Adversarial Noises Are Linearly Separable for (Nearly) Random Neural Networks

Huishuai Zhang[⋆] **Da Yu**[†] **Yiping Lu**[‡] **Di He**[♣]
[⋆]Microsoft Research Asia [†]Sun Yat-sen University [‡]Stanford University
[♣]School of Intelligence Science and Technology, Peking University

Abstract

Adversarial example, which is usually generated by adding imperceptible adversarial noise to a clean sample, is ubiquitous for neural networks. In this paper we unveil a surprising property of adversarial noises when they are put together, i.e., adversarial noises crafted by one-step gradient methods are linearly separable if equipped with the corresponding labels. We theoretically prove this property for a two-layer network with randomly initialized entries and the *neural tangent kernel* setup where the parameters are not far from initialization. The proof idea is to show the label information can be efficiently backpropagated to the input while keeping the linear separability. Our theory and experimental evidence further show that the linear classifier trained with the adversarial noises of the training data can well classify the adversarial noises of the test data, indicating that adversarial noises actually inject a distributional perturbation to the original data distribution. Furthermore, we empirically demonstrate that the adversarial noises may become *less* linearly separable when the above conditions are compromised while they are still much easier to classify than original features.

1 INTRODUCTION

Modern deep learning models have achieved great accuracy on vast intelligence tasks. However at the same time, they have been demonstrated vulnerable to adversarial examples, i.e., imperceptible perturbations can significantly change the output of a neural network at test time. This hinders the applicability of deep learning model on safety-critical tasks (Biggio et al., 2013; Szegedy et al., 2014).

Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206.

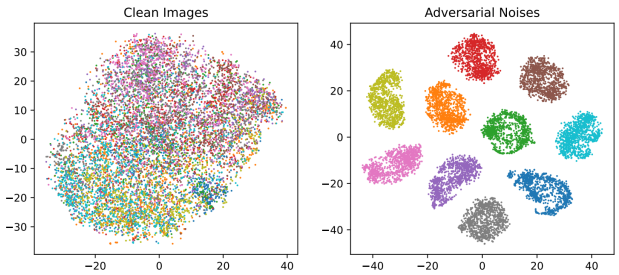

Figure 1: Comparison between the T-SNEs of clean CIFAR-10 images and those of adversarial noises. Points with the same color are from the same class. Adversarial noises from the same class are well clustered. Adversarial noises are generated with a random initialized ResNet-18 model.

Adversarial example is usually generated via finding a perturbed sample that maximizes the loss (untargeted attack Carlini et al. (2019)), i.e.,

$$\arg\max_{\boldsymbol{x}'\in\mathbb{B}(\boldsymbol{x},\epsilon)} \ell(\boldsymbol{x}', y; \theta). \tag{1}$$

Usually the above objective is solved by projected gradient descent (PGD) methods Goodfellow et al. (2014) for every sample pair $(\boldsymbol{x}, y)$. Therefore, the adversarial noise $\boldsymbol{x}' - \boldsymbol{x}$ is sample $(\boldsymbol{x}, y)$ specific and model (θ) specific. Most existing theoretical and empirical studies are mainly about this setting except the universal attacks (Moosavi-Dezfooli et al., 2017; Akhtar et al., 2018; Zhang et al., 2021).

In this paper, we study the adversarial noises from a population's perspective and ask

"What property do the adversarial noises exhibit when they are put together?"

Due to the complicated procedure of generating adversarial noises, one might think they must be scattered quite casually and disorderly. However, surprisingly, we observe that adversarial noises are well clustered with regards to the labels of the original samples. This is illustrated in Figure 1

where the adversarial noises of samples from three classes are projected into two-dimensional space via t-distributed Stochastic Neighbor Embedding (t-SNE) (Hinton & Roweis, 2002).

More specifically, in this paper we argue that under some assumption adversarial noises are almost linearly separable if they are equipped with the labels of the corresponding original samples, i.e., a new constructed dataset $\{(\text{adversarial noise } i,\ \text{label } i)\}_{i\in[n]}$ is linearly separable (as shown in Figure 2). This new property is important for us to better understand the behavior of adversarial noises.

We first study why such phenomenon happens. Specifically, we consider the adversarial noises generated by the single-step *Projected Gradient Descent (PGD)* algorithm (Goodfellow et al., 2014), i.e.[1],

$$\boldsymbol{x}^{adv} = \boldsymbol{x} + \eta \nabla_x \ell(\boldsymbol{x}, y; \theta) \tag{2}$$

with a suitable step size η. We theoretically prove that for a randomly-initialized two-layer neural network, the adversarial noises are linearly separable. We further prove that the linear separability also holds for the neural tangent kernel (NTK) regime where the weights are trained to fit the data while staying in a neighborhood of the initialization. The proof idea is to show the label information or the error of last layer, which are separable initially, can be efficiently back-propagated to the input and then a linear classifier is conceived to classify these adversarial noises perfectly. We deal with the correlation between forward and backward process via the Gaussian conditioning technique (Bayati & Montanari, 2011; Yang, 2020; Montanari & Wu, 2022). Throughout the proofs, we spend much effort to obtain high probability bounds. Such high probability bounds are not only stronger than expectation bounds but also critical to make the linear separability claim valid for all adversarial noises of the whole dataset. This is in contrast with previous studies (Bubeck et al., 2019; Montanari & Wu, 2022) that are to understand example-specific property of adversarial noises, e.g. why an adversarial noise is imperceptible but able to attack successfully.

The theory indicates that the linear separability of adversarial noises actually are generalizable to the test set, which is also verified in Figure 2. That is, a linear classifier trained on the adversarial noises of the training data points can well classify the adversarial noises of the test data points as long as they follow the same procedure of generation. This means the generation of adversarial noises actually inject a distributional perturbation to the original data distribution.

We also empirically explore the property of adversarial noises beyond the theoretical regime, especially for the case where the neural network is trained with a large learning rate, the case of other adversarial noise generation algorithms, and the case of adversarially trained models. Although the adversarial noises are not perfectly linearly separable in these wild scenarios, a consistent message is that they are much easier to fit than original features, i.e., a linear classifier on adversarial noises can achieve much higher accuracy than the best linear classifier on the original dataset.

[1]Here we consider the corresponding constraint in Equation 1 is l_2 ball and then the projection can be absorbed in the step size.

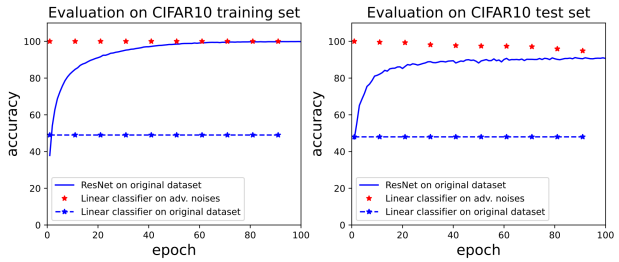

Figure 2: Training and test accuracy of **linear models** on adversarial noises, which are generated with ResNet-18 on CIFAR-10 over a standard training process of SGD with $lr = 0.001$.

Overall, our contribution can be summarized as follows.

- We unveil and theoretically prove a surprising phenomenon that adversarial noises are almost linearly separable for (nearly) random two-layer networks.
- We show that the linear separability of adversarial noises may be compromised when going beyond the theoretical regime, but they are still much easier to classify than original features.

1.1 Related work

There are some explanations why adversarial examples exist, e.g., the deep network classifiers being too linear locally because of ReLU like activations (Goodfellow et al., 2014), the boundary tilting hypothesis that the classification boundary is close to the submanifold of the training data (Tanay & Griffin, 2016), the isoperimery argument (Fawzi et al., 2018; Shafahi et al., 2019) and the dimpled manifold model (Shamir et al., 2021). There are also theoretical researches on the difficulty of adversarial learning difficulty, e.g., robust classifier requiring much more training data (Schmidt et al., 2018) and the computational intractability of building robust classifiers (Bubeck et al., 2019).

Specifically, (Ilyas et al., 2019) did an informative experiment showing that adversarial noises can create predictive signals. Beyond (Ilyas et al., 2019), our work first gives rigorous proof why this happens in theory, and then argues the prediction ability is so strong that the adversarial noises are as simple as being linear separable.

One related concept is the *label leakage* (Kurakin et al., 2017) that adversarial examples are crafted by using true label information in the single-step gradient methods and

hence may be easier to classify. Our results greatly extend/verify this concept by showing the adversarial noises are linearly separable.

Recent works explain why single-step PGD is able to attack deep models theoretically (Montanari & Wu, 2022; Bartlett et al., 2021; Bubeck et al., 2021; Daniely & Schacham, 2020). They show that for random neural networks, the output is roughly linear around the input sample. Then the high-dimensional statistics tell that a random weight vector has small inner product with a given input while at the same time a small perturbation can sufficiently change the output. Instead of the example-wise point of view, we unveil the population property of adversarial noises.

Our finding is also related with the concept *shortcut learning* (Beery et al., 2018; Niven & Kao, 2019; Geirhos et al., 2020) that deep models may rely on shortcuts to make predictions. Shortcuts are spurious features that are correlated with the label but not in a causal way. In this work, we show the linear separability makes adversarial noises perfect shortcut, which may hinder the classifier learns true features in the adversarial training. Our study is inspired by the finding that the data poisoning for availability attack is adding simple features (Yu et al., 2021) to the training data. We focus on the adversarial noises and analyze their linear-separability theoretically.

2 PROBLEM SETUP AND NOTATIONS

We study the distribution of adversarial noises of neural networks. Although the study is not constrained to specific networks, we analyze a simplified model to ease the technical exposure.

Specifically, we consider a two-layer neural network with input dimension d and width m:

$$f(\boldsymbol{x}; \boldsymbol{a}, \boldsymbol{W}) = \boldsymbol{a}^\top \sigma(\boldsymbol{W}\boldsymbol{x}), \tag{3}$$

where σ is the ReLU activation function which is applied coordinate-wisely, input $\boldsymbol{x} \in \mathbb{R}^d$, $\boldsymbol{W} \in \mathbb{R}^{m\times d}$, and readout layer $\boldsymbol{a} \in \mathbb{R}^m$. The network parameters are initialized as follows. Each entry of $\boldsymbol{W}$ is independently generated from $\mathcal{N}(0, 1/d)$, and each entry of $\boldsymbol{a}$ is independently generated from $\mathcal{N}(0, 1/m)$. Moreover, $\boldsymbol{W}$ and $\boldsymbol{a}$ are independent from each other. We use a new notation θ to represent the whole trainable parameters in the network, i.e., here $\theta = \{\boldsymbol{W}, \boldsymbol{a}\}$.

We consider binary classification task with a dataset $\{(\boldsymbol{x}_i, y_i)\}_{i\in[n]}$, where $\boldsymbol{x}_i \in \mathbb{R}^d$ and $y_i \in \{-1, +1\}$ for $i = 1, ..., n$. We use a negative log sigmoid loss, i.e.,

$$\ell(\boldsymbol{x}) = -\log s(yf(\boldsymbol{x})), \tag{4}$$

where $s(z) = \frac{1}{1+e^{-z}}$ is the sigmoid function.

We consider the one-step gradient method to generate the adversarial noise, i.e.,

$$\boldsymbol{r}_x = \frac{\partial \ell(\boldsymbol{x})}{\partial \boldsymbol{x}} = -(1 - s(yf(\boldsymbol{x})))y\nabla_x f(\boldsymbol{x}), \tag{5}$$

where $\nabla_x f(\boldsymbol{x})$ is the gradient with respect to the input and we may omit the subscript when it is clear from the context. For the two-layer neural network (Equation 3), this gradient is given by

$$\nabla f(\boldsymbol{x}) = \boldsymbol{W}^\top \boldsymbol{D}_x \boldsymbol{a}, \tag{6}$$

where $\boldsymbol{D}_x \in \mathbb{R}^{m\times m}$ is a diagonal matrix and the diagonal entries are given by $\sigma'(\boldsymbol{W}\boldsymbol{x})$. The adversarial example is given by

$$\boldsymbol{x}^{adv} = \boldsymbol{x} + \eta \boldsymbol{r}_x, \tag{7}$$

where η is step size has magnitude $O(1)$. Here we assume the ball constraint in Equation 1 is measured in l_2 distance and hence the projection can be removed. It is interesting and important to extend the analysis to other distances which are empirically verified in Section 4

We next state the mathematical definition of linear separability for a binary-label dataset.

Definition 1 (Linearly separable)**.** *We say a set $\{\boldsymbol{x}_i, y_i\}_{i\in[n]}$ with $y_i \in \{+1, -1\}$ linearly separable if $\exists \boldsymbol{v}$ such that $\forall i : \langle \boldsymbol{v}, y_i\boldsymbol{x}_i\rangle > 0$.*

Notations. In the sequel, we use $\|\boldsymbol{x}\|$ to denote the ℓ_2 norm of a vector $\boldsymbol{x}$, We use $\|\boldsymbol{M}\|_2$ and $\|\boldsymbol{M}\|_F$ to denote the spectral norm and the Frobenius norm of a matrix $\boldsymbol{M}$, respectively. The learning process is to minimize the average loss $\mathcal{L}(\theta) = \sum_{i=1}^n \ell(\theta; \boldsymbol{x}_i, y_i)/n$. We assume $\|\boldsymbol{x}_i\| = \sqrt{d}$ for all $i \in [n]$.

Besides, we also define the following notations to describe the bounds we derived. We write $f(\cdot) = \mathcal{O}(g(\cdot))$, $f(\cdot) = \Omega(g(\cdot))$ to denote $f(\cdot)/g(\cdot)$ is upper or lower bounded by a positive constant. We use $f(\cdot) = \Theta(g(\cdot))$ to denote that $f(\cdot) = \Omega(g(\cdot))$ and $f(\cdot) = \mathcal{O}(g(\cdot))$.

3 PROVABLE LINEAR SEPARABILITY OF ADVERSARIAL NOISES

In this section, we show that the adversarial noises exhibit surprising linearly-separable phenomenon when put together. We first analyze why such phenomenon exists for randomly initialized network. Then we extend the analysis to the NTK setting.

3.1 Linear Separability at Initialization

We claim that for a two-layer network at its initialization, the adversarial noises are linearly separable if equipped with corresponding labels, i.e., $\{\boldsymbol{r}_{x_i}, y_i\}_{i=1}^n$ is linearly separable.

Theorem 3.1. *For the two-layer network given by Equation 3 and the adversarial noises $\{\boldsymbol{r}_{x_i}\}_{i=1}^n$ generated by Equation 5, there exists $\boldsymbol{v}$ such that $\forall i \in [n], \langle \boldsymbol{v}, y_i \boldsymbol{r}_{x_i} \rangle > 0$ with high probability. Specifically $\boldsymbol{v} = -\boldsymbol{W}^\top \boldsymbol{a}$ serves this purpose with probability at least $1 - 3Cn(e^{-c_1 d} + e^{-c_2 m})$ where C, c_1, c_2 are some constants.*

The adversarial noise in Equation 5 is $\boldsymbol{r}_{x_i} = -(1 - s(yf(\boldsymbol{x}))) y_i \nabla f(\boldsymbol{x}_i)$, where $1 - s(yf(\boldsymbol{x})) > 0$ because of the sigmoid function. Hence it is sufficient to show that $\langle -\boldsymbol{v}, \nabla f(\boldsymbol{x}_i) \rangle > 0$ holds for all i. Next we give a proof outline and the full derivation is deferred to Appendix 6.

Proof Outline. To give an intuitive idea why the claim is probably true, for a generic input $\boldsymbol{x}$ and $\boldsymbol{v} = -\boldsymbol{W}^\top \boldsymbol{a}$, we calculate the expectation and the variance of $\langle -\boldsymbol{v}, \nabla f(\boldsymbol{x}) \rangle$ for a simplified case: $\boldsymbol{D}_x$ is random and independent from all others $\{\boldsymbol{W}, \boldsymbol{a}, \boldsymbol{x}\}$. We are safe to ignore the subscript in $\boldsymbol{D}_x$ for this case.

Because of the property of ReLU activation, we further assume that the diagonal entries of $\boldsymbol{D}$ are independently and randomly sampled from $\{0, 1\}$ with equal probability, i.e., for all $k \in [m]$

$$\boldsymbol{D}(k,k) = \begin{cases} 0, & \text{with probability } 0.5, \\ 1, & \text{with probability } 0.5. \end{cases}$$

Then we can compute $\mathbb{E}\langle -\boldsymbol{v}, \nabla f(\boldsymbol{x}) \rangle$ and $\mathrm{Var}\langle -\boldsymbol{v}, \nabla f(\boldsymbol{x}) \rangle$ as follows,

$$\begin{aligned} &\mathbb{E}\langle -\boldsymbol{v}, \nabla f(\boldsymbol{x}) \rangle \\ &= \mathbb{E}[\boldsymbol{a}^\top \boldsymbol{W}\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{a}] \\ &= \mathbb{E}\left[\mathrm{Tr}(\boldsymbol{a}\boldsymbol{a}^\top \boldsymbol{W}\boldsymbol{W}^\top \boldsymbol{D})\right] \\ &= \mathrm{Tr}\left(\mathbb{E}[\boldsymbol{a}\boldsymbol{a}^\top]\mathbb{E}[\boldsymbol{W}\boldsymbol{W}^\top]\mathbb{E}[\boldsymbol{D}]\right) \\ &= \mathrm{Tr}\left(\frac{1}{m}\boldsymbol{I}_{m\times m} \cdot \boldsymbol{I}_{m\times m} \cdot \frac{1}{2}\boldsymbol{I}_{m\times m}\right) \\ &= \frac{1}{2}, \end{aligned} \tag{8}$$

and

$$\begin{aligned} &\mathrm{Var}\langle -\boldsymbol{v}, \nabla f(\boldsymbol{x}) \rangle \\ &= \mathbb{E}\left[\left(\boldsymbol{a}^\top \boldsymbol{W}\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{a}\right)^2\right] - \left(\mathbb{E}[\boldsymbol{a}^\top \boldsymbol{W}\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{a}]\right)^2 \\ &= \left(\frac{1}{4} + \frac{5}{4m} + \frac{1}{d} + \frac{2}{md}\right) - \frac{1}{4} \\ &= \frac{5}{4m} + \frac{1}{d} + \frac{2}{md} \end{aligned} \tag{9}$$

We note that the computation of $\mathbb{E}\left(\boldsymbol{a}^\top \boldsymbol{W}\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{a}\right)^2$ is quite complicated and heavily relies on the property of $\boldsymbol{a}, \boldsymbol{W}$ being Gaussian and the independence between $\boldsymbol{a}, \boldsymbol{W}$ and $\boldsymbol{D}$. By Chebyshev inequality, we can show that $\langle -\boldsymbol{v}, \nabla f(\boldsymbol{x}) \rangle > \frac{1}{2} - \delta$ with probability at least $1 - \frac{2}{\delta^2 d}$ assuming that $m > 1.2d + 2$. Taking the union bound, we can prove the claim holds with probability $1 - n\frac{2}{\delta^2 d}$. Thus, it requires $d \gg n$ to claim that the theorem holds with high probability. To obtain a tighter bound, it requires more elaborate concentration inequality, which is deferred to Appendix 6.

Next we consider the case where $\boldsymbol{D}_x$ is exactly $\sigma'(\boldsymbol{W}\boldsymbol{x})$.

This makes the analysis a bit harder as the $\boldsymbol{W}, \boldsymbol{W}^\top$ and $\boldsymbol{D}_x$ are correlated. We prove the claim via the technique of probability concentration and Gaussian conditioning (Yang, 2020; Montanari & Wu, 2022), where we use a lemma as follows.

Lemma 3.2 (Lemma 3.1 in (Montanari & Wu, 2022))**.** *Let $\boldsymbol{X} \in \mathbb{R}^{m\times d}$ which has i.i.d. standard Gaussian entries, and $\boldsymbol{A}_1 \in \mathbb{R}^{k_1 \times m}, \boldsymbol{A}_2 \in \mathbb{R}^{d\times k_2}$. Let $\boldsymbol{Y} = h_1(\boldsymbol{A}_1\boldsymbol{X}, \boldsymbol{X}\boldsymbol{A}_2, \boldsymbol{Z}_1)$ with $\boldsymbol{Z}_1$ independent of $\boldsymbol{X}$, $\boldsymbol{A}_2 = h_2(\boldsymbol{A}_1\boldsymbol{X}, \boldsymbol{Z}_2)$ with $\boldsymbol{Z}_2$ independent of $\boldsymbol{X}$. We assume that $(\boldsymbol{A}_1, \boldsymbol{Z}_1, \boldsymbol{Z}_2)$ is independent of $\boldsymbol{X}$. Then there exists $\tilde{\boldsymbol{X}} \in \mathbb{R}^{m\times d}$ which has the same distribution with $\boldsymbol{X}$ and is independent of $\boldsymbol{Y}$, such that*

$$\begin{aligned} \boldsymbol{X} =& \Pi_{\boldsymbol{A}_1}^{\perp} \tilde{\boldsymbol{X}} \Pi_{\boldsymbol{A}_2}^{\perp} + \Pi_{\boldsymbol{A}_1}^{\perp} \boldsymbol{X} \Pi_{\boldsymbol{A}_2} \\ &+ \Pi_{\boldsymbol{A}_1} \boldsymbol{X} \Pi_{\boldsymbol{A}_2}^{\perp} + \Pi_{\boldsymbol{A}_1} \boldsymbol{X} \Pi_{\boldsymbol{A}_2}, \end{aligned}$$

where $\Pi_{\boldsymbol{A}_1} \in \mathbb{R}^{m\times m}$ is the projection operator projecting onto the subspace spanned by the rows of $\boldsymbol{A}_1$, $\Pi_{\boldsymbol{A}_2} \in \mathbb{R}^{d\times d}$ is the projection operator projecting onto the subspace spanned by the columns of $\boldsymbol{A}_2$, and $\Pi_{\boldsymbol{A}_1}^{\perp} := \boldsymbol{I}_m - \Pi_{\boldsymbol{A}_1}$, $\Pi_{\boldsymbol{A}_2}^{\perp} := \boldsymbol{I}_d - \Pi_{\boldsymbol{A}_2}$.

The proof of Lemma 3.2 is in Appendix A.1 of Montanari & Wu (2022). By using Lemma 3.2, where plugging in $\boldsymbol{X} \leftarrow \boldsymbol{W}$, $\boldsymbol{A}_1 \leftarrow 0, \boldsymbol{A}_2 \leftarrow \boldsymbol{x}$, $\boldsymbol{Y} \leftarrow \boldsymbol{D}_x = \sigma'(\boldsymbol{W}\boldsymbol{x})$ and $\Pi_x = \frac{1}{d}\boldsymbol{x}\boldsymbol{x}^\top$, we have

$$\boldsymbol{W} = \tilde{\boldsymbol{W}} \Pi_x^{\perp} + \boldsymbol{W} \Pi_x, \tag{10}$$

where $\tilde{\boldsymbol{W}}$ has the same marginal distribution as $\boldsymbol{W}$ and is independent of $\boldsymbol{D}_x$. Consequently we have

$$\begin{aligned} &\boldsymbol{a}^\top \boldsymbol{W}\boldsymbol{W}^\top \boldsymbol{D}_x \boldsymbol{a} \\ &= \frac{1}{d}\boldsymbol{a}^\top \boldsymbol{W}\boldsymbol{x}\boldsymbol{x}^\top \boldsymbol{W}^\top \boldsymbol{D}_x \boldsymbol{a} + \boldsymbol{a}^\top \tilde{\boldsymbol{W}} \Pi_x^{\perp} \tilde{\boldsymbol{W}}^\top \boldsymbol{D}_x \boldsymbol{a} \\ &= \frac{1}{d}\boldsymbol{a}^\top \boldsymbol{W}\boldsymbol{x}\boldsymbol{x}^\top \boldsymbol{W}^\top \boldsymbol{D}_x \boldsymbol{a} + \boldsymbol{a}^\top \bar{\boldsymbol{W}}\bar{\boldsymbol{W}}^\top \boldsymbol{D}_x \boldsymbol{a}, \end{aligned} \tag{11}$$

where $\bar{\boldsymbol{W}} \in \mathbb{R}^{m\times(d-1)}$ has Gaussian entries with mean 0 and variance $\frac{1}{d}$, independent of $\boldsymbol{D}_x, \boldsymbol{a}$.

For the first term in Equation 11, let $\boldsymbol{h} = \boldsymbol{W}\boldsymbol{x}$, then with high probability $\|\boldsymbol{h}\| \approx \sqrt{m}$ and $\|\boldsymbol{D}_x\boldsymbol{h}\| \approx \sqrt{m/2}$. Given $\boldsymbol{h}$, we have $\boldsymbol{a}^\top \boldsymbol{h} \sim \mathcal{N}(0, \|\boldsymbol{h}\|^2/m)$ and $\boldsymbol{h}^\top \boldsymbol{D}_x \boldsymbol{a} \sim$

$\mathcal{N}(0, \|\boldsymbol{D}_x\boldsymbol{h}\|^2/m)$. Then we have

$$\begin{aligned}&\frac{1}{d}\left|\boldsymbol{a}^\top\boldsymbol{W}\boldsymbol{x}\boldsymbol{x}^\top\boldsymbol{W}^\top\boldsymbol{D}_x\boldsymbol{a}\right|\\&=\frac{1}{d}\left|\boldsymbol{a}^\top\boldsymbol{h}\boldsymbol{h}^\top\boldsymbol{D}_x\boldsymbol{a}\right|\\&=\frac{1}{d}\left|\boldsymbol{a}^\top\boldsymbol{h}\right|\cdot\left|\boldsymbol{h}^\top\boldsymbol{D}_x\boldsymbol{a}\right|.\end{aligned}\tag{12}$$

We can bound the right hand side of Equation 12 with high probability by using the following two lemmas.

Lemma 3.3. *Suppose $\|\boldsymbol{x}\| = \sqrt{d}$ and $\boldsymbol{W}$ is a Gaussian matrix with entry variance $1/d$. Let $\boldsymbol{h} = \boldsymbol{W}\boldsymbol{x}$, then we have*

$$\mathbb{P}\{\|\boldsymbol{h}\|^2 < 2m\} > 1 - e^{m/7}.\tag{13}$$

The proof of this lemma is based on the tail bound of χ^2 distribution.

Lemma 3.4. *Suppose $\|\boldsymbol{x}\| = \sqrt{d}$, $\boldsymbol{W}$ is a Gaussian matrix with entry variance $1/d$ and $\boldsymbol{a}$ is a Gaussian vector with entry variance $1/m$. Let $\boldsymbol{h} = \boldsymbol{W}\boldsymbol{x}$ and $\boldsymbol{D}_x = \sigma'(\boldsymbol{W}\boldsymbol{x})$. Then with probability at least $1 - e^{-m/7} - 4e^{-c_2 d/4}$, we have*

$$|\boldsymbol{a}^\top\boldsymbol{h}| < \sqrt{c_2 d}, \quad |\boldsymbol{a}^\top\boldsymbol{D}_x\boldsymbol{h}| < \sqrt{c_2 d},\tag{14}$$

where c_2 is some constant.

Thus if choosing $c_2 < 1/64$, we prove that the Equation 12 smaller than $1/64$ with probability at least $1 - n(e^{-m/7} + e^{-d/256})$.

For the second term in Equation 11, we can use the result for the case where $\boldsymbol{D}_x$ is indepdent of $\boldsymbol{W}\boldsymbol{x}, \boldsymbol{a}$ and have a lower bound for it. Combining these two parts together, we prove the Theorem 3.1 with high probability. □

We have shown that the linear separability of adversarial noises for network at its random initialization. Then one question is whether the adversarial examples are linearly separable, i.e., if there exists one $\boldsymbol{v}'$ such that $\langle \boldsymbol{v}', y_i\boldsymbol{x}_i^{adv}\rangle > 0$ for all $i \in [n]$. This is true if the input dimension and the network width are much larger than the number of input samples. In this case we can find a linear classifier that lives in a subspace perpendicular to the linear space spanned by $\{\boldsymbol{x}_i\}_{i=1}^n$.

Corollary 3.4.1. *For the two-layer network defined in Equation 3 and the adversarial samples given by $\boldsymbol{x}_i^{adv} = \boldsymbol{x}_i + \eta\boldsymbol{r}_i$, if $d > poly(n)$ there exists $\boldsymbol{v}' = -\Pi_{\boldsymbol{X}}^{\perp}\boldsymbol{W}^\top\boldsymbol{a}$ such that $\forall i : \langle \boldsymbol{v}', \boldsymbol{x}_i^{adv}\rangle > 0$ with high probability.*

Proof. The idea is that we can make the classifier staying in the orthogonal subspace of $\boldsymbol{X}\boldsymbol{X}^\top$ while can still linearly separates the adversarial samples.

We note that $\langle \boldsymbol{v}', \boldsymbol{x}^{adv}\rangle = \langle \boldsymbol{v}', \boldsymbol{x}\rangle + \langle \boldsymbol{v}', -\eta(1-p)\nabla f(\boldsymbol{x})\rangle = \langle \boldsymbol{v}', -\eta(1-p)\nabla f(\boldsymbol{x})\rangle$. We next prove with high probability

$$\boldsymbol{a}^\top\boldsymbol{W}\Pi_{\boldsymbol{X}}^{\perp}\boldsymbol{W}^\top\boldsymbol{D}_x\boldsymbol{a} > 0.\tag{15}$$

The above is indeed true because we can use Gaussian conditioning, i.e.,

$$\boldsymbol{a}^\top\boldsymbol{W}\Pi_{\boldsymbol{X}}^{\perp}\boldsymbol{W}^\top\boldsymbol{D}_x\boldsymbol{a} = \boldsymbol{a}^\top\bar{\boldsymbol{W}}\bar{\boldsymbol{W}}^\top\boldsymbol{D}_x\boldsymbol{a},\tag{16}$$

where $\bar{\boldsymbol{W}} \in \mathbb{R}^{m\times(d-n)}$ with $i.i.d.$ Gaussian entries with mean 0 and variance $1/d$. Then following the argument in the proof of Theorem 3.1, we complete the proof. □

Remark 1. *If d is not larger than n, then there may not exist a valid v' in Corollary 3.4.1.*

In this setting, there is not enough randomness in $\boldsymbol{W}$ to exploit. One possible choice is to increase the energy of the adversarial signal (by increasing the step size of Equation 7) to overcome the effect of the original input $\boldsymbol{x}$. By choosing $\eta = d^{1/4}$, the adversarial noise is still small compared with the original signal, i.e., $\frac{\|\boldsymbol{x}^{adv}-\boldsymbol{x}\|}{\|\boldsymbol{x}\|} = O(d^{-1/4})$ but the effect of the adversarial noise overweighs that of the original signal, i.e., $\frac{|\langle -\boldsymbol{W}^\top\boldsymbol{a}, \boldsymbol{x}^{adv}-\boldsymbol{x}\rangle|}{|\langle -\boldsymbol{W}^\top\boldsymbol{a}, \boldsymbol{x}\rangle|} = O(d^{1/4})$. Thus the adversarial examples may still be linearly separable in this case.

3.2 Linear Separability in NTK Regime

We have established the linear separability of adversarial noises for two-layer networks at initialization. In this section, we study the behavior of the adversarial noises when the network is slightly trained, i.e., the weights are not far from initialization. By the convergence theory of training neural network in *Neural Tangent Kernel* (NTK) regime, the network parameter can fit the training data perfectly even in a small neighborhood around initialization as long as the width of the network is large enough (Jacot et al., 2018; Allen-Zhu et al., 2018; Du et al., 2019; Chizat & Bach, 2018; Zou et al., 2018; Zhang et al., 2019). A typical result reads as follows, which we adapt to our notations.

Lemma 3.5 (Theorem 1 in (Allen-Zhu et al., 2018))**.** *Suppose a two-layer neural network defined by Equation 3 and a distinguishable dataset with n data points. If the network width $m \geq \Omega(poly(n)\cdot d)$, starting from random initialization θ, with probability at least $1 - e^{\Omega(\log^2 m)}$, then gradient descent with learning rate $\Theta\left(\frac{d}{poly(n)}\right)$ finds $\{\boldsymbol{W}^*, \boldsymbol{a}^*\}$ such that $\mathcal{L}(\boldsymbol{W}^*, \boldsymbol{a}^*) \leq \epsilon$ and $\|\boldsymbol{W}^* - \boldsymbol{W}\|_2 \leq \frac{1}{\sqrt{m}}$ and $\|\boldsymbol{a}^* - \boldsymbol{a}\| \leq \frac{1}{\sqrt{m}}$.*

Based on this result of NTK convergence, we can see that when the loss is minimized, the learned parameters are still very close to the initialization especially as the width becomes large. Thus, it is possible for us to show that the adversarial noises at the NTK solution are linear separable.

Theorem 3.6. *For the two-layer network defined in Equation 3, the NTK solution $\{\boldsymbol{W}^*, \boldsymbol{a}^*\}$ satisfying Lemma 3.5, and the adversarial noises $\{\boldsymbol{r}_i\}_{i=1}^n$ given by Equation 5, there exists $\boldsymbol{v}$ such that $\forall i : \langle \boldsymbol{v}, y_i \boldsymbol{r}_i \rangle > 0$. Specifically $\boldsymbol{v} = -\boldsymbol{W}^\top \boldsymbol{a}$ serves this purpose with high probability at least $1 - Cne^{-\Omega(m/\log^2 m)-cd}$ for some constants C, c.*

Proof Outline. The proof relies on that the NTK solution is very close to the initialization. We ignore the $1 - s(yf(\boldsymbol{x}))$ term and only calculate

$$\begin{aligned}
&\langle \boldsymbol{W}^\top \boldsymbol{a}, \nabla_x f(\boldsymbol{x}; \boldsymbol{W}^*, \boldsymbol{a}^*) \rangle \\
&= \boldsymbol{a}^\top \boldsymbol{W} \boldsymbol{W}^{*\top} \boldsymbol{D}_x^* \boldsymbol{a}^* \\
&= \boldsymbol{a}^\top \boldsymbol{W} \boldsymbol{W}^\top \boldsymbol{D}_x \boldsymbol{a} + \boldsymbol{a}^\top \boldsymbol{W} (\Delta \boldsymbol{W})^\top \boldsymbol{D}_x \boldsymbol{a} \\
&\quad + \boldsymbol{a}^\top \boldsymbol{W} \boldsymbol{W}^\top (\Delta \boldsymbol{D}_x) \boldsymbol{a} + \boldsymbol{a}^\top \boldsymbol{W} \boldsymbol{W}^\top \boldsymbol{D}_x (\Delta \boldsymbol{a}) \\
&\quad + \boldsymbol{a}^\top \boldsymbol{W} (\Delta \boldsymbol{W}^\top \Delta \boldsymbol{D}_x \boldsymbol{a} + \Delta \boldsymbol{W}^\top \boldsymbol{D}_x \Delta \boldsymbol{a} \\
&\quad + \boldsymbol{W}^\top \Delta \boldsymbol{D}_x \Delta \boldsymbol{a} + \Delta \boldsymbol{W}^\top \Delta \boldsymbol{D}_x \Delta \boldsymbol{a}) \qquad (17)
\end{aligned}$$

where $\Delta \boldsymbol{W} = \boldsymbol{W}^* - \boldsymbol{W}, \Delta \boldsymbol{a} = \boldsymbol{a}^* - \boldsymbol{a}$ and $\Delta \boldsymbol{D}_x = \boldsymbol{D}_x^* - \boldsymbol{D}_x$.

For the first term of Equation 17, by Theorem 3.1 we have with high probability

$$\boldsymbol{a}^\top \boldsymbol{W} \boldsymbol{W}^\top \boldsymbol{D}_x \boldsymbol{a} > 1/32. \qquad (18)$$

For the second term of Equation 17, we note that with high probability $\|\boldsymbol{D}_x \boldsymbol{a}\| \in (\frac{1}{2} - \delta, \frac{1}{2} + \delta)$ and $\left\|\boldsymbol{a}^\top \boldsymbol{W}\right\| \in (1 - \delta, 1 + \delta)$, and hence

$$|\boldsymbol{a}^\top \boldsymbol{W} (\Delta \boldsymbol{W})^\top \boldsymbol{D}_x \boldsymbol{a}| \le O(\frac{1}{\sqrt{m}}). \qquad (19)$$

We next bound the third term of Equation 17. From the convergence proof in the NTK regime, we have $\|\Delta \boldsymbol{D}_x\|_0 < \frac{m}{\log^2 m}$ with probability at least 1 -$e^{-\Omega(m/\log^2 m)}$ (Allen-Zhu et al., 2018, Lemma 8.2). Hence with high probability $\|(\Delta \boldsymbol{D}_x) \boldsymbol{a}\| \le \frac{1}{\log m}$. We need the following lemma to get an overall high probability bound.

Lemma 3.7 (Lemma 7.3 in (Allen-Zhu et al., 2018))**.** *For all sparse vectors $\boldsymbol{u}$ with $\|\boldsymbol{u}\|_0 \le O(\frac{m}{\log^2 m})$, we have with probability at least $1 - e^{-\Omega(m)}$,*

$$|\boldsymbol{a}^\top \boldsymbol{W} \boldsymbol{W}^\top \boldsymbol{u}| \le 2 \|\boldsymbol{u}\|. \qquad (20)$$

Thus, we have

$$|\boldsymbol{a}^\top \boldsymbol{W} \boldsymbol{W}^\top (\Delta \boldsymbol{D}_x \boldsymbol{a})| \le \frac{1}{\log m}. \qquad (21)$$

We next bound the fourth term of Equation 17

$$\begin{aligned}
&|\boldsymbol{a}^\top \boldsymbol{W} \boldsymbol{W}^\top \boldsymbol{D}_x (\Delta \boldsymbol{a})| \\
&\le \left\|\boldsymbol{a}^\top \boldsymbol{W}\right\| \cdot \left\|\boldsymbol{W}^\top\right\|_2 \cdot \|\boldsymbol{D}_x (\Delta \boldsymbol{a})\| \\
&\le 2 \cdot \sqrt{\frac{m}{d}} \cdot \frac{1}{\sqrt{m}} \\
&= O(\frac{1}{\sqrt{d}}). \qquad (22)
\end{aligned}$$

For the higher order terms in Equation 17, we can similarly bound them one by one.

Hence by combining bounds in (18), (19), (21) and (22) together and taking the union bound, we complete the proof. □

Beyond the NTK setting, we discuss the possible extensions here. First it is possible to extend the current results to the multi-layer neural network setting, as it has been demonstrated that a multi-layer neural network near its initialization also behaves like linear function with respect to the input (Allen-Zhu et al., 2018; Bubeck et al., 2021; Montanari & Wu, 2022). Second, it is desirable if one can extend the result to the PGD with respect to the l_∞ constraint. The extension towards this direction is not easy based on current technique. One main difficulty is that the sign operation in PGD would break the Gaussian property of the adversarial noises. One can hardly exploit the Gaussian conditioning to give a proof. Nonetheless, We will do empirical experiments and verify the distributional property of the adversarial noises for all these settings.

4 VERIFY LINEAR SEPARABILITY OF ADVERSARIAL NOISES IN PRACTICE

In this section, we empirically verify the linear separability of adversarial noises. Specifically, we will first verify our theories' prediction that the adversarial noises are indeed linearly separable for neural network at/near its random initialization. Then we go beyond the theoretical regime and explore the case where the networks are sufficiently trained and the case where the adversarial noises are generated with multiple-step PGD. We next describe the general setup of our experiments.

4.1 General Setup of Experiments

The target model architecture is the ResNet-18 model (He et al., 2016b) or a two-layer convolutional neural network. We train the target models on the CIFAR-10 dataset (Krizhevsky & Hinton, 2009) with standard random cropping and flipping as data augmentation. All models are trained for 100 epochs with a batchsize of 128.

To test the linear separability of adversarial noises, we train linear models that use the generated adversarial noises as

input and the labels of corresponding original examples as their labels. All perturbations are flattened into one-dimensional vectors. The higher the training accuracy of linear model, the better the linear separability. All linear models are trained for 50 steps with the L-BFGS optimizer (Liu & Nocedal, 1989). All the experiments are run with one single Tesla V100 GPU.

4.2 Adversarial Noises Within Theoretical Regime

We first verify our theoretical findings. For the initialization setup, we use random Gaussian (Kaiming initialization (He et al., 2016a)) to initialize the neural network and test the linear separability of its adversarial noises.

We do not directly work with the neural tangent kernel. Instead we use a small constant learning rate to mimic the case that the model is close to initialization across training. We take a snapshot of the model every 10 epochs of training. We generate the adversarial noises with respect to each snapshot and then train a linear classifier on them accordingly.

All adversarial noises are generated with single-step PGD. In addition to the training accuracy of linear models, we also report how well these linear models "generalize" to the adversarial noises on test data points, the so-called *test accuracy* of the linear models on adversarial noises.

Verification with two-layer networks. We conduct experiments to exactly show how two-layer networks behave with varying widths. Specifically, we use two-layer ReLU convolutional networks with width 4, 16, 64, 128, 256 to classify the CIFAR-10 dataset. We train all networks with learning rate 0.01 for 50 epochs. From Figure 3 we can see that although the two-layer CNN can only give 80% accuracy on the original data set, the adversarial noises are very easy to classify. The linear separability of adversarial noises is almost perfect for all widths across the whole training procedure. Therefore, our theory predicts empirical observations well and in practice the width requirement can be very weak to observe such phenomenon.

Verification with ResNet18 models. We plot the result of ResNet18 with CIFAR-10 classification in Figure 2. As the model is trained, the ResNet's accuracy on the original training data increases to 100% steadily. We see that a linear model cannot fit the original training data well but a linear model can fit the adversarial noises perfectly from the initialization to the end of the training. This finding confirms that our theoretical findings do hold in practical networks.

We also observe that the ability of the linear classifier generalizes to the "test data", i.e., the linear model trained on adversarial noises of the training data performs well on the adversarial noises of the test data. This finding implies that

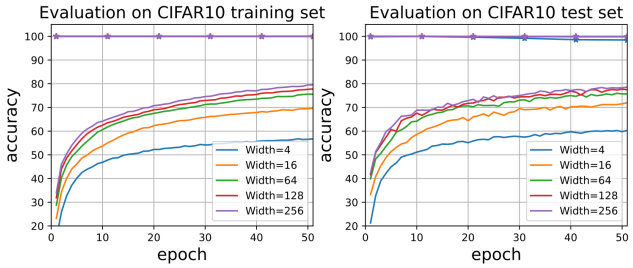

Figure 3: Training and test accuracy of **linear models** on adversarial noises. The noises are generated by using a two-layer ReLU convolutional neural networks with varying widths $\{4, 16, 64, 128, 256\}$. The networks are trained with lr=0.01 on CIFAR-10 task.

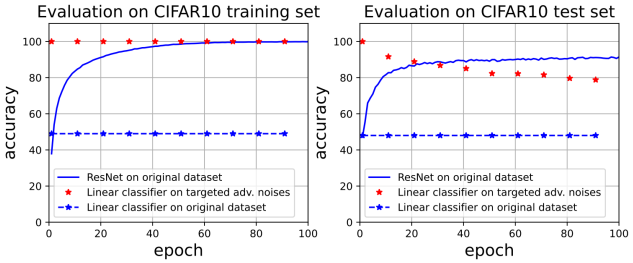

Figure 4: Training and test accuracy of **linear models** on targeted adversarial noises. The noises are generated by single-step PGD for targeted attack. The network is trained with lr=0.001 on CIFAR-10 task.

although the adversarial noises are designed with respect to specific samples, they actually introduce a new distribution on $(\boldsymbol{x}; y)$ to perturb the original data distribution. This new perspective may inspire new ways to defend against the adversarial noises.

Besides the above untargeted attacks, we also verify the linear separability of adversarial noises of the targeted attacks for comprehensiveness. Specifically, the adversarial noises are generated to make the network output a target label. We use $(\text{original label} + 1)\%10$ as the target label and use $(\text{adversarial noise}, \text{target label})$ as the inputs for the linear models. From Figure 4, we have similar observation as the the case of untargeted attacks.

4.3 Adversarial Noises Beyond Theoretical Regime

In this section, we go beyond the theoretical regime and see how the adversarial noises behave in the wild.

If using large learning rate, though not perfectly linearly separable, adversarial noises are still easy features. We first test the linear separability of adversarial noises for network that is sufficiently trained with a large learning rate. Specifically for the same ResNet-18 and CIFAR-10, we use learning rate $\text{lr} = 0.1$ instead of 0.001 in previous subsection

so that the model is no longer close to initialization after training. Similarly, we take a snapshot of the model every 10 epochs of training. We generate the adversarial noises with respect to each snapshot and then train a linear classifier on them accordingly. We plot the result in Figure 5. We can see that indeed, for this setting, the model is trained with a large learning rate, which goes beyond the regime of theoretical characterization, and gets further and further from the initialization with iterations. The linear separability becomes compromised as the training proceeds.

Apart from the reason that the weights move far away from initialization, we identify that the errors at the last layer become less separable as the training loss becomes small. At initialization, all the output activation is random and the only signal in the last layer gradient is the label. After training, we gradually learn label's information which makes the signal in the last layer gradient is not that informative and separable. We can alleviate this effect by tuning the softmax temperature of generating adversarial noises as shown in Figure 5. We note that the temperatures of softmax are only used for generating adversarial noises while not affecting the training of ResNet models. The larger T, the more uniform the softmax output.

Even though the adversarial noises are not perfectly linearly separable, they are still easy features, much easier than original features, e.g., the accuracy of linear classifier on adversarial noises are higher than that on original features (see Figure 5). Moreover, if we replace the linear classifier with a two-layer neural network, the adversarial noises can still be perfectly fit.

Similar phenomenon also holds for multi-step PGD and adversarially trained models. In this part, we consider the adversarial noises generated by the final models trained either standardly or adversarially. For adversarial training, we adopt the setup in Madry et al. (2018) that uses 7 steps of PGD with a stepsize of $2/255$ and the CIFAR-10 dataset.

For generating the adversarial noises, we test PGD with 5, 10, and 100 steps, where ϵ is set as usual $8/255$. We also plot the linear separability of clean data for a comparison. The results are in Figure 6.

Moreover, as another verification, we run adversarial training with the same step size and ϵ as that used to generate adversarial noise, i.e., L_∞ bound $8/255$. We plot the linear separability of adversarial noises along the training trajectories in Figure 7 and observe similar phenomenon as before.

We can see that the number of PGD steps does not affect the linear separability much. The adversarial noises are easier to fit than the original data for both standardly-trained and adversarially-trained models. Moreover, the adversarially trained model has substantially better linear separability than the standardly trained model. We speculate that the adversarially trained model has larger training losses and

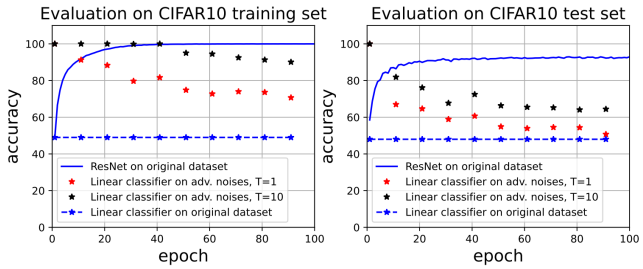

Figure 5: Training and test accuracy of **linear models** on adversarial noises. The noises are generated by using ResNet-18 models trained with **lr=0.1** on CIFAR-10 task and using two softmax temperatures $T = 1$ and $T = 10$.

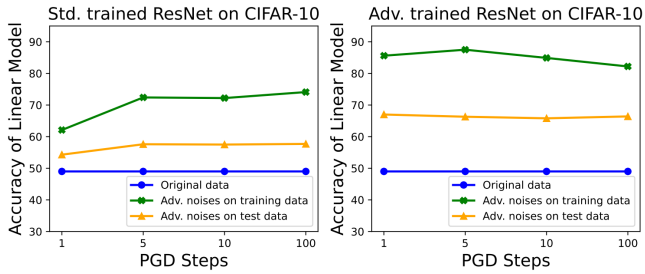

Figure 6: Training accuracy of linear models on adversarial noises generated with standardly/adversarially trained models. The blue line is training accuracy on original data.

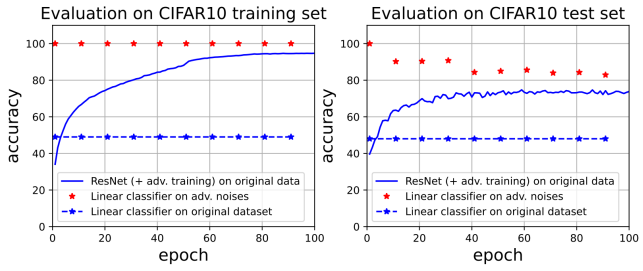

Figure 7: Training and test accuracy of linear models on adversarial noises with L_∞ bound $8/255$. The noises are generated with ResNet-18. We use the adversarial training setup in Madry et al. (2018) to train the target model. To evaluate the accuracy of linear models, we use the same setup as that in adversarial training to generate adversarial noises.

hence larger error at the last layer, which shares similar effect to tuning the temperature as shown in Figure 5.

Although the adversarial noises may not be perfectly linearly separable for these wild scenarios, one consistent message is that the adversarial noises are still much easier to fit than original data. The linear classifier still generalizes to the adversarial noises on test data to some extend, which indicates adversarial noises inject distributional perturbation to the original data distribution.

More experiments We add experiments with L2 norm constraint in Appendix 7, which demonstrates similar behaviors of adversarial noises to that in Figure 4. We also present more experiments on VGG11 and wide ResNet (WRN28-4) in Appendix 7. The behaviors of adversarial noises for different architectures are similar to that shown in Figure 4. This implies that the linear separability of adversarial noises is an essential property that holds universally for different architectures.

5 CONCLUSION AND LIMITATION

In this paper, we unveil a phenomenon that adversarial noises are almost linearly separable for nearly random neural network. We theoretically prove why this happens. One key message is that the adversarial noises are easy to fit for no matter nearly random network or fully trained network. We think that such a distributional perspective of adversarial noises is a novel and important view to understand the behavior of adversarial examples. One thing left to explore is whether such phenomenon still exists for more network architectures Bortolussi et al. (2022).

One limitation of the work is that we do not have a straightforward way to exploit such phenomenon to improve the adversarial attack or defense. One future direction could be designing more powerful universal attacks based on our finding that the adversarial noise is rarely related to the input sample. Another direction is to understand the robust generalization gap. We argue that the easy-to-fit property of adversarial noises make them strong disruptive signals during adversarial training. Hence the neural network may fit these adversarial noises rather than the true features. This may partially answer why adversarial training is not that efficient for learning original features, which usually leads to deteriorated performance on clean test data.

References

Naveed Akhtar, Jian Liu, and Ajmal Mian. Defense against universal adversarial perturbations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3389–3398, 2018.

Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. *arXiv preprint arXiv:1811.03962*, 2018.

Peter L Bartlett, Sébastien Bubeck, and Yeshwanth Cherapanamjeri. Adversarial examples in multi-layer random relu networks. *arXiv preprint arXiv:2106.12611*, 2021.

Mohsen Bayati and Andrea Montanari. The dynamics of message passing on dense graphs, with applications to compressed sensing. *IEEE Transactions on Information Theory*, 57(2):764–785, 2011.

Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings of the European conference on computer vision (ECCV)*, 2018.

Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 387–402. Springer, 2013.

Luca Bortolussi, Ginevra Carbone, Luca Laurenti, Andrea Patane, Guido Sanguinetti, and Matthew Wicker. On the robustness of bayesian neural networks to adversarial attacks. *arXiv preprint arXiv:2207.06154*, 2022.

Sébastien Bubeck, Yin Tat Lee, Eric Price, and Ilya Razenshteyn. Adversarial examples from computational constraints. In *International Conference on Machine Learning*, pp. 831–840. PMLR, 2019.

Sébastien Bubeck, Yeshwanth Cherapanamjeri, Gauthier Gidel, and Rémi Tachet des Combes. A single gradient step finds adversarial examples on random two-layers neural networks. *arXiv preprint arXiv:2104.03863*, 2021.

Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian Goodfellow, Aleksander Madry, and Alexey Kurakin. On evaluating adversarial robustness. *arXiv preprint arXiv:1902.06705*, 2019.

Lenaic Chizat and Francis Bach. On the global convergence of gradient descent for over-parameterized models using optimal transport. In *Advances in Neural Information Processing Systems 31*. 2018.

Amit Daniely and Hadas Schacham. Most ReLU networks suffer from l2 adversarial perturbations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Simon S Du, Jason D Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning (ICML)*, 2019.

Alhussein Fawzi, Hamza Fawzi, and Omar Fawzi. Adversarial vulnerability for any classifier. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2020.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016a.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, pp. 630–645. Springer, 2016b.

Geoffrey E Hinton and Sam Roweis. Stochastic neighbor embedding. In *Advances in neural information processing systems*, volume 15, 2002.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Anish Athalye, Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems*, pp. 8571–8580, 2018.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. 2009.

Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. In *International Conference on Learning Representations (ICLR)*, 2017.

Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 1989.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations (ICLR)*, 2018.

Andrea Montanari and Yuchen Wu. Adversarial examples in random neural networks with general activations. *arXiv preprint arXiv:2203.17209*, 2022.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Timothy Niven and Hung-Yu Kao. Probing neural network comprehension of natural language arguments. *arXiv preprint arXiv:1907.07355*, 2019.

Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially robust generalization requires more data. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 5019–5031, 2018.

Ali Shafahi, Mahyar Najibi, Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Adi Shamir, Odelia Melamed, and Oriel BenShmuel. The dimpled manifold model of adversarial examples in machine learning. *arXiv preprint arXiv:2106.10151*, 2021.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2015.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR)*, 2014.

Thomas Tanay and Lewis Griffin. A boundary tilting persepective on the phenomenon of adversarial examples. *arXiv preprint arXiv:1608.07690*, 2016.

Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010.

Greg Yang. Tensor programs iii: Neural matrix laws. *arXiv preprint arXiv:2009.10685*, 2020.

Da Yu, Huishuai Zhang, Wei Chen, Jian Yin, and Tie-Yan Liu. Indiscriminate poisoning attacks are shortcuts. *arXiv preprint arXiv:2111.00898*, 2021.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *The British Machine Vision Conference (BMVC)*, 2016.

Chaoning Zhang, Philipp Benz, Chenguo Lin, Adil Karjauv, Jing Wu, and In So Kweon. A survey on universal adversarial attack. *arXiv preprint arXiv:2103.01498*, 2021.

Huishuai Zhang, Da Yu, Mingyang Yi, Wei Chen, and Tie-Yan Liu. Convergence theory of learning over-parameterized resnet: A full characterization. *arXiv preprint arXiv:1903.07120*, 2019.

Difan Zou, Yuan Cao, Dongruo Zhou, and Quanquan Gu. Stochastic gradient descent optimizes over-parameterized deep ReLU networks. *arXiv preprint arXiv:1811.08888*, 2018.

6 Some Proofs in Theorem 3.1

6.1 The Independent Case

We first prove the high probability bound for the case: $\boldsymbol{D}_x$ is random and independent from all others $\{\boldsymbol{W}, \boldsymbol{a}, \boldsymbol{x}\}$, which is restated as the following lemma.

Lemma 6.1. *Suppose that $\boldsymbol{W} \in \mathbb{R}^{m\times d}$ whose entries are i.i.d. sampled from $\mathcal{N}(0, 1/d)$, $\boldsymbol{a} \in \mathbb{R}^m$ whose entries are i.i.d. sampled from $\mathcal{N}(0, 1/m)$, $\boldsymbol{D}$ is a diagonal matrix whose diagonal entries are i.i.d., sampled from $Bernoulli(\frac{1}{2})$. We further assume that $\boldsymbol{a}, \boldsymbol{W}$ and $\boldsymbol{D}$ are mutually independent. Then we have with probability at least $1 - 3Cn(e^{-c_1 d} + e^{-c_2 m})$ where C, c_1, c_2 are some constants,*

$$\boldsymbol{a}^\top \boldsymbol{W}\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{a} > \frac{1}{32}. \tag{23}$$

Proof. We note that

$$\boldsymbol{a}^\top \boldsymbol{W}\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{a} = \boldsymbol{a}^\top \boldsymbol{D}\boldsymbol{W}\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{a} + \boldsymbol{a}^\top (\boldsymbol{I} - \boldsymbol{D})\boldsymbol{W}\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{a}. \tag{24}$$

For the first term, $\boldsymbol{a}^\top \boldsymbol{D}\boldsymbol{W}\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{a} = \left\|\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{a}\right\|^2$. Given $\boldsymbol{D}\boldsymbol{a}$, we have $\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{a} \sim \mathcal{N}\left(0, \frac{\|\boldsymbol{D}\boldsymbol{a}\|^2}{d}\boldsymbol{I}_{d\times d}\right)$ and hence $\left\|\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{a}\right\|^2 \overset{d}{=} \frac{\|\boldsymbol{D}\boldsymbol{a}\|^2}{d}\chi^2_d$.

We need a bound on the tail probability of χ^2_d.

Lemma 6.2. *Suppose $X \sim \chi^2_d$, i.e., chi square distribution with freedom d. Then we have*

$$\mathbb{P}\{X < zd\} \le (ze^{1-z})^{d/2}, \quad \textit{for } z < 1, \tag{25}$$

$$\mathbb{P}\{X > zd\} \le (ze^{1-z})^{d/2}, \quad \textit{for } z > 1. \tag{26}$$

Hence

$$\mathbb{P}\left\{\left\|\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{a}\right\|^2 > \frac{\|\boldsymbol{D}\boldsymbol{a}\|^2}{d}\cdot\frac{d}{2}\right\} \ge 1 - \left(\frac{e^{1/2}}{2}\right)^{d/2} > 1 - e^{-m/11}. \tag{27}$$

We note that $\|\boldsymbol{a}\|^2 \sim \frac{1}{m}\chi^2_m$. Hence $\mathbb{P}\{\|\boldsymbol{a}\|^2 < z\} \le (ze^{1-z})^{m/2}$ for $z < 1$ and $\mathbb{P}\{\|\boldsymbol{a}\|^2 > z\} \le (ze^{1-z})^{m/2}$ for $z > 1$.

The diagonal entries of $\boldsymbol{D}$ are Bernoulli random variables, and hence $\mathrm{Tr}(\boldsymbol{D})$ is a Binomial random variable with parameter $(m, \frac{1}{2})$. Due to the Hoeffding-type tail bound of Binomial random variable, we have for $z < 1/2$

$$\mathbb{P}\{\mathrm{Tr}(\boldsymbol{D}) < zm\} < \exp\left(-2m\left(\frac{1}{2} - z\right)^2\right). \tag{28}$$

Define an event $E_1 := \{\mathrm{Tr}(\boldsymbol{D}) > \frac{1}{4}m\}$ and then its probability is at least $1 - e^{-m/8}$. On event E_1, we can show $\mathbb{P}\{\|\boldsymbol{D}\boldsymbol{a}\|^2 > 1/8\} > 1 - e^{-(\log\sqrt{2} - \frac{1}{4})m} > 1 - e^{-m/11}$. Then define another event $E_2 := \{\|\boldsymbol{D}\boldsymbol{a}\|^2 > 1/8\}$ whose probability is at least $1 - e^{-m/8} - e^{-m/11}$.

Hence for the first term we have with probability at least $1 - e^{-m/8} - 2e^{-m/11}$

$$\boldsymbol{a}^\top \boldsymbol{D}\boldsymbol{W}\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{a} > \frac{1}{16}. \tag{29}$$

For the second term, let D denote the set of index j that $\boldsymbol{D}_{j,j} = 1$, $\bar{D}$ denote the set of index j that $\boldsymbol{D}_{j,j} = 0$ and $\boldsymbol{w}_k$ denote the vector of the k-th column of $\boldsymbol{W}$. Given $\{\boldsymbol{D}, \boldsymbol{a}\}$ we have

$$\boldsymbol{a}^\top (\boldsymbol{I} - \boldsymbol{D})\boldsymbol{W}\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{a} = \sum_{k=1}^{d} (\boldsymbol{w}_{k,\bar{D}}^\top \boldsymbol{a}_{\bar{D}})(\boldsymbol{w}_{k,D}^\top \boldsymbol{a}_D) \tag{30}$$

We note that $\boldsymbol{w}_{k,\bar{D}}^\top \boldsymbol{a}_{\bar{D}} \sim \mathcal{N}(0, \frac{\|\boldsymbol{a}_{\bar{D}}\|^2}{d})$ and $\boldsymbol{w}_{k,D}^\top \boldsymbol{a}_D \sim \mathcal{N}(0, \frac{\|\boldsymbol{a}_D\|^2}{d})$. They are independent from each other and their product is a sub-exponential random variable, with sub-exponential norm $K = \frac{2\|\boldsymbol{a}_{\bar{D}}\|\|\boldsymbol{a}_D\|}{\pi d}$.

Definition 2. *The sub-exponential norm of X is defined to be*

$$\|X\|_{\psi_1} = \sup_{p \geq 1} p^{-1} (\mathbb{E}|X|^p)^{1/p}. \tag{31}$$

For the sum of sub-exponential random variables, we have the following Bernstein-type bound.

Lemma 6.3 (Corollary 5.17 in (Vershynin, 2010))**.** *Let $X_1, ..., X_N$ be independent centered sub-exponential random variables, and let $K = \max_i \|X_i\|_{\psi_1}$. Then, for every $\epsilon \geq 0$, we have*

$$\mathbb{P}\left\{\left|\sum_{i=1}^{N} X_i\right| \geq \epsilon N\right\} \leq 2\exp\left[-c\min\left(\frac{\epsilon^2}{K^2}, \frac{\epsilon}{K}\right) N\right] \tag{32}$$

where $c > 0$ is an absolute constant.

Using the sub-exponential Bernstein inequality, we have

$$\mathbb{P}\left\{\left|\sum_{k=1}^{d} (\boldsymbol{w}_{k,\bar{D}}^\top \boldsymbol{a}_{\bar{D}})(\boldsymbol{w}_{k,D}^\top \boldsymbol{a}_D)\right| \geq \epsilon d\right\} \leq 2\exp\left[-c\min\left(\frac{\epsilon^2}{K^2}, \frac{\epsilon}{K}\right) d\right]. \tag{33}$$

Define an event $E_3 := \{\|\boldsymbol{a}\|^2 < 2\}$ whose probability is at least $1 - e^{-(0.5-\log\sqrt{2})m} > 1 - e^{-m/7}$.

On the intersection of E_2 and E_3, we have $\|\boldsymbol{a}_D\|^2 \geq \frac{\|\boldsymbol{a}\|^2}{16}$ and hence $\frac{\|\boldsymbol{a}_D\|^2}{\|\boldsymbol{a}_{\bar{D}}\|^2} \geq \frac{1}{15}$, whose probability is at least $1 - e^{-m/8} - e^{-m/11} - e^{-m/7} > 1 - 3e^{-m/7}$.

On the event of $E_2 \cap E_3$ and taking $\epsilon = \frac{\|\boldsymbol{D}\boldsymbol{a}\|^2}{4d}$, the probability in Equation 33 is smaller than $2\exp\left[-c\frac{\pi^2 d}{960}\right]$.

Hence for the second term, we have with probability at least $1 - 3e^{-m/7} - 2e^{-c\pi^2 d/960}$,

$$|\boldsymbol{a}^\top(\boldsymbol{I} - \boldsymbol{D})\boldsymbol{W}\boldsymbol{W}^\top\boldsymbol{D}\boldsymbol{a}| \leq \frac{1}{32}. \tag{34}$$

Hence combining with the bound on the first term, we have that $\boldsymbol{a}^\top\boldsymbol{W}\boldsymbol{W}^\top\boldsymbol{D}\boldsymbol{a} \geq \frac{1}{32}$ holds with probability at least $1 - e^{-m/8} - 2e^{-m/11} - 3e^{-m/7} - 2e^{-c\pi^2 d/960}$. By taking the union bound over the training sample n, we complete the proof.

□

6.2 Proof of Lemma 3.4

Proof. We note that $\boldsymbol{h} \sim \mathcal{N}(0, \boldsymbol{I}_m)$. Hence because of the tail bound of the χ^2_m, we have

$$\mathbb{P}\{\|\boldsymbol{h}\|^2 \leq 2m\} > 1 - (2e^{-1})^{m/2} > 1 - e^{-m/7}. \tag{35}$$

Given $\boldsymbol{h}$, we have $\boldsymbol{a}^\top\boldsymbol{h} \sim \mathcal{N}(0, \|\boldsymbol{h}\|^2/m)$. On the event of $\{\|\boldsymbol{h}\|^2 \leq 2m\}$, for some constant c_2, $\mathbb{P}\{|\boldsymbol{a}^\top\boldsymbol{h}| < \sqrt{c_2 d}\} > 1 - 2\Phi(-\sqrt{c_2 d/2}) > 1 - 2e^{-c_2 d/4}$. Hence we have

$$|\boldsymbol{a}^\top\boldsymbol{h}| < \sqrt{c_2 d} \tag{36}$$

holds with probability at least $1 - e^{-m/7} - 2e^{-c_2 d/4}$.

On the event of $\{\|\boldsymbol{h}\|^2 \leq 2m\}$, we have that $\mathbb{P}\{\|\boldsymbol{D}_x\boldsymbol{h}\|^2 \leq 2m\} = 1$ and $\mathbb{P}\{|\boldsymbol{a}^\top\boldsymbol{D}_x\boldsymbol{h}| < \sqrt{c_2 d}\} > 1 - 2e^{-c_2 d/4}$.

Combining the above two terms together, we complete the proof. □

7 More Experiments

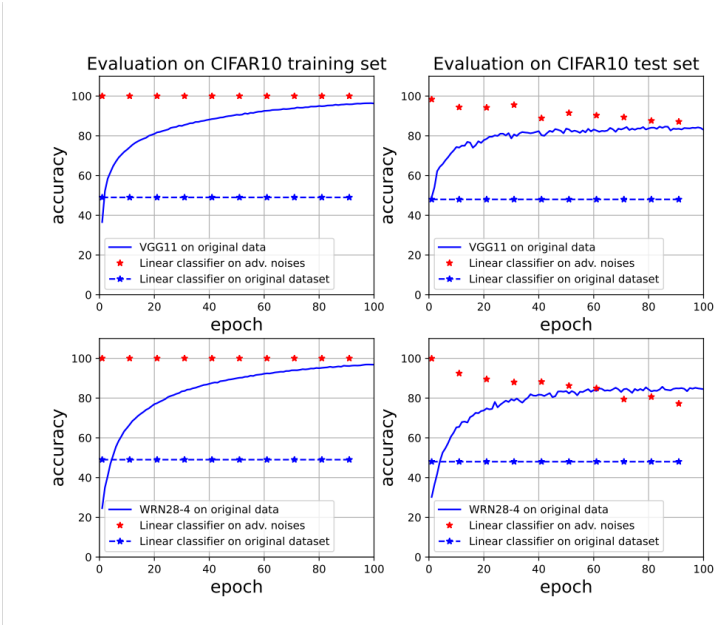

Figure 8: Training and test accuracy of linear models on adversarial noises with L_∞ bound $8/255$. The noises are generated with VGG11 (Simonyan & Zisserman, 2015) and Wide ResNet28-4 (Zagoruyko & Komodakis, 2016). We use a standard training process of SGD with $lr = 0.001$.

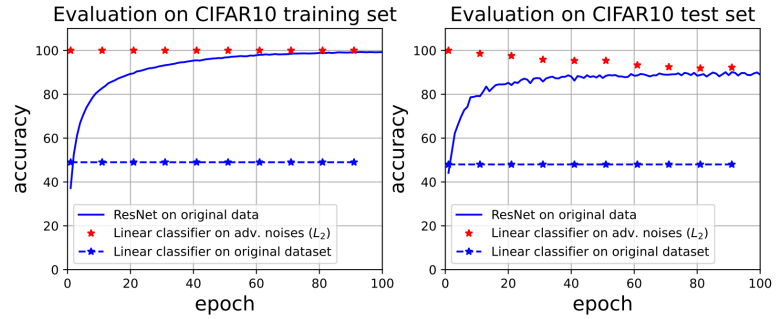

Figure 9: Training and test accuracy of linear models on adversarial noises with a L_2 norm bound 0.5. The noises are generated with ResNet-18. We use a standard training process of SGD with $lr = 0.001$.