Interpolating Noisy Datasets hurt Adversarial Robustness

Amartya Sanyal & Varun Kanade
Department of Computer Science
University of Oxford
Oxford, UK
{amartya.sanyal,varunk}@cs.ox.ac.uk

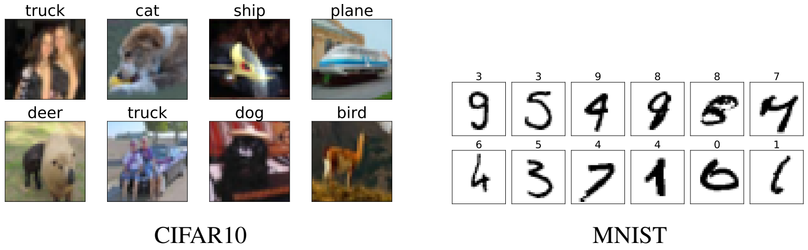
Puneet K. Dokania & Philip H.S. Torr
Department of Engineering Science
University of Oxford
Five AI
{puneet,phst}@robots.ox.ac.uk

Abstract

When trained with SGD, deep neural networks essentially achieve zero training error, even in the presence of label noise, while also exhibiting good generalization on natural test data, something referred to as benign overfitting [2, 8]. However, these models are vulnerable to adversarial attacks. We identify label noise as one of the causes for adversarial vulnerability, and provide theoretical and empirical evidence in support of this. Surprisingly, we find several instances of label noise in datasets such as MNIST and CIFAR, and that robustly trained models incur training error on some of these, i.e. they don't fit the noise. We believe this highlights the importance of removing label noise from dataset as well as protecting the integrity of the dataset curation process.By means of simple theoretical setups, we show how the choice of representation can drastically affect adversarial robustness. We also provide some experimental evidence how incorporating better inductive biases can help improve robustness.

1 Introduction

Modern machine learning methods achieve a very high accuracy on wide range of tasks, e.g. in computer vision, natural language processing etc. However, especially in vision tasks, they have been shown to be highly vulnerable to small adversarial perturbations that are imperceptible to the human eye [9, 7, 11] . This vulnerability poses serious security concerns when these models are deployed in real-world tasks (cf. [28, 31, 15, 21]). A large body of research has been devoted to crafting defences to protect neural networks from adversarial attacks (e.g. [11, 27, 35, 24, 38]). However, such defences have usually been broken by future attacks [1, 34]. This arms race between attacks and defenses suggests that to create a truly robust model would require a deeper understanding of the source of this vulnerability.



CIFAR10

MNIST

Figure 1: Label noise in CIFAR10 and MNIST. Text above the image indicates the training set label.

NeurIPS 2020 Workshop on Dataset Curation and Security

Our goal in this paper is not to propose new defenses, but to highlight the importance of proper dataset curation for adversarial robustness. This becomes especially relevant with interpolating models like deep neural networks. Starting with the celebrated work of Zhang *et al.* [37] it has been observed that neural networks trained with SGD are capable of memorizing large amounts of label noise. Recent theoretical work (e.g. [23, 4, 3, 13, 6, 5, 2, 26, 8]) has also sought to explain why fitting training data perfectly does not lead to a large drop in test accuracy, as the classical notion of overfitting might suggest. This is commonly referred to as *memorization* or *interpolation*. We show through simple theoretical models, as well as experiments on standard datasets, that there are scenarios where label noise causes significant adversarial vulnerability, even when high natural (test) accuracy can be achieved. Surprisingly, we find that label noise is not at all uncommon in datasets such as MNIST and CIFAR-10 (see Figure 1).

2 Theoretical Setting

We develop a simple theoretical framework to demonstrate how overfitting, even very minimal, label noise causes significant adversarial vulnerability. For notations please see Appendix B.

The following result provides a sufficient condition under which even a small amount of label noise causes any classifier that fits the training data perfectly to have significant adversarial error. Informally, Theorem 1 states that if the data distribution has significant probability mass in a union of (a relatively small number of, and possibly overlapping) balls, each of which has roughly the same probability mass (cf. Eq. (1)), then even a small amount of label noise renders this entire region vulnerable to adversarial attacks to classifiers that fit the training data perfectly.

Theorem 1. *Let c be the target classifier, and let $\mathcal{D}$ be a distribution over $(\mathbf{x}, y)$, such that $y = c(\mathbf{x})$ in its support. Using the notation $\mathbb{P}_D[A]$ to denote $\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}[\mathbf{x} \in A]$ for any measurable subset $A \subseteq \mathbb{R}^d$, suppose that there exist $c_1 \geq c_2 > 0$, $\rho > 0$, and a finite set $\zeta \subset \mathbb{R}^d$ satisfying*

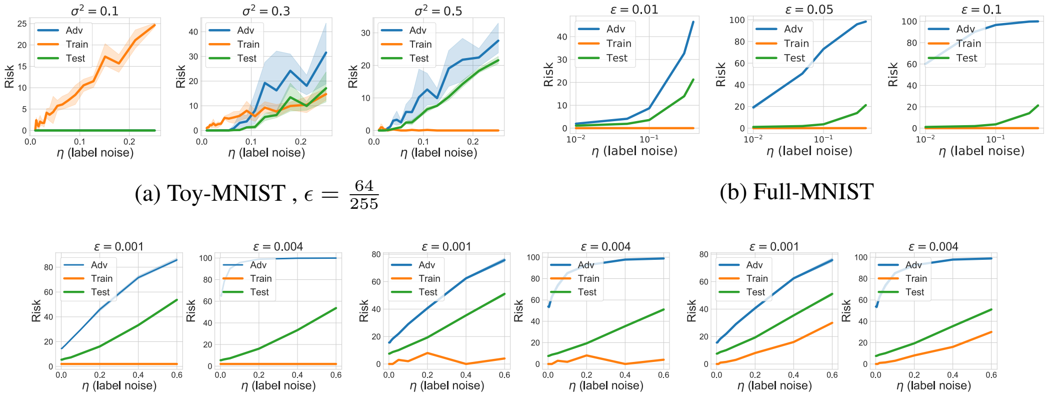
$$\mathbb{P}_{\mathcal{D}}\left[\bigcup_{\mathbf{s}\in\zeta} \mathcal{B}_\rho^p(\mathbf{s})\right] \geq c_1 \quad \textit{and} \quad \forall \mathbf{s} \in \zeta,\ \mathbb{P}_{\mathcal{D}}\left[\mathcal{B}_\rho^p(\mathbf{s})\right] \geq \frac{c_2}{|\zeta|} \tag{1}$$

where $\mathcal{B}_\rho^p(\mathbf{s})$ represents a ℓ_p-ball of radius ρ around $\mathbf{s}$. Further, suppose that each of these balls contain points from a single class i.e. for all $\mathbf{s} \in \zeta$, for all $\mathbf{x}, \mathbf{z} \in \mathcal{B}_\rho^p(\mathbf{s}) : c(\mathbf{x}) = c(\mathbf{z})$.

Let $\mathcal{S}_m$ be a dataset of m i.i.d. samples drawn from $\mathcal{D}$, which subsequently has each label flipped independently with probability η. For any classifier f that perfectly *fits the training data $\mathcal{S}_m$ i.e. $\forall\ \mathbf{x}, y \in \mathcal{S}_m, f(\mathbf{x}) = y$, $\forall \delta > 0$ and $m \geq \frac{|\zeta|}{\eta c_2} \log\left(\frac{|\zeta|}{\delta}\right)$, with probability at least $1 - \delta$, $\mathcal{R}_{\mathrm{Adv},2\rho}(f; \mathcal{D}) \geq c_1$.*

The goal is to find a relatively small set ζ that satisfies the condition as this will mean that even for modest sample sizes, the trained models have significant adversarial error. We remark that it is easy to construct concrete instantiations of problems that satisfy the conditions of the theorem, e.g. each class represented by a spherical (truncated) Gaussian with radius ρ, with the classes being well-separated satisfies Eq. (1). The main idea of the proof is that there is sufficient probability mass for points which are within distance 2ρ of a training datum that was mislabelled. We note that the generality of the result, namely that *any* classifier (including neural networks) that fits the training data must be vulnerable irrespective of its structure, requires a result like Theorem 1. For instance, one could construct the classifier h, where $h(\mathbf{x}) = c(\mathbf{x})$, if $(\mathbf{x}, b) \notin \mathcal{S}_m$ for $b = 0, 1$, and $h(\mathbf{x}) = y$ if $(\mathbf{x}, y) \in \mathcal{S}_m$. Note that the classifier h agrees with the target c on *every* point of $\mathbb{R}^d$ except the mislabelled training examples, and as a result these examples are the only source of vulnerability. The complete proof is presented in Appendix B.1.

There are a few things to note about Theorem 1. First, the lower bound on adversarial error applies to any classifier f that fits the training data $\mathcal{S}_m$ perfectly and is agnostic to the type of model f is. Second, for a given c_1, there maybe multiple ζs that satisfy the bounds in (1) and the adversarial risk holds for all of them. Thus, smaller the value of $|\zeta|$ the smaller the size of the training data it needs to fit and it can be done by simpler classifiers. Third, if the distribution of the data is such that it is concentrated around some points then for a fixed c_1, c_2, a smaller value of ρ would be required to satisfy (1) and thus a weaker adversary (smaller perturbation budget 2ρ) can cause a much larger adversarial error.



(a) Toy-MNIST , $\epsilon = \frac{64}{255}$

(b) Full-MNIST

(c) ResNet18 (CIFAR10)

(d) DenseNet121 (CIFAR10)

(e) VGG19 (CIFAR10)

Figure 2: Adversarial Error increases with increasing label noise η. Shaded region indicates 95% confidence interval. Absence of shaded region indicates that it is invisible due to low variance.

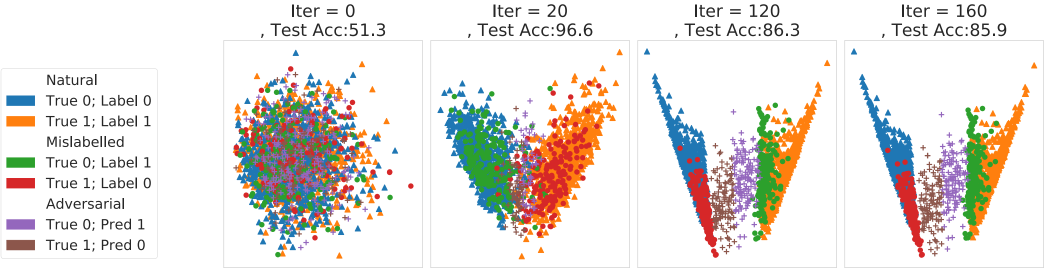


Figure 3: Two dimensional PCA projections of the original correctly labelled (blue and orange), original mis-labelled (green and red), and adversarial examples (purple and brown) at different stages of training. The correct label for *True 0* (blue), *Noisy 0* (green), *Adv 0* (purple +) are the same i.e. 0 and similar for the other class.

In practice, classifiers exhibit much greater vulnerability than purely arising from the presence of memorized noisy data. Experiments in Section 3.1 show how label noise causes vulnerability in a toy MNIST model, the full MNIST and CIFAR10 for a variety of architectures.

3 Experiments on Overfitting Label Noise

In Section 2, we provided theoretical settings to highlight how fitting label noise hurts adversarial robustness. In this section, we provide empirical evidence on synthetic data inspired by the theory and on the standard datasets: MNIST [20] and CIFAR10 [19].

3.1 Overfitting label noise decreases adversarial accuracy

We design a simple binary classification problem, *toy-MNIST*, and show that when fitting a complex classifier on a training dataset with label noise, adversarial vulnerability increases with the amount of label noise, and that this vulnerability is caused by the label noise. The problem is constructed by selecting two random images from MNIST: one “0” and one “1”. Each training/test example is generated by selecting one of these images and adding i.i.d. Gaussian noise sampled from $\mathcal{N}\left(0, \sigma^2\right)$. We create a training dataset of 4000 samples by sampling uniformly from either class. Finally, η fraction of the training data is chosen randomly and its labels are flipped. We train a neural network with four fully connected layers followed by a softmax layer and minimize the cross-entropy loss using an SGD optimizer until the training error becomes zero. Then, we attack this network with a *strong* ℓ_∞ PGD adversary [24] with $\epsilon = \frac{64}{255}$ for 400 steps with a step size of 0.01.

In Figure 2a, we plot the adversarial error, test error and training error as the amount of label noise (η) varies, for three different values of sample variance (σ^2). For low values of σ^2 $(\sigma^2 = 0.1)$, the training data from each class is all concentrated around the same point; as a result these models are unable to memorize the label noise and the training error is high. In this case, over-fitting label noise is impossible and the test error, as well as the adversarial error, is low. However, as σ^2 increases to $\sigma^2 = 0.5$, the neural network is flexible enough to use the "noise component" to extract features that allow it to memorize label noise and fit the training data perfectly. This brings the training error down to zero, while causing the test error to increase, and the adversarial error even more so. This is in line with Theorem 1.

Remark 1. *The case when $\sigma^2 = 0.3$ is particularly interesting; when the label noise is low and the training error is high, there is no overfitting and the test error and the adversarial error is zero. When the network starts memorizing label noise (i.e. train error gets lesser than label noise), test error still remains very low but adversarial error increases rapidly.*

We perform a similar experiment on the full MNIST dataset trained on a 4-layered Convolutional Neural Network. For varying values of η, for a uniformly randomly chosen η fraction of the training data we assigned the class label randomly. We compute the natural test accuracy and the adversarial test accuracy for when the network is attacked with a ℓ_∞ bounded PGD adversary for varying perturbation budget ϵ, with a step size of 0.01 and for 20 steps and plot the results in Figure 2b. We repeat the same experiment for CIFAR10 with a DenseNet121 [17], ResNet18 [14], and VGG19 [33] to test the phenomenon across multiple state of the art architectures and plot the results in Figure 2c to 2e. The results on both datasets show that the effect of over-fitting label noise on adversarial error is even more clearly visible here; for the same PGD adversary the adversarial error jumps sharply with increasing label noise, while the growth of natural test error is much slower. This confirms the hypothesis that benign overfitting may not be so benign when it comes to adversarial error.

For the toy-MNIST problem, we plot a 2-d projection (using PCA) of the learned representations (activations before the last layer) at various stages of training in Figure 3. (We remark that the simplicity of the data model ensures that even a 1-d PCA projection suffices to perfectly separate the classes when there is no label noise; however, the representations learned by a neural network in the presence of noise maybe very different!) We highlight two key observations: (i) The bulk of adversarial examples ("$+$"-es) are concentrated around the mis-labelled training data ("$\circ$"-es) of the opposite class. For example, the purple $+$-es (Adversarially perturbed: True: 0, Pred:1) are very close to the green $\circ$-es (Mislabelled: True:0, Pred: 1). This provides empirical validation for the hypothesis that if there is a mis-labelled data-point in the vicinity that has been fit by the model, an adversarial example is created by moving towards that data point as predicted by Theorem 1. (ii) The mis-labelled training data take longer to be fit by the classifier. For example by iteration 20, the network actually learns a fairly good representation and classification boundary that correctly fits the clean training data (but not the noisy training data). At this stage, the number of adversarial examples are much lower as compared to Iteration 160, by which point the network has completely fit the noisy training data. Thus early stopping helps in avoiding *memorizing* the label noise, and consequently also reduces adversarial vulnerability. Early stopping has indeed been used as a defence in quite a few recent papers in context of adversarial robustness [36, 16], as well as learning in the presence of label-noise [22]. Our work provides an explanation regarding *why* early stopping may reduce adversarial vulnerability by avoiding fitting noisy training data.

4 Conclusion

Recent research has largely shone a positive light on interpolation (zero training error) by highly over-parameterized models even in the presence of label noise. While overfitting noisy data may not harm generalisation, we have shown that this can be severely detrimental to robustness. This raises a new security threat where label noise can be inserted into datasets to make the models learnt from them vulnerable to adversarial attacks without hurting their test accuracy. As a result, a) further research into learning without memorization is ever more important [30, 32], b) Importance of integrity of dataset curation processes is ever more important to prevent the injection of said noise especially when they will be used with deep neural networks.

References

[1] Athalye, A., Carlini, N., and Wagner, D. (2018). Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*.

[2] Bartlett, P. L., Long, P. M., Lugosi, G., and Tsigler, A. (2020). Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, page 201907378.

[3] Belkin, M., Hsu, D. J., and Mitra, P. (2018a). Overfitting or perfect fitting? risk bounds for classification and regression rules that interpolate. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 2300–2311. Curran Associates, Inc.

[4] Belkin, M., Ma, S., and Mandal, S. (2018b). To understand deep learning we need to understand kernel learning. In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 541–549, Stockholmsmässan, Stockholm Sweden. PMLR.

[5] Belkin, M., Rakhlin, A., and Tsybakov, A. B. (2019a). Does data interpolation contradict statistical optimality? In K. Chaudhuri and M. Sugiyama, editors, *Proceedings of Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, pages 1611–1619. PMLR.

[6] Belkin, M., Hsu, D., and Xu, J. (2019b). Two models of double descent for weak features. *arXiv:1903.07571*.

[7] Biggio, B. and Roli, F. (2018). Wild patterns. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. ACM.

[8] Chatterji, N. S. and Long, P. M. (2020). Finite-sample analysis of interpolating linear classifiers in the overparameterized regime. *arXiv:2004.12019*.

[9] Dalvi, N., Domingos, P., Mausam, Sanghai, S., and Verma, D. (2004). Adversarial classification. In *Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining - KDD2004*. ACM Press.

[10] Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., and Wichmann, F. A. (????). Shortcut learning in deep neural networks.

[11] Goodfellow, I. J., Shlens, J., and Szegedy, C. (2014). Explaining and Harnessing Adversarial Examples. *arXiv preprint arXiv:1412.6572*.

[12] Hanin, B. and Rolnick, D. (2019). Complexity of linear regions in deep networks. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2596–2604, Long Beach, California, USA. PMLR.

[13] Hastie, T., Montanari, A., Rosset, S., and Tibshirani, R. J. (2019). Surprises in high-dimensional ridgeless least squares interpolation. *arXiv:1903.08560*.

[14] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778. IEEE.

[15] Hendrycks, D., Zhao, K., Basart, S., Steinhardt, J., and Song, D. (2019a). Natural adversarial examples. *arXiv:1907.07174*.

[16] Hendrycks, D., Lee, K., and Mazeika, M. (2019b). Using pre-training can improve model robustness and uncertainty. *Proceedings of the International Conference on Machine Learning*.

[17] Huang, G., Liu, Z., Weinberger, K. Q., and van der Maaten, L. (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, volume 1, page 3.

[18] Jacobsen, J.-H., Behrmann, J., Zemel, R., and Bethge, M. (2019). Excessive invariance causes adversarial vulnerability. In *International Conference on Learning Representations*.

[19] Krizhevsky, A. and Hinton, G. (2009). Learning multiple layers of features from tiny images. Technical report, Citeseer.

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

[21] Li, J., Schmidt, F., and Kolter, Z. (2019a). Adversarial camera stickers: A physical camera-based attack on deep learning systems. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3896–3904, Long Beach, California, USA. PMLR.

[22] Li, M., Soltanolkotabi, M., and Oymak, S. (2019b). Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks. *arXiv:1903.11680*.

[23] Liang, T. and Rakhlin, A. (2018). Just interpolate: Kernel "ridgeless" regression can generalize. *arXiv:1808.00387*.

[24] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*.

[25] Montasser, O., Hanneke, S., and Srebro, N. (2019). Vc classes are adversarially robustly learnable, but only improperly. In A. Beygelzimer and D. Hsu, editors, *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pages 2512–2530, Phoenix, USA. PMLR.

[26] Muthukumar, V., Vodrahalli, K., Subramanian, V., and Sahai, A. (2020). Harmless interpolation of noisy data in regression. *IEEE Journal on Selected Areas in Information Theory*, pages 1–1.

[27] Papernot, N., McDaniel, P., Wu, X., Jha, S., and Swami, A. (2015). Distillation as a defense to adversarial perturbations against deep neural networks. *arXiv:1511.04508*.

[28] Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., and Swami, A. (2017). Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*. ACM.

[29] Sanyal, A., Dokania, P., Kanade, V., and Torr, P. H. (2020a). Robustness via deep low-rank representations. arxiv:1804.07090.

[30] Sanyal, A., Torr, P. H., and Dokania, P. K. (2020b). Stable rank normalization for improved generalization in neural networks and {gan}s. In *International Conference on Learning Representations*.

[31] Schönherr, L., Kohls, K., Zeiler, S., Holz, T., and Kolossa, D. (2018). Adversarial attacks against automatic speech recognition systems via psychoacoustic hiding. *arXiv:1808.05665*.

[32] Shen, Y. and Sanghavi, S. (2019). Learning with bad training data via iterative trimmed loss minimization. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5739–5748, Long Beach, California, USA. PMLR.

[33] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[34] Tramer, F., Carlini, N., Brendel, W., and Madry, A. (2020). On adaptive attacks to adversarial example defenses. *arXiv:2002.08347*.

[35] Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., and McDaniel, P. (2018). Ensemble adversarial training: Attacks and defenses. In *International Conference on Learning Representations*.

[36] Wong, E., Rice, L., and Kolter, J. Z. (2020). Fast is better than free: Revisiting adversarial training. In *International Conference on Learning Representations*.

[37] Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2016). Understanding deep learning requires rethinking generalization. *International Conference on Learning Representations (ICLR)*.

[38] Zhang, H., Yu, Y., Jiao, J., Xing, E. P., Ghaoui, L. E., and Jordan, M. I. (2019). Theoretically principled trade-off between robustness and accuracy. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 7472–7482. PMLR.

A Related Work

[25] established that there are concept classes with finite VC dimensions i.e. are *properly* PAC-learnable but are only *improperly* robustly PAC learnable. This implies that to learn the problem with small adversarial error, a different class of models (or representations) needs to be used whereas for small natural test risk, the original model class (or representation) can be used. Recent empirical works have also shown evidence towards this (eg. [29]).

Hanin and Rolnick [12] have shown that though the number of possible linear regions that can be created by a deep ReLU network is exponential in depth, in practice for networks trained with SGD this tends to grow only linearly thus creating much simpler decision boundaries than is possible due to sheer expresssivity of deep networks. Experiments on the data models from our theoretical settings indeed show that adversarial training indeed produces more "complex" decision boundaries

Jacobsen *et al.* [18] have discussed that excessive invariance in neural networks might increase adversarial error. However, their argument is that excessive invariance can allow sufficient changes in the semantically important features without changing the network's prediction. They describe this as Invariance-based adversarial examples as opposed to perturbation based adversarial examples. We show that excessive (incorrect) invariance might also result in perturbation based adversarial examples.

Another contemporary work [10] discusses a phenomenon they refer to as *Shortcut Learning* where deep learning models perform very well on standard tasks like reducing classification error but fail to perform in more difficult real world situations. We discuss this in the context of models that have small test error but large adversarial error and provide and theoretical and empirical to discuss why one of the reasons for this is sub-optimal representation learning.

B Proofs and Extral Notations for Section 2

In this section, we present the formal proofs to the theorems stated in Section 2 as well as define the notations that were left undefined.

We formally define the notions of natural (test) error and adversarial error.

Definition 1 (Natural and Adversarial Error)**.** *For any distribution $\mathcal{D}$ defined over $(\mathbf{x}, y) \in \mathbb{R}^d \times \{0,1\}$ and any binary classifier $f : \mathbb{R}^d \to \{0,1\}$,*

- *the* natural *error is*

$$\mathcal{R}(f; \mathcal{D}) = \mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}} \left[f(\mathbf{x}) \neq y \right], \tag{2}$$

- *if $\mathcal{B}_\gamma(\mathbf{x})$ is a ball of radius $\gamma \geq 0$ around $\mathbf{x}$ under some norm*[1]*, the γ-*adversarial *error is*

$$\mathcal{R}_{\mathrm{Adv},\gamma}(f; \mathcal{D}) = \mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}} \left[\exists \mathbf{z} \in \mathcal{B}_\gamma(\mathbf{x}) ; f(\mathbf{z}) \neq y \right], \tag{3}$$

B.1 Proofs of Section 2

Theorem 1. *Let c be the target classifier, and let $\mathcal{D}$ be a distribution over $(\mathbf{x}, y)$, such that $y = c(\mathbf{x})$ in its support. Using the notation $\mathbb{P}_{\mathcal{D}}[A]$ to denote $\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}[\mathbf{x} \in A]$ for any measurable subset $A \subseteq \mathbb{R}^d$, suppose that there exist $c_1 \geq c_2 > 0$, $\rho > 0$, and a finite set $\zeta \subset \mathbb{R}^d$ satisfying*

$$\mathbb{P}_{\mathcal{D}} \left[\bigcup_{\mathbf{s}\in\zeta} \mathcal{B}_\rho^p(\mathbf{s}) \right] \geq c_1 \quad \textit{and} \quad \forall \mathbf{s} \in \zeta,\ \mathbb{P}_{\mathcal{D}} \left[\mathcal{B}_\rho^p(\mathbf{s}) \right] \geq \frac{c_2}{|\zeta|} \tag{1}$$

where $\mathcal{B}_\rho^p(\mathbf{s})$ represents a ℓ_p-ball of radius ρ around $\mathbf{s}$. Further, suppose that each of these balls contain points from a single class i.e. for all $\mathbf{s} \in \zeta$, for all $\mathbf{x}, \mathbf{z} \in \mathcal{B}_\rho^p(\mathbf{s}) : c(\mathbf{x}) = c(\mathbf{z})$.

Let $\mathcal{S}_m$ be a dataset of m i.i.d. samples drawn from $\mathcal{D}$, which subsequently has each label flipped independently with probability η. For any classifier f that perfectly *fits the training data $\mathcal{S}_m$ i.e. $\forall\ \mathbf{x}, y \in \mathcal{S}_m, f(\mathbf{x}) = y$, $\forall \delta > 0$ and $m \geq \frac{|\zeta|}{\eta c_2} \log\left(\frac{|\zeta|}{\delta}\right)$, with probability at least $1 - \delta$, $\mathcal{R}_{\mathrm{Adv},2\rho}(f; \mathcal{D}) \geq c_1$.*

[1] Throughout, we will mostly use the (most commonly used) ℓ_∞ norm, but the results hold for other norms.

Proof of Theorem 1. From (1), for any ζ and $s \in \zeta$,

$$\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\mathbf{x} \in \mathcal{B}_\rho\left(s\right)\right] \geq \frac{c_2}{|\zeta|}$$

As the sampling of the point and the injection of label noise are independent events,

$$\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\mathbf{x} \in \mathcal{B}_\rho\left(s\right) \wedge \mathbf{x} \text{ gets mislabelled}\right] \geq \frac{c_2\eta}{|\zeta|}$$

Thus,

$$\begin{aligned}\mathbb{P}_{\mathcal{S}_m\sim\mathcal{D}^m}\left[\exists\left(\mathbf{x},y\right) \in \mathcal{S}_m : \mathbf{x} \in \mathcal{B}_\rho\left(s\right) \wedge \mathbf{x} \text{ is mislabelled}\right] &\geq 1 - \left(1 - \frac{c_2\eta}{|\zeta|}\right)^m \\ &\geq 1 - \exp\left(\frac{-c_2\eta m}{|\zeta|}\right)\end{aligned}$$

Substituting $m \geq \frac{|\zeta|}{\eta c_2}\log\left(\frac{|\zeta|}{\delta}\right)$ and applying the union bound over all $s \in \zeta$, we get

$$\mathbb{P}_{\mathcal{S}_m\sim\mathcal{D}^m}\left[\forall s \in \zeta,\ \exists\left(\mathbf{x},y\right) \in \mathcal{S}_m : \mathbf{x} \in \mathcal{B}_\rho\left(s\right) \wedge \mathbf{x} \text{ is mislabelled}\right] \geq 1 - \delta \tag{4}$$

As for all $\mathbf{s} \in \mathbb{R}^d$ and $\forall\mathbf{x},\mathbf{z}, \in \mathcal{B}^p_\rho\left(\mathbf{s}\right),\ \left\|\mathbf{x}-\mathbf{z}\right\|_p \leq 2\rho$, we have that

$$\begin{aligned}\mathcal{R}_{\text{Adv},2\rho}(f;\mathcal{D}) &= \mathbb{P}_{\mathcal{S}_m\sim\mathcal{D}^m}\left[\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\exists\mathbf{z} \in \mathcal{B}_{2\rho}\left(\mathbf{x}\right)\ \wedge y \neq f\left(\mathbf{z}\right)\right]\right] \\ &= \mathbb{P}_{\mathcal{S}_m\sim\mathcal{D}^m}\left[\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\exists\mathbf{z} \in \mathcal{B}_{2\rho}\left(\mathbf{x}\right)\ \wedge c\left(\mathbf{z}\right) \neq f\left(\mathbf{z}\right)\right]\right] \\ &\geq \mathbb{P}_{\mathcal{S}_m\sim\mathcal{D}^n}\left[\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\mathbf{x} \in \bigcup_{s\in\zeta}\mathcal{B}^p_\rho\left(s\right) \wedge \left\{\exists\mathbf{z} \in \mathcal{B}_{2\rho}\left(\mathbf{x}\right) : c\left(\mathbf{z}\right) \neq f\left(\mathbf{z}\right)\right\}\right]\right] \\ &= \mathbb{P}_{\mathcal{S}_m\sim\mathcal{D}^m}\left[\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\exists\mathbf{s} \in \zeta : \mathbf{x} \in \mathcal{B}^p_\rho\left(s\right) \wedge \left\{\exists\mathbf{z} \in \mathcal{B}_\rho\left(\mathbf{s}\right) : c\left(\mathbf{z}\right) \neq f\left(\mathbf{z}\right)\right\}\right]\right] \\ &= \mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\mathbf{x} \in \bigcup_{s\in\zeta}\mathcal{B}^p_\rho\left(s\right)\right] \quad \text{w.p. atleast } 1-\delta \\ &\geq c_1 \quad \text{w.p. } 1-\delta\end{aligned}$$

where c is the true concept for the distribution $\mathcal{D}$. The second equality follows from the assumptions that each of the balls around $\mathbf{s} \in \zeta$ are pure in their labels. The second last equality follows from (4) by using the $\mathbf{x}$ that is guaranteed to exist in the ball around $\mathbf{s}$ and be mis-labelled with probability atleast $1-\delta$. The last equality follows from Assumption (4). □