Choice of Representation Matters for 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

Different components in a machine learning system - for example the choice of model architecture and the learning algorithms are responsible for adding inductive biases in a trained model. For example, standard training procedures bias neural networks towards learning "simple" classification boundaries [29], convolutional networks are location invariant [18], and MLPs ignore permutation in the input space among others. These biases manifest themselves in the representations of raw data learned by the model. While some previous work have showed the importance of invariances for purposes of generalization and optimization, in this work, we show the importance of proper inductive biases for adversarial robustness. 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. This has largely been made possible by incorporating proper inductive biases; and that has helped achieve better generalization and accelerate optimization. Convolutional Layers [16] and Recurrent Layers [7] provides domain-dependant inductive biases by introducing translation invariance and temporal invariance respectively. Inductive Biases in Batch Normalization [11] heavily sped up optimization of deep networks. The positional embeddings in transformer architectures [34] allow the model to encode absolute and relative positions and positionally invariant relationships. In Language models, using Byte Pair Encoding [28, 33] for words reflects the inductive bias that words composed of similar subwords are related. Data augmentation, for example in the form of random cropping, has long been used to achieve better generalization behaviour in computer vision [15]

However, especially in vision tasks, they have been shown to be highly vulnerable to small adversarial perturbations that are imperceptible to the human eye [3, 2, 6] . This vulnerability poses serious security concerns when these models are deployed in real-world tasks (cf. [24, 27, 10, 17]). A large body of research has been devoted to crafting defences to protect neural networks from adversarial attacks (e.g. [6, 23, 31, 19, 38]). However, such defences have usually been broken by future attacks [1, 30]. 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. Our goal in this paper is not to propose new defenses, but to provide better answers to the question: what causes adversarial vulnerability? In particular, we look at *improper* representation learning by way of incorrect inductive biases as a source of adversarial vulnerability.

1st NeurIPS workshop on Interpretable Inductive Biases and Physically Structured Learning (2020), virtual.

Recent works [32, 38] have argued that the trade-off between robustness and accuracy might be unavoidable. However, their setting involves a distribution that is not robustly separable by any classifier. In such a situation there is indeed a trade-off between robustness and accuracy. In this paper, we focus on settings where robust classifiers exist, which is a more realistic scenario for real-world data. At least for vision, one may well argue that "humans" are robust classifiers, and as a result we would expect that classes are well-separated at least in some representation space. In fact, Yang *et al.* [35] show that classes are already well-separated in the input space. In such situations, there is no need for robustness to be at odds with accuracy. A more plausible scenario which we posit, and provide theoretical evidence in support of in Theorem 2, is that depending on the choice of representations, the trade-off may exist or can be avoided. Recent empirical work [25, 20] has also established that modifying the training objective to favour certain inductive biass in the learned representations can automatically lead to improved robustness.

On a related note, it has been suggested in recent works that adversarially robust learning may require more "complex" decision boundaries, and as a result may require more data [29, 26, 36, 19]. However, the question of decision boundaries in neural networks is subtle as the network learns a *feature representation* as well as a decision boundary on top of it. We develop concrete theoretical examples in Theorem 1 and 2 to establish that choosing one feature representation over another may lead to *visually* more complex decision boundaries on the input space, though these are not necessarily more complex in terms of statistical learning theoretic concepts such as VC dimension.

Summary of Contributions

1. The choice of the representation (and hence the shape of the decision boundary) can be important for adversarial accuracy even when it doesn't affect natural test accuracy. (See Theorem 1)
2. There exists data distributions and training algorithms, which when trained with (some fraction of) random label noise have the following property: (i) using one representation, it is possible to have high natural and robust test accuracies but at the cost of having training error; (ii) using another representation, it is possible to have no training error (including fitting noise) and high test accuracy, but low robust accuracy. (See Theorem 2).
3. To demonstrate the benefit of representation learning for adversarial robustness, we show that learning richer representation by training with more fine-grained labels, subclasses within each class, leads to higher robust accuracy.

2 Representation Learning and Adversarial Robustness

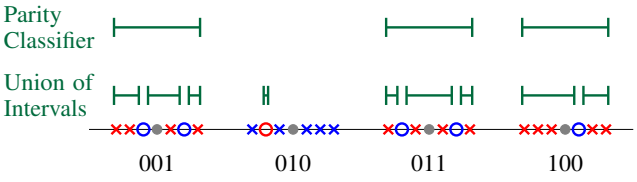
The choice of inductive biases incorporated in a model affects representations and introduces desirable and possibly even undesirable (cf. [18]) invariances; for example, training convolutional networks are invariant to (some) translations, while training fully connected networks are invariant to permutations of input features. This means that fully connected networks can learn even if the pixels of each training image in the training set are permuted with a fixed permutation [37]. This invariance is worrying as it means that such a network can effectively classify a matrix (or tensor) that is visually nothing like a real image into an image category.

In this section we present a result to show that there exists a data distribution where proper representation is necessary for small adversarial error as well as small test error whereas another representation can provide low test error but necessarily have large adversarial error. Interestingly, the representation that can achieve small adversarial error can look visually more complex due to larger number of distinct linear regions in its decision boundary. However, statistically it will have a smaller VC dimension than its counterpart.We first present the theorems with a proof sketch for ease of understanding and the more detailed proofs in Appendix B.

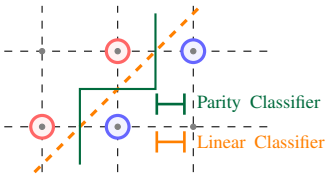
Theorem 1. *For some universal constant c, and any $0 < \gamma_0 < 1/\sqrt{2}$, there exists a family of distributions $\mathcal{D}$ defined on $\mathcal{X} \times \{0, 1\}$ where $\mathcal{X} \subseteq \mathbb{R}^2$ such that for all distributions $\mathcal{P} \in \mathcal{D}$, and denoting by $\mathcal{S}_m = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_m, y_m)\}$ a sample of size m drawn i.i.d. from $\mathcal{P}$,*

(i) For any $m \geq 0$, $\mathcal{S}_m$ is linearly separable i.e., $\forall (\mathbf{x}_i, y_i) \in \mathcal{S}_m$, there exist $\mathbf{w} \in \mathbb{R}^2, w_0 \in \mathbb{R}$ s.t. $y_i \left(\mathbf{w}^\top \mathbf{x}_i + w_0\right) \geq 0$. Furthermore, for every $\gamma > \gamma_0$, any linear separator f that perfectly fits the training data $\mathcal{S}_m$ has $\mathcal{R}_{\mathrm{Adv},\gamma}(f; \mathcal{P}) \geq 0.0005$, even though $\mathcal{R}(f; \mathcal{P}) \to 0$ as $m \to \infty$.

(ii) There exists a function class $\mathcal{H}$ such that for some $m \in O(\log(\delta^{-1}))$, any $h \in \mathcal{H}$ that perfectly fits the $\mathcal{S}_m$, satisfies with probability at least $1 - \delta$, $\mathcal{R}(h; \mathcal{P}) = 0$ and $\mathcal{R}_{\mathrm{Adv},\gamma}(h; \mathcal{P}) = 0$, for any $\gamma \in [0, \gamma_0 + 1/8]$.



(a) Both Parity and Union of Interval classifier predicts red if inside any green interval and blue if outside all intervals. The ×-es are correctly labelled and the ○-es are mis-labelled points. Reference integer points on the line labelled in *binary*.



(b) Robust generalization needs more complex boundaries

Figure 1: Visualization of the distribution and classifiers used in the Proof of Theorem 1 and 2. The Red and Blue indicate the two classes.

A complete proof of this result appears in Appendix B.1, but first, we provide a sketch of the key idea here.The distributions in family $\mathcal{D}$ will be supported on balls of radius at most $1/\sqrt{2}$ on the integer lattice in $\mathbb{R}^2$. The *true* class label for any point $\mathbf{x}$ is provided by the parity of $a+b$, where (a,b) is the lattice point closest to $\mathbf{x}$. However, the distributions in $\mathcal{D}$ are chosen to be such that there is also a linear classifier that can separate these classes, e.g. a distribution only supported on balls centered at the points (a,a) and $(a,a+1)$ for some integer a (See Figure 1b). *Visually* learning the classification problem using the parity of $a+b$ results in a seemingly more complex decision boundary, a point that has been made earlier regarding the need for more complex boundaries to achieve adversarial robustness [22, 4]. However, it is worth noting that this complexity is not rooted in any *statistical theory*, e.g. the VC dimension of the classes considered in Theorem 1 is essentially the same (even lower for $\mathcal{H}$ by 1). This *visual* complexity arises purely due to the fact that the linear classifier looks at a geometric representation of the data whereas the parity classifier looks at the binary representation of the sum of the nearest integer of the coordinates. In the case of neural networks, recent works [13] have indeed provided empirical results to support that excessive invariance (eg. rotation invariance) increases adversarial error.

In this section, we show how the choice of representation is important in the presence of label noise to learn an adversarially robust classifier. Informally, we show that if the *correct* representation is used, then in the presence of label noise, it will be impossible to fit the training data perfectly, but the classifier that best fits the training data,will have good test accuracy and adversarial accuracy. However, using an "incorrect" representation, we show that it is possible to find a classifier that has no training error, has good test accuracy, but has high *adversarial error*. We posit this as an (partial) explanation of why classifiers trained on real data (with label noise) have good test accuracy, while still being vulnerable to adversarial attacks.

Theorem 2. *For any $n \in \mathbb{Z}_+$, there exists a family of distributions $\mathcal{D}^n$ over $\mathbb{R} \times \{0,1\}$ and function classes $\mathcal{C}, \mathcal{H}$, such that for any $\mathcal{P}$ from $\mathcal{D}^n$, and for any $0 < \gamma < 1/4$, and $\eta \in (0, 1/2)$ if $\mathcal{S}_m = \{(\mathbf{x}_i, y_i)\}_{i=1}^m$ denotes a sample of size m drawn from $\mathcal{P}$ where*

$$m = O\left(\max\left\{n \log \frac{n}{\delta}\left(\frac{(1-\eta)}{(1-2\eta)^2} + 1\right), \frac{n}{\eta\gamma^2}\log\left(\frac{n}{\gamma\delta}\right)\right\}\right)$$

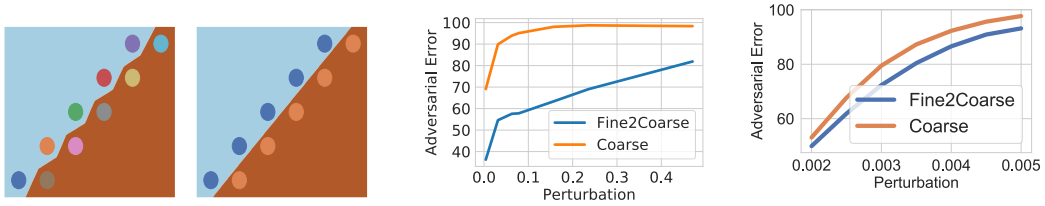
and if $\mathcal{S}_{m,\eta}$ denotes the sample where each label is flipped independently with probability η.

(i) the classifier $c \in \mathcal{C}$ that minimizes the training error on $\mathcal{S}_{m,\eta}$, has $\mathcal{R}(c;\mathcal{P}) = 0$ and $\mathcal{R}_{\mathrm{Adv},\gamma}(c;\mathcal{P}) = 0$ for $0 \le \gamma < 1/4$.

(ii) there exist $h \in \mathcal{H}$, h has zero training error on $\mathcal{S}_{m,\eta}$, and $\mathcal{R}(h;\mathcal{P}) = 0$. However, for any $\gamma > 0$, and for any $h \in \mathcal{H}$ with zero training error on $\mathcal{S}_{m,\eta}$, $\mathcal{R}_{\mathrm{Adv},\gamma}(h;\mathcal{P}) \ge 0.1$.

Furthermore, the required $c, h \in \mathcal{C}, \mathcal{H}$ can be computed in $O\left(\mathrm{poly}\left(n\right), \mathrm{poly}\left(\frac{1}{\frac{1}{2}-\eta}\right), \mathrm{poly}\left(\frac{1}{\delta}\right)\right)$.

We sketch the proof here and present the complete the proof in Appendix B.1; as in Theorem 1 we will make use of parity functions, though the key point is the representations used. Let $\mathcal{X} = [0, N]$, where $N = 2^n$, we consider distributions that are supported on intervals $(i - 1/4, i + 1/4)$ for $i \in \{1, \ldots, N-1\}$ (See Figure 1a), but any such distribution will only have a small number, $O(n)$,



(a) MULTICLASS (b) NATURAL (c) CIFAR-100. (d) Restricted Imagenet

Figure 2: Assigning a separate class to each sub-population within the original class during training increases robustness by learning more meaningful representations.

intervals on which it is supported. The *true* class label is given by a function that depends on the parity of some hidden subsets S of bits in the bit-representation of the closest integer i, e.g. as in Figure 1a if $S = \{0, 2\}$, then only the least significant and the third least significant bit of i are examined and the class label is 1 if an odd number of them are 1 and 0 otherwise. Despite the noise, the *correct* label on any interval can be guessed by using the majority vote and as a result, the correct parity learnt using Gaussian elimination. (This corresponds to the class $\mathcal{C}$ in Theorem 2.) On the other hand it is also possible to learn the function as a union of intervals, i.e. find intervals, $I_1, I_2, \ldots, I_k$ such that any point that lies in one of these intervals is given the label 1 and any other point is given the label 0. By choosing intervals carefully, it is possible to fit *all the training data*, including noisy examples, but yet not compromise on *test accuracy* (Fig. 1a). Such a classifier, however, will be vulnerable to adversarial examples by applying Theorem 3. A classifier such as union of intervals ($\mathcal{H}$ in Theorem 2) is translation-invariant, whereas the parity classifier is not. This suggests that using classifiers, such as neural networks, that are designed to have too many built-in invariances might hurt its robustness accuracy. In Theorem 2, we present further experimental evidence that neural networks trained with SGD learn more linear-like (simpler) decision boundaries than is necessary for obtaining adversarial robustness.

3 Accounting for fine grained sub-populations leads to better robustness

One way to evaluate whether more meaningful representations lead to better robust accuracy is to use training data with more fine-grained labels (e.g. subclasses of a class); for example, one would expect that if different breeds of dogs are labelled differently the network will learn features that are relevant to that extra information. We show both using synthetic data, CIFAR100 [14], and Restricted Imagenet [32] that training on fine-grained labels does increase robust accuracy.

We hypothesize that learning more meaningful representations by accounting for fine-grained sub-populations within each class may lead to better robustness. We use the theoretical setup presented in Figure 1b. However, if each of the circles belonged to a separate class then the decision boundary would have to be necessarily more complex as it needs to, now, separate the balls that were previously within the same class. We test this hypothesis with two experiments. First, we test it on the the distribution defined in Theorem 1 where for each ball with label 1, we assign it a different label (say $\alpha_1, \cdots, \alpha_k$) and similarly for balls with label 0, we assign it a different label ($\beta_1, \cdots, \beta_k$). Now, we solve a multi-class classification problem for $2k$ classes with a deep neural network and then later aggregate the results by reporting all α_is as 1 and all β_is as 0.The resulting decision boundary is drawn in Figure 2a along with the decision boundary for natural training. Clearly, the margin for the multi-class model (and thus robustness) is greater than the naturally trained model.

Second, we also repeat the experiment with CIFAR-100 and Restricted Imagenet [32]. For CIFAR-100, we train a ResNet50 [9] on the fine labels of CIFAR100 and then aggregate the fine labels corresponding to a coarse label by summing up the logits of the fine classes corresponding to each coarse class. For restricted imagenet, we use the fine-coarse division mentioned in Table 1. We call this model the *Fine2Coarse* model and compare the adversarial risk of this network to a ResNet-50 trained directly on the coarse labels. Note that the model is end-to-end differentiable as the only addition is a layer to aggregate the logits corresponding to the fine classes pertaining to each coarse class. Thus PGD adversarial attacks can be applied out of the box. Figure 2c shows that for all perturbation budgets, *Fine2Coarse* has smaller adversarial risk than the naturally trained model.

4 Acknowledgement

We thank Vitaly Feldman and Chiyuan Zhang for providing us with data that helped to significantly speed up some parts of this work. We also thank Nicholas Lord for feedback on the draft. AS acknowledges support from The Alan Turing Institute under the Turing Doctoral Studentship grant TU/C/000023. VK is supported in part by the Alan Turing Institute under the EPSRC grant EP/N510129/1. PHS and PD are supported by the ERC grant ERC-2012-AdG 321162-HELIOS, EPSRC grant Seebibyte EP/M013774/1 and EPSRC/MURI grant EP/N019474/1. PHS and PD also acknowledges the Royal Academy of Engineering and FiveAI.

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

[3] 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.

[4] Degwekar, A., Nakkiran, P., and Vaikuntanathan, V. (2019). Computational limitations in robust classification and win-win results. 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 994–1028, Phoenix, USA. PMLR.

[5] Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., and Wichmann, F. A. (2015). Shortcut learning in deep neural networks.

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

[7] Graves, A., Mohamed, A.-r., and Hinton, G. (2013). Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. IEEE.

[8] 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.

[9] 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.

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

[11] Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift.

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

[13] Kamath, S., Deshpande, A., and Subrahmanyam, K. V. (2020). Invariance vs robustness of neural networks.

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

[15] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Conference on Neural Information Processing Systems*, pages 1097–1105.

[16] 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.

[17] Li, J., Schmidt, F., and Kolter, Z. (2019). 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.

[18] Liu, R., Lehman, J., Molino, P., Such, F. P., Frank, E., Sergeev, A., and Yosinski, J. (2018). An intriguing failing of convolutional neural networks and the coordconv solution. In S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*, pages 9628–9639.

[19] 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*.

[20] Mao, C., Gupta, A., Nitin, V., Ray, B., Song, S., Yang, J., and Vondrick, C. (2020). Multitask learning strengthens adversarial robustness.

[21] 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.

[22] Nakkiran, P. (2019). Adversarial robustness may be at odds with simplicity. *arXiv preprint-arXiv:1901.00532*.

[23] 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*.

[24] 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.

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

[26] Schmidt, L., Santurkar, S., Tsipras, D., Talwar, K., and Madry, A. (2018). Adversarially robust generalization requires more data. In *Conference on Neural Information Processing Systems*, pages 5014–5026.

[27] 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*.

[28] Sennrich, R., Haddow, B., and Birch, A. (2015). Neural machine translation of rare words with subword units.

[29] Shah, H., Tamuly, K., Raghunathan, A., Jain, P., and Netrapalli, P. (2020). The pitfalls of simplicity bias in neural networks. *Advances in Neural Information Processing Systems*, **33**.

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

[31] 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*.

[32] Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., and Madry, A. (2019). Robustness may be at odds with accuracy. In *International Conference on Learning Representations*.

[33] van Merriënboer, B., Sanyal, A., Larochelle, H., and Bengio, Y. (2017). Multiscale sequence modeling with a learned dictionary.

[34] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Conference on Neural Information Processing Systems*, pages 5998–6008.

[35] Yang, Y.-Y., Rashtchian, C., Zhang, H., Salakhutdinov, R., and Chaudhuri, K. (2020). Adversarial robustness through local lipschitzness. *arXiv:2003.02460*.

[36] Yin, D., Kannan, R., and Bartlett, P. (2019). Rademacher complexity for adversarially robust generalization. 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 7085–7094, Long Beach, California, USA. PMLR.

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

[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

[21] 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. [25]).

Hanin and Rolnick [8] 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.* [12] 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 [5] 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*
$$\mathcal{R}(f; \mathcal{D}) = \mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}\left[f\left(\mathbf{x}\right) \neq y\right], \tag{1}$$
- *if $\mathcal{B}_\gamma\left(\mathbf{x}\right)$ is a ball of radius $\gamma \geq 0$ around $\mathbf{x}$ under some norm*[1]*, the* γ-adversarial *error is*
$$\mathcal{R}_{\mathrm{Adv},\gamma}(f; \mathcal{D}) = \mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\exists \mathbf{z} \in \mathcal{B}_\gamma\left(\mathbf{x}\right); f\left(\mathbf{z}\right) \neq y\right], \tag{2}$$

B.1 Proofs of Section 2

Theorem 1. *For some universal constant c, and any $0 < \gamma_0 < 1/\sqrt{2}$, there exists a family of distributions $\mathcal{D}$ defined on $\mathcal{X} \times \{0,1\}$ where $\mathcal{X} \subseteq \mathbb{R}^2$ such that for all distributions $\mathcal{P} \in \mathcal{D}$, and denoting by $\mathcal{S}_m = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_m, y_m)\}$ a sample of size m drawn i.i.d. from $\mathcal{P}$,*

(i) For any $m \geq 0$, $\mathcal{S}_m$ is linearly separable i.e., $\forall (\mathbf{x}_i, y_i) \in \mathcal{S}_m$, there exist $\mathbf{w} \in \mathbb{R}^2, w_0 \in \mathbb{R}$ s.t. $y_i\left(\mathbf{w}^\top \mathbf{x}_i + w_0\right) \geq 0$. Furthermore, for every $\gamma > \gamma_0$, any linear separator f that perfectly fits the training data $\mathcal{S}_m$ has $\mathcal{R}_{\mathrm{Adv},\gamma}(f; \mathcal{P}) \geq 0.0005$, even though $\mathcal{R}(f; \mathcal{P}) \to 0$ as $m \to \infty$.

(ii) There exists a function class $\mathcal{H}$ such that for some $m \in O(\log(\delta^{-1}))$, any $h \in \mathcal{H}$ that perfectly fits the $\mathcal{S}_m$, satisfies with probability at least $1 - \delta$, $\mathcal{R}(h; \mathcal{P}) = 0$ and $\mathcal{R}_{\mathrm{Adv},\gamma}(h; \mathcal{P}) = 0$, for any $\gamma \in [0, \gamma_0 + 1/8]$.

Proof of Theorem 1. We define a family of distribution $\mathcal{D}$, such that each distribution in $\mathcal{D}$ is supported on balls of radius r around (i, i) and $(i+1, i)$ for positive integers i. Either all the balls

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

around (i, i) have the labels 1 and the balls around $(i+1, i)$ have the label 0 or vice versa. Figure 1b shows an example where the colors indicate the label.

Formally, for $r > 0$, $k \in \mathbb{Z}_+$, the (r, k)-1 bit parity class conditional model is defined over $(x, y) \in \mathbb{R}^2 \times \{0, 1\}$ as follows. First, a label y is sampled uniformly from $\{0, 1\}$, then and integer i is sampled uniformly from the set $\{1, \cdots, k\}$ and finally $\mathbf{x}$ is generated by sampling uniformly from the ℓ_2 ball of radius r around $(i + y, i)$.

In Lemma 1 we first show that a set of m points sampled iid from any distribution as defined above for $r < \frac{1}{2\sqrt{2}}$ is with probability 1 linear separable for any m. In addition, standard VC bounds show that any linear classifier that separates S_m for large enough m will have small test error. Lemma 1 also proves that there exists a range of γ, r such that for any distribution defined with r in that range, though it is possible to obtain a linear classifier with 0 training and test error, the minimum adversarial risk will be bounded from 0.

However while it is possible to obtain a linear classifier with 0 test error, all such linear classifiers has a large adversarial vulnerability. In Lemma 2, we show that there exists a different representation for this problem, which also achieves zero training and test error and in addition has zero adversarial risk for a range of r, γ where the linear classifier's adversarial error was atleast a constant.

□

Lemma 1 (Linear Classifier)**.** *There exists universal constants γ_0, ρ, such that for any perturbation $\gamma > \gamma_0$, radius $r \geq \rho$, and $k \in \mathbb{Z}_+$, the following holds. Let $\mathcal{D}$ be the family of (r, k)- 1-bit parity class conditional model, $\mathcal{P} \in \mathcal{D}$ and $\mathcal{S}_n = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_1)\}$ be a set of n points sampled i.i.d. from $\mathcal{P}$.*

1) *For any $n > 0$, S_n is linearly separable with probability 1 i.e. there exists a $h : (\mathbf{w}, w_0)$, $\mathbf{w} \in \mathbb{R}^2, w_0 \in \mathbb{R}$ such that the linear hyperplane $\mathbf{x} \to \mathbf{w}^\top \mathbf{x} + w_0$ separates $\mathcal{S}_n$ with probability 1:*

$$\forall (\mathbf{x}, y) \in \mathcal{S}_n \quad z\left(\mathbf{w}^\top \mathbf{x} + w_0\right) > 0 \quad \text{where } z = 2y - 1$$

2) *Further there exists an universal constant c such that for any $\epsilon, \delta > 0$ with probability $1 - \delta$ for any $\mathcal{S}_n$ with $n = c\frac{1}{\epsilon^2} \log \frac{1}{\delta}$, any linear classifier $\tilde{h}$ that separates $\mathcal{S}_n$ has $\mathcal{R}(\tilde{h}; \mathcal{P}) \leq \epsilon$.*
3) *Let $h : (\mathbf{w}, w_0)$ be any linear classifier that has $\mathcal{R}(h; \mathcal{P}_P) = 0$. Then, $\mathcal{R}_{\mathrm{Adv},\gamma}(h; \mathcal{P}) > 0.0005$.*

We will prove the first part for any $r < \frac{1}{2\sqrt{2}}$ by constructing a $\mathbf{w}, w_0$ such that it satisfies the constraints of linear separability. Let $\mathbf{w} = (1, -1)$, $w_0 = -0.5$. Consider any point $(\mathbf{x}, y) \in \mathcal{S}_n$ and $z = 2y - 1$. Converting to the polar coordinate system there exists a $\theta \in [0, 2\pi], j \in [0, \cdots, k]$ such that $\mathbf{x} = \left(j + \frac{z+1}{2} + r\cos(\theta), j + r\sin(\theta)\right)$

$$\begin{aligned}
z\left(\mathbf{w}^\top \mathbf{x} + w_0\right) &= z\left(j + \frac{z+1}{2} + r\cos(\theta) - j - r\sin(\theta) - 0.5\right) && \mathbf{w} = (1, -1)^\top \\
&= z\left(\frac{z}{2} + 0.5 + r\cos(\theta) - r\sin(\theta) - 0.5\right) \\
&= \frac{1}{2} + zr(\cos(\theta) - \sin(\theta)) && |\cos(\theta) - \sin(\theta)| < \sqrt{2},\ z \in \{-1, 1\} \\
&> \frac{1}{2} - r\sqrt{2} \\
&> 0 && r < \frac{1}{2\sqrt{2}}
\end{aligned}$$

Part 2 follows with simple VC bounds of linear classifiers.

Let the universal constants γ_0, ρ be 0.02 and $\frac{1}{2\sqrt{2}} - 0.008$ respectively. Note that there is nothing special about this constants except that *some* constant is required to bound the adversarial risk away from 0. Now, consider a distribution $\mathcal{P}$ 1-bit parity model such that the radius of each ball is atleast ρ. This is smaller than $\frac{1}{2\sqrt{2}}$ and thus satisfies the linear separability criterion.

Consider h to be a hyper-plane that has 0 test error. Let the ℓ_2 radius of adversarial perturbation be $\gamma > \gamma_0$. The region of each circle that will be vulnerable to the attack will be a circular segment with

the chord of the segment parallel to the hyper-plane. Let the minimum height of all such circular segments be r_0. Thus, $\mathcal{R}_{\mathrm{Adv},\gamma}(h;\mathcal{P})$ is greater than the mass of the circular segment of radius r_0. Let the radius of each ball in the support of $\mathcal{P}$ be r.

Using the fact that h has zero test error; and thus classifies the balls in the support of $\mathcal{P}$ correctly and simple geometry

$$\begin{aligned}\frac{1}{\sqrt{2}} &\geq r + (\gamma - r_0) + r \\ r_0 &\geq 2r + \gamma - \frac{1}{\sqrt{2}}\end{aligned} \tag{3}$$

To compute $\mathcal{R}_{\mathrm{Adv},\gamma}(h;\mathcal{P})$ we need to compute the ratio of the area of a circular segment of height r_0 of a circle of radius r to the area of the circle. The ratio can be written

$$A\left(\frac{r_0}{r}\right) = \frac{cos^{-1}\left(1-\frac{r_0}{r}\right) - \left(1-\frac{r_0}{r}\right)\sqrt{2\frac{r_0}{r} - \frac{r_0^2}{r^2}}}{\pi} \tag{4}$$

As (4) is increasing with $\frac{r_0}{r}$, we can evaluate

$$\begin{aligned}\frac{r_0}{r} &\geq \frac{2r - \frac{1}{\sqrt{2}} + \gamma}{r} && \text{Using (3)} \\ &\geq 2 - \frac{\frac{1}{\sqrt{2}} - 0.02}{r} && \gamma > \gamma_0 = 0.02 \\ &\geq 2 - \frac{\frac{1}{\sqrt{2}} - 0.02}{\frac{1}{\sqrt{2}} - 0.008} > 0.01 && r > \rho = \frac{1}{2\sqrt{2}} - 0.008\end{aligned}$$

Substituting $\frac{r_0}{r} > 0.01$ into Eq. (4), we get that $A\left(\frac{r_0}{r}\right) > 0.0005$. Thus, for all $\gamma > 0.02$, we have $\mathcal{R}_{\mathrm{Adv},\gamma}(h;\mathcal{P}) > 0.0005$.

Lemma 2 (Robustness of parity classifier)**.** *There exists a concept class $\mathcal{H}$ such that for any $\gamma \in \left[\gamma_0, \gamma_0 + \frac{1}{8}\right]$, $k \in \mathbb{Z}_+$, $\mathcal{P}$ being the corresponding (ρ, k) 1-bit parity class distribution where ρ, γ_0 are the same as in Lemma 1 there exists $g \in \mathcal{H}$ such that*

$$\mathcal{R}(g;\mathcal{P}) = 0 \qquad \mathcal{R}_{\mathrm{Adv},\gamma}(g;\mathcal{P}) = 0$$

Proof of Lemma 2. We will again provide a proof by construction. Consider the following class of concepts $\mathcal{H}$ such that $g_b \in \mathcal{H}$ is defined as

$$g\left((x_1, x_2)^\top\right) = \begin{cases} 1 & \text{if } [x_1] + [x_2] = b \,(\mathrm{mod}\ 2) \\ 1 - b & \text{o.w.} \end{cases} \tag{5}$$

where $[x]$ rounds x to the nearest integer and $b \in \{0, 1\}$. In Figure 1b, the green staircase-like classifier belongs to this class. Consider the classifier g_1. Note that by construction $\mathcal{R}(g_1;\mathcal{P}) = 0$. The decision boundary of g_1 that are closest to a ball in the support of $\mathcal{P}$ centered at (a, b) are the lines $x = a \pm 0.5$ and $y = b \pm 0.5$.

As $\gamma < \gamma_0 + \frac{1}{8}$, the adversarial perturbation is upper bounded by $\frac{1}{50} + \frac{1}{8}$. The radius of the ball is upper bounded by $\frac{1}{2\sqrt{2}}$, and as we noted the center of the ball is at a distance of 0.5 from the decision boundary. If the sum of the maximum adversarial perturbation and the maximum radius of the ball is less than the minimum distance of the center of the ball from the decision boundary, then the adversarial error is 0. Substituting the values,

$$\frac{1}{50} + \frac{1}{8} + \frac{1}{2\sqrt{2}} < 0.499 < \frac{1}{2}$$

This completes the proof. □

Theorem 2. *For any $n \in \mathbb{Z}_+$, there exists a family of distributions $\mathcal{D}^n$ over $\mathbb{R} \times \{0,1\}$ and function classes $\mathcal{C}, \mathcal{H}$, such that for any $\mathcal{P}$ from $\mathcal{D}^n$, and for any $0 < \gamma < 1/4$, and $\eta \in (0, 1/2)$ if $\mathcal{S}_m = \{(\mathbf{x}_i, y_i)\}_{i=1}^m$ denotes a sample of size m drawn from $\mathcal{P}$ where*

$$m = O\left(\max\left\{n \log \frac{n}{\delta}\left(\frac{(1-\eta)}{(1-2\eta)^2} + 1\right), \frac{n}{\eta\gamma^2}\log\left(\frac{n}{\gamma\delta}\right)\right\}\right)$$

and if $\mathcal{S}_{m,\eta}$ denotes the sample where each label is flipped independently with probability η.

(i) the classifier $c \in \mathcal{C}$ that minimizes the training error on $\mathcal{S}_{m,\eta}$, has $\mathcal{R}(c;\mathcal{P}) = 0$ and $\mathcal{R}_{\mathrm{Adv},\gamma}(c;\mathcal{P}) = 0$ for $0 \le \gamma < 1/4$.

(ii) there exist $h \in \mathcal{H}$, h has zero training error on $\mathcal{S}_{m,\eta}$, and $\mathcal{R}(h;\mathcal{P}) = 0$. However, for any $\gamma > 0$, and for any $h \in \mathcal{H}$ with zero training error on $\mathcal{S}_{m,\eta}$, $\mathcal{R}_{\mathrm{Adv},\gamma}(h;\mathcal{P}) \ge 0.1$.

Furthermore, the required $c, h \in \mathcal{C}, \mathcal{H}$ can be computed in $O\left(\mathrm{poly}\,(n), \mathrm{poly}\left(\frac{1}{\frac{1}{2}-\eta}\right), \mathrm{poly}\left(\frac{1}{\delta}\right)\right)$.

Proof of Theorem 2. We will provide a constructive proof to this theorem by constructing a distribution $\mathcal{D}$, two concept classes $\mathcal{C}$ and $\mathcal{H}$ and provide the ERM algorithms to learn the concepts and then use Lemma 3 and 4 to complete the proof.

Distribution: Consider the family of distribution $\mathcal{D}^n$ such that $\mathcal{D}_{S,\zeta} \in \mathcal{D}^n$ is defined on $\mathcal{X}_\zeta \times \{0,1\}$ for $S \subseteq \{1, \cdots, n\}, \zeta \subseteq \{1, \cdots, 2^n - 1\}$ such that the support of $\mathcal{X}_\zeta$ is a union of intervals.

$$\mathrm{supp}\,(\mathcal{X})_\zeta = \bigcup_{j \in \zeta} I_j \text{ where } I_j := \left(j - \frac{1}{4}, j + \frac{1}{4}\right) \tag{6}$$

We consider distributions with a relatively small support i.e. where $|\zeta| = O(n)$. Each sample $(\mathbf{x}, y) \sim \mathcal{D}_{S,\zeta}$ is created by sampling $\mathbf{x}$ uniformly from $\mathcal{X}_\zeta$ and assigning $y = c_S(\mathbf{x})$ where $c_S \in \mathcal{C}$ is defined below (7). We define the family of distributions $\mathcal{D} = \bigcup_{n \in \mathbb{Z}_+} \mathcal{D}^n$. Finally, we create $\mathcal{D}^\eta_{S,\zeta}$ -a noisy version of $\mathcal{D}_{S,\zeta}$, by flipping y in each sample (x, y) with probability $\eta < \frac{1}{2}$. Samples from $\mathcal{D}_{S,\zeta}$ can be obtained using the example oracle $\mathrm{EX}\,(\mathcal{D}_{S,\zeta})$ and samples from the noisy distribution can be obtained through the noisy oracle $\mathrm{EX}^\eta\,(\mathcal{D}_{S,\zeta})$

Concept Class $\mathcal{C}$: We define the concept class $\mathcal{C}^n$ of concepts $c_S : [0, 2^n] \to \{0,1\}$ such that

$$c_S(\mathbf{x}) = \begin{cases} 1, & \text{if } (\langle [\mathbf{x}] \rangle_b \text{ XOR } S) \text{ is odd.} \\ 0 & \text{o.w.} \end{cases} \tag{7}$$

where $[\cdot] : \mathbb{R} \to \mathbb{Z}$ rounds a decimal to its nearest integer, $\langle \cdot \rangle_b : \{0, \cdots, 2^n\} \to \{0,1\}^n$ returns the binary encoding of the integer, and $(\langle [\mathbf{x}] \rangle_b \text{ XOR } S) = \sum_{j \in S} \langle [x] \rangle_b [j] \mod 2$. $\langle [x] \rangle_b [j]$ is the j^{th} least significant bit in the binary encoding of the nearest integer to $\mathbf{x}$. It is essentially the class of parity functions defined on the bits corresponding to the indices in S for the binary encoding of the nearest integer to $\mathbf{x}$. For example, as in Figure 1a if $S = \{0, 2\}$, then only the least significant and the third least significant bit of i are examined and the class label is 1 if an odd number of them are 1 and 0 otherwise.

Concept Class $\mathcal{H}$: Finally, we define the concept class $\mathcal{H} = \bigcup_{k=1}^\infty \mathcal{H}_k$ where $\mathcal{H}_k$ is the class of union of k intervals on the real line $\mathcal{H}^k$. Each concept $h_I \in \mathcal{H}^k$ can be written as a set of k disjoint intervals $I = \{I_1, \cdots, I_k\}$ on the real line i.e. for $1 \le j \le k$, $I_j = [a, b]$ where $0 \le a \le b$ and

$$h_I(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in \bigcup_j I_j \\ 0 & \text{o.w.} \end{cases} \tag{8}$$

Now, we look at the algorithms to learn the concepts from $\mathcal{C}$ and $\mathcal{H}$ that minimize the train error. Both of the algorithms will use a majority vote to determine the correct (de-noised) label for each interval, which will be necessary to minimize the test error. The intuition is that if we draw a sufficiently large number of samples, then the majority of samples on each interval will have the correct label with a high probability.

Lemma 3 proves that there exists an algorithm $\mathcal{A}$ such that $\mathcal{A}$ draws $m = O\left(|\zeta|^2 \frac{(1-\eta)}{(1-2\eta)^2} \log \frac{|\zeta|}{\delta}\right)$ samples from the noisy oracle $\mathrm{EX}^{\eta}(\mathcal{D}_{s,\zeta})$ and with probability $1-\delta$ where the probability is over the randomization in the oracle, returns $f \in \mathcal{C}$ such that $\mathcal{R}(f; \mathcal{D}_{S,\zeta}) = 0$ and $\mathcal{R}_{\mathrm{Adv},\gamma}(f; \mathcal{D}_{S,\zeta}) = 0$ for all $\gamma < \frac{1}{4}$. As Lemma 3 states, the algorithm involves gaussian elimination over $|\zeta|$ variables and $|\zeta|$ majority votes (one in each interval) involving a total of m samples. Thus the algorithm runs in $O(\mathrm{poly}(m) + \mathrm{poly}(|\zeta|))$ time. Replacing the complexity of m and the fact that $|\zeta| = O(n)$, the complexity of the algorithm is $O\left(\mathrm{poly}\left(n, \frac{1}{1-2\eta}, \frac{1}{\delta}\right)\right)$.

Lemma 4 proves that there exists an algorithm $\widetilde{A}$ such that $\widetilde{A}$ draws

$$m > \max\left\{2|\zeta|^2 \log\frac{2|\zeta|}{\delta}\left(8\frac{(1-\eta)}{(1-2\eta)^2}+1\right), \frac{0.1|\zeta|}{\eta\gamma^2}\log\left(\frac{0.1|\zeta|}{\gamma\delta}\right)\right\}$$

samples and returns $h \in \mathcal{H}$ such that h has 0 training error, 0 test error and an adversarial test error of atleast 0.1. We can replace $|\zeta| = O(n)$ to get the required bound on m in the theorem. The algorithm to construct h visits every point atmost twice - once during the construction of the intervals using majority voting, and once while accommodating for the mislabelled points. Replacing the complexity of m, the complexity of the algorithm is $O\left(\mathrm{poly}\left(n, \frac{1}{1-2\eta}, \frac{1}{\gamma}, \frac{1}{\delta}\right)\right)$. This completes the proof. □

Lemma 3 (Parity Concept Class)**.** *There exists a learning algorithm $\mathcal{A}$ such that given access to the noisy example oracle $\mathrm{EX}^{\eta}(\mathcal{D}_{S,\zeta})$, $\mathcal{A}$ makes $m = O\left(|\zeta|^2 \frac{(1-\eta)}{(1-2\eta)^2} \log \frac{|\zeta|}{\delta}\right)$ calls to the oracle and returns a hypothesis $f \in \mathcal{C}$ such that with probability $1-\delta$, we have that $\mathcal{R}(f; \mathcal{D}_{S,\zeta}) = 0$ and $\mathcal{R}_{\mathrm{Adv},\gamma}(f; \mathcal{D}_{S,\zeta}) = 0$ for all $\gamma < \frac{1}{4}$.*

Proof. The algorithm $\mathcal{A}$ works as follows. It makes m calls to the oracle $\mathrm{EX}(\mathcal{D}_s^m)$ to obtain a set of points $\{(x_1, y_1), \cdots, (x_m, y_m)\}$ where $m \geq 2|\zeta|^2 \log\frac{2|\zeta|}{\delta}\left(8\frac{(1-\eta)}{(1-2\eta)^2}+1\right)$. Then, it replaces each x_i with $[x_i]$ ($[\cdot]$ rounds a decimal to the nearest integer) and then removes duplicate x_is by preserving the most frequent label y_i associated with each x_i. For example, if $\mathcal{S}_5 = \{(2.8,1), (2.9,0), (3.1,1), (3.2,1), (3.9,0)\}$ then after this operation, we will have $\{(3,1), (4,0)\}$.

As $m \geq 2|\zeta|^2 \log\frac{2|\zeta|}{\delta}\left(8\frac{(1-\eta)}{(1-2\eta)^2}+1\right)$, using $\delta_2 = \frac{\delta}{2}$ and $k = \frac{8(1-\eta)}{(1-2\eta)^2}\log\frac{2|\zeta|}{\delta}$ in Lemma 5 guarantees that with probability $1-\frac{\delta}{2}$, each interval will have atleast $\frac{8(1-\eta)}{(1-2\eta)^2}\log\frac{2|\zeta|}{\delta}$ samples.

Then for any specific interval, using $\delta_1 = \frac{2|\zeta|}{\delta}$ in Lemma 6 guarantees that with probability atleast $1-\frac{2|\zeta|}{\delta}$, the majority vote for the label in that interval will succeed in returning the de-noised label. Applying a union bound over all $|\zeta|$ intervals, will guarantee that with probability atleast $1-\delta$, the majority label of every interval will be the denoised label.

Now, the problem reduces to solving a parity problem on this reduced dataset of $|\zeta|$ points (after denoising, all points in that interval can be reduced to the integer in the interval and the denoised label). We know that there exists a polynomial algorithm using Gaussian Elimination that finds a consistent hypothesis for this problem. We have already guaranteed that there is a point in $\mathcal{S}_m$ from every interval in the support of $\mathcal{D}_{S,\zeta}$. Further, f is consistent on $\mathcal{S}_m$ and f is constant in each of these intervals by design. Thus, with probability atleast $1-\delta$ we have that $\mathcal{R}(f; \mathcal{D}_{S,\zeta}) = 0$.

By construction, f makes a constant prediction on each interval $\left(j-\frac{1}{2}, j+\frac{1}{2}\right)$ for all $j \in \zeta$. Thus, for any perturbation radius $\gamma < \frac{1}{4}$ the adversarial risk $\mathcal{R}_{\mathrm{Adv},\mathcal{D}_{S,\prime\zeta}}(f) = 0$. Combining everything, we have shown that there is an algorithm that makes $2|\zeta|^2 \log\frac{2|\zeta|}{\delta}\left(8\frac{(1-\eta)}{(1-2\eta)^2}+1\right)$ calls to the $\mathrm{EX}\left(\mathcal{D}^{\eta}_{S,\zeta}\right)$ oracle, runs in time polynomial in $|\zeta|, \frac{1}{1-2\eta}, \frac{1}{\delta}$ to return $f \in \mathcal{C}$ such that $\mathcal{R}(f; \mathcal{D}_{S,\zeta}) = 0$ and $\mathcal{R}_{\mathrm{Adv},\gamma}(f; \mathcal{D}_{S,\zeta}) = 0$ for $\gamma < \frac{1}{4}$. □

Lemma 4 (Union of Interval Concept Class)**.** *There exists a learning algorithm $\widetilde{\mathcal{A}}$ such that given access to a noisy example oracle makes $m = O\left(|\zeta|^2 \frac{(1-\eta)}{(1-2\eta)^2} \log \frac{|\zeta|}{\delta}\right)$ calls to the oracle and returns a hypothesis $h \in \mathcal{H}$ such that training error is 0 and with probability $1-\delta$, $\mathcal{R}(f; \mathcal{D}_{S,\zeta}) = 0$.*

Further for any $h \in \mathcal{H}$ that has zero training error on m' samples drawn from $\mathrm{EX}^{\eta}(\mathcal{D}_{S,\zeta})$ for $m' > \frac{|\zeta|}{10\eta\gamma^2} \log \frac{|\zeta|}{10\gamma\delta}$ and $\eta \in \left(0, \frac{1}{2}\right)$ then $\mathcal{R}_{\mathrm{Adv},\gamma}(f; \mathcal{D}_{S,\zeta}) \geq 0.1$ for all $\gamma > 0$.

Proof of Lemma 4. The first part of the algorithm works similarly to Lemma 3. The algorithm $\widetilde{\mathcal{A}}$ makes m calls to the oracle $\mathrm{EX}(\mathcal{D}_s^m)$ to obtain a set of points $\mathcal{S}_m = \{(x_1, y_1), \cdots, (x_m, y_m)\}$ where $m \geq 2|\zeta|^2 \log \frac{2|\zeta|}{\delta}\left(8\frac{(1-\eta)}{(1-2\eta)^2} + 1\right)$. $\widetilde{\mathcal{A}}$ computes $h \in \mathcal{H}$ as follows. To begin, let the list of intervals in h be I and $\mathcal{M}_z = \{\}$ Then do the following for every $(x, y) \in \mathcal{S}_m$.

1. let $z := [x]$,
2. Let $\mathcal{N}_z \subseteq \mathcal{S}_m$ be the set of all $(x, y) \in \mathcal{S}_m$ such that $|x - z| < 0.5$.
3. Compute the majority label $\tilde{y}$ of $\mathcal{N}_z$.
4. Add all $(x, y) \in \mathcal{N}_z$ such that $y \neq \tilde{y}$ to $\mathcal{M}_z$
5. If $\tilde{y} = 1$, then add the interval $(z - 0.5, z + 0.5)$ to I.
6. Remove all elements of $\mathcal{N}_z$ from $\mathcal{S}_m$ i.e. $\mathcal{S}_m := \mathcal{S}_m \setminus \mathcal{N}_z$.

For reasons similar to Lemma 3, as $m \geq 2|\zeta|^2 \log \frac{2|\zeta|}{\delta}\left(8\frac{(1-\eta)}{(1-2\eta)^2} + 1\right)$, Lemma 5 guarantees that with probability $1 - \frac{\delta}{2}$, each interval will have atleast $\frac{8(1-\eta)}{(1-2\eta)^2} \log \frac{2|\zeta|}{\delta}$ samples. Then for any specific interval, Lemma 6 guarantees that with probability atleast $1 - \frac{2|\zeta|}{\delta}$, the majority vote for the label in that interval will succeed in returning the de-noised label. Applying a union bound over all intervals, will guarantee that with probability atleast $1 - \delta$, the majority label of every interval will be the denoised label. As each interval inζ has atleast one point, all the intervals in ζ with label 1 will be included in I with probability $1 - \delta$. Thus, $\mathcal{R}(h; \mathcal{D}_{S,\zeta}) = 0$.

Now, for all $(x, y) \in \mathcal{M}_z$, add the interval $[x]$ to I if $y = 1$. If $y = 0$ then x must lie a interval $(a, b) \in I$. Replace that interval as follows $I := I \setminus (a, b) \cup \{(a, x), (x, b)\}$. As only a finite number of sets with Lebesgue measure of 0 were added or deleted from I, the net test error of h doesn't change and is still 0 i.e. $\mathcal{R}(h; \mathcal{D}_{S,\zeta}) = 0$

For the second part, we will invoke Theorem 3. To avoid confusion in notation, we will use Γ instead of ζ to refer to the sets in Theorem 3 and reserve ζ for the support of interval of $\mathcal{D}_{S,\zeta}$. Let Γ be any set of disjoint intervals of width $\frac{\gamma}{2}$ such that $|\Gamma| = \frac{0.1|\zeta|}{\gamma}$. This is always possible as the total width of all intervals in Γ is $\frac{0.1|\zeta|}{\gamma}\frac{\gamma}{2} = 0.1\frac{|\zeta|}{2}$ which is less than the total width of the support $\frac{|\zeta|}{2}$. c_1, c_2 from Eq. (9) is

$$c_1 = \mathbb{P}_{\mathcal{D}_{S,\zeta}}[\Gamma] = \frac{2 * 0.1|\zeta|}{2|\zeta|} = 0.1, \quad c_2 = \frac{2\gamma}{2|\zeta|}|\zeta| = \gamma$$

Thus, if h has an error of zero on a set of m' examples drawn from $\mathrm{EX}^{\eta}(\mathcal{D}_{S,\zeta})$ where $m' > \frac{0.1|\zeta|}{\eta\gamma^2} \log\left(\frac{0.1|\zeta|}{\gamma\delta}\right)$, then by Theorem 3, $\mathcal{R}_{\mathrm{Adv},\gamma}(h; \mathcal{D}_{S,\zeta}) > 0.1$.

Combining the two parts for

$$m > \max\left\{2|\zeta|^2 \log\frac{2|\zeta|}{\delta}\left(8\frac{(1-\eta)}{(1-2\eta)^2} + 1\right), \frac{0.1|\zeta|}{\eta\gamma^2} \log\left(\frac{0.1|\zeta|}{\gamma\delta}\right)\right\}$$

it is possible to obtain $h \in \mathcal{H}$ such that h has zero training error, $\mathcal{R}(\mathcal{D}_{S,\zeta}; h) = 0$ and $\mathcal{R}_{\mathrm{Adv},\gamma}(h; \mathcal{D}_{S,\zeta}) > 0.1$ for any $\gamma > 0$.

□

Lemma 5. *Given $k \in \mathbb{Z}_+$ and a distribution $\mathcal{D}_{S,\zeta}$, for any $\delta_2 > 0$ if $m > 2|\zeta|^2 k + 2|\zeta|^2 \log \frac{|\zeta|}{\delta_2}$ samples are drawn from $\mathrm{EX}(\mathcal{D}_{S,\zeta})$ then with probability atleast $1-\delta_2$ there are atleast k samples in each interval $\left(j - \frac{1}{4}, j + \frac{1}{4}\right)$ for all $j \in \zeta$.*

Proof of Lemma 5. We will repeat the following procedure $|\zeta|$ times once for each interval in ζ and show that with probability $\frac{\delta}{|\zeta|}$ the j^{th} run will result in atleast k samples in the j^{th} interval.

Corresponding to each interval in ζ, we will sample atleast m' samples where $m' = 2|\zeta| k + 2|\zeta| \log \frac{|\zeta|}{\delta_2}$. If z_i^j is the random variable that is 1 when the i^{th} sample belongs to the j^{th} interval, then j^{th} interval has atleast k points out of the m' points sampled for that interval with probability less than $\frac{\delta_2}{|\zeta|}$.

$$\begin{aligned}
\mathbb{P}\left[\sum_i z_i^j \le k\right] &= \mathbb{P}\left[\sum_i z_i^j \le (1-\delta)\mu\right] && \delta = 1 - \frac{k}{\mu}, \mu = \mathbb{E}\left[\sum_i z_i^j\right] \\
&\le \exp\left(-\left(1-\frac{k}{\mu}\right)^2 \frac{\mu}{2}\right) && \text{By Chernoff's inequality} \\
&\le \exp\left(-\left(\frac{m'}{2|\zeta|} - k + \frac{k^2|\zeta|}{2m'}\right)\right) && \mu = \frac{m'}{|\zeta|} \\
&\le \exp\left(k - \frac{m'}{2|\zeta|}\right) \le \frac{\delta_2}{|\zeta|}
\end{aligned}$$

where the last step follows from $m' > 2|\zeta| k + 2|\zeta| \log \frac{|\zeta|}{\delta_2}$. With probability atleast $\frac{\delta}{|\zeta|}$, every interval will have atleast k samples. Finally, an union bound over each interval gives the desired result. As we repeat the process for all $|\zeta|$ intervals, the total number of samples drawn will be atleast $|\zeta| m' = 2|\zeta|^2 k + 2|\zeta|^2 \log \frac{|\zeta|}{\delta_2}$. □

Lemma 6 (Majority Vote)**.** *For a given $y \in \{0,1\}$, let $S = \{s_1, \cdots, s_m\}$ be a set of size m where each element is y with probability $1-\eta$ and $1-y$ otherwise. If $m > \frac{8(1-\eta)}{(1-2\eta)^2} \log \frac{1}{\delta_1}$ then with probability atleast $1-\delta_1$ the majority of S is y.*

Proof of Lemma 6. Without loss of generality let $y = 1$. For the majority to be 1 we need to show that there are more than $\frac{m}{2}$ "1"s in S i.e. we need to show that the following probability is less than δ_1.

$$\begin{aligned}
\mathbb{P}\left[\sum s_i < \frac{m_1}{2}\right] &= \mathbb{P}\left[\sum s_i < \frac{m_1}{2\mu} * \mu + \mu - \mu\right] && \mu = \mathbb{E}\left[\sum s_i\right] \\
&= \mathbb{P}\left[\sum s_i < \left(1 - \left(1 - \frac{m_1}{2\mu}\right)\right)\mu\right] \\
&\le \exp\left(-\frac{(1-2\eta)^2}{8(1-\eta)^2}\mu\right) && \text{By Chernoff's Inequality} \\
&= \exp\left(-\frac{(1-2\eta)^2}{8(1-\eta)}m\right) && \because \mu = (1-\eta)m \\
&\le \delta_1 && \because m > \frac{8(1-\eta)}{(1-2\eta)^2}\log\frac{1}{\delta_1}
\end{aligned}$$

□

Theorem 3. *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 $c_1 \ge c_2 > 0$, $\rho > 0$, and a finite set $\zeta \subset \mathbb{R}^d$ satisfying*

$$\mathbb{P}_{\mathcal{D}}\left[\bigcup_{\mathbf{s}\in\zeta} \mathcal{B}_\rho^p(\mathbf{s})\right] \ge c_1 \quad \textit{and} \quad \forall \mathbf{s} \in \zeta,\ \mathbb{P}_{\mathcal{D}}\left[\mathcal{B}_\rho^p(\mathbf{s})\right] \ge \frac{c_2}{|\zeta|} \tag{9}$$

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 $1-\delta$, $\mathcal{R}_{\mathrm{Adv},2\rho}(f;\mathcal{D}) \geq c_1$.*

Proof of Theorem 3. From (9), for any ζ and $s \in \zeta$,

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

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

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

Thus,

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

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

$$\mathbb{P}_{\mathcal{S}_m\sim\mathcal{D}^m}\left[\forall s \in \zeta,\ \exists\,(\mathbf{x},y) \in \mathcal{S}_m : \mathbf{x} \in \mathcal{B}_\rho(s) \wedge \mathbf{x} \text{ is mislabelled}\right] \geq 1 - \delta \tag{10}$$

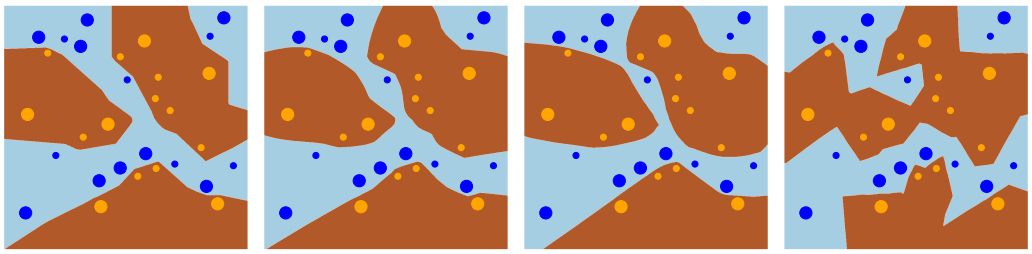
As for all $\mathbf{s} \in \mathbb{R}^d$ and $\forall \mathbf{x}, \mathbf{z}, \in \mathcal{B}_\rho^p(\mathbf{s})\,,\ \|\mathbf{x}-\mathbf{z}\|_p \leq 2\rho$, we have that

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

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 (10) 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 (10). □

C Complexity of decision boundaries

When neural networks are trained they create classifiers whose decisions boundaries are much simpler than they need to be for being adversarially robust. A few recent papers [22, 26] have discussed that robustness might require more complex classifiers. In Theorem 1 and 2 we discussed this theoretically and also why this might not violate the traditional wisdom of Occam's Razor. In particular, complex decision boundaries does not necessarily mean more complex classifiers in statistical notions of complexity like VC dimension. In this section, we show through a simple experiment how the decision boundaries of neural networks are not "complex" enough to provide large enough margins and are thus adversarially much more vulnerable than is possible.



(a) Shallow NN (b) Shallow-Wide NN (c) Deep NN (d) Large Margin

Figure 3: Decision boundaries of neural networks are much simpler than they should be.

Coarse Class	Fine Classes
Dog	Chihuahua, Japanese spaniel, Maltese dog, Pekinese, Shih-Tzu, Blenheim spaniel
Bird	cock, hen, ostrich, brambling, goldfinch, house finch
Insect	tiger beetle,ladybug,ground beetle, long-horned beetle, leaf beetle, dung beetle
Monkey	guenon, patas, baboon, macaque, langur, colobus
Car	jeep, limousine,cab, beach wagon, ambulance, convertible
Feline	leopard, snow leopard, jaguar, lion, cougar, lynx
Truck	tow truck, moving van, fire engine, pickup, garbage truck, police van
Fruit	Granny Smith, rapeseed, corn, acorn, hip, buckeye
Fungus	agaric, gyromitra, stinkhorn, earthstar, hen-of-the-woods, coral fungus
Boat	gondola, fireboat, speedboat, lifeboat, yawl, canoe

Table 1: Fine-grained classes in Restricted Imagenet

We train three different neural networks with ReLU activations, a shallow network (Shallow NN) with 2 layers and 100 neurons in each layer, a shallow network with 2 layers and 1000 neurons in each layer (Shallow-Wide NN), and a deep network with 4 layers and 100 neurons in each layer. We train them for 200 epochs on a binary classification problem as constructed in Figure 3. The distribution is supported on blobs and the color of each blob represent its label. On the right side, we have the decision boundary of a large margin classifier, which is simulated using a 1-nearest neighbour.

From Figure 3, it is evident that the decision boundaries of neural networks trained with standard optimizers have far *simpler* decision boundaries than is needed to be robust (eg. the 1- nearest neighbour is much more robust than the neural networks.)

Restricted Imagenet Settings There are 60 fine classes and 10 coarse classes with each coarse class having 6 distinct fine classes in them. The train set size is 77237 and the test set size is 3000. The fine classes within each coarse are balanced i.e. given a coarse class all the fine classes in it are equally represented in this dataset.