

Relationship between manifold smoothness and adversarial vulnerability in deep learning with local errors

Zijian Jiang(蒋子健), Jianwen Zhou(周健文), and Haiping Huang(黄海平)
Citation: Chin. Phys. B, 2021, 30 (**4**): 048702. DOI: 10.1088/1674-1056/abd68e
Journal homepage: http://cpb.iphy.ac.cn; http://iopscience.iop.org/cpb

What follows is a list of articles you may be interested in

Effective suppression of beta oscillation in Parkinsonian state via a noisy direct delayed feedback control scheme

Hai-Tao Yu(于海涛), Zi-Han Meng(孟紫寒), Chen Liu(刘晨), Jiang Wang(王江), and Jing Liu(刘静)
Chin. Phys. B, 2021, 30 (**3**): 038703. DOI: 10.1088/1674-1056/abd395

Computational prediction of RNA tertiary structures using machine learning methods

Bin Huang(黄斌), Yuanyang Du(杜渊洋), Shuai Zhang(张帅), Wenfei Li(李文飞), Jun Wang (王骏), and Jian Zhang(张建)†
Chin. Phys. B, 2020, 29 (**10**): 108704. DOI: 10.1088/1674-1056/abb303

Dynamics and coherence resonance in a thermosensitive neuron driven by photocurrent

Ying Xu(徐莹), Minghua Liu(刘明华), Zhigang Zhu(朱志刚), Jun Ma(马军)
Chin. Phys. B, 2020, 29 (**9**): 098704. DOI: 10.1088/1674-1056/ab9dee

Enhancing convolutional neural network scheme forrheumatoid arthritis grading with limited clinical data

Jian Tang(汤键), Zhibin Jin(金志斌), Xue Zhou(周雪), Weijing Zhang(张玮婧), Min Wu(吴敏), Qinghong Shen(沈庆宏), Qian Cheng(程茜), Xueding Wang(王学鼎), Jie Yuan(袁杰)
Chin. Phys. B, 2019, 28 (**3**): 038701. DOI: 10.1088/1674-1056/28/3/038701

Physical mechanism of mind changes and tradeoffs among speed, accuracy, and energy cost in brain decision making: Landscape, flux, and path perspectives

Han Yan(闫晗), Kun Zhang(张坤), Jin Wang(汪劲)
Chin. Phys. B, 2016, 25 (**7**): 078702. DOI: 10.1088/1674-1056/25/7/078702

SPECIAL TOPIC — Machine learning in statistical physics

Relationship between manifold smoothness and adversarial vulnerability in deep learning with local errors*

Zijian Jiang(蒋子健), Jianwen Zhou(周健文), and Haiping Huang(黄海平)†

PMI Laboratory, School of Physics, Sun Yat-sen University, Guangzhou 510275, *China*

(Received 9 July 2020; revised manuscript received 9 December 2020; accepted manuscript online 24 December 2020)

Artificial neural networks can achieve impressive performances, and even outperform humans in some specific tasks. Nevertheless, unlike biological brains, the artificial neural networks suffer from tiny perturbations in sensory input, under various kinds of adversarial attacks. It is therefore necessary to study the origin of the adversarial vulnerability. Here, we establish a fundamental relationship between geometry of hidden representations (manifold perspective) and the generalization capability of the deep networks. For this purpose, we choose a deep neural network trained by local errors, and then analyze emergent properties of the trained networks through the manifold dimensionality, manifold smoothness, and the generalization capability. To explore effects of adversarial examples, we consider independent Gaussian noise attacks and fast-gradient-sign-method (FGSM) attacks. Our study reveals that a high generalization accuracy requires a relatively fast power-law decay of the eigen-spectrum of hidden representations. Under Gaussian attacks, the relationship between generalization accuracy and power-law exponent is monotonic, while a non-monotonic behavior is observed for FGSM attacks. Our empirical study provides a route towards a final mechanistic interpretation of adversarial vulnerability under adversarial attacks.

Keywords: neural networks, learning

PACS: 87.18.Sn, 87.19.lv **DOI:** 10.1088/1674-1056/abd68e

1. Introduction

Artificial deep neural networks have achieved the state-of-the-art performances in many domains such as pattern recognition and even natural language processing.[1] However, deep neural networks suffer from adversarial attacks,[2,3] i.e., they can make an incorrect classification with high confidence when the input image is slightly modified yet maintaining its class label. In contrast, for humans and other animals, the decision making systems in the brain are quite robust to imperceptible pixel perturbations in the sensory inputs.[4] This immediately establishes a fundamental question: what is the origin of the adversarial vulnerability of artificial neural networks? To address this question, we can first gain some insights from recent experimental observations of biological neural networks.

A recent investigation of recorded population activity in the visual cortex of awake mice revealed a power law behavior in the principal component spectrum of the population responses,[5] i.e., the n^{th} biggest principal component (PC) variance scales as $n^{-\alpha}$, where α is the exponent of the power law. In this analysis, the exponent is always slightly greater than one for all input natural-image stimuli, reflecting an intrinsic property of a smooth coding in biological neural networks. It can be proved that when the exponent is smaller than $1+2/d$, where d is the manifold dimension of the stimuli set, the neural coding manifold must be fractal,[5] and thus slightly modified inputs may cause extensive changes in the outputs. In other words, the encoding in a slow decay of population variances would capture fine details of sensory inputs, rather than an abstract concept summarizing the inputs. For a fast decay case, the population coding occurs in a smooth and differentiable manifold, and the dominant variance in the eigen-spectrum captures key features of the object identity. Thus, the coding is robust, even under adversarial attacks. Inspired by this recent study, we ask whether the power-law behavior exists in the eigen-spectrum of the correlated hidden neural activity in deep neural networks. Our goal is to clarify the possible fundamental relationship between classification accuracy, the decay rate of activity variances, manifold dimensionality, and adversarial attacks of different nature.

Taking the trade-off between biological reality and theoretical analysis, we consider a special type of deep neural network, trained with a local cost function at each layer.[6] Moreover, this kind of training offers us the opportunity to look at the aforementioned fundamental relationship at each layer. The input signal is transferred by trainable feedforward weights, while the error is propagated back to adjust the feedforward weights via random quenched weights connecting the classifier at each layer. The learning is therefore guided by the target at each layer, and layered representations are cre-

*Project supported by the National Key R&D Program of China (Grant No. 2019YFA0706302), the start-up budget 74130-18831109 of the 100-talent-program of Sun Yat-sen University, and the National Natural Science Foundation of China (Grant No. 11805284).

†Corresponding author. E-mail: huanghp7@mail.sysu.edu.cn

© 2021 Chinese Physical Society and IOP Publishing Ltd

ated due to this hierarchical learning. These layered representations offer us the neural activity space for the study of the above fundamental relationship.

We remark the motivation and relevance of our model setting, i.e., deep supervised learning with local errors. As already known, the standard back propagation widely used in machine learning is not biologically plausible.[7] The algorithm has three unrealistic (in biological sense) assumptions: (i) errors are generated from the top layer and are thus non-local; (ii) a typical network is deep, thereby requiring a memory buffer for all layers' activities; (iii) weight symmetry is assumed for both forward and backward passes. In our model setting, the errors are provided by local classifier modules and are thus local. Updating the forward weight needs only the neural state variable in the corresponding layer [see Eq. (2)], without requiring the whole memory buffer. And finally, the error is backpropagated through a fixed random projection, allowing easy implementation of breaking the weight symmetry. The learning algorithm in our paper thus bypasses the above three biological implausibilities.[6] Moreover, this model setting still allows a deep network to transform the low-level features at earlier layers into high-level abstract features at deeper layers.[6,8] Taken together, the model setting offers us the opportunity to look at the fundamental relationship between classification accuracy, the power-law decay rate of activity variances, manifold dimensionality, and adversarial vulnerability at each layer.

2. Model

The deep neural network under investigation is shown in Fig. 1. We train the network of $L+1$ layers (numbered from 0) to classify the MNIST handwritten digits.[9] The first layer serves as an input layer, receiving a 784-dimensional vector from the MINST dataset. These sensory inputs are then transferred to the next layer in a feedforward manner. The l^{th} ($l \geq 1$) layer has n_l neurons connected to a random classifier of a fixed size 10, corresponding to the number of digit classes. The random classifier means that the connection weight to the classifier is pre-sampled from a zero-mean Gaussian distribution.[6] The depth L and each layer's width n_l are flexible according to our settings. The strength of the connection or the weight between neuron i at the layer $l-1$ and neuron j at the upper layer l is denoted by w_{ij}^l, which is trainable. Similarly, at the layer l, the weight between neuron i at this layer and neuron j at the neighboring random classifier is specified by J_{ij}^l, which is quenched.

For the forward propagation, e.g., at the layer l, we have an input $\boldsymbol{I}^l$ and the pre-activation is given by $z_i^l = \sum_j w_{ji}^l I_j^l$, simply summarizing the input pixels according to the corresponding weights. To ensure that the pre-activation is of the order one, a scaling factor is often applied to the pre-activation. However, the scaling is not necessary in our task. A non-linear transfer function is then applied to the pre-activation to obtain an activation defined by $y_i^l = f(z_i^l)$. Here, we use the rectified linear unit (ReLU) function as the non-linear function, i.e., $f(x) = \max(0,x)$. The input of the next layer $l+1$ is the activation of layer l, except for the first layer where the input is a 784-dimensional vector characterizing one handwritten digit. Meanwhile, the activation of the current layer is also fed to the random classifier, resulting in a classification score vector whose components $S_i^l = \sum_j J_{ji}^l y_j^l$. The score vector can be transformed into a classification probability by applying the softmax function, i.e., $P_i^l = e^{S_i^l} / \sum_j e^{S_j^l}$. P_i^l can be understood as the probability of the input-label prediction. To specify the cost function, we first define $\boldsymbol{h}$ as the one-hot representation for the label of an image. More precisely, $h_i = \delta_{i,q}$ (Kronecker delta function), where q is the digit number of the input image. Finally, the local error function at the layer l is given by $E^l = -\sum_i h_i \ln P_i^l$. The local error is nothing but the cross entropy between $\boldsymbol{P}^l$ and $\boldsymbol{h}$. The forward propagation process can be summarized as follows:

$$
\begin{aligned}
I_i^l &= \begin{cases} y_i^{l-1}, & l > 1 \\ \text{the } i\text{-th pixel in an input}, & l = 1, \end{cases} \\
z_i^l &= \sum_j w_{ji}^l I_j^l, \\
y_i^l &= \max(0, z_i^l), \\
S_i^l &= \sum_j J_{ji}^l y_j^l, \\
P_i^l &= \mathrm{e}^{S_i^l} / \sum_j \mathrm{e}^{S_j^l}, \\
E^l &= -\sum_i h_i \ln P_i^l, \qquad (1)
\end{aligned}
$$

where $h_i = \delta_{i,q}$ (Kronecker delta function) and q is the digit label of the input image.

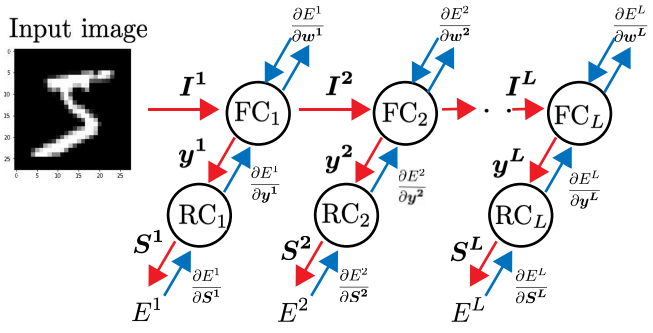


Fig. 1. Schematic illustration of deep neural networks trained with local errors. FC is the short-hand for the fully-connected module, which performs a non-linear processing. RC refers to the random classifier module, which projects the output of FC to a 10-dimensional decision space. The red and blue arrows indicate forward and backward propagations, respectively. For the forward propagation, the input image is transfered through the FCs in a layer wise manner. The FC at each layer l produces an activation $\boldsymbol{y}^l$, which serves as an input ($\boldsymbol{I}^{l+1}$) to the next layer, or is further weighted by quenched random weights of a RC ($\boldsymbol{S}^l$). After a random projection, the cross-entropy cost (E^l) is evaluated. The classification error is back-propagated to adjust the trainable network parameters, through gradients of the cost function with respect to feedforward weights ($\boldsymbol{w}^l$) between consecutive layers.

The local cost function E^l is minimized when $h_i = P_i$ for every i. The minimization is achieved by the gradient decent method. The gradient of the local error with respect to the weight of the feedforward layer can be calculated by applying the chain rule, given by

$$\frac{\partial E^l}{\partial w_{ij}^l} = \left(\sum_k (P_k^l - h_k) J_{jk}^l\right) f'(z_j^l) I_i^l. \tag{2}$$

Then, all trainable weights are updated by following $w_{ij}^l(t+1) = w_{ij}^l(t) - \eta \partial E^l / \partial w_{ij}^l$, where t and η indicate the learning step and the learning rate, respectively. In our settings, J_{ij}^l is a pre-fixed random Gaussian variable.[6] In practice, we use the mini-batch-based stochastic gradient descent method. The entire training dataset is divided into hundreds of mini-batches; after sweeping through all mini-batches, an epoch of learning is completed. The learning is then tested in an unseen/test dataset. For the standard MNIST dataset, the training set has 60000 images, and the test set has 10000 images. We can conclude whether a test image is correctly classified at the layer l by comparing the position of the maximum component of the classifier score vector $\boldsymbol{S}^l$ with that of the one-hot label $\boldsymbol{h}$, i.e., the image is correctly classified at the layer l if $\text{argmax}_i S_i^l = \text{argmax}_i h_i$. The test accuracy A_{t}^l (t means test) at layer l is then estimated by the number of correctly-classified images divided by the total number of images in the test set.

After learning, the input ensemble can be transfered throughout the network in a layer-wise manner. Then, at each layer, the activity statistics can be analyzed by the eigen-spectrum of the correlation matrix (or covariance matrix). We use principle component analysis (PCA) to obtain the eigen-spectrum, which gives variances along orthogonal directions in the descending order. For each input image, the population output of n_l neurons at the layer l can be thought of as a point in the n_l-dimensional activation space. It then follows that, for k input images, the outputs can be seen as a cloud of k points. The PCA first finds the direction with a maximal variance of the cloud, then chooses the second direction orthogonal to the first one, and so on. Finally, the PCA identifies n_l orthogonal directions and n_l corresponding variances. In our current setting, the n_l eigenvalues of the the covariance matrix of the neural manifold explain n_l variances. Arranging the n_l eigenvalues in the descending order leads to the eigen-spectrum whose behavior will be later analyzed in the next section.

To consider effects of adversarial examples, we add perturbations $\delta \boldsymbol{I}^l$ to the original input $\boldsymbol{I}^l$ at each layer. The perturbation is added in a layer-wise manner because of our model setting where a local classifier is present at each layer. The original input is obtained by the layer-wise propagation of an image from the test set. We consider two kinds of additive perturbations: one is the Gaussian white noise and the other is the fast gradient sign method (FGSM) noise,[10,11] representing black-box attacks and white-box attacks, respectively. Each component of the white noise is an *i.i.d* random number drawn from zero mean Gaussian distribution with different variances (attack/noise strength), and each component of the FGSM noise is taken from the gradient of the local cost function at each layer with respect to the immediate input of this layer. These types of perturbations are given as follows:

$$\delta \boldsymbol{I}_{\text{G}}^l(\varepsilon) = \varepsilon \cdot \boldsymbol{z}, \tag{3a}$$

$$\delta \boldsymbol{I}_{\text{F}}^l(\varepsilon) = \varepsilon \cdot \text{sgn}\left(\frac{\partial E^l}{\partial \boldsymbol{I}^l}\right), \tag{3b}$$

where $z_i \sim \mathcal{N}(0,1)$, ε denotes the perturbation magnitude, and $\text{sgn}(x) \equiv |x|/x$. In fact, the FGSM attack can be thought of as an ℓ_∞ norm ball attack around the original input image.

3. Results and discussion

In this section, we apply our model to clarify the possible fundamental relationship between classification accuracy, the decay rate of activity variances, manifold dimensionality, and adversarial attacks of different nature.

3.1. Test error decreases with depth

We first show that the deep supervised learning in our current setting works. Figure 2 shows that the training error decreases as the test accuracy increases (before early stopping) during training. We remark that it is challenging to rigorously prove the convergence of the algorithm we used in this study, as the deep learning cost landscape is highly non-convex, and the learning dynamics is non-linear in nature. As a heuristic way, we judge the convergence by the stable error rate (in the global sense), which is also common in other deep learning systems. As the layer goes deeper, the test accuracy grows until saturation despite a slight deterioration. This behavior provides an ideal candidate of deep learning to investigate the emergent properties of the layered intermediate representations after learning, without and with adversarial attacks.

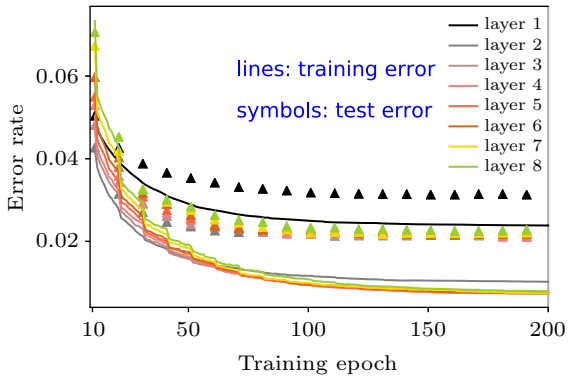


Fig. 2. Typical trajectories of training and test error rates versus training epoch. Lines indicate the train error rate, and the symbols are the test error rate. The network width of each layer is fixed to $N = 200$ (except the input layer), with 60000 images for training and 10000 images for testing. The initial learning rate $\eta = 0.5$ which is multiplied by 0.8 every ten epochs.

Next, we will study in detail how the test accuracy is related to the power-law exponent, how the test accuracy is related to the attack strength, and how the dimensionality of the layered representation changes with the exponent, under zero, weak, and strong adversarial attacks.

3.2. Power-law decay of dominant eigenvalues of the activity correlation matrix

A typical eigen-spectrum of our current deep learning model is given in Fig. 3. Notice that the eigen-spectrum is displayed in the log–log scale, then the slope of the linear fit of the spectrum gives the power-law exponent α. We use the first ten PC components to estimate α but not all for the following two reasons: (i) A waterfall phenomenon appears at the position around the 10^{th} dimension, which is more evident at higher layers. (ii) The first ten dimensions explain more than 95% of the total variance, and thus they capture the key information about the geometry of the representation manifold. The waterfall phenomenon in the eigen-spectrum can occur multiple times, especially for deeper layers [Fig. 3(a)], which is distinct from that observed in biological neural networks [see the inset of Fig. 3(a)]. This implies that the artificial deep networks may capture fine details of stimuli in a hierarchical manner. A typical example of obtaining the power-law exponent is shown in Fig. 3(b) for the fifth layer. When the stimulus size k is chosen to be large enough (e.g., $k \geq 2000$; $k = 3000$ throughout the paper), the fluctuation of the estimated exponent due to stimulus selection can be neglected.

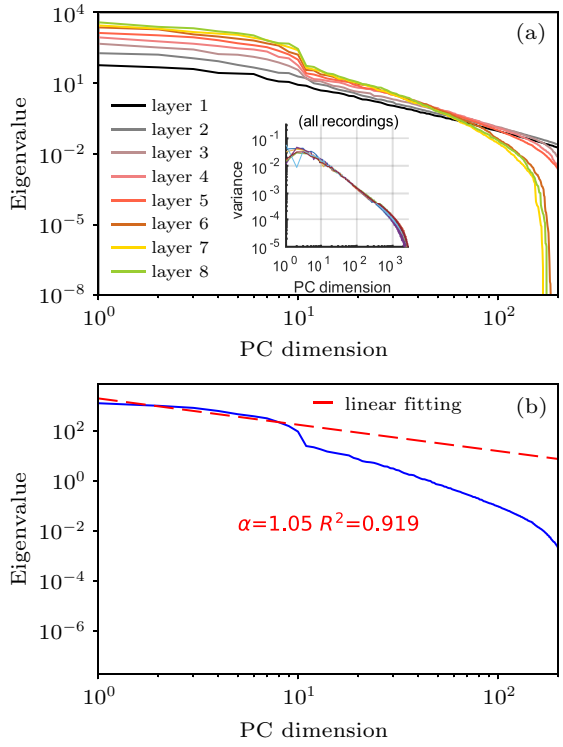


Fig. 3. Eigen-spectrum of layer-dependent correlated activities and the power-law behavior of dominant PC dimensions. (a) The typical eigen-spectrum of deep networks trained with local errors ($L = 8$, $N = 200$). Log-log scales are used. The inset is the eigen-spectrum measured in the visual cortex of mice (taken from Ref. [5]). (b) An example of extracting the power-law behavior at the fifth layer in (a). A linear fitting for the first ten PC components is shown in the log–log scale.

3.3. Effects of layer width on test accuracy and power-law exponent

We then explore the effects of the layer width on both test accuracy and power-law exponent. As shown in Fig. 4(a), the test accuracy becomes more stable with increasing layer width. This is indicated by an example of $n_l = 50$ which shows a large fluctuation of the test accuracy especially at deeper layers. We conclude that a few hundreds of neurons at each layer are sufficient for an accurate learning.

The power-law exponent also shows a similar behavior; the estimated exponent shows less fluctuations as the layer width increases. This result also shows that the exponent grows with layers. The deeper the layer is, the larger the exponent becomes. A larger exponent suggests that the manifold is smoother, because the dominant variance decays fast, leaving few space for encoding the irrelevant features in the stimulus ensemble. This may highlight that the depth in hierarchical learning is important for capturing key characteristics of sensory inputs.

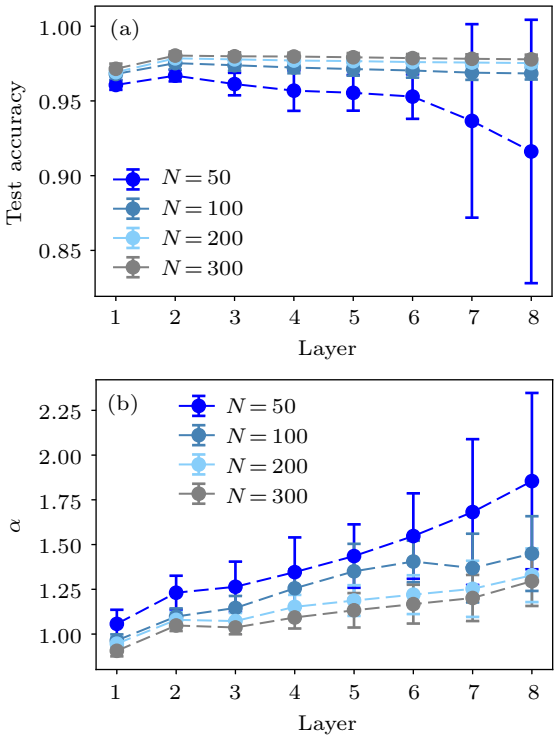


Fig. 4. Effects of network width on test accuracy and power-law exponent α. (a) Test accuracy versus layer. Error bars are estimated over 20 independently training models. (b) α versus layer. Error bars are also estimated over 20 independently training models.

3.4. Relationship between test accuracy and power-law exponent

In the following simulations, we use 200 neurons at each layer. The relationship between α and A_t^l is plotted in Fig. 5, with error bars indicating the standard errors across 50 independent trainings. The typical value of α grows along the

depth, implying that the coding manifold becomes smoother at higher layers. The clear separation of the α values between the earlier layer (e.g., the exponent of the first layer is less than one) and the deeper layers may indicate a transition from non-smooth (a slow decay of the eigen-spectrum) coding to smooth coding. Interestingly, the test accuracy does not increase with the depth for deeper layers, it just simply increases first and then quickly drops with a relatively small margin. On which layer the accuracy starts to be saturated seems to depend on the specific computational task. This suggests that too much smoothness will harm the generalization ability of the network, although the impact is not significant and thus limited.

3.5. Properties of the model under black-box attacks

We first consider the additive Gaussian white noise perturbation to the input representation at each layer. This kind of perturbation is also called the black-box attack, because it is not necessary to have access to the training details of the deep learning model, including architectures and loss function. Under this attack, the test accuracy decreases as the perturbation magnitude increases. As expected from the above results of manifold smoothness at deeper layers, the deeper layers become more robust against the black-box attacks of increasing perturbation magnitude [Figs. 6(a) and 6(b)]. The ε-dependence of the power-law exponent shows that the adversarial robustness correlates with the manifold smoothness at deeper layers, which highlights the fundamental relationship between the adversarial vulnerability and the fractal manifold.[5]

Inspecting carefully the behavior of α as a function of the test accuracy, we can identify two interesting regimes separated by a turning point at $\alpha \sim 1$ [Fig. 6(c)]. Below the turning point, increasing ε results in a clear decreasing in both A_t^l and α; Moreover, for the first three layers, a linear relationship can be identified. The linear fitting coefficient is as high as $R^2 \sim 0.99$. This observation is very interesting. The underlying mechanism is closely related to the monotonic behavior of the accuracy or exponent as a function of the attack strength. Let us define their respective functions as $f_a(\varepsilon)$ and $f_e(\varepsilon)$. Then a simple transformation leads to $\alpha = f_e(f_a^{-1}(A_t^l))$,

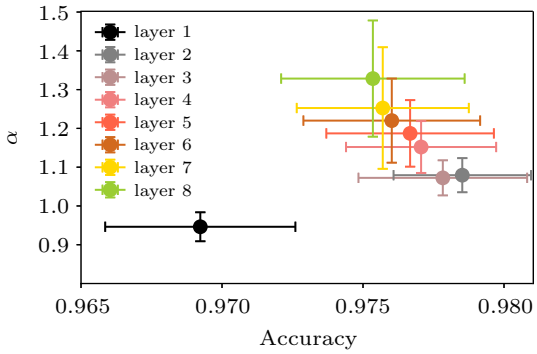


Fig. 5. The power-law exponent α versus test accuracy of the manifold. α grows along the depth, while the test accuracy has a turnover at the layer 2, and then decreases by a very small margin. Error bars are estimated over 50 independently training models.

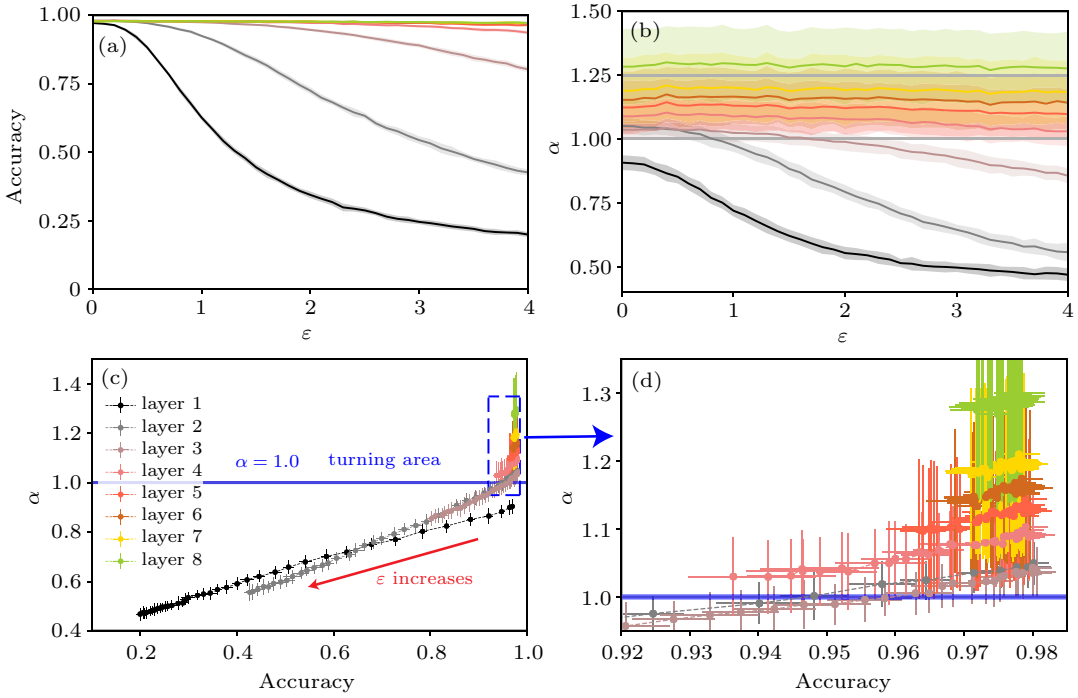


Fig. 6. Relationship between test accuracy and power-law exponent α when the input test data is attacked by independent Gaussian white noises. Error bars are estimated over 20 independently training models. (a) Accuracy versus ε. ε is the attack amplitude. (b) α versus ε. (c) Accuracy versus α over different values of ε. Different symbol colors refer to different layers. The red arrow points to the direction along which ε increases from 0.1 to 4.0, with an increment size of 0.1. The relationship of $A_t^l(\alpha)$ with increasing ε in the first three layers shows a linear function, with the slopes of 0.56, 0.86, and 1.04, respectively. The linear fitting coefficients R^2 are all larger than 0.99. Beyond the third layer, the linear relationship is not evident. For the sake of visibility, we enlarge the deeper-layer region in (d). A turning point $\alpha \approx 1.0$ appears. Above this point, the manifold seems to become smooth, and the exponent becomes stable even against stronger black-box attacks [see also (b)].

which suggests that a particular choice of f_a and f_e allows for the observed linear relationship, e.g., a linear function. Therefore, how the earlier layer is affected by adversarial examples with increasing strength plays a vital role in shaping the interesting linear regime. Above the turning point, the linear fitting fails, and instead, a non-linear relationship takes over. One key reason is that deeper layers are more robust against adversarial examples in our current setting.

3.6. Properties of the model under white-box attacks

We then consider the white-box attack, FGSM additive noise perturbation; the results are summarized in Fig. 7. The FGSM attack is much stronger than the black-box attack, as expected from the fact that the loss function and network architecture knowledge are both used by the FGSM attack. The first few layers display evident adversarial vulnerability, or completely fail to identity correct class of the input images, while the last deeper layers still show adversarial robustness to some extent. These deeper layers also maintain a relatively high value of α, although strong adversarial examples strongly reduce the manifold smoothness especially for layers next to the earlier layers. Under the FGSM attack, the α–A_t^l relationship becomes complicated, and the linear–nonlinear separation that occurs in the black-box attacks disappears. However, the high accuracy still implies a high value of α. In particular, when the white-box perturbation is weak, the system reduces the manifold smoothness by a small margin to get a high accuracy [Fig. 7(c)]. This can be interpreted as follows. Supported by the trained weights (no adversarial training), the manifold formed by the adversarial examples takes into account more details (or some special directions pointing to the decision boundary) of the adversarial examples. Under the FGSM attack, specific pixels with ℓ_∞ norm perturbations affecting strongly the loss function are particularly used to flip the decision output. In this sense, the competition (or trade-off) between the accuracy and the manifold smoothness captured by α is present. This may explain why there exists a peak in Fig. 7(c). The peak also appears in Fig. 7(b). Both types of peaks have the one-to-one correspondence. Note that black-box attacks have no such properties.

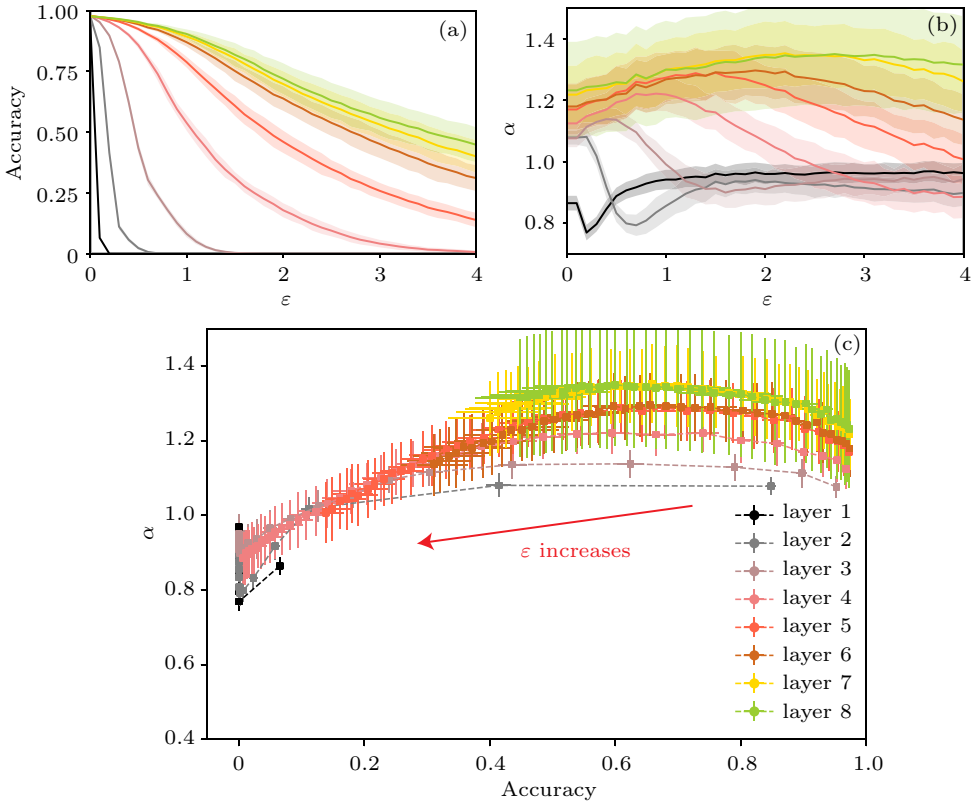


Fig. 7. Relationship between test accuracy and exponent α under the FGSM attack. Error bars are estimated over 20 independently training models. (a) A_t^l changes with ε. (b) α changes with ε. (c) A_t^l versus α over different attack magnitudes. ε increases from 0.1 to 4.0 with the increment size of 0.1. The plot shows a non-monotonic behavior different from that of white-box attacks in Fig. 6(c).

3.7. Relationship between manifold linear dimensionality and power-law exponent

The linear dimensionality of a manifold formed by data/representations can be thought of as a first approximation of intrinsic geometry of a manifold,[12,13] defined as follows:

$$D^l = \frac{\left(\sum_{i=1}^{N} \lambda_i\right)^2}{\sum_{i=1}^{N} \lambda_i^2}, \tag{4}$$

where $\{\lambda_i\}$ is the eigen-spectrum of the covariance matrix. Suppose the eigen-spectrum has a power-law decay behavior as the PC dimension increases, we simplify the dimensionality equation as follows:

$$D(\alpha) = \frac{\left(\sum_{n=1}^{N} n^{-\alpha}\right)^2}{\sum_{n=1}^{N} n^{-2\alpha}} \simeq \frac{(N^{1-\alpha}-1)^2(1-2\alpha)}{(N^{1-2\alpha}-1)(1-\alpha)^2}, \quad (5)$$

where N denotes the layer width, and the approximation $\sum_{n=1}^{N} n^{-\alpha} \simeq \int_1^N n^{-\alpha} \mathrm{d}n = \frac{N^{1-\alpha}-1}{1-\alpha}$ is used to get the second equality. In the thermodynamic limit, $D(\alpha) = [\varsigma(\alpha)]^2/\varsigma(2\alpha)$, where $\varsigma(\alpha)$ is the Reimann zeta function. Note that for a small N, a theoretical prediction of $D(\alpha)$ in Figs. 8(b) and 8(c) is obtained by using the first equality (sum).

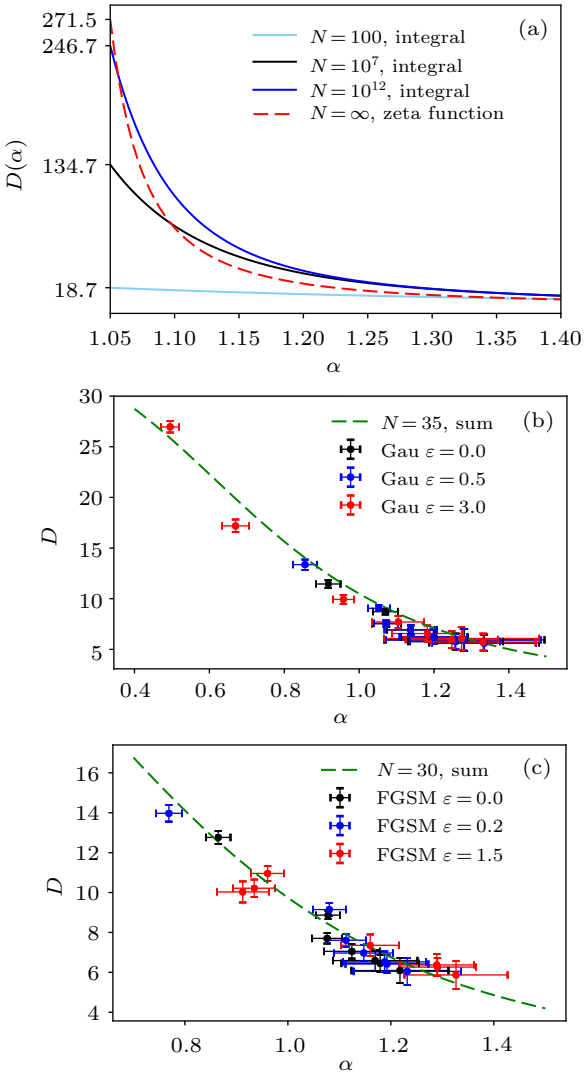


Fig. 8. Relationship between dimensionality D and power-law exponent. (a) $D(\alpha)$ estimated from the integral approximation and in the thermodynamic limit. N is the layer width. (b) $D(\alpha)$ under the Gaussian white noise attack. The dimensionality and the exponent are estimated directly from the layered representations given the immediate perturbed input for each layer [Eq. (4)]. We show three typical cases of attack: no noise with $\varepsilon = 0.0$, small noise with $\varepsilon = 0.5$, and strong noise with $\varepsilon = 3.0$. For each case, we plot eight results corresponding to eight layers. The green dashed line is the theoretical prediction [Eq. (5)], provided that $N = 35$. Error bars are estimated over 20 independently training models. (c) $D(\alpha)$ under the FGSM attack. The theoretical curve (dashed line) is computed with $N = 30$. Error bars are estimated over 20 independently training models.

The results are shown in Fig. 8. The theoretical prediction agrees roughly with simulations under zero, weak, and strong attacks of black-box and white-box types. This shows that using the power-law decay behavior of the eigen-spectrum in terms of the first few dominant dimensions to study the relationship between the manifold geometry and adversarial vulnerability of artificial neural networks is also reasonable, as also confirmed by many aforementioned non-trivial properties about this fundamental relationship. Note that when the network width increases, a deviation may be observed due to the waterfall phenomenon observed in the eigen-spectrum (see Fig. 3).

4. Conclusion

In this work, we study the fundamental relationship between the adversarial robustness and the manifold smoothness characterized by the power-law exponent of the eigen-spectrum. The eigen-spectrum is obtained from the correlated neural activity on the representation manifold. We choose deep supervised learning with local errors as our target deep learning model, because of its nice property of allowing for analyzing each layered representation in terms of both test accuracy and manifold smoothness. We then reveal that the deeper layer has a larger value of α, thereby possessing the adversarial robustness against both black- and white-box attacks. In particular, we find a turning point under the black-box attacks for the $A_t^l(\alpha)$ curve, separating linear and non-linear relationships. This turning point also signals the qualitative change of the manifold geometric property. Under the white-box attacks, the exponent-accuracy behavior becomes more complicated, as isotropic properties of attacks like in Gaussian white noise perturbations do not hold, which requires that a trade-off between manifold smoothness and test accuracy, given the normal trained weights (no adversarial training), should be taken.

All in all, although our study does not provide precise mechanisms underlying the adversarial vulnerability, the empirical works are expected to offer some intuitive arguments about the fundamental relationship between generalization capability and the intrinsic properties of representation manifolds inside the deep neural networks with biological plausibility (to some degree), encouraging future mechanistic studies towards the final goal of aligning machine perception and human perception.[4]

References

[1] Goodfellow I, Bengio Y and Courville A 2016 *Deep Learning* (Cambridge, MA: MIT Press)
[2] Carlini N and Wagner D 2017 *IEEE Symposium on Security and Privacy (SP)*, pp. 39–57
[3] Su J W, Vargas D V and Sakurai K 2019 *IEEE Transactions on Evolutionary Computation* **23** 828
[4] Zhou Z L and Firestone C 2019 *Nat. Commun.* **10** 1334

[5] Stringer C, Pachitariu M, Steinmetz N, Carandini M and Harris K D 2019 *Nature* **571** 361
[6] Mostafa H, Ramesh V and Cauwenberghs G 2018 *Frontiers in Neuroscience* **12** 608
[7] Lillicrap T P, Santoro A, Marris L, Akerman C J and Hinton G 2020 *Nature Reviews Neuroscience* **21** 335
[8] Yamins D L K and DiCarlo J 2016 *Nat. Neurosci.* **19** 356
[9] Lecun Y, Bottou L, Bengio Y and Haffner P 1998 *Proc. IEEE* **86** 2278
[10] Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow I and Fergus R 2014 *International Conference on Learning Representations (ICLR)*
[11] Goodfellow I, Shlens J and Szegedy C 2015 *International Conference on Learning Representations (ICLR)*
[12] Huang H P 2018 *Phys. Rev. E* **98** 062313
[13] Zhou J W and Huang H P 2021 *Phys. Rev. E* **103** 012315