Adversarial Robustness Guarantees for Random Deep Neural Networks: Supplementary Manuscript

Giacomo De Palma [1 2 3] **Bobak T. Kiani** [3 4] **Seth Lloyd** [2 3]

1. Proof of Theorem 1, Part II

Let

$$p_r = \mathbb{P}\left\{\exists\, x \in \mathcal{L}_r : \phi(x) = 0\right\} . \tag{1}$$

We define for any $0 \leq t \leq r$

$$f(t) = \frac{\phi(x_0 + t\, v)}{\sqrt{K(x_0 + t\, v,\, x_0 + t\, v)}} , \tag{2}$$

such that f is a centered Gaussian process on $[0, r]$ with covariance and feature map

$$\tilde{K}(s,t) = \frac{K(x_0 + s\, v,\, x_0 + t\, v)}{\sqrt{K(x_0 + s\, v,\, x_0 + s\, v)\, K(x_0 + t\, v,\, x_0 + t\, v)}} , \qquad \tilde{\Phi}(t) = \frac{\Phi(x_0 + t\, v)}{\|\Phi(x_0 + t\, v)\|} . \tag{3}$$

We have

$$\begin{aligned} \left\| \tilde{\Phi}(s) - \tilde{\Phi}(t) \right\| &= \frac{\left\| \|\Phi(x_0 + t\, v)\|\, \Phi(x_0 + s\, v) - \|\Phi(x_0 + s\, v)\|\, \Phi(x_0 + t\, v) \right\|}{\|\Phi(x_0 + s\, v)\|\, \|\Phi(x_0 + t\, v)\|} \\ &\leq \frac{\|\Phi(x_0 + s\, v) - \Phi(x_0 + t\, v)\| + \left| \|\Phi(x_0 + s\, v)\| - \|\Phi(x_0 + t\, v)\| \right|}{\|\Phi(x_0 + s\, v)\|} \\ &\leq \frac{2\, M\, |s - t|}{\|x_0 + s\, v\|_2} \leq \frac{2\, M\, |s - t|}{\|x_0\|_2 - r} . \end{aligned} \tag{4}$$

To conclude, we will need the following two results. The first is Rice's formula, which provides an upper bound to the number of zeroes of a one-dimensional Gaussian process on a given interval:

Theorem (Rice's formula (Adler & Taylor, 2009))**.** *Let $f : [0, r] \to \mathbb{R}$ be a centered Gaussian process with covariance $\tilde{K}$ such that $\tilde{K}(t, t) = 1$ for any $0 \leq t \leq r$. Then,*

$$\mathbb{E}\left| \{0 \leq t \leq r : f(t) = 0\} \right| = \frac{1}{\pi} \int_0^r \sqrt{\frac{\partial^2}{\partial s \partial t} \tilde{K}(s,t) \Big|_{s=t}}\, \mathrm{d}t . \tag{5}$$

The second is the following Lemma 1:

Lemma 1. *Let $\tilde{d}$ be the distance associated with $\tilde{K}$, and let us assume that for any $s,\, t \in [0, r]$*

$$\tilde{d}(s,t) \leq \tilde{C}\, |s - t| . \tag{6}$$

Then,

$$\frac{\partial^2}{\partial s \partial t} \tilde{K}(s,t) \Big|_{s=t} \leq \tilde{C}^2 . \tag{7}$$

[1]Scuola Normale Superiore, Pisa, Italy [2]Department of Mechanical Engineering, MIT, Cambridge MA, USA [3]Research Laboratory of Electronics, MIT, Cambridge MA, USA [4]Department of Electrical Engineering & Computer Science, MIT, Cambridge MA, USA. Correspondence to: Giacomo De Palma <giacomo.depalma@sns.it>, Bobak T. Kiani <bkiani@mit.edu>, Seth Lloyd <slloyd@mit.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

Proof. Let $\tilde{\Phi}$ be the feature map associated with $\tilde{K}$. We have

$$\begin{aligned}\frac{\partial^2}{\partial s \partial t}\tilde{K}(s,t)\bigg|_{s=t} &= \lim_{\epsilon\to 0}\frac{\left(\tilde{\Phi}(s+\epsilon)-\tilde{\Phi}(s)\right)\cdot\left(\tilde{\Phi}(t+\epsilon)-\tilde{\Phi}(t)\right)}{\epsilon^2}\\ &\leq \lim_{\epsilon\to 0}\frac{\left\|\tilde{\Phi}(s+\epsilon)-\tilde{\Phi}(s)\right\|\left\|\tilde{\Phi}(t+\epsilon)-\tilde{\Phi}(t)\right\|}{\epsilon^2} \leq \tilde{C}^2\,. \end{aligned} \tag{8}$$

□

Rice's formula and Lemma 1 imply

$$p_r \leq \mathbb{E}\left|\{0\leq t\leq r : \phi(x_0+t\,v)=0\}\right| \leq \frac{2\,M\,r}{\pi\left(\|x_0\|_2 - r\right)}\,, \tag{9}$$

and the claim follows.

2. Proof of Theorem 2

The proof of Theorem 2 is based on the following theorem, which formalizes the equivalence between deep neural networks with random weights and biases and Gaussian processes.

Theorem 1 (Master Theorem (Yang, 2019))**.** *Let $\phi^{(1)}, \ldots, \phi^{(L+1)}$ be the outputs of the layers of the random deep neural network defined in section 2 of the Main Manuscript. Let $K^{(1)}, \ldots, K^{(L+1)}$ be the kernels on $\mathbb{R}^{n_C^{(0)}\times\mathcal{I}^{(0)}}$ where $K^{(l)}$ is recursively defined as*

$$\mathbb{E}\left(\phi_{i,\alpha}^{(l)}(x)\,\phi_{j,\beta}^{(l)}(y)\right) = \delta_{ij}\,K_{\alpha,\beta}^{(l)}(x,y)\,, \qquad i,\,j=1,\ldots,n_C^{(l)}\,, \quad \alpha,\,\beta\in\mathcal{I}^{(l)}\,, \tag{10}$$

i.e., as the covariance of $\phi^{(l)}$, where the expectation is computed assuming that $\phi^{(1)}, \ldots, \phi^{(l-1)}$ are independent centered Gaussian processes with covariances $K^{(1)}, \ldots, K^{(l-1)}$.

Given $M\in\mathbb{N}$, let $\psi:\mathbb{R}^M\to\mathbb{R}$ be such that there exist $A,\,a,\,\epsilon>0$ such that for any $z\in\mathbb{R}^M$,

$$|\psi(z)| \leq \exp\left(A\,\|z\|_2^{2-\epsilon}+a\right)\,. \tag{11}$$

Then, in the limit $n_C^{(1)}, \ldots, n_C^{(L+1)}\to\infty$, we have for any $x^1, \ldots, x^M\in\mathbb{R}^{n_C^{(0)}\times\mathcal{I}^{(0)}}$

$$\frac{1}{n_C^{(L+1)}}\sum_{i=1}^{n_C^{(L+1)}}\psi\left(\phi_i^{(L+1)}\left(x^1\right),\ldots,\phi_i^{(L+1)}\left(x^M\right)\right) \xrightarrow{\text{a.s.}} \mathbb{E}_{Z\sim\mathcal{N}(0,\Sigma)}\,\psi(Z)\,, \tag{12}$$

with the covariance matrix Σ given by

$$\Sigma_{mm'} = K^{(L+1)}\left(x^m, x^{m'}\right)\,, \qquad m,\,m'=1,\ldots,M\,. \tag{13}$$

Remark 1. For finite width, the outputs of the intermediate layers of the random deep neural networks have a sub-Weibull distribution (Vladimirova et al., 2019).

The main consequence of Theorem 1 is that the final output ϕ is a centered Gaussian process:

Corollary 1. *The final output of the deep neural network ϕ is a centered Gaussian process with covariance $K=K^{(L+1)}$.*

Proof. Given $M\in\mathbb{N}$, let $\psi:\mathbb{R}^M\to\mathbb{R}$ be continuous and bounded. For any $x^1, \ldots, x^M\in\mathbb{R}^{n_C^{(0)}\times\mathcal{I}^{(0)}}$ we have from Theorem 1 in the limit $n_C^{(1)}, \ldots, n_C^{(L+1)}\to\infty$

$$\frac{1}{n_C^{(L+1)}}\sum_{i=1}^{n_C^{(L+1)}}\psi\left(\phi_i^{(L+1)}\left(x^1\right),\ldots,\phi_i^{(L+1)}\left(x^M\right)\right) \xrightarrow{\text{a.s.}} \mathbb{E}_{Z\sim\mathcal{N}(0,\Sigma)}\,\psi(Z)\,, \tag{14}$$

with Σ as in (13). Taking the expectation value on both sides of (14) we get, recalling that each $\phi_i^{(L+1)}$ has the same probability distribution as ϕ,

$$\lim_{n_C^{(1)},\ldots,n_C^{(L)}\to\infty} \mathbb{E}\,\psi\left(\phi\left(x^1\right),\ldots,\phi\left(x^M\right)\right) = \mathbb{E}_{Z\sim\mathcal{N}(0,\Sigma)}\,\psi(Z)\,, \tag{15}$$

and the claim follows. □

It is convenient to define for any $l = 1, \ldots, L$

$$K^{(l)}(x,y) = \sum_{\alpha\in\mathcal{I}^{(l)}} K^{(l)}_{\alpha,\alpha}(x,y)\,. \tag{16}$$

Let also

$$d^{(l)}(x,y)^2 = K^{(l)}(x,x) - 2\,K^{(l)}(x,y) + K^{(l)}(y,y) \tag{17}$$

be the RKHS distance associated with $K^{(l)}$.

We will prove by induction that for any $l = 1, \ldots, L+1$ there exist $C^{(l)}$, $M^{(l)} > 0$ such that $K^{(l)}$ satisfies Eq. (8) of the Main Manuscript with $C = C^{(l)}$ and $M = M^{(l)}$. The following subsections will prove the inductive step for each of the types of layer defined in section 2 of the Main Manuscript.

2.1. Input Layer

$\phi^{(1)}(x)$ is a centered Gaussian process with covariance as in (10) with

$$K^{(1)}_{\alpha,\beta}(x,y) = {\sigma_b^{(1)}}^2 + {\sigma_W^{(1)}}^2 \sum_{i=1}^{n_C^{(0)}} \sum_{\gamma\in\mathcal{P}^{(1)}} \frac{x_{i,\alpha+\gamma}\,y_{i,\beta+\gamma}}{n_C^{(0)}}\,, \qquad x,\,y\in\mathbb{R}^{n_C^{(0)}\times\mathcal{I}^{(0)}},\quad \alpha,\,\beta\in\mathcal{I}^{(1)}\,, \tag{18}$$

and

$$K^{(1)}(x,y) = \left|\mathcal{I}^{(1)}\right| {\sigma_b^{(1)}}^2 + \frac{\left|\mathcal{P}^{(1)}\right| {\sigma_W^{(1)}}^2}{n_C^{(0)}}\,x\cdot y\,, \tag{19}$$

therefore

$$\begin{aligned} d^{(1)}(x,y)^2 &= \frac{\left|\mathcal{P}^{(1)}\right| {\sigma_W^{(1)}}^2}{n_C^{(0)}}\,\|x-y\|_2^2\,,\\ K^{(1)}(x,x) &= \left|\mathcal{I}^{(1)}\right| {\sigma_b^{(1)}}^2 + \frac{\left|\mathcal{P}^{(1)}\right| {\sigma_W^{(1)}}^2}{n_C^{(0)}}\,\|x\|_2^2 \geq \frac{\left|\mathcal{P}^{(1)}\right| {\sigma_W^{(1)}}^2}{n_C^{(0)}}\,\|x\|_2^2\,, \end{aligned} \tag{20}$$

and $K^{(1)}$ satisfies Eq. (8) of the Main Manuscript with

$$C^{(1)} = \sigma_W^{(1)}\sqrt{\frac{\left|\mathcal{P}^{(1)}\right|}{n_C^{(0)}}}\,, \qquad M^{(1)} = \sqrt{\frac{\left|\mathcal{I}^{(0)}\right|}{\left|\mathcal{I}^{(1)}\right|}} = 1\,. \tag{21}$$

2.2. Nonlinear Layer

Let the $(l+1)$-th layer be a nonlinear layer. From Theorem 1, we can assume that $\phi^{(l)}$ is the centered Gaussian process with covariance $K^{(l)}$. We then have

$$K^{(l+1)}_{\alpha,\beta}(x,y) = {\sigma_b^{(l+1)}}^2 + {\sigma_W^{(l+1)}}^2 \sum_{\gamma\in\mathcal{P}^{(l+1)}} \mathbb{E}\left(\tau(u)\,\tau(v) : (u,v)\sim\mathcal{N}\left(0,\Sigma^{(l)}_{\alpha,\beta}(x,y)\right)\right)\,, \tag{22}$$

with

$$\Sigma^{(l)}_{\alpha,\beta}(x,y) = \begin{pmatrix} K^{(l)}_{\alpha+\gamma,\alpha+\gamma}(x,x) & K^{(l)}_{\alpha+\gamma,\beta+\gamma}(x,y) \\ K^{(l)}_{\beta+\gamma,\alpha+\gamma}(y,x) & K^{(l)}_{\beta+\gamma,\beta+\gamma}(y,y) \end{pmatrix}\,. \tag{23}$$

If τ is the ReLU activation function, (22) simplifies to

$$K^{(l+1)}_{\alpha,\beta}(x,y) = {\sigma_b^{(l+1)}}^2 + \frac{{\sigma_W^{(l+1)}}^2}{2} \sum_{\gamma \in \mathcal{P}^{(l+1)}} V^{(l)}_{\alpha+\gamma,\beta+\gamma}(x,y)\,, \tag{24}$$

where

$$V^{(l)}_{\alpha,\beta}(x,y) = \sqrt{K^{(l)}_{\alpha,\alpha}(x,x)\,K^{(l)}_{\beta,\beta}(y,y)}\;\Psi\left(\frac{K^{(l)}_{\alpha,\beta}(x,y)}{\sqrt{K^{(l)}_{\alpha,\alpha}(x,x)\,K^{(l)}_{\beta,\beta}(y,y)}}\right)\,, \tag{25}$$

with $\Psi : [-1,1] \to \mathbb{R}$ given by

$$\Psi(t) = \frac{\sqrt{1-t^2} + (\pi - \arccos t)\,t}{\pi}\,. \tag{26}$$

Remark 2. Since $\Psi(t) \geq 0$ for any $-1 \leq t \leq 1$, we have from (25) and (24) that $K^{(l+1)}_{\alpha,\beta}(x,y) \geq 0$ for any $x,\, y \in \mathbb{R}^{n^{(0)}} \times \mathcal{I}^{(0)}$ and any $\alpha,\, \beta \in \mathcal{I}^{(l+1)} = \mathcal{I}^{(l)}$.

Let

$$V^{(l)}(x,y) = \sum_{\alpha \in \mathcal{I}} V^{(l)}_{\alpha,\alpha}(x,y)\,. \tag{27}$$

Since $\Psi(1) = 1$, we get

$$V^{(l)}(x,x) = K^{(l)}(x,x)\,. \tag{28}$$

Since $\Psi(t) \geq t$ for any $-1 \leq t \leq 1$, we also get

$$V^{(l)}(x,y) \geq K^{(l)}(x,y)\,. \tag{29}$$

Moreover,

$$\begin{aligned} K^{(l+1)}(x,y) &= \left|\mathcal{I}^{(l+1)}\right| {\sigma_b^{(l+1)}}^2 + \frac{\left|\mathcal{P}^{(l+1)}\right| {\sigma_W^{(l+1)}}^2}{2}\, V^{(l)}(x,y) \\ &\geq \left|\mathcal{I}^{(l+1)}\right| {\sigma_b^{(l+1)}}^2 + \frac{\left|\mathcal{P}^{(l+1)}\right| {\sigma_W^{(l+1)}}^2}{2}\, K^{(l)}(x,y)\,, \end{aligned} \tag{30}$$

with equality for $y = x$. We then have

$$K^{(l+1)}(x,x) \geq \frac{\left|\mathcal{P}^{(l+1)}\right| {\sigma_W^{(l+1)}}^2}{2}\, K^{(l)}(x,x) \geq \frac{\left|\mathcal{P}^{(l+1)}\right| {\sigma_W^{(l+1)}}^2 {C^{(l)}}^2}{2}\, \|x\|_2^2\,, \tag{31}$$

where we have used the inductive hypothesis. Moreover,

$$d^{(l+1)}(x,y)^2 \leq \frac{\left|\mathcal{P}^{(l+1)}\right| {\sigma_W^{(l+1)}}^2}{2}\, d^{(l)}(x,y)^2 \leq \frac{\left|\mathcal{P}^{(l+1)}\right| {\sigma_W^{(l+1)}}^2 \left|\mathcal{I}^{(0)}\right| {C^{(l)}}^2}{2\left|\mathcal{I}^{(l)}\right|}\, \|x-y\|_2^2\,, \tag{32}$$

where we have used the inductive hypothesis again, and $K^{(l+1)}$ satisfies Eq. (8) of the Main Manuscript with

$$C^{(l+1)} = \sigma_W^{(l+1)} \sqrt{\frac{\left|\mathcal{P}^{(l+1)}\right|}{2}}\, C^{(l)}\,, \qquad M^{(l+1)} = \sqrt{\frac{\left|\mathcal{I}^{(0)}\right|}{\left|\mathcal{I}^{(l)}\right|}} = \sqrt{\frac{\left|\mathcal{I}^{(0)}\right|}{\left|\mathcal{I}^{(l+1)}\right|}}\,. \tag{33}$$

2.3. Skipped Connection

Let the $(l+1)$-th layer be a skipped connection. We have

$$K^{(l+1)}_{\alpha,\beta}(x,y) = K^{(l)}_{\alpha,\beta}(x,y) + K^{(l-k)}_{\alpha,\beta}(x,y)\,, \tag{34}$$

and

$$K^{(l+1)}(x,y) = K^{(l)}(x,y) + K^{(l-k)}(x,y)\,. \tag{35}$$

Since in section 2 of the Main Manuscript we have imposed $k \leq l-2$, we have from Remark 2 $K^{(l+1)}_{\alpha,\beta}(x,y) \geq 0$ for any $x,\, y \in \mathbb{R}^{n^{(0)}} \times \mathcal{I}^{(0)}$ and any $\alpha,\, \beta \in \mathcal{I}^{(l+1)}$. We then have

$$K^{(l+1)}(x,x) = K^{(l)}(x,x) + K^{(l-k)}(x,x) \geq \left(C^{(l)^2} + C^{(l-k)^2}\right) \|x\|_2^2\,, \tag{36}$$

where we have used the inductive hypothesis. Moreover,

$$d^{(l+1)}(x,y)^2 = d^{(l)}(x,y)^2 + d^{(l-k)}(x,y)^2 \leq \frac{\left|\mathcal{I}^{(0)}\right|}{\left|\mathcal{I}^{(l)}\right|}\left(C^{(l)^2} + C^{(l-k)^2}\right)\|x-y\|_2^2\,, \tag{37}$$

where we have used the inductive hypothesis again, and $K^{(l+1)}$ satisfies Eq. (8) of the Main Manuscript with

$$C^{(l+1)} = \sqrt{C^{(l)^2} + C^{(l-k)^2}}\,, \qquad M^{(l+1)} = \sqrt{\frac{\left|\mathcal{I}^{(0)}\right|}{\left|\mathcal{I}^{(l)}\right|}} = \sqrt{\frac{\left|\mathcal{I}^{(0)}\right|}{\left|\mathcal{I}^{(l+1)}\right|}}\,. \tag{38}$$

2.4. Pooling

Let the $(l+1)$-th layer be a pooling layer. Since in the architecture defined in section 2 of the Main Manuscript we have imposed the l-th layer to be a nonlinear convolutional layer, from Remark 2 we have $K^{(l)}_{\beta,\gamma}(x,x) \geq 0$ for any $\beta,\, \gamma \in \mathcal{I}^{(l)}$. We can assume from Theorem 1 that $\phi^{(l)}$ is a Gaussian process with covariance $K^{(l)}$. We then have

$$K^{(l+1)}_{\alpha,\beta}(x,y) = \sum_{\gamma\in\alpha,\,\delta\in\beta} K^{(l)}_{\gamma,\delta}(x,y)\,, \qquad \alpha,\, \beta \in \mathcal{I}^{(l+1)}\,, \tag{39}$$

and

$$K^{(l+1)}(x,y) = \sum_{\alpha\in\mathcal{I}^{(l+1)}} \sum_{\beta,\,\gamma\in\alpha} K^{(l)}_{\beta,\gamma}(x,y)\,. \tag{40}$$

We have from the inductive hypothesis

$$d^{(l+1)}(x,y)^2 \leq \frac{\left|\mathcal{I}^{(l)}\right|}{\left|\mathcal{I}^{(l+1)}\right|} d^{(l)}(x,y)^2 \leq \frac{\left|\mathcal{I}^{(0)}\right|}{\left|\mathcal{I}^{(l+1)}\right|} C^{(l)^2}\|x-y\|_2^2\,. \tag{41}$$

Moreover, since $K^{(l)}_{\beta,\gamma}(x,x) \geq 0$, we have

$$K^{(l+1)}(x,x) \geq \sum_{\alpha\in\mathcal{I}^{(l+1)}} \sum_{\beta\in\alpha} K^{(l)}_{\beta,\beta}(x,x) = K^{(l)}(x,x) \geq C^{(l)^2}\|x\|_2^2\,, \tag{42}$$

where we have used the inductive hypothesis again, and $K^{(l+1)}$ satisfies Eq. (8) of the Main Manuscript with

$$C^{(l+1)} = C^{(l)}\,, \qquad M^{(l+1)} = \sqrt{\frac{\left|\mathcal{I}^{(0)}\right|}{\left|\mathcal{I}^{(l+1)}\right|}}\,. \tag{43}$$

2.5. Flattening

From Theorem 1, we can assume that $\phi^{(L)}$ is the centered Gaussian process with covariance $K^{(L)}$. The proof is completely analog to the proof in subsection 2.2, and $K^{(L+1)}$ satisfies Eq. (8) of the Main Manuscript with

$$C = C^{(L+1)} = \sigma_W^{(L+1)}\sqrt{\frac{\left|\mathcal{I}^{(L_f)}\right|}{2}}\,C^{(L)}\,, \qquad M = M^{(L+1)} = \sqrt{\frac{\left|\mathcal{I}^{(0)}\right|}{\left|\mathcal{I}^{(L_f)}\right|}}\,. \tag{44}$$

3. Proof of Theorem 3

The upper bound for $0 < \epsilon \leq \frac{1}{\sqrt{n}}$ has been proven in (Price, 2016). Let $\frac{1}{\sqrt{n}} < \epsilon < 1$, and let $m \in \mathbb{N}$ be such that

$$2 \leq m \leq n\,, \qquad \frac{1}{m} \leq \epsilon^2 < \frac{1}{m-1}\,. \tag{45}$$

We consider the lattice

$$\mathcal{L}_m = \frac{\mathbb{Z}^n}{m} \cap \mathcal{B}_1\,. \tag{46}$$

For any $x \in \mathcal{B}_1$, there exists $y \in \mathcal{L}_m$ such that for any $i = 1, \ldots, n$

$$|x_i - y_i| \leq \min\left(\frac{1}{m},\, |x_i|\right)\,. \tag{47}$$

We have

$$\begin{aligned}(x-y)^2 &\leq \frac{1}{m^2}\left|\left\{i : |x_i| > \frac{1}{m}\right\}\right| + \sum_{i\,:\,|x_i|\leq\frac{1}{m}} |x_i|^2 \leq \frac{1}{m}\sum_{i:|x_i|>\frac{1}{m}} |x_i| + \frac{1}{m}\sum_{i\,:\,|x_i|\leq\frac{1}{m}} |x_i| = \frac{\|x\|_1}{m} \\ &\leq \frac{1}{m} \leq \epsilon^2\,,\end{aligned} \tag{48}$$

therefore $N(\epsilon) \leq |\mathcal{L}_m|$. The claim follows since

$$|\mathcal{L}_m| = \sum_{k=0}^{m-1} 2^k \binom{n}{k}\binom{m-1}{k} \leq (2\,n)^{m-1} \leq (2\,n)^{\frac{1}{\epsilon^2}}\,. \tag{49}$$

4. Lemmas

4.1. Proof of Lemma 1

Since $\hat{K} \leq K$ we have

$$\begin{aligned}d(x,y)^2 &= \begin{pmatrix} 1 \\ -1 \end{pmatrix}\begin{pmatrix} K(x,x) & K(x,y) \\ K(y,x) & K(y,y)\end{pmatrix}\begin{pmatrix} 1 & -1 \end{pmatrix} \\ &\geq \begin{pmatrix} 1 \\ -1 \end{pmatrix}\begin{pmatrix} \hat{K}(x,x) & \hat{K}(x,y) \\ \hat{K}(y,x) & \hat{K}(y,y)\end{pmatrix}\begin{pmatrix} 1 & -1 \end{pmatrix} = \hat{d}(x,y)^2\,.\end{aligned} \tag{50}$$

4.2. Proof of Lemma 2

We have for any $x \in \mathcal{B}_r^1$

$$\begin{aligned}|K(x,x_0) - K(x_0,x_0)| &= |(\Phi(x) - \Phi(x_0)) \cdot \Phi(x_0)| \leq \|\Phi(x) - \Phi(x_0)\|\,\|\Phi(x_0)\| \\ &\leq M\,C\,\|x - x_0\|_2\,\sqrt{K(x_0,x_0)} \leq M\,C\,r\sqrt{K(x_0,x_0)}\,,\end{aligned} \tag{51}$$

where we have used Eq. (8) of the Main Manuscript and that $\|x - x_0\|_2 \leq r$, and the claim follows.

4.3. Proof of Lemma 3

We have for any $x \in \mathcal{B}_r^1$

$$\hat{K}(x,x) = \|\Phi(x) - \Phi(x_0)\|^2 - \frac{(K(x_0,x_0) - K(x,x_0))^2}{K(x_0,x_0)} \leq M^2\,C^2\,\|x - x_0\|^2 \leq M^2\,C^2\,r^2\,, \tag{52}$$

where we have used Eq. (8) of the Main Manuscript.

5. Experimental Details

5.1. Adversarial Attack Methods

To find adversarial examples, various different adversarial attack methods were used. The list of adversarial attack used for each norm are given in Table 1. Hyperparameters for adversarial attack algorithms were also varied to find adversarial examples at the smallest norm possible. All attacks were performed using the python package Foolbox (Rauber et al., 2018).

For adversarial attacks on random neural networks (Main Manuscript, section 4), attacks were performed on random inputs where each input was sampled from the uniform distribution bounded by $[0, 1]$. For random neural networks, weights were chosen randomly according to a normal distribution with variance scaled depending on the number of neurons in prior and posterior layers (Glorot & Bengio, 2010). For each random neural network constructed, attacks were performed on batches of 10 randomized inputs. To be consistent with attacks in the adversarial literature, neural networks were constructed with two output neurons to perform binary classification. No activation (*e.g.,* softmax) was included in the final layer and attacks were directly performed on output logits.

Inputs were bounded by $[0, 1]$ in every dimension or pixel. Thus, attack algorithms were restricted in their operation within these bounds. In the case of random inputs, inputs were adversarially attacked to change their binary classification. In the cases where train or test data was provided (*i.e.,* in the cases of MNIST and CIFAR10), inputs were adversarially attacked to change the classification provided by the network (not necessarily the correct classification).

Adversarial Attack Methods	
ℓ_1 **Norm Attacks**	EAD Attack (Zhao et al., 2018)
	Pointwise Attack (Schott et al., 2018)
	Saliency Map Attack (Papernot et al., 2015)
	Sparse L1 Basic Iterative Method (Tramer & Boneh, 2019)
ℓ_2 **Norm Attacks**	Basic Iterative Method (Kurakin et al., 2018)
	Carlini Wagner Attack (Carlini & Wagner, 2017)
	Decoupled Direction Norm Attack (Rony et al., 2019)
ℓ_∞ **Norm Attacks**	Basic Iterative Method (Kurakin et al., 2018)
	Momentum Iterative Method (Dong et al., 2018)
	Adam Projected Gradient Descent (Madry et al., 2019; Carlini & Wagner, 2017)

Table 1. List of attack algorithms used to find the closest adversarial example for each norm. Attack algorithms were performed with varying hyperparameters. Among the adversarial examples given by the various attacks, the adversarial example with the smallest distance norm from the starting point is assumed to be the closest adversarial example.

5.2. Network Architectures

Various different networks were empirically studied. Layer sequences for the various networks are provided in Table 2. Weights in all networks were initialized randomly with variance inversely proportional to the size of the previous layer, often termed He initialization (He et al., 2015). In all cases, to conform to the standards provided in the Foolbox toolbox (Rauber et al., 2018), output layers contained two neurons, one for each class in the binary classification task. In all cases, the nonlinear activation used was the rectified linear unit (ReLU).

Inputs to all the networks are assumed to be 2-dimensional images with 3 channels. The only exception to this case is for networks trained on MNIST data where inputs have only one channel.

5.3. Trained Networks

For analysis on trained networks (Main Manuscript, subsection 4.1), networks were trained on either MNIST or CIFAR10 data under the task of binary classification. A softmax activation was placed on the two neurons of the last layer of all networks. For the case of MNIST, the binary classification task was determining if a digit is odd or even. For CIFAR10, image classes were assigned to binary categories of either {airplane, bird, deer, frog, ship} or {automobile, cat, dog, horse, truck}. All networks were trained to minimize categorical cross-entropy using the Adam optimizer (Kingma & Ba, 2014). Batch normalization was not included in any networks and none of the networks used dropout (Srivastava et al., 2014). For CIFAR10 data, networks were trained for 25 epochs on the complete training set with a learning rate of 0.0001. For MNIST

Simplified Residual Network	"LeNet" Style (LeCun et al., 1998) Network	Fully Connected Network	Simple Convolutional Network
Residual Block* (32 channels)	3x3 Conv - ReLU (128 Channels)	Flatten	3x3 Conv - ReLU (128 Channels)
Residual Block** (64 channels)	3x3 Conv - ReLU (128 Channels)	Fully Connected - ReLU (100 Neurons)	3x3 Conv - ReLU (128 Channels)
Residual Block** (128 channels)	2x2 Average Pooling	Fully Connected - ReLU (100 Neurons)	3x3 Conv - ReLU (128 Channels)
Flatten	3x3 Conv - ReLU (128 Channels)	Output Layer	3x3 Conv - ReLU (128 Channels)
Output Layer	3x3 Conv - ReLU (128 Channels)		Flatten
	2x2 Average Pooling		Output Layer
	Flatten		
	Fully Connected - ReLU (512 Neurons)		
	Fully Connected - ReLU (512 Neurons)		
	Output Layer		

* residual blocks contain 2 3x3 convolutional layers each followed by ReLU activation

** first convolution layer in residual block has stride set to 2 (feature map size is halved)

Table 2. Layer sequences for the various networks empirically studied. Layer sequences should be read from top to bottom.

data, networks were trained for 15 epochs on the complete training set with a learning rate of 0.0001. For the simplified residual network, trained networks achieved an average binary classification accuracy of 99.8% and 99.4% on the MNIST training and test sets respectively. Furthermore, the same network architecture trained on CIFAR10 achieved an average of 98.8% and 82.5% accuracy on training and test sets.

Adversarial attacks were performed on batches of 20 random images or 20 randomly chosen images from the training or test sets. Median adversarial distances in Figure 3 of the Main Manuscript were taken from a sample size of 5000 points – 250 trained networks each attacking 20 random images.

6. Supplementary Figures

Adversarial attacks on random networks were performed on various architectures. In this section, we include figures for attacks performed on networks not included in the Main Manuscript.

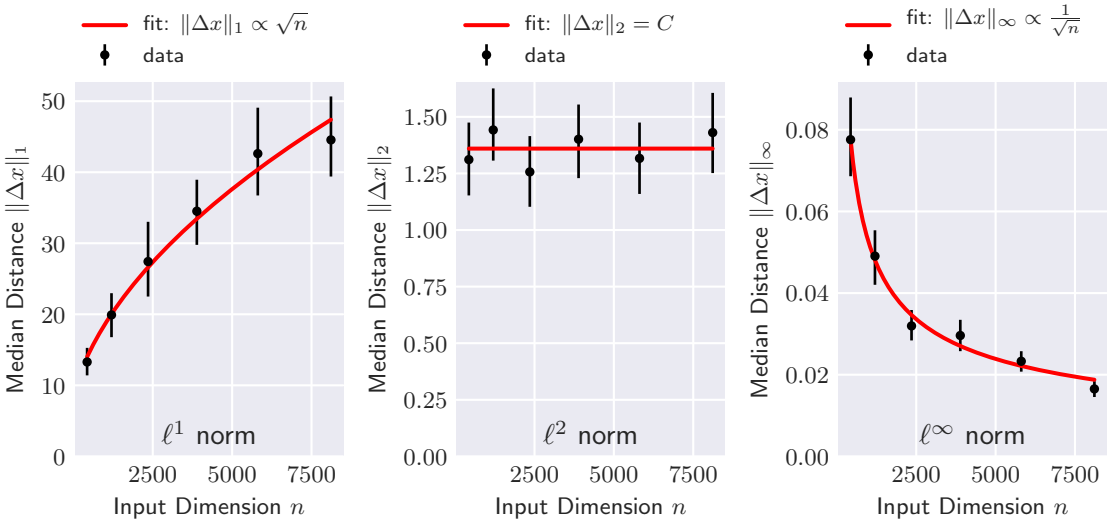

Figure 1. **Random untrained convolutional networks:** The median ℓ^p distances of closest adversarial examples from their respective inputs for $p = 1, 2, \infty$ scale as predicted in Remark 5 of the Main Manuscript for a "LeNet" (LeCun et al., 1998) architecture (see subsection 5.2 for full description of network). Error bars span ± 5 percentiles from the median. For each input dimension, results are calculated from 2000 samples (200 random networks each attacked at 10 random points). See section 5 for further details on how experiments were performed.

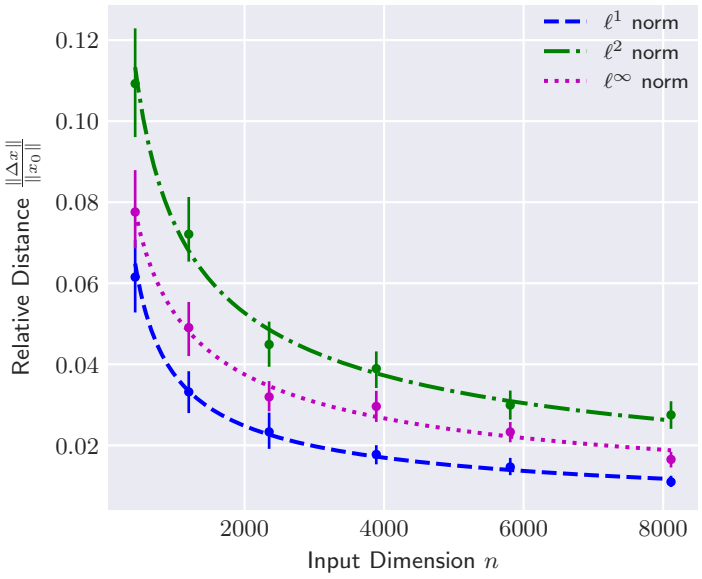

Figure 2. **Random untrained networks:** Median relative distance of closest adversarial examples $\|\Delta x\|_p / \|x_0\|_p$ from their respective inputs ($p \in \{1, 2, \infty\}$) scale with the input dimension n as $O(1/\sqrt{n})$ in all norms for a "LeNet" (LeCun et al., 1998) architecture (see subsection 5.2 for full description of network). Error bars span ± 5 percentiles from the median. For each input dimension, results are calculated from 2000 samples (200 random networks each attacked at 10 random points).

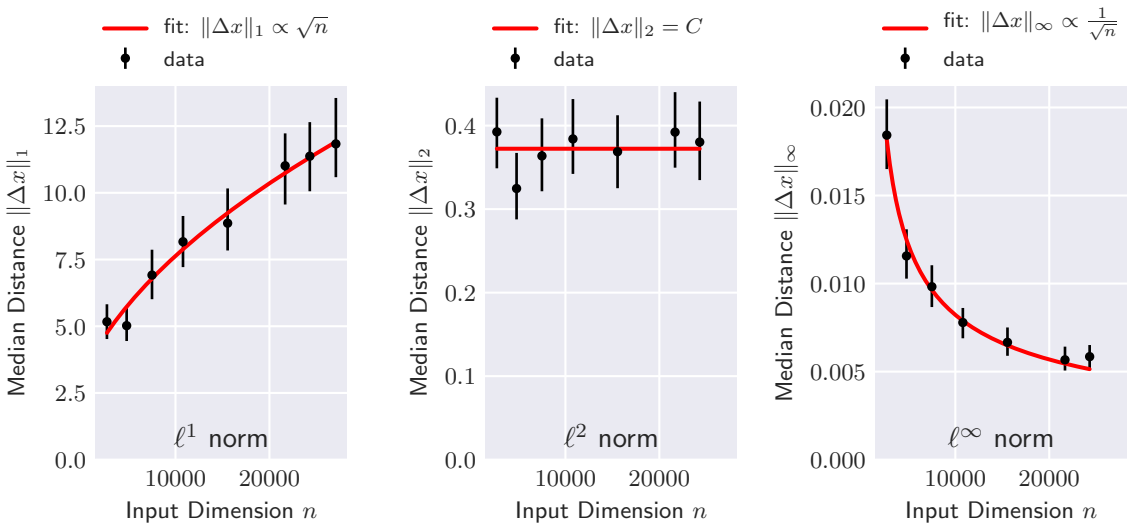

Figure 3. **Random untrained networks:** Median distance of closest adversarial examples $\|\Delta x\|_p$ from their respective inputs ($p \in \{1, 2, \infty\}$) scale as predicted in Remark 5 of the Main Manuscript for a fully connected network (see subsection 5.2 for full description of network). Error bars span ± 5 percentiles from the median. For each input dimension, results are calculated from 2000 samples (200 random networks each attacked at 10 random points). See section 5 for further details on how experiments were performed.

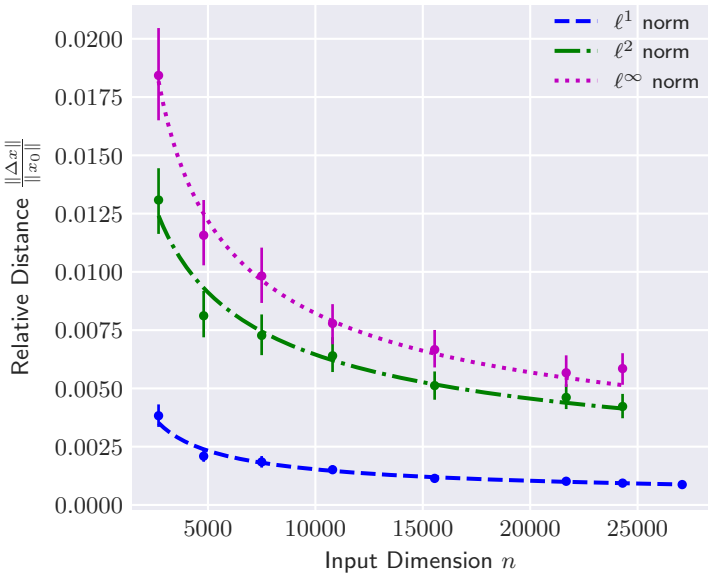

Figure 4. **Random untrained networks:** Median relative distance of closest adversarial examples $\|\Delta x\|_p / \|x_0\|_p$ from their respective inputs ($p \in \{1, 2, \infty\}$) scale with the input dimension n as $O(1/\sqrt{n})$ in all norms for a fully connected network (see subsection 5.2 for full description of network). Error bars span ± 5 percentiles from the median. For each input dimension, results are calculated from 2000 samples (200 random networks each attacked at 10 random points).

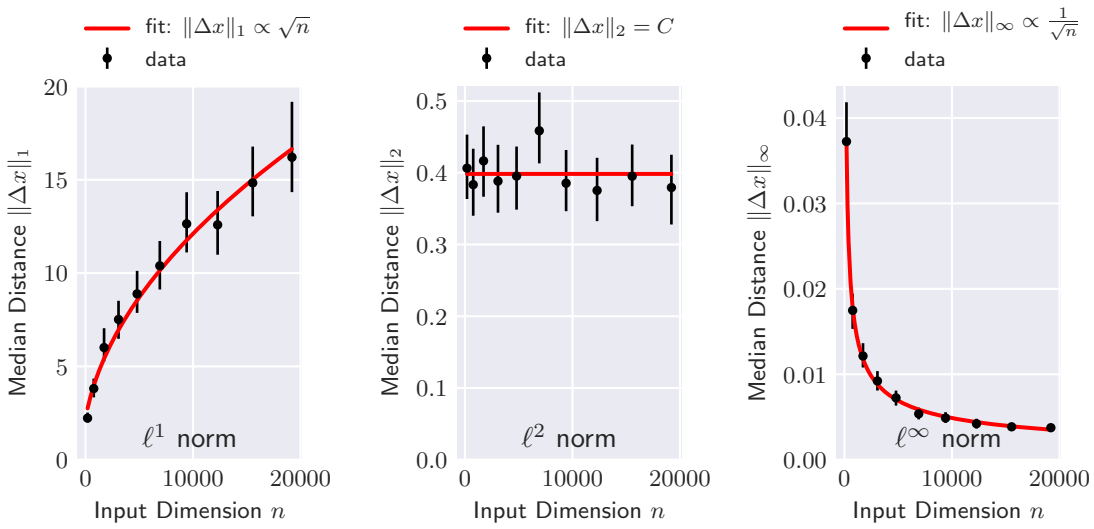

Figure 5. **Random untrained networks:** Median distance of closest adversarial examples $\|\Delta x\|_p$ from their respective inputs ($p \in \{1, 2, \infty\}$) scale as predicted in Remark 5 of the Main Manuscript for a simple convolutional network (see subsection 5.2 for full description of network). Error bars span ± 5 percentiles from the median. For each input dimension, results are calculated from 2000 samples (200 random networks each attacked at 10 random points). See section 5 for further details on how experiments were performed.

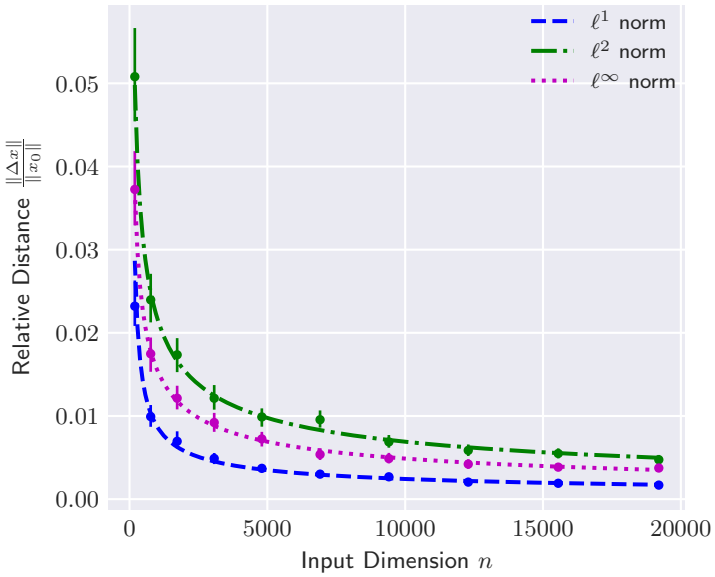

Figure 6. **Random untrained networks:** Median relative distance of closest adversarial examples $\|\Delta x\|_p / \|x_0\|_p$ from their respective inputs ($p \in \{1, 2, \infty\}$) scale with the input dimension n as $O(1/\sqrt{n})$ in all norms for a simple convolutional network (see subsection 5.2 for full description of network). Error bars span ± 5 percentiles from the median. For each input dimension, results are calculated from 2000 samples (200 random networks each attacked at 10 random points).

References

Adler, R. and Taylor, J. *Random Fields and Geometry*. Springer Monographs in Mathematics. Springer New York, 2009. ISBN 9780387481166.

Carlini, N. and Wagner, D. Towards Evaluating the Robustness of Neural Networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 39–57, May 2017. doi: 10.1109/SP.2017.49. ISSN: 2375-1207.

Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., and Li, J. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9185–9193, 2018.

Glorot, X. and Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256, 2010.

He, K., Zhang, X., Ren, S., and Sun, J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. *arXiv:1502.01852 [cs]*, February 2015. arXiv: 1502.01852.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv:1412.6980 [cs]*, 2014. arXiv: 1412.6980.

Kurakin, A., Goodfellow, I. J., and Bengio, S. Adversarial examples in the physical world. In *Artificial Intelligence Safety and Security*, pp. 99–112. Chapman and Hall/CRC, 2018.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. ISSN 0018-9219. doi: 10.1109/5.726791.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards Deep Learning Models Resistant to Adversarial Attacks. *arXiv:1706.06083 [cs, stat]*, September 2019. arXiv: 1706.06083.

Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z. B., and Swami, A. The Limitations of Deep Learning in Adversarial Settings. In *2016 IEEE European Symposium on Security and Privacy*, November 2015. arXiv: 1511.07528.

Price, E. Maurey's empirical method. *CS395T: Sublinear Algorithms, Lecture Notes*, 13, October 2016.

Rauber, J., Brendel, W., and Bethge, M. Foolbox: A Python toolbox to benchmark the robustness of machine learning models. *arXiv:1707.04131 [cs, stat]*, March 2018. arXiv: 1707.04131.

Rony, J., Hafemann, L. G., Oliveira, L. S., Ayed, I. B., Sabourin, R., and Granger, E. Decoupling direction and norm for efficient gradient-based l2 adversarial attacks and defenses. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4322–4330, 2019.

Schott, L., Rauber, J., Bethge, M., and Brendel, W. Towards the first adversarially robust neural network model on MNIST. *arXiv:1805.09190 [cs]*, September 2018. arXiv: 1805.09190.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

Tramer, F. and Boneh, D. Adversarial Training and Robustness for Multiple Perturbations. In Wallach, H., Larochelle, H., Beygelzimer, A., Alché-Buc, F. d., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, pp. 5866–5876. Curran Associates, Inc., 2019.

Vladimirova, M., Verbeek, J., Mesejo, P., and Arbel, J. Understanding priors in bayesian neural networks at the unit level. In *International Conference on Machine Learning*, pp. 6458–6467, 2019.

Yang, G. Wide feedforward or recurrent neural networks of any architecture are gaussian processes. In *Advances in Neural Information Processing Systems*, pp. 9947–9960, 2019.

Zhao, P., Liu, S., Wang, Y., and Lin, X. An ADMM-Based Universal Framework for Adversarial Attacks on Deep Neural Networks. *arXiv:1804.03193 [cs, stat]*, April 2018. arXiv: 1804.03193.