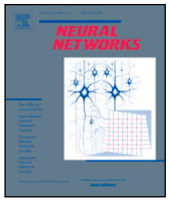
Neural Networks 180 (2024) 106711

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet



Full Length Article

How adversarial attacks can disrupt seemingly stable accurate classifiers

Oliver J. Sutton [a,*], Qinghua Zhou [a], Ivan Y. Tyukin [a], Alexander N. Gorban [b], Alexander Bastounis [a], Desmond J. Higham [c]

[a] Department of Mathematics, King's College London, London, UK
[b] School of Computing and Mathematical Sciences, University of Leicester, Leicester, UK
[c] School of Mathematics, University of Edinburgh, Edinburgh, UK

ARTICLE INFO

Keywords:
Neural networks
Adversarial attacks
Stability
Measure concentration theory

ABSTRACT

Adversarial attacks dramatically change the output of an otherwise accurate learning system using a seemingly inconsequential modification to a piece of input data. Paradoxically, empirical evidence indicates that even systems which are robust to large random perturbations of the input data remain susceptible to small, easily constructed, adversarial perturbations of their inputs. Here, we show that this may be seen as a fundamental feature of classifiers working with high dimensional input data. We introduce a simple generic and generalisable framework for which key behaviours observed in practical systems arise with high probability—notably the simultaneous susceptibility of the (otherwise accurate) model to easily constructed adversarial attacks, and robustness to random perturbations of the input data. We confirm that the same phenomena are directly observed in practical neural networks trained on standard image classification problems, where even large additive random noise fails to trigger the adversarial instability of the network. A surprising takeaway is that even small margins separating a classifier's decision surface from training and testing data can hide adversarial susceptibility from being detected using randomly sampled perturbations. Counter-intuitively, using additive noise during training or testing is therefore inefficient for eradicating or detecting adversarial examples, and more demanding adversarial training is required.

1. Introduction

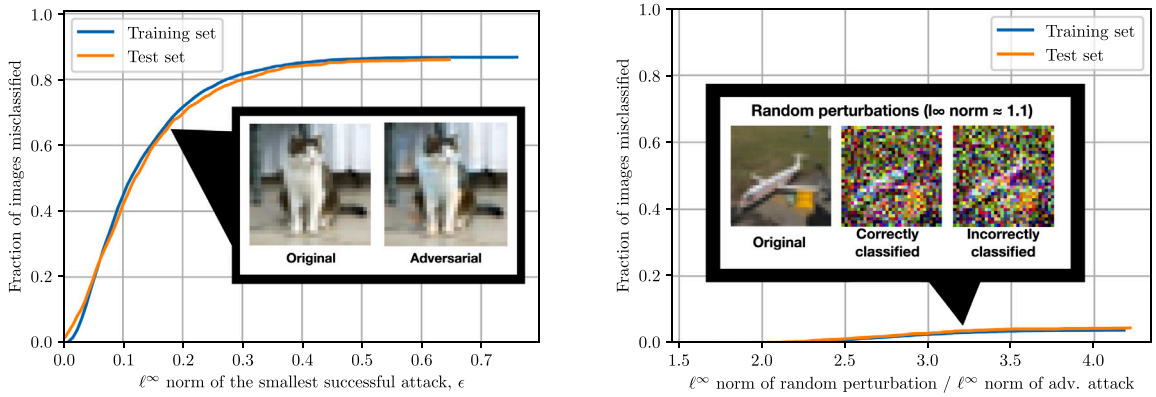
Adversarial attacks aim to slightly modify a piece of input data in such a way as to significantly change the output of a model. The sensitivity of neural networks to small perturbations like these has been widely studied since they were first reported in deep networks in Szegedy et al. (2014). Simple algorithms exist which enable a malicious attacker to produce adversarial perturbations quite easily in many cases (Chaubey, Agrawal, Barnwal, Guliani, & Mehta, 2020). Recent works (Bastounis et al., 2023; Bastounis, Hansen, & Vlačić, 2021) have shown that such instabilities are somewhat inevitable, even in relatively small networks consisting of just two layers where the number of neurons is linear in the input data dimension. It is remarkable, therefore, that these same instabilities are rarely triggered by random perturbations to the input data – even when these random perturbations may be much larger than destabilising adversarial perturbations.

This *paradox of apparent stability* is demonstrated in Fig. 1 for a standard convolutional neural network trained on CIFAR-10 images (Krizhevsky, 2009). Although the majority of images in both the training and test data sets are susceptible to small adversarial attacks (panel (*a*)), random perturbations even an order of magnitude larger mostly fail to cause the images to be misclassified (panel (*b*)). Further experimental results on other image classification datasets, including using pre-trained foundation models, are summarised in Table 1 and discussed further in Section 3.

Several explanations for the causes of adversarial examples have been proposed in the literature. An early work on the subject (Goodfellow, Shlens, & Szegedy, 2015) suggested that adversarial images simply live in regions of the data space to which the data distribution assigns low probability. A variant of this idea, discussed in Khoury and Hadfield-Menell (2018), suggests that adversarial attacks perturb inputs in a way that moves them in an orthogonal direction to the local data manifold. This results in adversarial images which exist in a region of data space where no training data could have been sampled, and the decision surfaces of the network are therefore relatively pathological. Other suggested mechanisms include the dimpled manifold hypothesis (Shamir, Melamed, & BenShmuel, 2022), boundary tilting (Tanay & Griffin, 2016), and the existence of uncountably large families of special distributions for which instabilities are expected (Bastounis et al., 2021). However, none of these frameworks rigorously account for

* Corresponding author.
E-mail address: oliver.sutton@kcl.ac.uk (O.J. Sutton).

https://doi.org/10.1016/j.neunet.2024.106711
Received 6 February 2024; Received in revised form 27 June 2024; Accepted 5 September 2024
Available online 12 September 2024



(a) Cumulative histogram of sizes of successful adversarial attacks.

(b) Cumulative histogram of sizes of successful random perturbations.

Fig. 1. Histograms showing the fraction of images from the 'aeroplane-vs-cat' binary classification problem (from the CIFAR-10 dataset) which were misclassified after either (a) an adversarial attack (as the fraction of ordinarily correctly classified images) or (b) a random perturbation of different sizes (as the fraction of images which were susceptible to adversarial attacks), measured as the maximum absolute change to an individual pixel channel (the ℓ^∞ norm). For adversarial attacks, this represents the smallest misclassifying attack in the adversarial direction. For the random perturbations, we record the smallest ℓ^∞ norm among 2000 misclassifying perturbations sampled from the Euclidean ball with radius 5ϵ, where ϵ is the Euclidean norm of the smallest successful adversarial attack found for each image. Examples are shown at the size of their respective perturbation norms. Full details of the experimental results are given in Section 7.

Table 1
A summary of the performance of networks trained on different standard image classification benchmark datasets. We split each dataset into a set of binary classification problems, and the results in this table are reported in the form 'train, test' and as the median over the binary classification problems of the form 'class i-vs-class $i+1$' within each dataset. The symbol '-' denotes values which were not computed. Full details of the experimental setup and results are given in Sections 7.1 and 7.2. Adversarial attack susceptibility is measured as the proportion of images in both classes of each problem which were susceptible to an adversarial attack. Random attack susceptibility with fixed δ is measured as the proportion of adversarially susceptible images which were misclassified after applying any of 2000 randomly sampled perturbations with Euclidean norm up to δ times as large as that of the smallest adversarial perturbation identified on that image. The results on GTSRB were computed using a subset of classes in the original dataset, see Section 7.2.3 for details. Results for ImageNet were evaluated using pre-trained ResNet50 and VGG19 models from Tensorflow on images from the ImageNet validation set, as described in Section 7.2.4.

	CIFAR-10	Fashion MNIST	GTSRB	ImageNet (ResNet50)	ImageNet (VGG19)
Accuracy	99.70%, 95.80%	99.51%, 99.4%	98.32%, 98.51%	–, 70.8%	–, 66.52%
Adversarial attack susceptibility	91.88%, 89.96%	53.58%, 53.01%	77.53%, 77.00%	–, 94.2%	–, 97.07%
Random attack susceptibility ($\delta = 2$)	0.02%, 0.17%	0.07%, 0.09%	0.36%, 0.36%	–	–
Random attack susceptibility ($\delta = 5$)	2.65%, 2.57%	10.71%, 13.35%	5.76%, 5.1%	–	–
Random attack susceptibility ($\delta = 10$)	41.19%, 40.57%	56.84%, 57.43%	39.26%, 36.07%	–, 2.5%	–, 1.4%
Input dimension	$32 \times 32 \times 3$	$28 \times 28 \times 1$	$30 \times 30 \times 3$	$224 \times 224 \times 3$	$224 \times 224 \times 3$
Number of classes	10	10	6	1000	1000

and explain the paradoxical simultaneous robustness of these classifiers to random perturbations whose size could be several times larger than that of the adversarial ones.

Here, we suggest a resolution to the paradox rooted in ideas from the theory of concentration of measure, and related properties of high dimensional probability distributions. The simple, realistic framework we introduce captures the key features of the paradox which are observed in practice (precise definitions of these terms are given in Section 4):

Accuracy: The classifier correctly labels non-perturbed data.

Apparent robustness/stability: There is a vanishingly small probability that a sampled data point will be misclassified after a large random perturbation is applied to it.

Vulnerability: Yet, with high probability, any sampled data point is susceptible to a very small adversarial perturbation that changes the predicted class.

Computability: An optimal destabilising perturbation can be computed from knowledge of the loss function gradient.

Our theoretical investigation reveals a tension between different notions of what it means for a classifier to be stable, a subtlety which is rarely discussed in practice. A problem may be *deterministically unstable* in the sense that for a given data point there exists an arbitrarily small destabilising perturbation which may be exploited by an attacker, while the fact that this instability is extremely unlikely to be triggered by random noise renders the problem *probabilistically stable*. This is a dangerous situation for a performance-critical classifier: even though the performance appears excellent at test time, adversarial instabilities and the lack of deterministic robustness lurk awaiting an unscrupulous attacker and cannot be efficiently detected at random. An important feature of our theoretical framework (developed in Section 4) for understanding the paradox of apparent stability is that it can be studied at various levels of generality. This enables us to distil the fundamental origins of the paradox without unnecessary technical details, through a hierarchy of models incorporating different levels of complexity.

Our findings are directly supported by extensive experimental results, summarised in Section 3 and discussed in detail in Section 7. The results demonstrate the paradox of apparent stability, and reveal some of its real-world implications. We show the effectiveness of simple adversarial attacks on a variety of different standard datasets and models, and contrast this against their robustness to large random perturbations. These experiments confirm the predictions of our theoretical results: to observe cases in which random perturbations cause labels to swap, many perturbations must be taken, and with a significantly larger amplitude than that of the smallest adversarial perturbation affecting the same image. An immediate practical consequence of this is to shed new light on algorithms which aim to ensure or certify adversarial robustness by adding random noise to inputs, such as those discussed in Cohen, Rosenfeld, and Kolter (2019), Li, Chen, Wang, and Carin (2019), Ye et al. (2024). Our investigation reveals that this is computationally inefficient in high dimension, since it requires an exponentially

large number of perturbed samples per data point to expect to observe just one which causes a misclassification of even a highly susceptible input. We also find that in genuinely high-dimensional settings adding random noise at training time significantly degrades the trained model's accuracy, which appears to outweigh any marginal improvements in adversarial robustness. The relevant spectrum of tasks to which the above applies includes popular image classification problems. This implies that data pre-processing involving an appropriate dimensionality reduction may be needed to bring out the benefits of robustness induced by random data augmentation at training.

To study the paradox of apparent stability, we begin with a simple model which nonetheless exhibits key features of the paradox, and build it up to expose how different phenomena appear and persist as the model becomes more general. We first consider a single fixed data point (i.e. without any data sampling process) sitting close to a (locally) linear decision surface in Section 4.1. We prove that in this situation, the probability of randomly sampling a perturbation which causes the data point to be misclassified is exponentially small in the data dimension. This already shows that dimensionality is a fundamental component in trying to understand the relationship between random and adversarial perturbations, and therefore in resolving the paradox of apparent stability. This result also shows that algorithms which aim to detect or defend against adversarial susceptibility using random data perturbations, such as those discussed in Cohen et al. (2019), Li et al. (2019), Ye et al. (2024), may in fact require computational complexities which are exponentially large with respect to the data dimension.

We build on this in Section 4.2 by considering a binary classification problem with a linear classifier. Data from both classes are sampled from distributions satisfying a mild non-degeneracy condition, formulated as a simplified version of the Smeared Absolute Continuity (SmAC) condition introduced in Gorban, Golubkov, Grechuk, Mirkes, and Tyukin (2018) (see Definition 3). Despite its simplicity, we prove that this setting already exhibits the four characteristics above of the paradox of apparent stability. We then significantly generalise the setup in Section 4.3 to show that these phenomena persist when data are sampled from two arbitrary distributions and classified using nonlinear decision surfaces. Once again, this setup reveals the same fundamental characteristics of the paradox of apparent stability. This setup admits various further generalisations, which are discussed in Section 4.4.

As a counterpoint to the findings discussed above, Section 5 reveals a subtle, yet important, modification which can be made to the setup which causes the discrepancy between adversarial and random perturbations to disappear. Specifically, we construct a scenario in which the typical distance from a sampled data point to the classifier's (linear) decision boundary approaches zero in high dimensions. In this case, the probability of a random perturbation to input data causing a misclassification is separated away from zero by a constant for arbitrarily large data dimension, rather than exponentially decreasing as in the previous scenarios. Having small distances from typical data points to the decision surface is clearly undesirable in any practical application, since it means that the classifier itself is extremely sensitive to small changes in the data. However, this setup reveals that it is in fact the presence of an appropriate margin between typical data points and the decision surface which manifests itself as the paradox of simultaneous apparent robustness to large random perturbations and vulnerability to small adversarial attacks.

The paper is organised as follows. In Section 2, we introduce mathematical notation that is used throughout the text. Section 3 gives an overview of the paradox of apparent stability as it manifests itself in real image classification problems. Our main theoretical results are presented in Section 4, introducing a hierarchy of simple models which expose the fundamental origins of the paradox. The alternative model analysed in Section 5 reveals the link between the existence of non-zero classification margins and the ability to determine susceptibility to adversarial examples using random perturbations. In Section 6 we discuss these analytical and empirical findings and relate them to prior work and knowledge in the area. Section 7 provides a comprehensive description of our experimental setup and numerical results. Section 8 concludes the paper. Proofs of all statements and auxiliary technical results are provided in the Appendix.

2. Notation

We use the following notation throughout:

- $x \cdot y$ denotes the inner product of $x, y \in \mathbb{R}^n$ and $\|x\| = \sqrt{x \cdot y}$ denotes the Euclidean (ℓ^2) norm,
- the ℓ^1 norm of a vector in $\mathbb{R}^n$ is defined to be the sum of the absolute values of its components,
- the ℓ^∞ norm of a vector in $\mathbb{R}^n$ is defined to be the maximum of the absolute values of its components,
- $\mathbb{B}^n_r(c) = \{x \in \mathbb{R}^n : \|x - c\| \leq r\}$ denotes the Euclidean ball in $\mathbb{R}^n$ with radius $r > 0$ centered at $c \in \mathbb{R}^n$, and we use the abbreviation $\mathbb{B}^n = \mathbb{B}^n_1(0)$,
- $V^n = \frac{\pi^{\frac{n}{2}}}{\Gamma(\frac{n}{2}+1)}$ denotes the n-dimensional volume of $\mathbb{B}^n$ (the unit ball, usually assumed centred at 0), and $V^n_{\text{cap}}(r,h)$ denotes the volume of the cap with height h of the n-dimensional ball of radius r, i.e. the volume of the set $\{x \in \mathbb{R}^n : \|x\| < r \text{ and } x_1 > 1-h\}$, where $x_1 = x \cdot \mathbf{e}_1$ and $\mathbf{e}_1 = (1, 0, \ldots, 0)^\top \in \mathbb{R}^n$; if $S \subset \mathbb{R}^n$ then $V^n(S)$ denotes the n-dimensional volume of the set S,
- for a set $S \subset \mathbb{R}^n$, we use $\mathcal{U}(S)$ to denote the uniform distribution on S, and $\mathbb{I}_S : S \to \{0,1\}$ to denote the indicator function of S, such that $\mathbb{I}_S(x) = 1$ for $x \in S$ and 0 otherwise,
- the function $\Phi : \mathbb{R} \to \mathbb{R}$ denotes the standard Gaussian cumulative distribution function

$$\Phi(s) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{s} e^{-\frac{\xi^2}{2}} d\xi.$$

3. The paradox of apparent stability demonstrated on standard datasets

The phenomenon of simultaneous susceptibility to adversarial attacks and robustness to random noise can be clearly demonstrated using standard image classification benchmark datasets. Here, we summarise results presented in detail in Section 7, calculated using CIFAR-10 (Krizhevsky, 2009), Fashion MNIST (Xiao, Rasul, & Vollgraf, 2017), the German Traffic Sign Recognition Benchmark (GTSRB) (Stallkamp, Schlipsing, Salmen, & Igel, 2012), and ImageNet (Russakovsky et al., 2015). Our experimental setup is described in detail in Section 7.1.

To present the phenomenon in the simplest possible setting, we split each of CIFAR-10, Fashion-MNIST and GTSRB into a set of binary classification problems, one for each pair of classes in the dataset. A separate network (each with the same convolutional structure in the form of a truncated VGG network (Simonyan & Zisserman, 2015)) was trained using Tensorflow (Abadi et al., 2015) for each of these problems, and each point in the training and test set was assessed for its susceptibility to adversarial examples using a gradient-based attack algorithm. On images which were susceptible to an adversarial attack with Euclidean norm ϵ, we applied 2000 perturbations randomly sampled from the Euclidean ball with radius $\delta\epsilon$ for each value of δ in the set $\{1, 2, 5, 10\}$. This measures the sensitivity of the network to random perturbations around the training and test images.

We complement this investigation of binary classification problems with an analogous analysis of the adversarial and random susceptibility of pre-trained models (with VGG19 (Simonyan & Zisserman, 2015) and ResNet50 (He, Zhang, Ren, & Sun, 2016) architectures, from Tensorflow) using images from the ImageNet validation set. These models are trained to classify ImageNet images into 1,000 classes, and demonstrate the presence of the paradox of apparent stability in real-world models.

The empirical essence of the phenomenon is illustrated in Fig. 1 using the CIFAR-10 dataset: while the networks were easily fooled

by relatively small adversarial perturbations which appear to make little perceptual difference to the image, they were remarkably robust to randomly sampled perturbations. Here we demonstrate this in the broadly representative case of the 'aeroplane-vs-cat' binary classification problem. Comparing the inset examples in Figs. 1(a) and 1(b), the modification made by the adversarial perturbation does not alter the overall perception of the image as that of a 'cat'. Moreover, it is difficult to tell by just looking at these images which one of them has been subjected to an adversarial attack. It is nearly equally difficult to make out the aeroplane in the (correctly classified) randomly perturbed image. Note that, since the original images have pixel channel values in $[0, 1]$, a perturbation with ℓ^∞ norm greater than 1 represents a drastic change to the contents of the image, yet one which was rarely able to cause the network to misclassify its input.

A summary of the accuracy and susceptibility of the classifiers is presented in Table 1, although the figures alone make it clear that even when the random perturbations are sampled to be five times as large as the known adversarial perturbation ($\delta = 5$), they still mostly fail to trigger the adversarial susceptibility of the network. The effects are broadly consistent across all the datasets we examined, with negligible difference between the training and test data. Further details of the experimental setup and full results for this and the remaining classification problems are explored in Section 7.2.

We also provide the results of experiments on CIFAR-10 into incorporating additive random noise to data at training time to assess the impact this may have on adversarial susceptibility (the experimental setup is described in Section 7.1.4 and the results are presented in Section 7.2.1). The conclusion we draw from these experiments is that training with even large random perturbations does not significantly decrease the susceptibility to adversarial attacks, and is responsible for a large drop in accuracy.

4. The essence of the paradox

To understand the origins of the paradox of apparent stability, we show how a hierarchy of simple, yet reasonably generic, theoretical models can explain the behaviour observed empirically. First, in Section 4.1 we show that for a fixed data point close to a model's (locally linear) decision boundary, randomly sampled noise is very unlikely to detect adversarial instabilities in high dimensions. Since this example does not assume any sampling distribution for the data point, it provides a generic setup for understanding the difference between random and adversarial perturbations.

We generalise this to a second scenario in Section 4.2, where data from two classes are sampled from a reasonably general class of distributions, and classified using a linear classifier. The data distributions are only assumed to satisfy a mild non-degeneracy condition (known as the SmAC condition; Definition 3). Despite its apparent simplicity, this setup already simultaneously presents all of the symptoms of the paradox of apparent stability: with high probability, data points are accurately classified (Theorem 4) and susceptible to small adversarial perturbations (Theorem 5), yet with high probability randomly sampled perturbations do not cause data to be misclassified (Theorem 6). Moreover, gradient-based algorithms are efficient for constructing the adversarial attack (Theorem 7), and successful attacks are even universal in the sense that they also cause other data points with the same class to be misclassified with high probability (Theorem 8).

This setup is generalised further in Section 4.3, providing versions of the same key results. No assumptions are placed on the data distributions, and the results require only that the classifier's decision boundary is a Lipschitz warping of a plane in its normal direction. We also show how the results from Section 4.2 may be obtained as corollaries of these general results.

Further generalisations of our results are considered in Section 4.4.

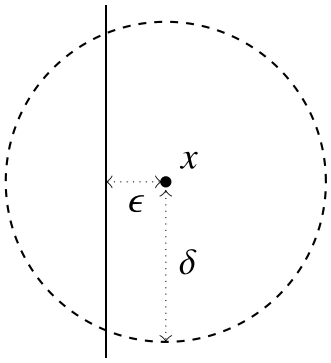


Fig. 2. A data point x and the (locally linear) decision surface of a classifier f (solid line). The point x is susceptible to an adversarial attack of size ϵ, and randomly perturbed using random noise of size $\leq \delta$. These perturbed points are sampled from the within dashed ball.

4.1. Random perturbations are inefficient for detecting adversarial instability

We first consider the simple setup illustrated in Fig. 2. Suppose we wish to estimate the susceptibility of a fixed data point x to adversarial attack. This may be measured as the distance from x to the decision surface of a classifier. For simplicity, we suppose that this decision surface is locally linear near x, and we denote the shortest distance from x to the decision surface by ϵ. We may therefore say that the point x is susceptible to an adversarial attack of size ϵ, since this is the smallest perturbation which would push x across the decision boundary. In keeping with the setup of the paradox, we attempt to estimate the (unknown) size of ϵ by randomly perturbing the data point x using noise of size δ. We may do this by measuring the proportion of random perturbations which fall on the opposite side of the decision surface. Unfortunately, as described in Theorem 1, this process is extremely inefficient in high dimensions.

Theorem 1 (*Random Perturbations are Inefficient for Detecting Adversarial Instability*)**.** *Let $x \in \mathbb{R}^n$ and let Π be a planar decision surface passing distance $\epsilon > 0$ from x. Suppose (without loss of generality since the setup is invariant to rigid translations) that Π passes through the origin. Suppose that points z are sampled uniformly from a ball of radius $\delta \geq \epsilon$ around x. Then, the probability of sampling a point z with a different classification from x decreases exponentially with the dimension n. Specifically, if Π has normal vector ν (with $\|\nu\| = 1$) then*

$$P(z \sim \mathcal{U}(\mathbb{B}^n_\delta(x)) \ : \ \operatorname{sign}(z \cdot \nu) \neq \operatorname{sign}(x \cdot \nu)) \leq \frac{1}{2}\left(1 - \frac{\epsilon^2}{\delta^2}\right)^{\frac{n}{2}}.$$

This theorem is proved in Appendix A. The clear implication is that exponentially many perturbed data samples would be required to expect to find any which are misclassified. This remains true even when the sampled noise is much larger than the size ϵ of the adversarial attack affecting x. Since this does not depend on any data distribution of x, it provides a first hint at the foundations of the paradox.

It is interesting to ask whether this finding is due to the choice of sampling noise uniformly from a ball around x. The answer to this question is 'no', due to concentration properties of data distributions in high dimensions. For example, points sampled from a uniform distribution on the cube $[-1, 1]^d$ and those sampled from a Gaussian distribution (with mean 0 and unit variance) both concentrate such that $\frac{1}{\sqrt{d}}\|x\|$ is almost constant with high probability in high dimensions (Ledoux, 2001). Data from these distributions therefore behaves very similarly to data sampled uniformly from a ball (albeit a ball with radius growing with $\sqrt{d}$).

4.2. A simple theoretical model captures the essence of the paradox

To more completely understand the paradox, in this section we show how it manifests itself in the case where data points sampled from two classes are classified using a linear classifier. We adopt the simple yet reasonably flexible assumption that each data class is sampled from a distribution supported somewhere within a ball and satisfying a mild non-degeneracy requirement (Definition 3). A significantly more general version of this model is analysed in Section 4.3, with fewer constraints on the distributions and a classifier which is permitted to use a more general nonlinear decision surface. The results and conclusions remain largely qualitatively similar. In particular, the simultaneous co-existence of high accuracy, the typicality of data susceptible to adversarial attacks, and the rarity of destabilising random perturbations with bounded Euclidean norm all extend to the more general model with nonlinear decision boundary (see Theorems 9, 11, 14 and Corollaries 10, 12, 15).

Let us now formally define the setup considered in this section. To assess the probabilities and typicality of events we need to define an appropriate class of data distributions. This class should be sufficiently flexible to capture uncertainties and the lack of precise knowledge about data distributions while also tractable enough to enable a mathematical assessment of the setup. One such class of distributions is those satisfying the Smeared Absolute Continuity property (Gorban et al., 2018).

Definition 2 (*Smeared Absolute Continuity (SmAC) (Gorban et al., 2018)*)**.** Let $x_1, \dots, x_M \in \mathbb{R}^n$ be random variables. The joint distribution of $x_1, \dots, x_M$ has the SmAC property if there exist constants $\alpha > 0$, $\beta \in (0,1)$, and $\gamma > 0$ such that for every positive integer n, any convex set $S \subset \mathbb{R}^n$ such that

$$\frac{V^n(S)}{V^n(\mathbb{B}^n)} \leq \alpha^n,$$

any index $i \in \{1, 2, \dots, M\}$ and any points $y_1, \dots, y_{i-1}, y_{i+1}, \dots, y_M \in \mathbb{R}^n$, we have

$$P(x_i \in \mathbb{B}^n \setminus S \mid x_j = y_j,\ \forall\ j \neq i) \geq 1 - \gamma\beta^n.$$

In this work, however, we do not wish to consider a joint distribution over multiple random variables as our main case focuses on a single point drawn from a distribution. Furthermore, to make the technical analysis simpler and clearer it is beneficial to assume the existence of a probability density function that is associated with the probability measure. Therefore, in what follows we adopt the following restricted single-particle version of the SmAC property which, for the sake of brevity, will be referred to as SmAC in the rest of the paper:

Definition 3 (*Single-Particle SmAC with Bounded Density*)**.** A distribution D on $\mathbb{R}^n$ is said to satisfy the *single-particle smeared absolute continuity condition with bounded density* if it possesses a density $p : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ and there exists a centre point $c \in \mathbb{R}^n$ and radius $r > 0$ such that $p(x) > 0$ only for points x in the ball $\mathbb{B}^n_r(c)$, and there exists a constant growth parameter $A > 0$ such that

$$\sup_{x \in \mathbb{B}^n_r(c)} p(x) \leq \frac{A}{V^n r^n}.$$

We note that if the growth property is satisfied with $A = 1$, then the distribution is simply the uniform distribution on the ball $\mathbb{B}^n_r(c)$.

Suppose that two classes of data are each sampled from data distributions D_0 and D_1 on $\mathbb{R}^n$, each satisfying Definition 3. For simplicity, we suppose that these distributions are each supported in a ball with radius 1, with centres given by $c_0 = -\epsilon\mathbf{e}_1$ for class 0 and $c_1 = \epsilon\mathbf{e}_1$ for class 1. We further suppose that both distributions satisfy the growth bound with the same parameter A. For brevity, we also define the combined distribution D_ϵ which samples a point from D_0 with label 0 with probability $\frac{1}{2}$, and samples a point from D_1 with label 1 with probability $\frac{1}{2}$. The geometry of this setup is illustrated in Fig. 3.

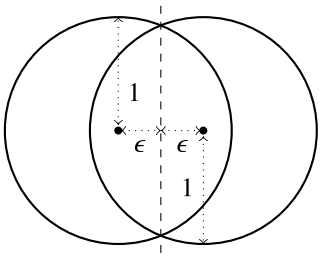


Fig. 3. Two unit balls with centres separated by distance 2ϵ, and the decision surface of the classifier f (dashed).

The classification function $f : \mathbb{R}^n \to \{0, 1\}$ with the highest accuracy which can be defined for this data model without further knowledge of the distributions is given by the simple linear separator

$$f(x) = \begin{cases} 0 & \text{if } x_1 < 0, \\ 1 & \text{otherwise.} \end{cases} \tag{1}$$

This classifier does not necessarily return the correct label in all cases since, for $\epsilon \in (0, 1)$, the two data classes overlap. Despite this, misclassified points are rare in the high dimensional setting, even when the two balls from which points are sampled have only a small separation between their centres. More precisely, the probability that this classifier is correct converges exponentially to 1 as the data dimension grows. This result is proven in Appendix B.1.

Theorem 4 (*The Classifier is Accurate*)**.** *For any $\epsilon > 0$, the probability that the classifier applies the correct label to a randomly sampled data point grows exponentially to 1 with dimension n, specifically*

$$P((x, \ell) \sim D_\epsilon : f(x) = \ell) \geq 1 - \frac{1}{2} A (1 - \epsilon^2)^{\frac{n}{2}}.$$

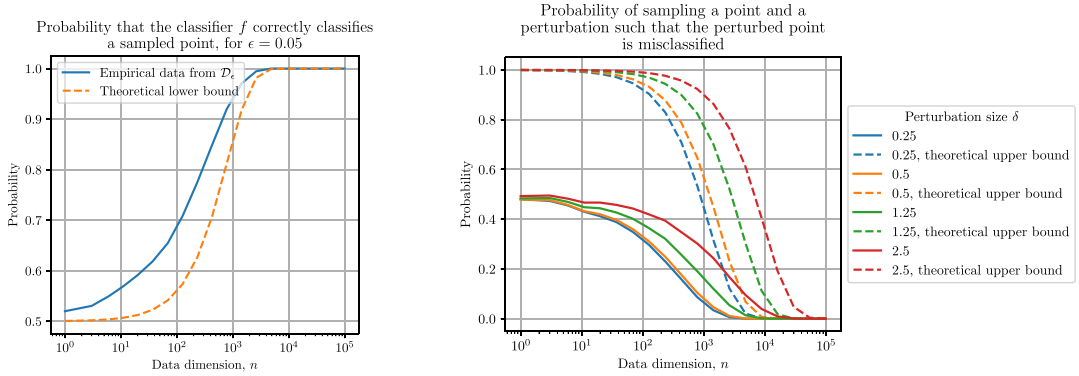
The sharpness of this result is verified empirically in Fig. 4(a), computed for $A = 1$. We observe that by $n = 10{,}000$, the probability of sampling a point which will be misclassified is virtually 0. To put this and the following results into context, the $32 \times 32 \times 3$ images used in CIFAR-10 have 3,072 attributes, while the size of $256 \times 256 \times 3$ commonly used for the images in ImageNet have 196,608 attributes, placing them firmly within the range of dimensionalities where the effects described here are active.

On the other hand, even accurately classified points in this model are still close to the decision surface since the ball centres are only separated by distance ϵ. Because of this, for any $\delta > \epsilon$, there are points sampled from each class which are susceptible to an adversarial attack $s \in \mathbb{R}^n$ with $\|s\| \leq \delta$ which causes f to predict the wrong class. Moreover, in high dimensions, data points sampled from such a distribution concentrate at distance ϵ from this decision surface, meaning that the probability of sampling a point which is susceptible to an adversarial attack is high. This may be encapsulated in the following result, which is proven in Appendix B.2.

Theorem 5 (*Susceptible Data Points are Typical*)**.** *For any $\epsilon \geq 0$ and $\delta \in [\epsilon, 1 + \epsilon]$, the probability that a randomly sampled data point is susceptible to an adversarial attack with Euclidean norm δ grows exponentially to 1 with the dimension n, specifically*

$$P\big((x, \ell) \sim D_\epsilon \ :\ \textit{there exists } s \in \mathbb{B}^n_\delta \textit{ such that } f(x + s) \neq \ell\big) \geq 1 - \frac{1}{2} A (1 - (\delta - \epsilon)^2)^{\frac{n}{2}}.$$

Although this susceptibility may therefore be viewed as typical in high dimensions, however, the probability of detecting it by sampling random perturbations of data points is paradoxically very small, as shown by the following result which is proven in Appendix B.3.



(a) Accuracy of the classifier (Theorem 4)

(b) Destabilisation probability (Theorem 6)

Fig. 4. Comparison of the theoretical bounds in Theorems 4 and 6 against empirical results computed using 10,000 data points sampled from $\mathcal{D}_\epsilon$, with $\epsilon = 0.05$, and 10,000 perturbations sampled from $\mathcal{U}(\mathbb{B}_\delta^n)$ for various values of δ. We see that, even for perturbations 50 times larger than the separation distance between the balls (i.e. $\delta = 2.5$), the probability of randomly sampling a perturbation which changes the classification of a random data point is very small in high dimensions.

Theorem 6 (*Destabilising Perturbations are Rare*)**.** *For any $\delta > \epsilon \geq 0$, the probability that a randomly selected perturbation with Euclidean norm δ causes a randomly sampled data point to be misclassified is bounded from above as:*

$$P\big((x,\ell) \sim \mathcal{D}_\epsilon, s \sim \mathcal{U}(\mathbb{B}_\delta^n) \ : \ f(x+s) \neq \ell\big) \leq A\left(1 - \left(\frac{\epsilon}{1+\delta}\right)^2\right)^{\frac{n}{2}}.$$

In particular, when δ is independent of dimension n, this probability converges to 0 exponentially with n.

This probability bound is compared against empirically sampled data in Fig. 4(b). While the bound is not particularly sharp in low dimensions, it accurately describes the key phenomenon which is the convergence of the probability to 0 in high dimensions. This phenomenon is startlingly persistent, even when the magnitude of the sampled perturbations is 50 times greater than the distance between the centres of the spheres (when $\delta = 2.5$).

We note that some care needs to be taken when considering perturbations with fixed ℓ^∞ norms. The corresponding ℓ^2 norm of these perturbations scales as $\sqrt{n}$, affecting convergence to 0 of the probability of destabilisation (see Theorem 6).

Even though randomly sampled perturbations are unlikely to affect the classifier, it is often straightforward to construct special adversarial perturbations which will affect a specific data point. Common algorithms for constructing adversarial attacks work by perturbing the target input in such a way as to increase an appropriate loss function. Gradient-based methods for this, such as the Fast Gradient Sign Method (Goodfellow et al., 2015), compute the gradient of the loss function with respect to the components of the input, evaluated at the target input with its true class. Perturbing the input in the direction of this gradient therefore moves it in the direction of steepest ascent of the loss function locally, thereby representing a good candidate for an adversarial direction. The minimal scaling to be applied to this adversarial direction, required to form the final adversarial input, can then be determined via a line search in the adversarial direction.

In the case of this model setup, such an algorithm (with a standard choice of loss function) will successfully provide the optimal direction for an adversarial attack: the most direct path to move the input along in order to cross the decision surface. To show this, we first observe that the classifier f in (1) can be equivalently defined as $f(x) = H(g(x))$, where $H : \mathbb{R} \to \{0,1\}$ denotes the (piecewise constant) Heaviside function, and the linear function $g(x) = \mathbf{e}_1 \cdot x - \frac{1}{2}$. To construct gradient-based attacks, we use a differentiable version $\tilde{f}$ of f constructed as $\tilde{f}(x) = \sigma(g(x))$, where $\sigma : \mathbb{R} \to (0,1)$ is a continuously differentiable version of the Heaviside function which is monotonically increasing with $\sigma(0) = \frac{1}{2}$. An example of such a function is the standard sigmoid function. Then, the following result, proved in Appendix B.4, shows that gradient-based attacks on this classifier will always return the optimal attack direction.

Theorem 7 (*Gradient-Based Methods Find the Optimal Adversarial Attack*)**.** *Let $L : \mathbb{R}_{>0} \to \mathbb{R}$ denote any differentiable, monotonically increasing loss function, and let $(x,\ell) \sim \mathcal{D}_\epsilon$. Then, with probability 1 with respect to the sample (x,ℓ), the gradient of the loss $L(|\tilde{f}(x) - \ell|)$ with respect to the components of x corresponds to a positive multiple of the optimal attack direction $(1-2\ell)\mathbf{e}_1$.*

A further aspect of this model problem is that successful adversarial attacks are universal in high dimensions. To state this property mathematically, we define the *destabilisation margin* to be the distance by which a destabilising perturbation pushes a data point across the decision threshold of the classifier (1). This is measured by the functions $d_\ell : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ associated with each class $\ell = 0, 1$, where, for a data point x and a perturbation s,

$$d_0(x,s) = \max\{x_1 + s_1, 0\},$$

and

$$d_1(x,s) = \max\{-x_1 - s_1, 0\}.$$

The following result then holds, as proven in Appendix B.5.

Theorem 8 (*Universality of Adversarial Attacks*)**.** *Let $\epsilon \geq 0$ and suppose that $x, z \sim \mathcal{D}_\epsilon$ are independently sampled points with the same class label ℓ. For any $\gamma \in (0,1]$, the probability that x is destabilised by all perturbations $s \in \mathbb{R}^n$ which destabilise z with destabilisation margin $d_\ell(z,s) > \gamma$ converges exponentially to 1 as the dimension n increases. Specifically, for $\ell \in \{0,1\}$ and $z \in \mathbb{R}^n$, let $S_z = \{s \in \mathbb{R}^n \ : \ d_\ell(z,s) > \gamma\}$. Then,*

$$P(x, z \sim \mathcal{D}_\ell \ : \ f(x+s) \neq \ell \text{ for all } s \in S_z) \geq \left(1 - A\left(1 - \frac{\gamma^2}{4}\right)^{\frac{n}{2}}\right)^2$$

This bound shows that in high dimensions we may expect pairs of sampled points to share their sets of adversarial perturbations. The dependence on the margin γ by which the perturbation destabilises z is an interesting feature. Roughly speaking, the result suggests that in low dimensions only severe perturbations which push points a long way past the decision threshold may be regarded as universal in the sense of having a high probability of destabilising other sampled points. As the dimension n increases, however, perturbations which produce smaller and smaller margins on individual points become universal in the sense that they have a constant probability of destabilising other sampled points.

4.3. A generalised theoretical model

We now show that the simple case presented in Section 4.2 extends to more general cases in which the classification surface is not assumed to be flat, and the data are sampled from more general distributions. To demonstrate that these abstract results are true generalisations of the results proven in Section 4.2, we derive corollaries to each result for a general SmAC distribution with a flat decision surface. These corollaries are therefore directly comparable with the results in Section 4.2 for specific indicated values of the parameters.

Let $v, w \in \mathbb{R}^n$ with $\|v\| = 1$, and define the plane

$$\pi = \{x \in \mathbb{R}^n \ : \ (x - w) \cdot v = 0\} \subset \mathbb{R}^n,$$

which passes through w and is normal to the vector v. Denote by $\Pi : \mathbb{R}^n \to \pi$ the orthogonal projection operator onto π in the Euclidean inner product, given by

$$\Pi x = x - ((x - w) \cdot v)v.$$

Let $\phi : \pi \to \mathbb{R}$ be continuous, and define the surface

$$S = \{x \in \mathbb{R}^n \ : \ x - \phi(\Pi x)v \in \pi\} \subset \mathbb{R}^n.$$

A projector $\Gamma : \mathbb{R}^n \to S$ onto the surface S (along the vector v) can be defined by

$$\Gamma x = \Pi x + \phi(\Pi x)v.$$

We also introduce the signed directed distance function $d_\pi : \mathbb{R}^n \to \mathbb{R}$ measuring the signed distance from a point x to the plane π along the normal vector v, given by

$$d_\pi(x) = (x - \Pi x) \cdot v = (x - w) \cdot v,$$

and $d_S : \mathbb{R}^n \to \mathbb{R}$ measuring the signed distance from a point x to the surface S along the vector v, given by

$$\begin{aligned} d_S(x) &= (x - \Gamma(x)) \cdot v = (x - \Pi x) \cdot v - \phi(\Pi x) \\ &= d_\pi(x) - \phi(\Pi x). \end{aligned}$$

Finally, we can define the distance from a point x to the surface S by

$$\sigma(x) = \inf_{\hat{y} \in S} \|x - \hat{y}\|,$$

noting the trivial inequality

$$\sigma(x) \leq |d_S(x)|, \tag{2}$$

for any $x \in \mathbb{R}^n$, since d_S only measures distance to S in the direction of v while σ measures the shortest distance to S in any direction.

With these constructions, we can define a binary classifier with decision surface S as the function $f : \mathbb{R}^n \to \{0, 1\}$ given by

$$f(x) = \begin{cases} 0 & \text{if } d_S(x) \leq 0, \\ 1 & \text{otherwise.} \end{cases} \tag{3}$$

To show how our previous results extend into this more general case, suppose that data points of class 0 are sampled from a distribution $\mathcal{D}$ on $\mathbb{R}^n$, and that data points of class 1 are sampled from the distribution $\mathcal{D}'$ on $\mathbb{R}^n$. In the interests of simplicity, we only study the behaviour of the classifier for data from the class 0, as the result for the class 1 is analogous. We study this more general model in parallel with the results of Section 4.2.

We first observe that the accuracy of the classifier may be controlled in an analogous way to the simple case in Section 4.2. The supremum in this result (and the suprema and infima in subsequent results) is simply present to ensure an optimal balancing for the two terms; a valid (though possibly sub-optimal) result may be obtained by selecting any value of $\alpha \geq 0$.

Theorem 9 (*Accuracy of the Classifier f*)**.** *Let $x \sim \mathcal{D}$. Then, the probability that x is correctly classified as class 0 by the classifier f is at least*

$$\begin{aligned} &P(x \sim \mathcal{D} \ : \ f(x) = 0) \\ &\quad \geq \sup_{\alpha \geq 0} \big[P(x \sim \mathcal{D} \ : \ |\phi(\Pi x)| \leq \alpha) \\ &\qquad\qquad - P(x \sim \mathcal{D} \ : \ d_\pi(x) > -\alpha) \big]. \end{aligned}$$

The proof of this result is given in Appendix D.1. The first term appearing on the right hand side controls how far the surface S may be expected to deviate from the plane π (and is therefore simply 1 in the case when $\phi \equiv 0$ and so $S = \pi$; in this case the optimal balancing of the terms will be obtained when $\alpha = 0$). The second term, on the other hand, estimates the probability that a point is correctly classified by the plane placed parallel to π, but offset by distance α to account for the variability of ϕ.

We demonstrate this result in the setting of a linear classifier with a distribution $\mathcal{E}$ which satisfies the SmAC condition of Definition 3 with radius $r > 0$ and centre c such that $d_\pi(c) = -\eta$ for some $\eta \in [0, r)$. Then, Theorem 9 takes the following form, from which we obtain Theorem 4 when $r = 1$ and $\eta = \epsilon$.

Corollary 10 (*Accuracy for SmAC Distributions*)**.** *Suppose that points with label 0 are sampled from the distribution $\mathcal{E}$, and suppose that $\phi \equiv 0$. Then, for $x \sim \mathcal{E}$, the probability that the classifier f correctly assigns x class 0 is at least*

$$P(x \sim \mathcal{E} \ : \ f(x) = 0) \geq 1 - \frac{1}{2} A\left(1 - \left(\frac{\eta}{r}\right)^2\right)^{\frac{n}{2}}.$$

We may also prove a generalised version of the susceptibility result of Theorem 5 in our abstract setting in Appendix D.2. The probability of sampling a data point which is susceptible to an adversarial attack of size δ may be bounded from below as in the following result. The form of this result is similar to that of Theorem 9, although we note the crucial difference in the second term.

Theorem 11 (*Susceptibility to Adversarial Perturbations*)**.** *Suppose that points with label 0 are sampled from the distribution $\mathcal{D}$. Then, for any $\delta > 0$, the probability that a point sampled at random from the class 0 is susceptible to an adversarial attack with Euclidean norm δ is at least*

$$\begin{aligned} &P(x \sim \mathcal{D} \ : \ \textit{there exists } s \in \mathbb{B}^n_\delta \textit{ with } f(x + s) \neq 0) \\ &\quad \geq \sup_{\alpha \geq 0} \big[P(x \sim \mathcal{D} \ : \ |\phi(\Pi x)| \leq \alpha) \\ &\qquad\qquad - P(x \sim \mathcal{D} \ : \ d_\pi(x) \leq \alpha - \delta) \big]. \end{aligned}$$

When applied to the SmAC distribution $\mathcal{E}$, this result takes the following form, from which we obtain Theorem 5.

Corollary 12 (*Susceptibility for SmAC Distributions*)**.** *Suppose that points with label 0 are sampled from the distribution $\mathcal{E}$, and suppose that $\phi \equiv 0$. Then, for any $\delta \in [\eta, r]$, the probability that a point sampled at random from the class 0 is susceptible to an adversarial attack with Euclidean norm δ is at least*

$$\begin{aligned} &P(x \sim \mathcal{E} \ : \ \textit{there exists } s \in \mathbb{B}^n_\delta \textit{ with } f(x + s) \neq 0) \\ &\quad \geq 1 - \frac{1}{2} A\left(1 - \left(\frac{\delta - \eta}{r}\right)^2\right)^{\frac{n}{2}}. \end{aligned}$$

We next derive a generalised version of Theorem 6, which bounds the probability of sampling a random perturbation which is adversarial for f. For this result, we assume that the surface S has some regularity, in the sense that the function ϕ is Lipschitz with constant $L \geq 0$; i.e. for any $\hat{x}, \hat{y} \in \pi$ we have $|\phi(\hat{x}) - \phi(\hat{y})| \leq L\|\hat{x} - \hat{y}\|$. Geometrically, for any $x \in \mathbb{R}^n$ this defines a cone of points containing x in which f is guaranteed to be constant. This property allows us to prove the following lower bound on σ by d_S in Appendix D.3, which may be viewed as a companion to (2)

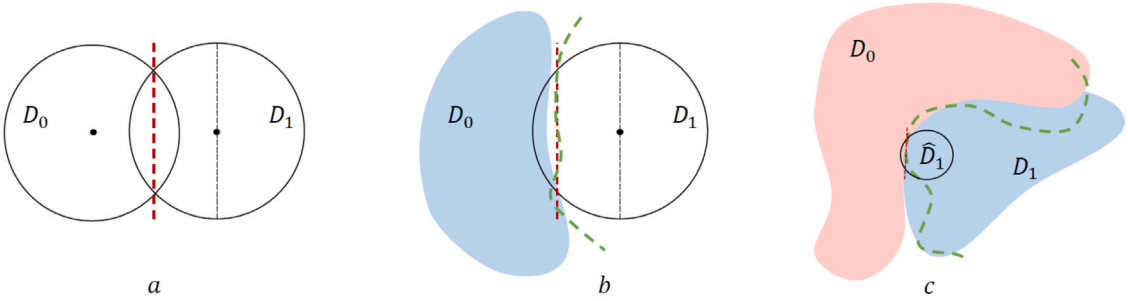


Fig. 5. Different scenarios to which the simple two ball model may be generalised.

Lemma 13 (*Lipschitz Regularity Gives Control of σ*)**.** *Suppose that ϕ is Lipschitz with parameter L. Then, for any $x \in \mathbb{R}^n$,*

$$\sigma(x) \geq |d_S(x)| \sin\theta, \tag{4}$$

where $\theta = \arctan(L^{-1})$.

This crucial property allows us to prove the following generalisation of Theorem 6 in Appendix D.3, indicating that destabilising random perturbations may be expected to be rare.

Theorem 14 (*Probability of Sampling Misclassifying Random Perturbations*)**.** *Suppose that points with label 0 are sampled from the distribution $\mathcal{D}$, and suppose that ϕ is Lipschitz with parameter L. Then, for any $\delta > 0$, the probability that a point sampled at random from the class 0 will be misclassified after the application of a perturbation randomly sampled uniformly from $\mathbb{B}^n_\delta$ is bounded by*

$$\begin{aligned}
&P(x \sim \mathcal{D}, s \sim \mathbb{B}^n_\delta \,:\, f(x+s) \neq 0) \\
&\leq \inf_{\substack{\alpha,\gamma \geq 0 \\ t \in T(L)}} \Big[P(x \sim \mathcal{D} \,:\, |\phi(\Pi x)| \geq \alpha) \\
&\qquad + P\Big(x \sim \mathcal{D} \,:\, d_\pi(x) \geq -\alpha - \frac{t}{\sin\theta}\Big) \\
&\qquad + \Delta(L)\frac{1}{2}\Big(1 - \Big(\frac{t}{\delta} - L\Big)^2\Big)^{\frac{n}{2}} \cdot \\
&\qquad\quad \cdot \big(P(x \sim \mathcal{D} \,:\, d_\pi(x) \leq \gamma - t) \\
&\qquad\qquad + P(x \sim \mathcal{D} \,:\, |\phi(\Pi x)| > \gamma)\big)\Big],
\end{aligned}$$

where $\Delta(L) = 1$ for $L \leq 1$ and 0 for $L > 1$, and the set $T(L) = [\min\{L, 1\}\delta, \delta]$.

For the SmAC distribution $\mathcal{E}$, Theorem 14 produces the following corollary (proved in Appendix D.3) from which Theorem 6 follows when $r = 1$ and $\eta = \epsilon$. In this case, we have $L = 0$ and so $\theta = \frac{\pi}{2}$ and $\sin\theta = 1$.

Corollary 15 (*Destabilising Random Perturbations are Rare for SmAC Distributions*)**.** *Suppose that points with label 0 are sampled from the distribution $\mathcal{E}$, and suppose that $\phi \equiv 0$. Then, for any $\delta \in [\eta, r]$, the probability that a point sampled a random from the class 0 is misclassified after the application of a perturbation sampled uniformly from the ball $\mathbb{B}^n_\delta$ is bounded by*

$$P(x \sim \mathcal{E}, s \sim \mathbb{B}^n_\delta \,:\, f(x+s) \neq 0) \leq A\Big(1 - \Big(\frac{\eta}{r+\delta}\Big)^2\Big)^{\frac{n}{2}}.$$

Finally, we also obtain a generalised analogue of the universality result of Theorem 8. We define the notion of the *destabilisation margin* in this setting to be the distance by which a perturbation pushes a data point across the decision threshold of the classifier (3). This is measured for class 0 by the function $d_0 : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$, where, for a data point x and a perturbation s,

$$d_0(x, s) = \max\{d_S(x+s), 0\}.$$

The following result then holds, as proven in Appendix D.4.

Theorem 16 (*Universality of Adversarial Attacks*)**.** *Suppose that $x, z \sim \mathcal{D}$ are independently sampled points with label 0, and suppose that ϕ is Lipschitz with parameter L. For any $\delta, \gamma \in \mathbb{R}$, the probability that x is destabilised by all perturbations $s \in \mathbb{B}^n_\delta$ which destabilise z with destabilisation margin $d_0(z, s) > \gamma$ is bounded from below by*

$$\begin{aligned}
&P(x, z \sim \mathcal{D} \,:\, f(x+s) \neq 0 \text{ for all } s \in S_z(\delta)) \\
&\geq \sup_{\alpha \geq 0, t \in \mathbb{R}} \Big[\Big(P(z \sim \mathcal{D} \,:\, |\phi(\Pi z)| \leq \alpha) \\
&\qquad - P(z \sim \mathcal{D} \,:\, d_\pi(z) > t + \chi) \Big) \cdot \\
&\qquad \cdot \Big(P(x \sim \mathcal{D} \,:\, |\phi(\Pi x)| \leq \alpha) \\
&\qquad - P(x \sim \mathcal{D} \,:\, d_\pi(x) \leq t - \chi) \Big) \Big],
\end{aligned}$$

where $\chi = \frac{1}{2}\gamma - L\delta - \alpha$, and for $z \in \mathbb{R}^n$ and $\delta \in \mathbb{R}$, we define $S_z(\delta) = \{s \in \mathbb{B}^n_\delta \,:\, d_0(z, s) > \gamma\}$.

For the SmAC distribution $\mathcal{E}$, this result takes the form shown in Corollary 17. Theorem 8 follows from this result in the case when $r = 1$ and $\eta = \epsilon$. Interestingly, this result does not depend on the perturbation size δ, due to the fact that the decision surface is assumed to be flat.

Corollary 17 (*Universality of Adversarial Perturbations for SmAC Distributions*)**.** *Suppose that points with label 0 are sampled from the distribution $\mathcal{E}$, and suppose that $\phi \equiv 0$. For any $\gamma \in \mathbb{R}$, the probability that x is destabilised by all perturbations $s \in \mathbb{B}^n$ which destabilise z with destabilisation margin $d_0(z, s) > \gamma$ is bounded from below by*

$$\begin{aligned}
&P(x, z \sim \mathcal{E} \,:\, f(x+s) \neq 0 \text{ for all } s \in S_z) \\
&\qquad \geq \Big(1 - A\Big(1 - \frac{\gamma^2}{4r^2}\Big)^{\frac{n}{2}}\Big)^2
\end{aligned}$$

4.4. Further generalisations

Despite their simplicity, the models presented above cover a wide variety of settings. The results in Section 4.2 include data sampled from many common distributions such as uniform distributions and truncated Gaussian distributions. This setup is depicted in Fig. 5a. The results may be naturally extended to a case where only one of the data classes is sampled from a distribution satisfying the SmAC property, or where the classifier's decision surface is only locally linear (such as ReLU networks), as illustrated in Fig. 5b. Furthermore, the results may be applied in a fully local sense, to locally SmAC distributions, and locally linear classifiers. This generalisation is shown in Fig. 5c.

The generalised setup introduced in Section 4.3 already incorporates general data distributions, and only assumes that the classifier's decision boundary is a Lipschitz warping of a plane in its normal direction. This setup can also be directly extended to incorporate other 'wiggly' decision surfaces. For instance, if S cannot be expressed as a modification of a hyperplane in its normal direction, one could instead consider the surfaces defined by the upper and lower graphs of S with respect to π. For example, if S is given by a multi-valued function, one could instead just take the maximum or minimum values, and where

necessary work with a Lipschitz extension of these surfaces. Our results extend to this case, albeit with some additional looseness reflecting the 'uncertainty' this imposes on the location of the decision surface of the classifier.

Our results also naturally extend to standard multiclass classification problems. In this case, the decision boundary separating any pair of classes may be viewed locally as a binary classifier, and our results therefore apply locally (as in Fig. 5c, for example). In regions of data space where a small number of classes meet (relative to the data dimension), analogous versions of the results will hold. This is because in these regions we can apply our result to the boundary between the sampled data point and each other class separately, and collect them together. The exponential nature of our bounds in the data dimension will therefore dwarf the additional looseness introduced by considering the class boundaries separately. To treat the situation when the number of classes meeting near a sampled data point is large relative to the data dimension, additional theoretical developments would be required. However, standard geometric arguments would suggest that these regions of data space would have only a very small measure, implying that data points from non-degenerate distributions are unlikely to be sampled from them.

We do not attempt to treat all these generalised scenarios here, in order to present the main ideas in a simple framework.

5. Class separation margins hide adversarial susceptibility

A further intriguing component of the paradox of apparent stability is that it may no longer occur when the two data classes are separable, but have no margin separating them.[1] To model this situation, we introduce the *two half-balls model*. The model comprises two data classes, with binary labels $\{0, 1\}$, each sampled uniformly from a half-ball in dimension $n > 0$. These half-balls have their flat face parallel to each other and are separated by distance $2\epsilon \geq 0$. Data of class 0 are sampled uniformly from the half-ball $D_0 = \{x \in \mathbb{R}^n \ : \ x + \epsilon\, \mathbf{e}_1 \in H^-\mathbb{B}^n\}$, where $\mathbf{e}_1 = (1, 0, \ldots, 0)^\top \in \mathbb{R}$, while data from class 1 are sampled uniformly from $D_1 = \{x \in \mathbb{R}^n \ : \ x - \epsilon\, \mathbf{e}_1 \in H^+\mathbb{B}^n\}$. Here, we use the notation $H^-\mathbb{B}^n = \{x \in \mathbb{B}^n \ : \ x \cdot \mathbf{e}_1 < 0\}$ and $H^+\mathbb{B}^n = \{x \in \mathbb{B}^n \ : \ x \cdot \mathbf{e}_1 > 0\}$. Any pair of data points x, y sampled with opposite classes therefore satisfy $\|x - y\| \geq 2\epsilon$. We denote the combined distribution by $\mathcal{D}_\epsilon = \mathcal{U}(D_0 \cup D_1)$.

A classification function which correctly labels this data for any $\epsilon \geq 0$ can be defined by

$$f(x) = \begin{cases} 0 & \text{if } x_1 < 0, \\ 1 & \text{otherwise.} \end{cases} \tag{5}$$

Data points sampled from either class are separated from the decision surface of this classifier by distance at least ϵ. On the other hand, for any $\delta > \epsilon$, there are clearly points sampled from near the boundary of each class susceptible to perturbations $s \in \mathbb{B}^n_\delta$ such that $f(x + s) \neq f(x)$. In high dimensions, concentration effects ensure that data points sampled from either class concentrate close to the flat surface of their respective half ball, and therefore close to the decision surface. This means that the probability of sampling a point which is susceptible to an adversarial attack is high, as encapsulated in the following result, which is proved in Appendix C.1.

Theorem 18 (*Susceptible Data Points are Typical*). *For any $\epsilon \geq 0$ and $\delta \in [\epsilon, 1 + \epsilon]$, the probability that a randomly sampled data point is susceptible to an adversarial attack with Euclidean norm δ grows exponentially to 1 with the dimension n, specifically*

$$P\big(x \sim \mathcal{D}_\epsilon \ : \ \text{there exists } s \in \mathbb{B}^n_\delta \text{ such that } f(x + s) \neq f(x)\big) \geq 1 - (1 - (\delta - \epsilon)^2)^{n/2}.$$

[1] We note that the two balls model of Section 4.2 is unable to capture this scenario since when $\epsilon = 0$ the two balls overlap and the classifier is just 50% accurate.

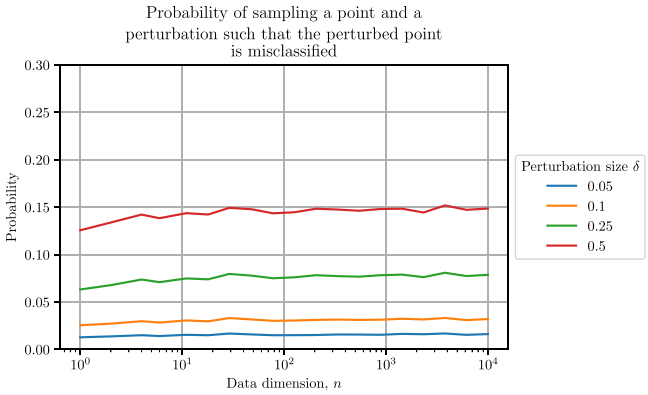


Fig. 6. Two half balls model with $\epsilon = 0$ — The empirical probability of sampling a point and perturbation of size δ such that the perturbed data point is misclassified. This empirical data was computed by sampling 10,000 points from the half-ball distribution and 10,000 perturbations from $\mathcal{U}(\mathbb{B}^n_\delta)$.

Analogously to Theorem 6, we also derive the following bound on the probability of a random perturbation destabilising a sampled data point, the proof of which is in Appendix C.2.

Theorem 19 (*Destabilising Perturbations are Rare*). *For any $\delta > \epsilon \geq 0$, the probability that a randomly selected perturbation with Euclidean norm δ causes a randomly sampled data point to be misclassified is bounded from above as*

$$P\big(x \sim \mathcal{D}_\epsilon, s \sim \mathcal{U}(\mathbb{B}^n_\delta) \ : \ f(x + s) \neq f(x)\big) \leq \frac{1}{4}\left(1 - \left(\frac{\epsilon}{\delta}\right)^2\right)^{n/2}.$$

Surprisingly, for $\epsilon = 0$ (when the two half balls meet along their flat faces) this probability does not converge to zero with increasing dimension n. To illustrate that this is not simply a looseness in the bound, we present empirical data in Fig. 6 demonstrating that the probability of sampling a destabilising perturbation at random remains approximately constant, even in high dimensions.

A deeper theoretical analysis reveals that the probability of a label swap in the model with $\epsilon = 0$ is always separated away from zero for all dimensions $n > 1$. In particular, the following result holds, the proof of which is provided in Appendix C.3.

Theorem 20 (*No Place to Hide When Margins are Zero*). *Consider the two half-balls model with $\epsilon = 0$, $n > 1$, and let $\delta > 0$. Then*

$$\lim_{n\to\infty} P(x \sim \mathcal{D}_\epsilon, \ s \sim \mathcal{U}(\mathbb{B}^n_\delta) : \ f(x + s) \neq f(x)) \geq \sup_{p \in (0,1)} 2p\left(1 - \Phi\left(\frac{\sqrt{2|\log(1-p)|}}{\delta}\right)\right),$$

where Φ is the standard cumulative distribution function.

According to Theorem 20, the probability of label swaps due to additive and independent random perturbations sampled from $\mathcal{U}(\mathbb{B}^n_\delta)$ does not converge to zero when n grows arbitrarily large in this model when $\epsilon = 0$. This is in stark contrast with the case when the separation margin ϵ is non-zero, where the analogous upper bound from Theorem 19 goes to zero with increasing dimension n. We conclude from these results that it is the presence of a non-zero margin $\epsilon > 0$ separating pairs of typical data points that is responsible for 'hiding' the adversarial susceptibility of the classifier such that it cannot be efficiently detected using random perturbations.

6. Discussion and relation to prior work

6.1. Existence of adversarial examples

Since the seminal work (Szegedy et al., 2014) reporting the discovery of adversarial examples in deep neural networks, the topic of adversarial examples as well as their origins and the mechanisms behind their occurrence have been the focus of significant attention in theoretical and computational machine learning communities. One hypothesis, expressed in Szegedy et al. (2014) was that the existence of the adversarial examples could be attributed to the inherent instabilities – i.e., large Jacobian norms leading to large Lipschitz constants for the classification maps. Theorems 5, 6 (see also Theorems 18 and 19 in Section 5) show that whilst the latter mechanism may indeed constitute a feasible route for adversarial examples to occur, our presented framework reveals a simple pathway for adversarial data to emerge naturally in systems without large Jacobian norms.

6.2. Fragility of adversarial examples

It has been empirically observed in Gupta, Dasgupta, and Akhtar (2020), Kurakin, Goodfellow, and Bengio (2018) that the capability of adversarial examples to fool the classifiers for which they have been designed can be hindered by perturbations and transformations which are naturally present in real-world environments. Here we show and prove (Theorems 6 and 19) that in the vicinity of the target images, adversarial examples may indeed occupy sets whose Lebesgue measure is exponentially small. Hence, the addition of a small but appropriate perturbation to an example of that type will have the capability to make it non-adversarial. Our results also show that simply adding random noise to an adversarially attacked image is very unlikely to produce something which would be correctly classified. Taken together, these two observations suggest that random image perturbations have a significantly different effect on standard image classification models from natural environmental changes to images.

6.3. Certifying robustness of classifiers to adversarial perturbations

There is a body of work in the literature dedicated to detecting, mitigating, and defending against adversarial attacks using randomly sampled noise; see, for example, the algorithms discussed in Cohen et al. (2019), Li et al. (2019), Ye et al. (2024) amongst many others. If many such randomly sampled perturbations are used, our results suggest that only a small fraction of them would change the classification of an image. Indeed, this fraction is exponentially small (in the data dimension n) when the classifier's decision surface is locally linear around the perturbed data point (Theorems 6 and 14). Equivalently, this suggests that such algorithms would need to take exponentially many (in n) samples to find even one which changes the classification. This implies that, to reliably detect or defend against an adversarial attack, algorithms based on this approach require an exponentially large computational complexity.

6.4. Universal adversarial perturbations

Another striking feature of adversarial examples is the existence of seemingly universal adversarial perturbations. These are small image-agnostic perturbations which can be applied to most, if not all, images in a dataset to cause the image to be misclassified by a given model. The phenomenon of universal adversarial perturbations was first reported in Moosavi-Dezfooli, Fawzi, Fawzi, and Frossard (2017) and since then observed in a wide range of tasks and architectures (Chaubey et al., 2020). Several explanations justifying the existence of universal adversarial perturbations have been proposed in the literature. This includes the view that universal perturbations may exploit correlated lower-dimensional structures in the classifier's decision boundaries. It has been less clear how to explain the simultaneous existence, fragility, typicality, and universality of adversarial perturbations. Theorems 5, 6, and 8 show that the combination of these correlations with the high dimensionality of data may explain the co-existence of the typicality of adversarial examples, their fragility, and at the same time universality.

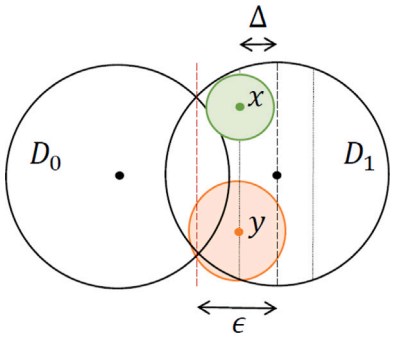


Fig. 7. Adversarial susceptibility of seemingly stable classifiers. Points x and y are in the Δ thickening of disc intersecting the ball D_1 along one of its largest equators. For n sufficiently large, most points sampled from $\mathcal{U}(D_1)$ belong to this domain. Both x and y are $(\epsilon-\Delta)$-stable. At the same time, they are also δ-stable with confidence $\upsilon \approx 1$.

6.5. Notions of stability

Our results reveals a new unexplored relationship between stability and the existence of adversarial data. We show that the ubiquitous presence of adversarial perturbations which destabilise the classifier is not contradictory to the robustness of the classifier to random perturbations of the data. If we view the former as a form of *deterministic instability* (i.e. there exist small, and potentially arbitrarily small, destabilising perturbations which can be constructed by an attacker), and the latter as a form of *probabilistic stability* (destabilising perturbations are unlikely to be sampled at random), it becomes apparent that the probabilistic stability is in fact masking the underlying instability. Since these two notions of stability are clearly not equivalent, it is imperative to understand the difference between the two. To clarify this intriguing relationship, let us first recall two relevant definitions of stability (cf. Huang, Yang, Liu, Jia, Ma, and Zhang (2022)).

Definition 21 (ϵ-*Stability*)**.** The classification map $f : \mathbb{R}^n \to \{0,1\}$ is ϵ-stable at x if

$$f(x+s) = f(x) \quad \text{for all} \quad s \in \mathbb{B}^n_\epsilon.$$

Otherwise, if there is an $s \in \mathbb{B}^n_\epsilon$ for which $f(x+s) \neq f(x)$, we say that the classification map f is not ϵ-stable at x, or that f is ϵ-unstable at x.

Definition 22 (ϵ-*Stability with Confidence* υ)**.** Let μ be a probability distribution on $\mathbb{B}^n_\epsilon$. The classification map $f : \mathbb{R}^n \to \{0,1\}$ is ϵ-stable at x with confidence υ w.r.t. the distribution μ if

$$P(s \sim \mu \ : \ f(x+s) = f(x)) \geq \upsilon.$$

At the core of the phenomenon explored in Theorems 5 and 6 is the fact that a "typical" point x is δ-stable with confidence υ with respect to perturbations sampled from $\mathcal{U}(\mathbb{B}^n_\delta)$, where υ approaches 1 exponentially in n. This makes the finding of adversarial perturbations by adding random samples $s \sim \mathcal{U}(\mathbb{B}^n_\delta)$ difficult and unlikely.

At the same time, for n sufficiently large, typical points are located in some $\Delta < \epsilon$ vicinity of the equators of the n-dimensional unit balls supporting $\mathcal{D}_0$ and $\mathcal{D}_1$. This implies that these typical points are $\epsilon - \Delta$-stable in the sense of Definition 21. This is visualised in the diagram shown in Fig. 7. In the absence of the margin ϵ separating the centres of $\mathcal{D}_0$ and $\mathcal{D}_1$, there is no room to "hide" adversarial examples among random perturbations. This leads to the intriguing observation that, for some appropriate value of ϵ:

The existence and prevalence of adversarial examples which are undetectable via random perturbations can be enabled by the ϵ-stability of 'typical' data samples.

This is further illustrated through the two half-balls model investigated in Section 5. The choice of sampling two data classes were sampled from complementary half-balls separated by margin $\epsilon \geq 0$ was motivated by its ability to represent two separable classes without any margin or overlap. As shown numerically in Fig. 6, in the absence of margins (which is an admissible case in the setup adopted in Shafahi, Huang, Studer, Feizi, and Goldstein (2019)) the probability of registering misclassifications due to random perturbation is significant and does not change much with dimension. This is confirmed theoretically by Theorem 20.

6.6. Other theoretical frameworks explaining the phenomenon of adversarial examples

Several works have presented feasible mechanisms explaining some elements of the paradox considered in this work. For example in Shafahi et al. (2019), Tyukin, Higham, and Gorban (2020) the authors exploited concentration of measure arguments to determine conditions when small destabilising perturbations can be typical in high dimensional settings. In Fawzi, Moosavi-Dezfooli, and Frossard (2016) the authors looked at the relationships between the relative sizes of class-altering perturbations in random directions and their worst-case counterparts (adversarial). Sample-inefficiency of robust training with random noise as well as the impact of the choice of norms have been discussed in Khoury and Hadfield-Menell (2018). Relevant geometric concepts explaining the feasibility of the expected emergence of adversarial examples have been suggested in Shamir et al. (2022) (the dimpled manifold hypothesis) and Tanay and Griffin (2016) (the boundary tilting mechanism).

In our work, we focused on presenting a single simple theoretical framework that could holistically explain the simultaneous rarity of destabilising random perturbations, the typicality of adversarial examples (see Fig. 6 and the discussion below), their universality, their potential fragility, and the relationship between the presence of non-zero independent on dimension separation margins (i.e. stability) and the possibility to successfully hide vulnerability to adversarial perturbations in the apparent robustness to random perturbations. Revealing the connection between all these phenomena within a single setting is a key feature of our framework.

The typicality of such coexistence in a broad class of problems distinguishes our work from other relevant theories and explanations focussing on showing the existence of tasks in which instabilities are expected in otherwise accurate classifiers (see e.g. Bastounis et al. (2021), Bastounis et al. (2023)).

7. Experimental investigation of the paradox of apparent stability

We experimentally explored the paradox of apparent stability using several standard benchmark image classification datasets. We first describe the experimental methodology in Section 7.1, and the results are reported in Section 7.2. The results for each benchmark are reported separately in the following sections:

- results for the CIFAR-10 dataset (Krizhevsky, 2009) are in Section 7.2.1
- results for the Fashion MNIST dataset (Xiao et al., 2017) are in Section 7.2.2
- results for the German Traffic Sign Recognition Benchmark (GTSRB) (Stallkamp et al., 2012) are in Section 7.2.3
- results for the ImageNet benchmark (Russakovsky et al., 2015) are in Section 7.2.4

Table 2
Architecture used for the binary classification problems. All convolutional layers do not pad their output, and are followed by a leaky ReLU activation function with leakiness parameter 0.1. The final dense layer has a standard sigmoid activation function. The number of trainable parameters depends on the size of the input data, and we use CIFAR-10 as an example.

Layer	Size	Output channels	Number of trained parameters
Conv-1	3×3	64	1792
Conv-2	3×3	64	36,928
Max pool	2×2		
Conv-3	3×3	128	73,856
Conv-4	3×3	128	147,584
Max pool	2×2		
Conv-5	3×3	256	295,168
Conv-6	3×3	256	590,080
Global max pool			
Dense		512	131,584
Dense		1	513

To present the phenomenon in the simplest possible setting, for CIFAR-10, Fashion MNIST and GTSRB, we arranged the n classes of each benchmark dataset into $\frac{1}{2}n(n-1)$ binary classification problems. A convolutional neural network was trained and assessed for each problem using a standardised protocol described below. To complement these experiments, we also investigated two pre-trained foundation models on the 1000-class image classification ImageNet benchmark, as described in Section 7.1.5.

The results here are presented normalised to the setting of images with pixel values in $[0, 1]$, regardless of the native scaling of the datasets or pre-trained models. This enables us to conveniently and comparably discuss the sizes of individual adversarial or random perturbations.

These results were computed using the CREATE HPC facilities at King's College London (King's College London, 2024).

7.1. Experimental setup

7.1.1. Network architecture

Convolutional neural networks were trained on each of these problems, using a similar architecture and training regime for each problem. Here, we describe the default settings, and any variations made for specific datasets are documented in the section describing the results computed on that dataset. We used a simplification of the VGG architecture (Simonyan & Zisserman, 2015), the details of which are given in Table 2.

For each pair-wise binary classification problem, the classes were assigned the labels 0 and 1, for compatibility with a standard sigmoid function on the output node of the network. A mean square error loss function was used to train the network in Tensorflow (Abadi et al., 2015) using stochastic gradient descent using a batch size of 128 for 100 epochs with Nesterov momentum parameter 0.9 and an initial learning rate of 0.1, which was halved every 20 epochs. Dropout was used on the convolutional layers during training, with a parameter of 0.4.

For the binary classification problem of distinguishing class i from class j, we denote the training set by $\mathcal{X}_{i,j}$, and the test set by $\mathcal{Y}_{i,j}$. The subsets of training and test images which were correctly classified by the network are then denoted by $\mathcal{X}^{\text{corr}}_{i,j} \subset \mathcal{X}_{i,j}$ and $\mathcal{Y}^{\text{corr}}_{i,j} \subset \mathcal{Y}_{i,j}$ respectively.

We are therefore able to compute the training and test accuracy of the network for the binary classification problem involving class i and class j as the percentages

$$100\frac{\text{card}(\mathcal{X}^{\text{corr}}_{i,j})}{\text{card}(\mathcal{X}_{i,j})} \quad \text{and} \quad 100\frac{\text{card}(\mathcal{X}^{\text{corr}}_{i,j})}{\text{card}(\mathcal{X}_{i,j})}, \tag{6}$$

respectively, where we use card to denote the cardinality of a set.

7.1.2. Adversarial attacks

To investigate the susceptibility of the networks to adversarial attacks, we used a standard gradient-based algorithm on a loss function, which can be viewed as an Euclidean version of the Fast Gradient Sign Method (FGSM) introduced in Goodfellow et al. (2015). Specifically, if $L(x, y, N)$ denotes the mean square error loss function evaluated on the neural network N at the target image x with label ℓ, we compute the adversarial attack direction as

$$a(x) = \frac{\nabla_x L(x, \ell, N)}{\|\nabla_x L(x, \ell, N)\|},$$

where $\|\cdot\|$ denotes the Euclidean norm. We then tested 256 equally-spaced scalings $\epsilon \in [0, 5]$ to determine the smallest value such that $|\ell - N(x + \epsilon a(x))| > \frac{1}{2}$. This value of ϵ therefore gives the Euclidean norm of the smallest perturbation (among those tested) in the direction of $a(x)$ such that the network therefore predicts the wrong class for the attacked image. The value of ϵ therefore provides an upper bound on the minimal Euclidean distance of the image x from the decision surface of the neural network N.

For the class i vs class j binary classification problem, we use $\mathcal{X}^{\text{adv}}_{i,j} \subset \mathcal{X}^{\text{corr}}_{i,j}$ to denote the set of training images $x \in \mathcal{X}^{\text{corr}}_{i,j}$ such that x was correctly classified by the network, but $x + \epsilon a(x)$ was misclassified for at least one of our tested values of ϵ. The equivalent subset of the test set is denoted by $\mathcal{Y}^{\text{adv}}_{i,j} \subset \mathcal{Y}^{\text{corr}}_{i,j}$. We may then define the *adversarial susceptibility* of the network for the training and test sets as the percentages

$$100\frac{\text{card}(\mathcal{X}^{\text{adv}}_{i,j})}{\text{card}(\mathcal{X}^{\text{corr}}_{i,j})}, \text{ and } 100\frac{\text{card}(\mathcal{Y}^{\text{adv}}_{i,j})}{\text{card}(\mathcal{Y}^{\text{corr}}_{i,j})}, \tag{7}$$

respectively, where we use card to denote the cardinality of a set.

7.1.3. Random perturbations

To assess the effect on the network of random perturbations to the images, we sampled a set P of 2000 random perturbations from a uniform distribution on the d-dimensional ball with Euclidean norm ≤ 1, where d denotes the number of individual attributes of an image from the dataset. Then, for each pair i, j of classes, we performed the following experiment. For each image x in the subsets $\mathcal{X}^{\text{adv}}_{i,j}$ and $\mathcal{Y}^{\text{adv}}_{i,j}$ of the training and test sets which were susceptible to an adversarial attack, we constructed the perturbed image $x+\delta\epsilon s$ for each $s \in P$, where ϵ denotes the Euclidean norm of the smallest successful adversarial attack on x, scaled by each value of $\delta \in \{1, 2, 5, 10\}$ sequentially. In other words, we evaluated the network on an image which was perturbed by a random perturbation with Euclidean norm scaled by a fixed multiple of that of the (known successful) adversarial attack.

For the class i vs class j binary classification problem, we define the set $\mathcal{X}^{\text{rand},\delta}_{i,j} \subset \mathcal{X}^{\text{adv}}_{i,j}$ as the set of images which were susceptible to one or more random perturbations with scaling factor δ, as described above. The set $\mathcal{Y}^{\text{rand},\delta}_{i,j} \subset \mathcal{Y}^{\text{adv}}_{i,j}$ is defined analogously on the test set of images.

This enables us to define the *random perturbation susceptibility* of each network for the training and test sets as the percentages

$$100\frac{\text{card}(\mathcal{X}^{\text{rand},\delta}_{i,j})}{\text{card}(\mathcal{X}^{\text{adv}}_{i,j})}, \text{ and } 100\frac{\text{card}(\mathcal{Y}^{\text{rand},\delta}_{i,j})}{\text{card}(\mathcal{Y}^{\text{adv}}_{i,j})}, \tag{8}$$

respectively for each tested value of δ, where we use card to denote the cardinality of a set.

7.1.4. Training with random perturbations

We explored the effect of applying additive random noise to images during training on adversarial robustness. For simplicity, we only explored this using the CIFAR-10 benchmark. To do this we inserted a layer at the beginning of the network architecture described in Table 2 which sampled noise from a prescribed distribution independently for each input and added it to the input. The precise random perturbation added to each image is therefore different each time the image is presented to the network during training. The random perturbation layer is only active during training, so does not affect how the trained network is assessed at test time. We experimented with noise sampled uniformly from the cube $[-a, a]^n$ (i.e. with maximum ℓ^∞ norm $a > 0$ and with noise sampled from the ball $\mathbb{B}^n_b$ (i.e. with maximum Euclidean norm $b > 0$), with $a \in \{0.1, 0.5, 1.0\}$, $b \in \{3.2, 16, 32\}$, where n is the dimension of a single image in the dataset. These values of a and b were selected to ensure that for each pair of a and b values the samples from each distribution would have approximately the same Euclidean norm on average. This enables us to observe whether the sampling distribution makes a significant impact on the results, independently of the magnitude. Each network was otherwise trained exactly as described in Section 7.1.1.

Table 3
CIFAR-10 — Class names associated with each class index.

Index	Name
0	Aeroplane
1	Automobile
2	Bird
3	Cat
4	Deer
5	Dog
6	Frog
7	Horse
8	Ship
9	Truck

7.1.5. ImageNet experimental setup

Experiments using the ImageNet image classification benchmark (Russakovsky et al., 2015) were performed using the pretrained VGG 19 (Simonyan & Zisserman, 2015) and ResNet50 (He et al., 2016) neural networks available from Tensorflow (Abadi et al., 2015). These architectures were selected because the VGG-19 network resembles the smaller networks we trained for the other datasets, while the ResNet50 architecture enables us to compare how our findings translate to a significantly different family of models. For these experiments, we sampled 20,480 images from the standard validation split of the ImageNet dataset, and assessed the accuracy, adversarial susceptibility and random susceptibility of each network as described above. Since ImageNet has 1000 classes, we ensured that every class was represented in the sampled data, although did not require the same number of images from each class. Our notions of adversarial and random susceptibility in this setting are 'one-vs-all': an adversarial attack or random perturbation is considered to cause a misclassification if it causes the predicted class label to change to any other class. Since our aim is simply to understand the relationship between random and worst-case perturbations, this treatment does not account for the widely-reported close semantic similarity between various pairs of ImageNet classes (see Beyer, Hénaff, Kolesnikov, Zhai, and van den Oord (2020), for example).

7.2. Experimental results

7.2.1. Experimental results on CIFAR-10

The English names associated with each of the 10 classes are provided in Table 3.

Network performance. The training and test accuracy of the networks trained on each of the binary classification problems is shown in Table 4. The mean accuracy on the training set of the networks trained for all of the binary classification problems was 99.57% (standard deviation 0.24), with a minimum of 98.74%. In comparison, the mean accuracy on the test set was 94.09% (standard deviation 3.78), with a minimum of 82.6%. These figures indicate that the networks were generally quite capable of learning these binary classification problems, despite the fact they were trained using the same regime for only 100 training epochs each, and no specific tweaks were applied to improve the performance of any network.

Table 4
CIFAR-10 — Accuracy of the networks on the binary classification problems, reported in the form 'train accuracy, test accuracy', where accuracy is calculated as the percentage of images which were correctly classified. The row and column headers indicate the classes used in each binary classification problem.

	1	2	3	4	5	6	7	8	9
0	99.88, 96.45	99.31, 91.70	99.40, 95.20	99.36, 94.65	99.67, 95.45	99.25, 96.05	99.49, 96.45	99.73, 94.10	99.77, 95.40
1		99.67, 96.65	99.46, 95.95	99.78, 98.10	99.75, 97.55	99.17, 96.85	99.91, 98.80	99.72, 96.85	99.77, 93.65
2			99.08, 85.20	99.70, 87.25	99.35, 87.05	99.68, 91.05	99.63, 92.90	99.58, 95.30	99.42, 95.80
3				98.74, 86.70	99.77, 82.60	99.09, 88.80	99.68, 91.35	99.53, 96.05	99.47, 95.05
4					99.59, 90.60	99.84, 94.90	99.78, 90.30	99.58, 97.20	99.42, 96.75
5						99.79, 93.95	99.72, 90.40	99.60, 96.85	99.41, 96.05
6							99.85, 96.70	99.40, 97.10	99.72, 97.20
7								99.70, 97.65	99.85, 97.60
8									99.78, 95.80

Adversarial attacks. We report the adversarial susceptibility of each network (as defined in Section 7.1.2) in Table 5. On average over all the binary classification problems, 85.0% of the training images were susceptible to an adversarial attack (standard deviation 9.71) with a minimum of 70.28%, while the average on the test set was 79.48% (standard deviation 7.91) with a minimum of 69.82%. We note that both minima were attained on the same task 'frog-vs-ship' (6-vs-8). In the vast majority of the binary classification problems, over 80% of images in the training and test sets could be adversarially attacked in such a way that they would be misclassified by the network. This demonstrates the susceptibility of all of the networks to adversarial attacks, implying that the decision surface in each case passes close to most of the points in the training and test sets.

To measure just how close the decision surface passes to each data point, we also explore the sizes of the computed adversarial perturbations measured in several norms. In Table 6 we show the mean and standard deviations over each training and test set of the Euclidean norms of the smallest computed adversarial attack on each image. Similarly, Table 7 shows the mean and standard deviations over each training and test set of the ℓ^1 norms of the adversarial attacks, while Table 8 shows the same information for the ℓ^∞ norms.

The summary statistics reported in these tables are broken down in violin plots for a representative sample of the binary classification problems (selected, for simplicity, as the 'i-vs-$(i+1)$' problems). These show an approximation of the distribution of the Euclidean norms (Fig. 9), ℓ^∞ norms (Fig. 10) and ℓ^1 norms (Fig. 8) of adversarial attacks. In each case, this is the distribution across the whole training or test set of the norm of the smallest misclassifying adversarial attack found for each image using the algorithm described in Section 7.1.2. It is clear from these plots that for the majority of the adversarial attacks the largest change to any individual pixel value is comparatively small: the ℓ^∞ norm is less than 0.2 for most of the images across all tasks. The ℓ^1 norms, on the other hand, compute the sum of the absolute values of all changes to all pixels, so are expected to be a much larger value. Scaling these ℓ^1 norms by the number of pixel channels ($32 \times 32 \times 3 = 3072$), we obtain the mean absolute change to a single pixel. Taking 100 as a representative maximum value for the ℓ^1 norm across the majority of cases, we can therefore observe that this would correspond to a mean absolute change of approximately 0.03. Comparing this value to a similarly representative value of less than 0.5 for the ℓ^∞ norm of the adversarial attack, it is clear that this implies that the attacks are typically very localised since most of the change must be focused in just a few pixels.

The plots also indicate that the networks trained on certain tasks (such as 'bird-vs-cat' (2-vs-3) and 'cat-vs-deer' (3-vs-4) seem to be much more susceptible to small adversarial attacks. We mean this in the sense that while the overall attack susceptibility (Table 5) is quite typical, the attacks themselves on these classes appear to have much smaller norms.

The conclusion from these experiments is that most points in all of the training and test sets lie very close to the decision surface of the neural network, implying that the networks are susceptible to small perturbations to most of their training and test data.

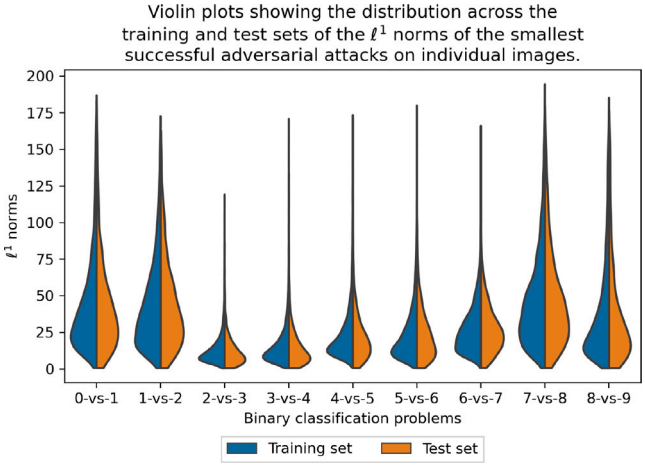


Fig. 8. CIFAR-10 — The distribution of the ℓ^1 norms of the successful adversarial attacks found for each image using the algorithm in Section 7.1.2, shown for a representative sample of the binary classification problems. The plotted distributions were fitted to the data using a standard Kernel Density Estimation algorithm and therefore only provide an approximation of the true empirical distribution.

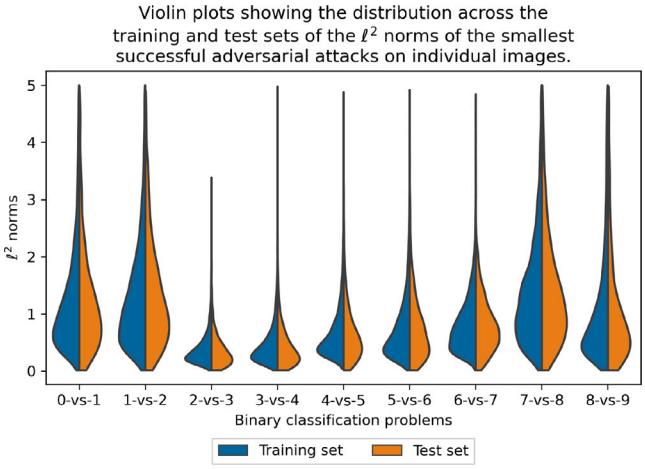


Fig. 9. CIFAR-10 — The distribution of the Euclidean norms of the successful adversarial attacks found for each image using the algorithm in Section 7.1.2, shown for a representative sample of the binary classification problems. The plotted distributions were fitted to the data using a standard Kernel Density Estimation algorithm and therefore only provide an approximation of the true empirical distribution.

Random perturbations. To explore whether the adversarial sensitivities described above could be triggered by random perturbations to the input data, we used the approach outlined in Section 7.1.3. We report the random perturbation susceptibility of each network (as defined in Section 7.1.3) in Table 9 for $\delta = 1$, Table 10 for $\delta = 2$, Table 11 for $\delta = 5$, and Table 12 for $\delta = 10$.

Table 5
CIFAR-10 — Susceptibility of the networks to adversarial attacks, reported in the form 'train susceptibility, test susceptibility', where susceptibility is calculated as in (7) as the percentage of images from the training set and test set which were misclassified after an adversarial attack using the algorithm described in Section 7.1.2. The row and column headers indicate the classes used in each binary classification problem.

	1	2	3	4	5	6	7	8	9
0	88.33, 86.88	95.69, 96.24	86.74, 86.08	95.19, 94.35	85.27, 83.97	83.43, 81.68	89.47, 88.02	92.78, 91.87	74.23, 73.69
1		86.25, 86.08	84.52, 83.79	93.59, 92.92	89.94, 88.83	85.23, 85.91	97.17, 97.17	89.62, 88.80	87.85, 88.04
2			91.88, 89.96	99.54, 99.66	96.93, 96.50	96.75, 96.05	92.97, 93.16	85.14, 83.89	92.73, 92.75
3				93.85, 92.85	99.86, 99.82	98.49, 98.65	98.95, 98.63	91.93, 92.04	78.84, 78.12
4					98.97, 98.34	98.99, 98.79	99.77, 99.56	80.96, 80.45	90.02, 90.08
5						96.19, 95.48	99.61, 99.61	84.14, 83.32	72.53, 71.63
6							98.63, 98.76	70.28, 69.82	87.35, 88.22
7								82.60, 83.56	96.85, 96.82
8									79.41, 78.03

Table 6
CIFAR-10 — Means and standard deviations of the Euclidean norms of the smallest successful adversarial attack on each image in the training and test set, reported in the form 'mean (standard deviation)'. The numbers in the row and column headers indicate the classes used in each binary classification problem. The 'train' row shows the values computed over the training set, while the 'test' row shows the values computed over the test set.

	1	2	3	4	5	6	7	8	9
0 train	1.25 (0.92)	0.76 (0.60)	1.19 (0.86)	0.96 (0.84)	1.32 (0.94)	1.21 (0.83)	1.24 (0.92)	0.84 (0.81)	1.20 (1.01)
test	1.25 (0.94)	0.79 (0.66)	1.21 (0.92)	0.97 (0.81)	1.36 (0.97)	1.24 (0.83)	1.26 (0.94)	0.83 (0.83)	1.25 (1.03)
1 train		1.32 (0.93)	1.00 (0.58)	1.13 (0.82)	1.27 (0.71)	0.96 (0.63)	1.50 (0.92)	1.08 (0.86)	0.75 (0.74)
test		1.35 (0.96)	1.00 (0.58)	1.14 (0.86)	1.28 (0.70)	0.99 (0.62)	1.47 (0.91)	1.06 (0.87)	0.74 (0.75)
2 train			0.40 (0.31)	0.48 (0.37)	0.64 (0.53)	0.56 (0.45)	0.86 (0.70)	1.05 (0.72)	1.21 (0.89)
test			0.37 (0.32)	0.47 (0.41)	0.63 (0.58)	0.57 (0.51)	0.86 (0.75)	1.06 (0.75)	1.22 (0.90)
3 train				0.47 (0.41)	0.40 (0.26)	0.74 (0.55)	0.58 (0.45)	1.27 (0.87)	0.72 (0.45)
test				0.44 (0.40)	0.35 (0.31)	0.73 (0.58)	0.57 (0.46)	1.29 (0.89)	0.72 (0.46)
4 train					0.68 (0.50)	0.54 (0.40)	0.65 (0.44)	1.17 (0.90)	1.03 (0.75)
test					0.67 (0.53)	0.53 (0.43)	0.63 (0.50)	1.21 (0.91)	1.07 (0.76)
5 train						0.75 (0.58)	0.73 (0.50)	1.33 (0.90)	0.94 (0.68)
test						0.77 (0.59)	0.70 (0.53)	1.32 (0.90)	0.98 (0.71)
6 train							0.82 (0.49)	1.38 (1.06)	0.94 (0.56)
test							0.83 (0.51)	1.42 (1.08)	0.97 (0.58)
7 train								1.39 (0.93)	1.05 (0.69)
test								1.41 (0.97)	1.05 (0.66)
8 train									1.14 (1.02)
test									1.13 (1.05)

Table 7
CIFAR-10 — Means and standard deviations of the ℓ^1 norms of the successful adversarial attacks on each training and test set, reported in the form 'mean (standard deviation)'. The numbers in the row and column headers indicate the classes used in each binary classification problem. The 'train' row shows the values computed over the training set, while the 'test' row shows the values computed over the test set.

	1	2	3	4	5	6	7	8	9
0 train	43.58 (31.63)	24.81 (19.52)	40.77 (30.04)	32.93 (29.12)	43.40 (31.07)	40.08 (27.55)	43.21 (32.21)	27.75 (26.27)	41.17 (34.03)
test	43.75 (32.53)	25.80 (21.35)	41.29 (31.76)	33.23 (28.07)	44.58 (31.90)	41.01 (27.42)	43.99 (32.61)	27.31 (26.69)	42.73 (35.09)
1 train		43.67 (30.41)	31.26 (18.66)	37.92 (26.88)	41.80 (23.68)	30.91 (20.83)	50.20 (30.37)	36.27 (28.58)	23.48 (22.71)
test		44.62 (31.28)	31.42 (18.83)	38.38 (28.13)	42.21 (23.44)	31.85 (20.46)	49.17 (29.95)	35.49 (28.86)	23.24 (23.52)
2 train			13.21 (10.15)	15.77 (12.02)	20.75 (17.32)	18.44 (14.49)	28.16 (23.22)	34.41 (23.75)	39.90 (29.16)
test			12.27 (10.59)	15.37 (13.53)	20.27 (18.76)	19.03 (16.61)	28.24 (24.51)	34.76 (24.91)	40.46 (29.26)
3 train				15.75 (13.50)	12.88 (8.26)	23.30 (17.50)	19.21 (14.39)	43.05 (29.57)	23.97 (15.48)
test				14.86 (13.22)	11.22 (9.67)	23.05 (18.58)	18.90 (15.04)	43.74 (30.26)	24.07 (15.69)
4 train					23.18 (17.02)	18.52 (13.60)	19.96 (13.64)	38.80 (30.40)	33.26 (23.93)
test					22.81 (18.14)	18.22 (14.59)	19.39 (15.33)	40.01 (30.97)	34.66 (24.06)
5 train						23.89 (18.77)	23.76 (15.43)	45.22 (30.63)	30.60 (22.05)
test						24.53 (19.03)	22.78 (16.78)	45.22 (30.79)	31.86 (23.00)
6 train							27.77 (16.80)	46.02 (35.74)	30.55 (18.78)
test							27.97 (17.57)	47.44 (36.49)	31.66 (19.39)
7 train								47.05 (31.64)	35.33 (22.86)
test								47.78 (33.38)	35.27 (22.17)
8 train									38.53 (34.28)
test									38.29 (35.17)

The remarkable story shown by this data is that the networks are almost universally insensitive to random perturbations to the images, even when those perturbations become quite drastic. This puzzling feature is demonstrated in Figs. 11 and 12, where we show examples from the 'aeroplane-vs-cat' binary classification problem (0-vs-3), of images of a cat and an aeroplane (original image in panel (a)) which were correctly classified by the network, alongside the same image after an adversarial attack which successfully changed the network's predicted class in panel (b). This adversarial attack makes only a small change to the image (the largest change to any single pixel channel is 0.19

Table 8
CIFAR-10 — Means and standard deviations of the ℓ^∞ norms of the smallest successful adversarial attack on each image in the training and test set, reported in the form 'mean (standard deviation)'. The numbers in the row and column headers indicate the classes used in each binary classification problem. The 'train' row shows the values computed over the training set, while the 'test' row shows the values computed over the test set.

	1	2	3	4	5	6	7	8	9
0 train	0.14 (0.11)	0.09 (0.08)	0.13 (0.09)	0.11 (0.10)	0.16 (0.12)	0.14 (0.10)	0.14 (0.11)	0.10 (0.10)	0.14 (0.12)
test	0.14 (0.11)	0.10 (0.09)	0.13 (0.10)	0.12 (0.10)	0.17 (0.13)	0.14 (0.10)	0.15 (0.12)	0.10 (0.10)	0.14 (0.12)
1 train		0.16 (0.12)	0.13 (0.08)	0.13 (0.10)	0.15 (0.09)	0.11 (0.07)	0.18 (0.12)	0.13 (0.11)	0.10 (0.10)
test		0.16 (0.13)	0.13 (0.08)	0.13 (0.11)	0.16 (0.09)	0.11 (0.07)	0.18 (0.12)	0.13 (0.11)	0.09 (0.10)
2 train			0.05 (0.04)	0.06 (0.04)	0.08 (0.07)	0.07 (0.06)	0.11 (0.10)	0.13 (0.09)	0.15 (0.12)
test			0.04 (0.04)	0.05 (0.05)	0.08 (0.07)	0.07 (0.06)	0.11 (0.11)	0.13 (0.10)	0.15 (0.12)
3 train				0.05 (0.05)	0.06 (0.04)	0.10 (0.07)	0.08 (0.06)	0.15 (0.11)	0.09 (0.06)
test				0.05 (0.05)	0.05 (0.05)	0.10 (0.08)	0.07 (0.07)	0.15 (0.11)	0.09 (0.06)
4 train					0.08 (0.06)	0.06 (0.05)	0.09 (0.07)	0.15 (0.11)	0.13 (0.10)
test					0.07 (0.06)	0.06 (0.06)	0.09 (0.07)	0.15 (0.11)	0.13 (0.10)
5 train						0.10 (0.08)	0.10 (0.08)	0.15 (0.11)	0.12 (0.09)
test						0.10 (0.08)	0.10 (0.08)	0.15 (0.11)	0.12 (0.10)
6 train							0.10 (0.06)	0.17 (0.13)	0.11 (0.07)
test							0.10 (0.06)	0.17 (0.14)	0.12 (0.07)
7 train								0.16 (0.11)	0.13 (0.09)
test								0.16 (0.12)	0.13 (0.08)
8 train									0.15 (0.14)
test									0.15 (0.15)

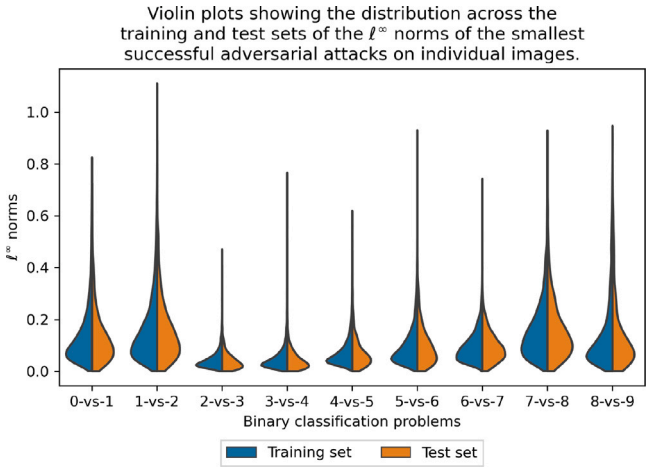


Fig. 10. CIFAR-10 — The distribution of the ℓ^∞ norms of the successful adversarial attacks found for each image using the algorithm in Section 7.1.2, shown for a representative sample of the binary classification problems. The plotted distributions were fitted to the data using a standard Kernel Density Estimation algorithm and therefore only provide an approximation of the true empirical distribution.

for the cat and 0.05 for the aeroplane). When the image is perturbed using a large random perturbation, as shown in panel (c), however, the network still produces the correct classification. For comparison, in panel (d) we show a random perturbation which caused the network to misclassify the image. Both of these random perturbations were obtained using $\delta = 10$ as in Section 7.1.3 and therefore have similar norms. To the human eye, however, there is no significant difference between the two randomly perturbed images, or between the original image and the adversarially attacked image. Even in these cases where a random perturbation was found which caused a misclassification, it is to be noted that only a small fraction of the 2000 sampled random perturbations did so (4.15% for the cat and 0.2% for the aeroplane).

The data from Tables 9–12 shows that as the scale of the random perturbations increases (as controlled by δ), so too does the probability of causing a perturbed image to be misclassified. In itself, this observation is unsurprising, but the data in Table 11 shows that, even when the random perturbations are scaled to be 5 times larger than the known adversarial attack (measured in the Euclidean norm, corresponding to $\delta = 5$), typically fewer than 5% of images were misclassified after applying any of the random perturbations.

In Figs. 13(b)–13(a) we show the distributions of the smallest random perturbation found to cause an image to be misclassified, as measured in the Euclidean, ℓ^∞ and ℓ^1 norms respectively, for $\delta = 10$ on a representative sample of the binary classification problems (the 'i-vs-$(i+1)$' problems, cf. Figs. 9–8). Recall that the random perturbations were sampled from a ball with Euclidean norm less than or equal to 1 (although high dimensional concentration phenomena ensure that all of the random perturbations have Euclidean norm very close to 1), and were scaled by δ times the Euclidean norm of the smallest successful adversarial attack when used to attack each image. This explains the underlying similarity between these distributions and those in Figs. 9–8, which show the size distributions of the adversarial attacks. However, it is readily apparent here once again that significantly larger random perturbations are required as compared to adversarial perturbations. This is visible (and shown in more detail for the 'cat-vs-aeroplane' problem (0-vs-3) in Fig. 14 from the fact that the random perturbation distributions appear to have much thicker tails than those for the adversarial perturbations; if simply a fixed fraction of all random perturbations were successful in causing an image to be misclassified then the distributions would shrink by a constant factor along their length.

Together, this evidence indicates that the decision surface does not pass close to the image in all directions, but rather only in one or a few specific adversarial directions.

Training with random perturbations. For brevity, we only report the results of these experiments for the representative subset of 'class i-vs-class $i+1$' binary classification problems. These are given in Tables 13, 15 and 17 for random perturbations sampled uniformly from the cube $[-a, a]^n$ with $a \in \{0.1, 0.5, 1.0\}$ respectively, and Tables 14, 16 and 18 for noise sampled uniformly from the ball $\mathbb{B}_b^n$ with $b \in \{3.2, 16, 32\}$ respectively (see Section 7.1.4 for details of the experimental setup). These results are also plotted against the size of the sampled perturbations in Figs. 15 and 16 for perturbations sampled from the cube and ball respectively. From these results, it is clear that additive random noise has little impact on the adversarial susceptibility of the networks, and large perturbations cause the networks' accuracy to decrease.

It should be stressed that additive noise sampled from the cube $[-1, 1]$ (the largest cube we tested) represents a significant modification to an image where each pixel value is in $[0, 1]$. We also recall that the adversarial susceptibility is only calculated as the fraction of correctly

Table 9
CIFAR-10 — Susceptibility of the networks to random perturbations, as described in Section 7.1.3 for $\delta = 1$. This is reported in the form 'train susceptibility, test susceptibility', where susceptibility is calculated as in (8) as the percentage of adversarially attackable images from each set which were misclassified after applying any of the 2000 random perturbations. The row and column headers indicate the classes used in each binary classification problem. Here, we use 0 without any trailing decimal places to indicate a value which was actually zero, and not simply rounded to zero when rounding to two decimal places.

	1	2	3	4	5	6	7	8	9
0	0, 0.06	0, 0.06	0, 0.06	0, 0.11	0, 0.12	0.01, 0	0.01, 0.12	0.10, 0.12	0.01, 0
1		0, 0.06	0, 0	0.01, 0.11	0, 0	0.04, 0.06	0, 0	0, 0.06	0.01, 0
2			0, 0.26	0.01, 0.06	0, 0.12	0, 0	0, 0	0, 0.06	0, 0
3				0.04, 0.12	0, 0.18	0, 0	0, 0.06	0, 0	0, 0
4					0, 0.06	0, 0.05	0, 0	0, 0.06	0.02, 0.06
5						0, 0.06	0, 0.06	0, 0	0, 0
6							0, 0	0, 0	0.01, 0.06
7								0, 0	0, 0
8									0.05, 0

Table 10
CIFAR-10 — Susceptibility of the networks to random perturbations, as described in Section 7.1.3 for $\delta = 2$. This is reported in the form 'train susceptibility, test susceptibility', where susceptibility is calculated as in (8) as the percentage of adversarially attackable images from each set which were misclassified after applying any of the 2000 random perturbations. The row and column headers indicate the classes used in each binary classification problem. Here, we use 0 without any trailing decimal places to indicate a value which was actually zero, and not simply rounded to zero when rounding to two decimal places.

	1	2	3	4	5	6	7	8	9
0	0.17, 0.42	0.02, 0.06	0, 0.06	0.07, 0.17	0.04, 0.31	0.01, 0	0.04, 0.29	1.11, 1.33	0.89, 0.78
1		0.02, 0.06	0.01, 0	0.03, 0.27	0, 0	0.05, 0.06	0, 0	0.59, 0.81	0.06, 0.36
2			0.02, 0.26	0.02, 0.12	0.06, 0.24	0.07, 0.06	0.01, 0.12	0, 0.19	0, 0
3				0.11, 0.31	0.01, 0.24	0.01, 0	0, 0.06	0.01, 0	0, 0
4					0, 0.17	0.03, 0.05	0, 0.06	0.05, 0.06	0.03, 0.11
5						0.03, 0.06	0, 0.11	0.01, 0	0.03, 0.07
6							0, 0	0.03, 0	0.01, 0.06
7								0, 0	0.04, 0
8									0.97, 1.47

Table 11
CIFAR-10 — Susceptibility of the networks to random perturbations, as described in Section 7.1.3 for $\delta = 5$. This is reported in the form 'train susceptibility, test susceptibility', where susceptibility is calculated as in (8) as the percentage of adversarially attackable images from each set which were misclassified after applying any of the 2000 random perturbations. The row and column headers indicate the classes used in each binary classification problem. Here, we use 0 without any trailing decimal places to indicate a value which was actually zero, and not simply rounded to zero when rounding to two decimal places.

	1	2	3	4	5	6	7	8	9
0	9.67, 10.56	1.55, 2.38	3.87, 4.58	9.94, 10.30	4.06, 4.93	1.75, 1.85	7.52, 8.42	8.93, 9.43	14.85, 16.57
1		7.61, 7.33	0.58, 0.25	5.99, 6.20	1.04, 1.27	1.11, 1.14	3.67, 3.12	11.06, 11.57	0.53, 0.75
2			0.12, 0.85	0.27, 0.40	0.61, 1.31	2.02, 2.46	1.65, 1.56	0.32, 0.56	2.30, 2.14
3				0.53, 0.99	0.25, 1.03	0.63, 0.63	0.23, 0.33	2.42, 2.94	0.34, 0.40
4					0.11, 0.56	0.44, 0.59	0.18, 0.56	3.46, 3.39	3.83, 3.96
5						2.97, 4.91	0.11, 0.33	2.36, 2.66	9.51, 9.74
6							0, 0.10	2.86, 3.24	0.07, 0.23
7								2.65, 2.57	1.40, 1.48
8									10.37, 10.37

Table 12
CIFAR-10 — Susceptibility of the networks to random perturbations, as described in Section 7.1.3 for $\delta = 10$. This is reported in the form 'train susceptibility, test susceptibility', where susceptibility is calculated as in (8) as the percentage of adversarially attackable images from each set which were misclassified after applying any of the 2000 random perturbations. The row and column headers indicate the classes used in each binary classification problem.

	1	2	3	4	5	6	7	8	9
0	41.49, 40.57	14.02, 16.71	29.34, 32.15	47.37, 49.55	27.49, 30.07	27.92, 28.11	50.40, 52.41	41.44, 42.16	50.82, 49.72
1		41.82, 42.67	29.38, 28.98	27.18, 28.96	41.01, 43.16	15.90, 17.79	30.90, 29.43	36.98, 36.92	29.42, 31.09
2			4.76, 6.78	16.74, 18.75	9.12, 10.48	34.10, 34.42	22.70, 23.11	19.20, 20.08	32.67, 31.80
3				5.54, 6.58	9.57, 11.04	19.86, 19.24	6.99, 8.49	28.58, 29.07	26.54, 28.01
4					5.59, 6.40	14.37, 14.77	3.87, 4.73	38.19, 39.58	31.07, 33.56
5						43.20, 42.81	6.99, 9.00	22.66, 23.23	56.10, 58.79
6							16.45, 16.70	26.87, 30.53	8.64, 8.92
7								42.90, 44.36	26.44, 26.14
8									45.04, 45.08

classified images which are susceptible to adversarial attacks, meaning that the drop in accuracy of the classifier is implicitly decreasing the pool of images which were tested for adversarial attacks. Interestingly, the average norm of the successful adversarial attacks does seem to increase with the size of the random perturbations applied during training. However, this could once again be due to the observed drop in accuracy: training and test points which were near the decision boundary of the original classifier trained without perturbations would be those which were susceptible to the smallest adversarial attacks. These would also be the points which would be most likely to be misclassified by the less accurate classifiers trained with randomly perturbed data, so would not be included when the adversarial attacks

Table 13
CIFAR-10 — Performance results when images are randomly perturbed during training using additive random noise sampled from the cube $[-a, a]^n$ with $a = 0.1$. The abbreviation 'Adv.' should be read as 'Adversarial'. The quantities computed are defined in Section 7.1. Accuracy and susceptibility are reported as percentages. The norms of the adversarial attacks are reported in the form 'mean (standard deviation)', calculated by averaging over all of the correctly classified and adversarially susceptible images in each of the training and test sets.

		0 vs 1	1 vs 2	2 vs 3	3 vs 4	4 vs 5	5 vs 6	6 vs 7	7 vs 8	8 vs 9
Accuracy	train	99.85	99.39	96.93	98.85	98.71	98.29	99.87	99.57	99.33
	test	96.40	96.60	82.45	85.90	87.45	92.45	96.35	98.05	94.75
Adv. susceptibility	train	87.85	95.14	90.49	99.96	99.46	98.67	98.87	84.85	95.94
	test	85.63	95.19	90.12	99.88	99.43	98.86	98.75	85.06	95.30
Adv. attack ℓ^1 norm	train	23.14 (35.17)	19.11 (27.94)	10.78 (17.35)	11.94 (16.84)	13.57 (18.69)	19.38 (26.04)	14.23 (19.03)	20.55 (33.22)	22.12 (32.41)
	test	23.33 (36.73)	19.75 (29.04)	10.42 (17.82)	11.92 (18.32)	12.66 (18.25)	20.58 (27.78)	14.39 (19.51)	20.68 (33.56)	22.46 (33.46)
Adv. attack ℓ^2 norm	train	0.68 (1.04)	0.57 (0.83)	0.32 (0.52)	0.34 (0.47)	0.40 (0.56)	0.59 (0.79)	0.42 (0.55)	0.60 (0.96)	0.64 (0.94)
	test	0.69 (1.08)	0.59 (0.87)	0.31 (0.54)	0.34 (0.51)	0.38 (0.54)	0.63 (0.84)	0.42 (0.57)	0.60 (0.97)	0.65 (0.97)
Adv. attack ℓ^∞ norm	train	0.07 (0.12)	0.06 (0.09)	0.03 (0.06)	0.04 (0.05)	0.04 (0.06)	0.08 (0.11)	0.05 (0.06)	0.07 (0.11)	0.08 (0.11)
	test	0.08 (0.12)	0.06 (0.10)	0.03 (0.06)	0.04 (0.06)	0.04 (0.06)	0.08 (0.11)	0.05 (0.06)	0.07 (0.11)	0.08 (0.12)

Table 14
CIFAR-10 — Performance results when images are randomly perturbed during training using additive random noise sampled from the ball $\mathbb{B}_b^n$ with $b = 3.2$. The abbreviation 'Adv.' should be read as 'Adversarial'. The quantities computed are defined in Section 7.1. Accuracy and susceptibility are reported as percentages. The norms of the adversarial attacks are reported in the form 'mean (standard deviation)', calculated by averaging over all of the correctly classified and adversarially susceptible images in each of the training and test sets.

		0 vs 1	1 vs 2	2 vs 3	3 vs 4	4 vs 5	5 vs 6	6 vs 7	7 vs 8	8 vs 9
Accuracy	train	99.83	99.69	95.33	99.02	99.39	99.45	99.82	98.64	99.45
	test	96.10	96.90	84.20	86.25	89.05	92.80	96.80	97.15	95.05
Adv. susceptibility	train	90.03	91.64	92.53	99.95	99.59	99.03	98.04	94.83	96.92
	test	88.29	91.12	92.58	100.00	99.38	98.81	97.99	94.96	96.79
Adv. attack ℓ^1 norm	train	25.16 (36.97)	18.58 (27.80)	9.94 (17.17)	11.45 (16.26)	11.95 (16.51)	17.81 (24.96)	14.97 (20.01)	22.23 (33.69)	22.58 (32.08)
	test	25.23 (37.82)	18.71 (28.24)	9.69 (16.88)	11.55 (18.15)	11.25 (16.89)	18.76 (25.94)	15.12 (21.37)	22.50 (34.12)	23.19 (33.62)
Adv. attack ℓ^2 norm	train	0.74 (1.10)	0.56 (0.83)	0.30 (0.53)	0.32 (0.46)	0.35 (0.49)	0.56 (0.78)	0.44 (0.59)	0.65 (0.99)	0.65 (0.93)
	test	0.74 (1.12)	0.56 (0.84)	0.30 (0.52)	0.33 (0.51)	0.33 (0.50)	0.58 (0.80)	0.44 (0.62)	0.66 (1.00)	0.67 (0.98)
Adv. attack ℓ^∞ norm	train	0.08 (0.13)	0.06 (0.10)	0.04 (0.07)	0.03 (0.05)	0.04 (0.06)	0.07 (0.11)	0.05 (0.07)	0.07 (0.12)	0.08 (0.11)
	test	0.08 (0.13)	0.06 (0.10)	0.04 (0.07)	0.03 (0.06)	0.04 (0.06)	0.08 (0.11)	0.05 (0.07)	0.07 (0.12)	0.08 (0.12)

Table 15
CIFAR-10 — Performance results when images are randomly perturbed during training using additive random noise sampled from the cube $[-a, a]^n$ with $a = 0.5$. The abbreviation 'Adv.' should be read as 'Adversarial'. The quantities computed are defined in Section 7.1. Accuracy and susceptibility are reported as percentages. The norms of the adversarial attacks are reported in the form 'mean (standard deviation)', calculated by averaging over all of the correctly classified and adversarially susceptible images in each of the training and test sets.

		0 vs 1	1 vs 2	2 vs 3	3 vs 4	4 vs 5	5 vs 6	6 vs 7	7 vs 8	8 vs 9
Accuracy	train	92.82	94.43	91.57	92.89	95.75	94.60	98.96	97.50	89.54
	test	89.50	92.05	81.50	82.80	86.95	88.80	94.65	95.40	85.35
Adv. susceptibility	train	76.02	98.56	98.21	99.26	98.55	94.27	99.20	95.57	83.88
	test	74.36	98.32	97.98	99.03	98.45	93.24	99.26	95.07	83.54
Adv. attack ℓ^1 norm	train	35.56 (50.17)	21.88 (33.12)	24.70 (34.77)	22.03 (33.51)	28.38 (35.91)	28.85 (37.80)	22.88 (29.34)	27.83 (39.89)	45.61 (55.12)
	test	35.17 (50.80)	21.82 (32.95)	25.02 (36.26)	21.81 (33.94)	27.13 (35.90)	29.06 (38.87)	23.70 (31.22)	27.50 (39.63)	45.39 (54.85)
Adv. attack ℓ^2 norm	train	1.01 (1.44)	0.65 (0.98)	0.70 (1.00)	0.61 (0.91)	0.82 (1.04)	0.87 (1.13)	0.67 (0.85)	0.78 (1.11)	1.27 (1.53)
	test	1.01 (1.47)	0.65 (0.98)	0.71 (1.04)	0.60 (0.93)	0.78 (1.04)	0.88 (1.17)	0.69 (0.91)	0.77 (1.10)	1.27 (1.53)
Adv. attack ℓ^∞ norm	train	0.10 (0.15)	0.07 (0.11)	0.07 (0.10)	0.06 (0.09)	0.08 (0.11)	0.10 (0.13)	0.07 (0.09)	0.08 (0.12)	0.13 (0.17)
	test	0.11 (0.16)	0.07 (0.11)	0.07 (0.11)	0.06 (0.09)	0.08 (0.11)	0.10 (0.13)	0.07 (0.10)	0.08 (0.12)	0.13 (0.17)

Table 16
CIFAR-10 — Performance results when images are randomly perturbed during training using additive random noise sampled from the ball $\mathbb{B}_b^n$ with $b = 16$. The abbreviation 'Adv.' should be read as 'Adversarial'. The quantities computed are defined in Section 7.1. Accuracy and susceptibility are reported as percentages. The norms of the adversarial attacks are reported in the form 'mean (standard deviation)', calculated by averaging over all of the correctly classified and adversarially susceptible images in each of the training and test sets.

		0 vs 1	1 vs 2	2 vs 3	3 vs 4	4 vs 5	5 vs 6	6 vs 7	7 vs 8	8 vs 9
Accuracy	train	95.08	96.21	88.05	93.85	96.42	90.75	98.97	97.06	89.75
	test	91.10	94.40	78.30	82.30	85.45	85.65	94.95	95.50	85.45
Adv. susceptibility	train	79.44	97.78	93.16	99.38	99.16	92.32	99.36	96.46	85.86
	test	77.22	97.72	92.40	99.45	99.06	90.37	99.53	96.07	85.08
Adv. attack ℓ^1 norm	train	34.08 (47.83)	25.09 (36.56)	32.75 (41.78)	20.49 (31.84)	26.35 (34.00)	33.95 (41.52)	24.59 (31.05)	30.17 (42.56)	47.84 (56.68)
	test	32.84 (46.83)	24.62 (36.20)	32.60 (42.07)	20.69 (33.00)	25.62 (34.86)	33.66 (42.43)	25.21 (32.42)	29.99 (42.89)	46.48 (55.40)
Adv. attack ℓ^2 norm	train	0.99 (1.40)	0.74 (1.07)	0.93 (1.19)	0.56 (0.87)	0.76 (0.98)	1.02 (1.24)	0.71 (0.89)	0.83 (1.17)	1.33 (1.57)
	test	0.97 (1.39)	0.73 (1.06)	0.93 (1.20)	0.57 (0.90)	0.74 (1.00)	1.01 (1.26)	0.73 (0.93)	0.83 (1.17)	1.29 (1.54)
Adv. attack ℓ^∞ norm	train	0.10 (0.15)	0.08 (0.11)	0.10 (0.13)	0.06 (0.09)	0.08 (0.10)	0.12 (0.14)	0.07 (0.09)	0.09 (0.12)	0.14 (0.17)
	test	0.10 (0.15)	0.07 (0.11)	0.09 (0.13)	0.06 (0.09)	0.07 (0.10)	0.11 (0.14)	0.07 (0.10)	0.08 (0.12)	0.13 (0.16)

Table 17
CIFAR-10 — Performance results when images are randomly perturbed during training using additive random noise sampled from the cube $[-a, a]^n$ with $a = 1$. The abbreviation 'Adv.' should be read as 'Adversarial'. The quantities computed are defined in Section 7.1. Accuracy and susceptibility are reported as percentages. The norms of the adversarial attacks are reported in the form 'mean (standard deviation)', calculated by averaging over all of the correctly classified and adversarially susceptible images in each of the training and test sets.

		0 vs 1	1 vs 2	2 vs 3	3 vs 4	4 vs 5	5 vs 6	6 vs 7	7 vs 8	8 vs 9
Accuracy	train	85.65	89.84	82.44	85.20	85.20	90.75	93.56	94.50	76.09
	test	84.00	88.40	76.40	80.40	81.30	87.75	91.15	92.45	74.60
Adv. susceptibility	train	75.66	98.90	93.92	97.79	93.96	97.41	98.57	96.96	61.51
	test	74.40	98.81	92.41	98.07	93.97	97.26	98.08	97.13	61.93
Adv. attack ℓ^1 norm	train	47.29 (60.24)	22.74 (35.90)	38.53 (49.56)	25.62 (41.16)	46.07 (52.80)	26.32 (39.04)	39.02 (45.31)	31.71 (44.80)	55.25 (68.60)
	test	45.52 (58.87)	22.06 (35.82)	38.01 (49.77)	25.61 (41.41)	45.18 (52.78)	27.85 (41.27)	39.61 (46.00)	31.00 (44.60)	54.28 (67.73)
Adv. attack ℓ^2 norm	train	1.29 (1.64)	0.66 (1.04)	1.06 (1.36)	0.67 (1.08)	1.29 (1.48)	0.77 (1.14)	1.12 (1.30)	0.86 (1.21)	1.44 (1.78)
	test	1.25 (1.61)	0.64 (1.03)	1.05 (1.36)	0.67 (1.08)	1.26 (1.47)	0.82 (1.21)	1.13 (1.31)	0.84 (1.21)	1.41 (1.76)
Adv. attack ℓ^∞ norm	train	0.12 (0.15)	0.07 (0.11)	0.10 (0.13)	0.06 (0.10)	0.13 (0.15)	0.08 (0.12)	0.11 (0.13)	0.08 (0.12)	0.13 (0.16)
	test	0.11 (0.15)	0.06 (0.10)	0.10 (0.13)	0.06 (0.10)	0.12 (0.15)	0.08 (0.13)	0.11 (0.13)	0.08 (0.12)	0.12 (0.16)

Table 18
CIFAR-10 — Performance results when images are randomly perturbed during training using additive random noise sampled from the ball $\mathbb{B}_b^n$ with $b = 32$. The abbreviation 'Adv.' should be read as 'Adversarial'. The quantities computed are defined in Section 7.1. Accuracy and susceptibility are reported as percentages. The norms of the adversarial attacks are reported in the form 'mean (standard deviation)', calculated by averaging over all of the correctly classified and adversarially susceptible images in each of the training and test sets.

		0 vs 1	1 vs 2	2 vs 3	3 vs 4	4 vs 5	5 vs 6	6 vs 7	7 vs 8	8 vs 9
Accuracy	train	86.82	91.61	84.19	85.79	84.17	90.45	94.29	93.40	80.22
	test	84.55	89.85	78.55	80.25	80.80	88.10	91.35	91.20	77.25
Adv. susceptibility	train	72.14	98.50	94.35	98.18	92.23	97.63	98.26	98.73	68.44
	test	71.08	98.16	93.19	98.32	92.51	97.73	98.14	98.85	68.41
Adv. attack ℓ^1 norm	train	43.25 (58.85)	25.08 (38.16)	34.84 (47.12)	25.38 (40.59)	48.40 (54.21)	25.85 (39.18)	38.47 (45.03)	29.10 (42.32)	53.87 (66.98)
	test	42.76 (58.44)	24.30 (37.83)	34.82 (48.19)	25.47 (40.99)	46.92 (53.64)	27.88 (42.30)	39.74 (46.60)	28.83 (42.06)	54.82 (67.20)
Adv. attack ℓ^2 norm	train	1.18 (1.61)	0.72 (1.09)	0.95 (1.28)	0.67 (1.06)	1.36 (1.52)	0.75 (1.12)	1.10 (1.29)	0.78 (1.13)	1.41 (1.75)
	test	1.18 (1.61)	0.69 (1.07)	0.95 (1.31)	0.67 (1.07)	1.32 (1.50)	0.81 (1.22)	1.13 (1.33)	0.77 (1.13)	1.43 (1.76)
Adv. attack ℓ^∞ norm	train	0.11 (0.15)	0.07 (0.11)	0.09 (0.12)	0.06 (0.09)	0.13 (0.16)	0.07 (0.11)	0.10 (0.13)	0.07 (0.11)	0.13 (0.16)
	test	0.11 (0.16)	0.07 (0.11)	0.09 (0.13)	0.06 (0.09)	0.13 (0.15)	0.08 (0.12)	0.11 (0.13)	0.07 (0.11)	0.13 (0.16)

Table 19
Fashion MNIST — Class names associated with each class index.

Index	Name
0	T-shirt/top
1	Trouser
2	Pullover
3	Dress
4	Coat
5	Sandal
6	Shirt
7	Sneaker
8	Bag
9	Ankle boot

were computed. Consequently, while large additive random noise may eliminate some of the smallest adversarial attacks, it does so at the expense of a significant drop in accuracy.

7.2.2. Experimental results on the Fashion MNIST dataset

The Fashion MNIST dataset consists of 28×28 pixel grayscale image (which we converted to RGB by simply duplicating the channels), separated into 10 classes. The English names for these classes are given in Table 19 The network structure used in this case is similar to that described in Table 2, but with the layers Conv-5 and Conv-6 removed. The same training and evaluation procedures outlined in Section 7.1 were applied, and the results are given below. For brevity, we only present the results on the problems of the form 'class i-vs-class $i + 1$'.

Table 20 shows the accuracy and susceptibility to adversarial and random perturbations of the network trained on each binary classification problem. The random perturbations are categorised by their size in the Euclidean norm in the form of δ, since they are uniformly sampled at random from the ball with radius $\delta\epsilon$, where ϵ denotes the Euclidean norm of the smallest adversarial perturbation identified on each image, while Table 21 shows the norms of the adversarial attack constructed with the smallest Euclidean norm on each image.

Violin plots for the distributions of the ℓ^1, Euclidean and ℓ^∞ norms of the successful adversarial and random perturbations are given in Figs. 17 and 18.

7.2.3. Experimental results on the German Traffic Sign Recognition Benchmark dataset (GTSRB)

The German Traffic Sign Recognition Benchmark (GTSRB) dataset consists of RGB colour images with size $30 \times 30 \times 3$, divided into more than 40 classes. Here, we have demonstrated our results using six of these 40 classes, selected to be relatively visually distinct from each other, and therefore to produce binary classification problems which may be expected to be more robust to adversarial attacks. The names of these data classes is given in Table 22. The network structure used in this case is the same as the Fashion MNIST dataset in Section 7.2.2, which is as described in Table 2 but without Conv-5 or Conv-6. The experimental setup was otherwise as described in Section 7.1. For brevity, we report the results on the problems of the form 'class i-vs-class $i + 1$'.

Table 23 shows the accuracy and susceptibility to adversarial and random perturbations of the network trained on each binary classification problem. The random perturbations are categorised by their size in the Euclidean norm in the form of δ, since they are uniformly sampled at random from the ball with radius $\delta\epsilon$, where ϵ denotes the Euclidean norm of the smallest adversarial perturbation identified on each image, while Table 24 shows the norms of the adversarial attack constructed with the smallest Euclidean norm on each image.

Violin plots for the distributions of the ℓ^1, Euclidean and ℓ^∞ norms of the successful adversarial and random perturbations are shown in Figs. 19 and 20.

7.2.4. Experimental results on ImageNet

The ImageNet dataset (Russakovsky et al., 2015) consists of RGB images of various sizes from 1000 classes. We experimented using a

Table 20
Fashion MNIST — Accuracy and susceptibility of the networks to adversarial and random attacks, reported as percentages in the form 'train, test'.

	0 vs 1	1 vs 2	2 vs 3	3 vs 4	4 vs 5	5 vs 6	6 vs 7	7 vs 8	8 vs 9
Accuracy	99.44, 99.35	99.51, 99.40	97.73, 96.85	97.77, 96.45	99.95, 99.95	99.96, 99.80	99.98, 99.95	99.84, 99.70	99.41, 99.35
Adversarial susceptibility	56.42, 56.87	61.81, 62.73	26.29, 27.36	62.10, 61.43	29.51, 29.96	38.73, 38.28	26.95, 28.21	53.58, 53.01	55.36, 55.81
Random susceptibility ($\delta = 1$)	0, 0	0, 0	0.03, 0	0, 0	0, 0	0, 0	0.06, 0	0.02, 0	0, 0
Random susceptibility ($\delta = 2$)	0, 0	0, 0	0.26, 0.38	0.07, 0.08	0, 0	0.58, 0.39	1.30, 1.42	0.87, 1.04	0.05, 0.09
Random susceptibility ($\delta = 5$)	2.48, 3.27	0.45, 0.32	4.25, 5.47	7.56, 7.68	10.71, 13.52	22.06, 23.69	16.73, 19.15	30.39, 31.98	13.72, 13.35
Random susceptibility ($\delta = 10$)	52.12, 52.92	50.39, 49.32	54.30, 55.47	56.84, 55.44	56.63, 57.43	61.88, 59.55	67.92, 71.45	84.33, 84.58	84.77, 83.95

Table 21
Fashion MNIST — Means and standard deviations of the norms of the smallest successful adversarial attack on each image in the training and test set, reported in the form 'mean (standard deviation)'.

		0 vs 1	1 vs 2	2 vs 3	3 vs 4	4 vs 5	5 vs 6	6 vs 7	7 vs 8	8 vs 9
Adv. attack ℓ^1 norm	train	68.18 (30.79)	83.46 (34.56)	59.53 (36.77)	50.06 (34.27)	59.45 (26.16)	49.61 (27.06)	54.79 (26.78)	47.29 (25.40)	52.12 (24.77)
	test	68.12 (30.56)	83.34 (35.36)	59.12 (38.72)	49.78 (35.26)	61.72 (26.30)	52.11 (27.52)	57.76 (26.42)	47.41 (23.36)	52.84 (25.64)
Adv. attack ℓ^2 norm	train	2.75 (1.17)	3.07 (1.16)	2.39 (1.33)	1.76 (1.20)	2.60 (1.15)	2.40 (1.24)	2.63 (1.20)	2.22 (1.08)	2.32 (1.12)
	test	2.75 (1.15)	3.06 (1.19)	2.36 (1.40)	1.75 (1.24)	2.70 (1.16)	2.52 (1.26)	2.79 (1.17)	2.24 (1.01)	2.34 (1.14)
Adv. attack ℓ^∞ norm	train	0.45 (0.22)	0.41 (0.17)	0.34 (0.19)	0.23 (0.16)	0.29 (0.14)	0.32 (0.17)	0.33 (0.15)	0.32 (0.15)	0.28 (0.14)
	test	0.44 (0.21)	0.41 (0.17)	0.34 (0.19)	0.23 (0.17)	0.30 (0.14)	0.33 (0.17)	0.35 (0.15)	0.33 (0.15)	0.29 (0.15)

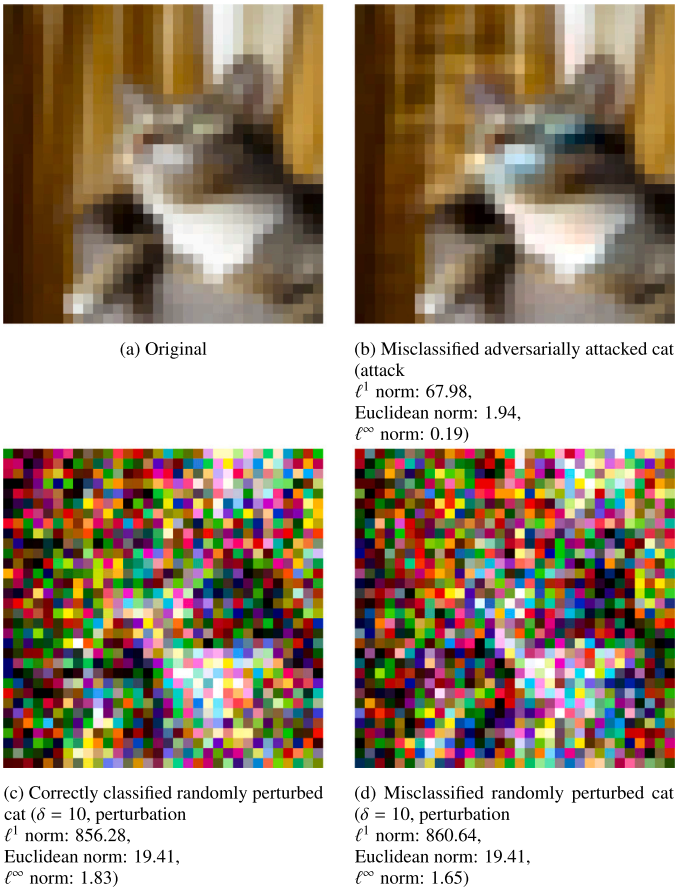
Table 22
GTSRB — Class names associated with each class index.

Index	Class name
0	Speed limit (20 km/h)
1	End of no passing for vehicles > 3.5 tons
2	Keep right
3	Turn right ahead
4	Road work
5	General caution
6	End of speed limit (80 km/h)

pre-trained VGG19 (Simonyan & Zisserman, 2015) and ResNet50 (He et al., 2016) network as described in Section 7.1.5.

Table 25 summarises the results of these experiments. In contrast to previous experiments, we only report random susceptibility for $\delta = 10$. Experiments with random perturbations for $\delta \in \{1, 2, 5\}$ produced virtually zero misclassifications, despite the apparently high adversarial susceptibility of the networks, and for brevity the detailed results are not reported here.

Figs. 21 and 22 show distributions of the sizes of successful adversarial attacks measured in different norms for each model. It is clear from these plots that in both cases the majority of images are susceptible to small adversarial perturbations. For ResNet50, 93.3% of images were susceptible to an adversarial attack with ℓ^∞ norm (measuring the absolute value of the largest change to any individual pixel) less than 0.1, while this was 94.3% for VGG19. Despite this, the results in Table 25 show that $\leq 2.5\%$ of images were misclassified by any of the 2000 random attacks sampled from the ball with radius 10 times larger than the Euclidean norm of the adversarial attack. We observe that the rate of misclassification after these large random perturbations is significantly smaller than for the other datasets. The theoretical results presented above suggest that this may be due to the much higher dimensionality of the images. Both of the pretrained networks we use accept inputs of size $224 \times 224 \times 3$, meaning that they have 150,528 individual attributes, in contrast with 3,072 attributes for CIFAR-10 images.



(a) Original

(b) Misclassified adversarially attacked cat (attack ℓ^1 norm: 67.98, Euclidean norm: 1.94, ℓ^∞ norm: 0.19)

(c) Correctly classified randomly perturbed cat ($\delta = 10$, perturbation ℓ^1 norm: 856.28, Euclidean norm: 19.41, ℓ^∞ norm: 1.83)

(d) Misclassified randomly perturbed cat ($\delta = 10$, perturbation ℓ^1 norm: 860.64, Euclidean norm: 19.41, ℓ^∞ norm: 1.65)

Fig. 11. CIFAR-10 — An example of an adversarially attacked image of a cat, taken from the 'cat-vs-aeroplane' binary classification problem (0-vs-3), alongside examples of large random perturbations to the same image which did and did not cause the network to misclassify the image. Of the 2,000 sampled random perturbations, 83 (4.15%) caused this image to be misclassified. Components of the modified image which were outside the range $[0, 1]$ have been clipped into the range for plotting, although not for the classification.

8. Conclusion

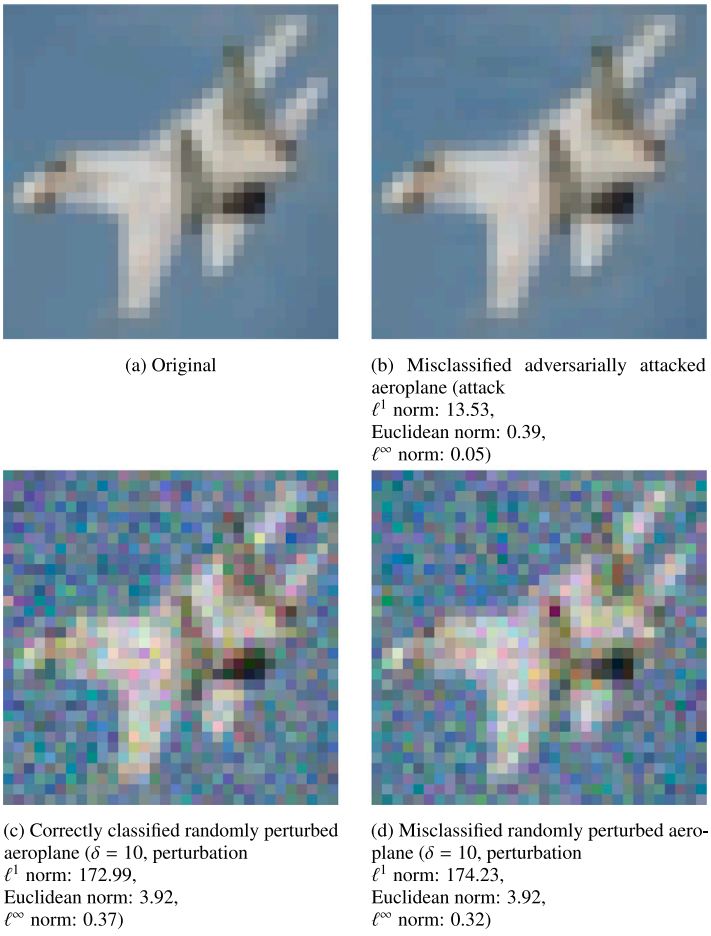
Our new framework for studying the paradox of apparent stability in classification problems allows for rigorous probabilistic bounds that are consistent with empirical observations concerning the simultaneous vulnerability to easily constructed worst-case adversarial attacks (Theorems 5 and 7) which may universally affect a whole data class (Theorem 8), and robustness against randomly sampled perturbations (Theorem 6). The results are generic in the sense that they deal with small perturbations under which any smooth and accurate classifier will behave like the optimal linear classifier (1). As illustrated in Fig. 5 and Section 4.4, the setup can be generalised to cover a broad range of input distributions and classification boundaries and multi-class setups. In addition to quantifying vulnerabilities, our analysis also raises new

Table 23
GTSRB — Accuracy and susceptibility of the networks to adversarial and random attacks, reported in the form 'train, test'.

	0 vs 1	1 vs 2	2 vs 3	3 vs 4	4 vs 5
Accuracy	96.84, 98.51	98.04, 97.63	99.95, 99.28	98.32, 98.14	100.00, 100.00
Adversarial susceptibility	94.44, 94.70	34.66, 32.98	62.96, 66.14	77.53, 77.00	88.95, 89.35
Random susceptibility ($\delta = 1$)	0, 0	2.52, 1.84	0, 0	0, 0	0.12, 0
Random susceptibility ($\delta = 2$)	0, 0	3.06, 2.76	0.17, 0.36	0.44, 0.19	0.36, 0.41
Random susceptibility ($\delta = 5$)	32.18, 32.80	5.76, 3.23	4.96, 5.10	22.11, 23.63	2.63, 3.01
Random susceptibility ($\delta = 10$)	61.25, 62.40	15.65, 10.60	39.26, 36.07	83.26, 81.85	32.18, 31.64

Table 24
GTSRB — Means and standard deviations of the norms of the smallest successful adversarial attack on each image in the training and test set, reported in the form 'mean (standard deviation)'.

		0 vs 1	1 vs 2	2 vs 3	3 vs 4	4 vs 5
Adv. attack ℓ^1 norm	train	42.38 (31.92)	4.20 (7.13)	13.42 (14.52)	29.80 (22.24)	15.55 (12.56)
	test	46.09 (35.10)	4.00 (5.91)	12.22 (13.13)	28.61 (22.18)	15.36 (12.53)
Adv. attack ℓ^2 norm	train	1.84 (1.22)	0.25 (0.48)	0.74 (0.80)	1.70 (1.25)	0.85 (0.71)
	test	1.95 (1.29)	0.24 (0.35)	0.67 (0.72)	1.61 (1.24)	0.83 (0.71)
Adv. attack ℓ^∞ norm	train	0.31 (0.21)	0.05 (0.09)	0.15 (0.16)	0.33 (0.26)	0.17 (0.16)
	test	0.31 (0.20)	0.04 (0.06)	0.14 (0.14)	0.31 (0.25)	0.17 (0.16)



(a) Original

(b) Misclassified adversarially attacked aeroplane (attack ℓ^1 norm: 13.53, Euclidean norm: 0.39, ℓ^∞ norm: 0.05)

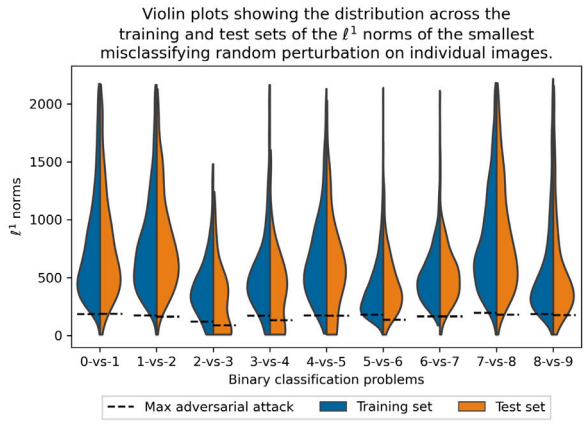
(c) Correctly classified randomly perturbed aeroplane ($\delta = 10$, perturbation ℓ^1 norm: 172.99, Euclidean norm: 3.92, ℓ^∞ norm: 0.37)

(d) Misclassified randomly perturbed aeroplane ($\delta = 10$, perturbation ℓ^1 norm: 174.23, Euclidean norm: 3.92, ℓ^∞ norm: 0.32)

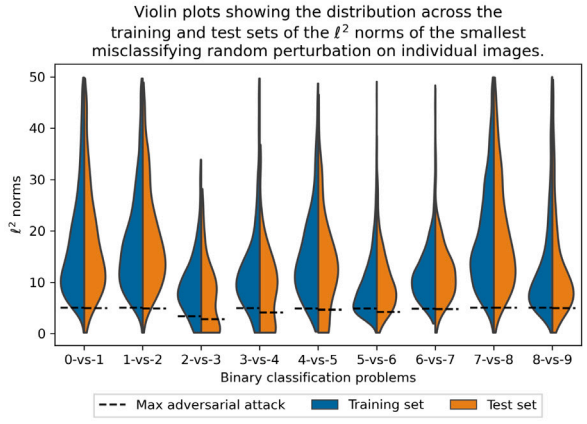
Fig. 12. CIFAR-10 — An example of an adversarially attacked image of an aeroplane, taken from the 'cat-vs-aeroplane' binary classification problem (0-vs-3), alongside examples of large random perturbations to the same image which did and did not cause the network to misclassify the image. Of the 2000 sampled random perturbations, 4 (0.2%) caused this image to be misclassified. Components of the modified image which were outside the range $[0, 1]$ have been clipped into the range for plotting, although not for the classification.

issues concerning the most relevant and useful notions of stability in classification.

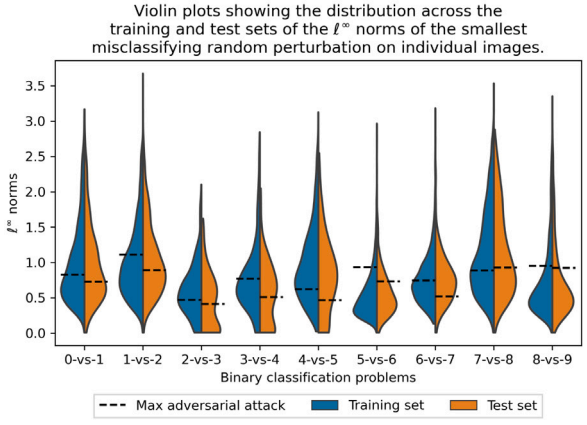
The overlapping unit ball model that we used, and the two half-ball model in Section 5, are closely tied to the use of the Euclidean norm. We note that there are several applications where spherical input data arises naturally, including remote sensing, climate change modelling, global ionospheric prediction and environmental governance, Feng, Huang, and Zhou (2023). It would of course be of interest to establish the extent to which these results can be extended to other choices of norm and input domain. We also note that more customised results



(a) ℓ^1 norms



(b) ℓ^2 norms



(c) ℓ^∞ norms

Fig. 13. CIFAR-10 — The distribution over all images in $\mathcal{X}_{i,j}^{\text{rand},10}$ (from the training set, see Section 7.1.3) and $\mathcal{Y}_{i,j}^{\text{rand},10}$ (from the test set) of the smallest norm of a random perturbation which caused the network to misclassify the image. Black dashed lines show the size of the largest adversarial attack required on each data set. These were fitted to the data using a standard Kernel Density Estimation algorithm and therefore only provide an approximation of the distribution.

could be investigated for specific classification tools by exploiting further information, for example, about the architecture, training regime and level of floating point accuracy.

Table 25
ImageNet — Accuracy and susceptibility of the networks to adversarial and random attacks, reported for 20,480 images from the validation set.

	ResNet50	VGG19
Accuracy	70.8%	66.52%
Adversarial attack susceptibility	94.2%	97.07%
Random attack susceptibility ($\delta = 10$)	2.5%	1.4%

CRediT authorship contribution statement

Oliver J. Sutton: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Qinghua Zhou:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Data curation. **Ivan Y. Tyukin:** Writing – review & editing, Writing – original draft, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Alexander N. Gorban:** Validation, Supervision, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Alexander Bastounis:** Conceptualization. **Desmond J. Higham:** Writing – review & editing, Writing – original draft, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Oliver J. Sutton reports financial support was provided by Engineering and Physical Sciences Research Council. Oliver J. Sutton reports financial support was provided by UK Research and Innovation. Qinghua Zhou reports financial support was provided by Engineering and Physical Sciences Research Council. Qinghua Zhou reports financial support was provided by UK Research and Innovation. Ivan Y. Tyukin reports financial support was provided by Engineering and Physical Sciences Research Council. Ivan Y. Tyukin reports financial support was provided by UK Research and Innovation. Alexander N. Gorban reports financial support was provided by Engineering and Physical Sciences Research Council. Alexander N. Gorban reports financial support was provided by UK Research and Innovation. Alexander Bastounis reports financial support was provided by Engineering and Physical Sciences Research Council. Desmond J. Higham reports financial support was provided by Engineering and Physical Sciences Research Council. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

We use openly available datasets.

Acknowledgements

O.J.S, Q.Z, A.N.G. and I.Y.T were supported in part by the UKRI, EPSRC [UKRI Turing AI Fellowship ARaISE EP/V025295/2 and UKRI Trustworthy Autonomous Systems Node in Verifiability EP/V026801/2]. D.J.H. and A.B. were supported by EPSRC grant EP/V046527/1.

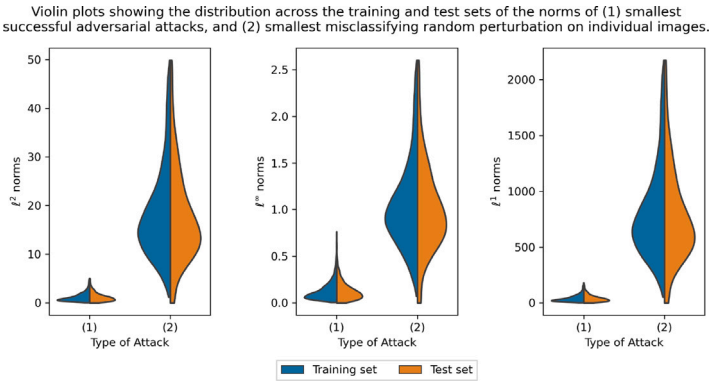


Fig. 14. CIFAR-10 — A direct comparison of the size distributions over all attackable images in the training and test sets of the smallest successful adversarial attack and smallest misclassifying random perturbation for the 'cat-vs-aeroplane' problem (0-vs-3), as measured in various norms. The plotted distributions were fitted to the data using a standard Kernel Density Estimation algorithm and therefore only provide an approximation of the true empirical distribution.

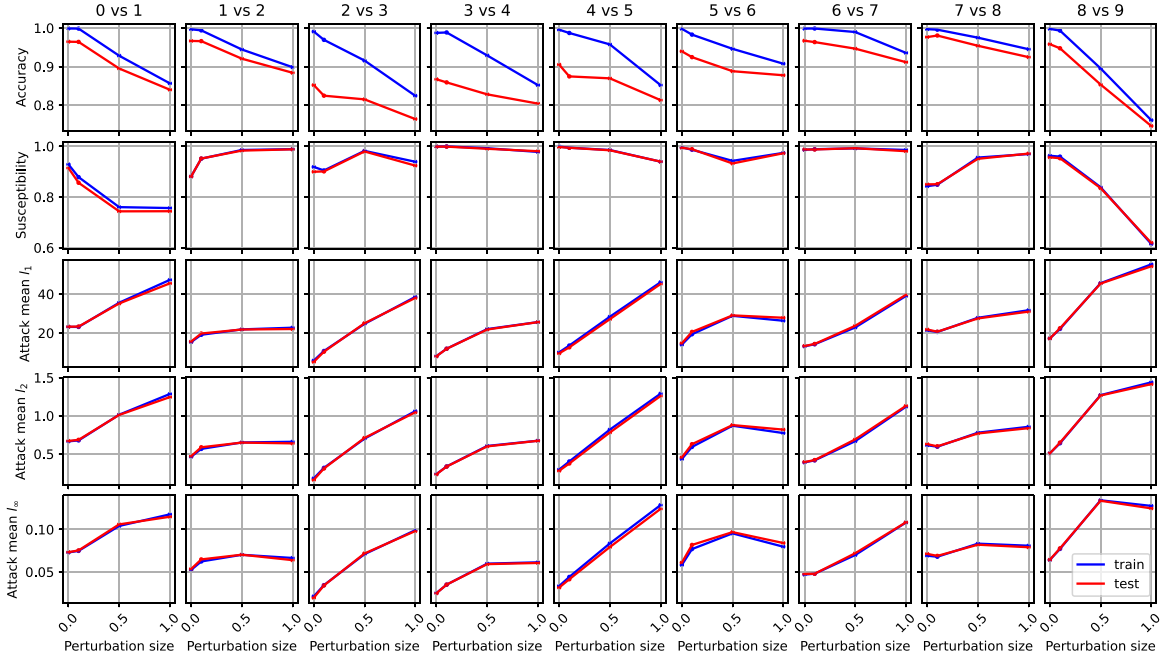


Fig. 15. CIFAR-10 — Plots showing how the performance of the network is affected by various magnitudes of random perturbations added to the images during training. This figure shows the results for random perturbations sampled from the cube $[-a,a]^n$ for $a \in \{0, 0.1, 0.5, 1\}$. This visualises the results in Tables 13, 15 and 17 compared to the previous data computed with no random perturbations (corresponding to $a = 0$). The data is plotted as separate lines for the training and test sets. 'Susceptibility' here refers to the adversarial susceptibility reported in the tables, and 'Attack mean ℓ_p' indicates the mean across each data set of the ℓ^p norm of the smallest adversarial perturbation affecting each image. The perturbation size plotted on the x axis is the size of a.

Appendix A. Proof of Theorem 1

Expanding the probability as an integral, using the fact that the density of the uniform distribution $\mathcal{U}(B^n_\delta(x))$ is just the reciprocal of the volume of a ball with radius δ, we have

$$P(z \sim \mathcal{U}(B^n_\delta(x)) \ : \ \text{sign}(z \cdot v) \neq \text{sign}(x \cdot v)) = \frac{1}{V^n \delta^n} \int_{B^n_\delta(x)} \mathbb{I}_{\{\text{sign}(z \cdot v) \neq \text{sign}(x \cdot v)\}} dz.$$

Since v is the fixed normal vector to the plane Π, the integral here is simply measuring the volume of a spherical cap. If we assume (without loss of generality) that $x \cdot v < 0$, then this cap may be expressed as the set

$$C = \{z \in \mathbb{R}^n \ : \ \|z\| \leq \delta \text{ and } z \cdot v \geq 0\}.$$

Since a spherical cap may be contained within a hemisphere of a different ball, we may prove the following bound:

Lemma 23 (*Spherical Cap Volume Bound*)**.** *Let n be a positive integer, and $r \geq h > 0$. Then,*

$$V^n_{\text{cap}}(r,h) \leq \frac{1}{2} V^n r^n \left(1 - \left(1 - \frac{h}{r}\right)^2\right)^{\frac{n}{2}}.$$

By assumption, the height of the cap C is $\delta - \epsilon$, and therefore

$$\int_C 1 dz \leq \frac{1}{2} V^n \delta^n \left(1 - \left(1 - \frac{\delta - \epsilon}{\delta}\right)^2\right)^{\frac{n}{2}},$$

and the result of the theorem follows.

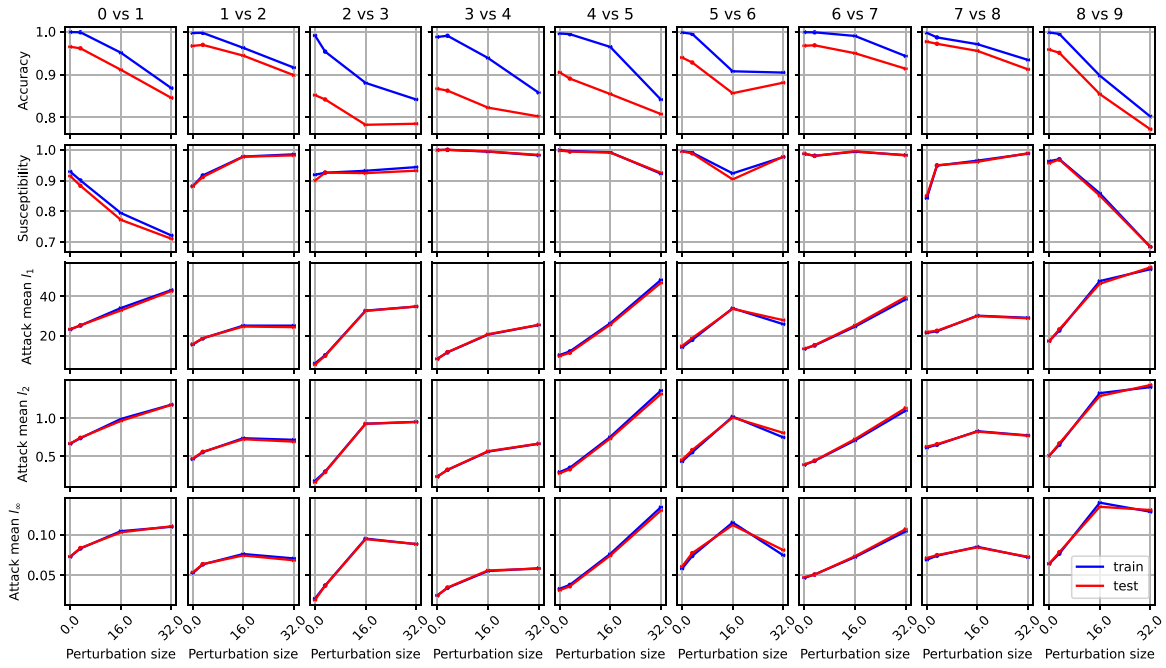


Fig. 16. CIFAR-10 — Plots showing how the performance of the network is affected by various magnitudes of random perturbations added to the images during training. This figure shows the results for random perturbations sampled from the ball $\mathbb{B}_b^n$ for $b \in \{0, 3.2, 16, 32\}$. This visualises the results in Tables 13, 15 and 17 compared to the previous data computed with no random perturbations (corresponding to $b = 0$). The data is plotted as separate lines for the training and test sets. 'Susceptibility' here refers to the adversarial susceptibility reported in the tables, and 'Attack mean ℓ_p' indicates the mean across each data set of the ℓ^p norm of the smallest adversarial perturbation affecting each image. The perturbation size plotted on the x axis is the size of a.

Appendix B. Proofs of results for the two balls model in Section 4.2

B.1. Proof of Theorem 4

Expanding the probability using the definition of the distribution $\mathcal{D}_\epsilon$, and the definition of the classifier f, we have

$$P((x,\ell) \sim \mathcal{D}_\epsilon \ : \ f(x) = \ell)$$
$$= \frac{1}{2}P(x \sim \mathcal{D}_0 \ : \ f(x) = 0) + \frac{1}{2}P(x \sim \mathcal{D}_1 \ : \ f(x) = 1)$$

The factor of $\frac{1}{2}$ is due to the fact that samples with either label are sampled with equal probability. Negating these two probabilities and expressing them as integrals using the densities p_0 and p_1 associated with $\mathcal{D}_0$ and $\mathcal{D}_1$ respectively (the existence of these densities is a requirement of the SmAC property), we have

$$P((x,\ell) \sim \mathcal{D}_\epsilon \ : \ f(x) = \ell)$$
$$= 1 - \frac{1}{2}\int_{D_0} \mathbb{I}_{\{x_1>0\}} p_0(x)dx - \frac{1}{2}\int_{D_1} \mathbb{I}_{\{x_1<0\}} p_1(x)dx.$$

The bound on the density p provided by the SmAC property in Definition 3 (recalling that $r = 1$ for both distributions) therefore implies that

$$P((x,\ell) \sim \mathcal{D}_\epsilon \ : \ f(x) = \ell)$$
$$\geq 1 - \frac{A}{2V^n}\left(\int_{D_0} \mathbb{I}_{\{x_1>0\}} dx + \int_{D_1} \mathbb{I}_{\{x_1<0\}} dx\right).$$

By symmetry, the two integrals have the same value, so we only compute the first. Since $\epsilon > 0$, this corresponds to the volume of a section of a ball which is smaller than a hemisphere, we may write it as

$$\int_{D_0} \mathbb{I}_{\{x_1>0\}} dx = V^n_{\text{cap}}(1, 1-\epsilon).$$

Lemma 23 implies that

$$V^n_{\text{cap}}(1, 1-\epsilon) \leq \frac{1}{2}V^n(1-\epsilon^2)^{\frac{n}{2}},$$

and the result therefore follows.

B.2. Proof of Theorem 5

Using the definition of the classification function f and conditioning on the class label, we may rewrite the probability in question as

$$P\big((x,\ell) \sim \mathcal{D}_\epsilon \ : \ \text{there exists } s \in \mathbb{B}^n_\delta \text{ such that } f(x+s) \neq \ell\big)$$
$$= \frac{1}{2}P(x \sim \mathcal{U}(D_0) \ : \ x_1 > -\delta) + \frac{1}{2}P(x \sim \mathcal{U}(D_1) \ : \ x_1 < \delta).$$

Symmetry implies that these two probabilities have the same value, so we only need to compute the first one. Negating the probability and expanding it as an integral using the density p_0 of $\mathcal{D}_0$, we find that

$$P(x \sim \mathcal{U}(D_0) \ : \ x_1 > -\delta) = 1 - \int_{D_0} \mathbb{I}_{\{x_1<-\delta\}} p_0(x)dx.$$

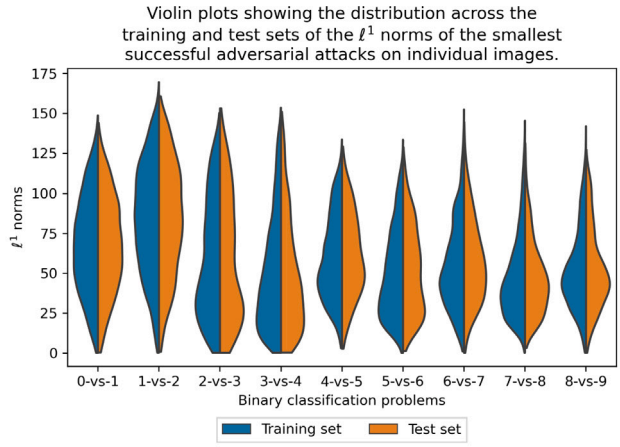
The bound on the density provided by the SmAC property (Definition 3) therefore implies that

$$P(x \sim \mathcal{U}(D_0) \ : \ x_1 > -\delta) \geq 1 - \frac{A}{V^n}\int_{D_0} \mathbb{I}_{\{x_1<-\delta\}} dx.$$

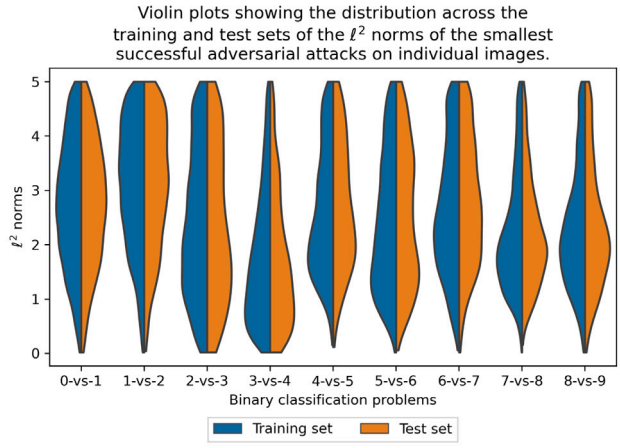
Since this integral corresponds to the volume of a spherical cap with height smaller than its radius (due to the fact that $1 > \delta > \epsilon > 0$), we may apply Lemma 23 to show that

$$\int_{D_0} \mathbb{I}_{\{x_1<-\delta\}} dx = V^n_{\text{cap}}(1, 1-(\epsilon-\delta))$$
$$\leq \frac{1}{2}V^n(1-(\delta-\epsilon)^2)^{\frac{n}{2}},$$

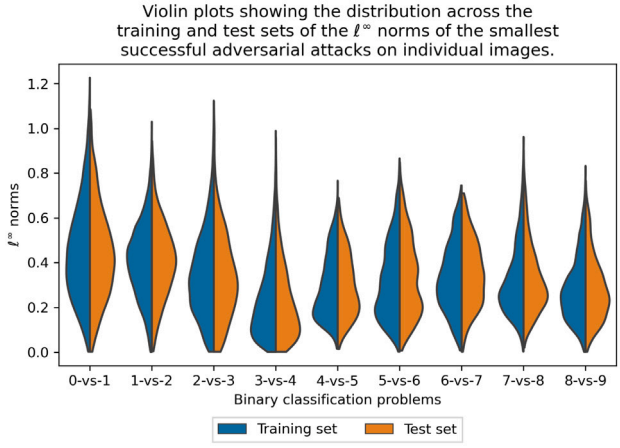
and the result therefore follows.



(a) ℓ^1 norms

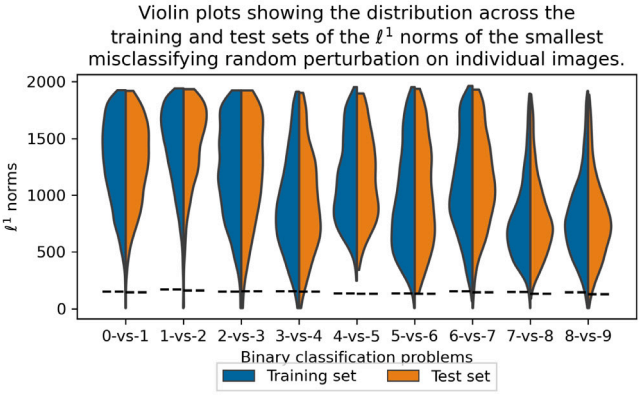


(b) ℓ^2 norms

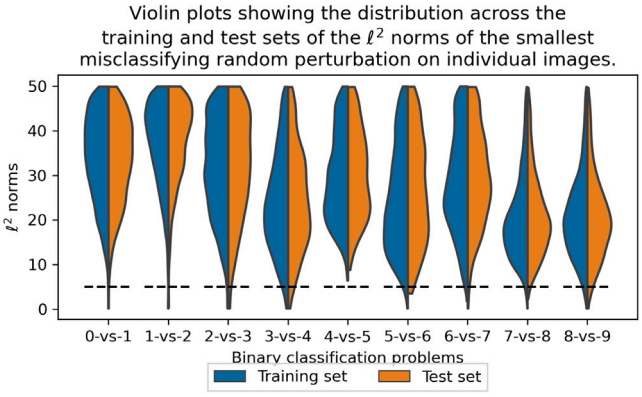


(c) ℓ^∞ norms

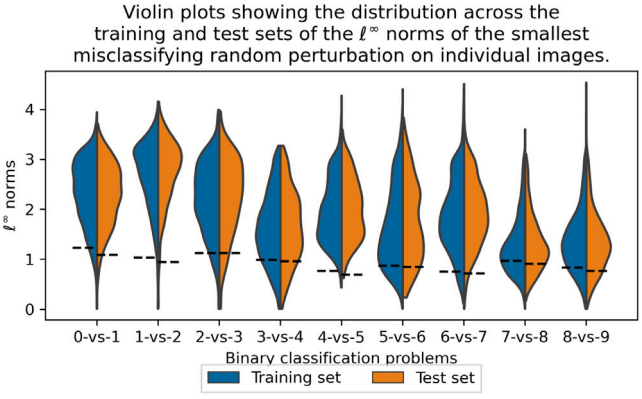
Fig. 17. Fashion MNIST — Distribution of norms of smallest successful adversarial attacks on each image.



(a) ℓ^1 norms



(b) ℓ^2 norms



(c) ℓ^∞ norms

Fig. 18. Fashion MNIST — Distribution of norms of the smallest misclassifying random perturbations found on each image. Black dashed lines indicate the size of the largest adversarial attack required on each data set.

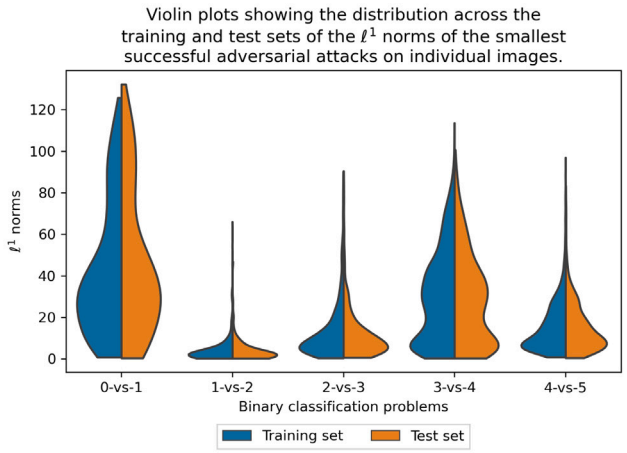
B.3. Proof of Theorem 6

To prove Theorem 6, we begin by expanding the probability by conditioning on the label value, finding

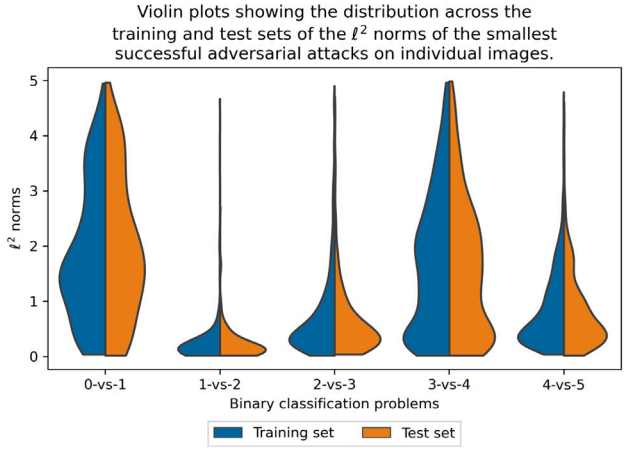
$$P((x,\ell) \sim D_\epsilon, s \sim \mathcal{U}(\mathbb{B}_\delta^n) \ : \ f(x+s) \neq \ell)$$

$$= \frac{1}{2} P(x \sim \mathcal{U}(D_0), s \sim \mathcal{U}(\mathbb{B}_\delta^n) \ : \ x_1 + s_1 > 0) + \frac{1}{2} P(x \sim \mathcal{U}(D_1), s \sim \mathcal{U}(\mathbb{B}_\delta^n) \ : \ x_1 + s_1 < 0).$$

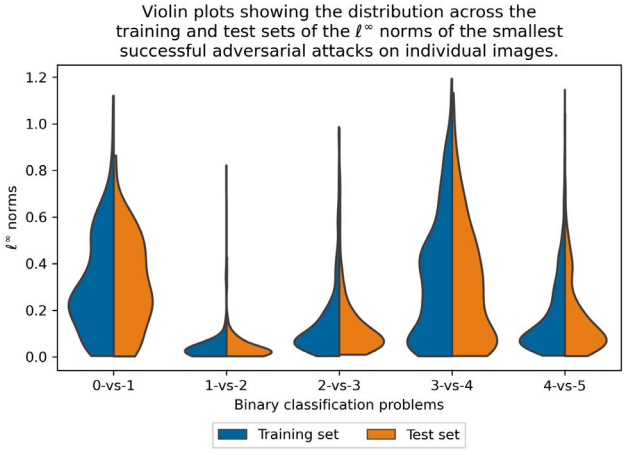
The symmetry here implies that these two probabilities are equal, so we proceed by only bounding the first. Expanding this as an integral



(a) ℓ^1 norms

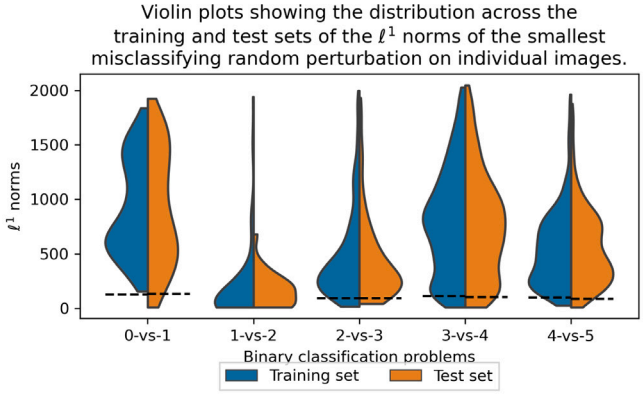


(b) ℓ^2 norms

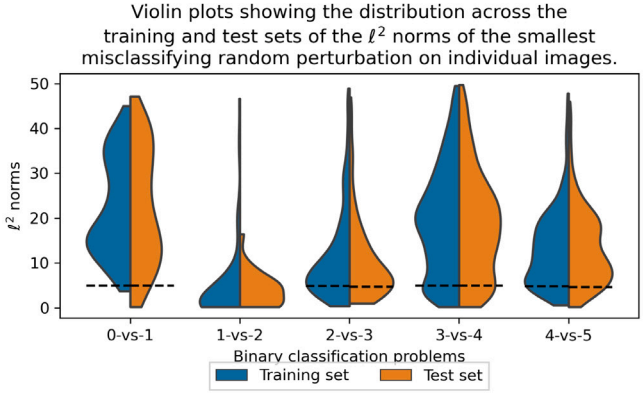


(c) ℓ^∞ norms

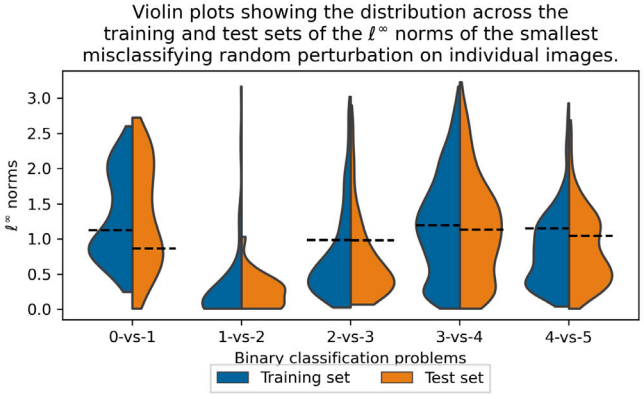
Fig. 19. GTSRB — Distribution of norms of smallest successful adversarial attacks on each image.



(a) ℓ^1 norms



(b) ℓ^2 norms



(c) ℓ^∞ norms

Fig. 20. GTSRB — Distribution of norms of the smallest misclassifying random perturbations found on each image. Black dashed lines indicate the size of the largest adversarial attack required on each data set.

using the density p_0 of D_0 and the fact that s is sampled from a uniform distribution, we observe that

$$P(x \sim \mathcal{U}(D_0), s \sim \mathcal{U}(\mathbb{B}_\delta^n) \ : \ x_1 + s_1 > 0)$$

$$= \frac{1}{V^n \delta^n} \int_{D_0} \int_{\mathbb{B}_\delta^n} \mathbb{I}_{\{s_1 > -x_1\}} ds \, p_0(x) \, dx.$$

The bound on the density provided by the SmAC property (Definition 3) therefore implies that

$$P(x \sim \mathcal{U}(D_0), s \sim \mathcal{U}(\mathbb{B}_\delta^n) \ : \ x_1 + s_1 > 0)$$

$$\leq \frac{A}{(V^n)^2 \delta^n} \int_{D_0} \int_{\mathbb{B}_\delta^n} \mathbb{I}_{\{s_1 > -x_1\}} ds dx.$$

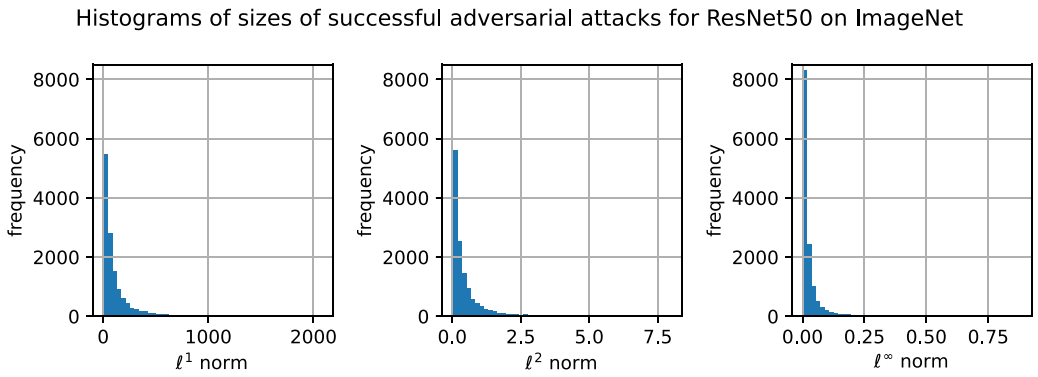


Fig. 21. ImageNet — Histograms showing the distribution of sizes of successful adversarial attacks on images from the ImageNet validation set for a pre-trained ResNet50 model.

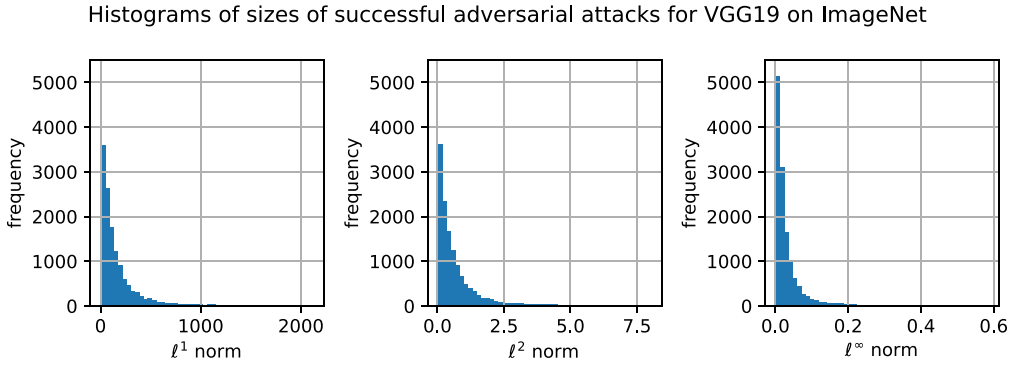


Fig. 22. ImageNet — Histograms showing the distribution of sizes of successful adversarial attacks on images from the ImageNet validation set for a pre-trained VGG19 model.

For each fixed value of x, the inner integral is just computing the volume of a section of a ball with radius δ. When $x_1 > 0$, this volume is at least a hemisphere, while when $x_1 < 0$ this volume is less than a hemisphere. On the other hand, for fixed s, the integral over x is also calculating the volume of a section of the unit ball. Since the volume of the ball concentrates in high dimensions about its equator, a section which is smaller than a hemisphere may be expected to have small volume, while a section larger than a hemisphere may be expected to have large volume. This is the intuition we apply here by splitting the integral into two parts: one for $x_1 < -t$ and one for $x_1 \geq -t$ for some arbitrary $t \in [0, \epsilon]$. In the first part, we will be able to obtain 'smallness' in our bound from the fact that we are integrating s over just a spherical cap, while in the second case we are integrating x over a spherical cap. We write this splitting as

$$\begin{aligned}\int_{D_0}\int_{\mathbb{B}^n_\delta} \mathbb{I}_{\{s_1>-x_1\}}\,ds\,dx \qquad (B.1)\\ = \int_{D_0}\int_{\mathbb{B}^n_\delta} \mathbb{I}_{\{x_1<-t\}}\mathbb{I}_{\{s_1>-x_1\}}\,ds\,dx\\ + \int_{D_0}\int_{\mathbb{B}^n_\delta} \mathbb{I}_{\{x_1>-t\}}\mathbb{I}_{\{s_1>-x_1\}}\,ds\,dx.\end{aligned}$$

The first term of this splitting may be bounded above by extending the indicator function over s to all those points with $s_1 > t$ (since $-x_1 > t$), which enables us to separate the integrals to find that

$$\begin{aligned}&\int_{D_0}\int_{\mathbb{B}^n_\delta} \mathbb{I}_{\{x_1<-t\}}\mathbb{I}_{\{s_1>-x_1\}}\,ds\,dx\\ &\leq \int_{D_0} \mathbb{I}_{\{x_1<-t\}}\,dx \int_{\mathbb{B}^n_\delta} \mathbb{I}_{\{s_1>t\}}\,ds.\end{aligned}$$

These integrals may be expressed as volumes in the form

$$\begin{aligned}&\int_{D_0} \mathbb{I}_{\{x_1<-t\}}\,dx \int_{\mathbb{B}^n_\delta} \mathbb{I}_{\{s_1>t\}}\,ds\\ &= (V^n - V^n_{\text{cap}}(1, 1-(\epsilon-t)))V^n_{\text{cap}}(\delta, \delta - t),\end{aligned}$$

and Lemma 23 and the fact that the volume of a spherical cap is non-negative implies that

$$\int_{D_0} \mathbb{I}_{\{x_1<-t\}}\,dx \int_{\mathbb{B}^n_\delta} \mathbb{I}_{\{s_1>t\}}\,ds \leq \frac{1}{2}(V^n)^2\delta^n\left(1-\left(\frac{t}{\delta}\right)^2\right)^{\frac{n}{2}}.$$

Returning to the second integral in (B.1), we may similarly bound the indicator function over s from above by simply the constant 1. This implies that

$$\begin{aligned}\int_{D_0}\int_{\mathbb{B}^n_\delta} \mathbb{I}_{\{x_1>-t\}}\mathbb{I}_{\{s_1>-x_1\}}\,ds\,dx &\leq \int_{D_0} \mathbb{I}_{\{x_1>-t\}}\,dx \int_{\mathbb{B}^n_\delta} ds\\ &= V^n_{\text{cap}}(1, 1-(\epsilon-t))V^n\delta^n,\end{aligned}$$

and Lemma 23 consequently provides

$$\begin{aligned}&\int_{D_0}\int_{\mathbb{B}^n_\delta} \mathbb{I}_{\{x_1>-t\}}\mathbb{I}_{\{s_1>-x_1\}}\,ds\,dx\\ &\leq \frac{1}{2}(V^n)^2\delta^n(1-(\epsilon-t)^2)^{\frac{n}{2}}.\end{aligned}$$

Combining these bounds and using the fact that $t \in [0, \epsilon]$ was arbitrary, we find that

$$\begin{aligned}&P((x,\ell) \sim \mathcal{D}_\epsilon, s \sim \mathcal{U}(\mathbb{B}^n_\delta) \,:\, f(x+s) \neq \ell)\\ &\leq \frac{1}{2}A \inf_{t\in[0,\epsilon]}\left[\left(1-\left(\frac{t}{\delta}\right)^2\right)^{\frac{n}{2}} + (1-(\epsilon-t)^2)^{\frac{n}{2}}\right],\end{aligned}$$

and the theorem follows by noting that, for $t = \frac{\epsilon\delta}{1+\delta}$, the two terms inside the infimum are equal (this choice of t is valid since $\frac{\delta}{1+\delta} \in [0,1]$ for $\delta \geq 0$).

B.4. Proof of Theorem 7

When $\ell = 0$, we have $|\tilde{f}(x) - \ell| = \sigma(g(x))$, since $\sigma(t) \in (0,1)$ for $t \in \mathbb{R}$. In this case, the attack may therefore be computed as

$$\mathbf{e}_1 \sigma'(g(x)) L'(\sigma(g(x))), \tag{B.2}$$

since $g'(x) = \mathbf{e}_1$. Since σ and L are assumed to be continuously differentiable and monotonically increasing, the Morse-Sard theorem (Morse, 1939) implies that $\sigma'(t), L'(t) > 0$ everywhere except on a set of Lebesgue measure zero. The SmAC property on the distribution $\mathcal{D}_\epsilon$ implies that the probability of sampling x from a set of Lebesgue measure zero is zero, and therefore with probability 1 the attack direction (B.2) is a positive multiple of $\mathbf{e}_1$ for all $t \in \mathbb{R}$, as required. Analogously, when $\ell = 1$ we have $|\tilde{f}(x) - \ell| = 1 - \sigma(g(x))$, and we therefore obtain a negative multiple of $\mathbf{e}_1$ with probability 1, and the result follows.

B.5. Proof of Theorem 8

Since the statement and setup are symmetric with respect to the class label ℓ, we focus only on the class 0 and the statement for class 1 follows analogously. In this case, the statement that

$$f(x+s) \neq \ell \text{ for all } s \in \mathbb{R}^n \text{ such that } d_\ell(z,s) > \gamma,$$

is implied by the condition that $z_1 < x_1 + \gamma$, and therefore

$$\begin{aligned} P(x, z \sim \mathcal{U}(D_0) \,:\, f(x+s) \neq 0 \text{ for all } s \in \mathbb{R}^n \\ \text{such that } d_0(z,s) > \gamma) \\ \geq P(x, z \sim \mathcal{U}(D_0) \,:\, z_1 < x_1 + \gamma). \end{aligned}$$

Introducing $t \in [0, \gamma]$, this probability is clearly at least

$$P(x, z \sim \mathcal{U}(D_0) \,:\, z_1 < x_1 + \gamma \text{ and } x_1 > -\epsilon - t).$$

We note that if $x_1 > -\frac{1}{2}\gamma - \epsilon$ and $z_1 < \frac{1}{2}\gamma - \epsilon$ then it follows that $z_1 < x_1 + \gamma$, and therefore

$$\begin{aligned} P(x, z \sim \mathcal{U}(D_0) \,:\, z_1 < x_1 + \gamma) \geq \\ P(x, z \sim \mathcal{U}(D_0) \,:\, z_1 < \tfrac{1}{2}\gamma - \epsilon \text{ and } x_1 > -\tfrac{1}{2}\gamma - \epsilon). \end{aligned}$$

This last probability has the property of involving two events which separately depend on the independent variables x and z, and may therefore be expressed as the product

$$P(z \sim \mathcal{U}(D_0) \,:\, z_1 < \tfrac{1}{2}\gamma - \epsilon) P(x \sim \mathcal{U}(D_0) \,:\, x_1 > -\tfrac{1}{2}\gamma - \epsilon).$$

Negating both of these probabilities and recalling the bound on the density provided by Definition 3 with $r = 1$, we obtain the lower bounds

$$P(z \sim \mathcal{U}(D_0) \,:\, z_1 < \tfrac{1}{2}\gamma - \epsilon) \geq 1 - \frac{A}{V^n} \int_{D_0} \mathbb{I}_{\{z_1 < \frac{1}{2}\gamma - \epsilon\}} dz,$$

and

$$P(x \sim \mathcal{U}(D_0) \,:\, x_1 > -\epsilon - \tfrac{1}{2}\gamma) \geq 1 - \frac{A}{V^n} \int_{D_0} \mathbb{I}_{\{x_1 < -\epsilon - \frac{1}{2}\gamma\}} dz.$$

Since D_0 is simply a ball for which the centre has first coordinate $-\epsilon$, it follows that both of these integrals are simply the volume of a spherical cap, with value $V^n_{\text{cap}}(1, 1 - \frac{1}{2}\gamma)$, and using Lemma 23, we therefore conclude that

$$P(x, z \sim \mathcal{U}(D_0) \,:\, z_1 < x_1 + \gamma) \geq \left(1 - A\left(1 - \frac{\gamma^2}{4}\right)^{\frac{n}{2}}\right)^2$$

Appendix C. Proofs of results for the two half balls model in Section 5

C.1. Proof of Theorem 18

Using the definition of the classification function f, we may rewrite the probability in question as

$$\begin{aligned} P\big(x \sim \mathcal{D}_\epsilon \,:\, \text{there exists } s \in \mathbb{R}^n \text{ with } \|s\| \leq \delta \\ \text{such that } f(x+s) \neq f(x)\big) = P(x \sim \mathcal{D}_\epsilon \,:\, |x_1| < \delta). \end{aligned}$$

Expanding the probability as an integral, and using the fact that $\mathcal{D}_\epsilon$ is a uniform distribution over two disjoint half-balls and therefore has density $(V^n)^{-1}$, we may further express this as

$$\begin{aligned} P(x \sim \mathcal{D}_\epsilon \,:\, |x_1| < \delta) = \frac{1}{V^n}\Big(\int_{D_0} \mathbb{I}_{\{x \,:\, -\delta < x_1 < -\epsilon\}} dx \\ + \int_{D_1} \mathbb{I}_{\{x \,:\, \epsilon < x_1 < \delta\}} dx \Big), \end{aligned}$$

and the remaining problem is to compute the two remaining integrals, the values of which are equal by symmetry. We may express the set $\{x \in D_1 \,:\, \epsilon < x_1 < \delta\}$, which geometrically represents the slab of the half-ball D_1 within $\delta - \epsilon$ distance of its planar face, as the complement of a spherical cap, implying

$$\int_{D_1} \mathbb{I}_{\{x \,:\, \epsilon < x_1 < \delta\}} dx = \frac{1}{2} V^n - V^n_{\text{cap}}(1, 1 - (\delta - \epsilon)),$$

and therefore

$$P(x \sim \mathcal{D}_\epsilon \,:\, |x_1| < \delta) = 1 - \frac{2V^n_{\text{cap}}(1, 1 - (\delta - \epsilon))}{V^n}.$$

We may further estimate this from below, to show the exponential behaviour of this quantity with respect to n, by enveloping the spherical cap within a small half-ball. The Pythagorean theorem implies that

$$\{x \in D_1 \,:\, \epsilon < x_1 < \delta\} \subset G$$

where

$$G = \{x \in \mathbb{R}^n \,:\, x_1 > \epsilon \text{ and } \|x - \epsilon\, \mathbf{e}_1\|^2 \leq 1 - (\delta - \epsilon)^2\},$$

and therefore

$$\frac{2V^n_{\text{cap}}(1, 1 - (\delta - \epsilon))}{V^n} \leq (1 - (\delta - \epsilon)^2)^{n/2},$$

which proves the theorem.

C.2. Proof of Theorem 19

Since D_0 and D_1 are disjoint half-balls of a unit ball, it follows that the density associated with $\mathcal{D}_\epsilon$ is simply $(V^n)^{-1}$, while the density associated with $\mathcal{U}(\mathbb{B}^n_\delta)$ is $(\delta^n V^n)^{-1}$. Writing the probability as an integral with this density, we therefore find that

$$\begin{aligned} P\big(x \sim \mathcal{D}_\epsilon, s \sim \mathcal{U}(\mathbb{B}^n_\delta) \,:\, f(x+s) \neq f(x)\big) \\ = \frac{1}{\delta^n (V^n)^2}\Big(\int_{D_0} \int_{B^n_\delta} \mathbb{I}_{\{x,s \,:\, s_1 > -x_1\}} ds dx \\ + \int_{D_1} \int_{B^n_\delta} \mathbb{I}_{\{x,s \,:\, s_1 < -x_1\}} ds dx \Big). \end{aligned}$$

Since the values of these two integrals are equal by symmetry, we proceed by only estimating the first. For $x \in D_0$, we have $x_1 < -\epsilon$, and therefore

$$\begin{aligned} \int_{D_0} \int_{B^n_\delta} \mathbb{I}_{\{x,s \,:\, s_1 > -x_1\}} ds dx \leq \int_{D_0} dx \int_{B^n_\delta} \mathbb{I}_{\{s \,:\, s_1 > \epsilon\}} ds \\ = \frac{1}{2} V^n \int_{B^n_\delta} \mathbb{I}_{\{s \,:\, s_1 > \epsilon\}} ds. \end{aligned}$$

The remaining integral over s now takes the form of the volume of a spherical cap, which we may bound by enveloping the cap in a small half-ball. Arguing as in the proof of Theorem 18, it follows that

$$\int_{\mathcal{B}^n_\delta} \mathbb{I}_{\{s\,:\,s_1>\epsilon\}}\, ds = V^n_{\text{cap}}(\delta, (\delta^2-\epsilon^2)^{1/2}) \leq \frac{1}{2}\delta^n V^n\left(1-\left(\frac{\epsilon}{\delta}\right)^2\right)^{n/2}.$$

The result therefore follows by combining the components above.

C.3. Proof of Theorem 20

Consider the hyperplane h passing through the origin and whose normal is $\mathbf{e}_1$. This hyperplane is the decision boundary of the classifier f (see (5)) which separates D_0 and D_1 in the sense that, with probability one, the classifier assigns correct labels to samples drawn from $\mathcal{D}_\epsilon$, $\epsilon = 0$.

Pick any $\Delta \in (0,1)$. The probability p that a point $x \sim \mathcal{D}_\epsilon$ lands within the Δ-distance from the hyperplane h is bounded from below as:

$$p > 1-(1-\Delta^2)^{\frac{n}{2}}.$$

Conversely, if one picks the value of $p \in (0,1)$, then the value of Δ corresponding to this probability must satisfy:

$$\Delta < (1-(1-p)^{\frac{2}{n}})^{\frac{1}{2}} = \rho(p,n).$$

In what follows, we are interested in the event

$$E_1(x,s,\delta,n):\ \ f(x+s)\neq f(x),\ \ s\sim\mathcal{U}(\mathbb{B}^n_\delta).$$

It is clear that the event

$$E_2(x,s,\delta,\Delta,n):\ \ f(x+s)\neq f(x),\ \ \ |x\cdot\mathbf{e}_1|\leq\Delta,\ \ s\sim\mathcal{U}(\mathbb{B}^n_\delta)$$

implies event $E_1(x,s,\delta,n)$. Hence

$$\begin{aligned} &P(f(x+s)\neq f(x),\ \ s\sim\mathcal{U}(\mathbb{B}^n_\delta))\geq\\ &P(f(x+s)\neq f(x),\ \ s\sim\mathcal{U}(\mathbb{B}^n_\delta) \text{ and } |x\cdot\mathbf{e}_1|\leq\Delta)\\ &= P(f(x+s)\neq f(x),\ \ s\sim\mathcal{U}(\mathbb{B}^n_\delta)\ \mid\ |x\cdot\mathbf{e}_1|\leq\Delta)p, \end{aligned} \tag{C.1}$$

where the last equality follows from the definition of the conditional probability and the fact that $p = P(|x\cdot\mathbf{e}_1|\leq\Delta)$ is the probability of $x\sim\mathcal{D}_\epsilon$ landing within the Δ-distance from the hyperplane h.

Consider the event

$$\begin{aligned} &E_3(x,s,\delta,p,n):\\ &\left[0\leq x\cdot\mathbf{e}_1\leq\Delta \text{ and } s\cdot\mathbf{e}_1\leq-\rho(p,n)\right] \text{ or}\\ &\qquad\left[-\Delta\leq x\cdot\mathbf{e}_1<0 \text{ and } s\cdot\mathbf{e}_1\geq\rho(p,n)\right]. \end{aligned}$$

Given that $\Delta<\rho(p,n)$, the event $E_3(x,s,\delta,p,n)$ implies $E_2(x,s,\delta,\Delta,n)$. Hence taking (C.1) into account, the following holds true:

$$P(E_1(x,s,\delta,n))\geq P(E_2(x,s,\delta,\Delta,n))\geq P(E_3(x,s,\delta,p,n)).$$

Therefore, a lower bound for $P(E_3(x,s,\delta,p,n))$ is also a lower bound for $P(E_1(x,s,\delta,n))$.

Noticing that x and s are independent, we obtain

$$\begin{aligned} P(E_3(x,s,\delta,p,n)) =& P(0\leq x\cdot\mathbf{e}_1\leq\Delta)P(s\cdot\mathbf{e}_1\leq-\rho(p,n))+\\ & P(-\Delta\leq x\cdot\mathbf{e}_1<0)P(s\cdot\mathbf{e}_1\geq\rho(p,n))\\ =&\frac{p}{2}P(s\cdot\mathbf{e}_1\leq-\rho(p,n))+\frac{p}{2}P(s\cdot\mathbf{e}_1\geq\rho(p,n)). \end{aligned}$$

Observe that the symmetry of $\mathcal{U}(\mathbb{B}^n_\delta)$ implies $P(s\cdot\mathbf{e}_1\geq\rho(p,n)) = P(s\cdot\mathbf{e}_1\leq-\rho(p,n))$, and hence

$$P(E_3(x,s,\delta,p,n)) = pP(s\cdot\mathbf{e}_1\geq\rho(p,n)).$$

Let us now bound the probability of $E_3(x,s,\delta,p,n)$ from below.

Case 1: $0<\delta\leq\rho(p,n)$. In this case

$$f(x+s)=f(x) \text{ for all } s\in\mathbb{B}^n_\delta,$$

and hence $P(E_3(x,s,\delta,p,n))=0$.

Case 2: $\delta>\rho(p,n)$. The probability of $P(s\cdot\mathbf{e}_1\geq\rho(p,n))$ is the ratio

$$\frac{V^n_{\text{cap}}(\delta,\delta-\rho(p,n))}{V^n_{\text{cap}}(\delta,\delta)},$$

where $V^n_{\text{cap}}(\delta,\delta-\rho(p,n))$ is the volume of the spherical cap whose radius is δ and whose height is $\delta-\rho(p,n)$.

Consider $V^n_{\text{cap}}(\delta,\delta-\rho(p,n))$ (Li, 2010):

$$V^n_{\text{cap}}(\delta,\delta-\rho(p,n)) = \frac{\pi^{(n-1)/2}}{\Gamma(\frac{n-1}{2}+1)}\delta^n\int_0^{\cos^{-1}(\rho(p,n)/\delta)}\sin^n(\theta)d\theta.$$

Rewriting the integral through the change of variables $t=\cos(\theta)$ results in

$$V^n_{\text{cap}}(\delta,\delta-\rho(p,n)) = \frac{\pi^{(n-1)/2}}{\Gamma(\frac{n-1}{2}+1)}\delta^n\int_{\rho(p,n)/\delta}^1(1-t^2)^{\frac{n-1}{2}}dt,$$

and hence

$$P(s\cdot\mathbf{e}_1\geq\rho(p,n)) = \frac{\int_{\rho(p,n)/\delta}^1(1-t^2)^{\frac{n-1}{2}}dt}{\int_0^1(1-t^2)^{\frac{n-1}{2}}dt}. \tag{C.2}$$

Let us now bound the integral

$$\int_{\rho(p,n)/\delta}^1(1-t^2)^{\frac{n-1}{2}}dt \tag{C.3}$$

from below. First, observe that

$$(1-t^2)\geq(1-t^2\alpha+\frac{t^4\alpha^2}{2}) \tag{C.4}$$

for any $\alpha>1$ and

$$0<t\leq\frac{\sqrt{2(\alpha-1)}}{\alpha}.$$

At the same time, using Taylor's theorem we see that if $t,\alpha>0$ there exists a $c\in(0,t\alpha^2)$ so that

$$\begin{aligned} &e^{-\alpha t^2} = 1-t^2\alpha+\frac{t^4\alpha^2}{2}-\frac{e^{-c}t^6\alpha^3}{3!}<1-t^2\alpha+\frac{t^4\alpha^2}{2}\\ &1-t^2\alpha+\frac{t^4\alpha^2}{2}>e^{-\alpha t^2}. \end{aligned} \tag{C.5}$$

Applying the same argument, one can conclude that for any $p\in(0,1)$ and all $n\geq1$ the following holds true:

$$(1-p)^{\frac{2}{n}} = e^{\log(1-p)\frac{2}{n}} > 1+\log(1-p)\frac{2}{n}.$$

This implies that

$$\rho(p,n)<\frac{\sqrt{2|\log(1-p)|}}{\sqrt{n}}$$

for all $n\geq1$.

Let

$$\tau(\alpha)=\frac{\sqrt{2(\alpha-1)}}{\alpha},$$

N be defined by

$$N(\alpha,p,\delta):=\max\left\{1,\frac{2|\log(1-p)|}{\delta^2\tau(\alpha)^2}\right\},$$

and

$$\beta(p,\delta,n):=\frac{\sqrt{2|\log(1-p)|}}{\sqrt{n}\delta}.$$

Suppose that $n>N(\alpha,p,\delta)$. Then we must have

$$\frac{\rho(p,n)}{\delta}<\beta(p,\delta,n)<\tau(\alpha).$$

In particular, for $n>N(\alpha,p,\delta)$, the integral (C.3) can be bounded from below as

$$\int_{\rho(p,n)/\delta}^1(1-t^2)^{\frac{n-1}{2}}dt>\int_{\beta(p,\delta,n)}^{\tau(\alpha)}(1-t^2)^{\frac{n-1}{2}}dt \tag{C.6}$$

Pick an

$$n > N(\alpha, p, \delta),$$

and consider the right-hand-side of (C.2). According to (C.6)

$$P(s \cdot \mathbf{e}_1 \geq \rho(p,n)) > \frac{\int_{\beta(p,\delta,n)}^{\tau(\alpha)} (1-t^2)^{\frac{n-1}{2}} dt}{\int_0^1 (1-t^2)^{\frac{n-1}{2}} dt}.$$

Invoking (C.4) and (C.5) we arrive at

$$P(s \cdot \mathbf{e}_1 \geq \rho(p,n)) > \frac{\int_{\beta(p,\delta,n)}^{\tau(\alpha)} e^{-\alpha \frac{t^2(n-1)}{2}} dt}{\int_0^1 (1-t^2)^{\frac{n-1}{2}} dt}.$$

Changing the integration variable as $t\sqrt{n-1} = \xi$ yields:

$$P(s \cdot \mathbf{e}_1 \geq \rho(p,n)) > \frac{\int_{\beta(p,\delta,n)\sqrt{n-1}}^{\tau(\alpha)\sqrt{n-1}} e^{-\alpha \frac{\xi^2}{2}} d\xi}{\int_0^{\sqrt{n-1}} \left(1 - \frac{\xi^2}{n-1}\right)^{\frac{n-1}{2}} d\xi}.$$

Note that (cf. Gorban, Tyukin, Prokhorov, and Sofeikov (2016), p. 135, inequality (23))

$$\left(1 - \frac{\xi^2}{(n-1)}\right)^{n-1} \leq e^{-\xi^2}$$

for all $n > 1$, $n \in \mathbb{N}$ and any $\xi^2 \geq 0$.

Therefore

$$P(s \cdot \mathbf{e}_1 \geq \rho(p,n)) > \frac{\int_{\beta(p,\delta,n)\sqrt{n-1}}^{\tau(\alpha)\sqrt{n-1}} e^{-\alpha \frac{\xi^2}{2}} d\xi}{\int_0^{\sqrt{n-1}} e^{-\frac{\xi^2}{2}} d\xi}.$$

for all $n > N(\alpha, p, \delta)$. Changing the integration variable yet again in the top integral as $\zeta = \sqrt{\alpha}\xi$ and pre-multiplying both the nominator and the denominator by $1/\sqrt{2\pi}$ results in:

$$P(s \cdot \mathbf{e}_1 \geq \rho(p,n)) > \frac{\frac{1}{\sqrt{\alpha}} \frac{1}{\sqrt{2\pi}} \int_{\sqrt{\alpha}\beta(p,\delta,n)\sqrt{n-1}}^{\sqrt{\alpha}\tau(\alpha)\sqrt{n-1}} e^{-\frac{\zeta^2}{2}} d\zeta}{\frac{1}{\sqrt{2\pi}} \int_0^{\sqrt{n-1}} e^{-\frac{\xi^2}{2}} d\xi}. \tag{C.7}$$

Recalling the standard cumulative distribution function

$$\Phi(s) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{s} e^{-\frac{\xi^2}{2}} d\xi,$$

the right-hand side of (C.7) becomes

$$\frac{\frac{1}{\sqrt{\alpha}} \left(\Phi\left(\sqrt{\alpha}\tau(\alpha)\sqrt{n-1}\right) - \Phi\left(\sqrt{\alpha}\beta(p,\delta,n)\sqrt{n-1}\right)\right)}{\Phi(\sqrt{n-1}) - \frac{1}{2}}.$$

Therefore, for any $p \in (0,1)$, $\delta > 0$, $\alpha > 1$, and $n > N(\alpha, p, \delta)$

$$P(E_3(x,s,\delta,p,n)) > \frac{p\left(\Phi\left(\sqrt{\alpha}\tau(\alpha)\sqrt{n-1}\right) - \Phi\left(\sqrt{\alpha}\beta(p,\delta,n)\sqrt{n-1}\right)\right)}{\sqrt{\alpha}\left(\Phi(\sqrt{n-1}) - \frac{1}{2}\right)}.$$

For any fixed $\alpha > 1$ and $n \to \infty$, the right-hand side of the above expression reduces to

$$\frac{p\frac{1}{\sqrt{\alpha}}\left(1 - \Phi\left(\sqrt{\alpha}\frac{\sqrt{2|\log(1-p)|}}{\delta}\right)\right)}{\frac{1}{2}}.$$

Given that the value of α can be chosen arbitrarily in $(1, \infty)$, in the limit

$$\lim_{n\to\infty} P(E_3(x,s,\delta,p,n)) \geq \sup_{\alpha>1} \frac{2p}{\sqrt{\alpha}}\left(1 - \Phi\left(\frac{\sqrt{\alpha}\sqrt{2|\log(1-p)|}}{\delta}\right)\right) = 2p\left(1 - \Phi\left(\frac{\sqrt{2|\log(1-p)|}}{\delta}\right)\right), \ \delta > 0.$$

Finally, taking sup over $p \in (0,1)$, results in the following asymptotic bound:

$$\sup_{p\in(0,1)} 2p\left(1 - \Phi\left(\frac{\sqrt{2|\log(1-p)|}}{\delta}\right)\right). \quad \square$$

Appendix D. Proofs of results for the general model in Section 4.3

D.1. Accuracy of the general model: Proof of Theorem 9 and Corollary 10

D.1.1. Proof of Theorem 9

We can measure the accuracy of the classifier for class 0 as

$$\mathrm{acc}_0(f) = P(x \sim \mathcal{D} \ : \ f(x) = 0) = P(x \sim \mathcal{D} \ : \ d_S(x) \leq 0).$$

For any $t \in \mathbb{R}$, the condition that $d_S(x) \leq t$ can be rewritten as

$$d_\pi(x) - \phi(\Pi x) \leq t,$$

which is implied by the condition that

$$d_\pi(x) + |\phi(\Pi x)| \leq t.$$

Introducing the events

$$\begin{aligned} A(\alpha) \ : \quad & x \sim \mathcal{D} \text{ is such that } |\phi(\Pi x)| \leq \alpha, \\ B(\beta) \ : \quad & x \sim \mathcal{D} \text{ is such that } d_\pi(x) \leq \beta, \end{aligned} \tag{D.1}$$

parameterised by the arbitrary values $\alpha \geq 0$, $\beta \in \mathbb{R}$, we find that

$$A(\alpha) \wedge B(t-\alpha) \quad \Rightarrow \quad d_\pi(x) + |\phi(\Pi x)| \leq t. \tag{D.2}$$

Putting these pieces together, we find that

$$\begin{aligned} \mathrm{acc}_0(f) &\geq P(x \sim \mathcal{D} \ : \ d_\pi(x) + |\phi(\Pi x)| \leq 0) \\ &\geq P(x \sim \mathcal{D} \ : \ A(\alpha) \wedge B(-\alpha)). \end{aligned}$$

Negating this event and applying the union bound, we therefore find that

$$\begin{aligned} \mathrm{acc}_0(f) \geq P(x \sim \mathcal{D} \ : \ |\phi(\Pi x)| \leq \alpha) \\ - P(x \sim \mathcal{D} \ : \ d_\pi(x) > -\alpha). \end{aligned}$$

Since $\alpha \geq 0$ was arbitrary, it therefore follows that

$$\begin{aligned} \mathrm{acc}_0(f) \geq \sup_{\alpha \geq 0} \big[& P(x \sim \mathcal{D} \ : \ |\phi(\Pi x)| \leq \alpha) \\ & - P(x \sim \mathcal{D} \ : \ d_\pi(x) > -\alpha) \big]. \end{aligned}$$

D.1.2. Proof of Corollary 10

Since $\phi \equiv 0$ in this case, it follows that

$$P(x \sim \mathcal{E} \ : \ |\phi(x)| \leq \alpha) = 1,$$

for all $\alpha \geq 0$. We may therefore take $\alpha = 0$ in the second term of Theorem 9, and we proceed by bounding

$$P(x \sim \mathcal{E} \ : \ d_\pi(x) \geq 0)$$

from above.

Recalling the bound on the density p of $\mathcal{E}$ in Definition 3, we have

$$P(x \sim \mathcal{E} \ : \ d_\pi(x) \geq 0) \leq \frac{A}{V^n r^n} \int_{\mathbb{B}_r^n(c)} \mathbb{I}_{\{x \ : \ d_\pi(x) \geq 0\}} dx,$$

and the definition of d_π implies that this is

$$\int_{\mathbb{B}_r^n(c)} \mathbb{I}_{\{x \ : \ d_\pi(x) \geq 0\}} dx = \int_{\mathbb{B}_r^n(c)} \mathbb{I}_{\{x \ : \ (x-w)\cdot v \geq 0\}} dx,$$

which is zero for $(w-c)\cdot v > r$ and simply a spherical cap otherwise. Note that the assumption that $d_\pi(c) = -\eta$ for some $\eta > 0$ implies that this spherical cap is less than half the ball $\mathbb{B}_r^n(c)$. Therefore, Lemma 23 implies that

$$P(x \sim \mathcal{E} \ : \ f(x) = 0) \geq 1 - \frac{1}{2} A\left(1 - \left(\frac{\eta}{r}\right)^2\right)^{\frac{n}{2}}$$

D.2. Susceptibility to adversarial perturbations of the general model: Proof of Theorem 11 and Corollary 12

D.2.1. Proof of Theorem 11

The susceptibility of points sampled from class 0 to an adversarial attack with Euclidean norm δ may be measured analogously using the function

$$\mathrm{sus}_0(f) = P(x \sim \mathcal{D} \ : \ \text{there exists } s \in \mathbb{B}^n_\delta \text{ with } f(x+s) \neq 0).$$

The set of points x satisfying the condition in this probability may be seen to be those contained in the union $R \cup T$ of the disjoint sets

$$R = \{x \in \mathbb{R}^n \ : \ d_S(x) > 0\}$$

and

$$T = \{x \in \mathbb{R}^n \ : \ d_S(x) \leq 0 \text{ and } \sigma(x) \leq \delta\};$$

in the first case, since these points are already misclassified it follows that $f(x+s) \neq 0$ for $s = 0 \in \mathbb{B}^n_\delta$, while in the second case the points are correctly classified but they lie within Euclidean distance δ of the decision surface S, due to the definition of σ. To simplify this condition slightly, we observe that

$$\{x \in \mathbb{R}^n \ : \ d_S(x) \geq -\delta\} \subset R \cup T,$$

and therefore

$$\mathrm{sus}_0(f) \geq P(x \sim \mathcal{D} \ : \ d_S(x) \geq -\delta)$$

Arguing as above, we have

$$d_S(x) = d_\pi(x) - \phi(\Pi x) \geq -\delta,$$

which is implied by the condition that

$$|\phi(\Pi x)| - d_\pi(x) \leq \delta.$$

Recalling the events $A(\alpha)$ and $B(\beta)$ from (D.1), we see that for any $\alpha \geq 0$ this event is in turn implied by the event

$$A(\alpha) \wedge \text{not } B(\alpha - \delta),$$

from which it follows that

$$\mathrm{sus}_0(f) \geq P(x \sim \mathcal{D} \ : \ A(\alpha) \wedge \text{not } B(\alpha - \delta)),$$

and negating this event and applying the union bound therefore implies that

$$\mathrm{sus}_0(f) \geq P(x \sim \mathcal{D} \ : \ A(\alpha)) - P(x \sim \mathcal{D} \ : \ B(\alpha - \delta)),$$

and, since $\alpha \geq 0$ was arbitrary,

$$\mathrm{sus}_0(f) \geq \sup_{\alpha \geq 0} \big[P(x \sim \mathcal{D} \ : \ |\phi(\Pi x)| < \alpha) - P(x \sim \mathcal{D} \ : \ d_\pi(x) < \alpha - \delta) \big].$$

D.2.2. Proof of Corollary 12

To prove the Corollary, we start from the result of Theorem 11. Setting $\phi \equiv 0$ and selecting $\alpha = 0$, we find that

$$P(x \sim \mathcal{E} \ : \ \text{there exists } s \in \mathbb{B}^n_\delta \text{ with } f(x+s) \neq 0) \geq 1 - P(x \sim \mathcal{E} \ : \ d_\pi(x) < -\delta).$$

To prove the result, we therefore bound this final term on the right from above.

Recalling the bound on the density p of $\mathcal{E}$ in Definition 3, we have

$$P(x \sim \mathcal{E} \ : \ d_\pi(x) < -\delta) \leq \frac{A}{V^n r^n} \int_{\mathbb{B}^n_r(c)} \mathbb{I}_{\{x \, : \, d_\pi(x) < -\delta\}} \, dx.$$

Here, the assumption that $\delta \in (\eta, r]$ implies that this integral is over a spherical cap which is smaller than a hemisphere, and so we conclude that

$$P(x \sim \mathcal{E} \ : \ d_\pi(x) < -\delta) \leq \frac{1}{2} A \Big(1 - \Big(\frac{\delta - \eta}{r} \Big)^2 \Big)^{\frac{n}{2}},$$

and the result follows.

D.3. Probability of sampling misclassifying random perturbations for the general model: Proof of Lemma 13, Theorem 14 and Corollary 15

D.3.1. Proof of Lemma 13

Geometrically, for any point $x \in \mathbb{R}^n$, the Lipschitz condition on ϕ defines a cone $C(x)$ of points y such that $y \in C(x)$ implies that $d_S(y) \leq 0$, where

$$C(x) = \{y \in \mathbb{R}^n \ : \ (y - \Gamma(x)) \cdot \nu \leq m \|y - \Gamma(x)\|\},$$

with $m = -\cos\theta$ and $\theta = \arctan(L^{-1})$.

Suppose that $z \in \mathbb{R}^n$ is a point which f classifies as class 0, i.e. such that $d_S(z) \leq 0$. The Lipschitz condition on ϕ provides a cone of points $C(z)$ containing z which are guaranteed to also be assigned class 0 by f. This allows us to use a geometric argument to find a lower bound on σ in terms of d_S. Placing a ball $\mathbb{B}^n_\epsilon(z)$ of radius ϵ around z for some $\epsilon \geq 0$, we can observe that the cone $C(z)$ is tangent to this ball when $\epsilon = |d_S(z)| \sin\theta$. This is due to the fact that z lies on the central axis of $C(z)$ (which is oriented in the direction of ν) and $|d_S(z)|$ therefore measures the distance from z to the vertex of $C(z)$. This therefore implies the lower bound that

$$\sigma(z) \geq |d_S(z)| \sin\theta,$$

which we may view as the companion to the upper bound (2). This allows us to control σ from below using d_S, which would not have been possible without such a regularity condition on the surface S.

D.3.2. Proof of Theorem 14

Define the probability of randomly sampling an adversarial perturbation as

$$\mathrm{rand}_0(f) = P(x \sim \mathcal{D}, s \sim \mathcal{U}(\mathbb{B}^n_\delta) \ : \ f(x+s) \neq 0),$$

for fixed δ as in the statement of the theorem. If x is correctly classified, and sampled with $\sigma(x) > \delta$ then there is no possibility of sampling a perturbation s which can destabilise it. We may therefore ignore these points, implying that

$$\mathrm{rand}_0(f) = P(x \sim \mathcal{D}, s \sim \mathcal{U}(\mathbb{B}^n_\delta) \ : \ f(x+s) \neq 0 \text{ and } (\sigma(x) \leq \delta \text{ or } f(x) \neq 0)).$$

Recalling the definition of f, we can rewrite this as

$$\mathrm{rand}_0(f) = P(x \sim \mathcal{D}, s \sim \mathcal{U}(\mathbb{B}^n_\delta) \ : \ d_S(x+s) > 0 \text{ and } (\sigma(x) \leq \delta \text{ or } d_S(x) > 0)).$$

To obtain an upper bound, we slightly refine this splitting of the points x by treating those points which are very close to the decision boundary along with those which are already misclassified. Specifically, let $t \in [0, \delta]$ and introduce

$$K = \{x \in \mathbb{R}^n \ : \ d_S(x) > 0 \text{ or } \sigma(x) \leq t\},$$

which contains those points which are misclassified by f alongside those points which are correctly classified but very close to the decision boundary, and

$$U = \{x \in \mathbb{R}^n \ : \ d_S(x) \leq 0 \text{ and } \sigma(x) \in (t, \delta]\},$$

which contains the correctly classified points which are in a small strip close to, but separated from, the decision boundary. Since these two sets are disjoint and between them contain all of the points which are susceptible to a perturbation of size δ, we have

$$\begin{aligned} &\mathrm{rand}_0(f) \\ &= P(x \sim \mathcal{D}, s \sim \mathcal{U}(\mathbb{B}^n_\delta) \ : \ x \in K \text{ and } f(x+s) \neq 0) \\ &\quad + P(x \sim \mathcal{D}, s \sim \mathcal{U}(\mathbb{B}^n_\delta) \ : \ x \in U \text{ and } f(x+s) \neq 0), \end{aligned} \tag{D.3}$$

and we proceed by obtaining bounds on these two terms separately. Analogously to the proof of Theorem 6, the philosophy here is that the first term is 'small' since it only contains those points which are

misclassified by a slightly worse classifier, while the second term is small because only a small fraction of the sampled perturbations $s \in \mathbb{B}^n_\delta$ are sufficiently large to push the points across the decision boundary.

To bound the first term of (D.3), we use the lower bound of Lemma 13 on σ in terms of d_S to show that the condition $\sigma(x) \leq t$ implies that $|d_S(x)| \leq \frac{t}{\sin\theta}$. From this, the set inclusion

$$K \subset V = \left\{ x \in \mathbb{R}^n \ : \ d_S(x) \geq -\frac{t}{\sin\theta} \right\}$$

follows, enabling us to simplify the term to be bounded as

$$\begin{aligned} &P(x \sim \mathcal{D}, s \sim \mathcal{U}(\mathbb{B}^n_\delta) \ : \ x \in K \text{ and } f(x+s) \neq 0) \\ &\leq P(x \sim \mathcal{D} \ : \ x \in K) \leq P(x \sim \mathcal{D} \ : \ x \in V). \end{aligned}$$

Recalling the definition of the events $A(\alpha)$ and $B(\beta)$ introduced in (D.1), for any $\alpha \geq 0$ the event $A(\alpha) \wedge B(-\alpha - \frac{t}{\sin\theta})$ implies that the event $x \notin V$ holds, and therefore

$$\begin{aligned} P(x \sim \mathcal{D} \ : \ x \in V) &= 1 - P(x \sim \mathcal{D} \ : \ x \notin V) \\ &\leq 1 - P\Big(A(\alpha) \wedge B\Big(-\alpha - \frac{t}{\sin\theta}\Big)\Big). \end{aligned}$$

Inverting this final probability, the union bound implies that

$$\begin{aligned} P(x \sim \mathcal{D} \ : \ x \in V) &\leq P(\text{not } A(\alpha)) \\ &\quad + P\Big(\text{not } B\Big(-\alpha - \frac{t}{\sin\theta}\Big)\Big), \end{aligned}$$

and since $\alpha \geq 0$ was arbitrary it follows that for any $t \in [0, \delta]$

$$\begin{aligned} &P(x \sim \mathcal{D} \ : \ x \in V) \qquad \text{(D.4)} \\ &\leq \inf_{\alpha \geq 0} \Big(P(x \sim \mathcal{D} \ : \ |\phi(\Pi x)| \geq \alpha) \\ &\qquad + P\Big(x \sim \mathcal{D} \ : \ d_\pi(x) \geq -\alpha - \frac{t}{\sin\theta}\Big) \Big), \end{aligned}$$

which completes our bound on the first term of (D.3).

Turning to the second term of (D.3), we can simplify things by including the set U into the larger set

$$U \subset G = \{x \in \mathbb{R}^n \ : \ d_S(x) < -t\},$$

where the inclusion holds due to the upper bound (2) on σ. The reason for this inclusion is that it allows us to study the intersection of the cone $C(x)$ of points with the same classification as x (the existence of which is ensured by the Lipschitz property on ϕ) with the ball of perturbed data points $\mathbb{B}^n_\delta(x)$. Specifically, for any $x \in G$, define the set

$$H(x) = \mathbb{B}^n_\delta(x) \setminus (\mathbb{B}^n_\delta(x) \cap C(x))$$

of perturbations of x which are taken outside the cone $C(x)$ of points guaranteed to be correctly classified.

Suppose that $L \leq 1$. Then, for $t > \delta L$ the set $H(x)$ may be included in a spherical cap which forms less than a hemisphere of $\mathbb{B}^n_\delta(x)$, and which may itself be contained in the larger spherical cap

$$H(x) \subset \{y \in \mathbb{B}^n_\delta(x) \ : \ (y - x) \cdot v > |d_S(x)| - \delta L\}.$$

Since $x \in G$ implies that $d_S(x) < -t$, it follows that

$$H(x) \subset J(x) = \{y \in \mathbb{B}^n_\delta(x) \ : \ (y - x) \cdot v > t - \delta L\},$$

and Lemma 23 implies that the volume of $J(x)$ may be bounded by

$$\frac{1}{2} V^n \delta^n \Big(1 - \Big(\frac{t}{\delta} - L\Big)^2\Big)^{\frac{n}{2}}$$

Consequently, since perturbations are sampled uniformly from $\mathbb{B}^n_\delta$, we obtain the bound

$$\begin{aligned} &P(x \sim \mathcal{D}, s \sim \mathcal{U}(\mathbb{B}^n_\delta) \ : \ x \in U \text{ and } f(x+s) \neq 0) \\ &\leq \frac{1}{2}\Big(1 - \Big(\frac{t}{\delta} - L\Big)^2\Big)^{\frac{n}{2}} P(x \sim \mathcal{D} \ : \ x \in G). \end{aligned}$$

To compute the probability of sampling $x \in G$, we once again recall the definition of the events $A(\alpha)$ and $B(\beta)$ introduced in (D.1), and observe that for any $\gamma \geq 0$

$$A(\gamma) \wedge \text{not } B(\gamma - t) \quad \Rightarrow \quad d_S(x) > -t,$$

and therefore, since

$$\begin{aligned} &P(x \sim \mathcal{D} \ : \ x \in G) = 1 - P(x \sim \mathcal{D} \ : \ d_S(x) > -t) \\ &\leq 1 - P(x \sim \mathcal{D} \ : \ A(\gamma) \wedge \text{not } B(\gamma - t)), \end{aligned}$$

it follows from negating this event, applying the union bound, and recalling that $\gamma \geq 0$ was arbitrary, that

$$\begin{aligned} &P(x \sim \mathcal{D} \ : \ x \in G) \qquad \text{(D.5)} \\ &\leq \inf_{\gamma \geq 0} \Big[P(x \sim \mathcal{D} \ : \ d_\pi(x) \leq \gamma - t) \qquad \text{(D.6)} \\ &\qquad + P(x \sim \mathcal{D} \ : \ |\phi(\Pi x)| > \gamma) \Big]. \qquad \text{(D.7)} \end{aligned}$$

For $L > 1$, however, it is not possible to take $t > \delta L$ since $t \in [0, \delta]$ and so selecting $t = \delta$ provides an optimal result here. In this case, the set U is empty, so this term is simply zero.

Combining the bounds (D.4) and (D.5), we therefore find that

$$\begin{aligned} &\text{rand}_0(f) \\ &\leq \inf_{\substack{\alpha,\gamma \geq 0 \\ t \in T(L)}} \Big[P(x \sim \mathcal{D} \ : \ |\phi(\Pi x)| \geq \alpha) \\ &\qquad + P\Big(x \sim \mathcal{D} \ : \ d_\pi(x) \geq -\alpha - \frac{t}{\sin\theta}\Big) \\ &\qquad + \Delta(L) \frac{1}{2}\Big(1 - \Big(\frac{t}{\delta} - L\Big)^2\Big)^{\frac{n}{2}} \cdot \\ &\qquad\quad \cdot \big(P(x \sim \mathcal{D} \ : \ d_\pi(x) \leq \gamma - t) \\ &\qquad\quad + P(x \sim \mathcal{D} \ : \ |\phi(\Pi x)| > \gamma)\big) \Big], \end{aligned}$$

where $\Delta(L) = 1$ for $L \leq 1$ and 0 for $L > 1$, and the set $T(L) = [\min\{L, 1\}\delta, \delta]$.

D.3.3. Proof of Corollary 15

Since $\phi \equiv 0$, it follows that $L = 0$ and therefore $\sin\theta = 1$. Applying these facts to the result of Theorem 14, and selecting $\alpha = \gamma = 0$, we immediately find that

$$\begin{aligned} &P(x \sim \mathcal{E}, s \sim \mathbb{B}^n_\delta \ : \ f(x+s) \neq 0) \qquad \text{(D.8)} \\ &\quad \leq \inf_{t \in [0,\delta]} \Big[P(x \sim \mathcal{E} \ : \ d_\pi(x) \geq -t) \\ &\qquad + \frac{1}{2}\Big(1 - \Big(\frac{t}{\delta}\Big)^2\Big)^{\frac{n}{2}} \big(P(x \sim \mathcal{E} \ : \ d_\pi(x) \leq -t)\big) \Big]. \end{aligned}$$

Using the crude bound

$$P\Big(x \sim \mathcal{E} \ : \ d_\pi(x) \geq -t\Big) \leq 1,$$

this may be simplified to

$$\begin{aligned} &P(x \sim \mathcal{E}, s \sim \mathbb{B}^n_\delta \ : \ f(x+s) \neq 0) \\ &\quad \leq \inf_{t \in [0,\delta]} \Big[P(x \sim \mathcal{E} \ : \ d_\pi(x) \geq -t) + \frac{1}{2}\Big(1 - \Big(\frac{t}{\delta}\Big)^2\Big)^{\frac{n}{2}} \Big]. \end{aligned}$$

Recalling the bound on the density p of $\mathcal{E}$ in Definition 3, we have

$$P(x \sim \mathcal{E} \ : \ d_\pi(x) \geq -t) \leq \frac{A}{V^n r^n} \int_{\mathbb{B}^n_r(c)} \mathbb{I}_{\{x \ : \ d_\pi(x) \geq -t\}} dx,$$

and, arguing as in the proof of Corollary 10, we note that this may be bounded by

$$P(x \sim \mathcal{E} \ : \ d_\pi(x) \geq -t) \leq \frac{1}{2} A \Big(1 - \Big(\frac{\eta - t}{r}\Big)^2\Big)^{\frac{n}{2}}$$

for any $t \in [0, \delta]$. Substituting this bound into (D.8) and selecting $t = \frac{\eta\delta}{r+\delta}$ (which is a valid choice of t because $\frac{\eta\delta}{r+\delta} \in [0, \frac{\eta}{r+1}]$ for $\delta \in [0, 1]$ and $\eta \in [0, r)$) produces the result.

D.4. Universality of adversarial perturbations for the general model: Proof of Theorem 16 and Corollary 17

D.4.1. Proof of Theorem 16

Since ϕ satisfies the Lipschitz condition with parameter L, a simple geometric argument shows that if $x \in \mathbb{R}^n$ is such that

$$d_S(z) \leq d_S(x) - 2L\delta + \gamma,$$

then $f(z+s) > \gamma \implies f(x+s) > 0$ for all $s \in \mathbb{B}^n_\delta$. Therefore, we bound the probability

$$P(x, z \sim \mathcal{D} \ : \ d_S(z) \leq d_S(x) - 2L\delta + \gamma).$$

For any $t \in \mathbb{R}$, this probability is at least the probability that

$$P(x, z \sim \mathcal{D} \ : d_S(z) \leq d_S(x) - 2L\delta + \gamma \\ \text{and } d_S(x) > t + L\delta - \frac{1}{2}\gamma).$$

When $d_S(x) > t + L\delta - \frac{1}{2}\gamma$, the condition that $d_S(z) \leq t - L\delta + \frac{1}{2}\gamma$ implies that $d_S(z) \leq d_S(x) - 2L\delta + \gamma$, and therefore the probability above is bounded from below by

$$P(x, z \sim \mathcal{D} \ : \ d_S(z) \leq t - L\delta + \frac{1}{2}\gamma \text{ and } d_S(x) > t + L\delta - \frac{1}{2}\gamma),$$

which may be expressed as the product

$$P(z \sim \mathcal{D} \ : \ d_S(z) \leq t - L\delta + \frac{1}{2}\gamma) \cdot \\ \cdot P(x \sim \mathcal{D} \ : \ d_S(x) > t + L\delta - \frac{1}{2}\gamma),$$

since x and z are sampled independently.

Arguing as in the proofs of the previous theorems, using the definitions of the events $A(\alpha)$ and $B(\beta)$ introduced in (D.1), we find that, for any $\alpha \geq 0$,

$$P(z \sim \mathcal{D} \ : \ d_S(z) \leq t - L\delta + \frac{1}{2}\gamma) \\ \geq P(z \sim \mathcal{D} \ : \ |\phi(\Pi z)| \leq \alpha) - P(z \sim \mathcal{D} \ : \ d_\pi(z) > t + \chi),$$

and

$$P(x \sim \mathcal{D} \ : \ d_S(x) > t + L\delta - \frac{1}{2}\gamma) \\ \geq P(x \sim \mathcal{D} \ : \ |\phi(\Pi x)| \leq \alpha) - P(x \sim \mathcal{D} \ : \ d_\pi(x) \leq t - \chi),$$

where $\chi = \frac{1}{2}\gamma - L\delta - \alpha$. The result of the theorem therefore follows since α and t were arbitrary.

D.4.2. Proof of Corollary 17

In this scenario $\phi \equiv 0$, and we have $L = 0$ and may therefore take $\alpha = 0$. This then implies that $\chi = \frac{1}{2}\gamma$, and so, selecting $t = -\eta$ (where we recall that $\eta = d_S(c)$ for the SmAC distribution $\mathcal{E}$), the bound from Theorem 16 becomes

$$P(x, z \sim \mathcal{E} \ : \ f(x+s) \neq 0 \text{ for all } s \in S_z(\delta)) \\ \geq \Big(1 - P(z \sim \mathcal{E} \ : \ d_\pi(z) > -\eta + \frac{1}{2}\gamma)\Big) \cdot \\ \cdot \Big(1 - P(x \sim \mathcal{E} \ : \ d_\pi(x) \leq -\eta - \frac{1}{2}\gamma)\Big).$$

Noting that the bound does not depend on δ, we may switch to using the generic S_z rather than $S_z(\delta)$.

The bound on the density guaranteed by the SmAC property in Definition 3 implies that

$$P(x, z \sim \mathcal{E} \ : \ f(x+s) \neq 0 \text{ for all } s \in S_z) \\ \geq \Big(1 - \frac{A}{V^n r^n} \int_{\mathbb{B}^n_r(c)} \mathbb{I}_{d_\pi(z) > -\eta + \frac{1}{2}\gamma} dz\Big) \cdot \\ \cdot \Big(1 - \frac{A}{V^n r^n} \int_{\mathbb{B}^n_r(c)} \mathbb{I}_{d_\pi(x) \leq -\eta - \frac{1}{2}\gamma} dx\Big),$$

and we observe that both integrals are simply the volume of a spherical cap with height $r - \frac{1}{2}\gamma$, and Lemma 23 therefore implies that

$$P(x, z \sim \mathcal{E} \ : \ f(x+s) \neq 0 \text{ for all } s \in S_z) \\ \geq \Big(1 - A\Big(1 - \frac{\gamma^2}{4r^2}\Big)^{\frac{n}{2}}\Big)^2$$

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., et al. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org. URL https://www.tensorflow.org/.

Bastounis, A., Gorban, A. N., Hansen, A. C., Higham, D. J., Prokhorov, D., Sutton, O., et al. (2023). The boundaries of verifiable accuracy, robustness, and generalisation in deep learning. In *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)*: *Vol. 14254*, LNCS, (pp. 530–541). http://dx.doi.org/10.1007/978-3-031-44207-0_44.

Bastounis, A., Hansen, A. C., & Vlačić, V. (2021). The mathematics of adversarial attacks in ai – why deep learning is unstable despite the existence of stable neural networks. arXiv:2109.06098.

Beyer, L., Hénaff, O. J., Kolesnikov, A., Zhai, X., & van den Oord, A. (2020). Are we done with imagenet? arXiv:2006.07159.

Chaubey, A., Agrawal, N., Barnwal, K., Guliani, K. K., & Mehta, P. (2020). Universal adversarial perturbations: A survey. arXiv:2005.08087.

Cohen, J., Rosenfeld, E., & Kolter, Z. (2019). Certified adversarial robustness via randomized smoothing. In K. Chaudhuri, & R. Salakhutdinov (Eds.), *Proceedings of machine learning research*: *Vol. 97*, *Proceedings of the 36th international conference on machine learning* (pp. 1310–1320). PMLR, URL https://proceedings.mlr.press/v97/cohen19c.html.

Fawzi, A., Moosavi-Dezfooli, S., & Frossard, P. (2016). Robustness of classifiers: from adversarial to random noise. In *Advances in neural information processing systems*: *Vol. 29*.

Feng, H., Huang, S., & Zhou, D.-X. (2023). Generalization analysis of cnns for classification on spheres. *IEEE Transactions on Neural Networks and Learning Systems*, *34*(9), 6200–6213. http://dx.doi.org/10.1109/TNNLS.2021.3134675.

Goodfellow, I. J., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples. In Y. Bengio, & Y. LeCun (Eds.), *3rd international conference on learning representations*. URL http://arxiv.org/abs/1412.6572.

Gorban, A., Golubkov, A., Grechuk, B., Mirkes, E., & Tyukin, I. (2018). Correction of AI systems by linear discriminants: Probabilistic foundations. *Information Sciences*, *466*, 303–322. http://dx.doi.org/10.1016/j.ins.2018.07.040, URL https://www.sciencedirect.com/science/article/pii/S0020025518305607.

Gorban, A. N., Tyukin, I. Y., Prokhorov, D. V., & Sofeikov, K. I. (2016). Approximation with random bases: Pro et contra. *Information Sciences*, *364*, 129–145.

Gupta, K. D., Dasgupta, D., & Akhtar, Z. (2020). Applicability issues of evasion-based adversarial attacks and mitigation techniques. In *2020 IEEE symposium series on computational intelligence* (pp. 1506–1515). IEEE.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE conference on computer vision and pattern recognition* (pp. 770–778). http://dx.doi.org/10.1109/CVPR.2016.90.

Huang, P., Yang, Y., Liu, M., Jia, F., Ma, F., & Zhang, J. (2022). ϵ-weakened robustness of deep neural networks. In *Proceedings of the 31st ACM SIGSOFT international symposium on software testing and analysis* (pp. 126–138).

Khoury, M., & Hadfield-Menell, D. (2018). On the geometry of adversarial examples. arXiv:1811.00525.

King's College London (2024). King's computational research, engineering and technology environment (CREATE). http://dx.doi.org/10.18742/rnvf-m076, Retrieved 2024-02-14.

Krizhevsky, A. (2009). *Learning multiple layers of features from tiny images*: *Tech. rep.*, University of Toronto.

Kurakin, A., Goodfellow, I. J., & Bengio, S. (2018). Adversarial examples in the physical world. In *Artificial intelligence safety and security* (pp. 99–112). Chapman and Hall/CRC.

Ledoux, M. (2001). *The concentration of measure phenomenon*: *Vol. 89*, American Mathematical Soc..

Li, S. (2010). Concise formulas for the area and volume of a hyperspherical cap. *Asian Journal of Mathematics & Statistics*, *4*(1), 66–70.

Li, B., Chen, C., Wang, W., & Carin, L. (2019). Certified adversarial robustness with additive noise. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems*: *Vol. 32*, Curran Associates, Inc., URL https://proceedings.neurips.cc/paper_files/paper/2019/file/335cd1b90bfa4ee70b39d08a4ae0cf2d-Paper.pdf.

Moosavi-Dezfooli, S.-M., Fawzi, A., Fawzi, O., & Frossard, P. (2017). Universal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1765–1773).

Morse, A. P. (1939). The behavior of a function on its critical set. *Annals of Mathematics*, *40*(1), 62–70, URL http://www.jstor.org/stable/1968544.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., et al. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, *115*(3), 211–252. http://dx.doi.org/10.1007/s11263-015-0816-y.

Shafahi, A., Huang, W., Studer, C., Feizi, S., & Goldstein, T. (2019). Are adversarial examples inevitable? In *International conference on learning representations*.

Shamir, A., Melamed, O., & BenShmuel, O. (2022). The dimpled manifold model of adversarial examples in machine learning. arXiv:2106.10151.

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556.

Stallkamp, J., Schlipsing, M., Salmen, J., & Igel, C. (2012). Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural Networks*, *32*, 323–332. http://dx.doi.org/10.1016/j.neunet.2012.02.016, Selected Papers from IJCNN 2011. URL https://www.sciencedirect.com/science/article/pii/S0893608012000457.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., et al. (2014). Intriguing properties of neural networks. arXiv:1312.6199.

Tanay, T., & Griffin, L. (2016). A boundary tilting persepective on the phenomenon of adversarial examples. arXiv:1608.07690.

Tyukin, I. Y., Higham, D. J., & Gorban, A. N. (2020). On adversarial examples and stealth attacks in artificial intelligence systems. In *2020 international joint conference on neural networks* (pp. 1–6). IEEE.

Xiao, H., Rasul, K., & Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv:1708.07747.

Ye, M., Yin, Z., Zhang, T., Du, T., Chen, J., Wang, T., et al. (2024). Unit: a unified look at certified robust training against text adversarial perturbation. In *Proceedings of the 37th international conference on neural information processing systems*. Red Hook, NY, USA: Curran Associates Inc..