}{2}\theta \in (0,1)$. Let R be a positive number such that $\|x^0 - x^*\| \le R$. Under assumptions A1-A4, the k-th estimate x^k given by PLUGIn-CS with constant step size $\eta = \theta 2^d$ satisfies*

$$\|x^k - x^*\| \le \alpha^k R + \frac{15\theta}{1-\alpha} 2^d \sqrt{n_0/m}\|\epsilon\|, \text{ and}$$
$$\|\mathcal{G}(x^k) - \mathcal{G}(x^*)\| \le 3\alpha^k R + \frac{45\theta}{1-\alpha} 2^d \sqrt{n_0/m}\|\epsilon\|$$

with probability at least $1 - 2(k+4)e^{-10n_0}$.

When $\theta = 1$, Theorem 1 reduces to the following corollary.

Corollary 1. *Let R be a positive number such that $\|x^0 - x^*\| \le R$. Under assumptions A1-A4, the k-th estimate x^k given by PLUGIn-CS with constant step size $\eta = 2^d$ satisfies*

$$\|x^k - x^*\| \le 2^{-k} R + 30 \cdot 2^d \sqrt{n_0/m}\|\epsilon\|, \text{ and}$$
$$\|\mathcal{G}(x^k) - \mathcal{G}(x^*)\| \le 2^{-k}(3R) + 90 \cdot 2^d \sqrt{n_0/m}\|\epsilon\|$$

with probability at least $1 - 2(k+4)e^{-10n_0}$.

Remark 1 (Contractive layers)**.** In A2, (5) states a lower bound on n_i with respect to the latent code dimension n_0 (up to log factors). While this bound strictly increases with layer depth i, it is not necessary for n_i to always increase with i (except in the first layer). For example, consider $n_i = \beta C_0 5^d n_0 d(2d - i)$ where β is any fixed number such that $\beta C_0 \in \mathbb{N}$ and $\beta \ge 4 + \log C_0$. It is easy to see $n_1 > n_2 > \cdots > n_d$, and we can also verify (see Appendix E) that such n_i satisfy (5). In this case, the network is contractive in each layer after the first, and Theorem 1 still applies.

Remark 2 (Initialization may depend on random weight matrices and sensing matrix)**.** The results of the theorem can still hold when x^0 is chosen randomly, dependent on the weight matrices A_i. In this case, suppose that $\|x^0 - x^*\| \le R$ with probability at least $1-\delta$. Then, the error bounds hold with probability at least $1 - 2(k+4)e^{-10n_0} - \delta$. This does not follow directly from the theorem as stated (which fixes x^0, then takes random weight matrices), but follows from the proof.

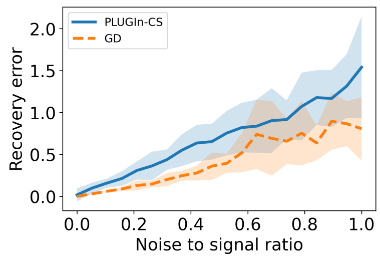
Remark 3 (Comparison to guarantees for gradient-based method)**.** Here we compare our results to the ones in [12], which uses (4) for iterations and considers a model with small noise, i.e., $\|\epsilon\| \lesssim \frac{\|x^*\|_2}{d^{42} 2^{d/2}}$. They show that when the weight matrices and sensing matrix are Gaussian with weight matrices sufficiently expansive at each layer, the iterates of the gradient-based method converge to a neighborhood of the target signal x^*. After sufficiently many iterations N, the iterates converge geometrically to a neighborhood of x^* of radius at most on the order of $2^{d/2}\|\epsilon\|$. This rate of convergence takes the form $(1 - C/2^d)$, thus giving slower convergence for deeper nets. On the other hand, we note that dependence on d is of relatively minor concern. Generative models usually have small depth in practice, our MNIST experiments (below) work well with depth 3, and typical applications use depth less than 8.

In comparison, Theorem 1 holds for any noise ϵ that does not depend on $\{A_i\}_{i \le d}$ or Φ. Under similar randomness assumptions, the iterates of PLUGIn-CS converge to a neighborhood of the latent code x^* of radius at most on the order of $2^d \sqrt{n_0/m}\|\epsilon\|$. This result can hold for networks with contractive layers and the rate of convergence is geometric starting at the initial iterate of PLUGIn-CS.

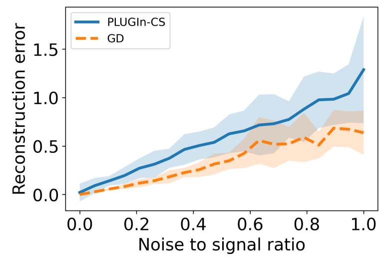
2 Numerical Experiments

In this section, we provide numerical experiments on synthetic data and MNIST images where the oberservations follow the model in (1). All experiments were conducted using Google Colaboratory.

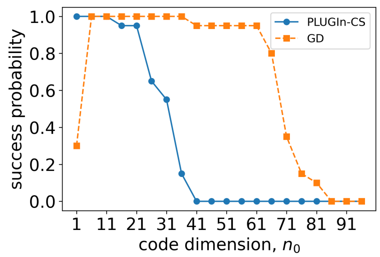
In the synthetic experiments, we let the generative prior be a 2-layer neural network $\mathcal{G}(z) = \sigma(A_2\sigma(A_1 x)))$, where the entries of weight matrix $A_i \in \mathbb{R}^{n_i \times n_{i-1}}$ are sampled from $\mathcal{N}(0, 1/n_i)$.



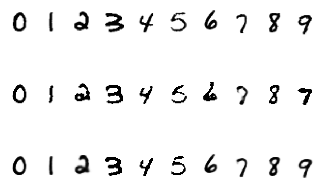
(a) Recovery error



(b) Reconstruction error



(c) Empirical success probability



(d) MNIST images

Figure 1: Comparison of performance of PLUGIn-CS with gradient descent (GD) is shown. Panels (a) and (b) show the dependence of relative recovery error with noise level-to-signal level from 20 independent trials. Panel (c) shows the empirical success probability versus the code dimension n_0 for noiseless problems. Panel (d) shows the result of recovering an image from compressive measurements. The top row corresponds to original image. The second and third row are images recovered using PLUGIn-CS and gradient descent, respectively.

We sample the target latent code x^* uniformly from $\mathbb{S}^{n_0-1}$, set the noise level as $\alpha \in \mathbb{R}$, and set the noise to be $\alpha\nu$ where ν is sampled uniformly from $\mathbb{S}^{n_3-1}$. Then we set $y = \Phi\mathcal{G}(x^*) + \alpha\nu$, where the entries of the compressive measurement matrix $\Phi \in \mathbb{R}^{m\times n_2}$ are sampled from $\mathcal{N}(0, 1/m)$. We run PLUGIn-CS and gradient descent each for 10,000 iterations or until the relative successive error is less than 10^{-13}, and set $\hat{x}$ to be the output. We use a fixed step size of 3 and 10 for PLUGIn-CS and gradient descent, respectively, with the gradient computed using PyTorch [23].

For the first experiment, we fix $n_0 = 10$, $n_1 = 400$, $n_2 = 300$, $m = 150$, and sample the noise level α uniformly in the interval $[0, 1]$. In figures 1a and 1b, the solid line corresponds to the performance of PLUGIn-CS and the dotted line represents the performance of gradient descent. Figure 1a shows the empirical dependence of the the relative recovery error $\|\hat{x} - x^*\|/\|x^*\|$ on the noise-to-signal ratio given, given by α, from 20 independent trials. Similarly, figure 1b shows the empirical dependence of the the relative reconstruction error $\|\mathcal{G}(\hat{x}) - \mathcal{G}(x^*)\|/\|\mathcal{G}(x^*)\|$ from 20 independent trials. The figures show that PLUGIn-CS can stably solve the compressive sensing problem (1) with a generative prior. For the second experiment, we fix $\alpha = 0$, $n_1 = 250$, $n_2 = 700$, $m = 150$, and sample the latent code dimension n_0 in the interval $[1, 100]$. In figures 1c, the solid line corresponds to the performance of PLUGIn-CS and the dotted line represents the performance of gradient descent. Figure 1c shows the empirical success probability from 20 independent trials.

We now empirically show that PLUGIn-CS can effectively recover MNIST images from compressive measurements and compare its performance to gradient descent. We trained a VAE [24] using Adam optimizer [25] with a learning rate of 0.001 and mini-batch size 100 on the MNIST dataset [26]. The decoder network in the VAE is a fully connected network with parameters $20 - 500 - 500 - 784$. The compressive sensing matrix $\Phi \in \mathbb{R}^{m\times 784}$ follows i.i.d $\mathcal{N}(0, 1/m)$ entries with $m = 150$ and the observation y satisfies $y = \Phi z^*$, where z^* is an image from the MNIST database. In all MNIST experiments, we use a fixed step size of $\eta = 1/\gamma$ for PLUGIn-CS, where γ is the product of the operator norms of the weight matrices; for gradient descent, we use a fixed step size of 1000. Similar to the synthetic experiment, we run PLUGIn-CS and gradient descent each for 10,000 iterations or until the relative successive error is less than 10^{-13}. In figure 1d, the images in the top row are the observations, the images in the second row and third row are the recovered images corresponding to PLUGIn-CS and gradient descent, respectively.

Acknowledgments and Disclosure of Funding

Y. Plan is partially supported by an NSERC Discovery Grant (GR009284), an NSERC Discovery Accelerator Supplement (GR007657), and a Tier II Canada Research Chair in Data Science (GR009243). O. Yılmaz is partially supported by an NSERC Discovery Grant (22R82411) and Pacific Institute for the Mathematical Sciences (PIMS) CRG 33: High-Dimensional Data Analysis. B. Joshi is partially supported by the Pacific Institute for the Mathematical Sciences (PIMS). (The research and findings may not reflect those of the Institute.)

References

[1] D.L. Donoho. De-noising by soft-thresholding. *IEEE Transactions on Information Theory*, 41(3):613–627, 1995.

[2] M. Aharon, M. Elad, and A. Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4322, 2006.

[3] Michael Elad and Michal Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15(12):3736–3745, 2006.

[4] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018.

[5] Hoo-Chang Shin, Neil A. Tenenholtz, Jameson K. Rogers, Christopher G. Schwarz, Matthew L. Senjem, Jeffrey L. Gunter, Katherine P. Andriole, and Mark Michalski. Medical image synthesis for data augmentation and anonymization using generative adversarial networks. In Ali Gooya, Orcun Goksel, Ipek Oguz, and Ninon Burgos, editors, *Simulation and Synthesis in Medical Imaging*, pages 1–11, Cham, 2018. Springer International Publishing.

[6] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4396–4405, 2019.

[7] Harold C. Burger, Christian J. Schuler, and Stefan Harmeling. Image denoising: Can plain neural networks compete with bm3d? In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2392–2399, 2012.

[8] Reinhard Heckel, Wen Huang, Paul Hand, and Vladislav Voroninski. Rate-optimal denoising with deep neural networks. *Information and Inference: A Journal of the IMA*, 06 2020. iaaa011.

[9] Ashish Bora, Ajil Jalal, Eric Price, and Alexandros G Dimakis. Compressed sensing using generative models. In *International Conference on Machine Learning*, pages 537–546. PMLR, 2017.

[10] Paul Hand and Vladislav Voroninski. Global guarantees for enforcing deep generative priors by empirical risk. *CoRR*, abs/1705.07576, 2017.

[11] Viraj Shah and Chinmay Hegde. Solving linear inverse problems using gan priors: An algorithm with provable guarantees. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4609–4613, 2018.

[12] Wen Huang, Paul Hand, Reinhard Heckel, and V. Voroninski. A provably convergent scheme for compressive sensing under random generative priors. *Journal of Fourier Analysis and Applications*, 27:1–34, 2018.

[13] Ganlin Song, Zhou Fan, and John Lafferty. Surfing: Iterative optimization over incrementally trained deep networks. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 15008–15017, 2019.

[14] Fabian Latorre, Armin eftekhari, and Volkan Cevher. Fast and provable admm for learning with generative priors. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[15] Paul Hand, Oscar Leong, and Vladislav Voroninski. Phase retrieval under a generative prior. *CoRR*, abs/1807.04261, 2018.

[16] Muhammad Asim, Fahad Shamshad, and Ali Ahmed. Solving bilinear inverse problems using deep generative priors. *CoRR*, abs/1802.04073, 2018.

[17] Paul Hand and Babhru Joshi. Global guarantees for blind demodulation with generative priors. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[18] Jorio Cocola, Paul Hand, and Vladislav Voroninski. No statistical-computational gap in spiked matrix models with generative network priors. *Entropy*, 23(1), 2021.

[19] Babhru Joshi, Xiaowei Li, Yaniv Plan, and Ozgur Yilmaz. PLUGIn: A simple algorithm for inverting generative models with recovery guarantees. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021.

[20] David R. Brillinger. *A Generalized Linear Model With "Gaussian" Regressor Variables*, pages 589–606. Springer New York, New York, NY, 2012.

[21] Yaniv Plan and Roman Vershynin. The generalized lasso with non-linear observations. *IEEE Transactions on Information Theory*, 62(3):1528–1537, 2016.

[22] Constantinos Daskalakis, Dhruv Rohatgi, and Manolis Zampetakis. Constant-expansion suffices for compressed sensing with generative priors. *CoRR*, abs/2006.04237, 2020.

[23] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

[24] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[25] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[26] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[27] Sjoerd Dirksen. Tail bounds via generic chaining. *Electron. J. Probab.*, 20(53):1–29, 2015.

[28] Jiri Matousek. *Lectures on discrete geometry*, volume 212. Springer Science & Business Media, 2013.

[29] Simon Foucart and Holger Rauhut. *A Mathematical Introduction to Compressive Sensing*. Birkhäuser, New York, NY, 2013.

[30] Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2018.

[31] Michel Talagrand. *The generic chaining: upper and lower bounds of stochastic processes*. Springer Science & Business Media, 2006.

Notations in proofs

For a positive integer n, let $[n] = \{1, 2, \dots, n\}$. For a vector x, let $\|x\|$ be its Euclidean norm; for a matrix A, let $\|A\|$ be its operator norm; for a matrix A and a set $\mathcal{T}$, let $\|A\|_{\mathcal{T}} := \sup_{x\in\mathcal{T}\setminus\{0\}} \frac{\|Ax\|}{\|x\|}$. Let $\mathbb{B}(x, r)$ be the Euclidean ball of radius r centered at x and let $\mathbb{B}^n(0, r)$ be the Euclidean ball in $\mathbb{R}^n$ with radius r, centered at origin. We use C and c to denote absolute constants (often c for small ones and C for large ones) which may vary from line to line. We also use c_0, C_0, C_1, etc., to denote particular absolute constants, which do not change throughout the paper.

We use $\mathbb{P}_{A_i}$ to denote that the probability is taken only with respect to A_i. In neural network $\mathcal{G} : \mathbb{R}^{n_0} \to \mathbb{R}^{n_d}$, let $\mathcal{G}_i : \mathbb{R}^{n_0} \to \mathbb{R}^{n_i}$ be the mapping that corresponds to the first i layers, i.e. $\mathcal{G}_i(x) = \sigma(A_i \dots \sigma(A_1 x)\dots)$. For its weight matrices, let $\tilde{A}_0 = I_{n_0}$ and $\tilde{A}_i = A_i A_{i-1} \cdots A_1$ for $i \in [d]$. For $x \in \mathbb{R}^{n_0}$, denote $x_0 = x$ and $x_i = \mathcal{G}_i(x)$ for $i \in [d]$

A Proof Outline

Our proofs for PLUGIn-CS builds upon the proofs for PLUGIn [19]. Here we include all the proofs for completeness, and note that many parts of these proofs are the same as in [19]. The main differences are the parts dealing with sensing matrix Φ. In particular, we added Lemma 9 and modified Lemma 5, Lemma 6 as well as *Proof of Theorem 1* to incorporate Φ in the new proofs.

Below we give a sketch for the proof of Theorem 1. For simplicity, we will only focus on analyzing one iteration of PLUGIn-CS with step size $\eta = 2^d$. The complete proof can be found in Appendix D.

A Special Case

Let us first look at the special case where $d = 1$, $\epsilon = 0$ and $\Phi = I$. The analysis here highlights some of the key ideas in our proofs, while its result Lemma 1 serves as a building block for proof in the general case. In this special case, PLUGIn-CS with $\eta = 2^d$ reduces to

$$x^{k+1} = x^k - 2A^\intercal \left[\sigma(Ax^k) - \sigma(Ax^*)\right]$$

where $\sigma = \mathrm{ReLU}$ and $A \in \mathbb{R}^{m\times n}$ is random with i.i.d. $\mathcal{N}\left(0, \frac{1}{m}\right)$ entries.

In fact, the first iterate provides an unbiased estimate of x^* when x^0 does not depend on A. Indeed, the rotation invariance property of the Gaussian distribution may be leveraged to show [20, 21], for any fixed x,

$$\mathbb{E}A^\intercal\sigma(Ax) = \tfrac{1}{2}x. \tag{7}$$

For completeness, we also include a proof for (7) in Appendix B, Lemma 2. Applying (7) to the first iteration gives

$$\begin{aligned}\mathbb{E}x^1 &= x^0 - 2\mathbb{E}A^\intercal\sigma(Ax^0) + 2\mathbb{E}A^\intercal\sigma(Ax^*)\\ &= x^0 - x^0 + x^* = x^*\end{aligned}$$

Thus, even the first iterate can be shown to be a good estimate by showing that x^1 concentrates around its mean. Further iterates are generally no longer unbiased estimators because they pick up complex dependence on the random matrix A. We overcome this by developing a series of uniform deviation inequalities, as below.

Let us suppose we have shown that, with high probability, $\|x^k - x^*\| \le r$ for some (small) constant $r > 0$. Then we wish to show that $\|x^{k+1} - x^*\| \le r/2$ with high probability. Notice that

$$\begin{aligned}-(x^{k+1} - x^*) &= 2A^\intercal \left[\sigma(Ax^k) - \sigma(Ax^*)\right] - (x^k - x^*)\\ \|x^{k+1} - x^*\| &= \sup_{u\in\mathbb{S}^{n-1}} 2\left\langle Au, \sigma(Ax^k) - \sigma(Ax^*)\right\rangle - \left\langle u, x^k - x^*\right\rangle\\ &= 2\sup_{u\in\mathbb{S}^{n-1}} Z(u, x^k; x^*)\end{aligned}$$

where

$$Z(u, v; x^*) := \langle Au, \sigma(Av) - \sigma(Ax^*)\rangle - \tfrac{1}{2}\langle u, v - x^*\rangle\,.$$

We wish to bound the supremum of random process $Z(u, x^k; x^*)$ over $u \in \mathbb{S}^{n-1}$. However, this process is challenging to analyze since x^k depends on A when $k \geq 1$. To alleviate this dependency, we bound by the supremum of $Z(u, v; x^*)$ over $(u, v) \in \mathcal{T}^0 := \mathbb{B}^n(0, 1) \times \mathbb{B}(x^*, r)$ instead. It is worth noting that $Z(u, v; x^*)$ is centred, namely $\mathbb{E}Z(u, v; x^*) = 0$ for any fixed (u, v). We now arrive at the estimate

$$\|x^{k+1} - x^*\| \leq 2 \sup_{(u,v)\in\mathcal{T}^0} Z(u, v; x^*) \quad \text{if } \|x^k - x^*\| \leq r. \tag{8}$$

The following Lemma 1 provides a bound on $\sup_{\mathcal{T}^0} Z(u, v; x^*)$. In fact, it is slightly more general because we replaced $\mathcal{T}^0$ with $\mathcal{T}_1 \times \mathcal{T}_2$ (this replacement is helpful when studying the general case $d > 1$). The complete proof of this lemma can be found in Appendix C. The proof idea is to first establish that $Z(u, v; x^*)$ has mixed (sub-Gaussian and sub-exponential) tail increments through Bernstein's inequality, and then apply the result from [27], which provides a general bound for the supremum of random processes with mixed tail increments.

Lemma 1. *Let $\sigma = \mathrm{ReLU}$. Fix $w \in \mathbb{R}^n$ and let $A \in \mathbb{R}^{m\times n}$ have i.i.d. $\mathcal{N}\left(0, \frac{1}{m}\right)$ entries. Define*

$$Z(u, v; w) := \langle Au, \sigma(Av) - \sigma(Aw)\rangle - \tfrac{1}{2}\langle u, v - w\rangle .$$

Suppose $\mathcal{T}_1, \mathcal{T}_2$ are sets (not depending on A) such that

$$\mathcal{T}_1 = \mathcal{S}_1 \cap \mathbb{B}^n(0, \alpha) \quad \text{and} \quad \mathcal{T}_2 = \mathcal{S}_2 \cap \mathbb{B}(w, \alpha r)$$

for some q-dimensional (affine) subspaces $\mathcal{S}_1, \mathcal{S}_2 \subseteq \mathbb{R}^n$ and real numbers $\alpha, r > 0$. Then for any $t \geq 1$,

$$\sup_{\substack{u\in\mathcal{T}_1\\ v\in\mathcal{T}_2}} |Z(u, v; w)| \leq C_1\alpha^2 r \left(\sqrt{\frac{q}{m}} + \frac{q}{m} + \sqrt{\frac{t}{m}} + \frac{t}{m}\right)$$

with probability at least $1 - e^{-t}$. Here $C_1 > 0$ is an absolute constant.

We can apply Lemma 1 to estimate (8) (with $\mathcal{S}_1 = \mathcal{S}_2 = \mathbb{R}^n$) and get, for example,

$$\|x^{k+1} - x^*\| \leq 2C_1 r \left(\sqrt{\tfrac{n}{m}} + \tfrac{n}{m} + \sqrt{\tfrac{n}{m}} + \tfrac{n}{m}\right) \leq \tfrac{1}{2} r$$

with probability at least $1 - e^{-n}$, provided that $m \geq (16C_1)^2 n$.

The General Case

Let us illustrate the proof idea with $d = 2$ (the extension to $d > 2$ is straightforward). Denote $x_i^k = \mathcal{G}_i(x^k)$ and $x_i^* = \mathcal{G}_i(x^*)$ for $i = 1, 2$. By adding and subtracting $2A_1^\intercal(x_1^k - x_1^*)$ and $2^2 A_1^\intercal A_2^\intercal(x_2^k - x_2^*)$, we can write PLUGIn-CS with $\eta = 2^d$ as

$$\begin{aligned} x^{k+1} - x^* &= x^k - x^* - 2^2 A_1^\intercal A_2^\intercal \Phi^\intercal[\Phi\mathcal{G}(x^k) - \Phi\mathcal{G}(x^*) - \epsilon] \\ &= (x^k - x^*) - 2A_1^\intercal\left(\sigma(A_1x^k) - \sigma(A_1x^*)\right) \\ &\quad + 2A_1^\intercal\left[(x_1^k - x_1^*) - 2A_2^\intercal\left(\sigma(A_2x_1^k) - \sigma(A_2x_1^*)\right)\right] \\ &\quad + 2^2 A_1^\intercal A_2^\intercal\left(I - \Phi^\intercal\Phi\right)\left(x_2^k - x_2^*\right) \\ &\quad + 2^2 A_1^\intercal A_2^\intercal \Phi^\intercal \epsilon. \end{aligned}$$

Similar to the special case above, we can get

$$\begin{aligned} \|x^{k+1} - x^*\| \leq &\sup_{u\in\mathbb{S}^{n_0-1}} 2Z_1(u, x^k) + \sup_{u\in\mathbb{S}^{n_0-1}} 2^2 Z_2(A_1u, x_1^k) \\ &+ 2^2\|A_2A_1\|\|(I - \Phi^\intercal\Phi)(x_2^k - x_2^*)\| + 2^2\|A_1^\intercal A_2^\intercal\Phi^\intercal\epsilon\| \end{aligned} \tag{9}$$

where (denote $x_0^* = x^*$)

$$Z_j(u, v) := \left\langle A_ju, \sigma(A_jv) - \sigma(A_jx_{j-1}^*)\right\rangle - \tfrac{1}{2}\left\langle u, v - x_{j-1}^*\right\rangle, \quad j = 1, 2.$$

Also assume that $\|x^k - x^*\| \leq r$, it remains to bound each term on the right hand side of (9). The first term can be bounded directly through Lemma 1 (with $t = 10n_0$). The last term is also easy to bound by the randomness of A_i (Appendix D, Lemma 6), in which case we have $\|A_1^\intercal A_2^\intercal \epsilon\| \le 15\sqrt{n_0/m}\|\epsilon\|$ with high probability.

For the second term, first notice that $\mathrm{range}(A_1)$ is a n_0-dimensional subspace in $\mathbb{R}^{n_1}$. Using the ideas from [9, 28], we can also show that $\mathrm{range}(\mathcal{G}_1)$ is contained in a union of N many n_0-dimensional (affine) subspaces, where $N \le (en_1/n_0)^{n_0}$. Furthermore, let $\mathcal{E}$ be the event such that mappings $A_1, A_2A_1, \mathcal{G}_1, \mathcal{G}$ all have Lipschitz constants being at most 3, then we can show (Appendix D, Lemma 8) that $\mathbb{P}(\mathcal{E}) \ge 1 - 3e^{-10n_0}$. Also on event $\mathcal{E}$ (note that $\|A_1\| \le 3$ and $\|x_1^k - x_1^*\| \le 3r$), we have

$$A_1\mathbb{S}^{n_0-1} \subseteq \mathrm{range}(A_1) \cap \mathbb{B}^{n_1}(0,3) = \mathcal{S}_1 \cap \mathbb{B}^{n_1}(0,3) =: \mathcal{T}_1$$
$$x_1^k \in \mathrm{range}(\mathcal{G}_1) \cap \mathbb{B}(x_1^*, 3r) \subseteq \cup_{j\in[N]} \left(\mathcal{S}_{1,j} \cap \mathbb{B}(x_1^*, 3r)\right) =: \cup_{j\in[N]}\mathcal{T}_{2,j}$$

where $\mathcal{S}_1$ and $\mathcal{S}_{2,j}$ are n_0-dimensional (affine) subspaces. Applying Lemma 1 on each $\mathcal{T}_1 \times \mathcal{T}_{2,j}$, followed by a union bound over $j \in [N]$, we get (denote $\mathcal{T}_2 = \cup_{j\in[N]}\mathcal{T}_{2,j}$)

$$\sup_{\mathcal{T}_1\times\mathcal{T}_2} Z_2(u,v) \le C_1(9r)\left(\sqrt{\tfrac{n_0}{n_2}} + \tfrac{n_0}{n_2} + \sqrt{\tfrac{t}{n_2}} + \tfrac{t}{n_2}\right)$$

with probability (over A_2 and conditioning on A_1) at least $1 - Ne^{-t}$. By choosing $t = 2n_0\log(en_1/n_0)$, we obtain a high probability bound for $\sup_{u\in\mathbb{S}^{n_0-1}} Z_2(A_1u, x_1^k)$.

For the third term, use the fact that $\|A_2A_1\| \le 3$ and $\|x_2^k - x_2^*\| \le 3r$ on $\mathcal{E}$, together with Lemma 9 we can obtain a high probability bound for $\|A_2A_1\|\|(I - \Phi^\intercal\Phi)(x_2^k - x_2^*)\|$.

Finally, if C_0 and c_0 are sufficiently large, we can thus show from (9) that, with high probability,

$$\|x^{k+1} - x^*\| \le \frac{1}{2}\left(r + 30\cdot 2^2\sqrt{n_0/m}\|\epsilon\|\right).$$

B Some Results on Gaussian Matrices

Here we state some results on Gaussian Matrices, which will be used in the proofs later.

Lemma 2 ([20, 21])**.** *Let $\sigma : \mathbb{R} \to \mathbb{R}$ be a positively homogeneous activation function. Let $A \in \mathbb{R}^{m\times n}$ have i.i.d. $\mathcal{N}\left(0, \frac{1}{m}\right)$ entries. Then for any $x \in \mathbb{R}^n$,*

$$\mathbb{E}A^\intercal\sigma(Ax) = \lambda x,$$

where $\lambda := \mathbb{E}\, g\cdot\sigma(g)$ with $g \sim \mathcal{N}(0,1)$. In particular, $\lambda = \frac{1}{2}$ when σ is ReLU.

Proof. Since σ is positively homogeneous, we can assume (without loss of generality) $x \in \mathbb{S}^{n-1}$. Denote by $a_j^\intercal$ the j-th row of A. Then

$$\mathbb{E}A^\intercal\sigma(Ax) = \mathbb{E}\sum_{j=1}^m \sigma(a_j^\intercal x)\, a_j = m\,\mathbb{E}\sigma(a_1^\intercal x)\, a_1 = \mathbb{E}\sigma(a^\intercal x)\, a$$

where $a := \sqrt{m}a_1 \sim \mathcal{N}(0, I_n)$. Take an orthogonal matrix U such that $Ux = \|x\|e_1 = e_1$ where $e_1 = (1, 0, \ldots, 0)^\intercal$. Note that by rotation invariance for standard Gaussian, Ua and a have the same distribution $\mathcal{N}(0, I_n)$, thus

$$\mathbb{E}\sigma(a^\intercal x)\, a = \mathbb{E}\sigma(a^\intercal U^\intercal e_1)\, U^\intercal U a = \mathbb{E}\sigma(a^\intercal e_1) U^\intercal a = U^\intercal\mathbb{E}\sigma(a^\intercal e_1)a = \lambda U^\intercal e_1 = \lambda x.$$

□

The following theorem is the concentration of (Gaussian) measure inequality for Lipschitz functions. Here we only state a one-sided version, though it is more commonly stated with a two-sided one, i.e., $\mathbb{P}\left(|f(g) - \mathbb{E}f(g)| \ge t\right) \le 2\exp\left(-t^2/(2L_f^2)\right)$.

Theorem 2. *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a Lipschitz function with Lipschitz constant L_f. Let $g \in \mathbb{R}^n$ be a random vector with independent $\mathcal{N}(0,1)$ entries. Then, for all $t > 0$,*

$$\mathbb{P}\left(f(g) - \mathbb{E}f(g) \ge t\right) \le \exp\left(-\frac{t^2}{2L_f^2}\right).$$

A proof of Theorem 2 can be found in [29, Chap. 8]. Based on this theorem, it is easy to prove the following results.

Lemma 3. *Let $A \in \mathbb{R}^{m\times n}$ have i.i.d. $\mathcal{N}(0,1)$ entries.*

(a) For any fixed point $s \in \mathbb{R}^n$, we have

$$\mathbb{P}\left(\|As\| \geq \sqrt{m}\|s\| + \sqrt{t}\|s\|\right) \leq e^{-t/2}, \quad \forall t > 0.$$

(b) For any fixed k-dimensional subspace $\mathcal{S} \subseteq \mathbb{R}^n$, we have

$$\mathbb{P}\left(\|A\|_{\mathcal{S}} \geq \sqrt{m} + \sqrt{k} + \sqrt{t}\right) \leq e^{-t/2}, \quad \forall t > 0$$

and

$$\mathbb{P}\left(\left|\|A\|_{\mathcal{S}} - \sqrt{m}\right| \geq \sqrt{k} + \sqrt{t}\right) \leq 2e^{-t/2}, \quad \forall t > 0.$$

Proof. (a) Without loss of generality, assume $\|s\| = 1$. Then $As \sim \mathcal{N}(0, I_m)$ and by Jensen's inequality, $\mathbb{E}\|As\| \leq \sqrt{\mathbb{E}\|As\|^2} = \sqrt{m}$. The result follows immediately from Theorem 2 (with $f(g) = \|g\|$ and $g = As$).

(b) Let U be an orthogonal matrix such that $U^\intercal \mathcal{S} = \operatorname{span}\{e_1, \ldots, e_k\} =: \mathcal{S}_0$, then $\|A\|_{\mathcal{S}} = \|AU\|_{\mathcal{S}_0}$. Also, since AU has the same distribution as A (by rotation invariance), we get

$$\mathbb{P}\left(\|A\|_{\mathcal{S}} \geq \sqrt{m} + \sqrt{k} + \sqrt{t}\right) = \mathbb{P}\left(\|A\|_{\mathcal{S}_0} \geq \sqrt{m} + \sqrt{k} + \sqrt{t}\right).$$

Notice that $\|A\|_{\mathcal{S}_0}$ is the operator norm for a particular sub-matrix (obtained by taking first k-columns) of A, so without loss of generality, we can assume $k = n$.

Let $f(A) = \|A\|$. Since $|f(A) - f(A')| \leq \|A - A'\|_F$, f is 1-Lipschitz when viewed as a mapping from $\mathbb{R}^{mn}$ to $\mathbb{R}$. By Theorem 2,

$$\mathbb{P}\left(f(A) \geq \mathbb{E}f(A) + \sqrt{t}\right) \leq e^{-t/2}, \quad \forall t > 0.$$

The one-sided result follows since $\sqrt{m} - \sqrt{n} \leq \mathbb{E}\|A\| \leq \sqrt{m} + \sqrt{n}$ (see, e.g., [30, Section 7.3]). The two-sided result follows by also considering $f(A) = -\|A\|$. $\square$

C Preliminaries and Proof for Lemma 1

Preliminaries

For $\alpha \geq 1$, the *ψ_α-norm* of a random variable X is defined as

$$\|X\|_{\psi_\alpha} := \inf\{t > 0 : \mathbb{E}\exp(|X|^\alpha / t^\alpha) \leq 2\}.$$

We say X is *sub-Gaussian* if $\|X\|_{\psi_2} < \infty$ and *sub-exponential* if $\|X\|_{\psi_1} < \infty$. The ψ_2 and ψ_1 norms are also called sub-Gaussian and sub-exponential norms respectively. Loosely speaking, a sub-Gaussian (or a sub-exponential) random variable has tail dominated by the tail of a Gaussian (or an exponential) random variable.

For independent, mean zero, sub-exponential random variables $X_1, \ldots, X_m$, their sum concentrates around zero. In particular, the following *Bernstein's Inequality* [30, Section 2.8] holds:

$$\mathbb{P}\left(\left|\sum_{i=1}^m X_i\right| \geq t\right) \leq 2\exp\left[-c\min\left(\frac{t^2}{\sum_{i=1}^m \|X_i\|_{\psi_1}^2}, \frac{t}{\max_i \|X_i\|_{\psi_1}}\right)\right].$$

The above inequality also suggests that $\sum_{i=1}^m X_i$ has a mixed tail, i.e., a tail consisting of both a sub-Gaussian part and a sub-exponential part. In our proof, we will use the following result from generic chaining for mixed tail processes.

Theorem 3 (Theorem 3.5 [27]). *If $(X_t)_{t\in T}$ has a mixed tail with respect to metric pair (d_1, d_2), i.e.*

$$\mathbb{P}\left(|X_t - X_s| \geq \sqrt{u} d_2(t,s) + u d_1(t,s)\right) \leq 2e^{-u}, \quad \forall u \geq 0.$$

Then there are constants $c, C > 0$ such that for any $u \geq 1$,

$$\mathbb{P}\left(\sup_{t\in T} |X_t - X_{t_0}| \geq C(\gamma_2(T,d_2) + \gamma_1(T,d_1)) + c(\sqrt{u}\Delta_{d_2}(T) + u\Delta_{d_1}(T))\right) \leq e^{-u}.$$

Here t_0 is any fixed point in T, $\gamma_\alpha(T,d)$ is the γ_α-functional and Δ_{d_i} is the diameter given by $\Delta_{d_i}(T) = \sup_{s,t\in T} d_i(s,t)$.

The γ_α-functional of (T,d) is defined as

$$\gamma_\alpha(T,d) := \inf_{(T_n)} \sup_{t\in T} \sum_{n=0}^{\infty} 2^{n/\alpha} d(t, T_n), \tag{10}$$

where the infimum is taken with respect to all *admissible* sequences. A sequence $(T_n)_{n\geq 0}$ of subsets of T is called *admissible* if $|T_0| = 1$ and $|T_n| \leq 2^{2^n}$ for all $n \geq 1$.

For our proof, we will use the following estimate on $\gamma_\alpha(T,d)$, which involves the generalized Dudley's integral [31, 27].

$$\gamma_\alpha(T,d) \leq C_{(\alpha)} \int_0^{\Delta_d(T)} (\log N(T,d,\varepsilon))^{1/\alpha} \, d\varepsilon, \tag{11}$$

where $C_{(\alpha)}$ is a constant depending only on α and $N(T,d,\varepsilon)$ is the *covering number*, i.e., the smallest number of balls (in metric d and with radius ε) needed to cover set T.

Proof for Lemma 1

We recall the statement of Lemma 1 below.

Lemma 1. *Let $\sigma = \mathrm{ReLU}$. Fix $w \in \mathbb{R}^n$ and let $A \in \mathbb{R}^{m\times n}$ have i.i.d. $\mathcal{N}\left(0, \frac{1}{m}\right)$ entries. Define*

$$Z(u,v;w) := \langle Au, \sigma(Av) - \sigma(Aw)\rangle - \tfrac{1}{2}\langle u, v-w\rangle.$$

Suppose $\mathcal{T}_1, \mathcal{T}_2$ are sets (not depending on A) such that

$$\mathcal{T}_1 = \mathcal{S}_1 \cap \mathbb{B}^n(0,\alpha) \quad \text{and} \quad \mathcal{T}_2 = \mathcal{S}_2 \cap \mathbb{B}(w, \alpha r)$$

for some q-dimensional (affine) subspaces $\mathcal{S}_1, \mathcal{S}_2 \subseteq \mathbb{R}^n$ and real numbers $\alpha, r > 0$. Then for any $t \geq 1$,

$$\sup_{\substack{u\in\mathcal{T}_1\\ v\in\mathcal{T}_2}} |Z(u,v;w)| \leq C_1 \alpha^2 r \left(\sqrt{\frac{q}{m}} + \frac{q}{m} + \sqrt{\frac{t}{m}} + \frac{t}{m}\right)$$

with probability at least $1 - e^{-t}$. Here $C_1 > 0$ is an absolute constant.

Proof. First, we establish that $Z(u,v;w)$ has a mixed tail.

Let $a_i^\intercal$ be the i-th row of A, then $a_i \sim \mathcal{N}(0, I_n/m)$. For $u \in \mathbb{B}^n(0,\alpha)$ and $v \in \mathbb{B}(w,\alpha r)$, define random variables

$$Z^i_{u,v} := \langle a_i, u\rangle \left[\sigma(\langle a_i, v\rangle) - \sigma(\langle a_i, w\rangle)\right] - \tfrac{1}{2m}\langle u, v-w\rangle, \quad i \in [m].$$

We have $\mathbb{E}Z^i_{u,v} = 0$ by Lemma 2, and

$$Z_{u,v} := \sum_{i=1}^m Z^i_{u,v} = \langle Au, \sigma(Av) - \sigma(Aw)\rangle - \tfrac{1}{2}\langle u, v-w\rangle = Z(u,v;w).$$

For the increments of $Z^i_{u,v}$, we have

$$\begin{aligned} Z^i_{u,v} - Z^i_{u',v'} &= \langle a_i, u\rangle \sigma(a_i^\intercal v) - \tfrac{1}{2m}\langle u, v\rangle - \langle a_i, u'\rangle \sigma(a_i^\intercal v') + \tfrac{1}{2m}\langle u', v'\rangle \\ &\quad - \langle a_i, u-u'\rangle \sigma(a_i^\intercal w) + \tfrac{1}{2m}\langle u-u', w\rangle \end{aligned}$$

$$
\begin{aligned}
&= \langle a_i, u\rangle \,\sigma(a_i^\intercal v) - \tfrac{1}{2m}\langle u, v\rangle - \left[\langle a_i, u\rangle \,\sigma(a_i^\intercal v') - \tfrac{1}{2m}\langle u, v'\rangle\right] \\
&\quad + \left[\langle a_i, u\rangle \,\sigma(a_i^\intercal v') - \tfrac{1}{2m}\langle u, v'\rangle\right] - \langle a_i, u'\rangle \,\sigma(a_i^\intercal v') + \tfrac{1}{2m}\langle u', v'\rangle \\
&\quad - \langle a_i, u - u'\rangle \,\sigma(a_i^\intercal w) + \tfrac{1}{2m}\langle u - u', w\rangle \\
&= \langle a_i, u\rangle \left[\sigma(a_i^\intercal v) - \sigma(a_i^\intercal v')\right] - \tfrac{1}{2m}\langle u, v - v'\rangle \\
&\quad + \langle a_i, u - u'\rangle \left[\sigma(a_i^\intercal v') - \sigma(a_i^\intercal w)\right] - \tfrac{1}{2m}\langle u - u', v' - w\rangle
\end{aligned}
$$

We can estimate its sub-exponential norm from Lemma 4, which gives

$$
\begin{aligned}
\|Z^i_{u,v} - Z^i_{u',v'}\|_{\psi_1} &\le C_2 m^{-1}\left(\|u\|\|v - v'\| + \|u - u'\|\|v' - w\|\right) \\
&\le C_2 \alpha m^{-1}\left(r\|u - u'\| + \|v - v'\|\right).
\end{aligned}
$$

By Bernstein's inequality,

$$
\mathbb{P}\left(|Z_{u,v} - Z_{u',v'}| \ge t\right) \le 2\exp\left(-c\min\left(\frac{t^2}{d_2^2}, \frac{t}{d_1}\right)\right)
$$

where the metrics d_i are given by

$$
d_2^2 = \frac{\alpha^2}{m}\left(r\|u - u'\| + \|v - v'\|\right)^2 \quad \text{and} \quad d_1 = \frac{\alpha}{m}\left(r\|u - u'\| + \|v - v'\|\right).
$$

Therefore $(Z_{u,v})_{(u,v)\in\mathcal{T}}$ has a mixed tail with respect to the metric pair (Cd_1, Cd_2) for some absolute constant C.

Next, we bound the supremum of $Z(u, v; w)$. Without loss of generality, we will assume that $q \ge 1$. (In fact, if $q = 0$, then $\mathcal{T}_1, \mathcal{T}_2$ are either empty set or singleton, in which case the result is trivial or follows directly from Bernstein's inequality).

Denote $\mathcal{T} := \mathcal{T}_1 \times \mathcal{T}_2$ and define a metric d on $\mathcal{T}$ as

$$
d\left((u, v), (u', v')\right) := r\|u - u'\| + \|v - v'\|.
$$

It is easy to see that $d_2 = \frac{\alpha}{\sqrt{m}} d$ and $d_1 = \frac{\alpha}{m} d$. Also note that $\gamma_i(\mathcal{T}, td) = t\gamma_i(\mathcal{T}, d)$ from definition (10). We can assume that $\mathcal{S}_1$ is a subspace[1], then $Z_{0,v} = 0$ for $v \in \mathcal{T}_2$. Thus by Theorem 3, we have

$$
\sup_{(u,v)\in\mathcal{T}} |Z_{u,v}| \lesssim \frac{\alpha}{\sqrt{m}}\gamma_2(\mathcal{T}, d) + \frac{\alpha}{m}\gamma_1(\mathcal{T}, d) + \sqrt{t}\frac{4\alpha^2 r}{\sqrt{m}} + t\frac{4\alpha^2 r}{m}
$$

with probability at least $1 - e^{-t}$. It remains to estimate $\gamma_i(\mathcal{T}, d)$.
From (11) we have

$$
\gamma_i(\mathcal{T}, d) \le C_3 \int_0^{\Delta_d(\mathcal{T})} \left(\log N(\mathcal{T}, d, \varepsilon)\right)^{1/i} \, d\varepsilon, \quad i = 1, 2.
$$

Let d_{ℓ_2} be the Euclidean metric. Note that one can always obtain a ε-covering on $\mathcal{T}$ (with metric d) from the product set of a $\varepsilon/2$-covering on $\mathcal{T}_1$ (with metric rd_{ℓ_2}) and a $\varepsilon/2$-covering on $\mathcal{T}_2$ (with metric d_{ℓ_2}). Moreover, note that $\mathcal{T}_1$ is contained in a q-dimensional ball of radius α and $\mathcal{T}_2$ is contained in a q-dimensional ball of radius αr. Hence

$$
\begin{aligned}
N(\mathcal{T}, d, \varepsilon) &\le N\left(\mathcal{T}_1,\, rd_{\ell_2},\, \varepsilon/2\right) \cdot N\left(\mathcal{T}_2,\, d_{\ell_2},\, \varepsilon/2\right) \\
&\le N\left(\alpha\mathbb{B}^q,\, rd_{\ell_2},\, \varepsilon/2\right) \cdot N\left(\alpha r\mathbb{B}^q,\, d_{\ell_2},\, \varepsilon/2\right) \\
&= N\left(\mathbb{B}^q,\, d_{\ell_2},\, \frac{\varepsilon}{2\alpha r}\right) \cdot N\left(\mathbb{B}^q,\, d_{\ell_2},\, \frac{\varepsilon}{2\alpha r}\right) \\
&\le \left(1 + \frac{4\alpha r}{\varepsilon}\right)^{2q}.
\end{aligned}
$$

[1] If $\mathcal{S}_1$ is an affine subspace, let $q' = q + 1$ and let $\mathcal{S}_1'$ be the q'-dimensional subspace containing $\mathcal{S}_1$ (and origin). One can proceed with $\mathcal{S}_1'$ and q' for the proof. Finally, notice that $\sqrt{\frac{q'}{m}} + \frac{q'}{m} \le 2\left(\sqrt{\frac{q}{m}} + \frac{q}{m}\right)$, so this will give the same result with only a different absolute constant. (In fact, in our application of Lemma 1 for the multi-layer proof, $\mathcal{S}_1$ is chosen as $\mathrm{range}(A_i \cdots A_1)$, which is always a subspace.)

Here the last line uses estimate $N(\mathbb{B}^q, d_{\ell_2}, \varepsilon) \le \left(1 + \frac{2}{\varepsilon}\right)^q$ for the covering number of unit balls (see e.g., [30, Section 4.2]).

Note the estimate[2] $\int_0^a \log\left(\frac{2a}{x}\right) dx = a(\log 2 + 1) < 2a$, we get

$$\gamma_1(\mathcal{T}, d) \le C_3 \int_0^{4\alpha r} 2q \log\left(1 + \frac{4\alpha r}{\varepsilon}\right) d\varepsilon \le 2C_3 q \int_0^{4\alpha r} \log\left(\frac{8\alpha r}{\varepsilon}\right) d\varepsilon \le 16 C_3 \alpha r q.$$

Also note the inequality $\sqrt{\log(1+x)} < \sqrt{2}\log(1+x)$ for $x \ge 1$, we have

$$\begin{aligned}
\gamma_2(\mathcal{T}, d) &\le C_3 \int_0^{4\alpha r} \sqrt{2q} \log^{\frac{1}{2}}\left(1 + \frac{4\alpha r}{\varepsilon}\right) d\varepsilon \\
&\le 2C_3\sqrt{q} \int_0^{4\alpha r} \log\left(1 + \frac{4\alpha r}{\varepsilon}\right) d\varepsilon \\
&\le 2C_3\sqrt{q} \int_0^{4\alpha r} \log\left(\frac{8\alpha r}{\varepsilon}\right) d\varepsilon \\
&\le 16 C_3 \alpha r \sqrt{q}.
\end{aligned}$$

Therefore with probability at least $1 - e^{-t}$,

$$\sup_{(u,v)\in\mathcal{T}} |Z_{u,v}| \le C_1 \alpha^2 r \left(\sqrt{\frac{q}{m}} + \frac{q}{m} + \sqrt{\frac{t}{m}} + \frac{t}{m}\right).$$

□

Lemma 4. *Let* $\sigma = \mathrm{ReLU}$. *For* $u, x, y \in \mathbb{R}^n$ *and* $g \sim \mathcal{N}(0, I_n)$, *the (mean zero) random variable*

$$Z^g := \langle g, u\rangle \left[\sigma(g^\intercal x) - \sigma(g^\intercal y)\right] - \tfrac{1}{2}\langle u, x - y\rangle$$

has sub-exponential norm $\|Z^g\|_{\psi_1} \le C_2 \|u\| \|x - y\|$, *where* C_2 *is an absolute constant.*

Proof. It is easy to see that Z^g is mean zero from Lemma 2. Also from the following two properties of ψ_1, ψ_2-norms (see [30, Section 2.7]):

$$\|X - \mathbb{E}X\|_{\psi_1} \lesssim \|X\|_{\psi_1} \quad \text{and} \quad \|XY\|_{\psi_1} \le \|X\|_{\psi_2}\|Y\|_{\psi_2},$$

we have (note that σ is 1-Lipschitz)

$$\|Z^g\|_{\psi_1} \lesssim \|\langle g, u\rangle\|_{\psi_2} \|\sigma(g^\intercal x) - \sigma(g^\intercal y)\|_{\psi_2} \lesssim \|\langle g, u\rangle\|_{\psi_2} \|\langle g, x - y\rangle\|_{\psi_2}.$$

The result follows by noting that $\|\langle g, u\rangle\|_{\psi_2} = \|g_1\|_{\psi_2}\|u\|$ where $g_1 \sim \mathcal{N}(0,1)$. □

D Proof for Theorem 1

Proof of Theorem 1. First we write

$$x^{k+1} - x^* = \theta\left(x^k - x^* - 2^d \tilde{A}_d^\intercal \Phi^\intercal [\Phi\mathcal{G}(x^k) - y]\right) + (1-\theta)(x^k - x^*).$$

For any fixed $r > 0$, using triangle inequality and Lemma 5 (with events $\mathcal{E}_i$ defined as in Lemma 5) we can conclude that if $\|x^k - x^*\| \le r$, then with probability at least $1 - \mathbb{P}(\mathcal{E}_1) - \mathbb{P}(\mathcal{E}_2) - \mathbb{P}(\mathcal{E}_3) - 2e^{-10n_0}$,

$$\|x^{k+1} - x^*\| \le \frac{\theta}{2}\left(r + 30 \cdot 2^d \sqrt{\frac{n_0}{m}}\|\epsilon\|\right) + |1-\theta| r = \alpha(r + \beta\varepsilon) \tag{12}$$

where

$$\alpha = \frac{\theta}{2} + |1-\theta|, \quad \beta = \frac{\theta/2}{|1-\theta| + \theta/2}, \quad \varepsilon = 30 \cdot 2^d \sqrt{n_0/m}\|\epsilon\|.$$

Now define a sequence $\{r_k\}_{k\in\mathbb{N}}$ such that $r_{k+1} = \alpha(r_k + \beta\varepsilon)$ and $r_0 = R$. We can find its general formula as follow:

$$r_{k+1} - \frac{\alpha\beta}{1-\alpha}\varepsilon = \alpha\left(r_k - \frac{\alpha\beta}{1-\alpha}\varepsilon\right) \quad \Rightarrow \quad r_k = \alpha^k\left(R - \frac{\alpha\beta}{1-\alpha}\varepsilon\right) + \frac{\alpha\beta}{1-\alpha}\varepsilon.$$

[2]This comes from the indefinite integral $\int \log\left(\frac{a}{x}\right) dx = x\log\left(\frac{a}{x}\right) + x + C$.

Next, by induction on k (i.e., apply (12) with $r = r_k$ for $k = 0, 1, 2, \dots$) we get

$$\|x^k - x^*\| \le r_k \le \alpha^k R + \frac{\alpha\beta}{1-\alpha}\varepsilon, \quad k \in \mathbb{N}. \tag{13}$$

Notice that the events $\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3$ remain unchanged throughout iterations, so (13) holds with probability at least $1 - \mathbb{P}(\mathcal{E}_1) - \mathbb{P}(\mathcal{E}_2) - \mathbb{P}(\mathcal{E}_3) - 2ke^{-10n_0}$.

Lastly, from Lemma 6, Lemma 8 and Lemma 9 we know $\mathbb{P}(\mathcal{E}_i) \le 3e^{-10n_0}$ for $i = 1, 2$ and $\mathbb{P}(\mathcal{E}_3) \le 2e^{-10n_0}$. Also, $\|\mathcal{G}(x^k) - \mathcal{G}(x^*)\| \le 3\|x^k - x^*\|$ on $\mathcal{E}_2^c$. This completes the proof. $\square$

Lemma 5. *Fix $r > 0$ and assume assumptions A1-A4 hold. If $\|x^k - x^*\| \le r$, then after one iteration according to PLUGIn-CS with step size $\eta = 2^d$, we have*

$$\|x^{k+1} - x^*\| \le \frac{1}{2}\left(r + 30 \cdot 2^d \sqrt{\frac{n_0}{m}}\|\epsilon\|\right)$$

with probability at least $1 - \mathbb{P}(\mathcal{E}_1) - \mathbb{P}(\mathcal{E}_2) - \mathbb{P}(\mathcal{E}_3) - 2e^{-10n_0}$.
Here $\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3$ are the events

$$\begin{aligned}
\mathcal{E}_1 &:= \left\{\|\tilde{A}_d^\intercal \Phi^\intercal \epsilon\| > 15\sqrt{n_0/m}\|\epsilon\|\right\}, \\
\mathcal{E}_2 &:= \left\{\max(L_{\tilde{A}_i}, L_{\mathcal{G}_i}) > 3 \textit{ for all } i \in [d]\right\} \quad \textit{and} \\
\mathcal{E}_3 &:= \left\{\|I - \Phi^\intercal\Phi\|_{\mathcal{R}} > \tfrac{1}{36\cdot 2^d}\right\},
\end{aligned}$$

where $L_{\mathcal{G}_i}$ and $L_{\tilde{A}_i}$ denote the Lipschitz constants of $\mathcal{G}_i, \tilde{A}_i : \mathbb{R}^{n_0} \to \mathbb{R}^{n_i}$ respectively, and

$$\|I - \Phi^\intercal\Phi\|_{\mathcal{R}} := \sup_{z \in \mathcal{R}\setminus\{0\}} \tfrac{\|(I-\Phi^\intercal\Phi)z\|}{\|z\|}$$

with $\mathcal{R} := \mathrm{range}(\mathcal{G}) - \mathrm{range}(\mathcal{G})$ being the Minkowski sum of $\mathrm{range}(\mathcal{G})$ and $-\mathrm{range}(\mathcal{G})$.

Proof. For $x \in \mathbb{R}^{n_0}$, denote $x_0 = x$ and $x_i = \mathcal{G}_i(x)$ for $i \in [d]$. Then

$$\begin{aligned}
x^{k+1} - x^* &= x^k - x^* - 2^d \tilde{A}_d^\intercal \Phi^\intercal [\Phi\mathcal{G}(x^k) - \Phi\mathcal{G}(x^*) - \epsilon] \\
&= (x_0^k - x_0^*) - 2\tilde{A}_1^\intercal(x_1^k - x_1^*) \\
&\quad + 2\tilde{A}_1^\intercal\left[(x_1^k - x_1^*) - 2A_2^\intercal(x_2^k - x_2^*)\right] \\
&\quad + \dots \\
&\quad + 2^{d-1}\tilde{A}_{d-1}^\intercal\left[(x_{d-1}^k - x_{d-1}^*) - 2A_d^\intercal(x_d^k - x_d^*)\right] \\
&\quad + 2^d\tilde{A}_d^\intercal(I - \Phi^\intercal\Phi)(x_d^k - x_d^*) \\
&\quad + 2^d\tilde{A}_d^\intercal\Phi^\intercal\epsilon
\end{aligned}$$

thus we can write

$$\begin{aligned}
\|x^{k+1} - x^*\| = \sup_{u \in \mathbb{S}^{n_0-1}} & 2\left(\langle A_1 u, x_1^k - x_1^*\rangle - \tfrac{1}{2}\langle u, x_0^k - x_0^*\rangle\right) \\
& + 2^2\left(\left\langle A_2\tilde{A}_1 u, x_2^k - x_2^*\right\rangle - \tfrac{1}{2}\left\langle \tilde{A}_1 u, x_1^k - x_1^*\right\rangle\right) \\
& + \dots \\
& + 2^d\left(\left\langle A_d\tilde{A}_{d-1}u, x_d^k - x_d^*\right\rangle - \tfrac{1}{2}\left\langle \tilde{A}_{d-1}u, x_{d-1}^k - x_{d-1}^*\right\rangle\right) \\
& - 2^d\left\langle \tilde{A}_d u, (I - \Phi^\intercal\Phi)(x_d^k - x_d^*)\right\rangle \\
& - 2^d\left\langle u, \tilde{A}_d^\intercal\Phi^\intercal\epsilon\right\rangle \\
\le \mathrm{I} + \mathrm{II} + \mathrm{III}&,
\end{aligned}$$

where

$$\mathrm{I} := \sum_{i=0}^{d-1} 2^{i+1} \sup_{u \in \mathbb{S}^{n_0-1}} Z_{i+1}\left(\tilde{A}_i u,\, x_i^k\right),$$

$$\mathrm{II} := 2^d \|\tilde{A}_d\| \|(I - \Phi^\intercal \Phi)(x_d^k - x_d^*)\|,$$
$$\mathrm{III} := 2^d \|\tilde{A}_d^\intercal \Phi^\intercal \epsilon\|$$

with

$$Z_j(u,v) := \left\langle A_j u, \sigma(A_j v) - \sigma(A_j x_{j-1}^*)\right\rangle - \tfrac{1}{2}\left\langle u, v - x_{j-1}^*\right\rangle, \quad j \in [d].$$

We will estimate I, II and III as below.

bound for I

On event $\mathcal{E}_2^c$, $\forall i \in [d-1]$ we have

$$\tilde{A}_i \mathbb{S}^{n_0-1} \subseteq \mathrm{range}(\tilde{A}_i) \cap \mathbb{B}^{n_i}(0,3) =: \mathcal{T}_1^i,$$
$$x_i^k \in \mathrm{range}(\mathcal{G}_i) \cap \mathbb{B}(x_i^*, 3r) =: \mathcal{T}_2^i.$$

By Lemma 7, there are $N_{\mathcal{G}_i}$ many n_0-dimensional affine subspaces $\{\mathcal{S}_{i,j}\}$ such that

$$\mathcal{T}_2^i \subseteq \cup_{j\in[N_{\mathcal{G}_i}]} \mathcal{T}_{2,j}^i \quad \text{where} \quad \mathcal{T}_{2,j}^i = \mathcal{S}_{i,j} \cap \mathbb{B}(x_i^*, 3r) \subseteq \mathbb{R}^{n_i} \text{ and } N_{\mathcal{G}_i} \le \psi_i := \prod_{j=1}^{i} \left(\frac{en_j}{n_0}\right)^{n_0}.$$

For $i \in [d-1]$, apply Lemma 1 on $\mathcal{T}_1^i \times \mathcal{T}_{2,j}^i$ followed by a union bound over $j \in [N_{\mathcal{G}_i}]$, we get

$$\sup_{\mathcal{T}_1^i \times \mathcal{T}_2^i} Z_{i+1}(u,v) \le C_1(9r)\left(\sqrt{\frac{n_0}{n_{i+1}}} + \frac{n_0}{n_{i+1}} + \sqrt{\frac{t_{i+1}}{n_{i+1}}} + \frac{t_{i+1}}{n_{i+1}}\right)$$

with probability (over A_{i+1} and conditioning on $\{A_j\}_{j\in[i]}$) at least $1 - \psi_i e^{-t_{i+1}}$.

Choose $t_{i+1} = 2\log\psi_i = 2n_0 \sum_{j=1}^{i} \log(\frac{en_j}{n_0})$, then we get

$$\mathbb{P}_{A_{i+1}}\left(\sup_{\mathcal{T}_1^i \times \mathcal{T}_2^i} Z_{i+1}(u,v) \le 9C_1 r \cdot 4\sqrt{\frac{2\log\psi_i}{n_{i+1}}}\right) \ge 1 - e^{-\log\psi_i}, \quad \forall i \in [d-1].$$

Also for $i = 0$, applying Lemma 1 on $\mathbb{B}^{n_0}(0,1) \times \mathbb{B}(x^*, r)$, we get

$$\sup_{\substack{u\in\mathbb{B}^{n_0}(0,1)\\ v\in\mathbb{B}(x^*,r)}} Z_1(u,v) \le C_1 r \cdot 4\sqrt{\frac{10n_0}{n_1}}$$

with probability (over A_1) at least $1 - e^{-10n_0}$.

Therefore under assumption A2 (with $C_0 \ge 160 \cdot 144^2 C_1^2$), we have

$$\begin{aligned}
\sum_{i=0}^{d-1} 2^{i+1} \sup_{u\in\mathbb{S}^{n_0-1}} Z_{i+1}\left(\tilde{A}_i u,\, x_i^k\right) &\le \frac{r}{144} + \sum_{i=1}^{d-1} 2^{i+1} \cdot \frac{36r}{144}\sqrt{\frac{2}{160\cdot 5^{i+1}}} \\
&= \frac{r}{144} + \frac{r}{4}\cdot\frac{1}{10}\sum_{i=1}^{d-1}\left(\frac{2}{\sqrt{5}}\right)^i \\
&< \frac{r}{4}\cdot\frac{1}{10}\sum_{i=0}^{\infty}\left(\frac{2}{\sqrt{5}}\right)^i \\
&< \frac{r}{4}
\end{aligned}$$

with probability at least $1 - \mathbb{P}(\mathcal{E}_2) - e^{-10n_0} - \sum_{i=1}^{d-1} e^{-\log\psi_i}$.

Also note that (assume $C_0 \ge 160 \cdot 144^2$)

$$\log\psi_i = n_0 \sum_{j=1}^{i} \log\left(\frac{en_j}{n_0}\right) \ge n_0 i \log(eC_0) > 11 n_0 i,$$

so $\sum_{i\ge1} e^{-\log\psi_i} \le \frac{e^{-11n_0}}{1-e^{-11n_0}} < e^{-10n_0}$.

bound for II

On event $\mathcal{E}_2^c \cap \mathcal{E}_3^c$, we have $\|\tilde{A}_d\| \leq 3$ and

$$\|(I - \Phi^\intercal\Phi)(x_d^k - x_d^*)\| \leq \frac{1}{36 \cdot 2^d}\|x_d^k - x_d^*\| \leq \frac{3r}{36 \cdot 2^d}.$$

Thus II $\leq r/4$.

bound for III

Note that on $\mathcal{E}_1^c$,

$$2^d\|\tilde{A}_d^\intercal\Phi^\intercal\epsilon\| \leq 15 \cdot 2^d\sqrt{n_0/m}\|\epsilon\|.$$

□

Lemma 6. *Under assumptions A1-A4, we have*

$$\mathbb{P}\left(\|A_1^\intercal A_2^\intercal \cdots A_d^\intercal\Phi^\intercal\epsilon\| \geq 15\sqrt{\frac{n_0}{m}}\|\epsilon\|\right) \leq 3e^{-10n_0}.$$

Proof. Denote $A_{d+1} = \Phi$ and $s_i = A_{i+1}^\intercal \cdots A_{d+1}^\intercal\epsilon$ for $i \in [d]$. Also let $s_{d+1} = \epsilon$ and $n_{d+1} = m$. For $i \in [d+1]$, by Lemma 3(a) we have

$$\mathbb{P}_{A_i}\left(\sqrt{n_i}\|A_i^\intercal s_i\| \leq \sqrt{n_{i-1}}\|s_i\| + \sqrt{t_i}\|s_i\|\right) \geq 1 - e^{-t_i/2}, \quad \forall t_i > 0.$$

Choose $t_1 = 20n_0$ and $t_j = n_{j-1}/4^{j-1}$ for $j > 1$, we get

$$\mathbb{P}_{A_1}\left(\|A_1^\intercal s_1\| \leq (1+\sqrt{20})\sqrt{\frac{n_0}{n_1}}\|s_1\|\right) \geq 1 - e^{-10n_0},$$
$$\mathbb{P}_{A_i}\left(\|A_i^\intercal s_i\| \leq (1+2^{-i+1})\sqrt{\frac{n_{i-1}}{n_i}}\|s_i\|\right) \geq 1 - e^{-n_{i-1}/4^i}, \quad i > 1.$$

Thus with probability at least $1 - e^{-10n_0} - \sum_{i=2}^{d+1} e^{-n_{i-1}/4^i}$,

$$\begin{aligned}
\|A_1^\intercal A_2^\intercal \cdots A_d^\intercal\Phi^\intercal\epsilon\| &\leq \left(1+\sqrt{20}\right)\sqrt{\frac{n_0}{n_1}} \cdot \prod_{i=2}^{d+1}\left(1 + \frac{1}{2^{i-1}}\right)\sqrt{\frac{n_{i-1}}{n_i}} \\
&\leq \left(1+\sqrt{20}\right)\sqrt{\frac{n_0}{m}} \cdot \prod_{i=1}^{\infty}\left(1 + \frac{1}{2^i}\right) \\
&< 15\sqrt{n_0/m}
\end{aligned}$$

where the last inequality uses estimate[3] $\prod_{i=1}^{\infty}\left(1 + \frac{1}{2^i}\right) \leq e$ and $(1+\sqrt{20})e < 15$.

It remains to show $\sum_{i=2}^{d+1} e^{-n_{i-1}/4^i} \leq 2e^{-10n_0}$ for the desired probability bound. Note that by assumption A2 (assume $C_0 \geq 40$),

$$\frac{n_i}{4^{i+1}} \geq \frac{1}{4}C_0 n_0 \sum_{j=0}^{i-1}\log\left(\frac{en_j}{n_0}\right) \geq 10n_0\, i.$$

Hence

$$\sum_{i=2}^{d+1} e^{-n_{i-1}/4^i} \leq \sum_{i=2}^{d+1} e^{-10n_0(i-1)} < \sum_{i=1}^{\infty} e^{-10n_0 i} = \frac{e^{-10n_0}}{1 - e^{-10n_0}} < 2e^{-10n_0}.$$

□

With ReLU (or positively homogeneous) activation functions, the range of neural network (in each layer) is contained in a union of affine subspaces. The following lemma, which is based on ideas and results in [9], gives a precise statement of this.

[3]For $\alpha > 0$, estimate $\sum_{j=1}^{\infty}\log\left(1 + \alpha 2^{-j}\right) \leq \sum_{j=1}^{\infty}\alpha 2^{-j} = \alpha$ holds, thus $\prod_{j=1}^{\infty}\left(1 + \frac{\alpha}{2^j}\right) \leq e^\alpha$.

Lemma 7. *If* $\min_{j\in[d]}\{n_j\} \geq n_0$*, then for* $i \in [d]$*,* $\operatorname{range}(\mathcal{G}_i)$ *is contained in a union of affine subspaces. Precisely,*

$$\operatorname{range}(\mathcal{G}_i) \subseteq \cup_{j\in[N_{\mathcal{G}_i}]}\mathcal{S}_{i,j} \quad \textit{where} \quad N_{\mathcal{G}_i} \leq \prod_{j=1}^{i}\left(\frac{en_j}{n_0}\right)^{n_0}.$$

Here each $\mathcal{S}_{i,j}$ *is some* n_0*-dimensional affine subspace (which depends on* $\{A_l\}_{l\in[i]}$*) in* $\mathbb{R}^{n_i}$*.*

Proof. The theory on hyperplane arrangements [28, Chapter 6.1] tells us that n hyperplanes in $\mathbb{R}^k$ (assume $n \geq k$) partition the space $\mathbb{R}^k$ into at most $\sum_{j=0}^{k}\binom{n}{j}$ regions[4].

Also for $k \in [n]$,

$$\sum_{j=0}^{k}\binom{n}{j} \leq \sum_{j=0}^{k}\frac{n^j}{j!} \leq \sum_{j=0}^{k}\frac{k^j}{j!}\left(\frac{n}{k}\right)^j \leq \left(\frac{n}{k}\right)^k\sum_{j=0}^{\infty}\frac{k^j}{j!} = \left(\frac{en}{k}\right)^k.$$

So consider $\operatorname{range}(\mathcal{G}_1) = \{\sigma(A_1x) : x \in \mathbb{R}^{n_0}\}$. Denote by a_j^1 ($j \in [n_1]$) the rows of A_1 and let H be the set of hyperplanes $H := \cup_{j\in[n_1]}\{x : \langle a_j^1, x\rangle = 0\}$. Then H partitions $\mathbb{R}^{n_0}$ into at most $(en_1/n_0)^{n_0}$ regions. Note that σ is linear in each of these regions (thus the mapping $\mathcal{G}_1$ is linear in each region), so $\operatorname{range}(\mathcal{G}_1)$ is contained in at most $(en_1/n_0)^{n_0}$ many n_0-dimensional (affine) subspace.

The result then follows by induction. □

The following lemma shows that the network $\mathcal{G}$ in our model is Lipschitz with high probability. This may be an interesting result on its own.

Lemma 8. *For mappings* $\mathcal{G}_i$*,* $\tilde{A}_i : \mathbb{R}^{n_0} \to \mathbb{R}^{n_i}$*, let* $L_{\mathcal{G}_i}$ *and* $L_{\tilde{A}_i}$ *be their Lipschitz constants respectively. Under assumptions A1 and A2, we have*

$$\mathbb{P}\left(\max\{L_{\tilde{A}_i}, L_{\mathcal{G}_i}\} \leq 3 \ \textit{ for all } i \in [d]\right) \geq 1 - 3e^{-10n_0}.$$

Proof. Denote $\tilde{\mathcal{R}}_0 = \mathcal{R}_0 = \mathbb{R}^{n_0}$ and

$$\mathcal{R}_j = \operatorname{range}(\mathcal{G}_j) - \operatorname{range}(\mathcal{G}_j), \quad \tilde{\mathcal{R}}_j = \mathcal{R}_j \cup \operatorname{range}(\tilde{A}_j), \quad j \in [d].$$

Note that $\tilde{A}_j$ is linear, so $\operatorname{range}(\tilde{A}_j)$ is a subspace in $\mathbb{R}^{n_i}$ with dimension at most n_0.

Since σ is 1-Lipschitz, we have

$$\begin{aligned}\|\mathcal{G}_i(x) - \mathcal{G}_i(x')\| &= \|\sigma(A_i\mathcal{G}_{i-1}(x)) - \sigma(A_i\mathcal{G}_{i-1}(x'))\| \\ &\leq \|A_i\left(\mathcal{G}_{i-1}(x) - \mathcal{G}_{i-1}(x')\right)\| \\ &\leq \|A_i\|_{\mathcal{R}_{i-1}}\|\mathcal{G}_{i-1}(x) - \mathcal{G}_{i-1}(x')\|.\end{aligned}$$

Hence

$$\|\mathcal{G}_i(x) - \mathcal{G}_i(x')\| \leq \left(\prod_{l=1}^{i}\|A_l\|_{\tilde{\mathcal{R}}_{l-1}}\right)\|x - x'\|, \quad \forall i \in [d].$$

Similarly,

$$\|\tilde{A}_ix - \tilde{A}_ix'\| \leq \left(\prod_{l=1}^{i}\|A_l\|_{\tilde{\mathcal{R}}_{l-1}}\right)\|x - x'\|, \quad \forall i \in [d].$$

By Lemma 7, $\operatorname{range}(\mathcal{G}_i)$ is contained in a union of $N_{\mathcal{G}_i}$ many n_0-dimensional affine subspaces, so $\mathcal{R}_i$ is contained in a union of at most $N_{\mathcal{G}_i}^2$ many $2n_0$-dimensional affine subspaces. Since every

[4]Such regions are also called k-faces or k-cells. Relative to each of the n hyperplanes, all points inside a region are on the same side.

$2n_0$-dimensional affine subspaces in $\mathbb{R}^{n_i}$ is also contained in a $(2n_0+1)$-dimensional subspace, we can further write this as

$$\tilde{\mathcal{R}}_i = \mathcal{R}_i \cup \mathrm{range}(\tilde{A}_i) \subseteq \cup_{j\in[N_{\mathcal{G}_i}^2+1]}\mathcal{S}_{i,j} \quad \text{where} \quad N_{\mathcal{G}_i} \leq \psi_i := \prod_{j=1}^{i}\left(\frac{en_j}{n_0}\right)^{n_0},$$

and each $\mathcal{S}_{i,j}$ is a $(2n_0+1)$-dimensional subspace in $\mathbb{R}^{n_i}$.

Thus by Lemma 3(b) and union bound we have, for $i \in [d-1]$,

$$\mathbb{P}_{A_{i+1}}\left(\sqrt{n_{i+1}}\|A_{i+1}\|_{\tilde{\mathcal{R}}_i} \geq \sqrt{n_{i+1}} + \sqrt{2n_0+1} + \sqrt{t_i}\right) \leq (\psi_i^2+1)e^{-t_i/2}, \quad \forall t_i > 0.$$

Choose $t_i = 26\log\psi_i = 26n_0\sum_{j=1}^{i}\log(\frac{en_j}{n_0}) > 2n_0+1$ we get

$$\mathbb{P}_{A_{i+1}}\left(\|A_{i+1}\|_{\tilde{\mathcal{R}}_i} \geq 1 + 2\sqrt{\frac{26\log\psi_i}{n_{i+1}}}\right) \leq e^{-10\log\psi_i}.$$

Under assumption A2 (with $C_0 \geq 2^2\cdot 26$), this implies

$$\mathbb{P}_{A_{i+1}}\left(\|A_{i+1}\|_{\tilde{\mathcal{R}}_i} \geq 1 + \frac{1}{2^{i+1}}\right) \leq e^{-10\log\psi_i}, \quad i \in [d-1].$$

Also by Lemma 3(b) with $t = 20n_0$ and assumption A2 (assume $C_0 \geq 2^2\cdot 26$), we have

$$\mathbb{P}_{A_1}\left(\|A_1\|_{\tilde{\mathcal{R}}_0} \geq 1 + \frac{1}{2}\right) \leq e^{-10n_0}.$$

Therefore with probability at least $1 - e^{-10n_0} - \sum_{i=1}^{d-1}e^{-10\log\psi_i}$,

$$\forall i \in [d], \quad \prod_{l=1}^{i}\|A_l\|_{\tilde{\mathcal{R}}_{l-1}} \leq \prod_{l=1}^{i}\left(1+\frac{1}{2^l}\right) \leq \prod_{l=1}^{\infty}\left(1+\frac{1}{2^l}\right) < 3.$$

Finally, note that $\log\psi_i \geq in_0$, so we have $\sum_{i=1}^{d-1}e^{-10\log\psi_i} \leq \sum_{i=1}^{\infty}e^{-10n_0 i} < 2e^{-10n_0}$. This completes the proof. □

Lemma 9. *Let $\mathcal{R} = \mathrm{range}(\mathcal{G}) - \mathrm{range}(\mathcal{G})$. If $\min_{j\in[d]}\{n_j\} \geq n_0$ and assumption A3 holds, then*

$$\mathbb{P}_\Phi\left(\|I - \Phi^\intercal\Phi\|_{\mathcal{R}} > \tfrac{1}{36\cdot 2^d}\right) \leq 2e^{-10n_0}.$$

Proof. Let $\psi = \prod_{j=1}^{d}\left(\frac{en_j}{n_0}\right)^{n_0}$. Similar to the proof in Lemma 8, we know that $\mathcal{R}$ is contained in a union of at most ψ^2 many $(2n_0+1)$-dimensional subspaces in $\mathbb{R}^{n_d}$.

From Lemma 3(b) and a union bound we get

$$\mathbb{P}_\Phi\left(|\|\Phi\|_{\mathcal{R}} - 1| \geq \sqrt{\frac{2n_0+1}{m}} + \sqrt{\frac{t}{m}}\right) \leq 2\psi^2 e^{-t/2}.$$

By choosing $t = 24n_0\sum_{j=1}^{d}\log(\frac{en_j}{n_0})$ and noticing that $\|I - \Phi^\intercal\Phi\|_{\mathcal{R}} = |\|\Phi\|_{\mathcal{R}} - 1|^2$, we have

$$\mathbb{P}_\Phi\left(\|I - \Phi^\intercal\Phi\|_{\mathcal{R}} \geq 4\frac{t}{m}\right) \leq 2\psi^2 e^{-t/2} \leq 2e^{-10n_0}.$$

This completes the proof with $c_0 \geq 96\cdot 36$ in assumption A3. □

E An Example of n_i

Here we show if $n_i = \beta C_0 5^d n_0 d(2d-i)$ where β is any fixed number such that $\beta C_0 \in \mathbb{N}$ and $\beta \geq 4 + \log C_0$, then n_i satisfy (5).

In fact, note that $2\log d < d$ and $\log(2\beta) < \beta$, we have

$$\begin{aligned}
\log\left(\prod_{j=0}^{i-1}\frac{en_j}{n_0}\right) &= 1 + \sum_{j=1}^{i-1}\log\left(\frac{en_j}{n_0}\right) \\
&\leq 1 + (d-1)\log\left(e\beta C_0 5^d \cdot 2d^2\right) \\
&= 1 + (d-1)[d\log 5 + 2\log d + \log(eC_0)] + (d-1)\log(2\beta) \\
&< 1 + d(d-1)[\log 5 + 1 + \log(eC_0)] + (d-1)\beta \\
&\leq \beta + d(d-1)\beta + (d-1)\beta \\
&= \beta d^2.
\end{aligned}$$

Since $n_i \geq C_0 5^d n_0(\beta d^2)$, it is easy to see that n_i satisfy (5).

Remark: A similar argument as above can also show that $n_i = \beta C_0 5^i n_0 i^2$ *satisfy* (5).

F Code Link

Codes for numerical experiments are available at https://github.com/babhrujoshi/PLUGIn.