PHILOSOPHICAL TRANSACTIONS A

royalsocietypublishing.org/journal/rsta

Survey

Article submitted to journal

**Subject Areas:**
deep learning, image classification, high dimensional analysis

**Keywords:**
adversarial attack, classification, concentration of measure, regulation

**Author for correspondence:**
Desmond J. Higham
e-mail: d.j.higham@ed.ac.uk

THE ROYAL SOCIETY PUBLISHING

# A Survey of Inevitability Results in AI Instability

Alexander Bastounis[1], Desmond J. Higham[2] and Ivan Tyukin[1]

[1]King's College London
[2]University of Edinburgh and Maxwell Institute

Over the past decade, there has been an explosion of activity in the design of algorithms for adversarially attacking AI systems; especially in the context of image classification. For example, a carefully crafted perturbation to an image that is imperceptible to the human eye may cause a sophisticated convolutional neural network to change classification. To deal with this vulnerability, algorithms that detect or defend against such attacks have also been proposed. It has been observed empirically that attackers have the upper hand. To explain these observations, various theoretical results have subsequently emerged. This article will review some recent rigorous results, emphasizing what assumptions they use, in terms of (a) the nature of the training data, (b) the network architectures, and (c) the information available to the attacker. We also discuss how the results may give guidelines for building more secure systems, and how this research area can inform the design of AI regulations.

## 1. Background, Motivation and Scope

Figure 1 illustrates three targeted *adversarial attacks*. The image on the upper left of the figure is correctly classified as a peacock by a sophisticated deep learning tool, in the form of a convolutional neural network [1]. The attacked version of this image shown in the upper right of Figure 1 has been subjected to a tiny perturbation, computed using an algorithm that has access to the inner workings of the network. The aim of the algorithm was to change the predicted class from peacock to scorpion. We see that this was achieved with a visually imperceptible perturbation. The lower left and right images show similar attacks where the new target classes were Rottweiler and toaster, respectively. Implementation details are given at the end of this manuscript.

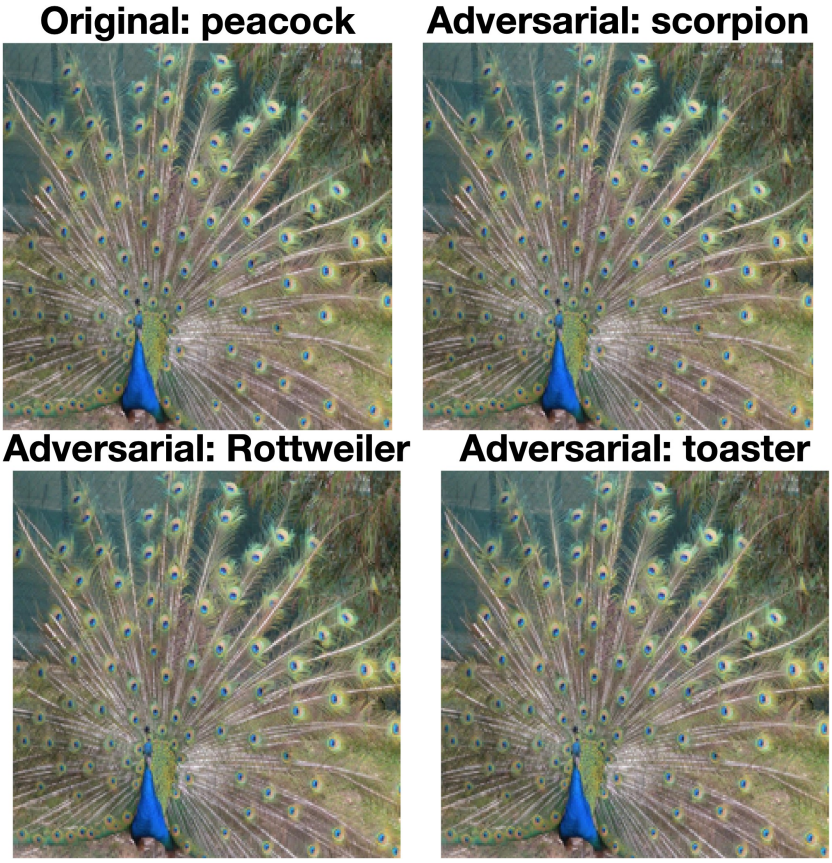


Figure 1: Upper left: an example from ImageNet that is correctly classified as a peacock by a convolutional neural network. Upper right: the classification has been changed to scorpion by making a carefully constructed perturbation that cannot be spotted by the human eye. Lower left and right: similar imperceptible adversarial attacks that alter the predicted class to Rottweiler and toaster, respectively.

We emphasize that Figure 1 has not been cherry-picked; empirical experiments suggest that this type of vulnerability to adversarial attack is present in all convolutional neural networks and applies to almost any image. We are seeing a fundamental lack of robustness; a small change to the input can cause a large change to the output. This phenomenon, which is widespread across the field of artificial intelligence (AI) has been intensively investigated over the past decade, following the seminal articles [2,3].

Adversarial attack algorithms come in various flavours [4]. For example, *white box* attacks require knowledge of the parameters in the AI tool, whereas *black box* attacks do not. In some circumstances, the attacker is not aiming for a perturbation that is imperceptible to a human. One such scenario involves using printable patches to cause a change in the predicted class. These can range from coin-sized printed stickers applied to static objects, [5], to A4-sized sheets of paper attached to clothing, [6]. Similarly, suitably coloured adversarial spectacles can fool facial recognition systems [7]. In these cases, the aim is simply to override an automated AI decision pipeline that does not involve humans. Targeted attacks, such as those shown in Figure 1, require a specific new classification, whereas an *untargeted* version just requires any change of class.

The vulnerability of AI systems to adversarial attack is clearly relevant to issues of safety and security. In the image classification setting, fooling AI in this way can have serious implications for facial recognition, car registration plate recognition, document processing, passport control, interpretation of medical scans, automated driving and content moderation.

These issues are of concern to regulators and policymakers. For example, the amended European Union AI act [8], which is now in force, focuses on high-risk AI systems, a broadly defined category that includes applications in healthcare, infrastructure, law enforcement, education and employment. The act requires that: *"High-risk AI systems shall be resilient as regards to attempts by unauthorised third parties to alter their use, behaviour, outputs or performance by exploiting the system vulnerabilities."* However, empirical evidence suggests that resilience is elusive. The 2025 *International AI Safety Report*[1] from a committee chaired by Yoshua Bengio, states that *"Improved understanding of model internals has advanced both adversarial attacks and defences without a clear winner."* Nicholas Carlini, formerly of Google DeepMind, claims more forcefully in a blog post[2] that defence algorithms published in top computer science conferences have proved surprisingly easy to overcome (as, for example, in [9]) and argues that *"we're not going to be able to deploy machine learning models as widely as we'd like if it's trivial to make them do bad things."*

Alongside safety and security, explainable AI—asking the system to give an easy-to-understand justification for a prediction—is also a desirable goal. However, this is immediately scuppered by vulnerability to adversarial attack; there is no plausible explanation for the upper right image in Figure 1 to be a scorpion. More fundamentally, researchers have also shown that the outputs from algorithms which try to provide explanations can be manipulated by adversarial attackers [10].

The cat-and-mouse world of designing and empirically testing attack and defence algorithms shows no sign of slowing down. Some researchers have therefore focused on the bigger picture question of whether there are fundamental vulnerabilities in AI systems that can be established rigorously. Results of this nature typically focus on *inevitability*: for a typical input, does a small class-changing perturbation always exist with high probability, and *computability*: can such a perturbation be found in practice? Of course in order to establish rigorous results, assumptions must be made, for example about the classes of AI system under study, the type of training and test data being used, the sense in which stability is being measured, whether worst-case or average performance is of interest, and what information is available to the attacker. This has led to a range of results that give theoretical insights into the computational boundaries of AI, and hence can inform the choices that need to be made by developers, policymakers and end-users. Such results also help to quantify the *tradeoff* that must be made between accuracy (performing well on training and test data) and stability (producing results that are not sensitive to small changes in the input).

In this survey, we describe theoretical results that have been derived in the field of adversarial attacks. We focus on image classification with deep learning networks and the case of small perturbations.

The manuscript is organised as follows. In section 2 we use the setting of linear classifiers to give a feel for the ease with which adversarial attacks can be computed. Section 3 summarizes bounds that can be derived for the worst case sensitivity of deep learning networks. The heart of the article is section 4, which reviews a range of theoretical results that provide insight into the vulnerability of classifiers. Results in this field typically quantify limitations, and hence are of a negative nature. However, it may be argued that they can indirectly have a positive effect by alerting developers and users to pitfalls and by identifying scenarios where results cannot be trusted. In section 5 we briefly mention how positive consequences may also arise directly from this theory. Section 6 provides brief conclusions.

We note that a related field where rigorous results have been established concerns *adversarial training*. Here, researchers study properties of the underlying optimization problems. For example, [11] considers a collection of classifiers and seeks one for which perturbations of a given

[1] https://www.gov.uk/government/publications/international-ai-safety-report-2025
[2] https://nicholas.carlini.com/writing

size have the least impact. We are concerned here with the properties that remain when such an "optimal" classifier has been identified.

## 2. Insights From Linearisation

Although practical attack algorithms are not the main focus of this survey, we give here a brief indication of how cheap strategies can be developed and why they might be successful. Following [2] we start with the simple case of binary classification with a linear classifier. Suppose an input data point $x \in \mathbb{R}^n$ is to be assigned to class A or class B according to

$$w^T x + b \geq 0 \Rightarrow \text{class A}, \qquad w^T x + b < 0 \Rightarrow \text{class B}.$$

Here, the weight vector $w \in \mathbb{R}^n$ and bias $b \in \mathbb{R}$ have been fitted to training data. Can a small perturbation to the input cause a change to the output classification? If we perturb $x$ to $x + \Delta x$ then, because the classifier is linear, the output changes by an amount $w^T \Delta x$. Suppose we restrict the perturbation in a componentwise sense, so that $|\Delta x_i| \leq \epsilon$ for $i = 1, 2, \ldots, n$, where $\epsilon > 0$. Then the biggest change is clearly given by choosing $\Delta x = \epsilon \operatorname{sign}(w)$ or $\Delta x = -\epsilon \operatorname{sign}(w)$. (This notation means that sign is applied separately to each component.) This allows us to increase or decrease the output by an amount $\epsilon \sum_{i=1}^n |w_i|$, which we may trivially rewrite as

$$n\,\epsilon \left( \frac{1}{n} \sum_{i=1}^{n} |w_i| \right).$$

We may interpret this result as follows: if the average weight is $O(1)$, then when the input dimension is high ($n$ large) we can make a significant change to the output with a small componentwise perturbation ($\epsilon$ small).

This argument remains relevant more generally because smooth functions respond almost linearly to small perturbations. Also, we can extend the ideas to other norms, as discussed for example in [12], and to targeted perturbations. We use $\|\cdot\|_p$ to the denote the vector $p$-norm, so $\|x\|_p := \left(\sum_{i=1}^n |x_i|^p\right)^{1/p}$ and given $p \geq 1$ we let $q$ be the Hölder conjugate, defined by $p^{-1} + q^{-1} = 1$ (with $q = \infty$ being the Hölder conjugate when $p = 1$, and vice versa).

To explain further, consider an image classification setting. We regard $n$ as the number of pixels and let $c$ denote the number of possible image classes. Suppose we have a smooth classification function $F : \mathbb{R}^n \to \mathbb{R}^c$ that can produce positive or negative values, with $F(x)_r > F(x)_s$ indicating that class $r$ is more likely than class $s$. So the predicted class for input $x$ corresponds to the largest component of $F(x)$. (In practice, we could use a softmax layer to convert these outputs into probabilities; however, it is more natural for us to work with the pre-softmax output, since this can take any real value.)

To be concrete, given $x$, we will suppose that the map $F$ assigns $x$ to class $r$. We will let $f : \mathbb{R}^n \to \mathbb{R}$ denote the map corresponding to component $r$ of $F$; that is $f(x) = F(x)_r$. We want to find a small perturbation $\Delta x$ that makes a large negative change in $f(x)$, in the hope that this will cause $x + \Delta x$ to be classified differently to $x$. When $\Delta x$ is small we have

$$f(x + \Delta x) \approx f(x) + \Delta x^T \nabla f(x). \tag{2.1}$$

(We are then effectively studying a linear classifier.) Suppose we measure the size of $\Delta x$ in the $p$-norm; that is, we restrict to $\|\Delta x\|_p \leq \epsilon$, for some small $\epsilon > 0$. Lemma 2.1 below shows that we can cause the most negative change, $-\epsilon \|\nabla f(x)\|_q$, in (2.1) with

$$\Delta x_i = -\epsilon \operatorname{sign}(\nabla f(x)_i) \frac{|\nabla f(x)_i|^{q/p}}{\|\nabla f(x)\|_q^{q-1}}. \tag{2.2}$$

**Lemma 2.1.** *Given $p \geq 1$ and $v \in \mathbb{R}^n$, we have*

$$\min_{u \in \mathbb{R}^n,\, \|u\|_p \leq \epsilon} u^T v = -\epsilon \|v\|_q,$$

*and for $v \neq 0$ this minimum is uniquely achieved by*

$$u_i = -\epsilon \operatorname{sign}(v_i) \frac{|v_i|^{q/p}}{\|v\|_q^{q-1}}.$$

*Proof.* The minimization result follows directly from the Hölder inequality [13]. Direct calculation shows that $u$ displayed in the lemma achieves the result. □

Figure 2 illustrates how the choice of $p$ affects the perturbation direction in the two-dimensional case.

A simple attack algorithm is then, for example, to regard (2.2) as a line search direction and iterate on $\epsilon$ to get a good approximation to the smallest perturbation that causes a change in classification. In the case $p = \infty$ we revert to the setting at the start of this section. For $p = 2$ (when Hölder becomes Cauchy–Schwarz) we have what could be called "gradient ascent."

Now consider a *target class*, $s$, and let $g : \mathbb{R}^n \to \mathbb{R}$ denote the corresponding map: $g(x) = F(x)_s$. Suppose we seek a small $p$-norm perturbation to $x$ such that $x + \Delta x$ is at least as likely to be in class $s$ as class $r$, according to the classifier; that is, we require $f(x + \Delta x) \leq g(x + \Delta x)$. Upon linearisation, this becomes $f(x) - g(x) + \Delta x^T (\nabla f(x) - \nabla g(x)) \leq 0$. Lemma 2.2 below shows that this is achieved with

$$\Delta x_i = -\operatorname{sign}(\nabla_{\text{diff}}(x)_i) \frac{f(x) - g(x)}{\| \left(\nabla_{\text{diff}}(x)\right)^{p/q} \|_p \, \|\nabla_{\text{diff}}(x)\|_q} | \left(\nabla_{\text{diff}}(x)\right)_i |^{p/q}, \tag{2.3}$$

where $\nabla_{\text{diff}}(x)$ denotes $\nabla f(x) - \nabla g(x)$.

**Lemma 2.2.** *Given $p \geq 1$, $a > 0$ and a nonzero vector $b \in \mathbb{R}^n$, the problem*

$$\min_{y \in \mathbb{R}^n} \|y\|_p \text{ such that } a + y^T b \leq 0, \tag{2.4}$$

*is uniquely solved by*

$$y_i = -\operatorname{sign}(b_i) \frac{a}{\|b^{p/q}\|_p \, \|b\|_q} |b_i|^{q/p}, \tag{2.5}$$

*for which*

$$\|y\|_p = \frac{a}{\|b\|_q}. \tag{2.6}$$

*Proof.* We know from the Hölder inequality [13] that $|y^T b| \leq \|y\|_p \|b\|_q$. Hence, to achieve the inequality in (2.4) we require $\|y\|_p \geq a / \|b\|_q$. It may be checked that $y$ in (2.5) achieves the inequality and satisfies (2.6), as desired. □

Hence, $\Delta x$ in (2.3) may be used as a line search direction in a targeted attack that attempts to change the classification from $r$ to $s$.

This approach can be varied or extended in many ways [4]. We briefly mention some key ideas here. For example, rather than directly attacking the output classification $F(x)$ it is possible to choose $\Delta x$ so that the loss function increases when the network is presented with $x + \Delta x$ and the correct label; for $p = \infty$ this corresponds to the Fast Gradient Sign Method from [2]. It can also be effective to regard each gradient computation as producing one step of an iterative process, as in traditional gradient descent [1], and it is generally necessary to constrain the pixel values so that they lie in a valid range. It is also possible to make use of more sophisticated nonlinear, constrained optimization algorithms [14]. Whether the output or the loss function is attacked, the corresponding partial derivatives (with respect to $x$) are available cheaply by a process akin to back propagation. If the attacker does not have access to these partial derivatives—the so-called black-box setting—then finite-differences may be used, or a proxy model could be built by the

attacker from which corresponding derivatives may be computed. The latter approach exploits the widely-reported *universality* effect, where the same perturbation is found to work well across different classification tools [15,16] .

The sense in which a perturbation is to be regarded as "small" is of course a matter of choice. As illustrated in Figure 2, the choice $p = \infty$ allows every pixel to be altered by the same amount. At the other extreme, $p = 1$ encourages relatively few pixels to be altered, akin to the extreme case of a one-pixel attack [17]. Rather than using a vector norm directly, it is also possible to consider componentwise perturbations, where each pixel is changed by some relative amount in order, for example, to avoid smudging backgrounds [18].

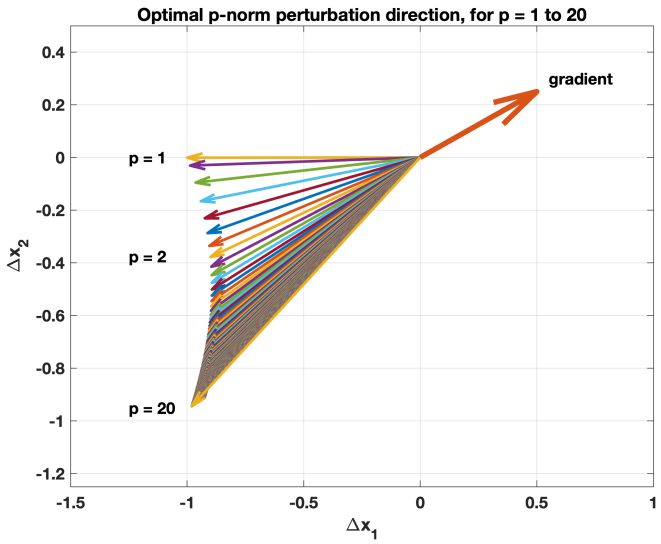


Figure 2: Illustration of Lemma 2.1 for dimension $n = 2$. Given the gradient vector indicated, we show the optimal direction $\Delta x$ subject to the constraint $\|\Delta x\|_p \leq \epsilon$ for $\epsilon = 1$, as characterised in (2.2). We see that $p = 1$ gives $\Delta x = [-1, 0]^T$, so that only the first pixel is altered. For $p = 2$ we have $\Delta x$ in the opposite direction to the gradient. In the limit $p \to \infty$ the value of $\Delta x$ tends to $[-1, -1]^T$, so that both pixels are perturbed by the same amount.

## 3. Worst Case Bound

From the perspective of applied and computational mathematics, vulnerability of classifiers to adversarial perturbations is an example of *ill-conditioning*—a small change to the input can cause a large change to the output [19]. We may then ask: what is the condition number of a deep learning classifier? We will consider the vanilla case of a feed-forward neural network with $L$ layers, and with $n_\ell$ neurons at layer $\ell$, [1,20]. This classifier may be regarded as a nonlinear map $F : \mathbb{R}^{n_1} \to \mathbb{R}^{n_L}$. Given an input $x \in \mathbb{R}^{n_1}$ we have

$$\begin{aligned} a^{[1]}(x) &= x, \\ a^{[\ell]}(x) &= \sigma^{[\ell]}\left(W^{[\ell]} a^{[\ell-1]}(x) + b^{[\ell]}\right) \in \mathbb{R}^{n_\ell}, \quad \text{for } \ell = 2, 3, \ldots, L, \end{aligned}$$

and we set $F(x) = a^{[L]}(x)$. Here, each $W^{[\ell]} \in \mathbb{R}^{n_\ell \times n_{\ell-1}}$ and $b^{[\ell]} \in \mathbb{R}^{n_\ell}$ are the weight matrices and bias vectors that are learned during the training phase. The nonlinear activation functions $\sigma^{[\ell]}$ are applied in a componentwise sense.

Basic calculus then shows that $F$ has a Jacobian of the form

$$\frac{\partial F}{\partial x} = D^{[L]}(x)\, W^{[L]}\, D^{[L-1]}(x)\, W^{[L-1]} \cdots D^{[2]}(x)\, W^{[2]}, \tag{3.1}$$

where

$$D^{[\ell]}(x) = \mathrm{diag}(\sigma^{[\ell]'}(W^{[\ell]}a^{[\ell-1]}(x) + b^{[\ell]})) \in \mathbb{R}^{n_\ell \times n_\ell}, \quad \text{for } \ell = 2, 3, \ldots, L.$$

The norm of the Jacobian characterizes sensitivity to small input perturbations [19].

The widely used sigmoid activation function $\sigma(x) = 1/(1+\exp(-x))$ has a derivative bounded above in modulus by $1/4$. Generally, if the derivatives of the $\sigma^{[\ell]}$ can be bounded above in modulus by $K$, then on taking the Euclidean norm in (3.1) we obtain

$$\|\frac{\partial F}{\partial x}\|_2 \leq \left(\prod_{\ell=2}^{L} \|W^{[\ell]}\|_2\right) K^{L-1}. \tag{3.2}$$

This expression shows that we can bound the worst-case sensitivity of a neural network by controlling the norms of the weight matrices, $\|W^{[\ell]}\|_2$. In fact, it is common to discourage large weight matrix norms during training for a slightly different reason; namely, as a form of regularization, to address the issue of overfitting. Several authors have noted that weight matrices play a key role in the condition number, or closely-related concept of Lipschitz constant [3,21–23], and in [24] the idea of *Parseval networks* was proposed. Here, weights are chosen so that the Lipschitz constant for each layer is less than one. However, as we discuss in the next section, there is an inherent trade-off between accuracy and stability, so that imposing such a constraint must, in some scenarios, degrade classification performance.

## 4. General Results in High Dimension

Current image classifiers operate in the world of high dimension—hundreds, thousands or millions of pixels in an image, with up to billions of model parameters. This is a regime where the fascinating, but highly counterintuitive, effects of high dimensional geometry [25] are present; concepts from this field offer one way to understand why classification can be challenging

To give a feel for the type of result that can be derived, we will consider a binary classification problem involving points on the unit sphere in $\mathbb{R}^n$. (We associate $n$ with the number of pixels.) We regard the first coordinate axis as the "north pole" and study the case where points in the northern hemisphere $x_1 > 0$ are in class A and points in the southern hemisphere $x_1 < 0$ are in class B. We will argue that the ground truth classifier, which uses the sign of $x_1$, is vulnerable to small perturbations. Suppose that each data point $x$ is drawn at random on the unit sphere, with each component of $x$ having the same distribution. Since $x_1^2 + x_2^2 + \cdots + x_n^2 = 1$, the expected value of each $x_i^2$, is $\mathbb{E}[x_i^2] = 1/n$. We claim that data points concentrate around the corresponding equator, that is, $x_1 \approx 0$. To show this we appeal to Markov's inequality [26] which says that for any $a > 0$ we have

$$\mathbb{P}(x_1^2 \geq a) \leq \frac{\mathbb{E}[x_1^2]}{a} = \frac{1}{na}.$$

Taking $a = 1/\sqrt{n}$, we obtain $\mathbb{P}(x_1^2 \geq 1/\sqrt{n}) \leq 1/\sqrt{n}$, or equivalently, $\mathbb{P}(|x_1| \geq 1/n^{\frac{1}{4}}) \leq 1/\sqrt{n}$. We conclude that in high dimension, $|x_1|$ must be small with high probability. Hence, a typical data point lies close to the (true) decision boundary; so its classification may be changed under a small perturbation. By continuity, the same conclusion must hold for any accurate classification algorithm.

This type of *concentration of measure* result, which can be dramatically sharpened and generalized, has been used by various authors to explain vulnerability to adversarial attack. Some results, such as the one above, make assumptions about the data distribution but do not place restrictions on the classifier. Other results require the type of classifier to be specified, such as a neural network with a given architecture, and show that it is possible to construct a training set

and test set for which accuracy is incompatible with stability. Most results in this area measure perturbation size in the 2-norm, however, other norms have also been studied.

In the following list, we summarize the main characteristics of a range of articles where rigorous theory for adversarial attacks has been developed. To give a high-level overview, Table 1 summarizes their key features.

- [2], as illustrated in section 2, considers binary classification with a linear classifier and measures perturbation size in $\infty$-norm. Without requiring any statistical assumptions about the training or test data, it is shown that in high dimension small attacks can cause large changes to the output.
- [27] studies linear and quadratic classifiers for binary classification. Robustness is measured in an average sense (average of smallest classification-changing perturbation in 2-norm). It is shown that robustness to random, rather than worst case, perturbations is higher by a factor proportional to $\sqrt{n}$, where $n$ is input dimension.
- [28] considers random data generated by applying a smooth function to a Guassian. Given $\eta > 0$, this work studies the probability that a classification-changing perturbation of size $\leq \eta$ exists. The analysis, which uses a Gaussian isoperimetric inequality, demonstrates increasing sensitivity in terms of input dimension. The work also shows that the same successful attack perturbation can be transferred from one classifier to another if they are both accurate.
- [29] provides universal bounds on the susceptibility of a general classifier to adversarial attacks. It is shown that if training and test data distributions are not excessively concentrated then most points admit adversarial perturbations which are small in terms of the dimension $n$. The approach has a similar feel to the example given at the start of this section, but uses a more sophisticated isoperimetric inequality. *"The idea is to show that, provided a class of data points takes up enough space, nearly every point in the class lies close to the class boundary."*
- [30] considers binary classification, with training and test data from two concentric spheres, with perturbations measured in the 2-norm. It uses concentration of measure arguments based on spherical caps to show that any model misclassifying a small constant fraction of a sphere can be fooled, in an average sense, by perturbations of size $1/\sqrt{n}$.
- [31] extends the idea in [2] to multi-layer neural networks and general $p$-norm perturbations, also giving conditions under which successful attacks can have size $1/\sqrt{n}$.
- [32] has a similar spirit to [29] in the sense that general classifiers are considered and the training/test data is assumed to come from a high-dimensional distribution where data is inevitably close to a decision boundary with high probability. The smeared Absolute Continuity condition from [33] is the key assumption on the data distribution. Further analysis under the smeared Absolute Continuity condition is given in [34]. Here a general setup for a linear classifier is defined that simultaneously captures a range of empirically observed features: (a) small class-changing perturbations exist with high probability, (b) they can be computed with a gradient-based attack algorithm, (c) a successful perturbation will be universal in the sense that it may be used to attack other data from the same class, (d) random perturbations make for ineffective attacks. The analysis is also extended to nonlinear decision boundaries.
- [35] considers the somewhat artificial case of ReLU neural networks with random Gaussian entries in the weight matrices, giving conditions under which a successful adversarial attack on an arbitrary input exists, in the small perturbation, high probability of success sense, as the input dimension increases. The authors give a helpful intuitive justification of this theoretical result, which resonates with the example at the start of section 2. In the case of a simple linear classifier $w^T x$, if $w$ and $x$ are high dimensional and random, then it is known that $w^T x \ll \|w\|_2 \, \|x\|_2$ with high probability. However, trivially, perturbing $x$ by an amount $\Delta x = \epsilon w$ causes the change $\epsilon \|w\|_2^2$. So a small $\epsilon$ is sufficient to make $w^T(x + \Delta x)$ have a different sign to $w^T x$.

- [36] considers binary classification with any classifier. For a specific training/test set distribution it is shown that a tradeoff exists where, in high dimension, good classification accuracy must inevitably expose a vulnerability to adversarial attack. In a similar manner to [2], the attack perturbation is measured in $\infty$-norm and chosen via the Hölder inequality.
- [37] shows that for a given feed forward neural network architecture, there exists a distribution of training/test data for which (a) there exist networks that have perfect accuracy but are vulnerable to small input perturbations, and (b) there also exist networks that have perfect accuracy and are robust. Moreover, pairs of stable and unstable networks can be arbitrarily close in parameter space.
- [38] also considers feed forward neural networks with a given depth and given layer sizes, and shows that (a) there exists a distribution of training/test data that is well-separated (the size of the adversarial perturbation is independent of the distance between the classes observed in the training/validation data) for which any accurate network from this class must be vulnerable to attack, whereas (b) there exists a larger network, with either greater depth or greater layer sizes, that is accurate and stable.

Table 1: Key aspects of some rigorous results concerning vulnerability to adversarial attack. Here alg denotes that that an algorithm is available for the attack and asymp denotes that full inevitability/arbitrarily small perturbation size arises in the limit of high dimension.

| | classifier | training data | attacked data | norm | alg? | asymp? |
|---|---|---|---|---|---|---|
| [2] | linear | arbitrary | arbitrary | $\|\cdot\|_\infty$ | Yes | Yes |
| [27] | linear/quadratic | generated | random | arbitrary | No | No |
| [28] | arbitrary | random | random | $\|\cdot\|_2$ | No | No |
| [29] | arbitrary | not rel. | random | $\|\cdot\|_p$ or geodesic | No | Yes |
| [30] | arbitrary | random | random | $\|\cdot\|_2$ | No | Yes |
| [31] | linearized | arbitrary | arbitrary | $\|\cdot\|_p$ | Yes | Yes |
| [32] | arbitrary | random | random | $\|\cdot\|_2$ | Yes | Yes |
| [34] | arbitrary | random | random | $\|\cdot\|_2$ | Yes | Yes |
| [35] | random NN/ReLU | N/A | arbitrary | $\|\cdot\|_2$ | No | Yes |
| [36] | arbitrary | random | random | $\|\cdot\|_\infty$ | Yes | Yes |
| [37] | neural network | random | random | $\|\cdot\|_2$ | Yes | Yes |
| [38] | neural network | specific | specific | arbitrary | Yes | No |

In several of the theoretical set-ups used in Table 1, the authors note that AI classifiers are typically much less vulnerable to random perturbations—the instability is only revealed by very specific, gradient-based perturbations. This effect has also been observed empirically. As a consequence, it is not sufficient to stress-test an AI system against purely random attacks. From a concentration of measure perspective, this effect can be understood from the result that, in high dimension, for a broad range of distributions, randomly chosen vectors are extremely likely to be almost orthogonal. So, in the linearised case discussed in section 2, a randomly chosen vector $\Delta x$ will typically have only a tiny component in the direction of the optimal perturbations given by Lemma 2.1 or Lemma 2.2. Section 6 of [34] also shows that the balance between the effectiveness of random and structured attacks can depend strongly on the margin between classes.

So far, we have considered the stability of an AI system with respect to input perturbations. It is also of interest to ask how much damage can be done if we are allowed to change the values of the model parameters. This question is relevant in the setting where an untrustworthy software team supplies code to third-party users. The idea of a *stealth attack* was introduced in [32] and

studied further in [39]. Here, a small number of weights or biases are altered in such a way that the system performance is unchanged on a large validation set, but the desired, adversarial, output is produced on a specific target input. It was shown that under reasonable assumptions a successful attack can be computed with high probability. We note that this is a type of sensitivity with respect to the system parameters, rather than the input data. Hence, a Jacobian bound such as (3.2) is not informative. Instead, this type of stealth attack relies on the ideas that (a) many neurons are redundant in the sense that they may be removed without significantly affecting performance, and (b) a replacement neuron can be inserted which essentially acts independently from the rest.

It is of interest to note that the stealth attack framework, and the underlying mathematical analysis, can be traced back to a more noble pursuit. Concentration of measure results show that in high dimension very many independently sampled points are very likely to be linearly separable; that is, they will all form vertices of the corresponding convex hull. For example, bounds from [40] show that 3 million uniformly random points in the unit Euclidean ball in $\mathbb{R}^{50}$ will all be vertices of their convex hull with probability over $0.99$. As mentioned in [40], this observation implies that an expensively-trained model can be cheaply fixed "on-the-fly": if an error is observed for certain input, then an extra neuron may be added which remains silent on previously seen data but fires, in a way that corrects the performance, on this new input. However, since AI models are typically massively over-parameterized and hence many parameters are redundant, it is also possible to alter the weights and biases of a small number of neurons in order to alter the performance adversarially on a specific target input; leading to the idea of stealth attacks.

## 5. Positive Consequences

Instabilities in AI systems may also be leveraged in a positive way, essentially turning attack into defence. Nightshade [41] can be used to imperceptibly poison a digital image in a way that degrades the training of a generative AI model. This allows an artist to protect their work against unwanted or illegal exploitation, given that unscrupulous third parties often choose to ignore opt-out or do-not-crawl directives. The related tool Glaze [42] imperceptibly perturbs images in a way that makes it difficult for generative models to copy their overall style. In a similar way, Fawkes [43] allows a user to add small perurbations (called cloaks) to photographs of themselves before making them publicly available. If the cloaked images are used as training data, the resulting facial recognition model will consistently misidentify normal photographs of the user.

Invisible perturbations may also be used for watermarking digital images, to assert ownership and monitor unauthorized alterations [44], although this is an area where a separate arms race is currently evolving [45].

## 6. Outlook and Conclusions

The vulnerabilities that we have discussed may be viewed as the AI equivalent of optical illusions. Transforming high-dimensional details into a low dimensional prediction inevitably entails a loss of information, and it is intuitively reasonable that there is a tradeoff between accuracy and robustness. Theoretical results in this area follow from assumptions that may involve the structure of the classifier, the nature and dimension of the training/test data and the manner in which perturbation size is measured. Although there is currently a gap between rigorous results and practical computation, we argue that (a) current theory gives a plausible high-level explanation for the empirically observed vulnerabilities in current AI classifiers and (b) attempts to impose generically-worded, unquantified, regulations on AI reliability must acknowledge the existence of inevitability results.

Looking ahead, we expect to see the assumptions underlying such inevitability results become more realistic and more easily verifiable; for example, building on concepts such as separability-based intrinsic dimensionality [46]. We also emphasize that this article has focused on attacking image classifiers, with no attempt to cover the rapidly growing literature on vulnerabilities in other areas of AI.

## Data Statement

Figure 1 was created within the the MATLAB Deep Learning Toolbox [47], using the demonstration code[3]. (The only significant edit was changing the target class from "great white shark" to "scorpion," then "Rottweiler," then "toaster".) The network under attack is `squeezenet`, an 18 layer convolutional neural network trained on over a million images from the ImageNet database `http://www.image-net.org`. ImageNet contains pictures from 1000 object categories.

Acknowledgements. DJH was supported by a Fellowship from the Leverhulme Trust and by the Advanced Grant "Numerical Analysis for Stable AI" 101198795 from the European Research Council. IT was supported by the UKRI Turing AI Fellowship EP/V025295/2.

## References

1. Goodfellow I, Bengio Y, Courville A. 2016 *Deep learning*. Adaptive computation and machine learning. The MIT Press.
2. Goodfellow IJ, Shlens J, Szegedy C. 2015 Explaining and Harnessing Adversarial Examples. In Bengio Y, LeCun Y, editors, *3rd International Conference on Learning Representations, San Diego, CA*.
3. Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow I, Fergus R. 2013 Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.
4. Costa JC, Roxo T, Proença H, Inácio PRM. 2024 How Deep Learning Sees the World: A Survey on Adversarial Attacks & Defenses. *IEEE Access* **12**, 61113–61136. (10.1109/ACCESS.2024.3395118)
5. Brown TB, Mané D, Roy A, Abadi M, Gilmer J. 2017 Adversarial patch. *arXiv:1712.09665*.
6. Thys S, Ranst WV, Goedemé T. 2019 Fooling Automated Surveillance Cameras: Adversarial Patches to Attack Person Detection. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* pp. 49–55. (10.1109/CVPRW.2019.00012)
7. Sharif M, Bhagavatula S, Bauer L, Reiter MK. 2019 A General Framework for Adversarial Examples with Objectives. *ACM Trans. Priv. Secur.* **22**. (10.1145/3317611)
8. European Parliament. 2023 Amendments adopted by the European Parliament on 14 June 2023 on the proposal for a regulation of the European Parliament and of the Council on laying down harmonised rules on artificial intelligence (Artificial Intelligence Act) and amending certain Union legislative acts. .
9. Carlini N. 2024 Cutting through buggy adversarial example defenses: fixing 1 line of code breaks SABRE. *arXiv preprint arXiv:2405.03672*.
10. Dombrowski AK, Alber M, Anders CJ, Ackermann M, Müller KR, Kessel P. 2019 Explanations can be manipulated and geometry is to blame. In Wallach HM, Larochelle H, Beygelzimer A, d'Alché Buc F, Fox EB, Garnett R, editors, *NeurIPS* pp. 13567–13578.
11. Bungert L, Stinson K. 2024 Gamma-convergence of a nonlocal perimeter arising in adversarial machine learning. *Calculus of Variations* **63**.
12. Huang R, Xu B, Schuurmans D, Szepesvári C. 2016 Learning with a Strong Adversary. *arXiv preprint arXiv:1511.03034*.
13. Horn RA, Johnson CR. 2013 *Matrix Analysis*. Cambridge; New York: Cambridge University Press 2nd edition.
14. Rony J, Granger E, Pedersoli M, Ben Ayed I. 2021 Augmented Lagrangian Adversarial Attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* pp. 7738–7747.
15. Khrulkov V, Oseledets I. 2018 Art of singular vectors and universal adversarial perturbations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
16. Moosavi-Dezfooli S, Fawzi A, Fawzi O, Frossard P. 2017 Universal Adversarial Perturbations. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* pp. 86–94.
17. Su J, Vargas DV, Sakurai K. 2019 One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation* **23**, 828–841.
18. Beerens L, Higham DJ. 2024 Adversarial ink: componentwise backward error attacks on deep learning. *IMA Journal of Applied Mathematics* **89**, 175–196.

[3] `https://uk.mathworks.com/help/deeplearning/ug/generate-adversarial-examples.html`.

19. Higham NJ. 2002 *Accuracy and Stability of Numerical Algorithms*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics second edition.
20. Higham CF, Higham DJ. 2019 Deep learning: An introduction for applied mathematicians. *SIAM Review* **61**, 860–891.
21. Budzinskiy S, Wenyi Fang LZ, Petersen P. 2025 Numerical Error Analysis of Large Language Models. *arXiv preprint arXiv:2503.10251*.
22. Bartlett PL, Foster DJ, Telgarsky MJ. 2017 Spectrally-normalized margin bounds for neural networks. In *Advances in Neural Information Processing Systems* vol. 30.
23. Zühlke MM, Kudenko D. 2025 Adversarial robustness of neural networks from the perspective of Lipschitz calculus: A survey. *ACM Comput. Surv.* **57**. (10.1145/3648351)
24. Cisse M, Bojanowski P, Grave E, Dauphin Y, Usunier N. 2017 Parseval Networks: Improving Robustness to Adversarial Examples. In Precup D, Teh YW, editors, *Proceedings of the 34th International Conference on Machine Learning* vol. 70*Proceedings of Machine Learning Research* pp. 854–863 International Convention Centre, Sydney, Australia. PMLR.
25. Ball K. 1997 An elementary introduction to modern convex geometry. *Flavors of Geometry* **31**, 1–58.
26. Grimmett GR, Stirzaker DR. 2001 Random processes. In *Probability and Random Processes* , . Oxford University Press. (10.1093/oso/9780198572237.003.0008)
27. Fawzi A, Fawzi O, Frossard P. 2018a Analysis of classifiers' robustness to adversarial perturbations. *Machine Learning* **107**, 481–508.
28. Fawzi A, Fawzi H, Fawzi O. 2018b Adversarial vulnerability for any classifier. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems* NIPS'18 p. 1186–1195 Red Hook, NY, USA. Curran Associates Inc.
29. Shafahi A, Huang W, Studer C, Feizi S, Goldstein T. 2019 Are adversarial examples inevitable?. *International Conference on Learning Representations, New Orleans, USA*.
30. Gilmer J, Metz L, Faghri F, Schoenholz SS, Raghu M, Wattenberg M, Goodfellow I. 2018 The Relationship Between High-Dimensional Geometry and Adversarial Examples. *arXiv:1801.02774*.
31. Simon-Gabriel CJ, Ollivier Y, Bottou L, Schölkopf B, Lopez-Paz D. 2019 First-Order Adversarial Vulnerability of Neural Networks and Input Dimension. In Chaudhuri K, Salakhutdinov R, editors, *Proceedings of the 36th International Conference on Machine Learning* vol. 97*Proceedings of Machine Learning Research* pp. 5809–5817. PMLR.
32. Tyukin IY, Higham DJ, Gorban AN. 2020 On adversarial examples and stealth attacks in artificial intelligence systems. In *2020 International Joint Conference on Neural Networks* pp. 1–6. IEEE.
33. 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. (https://doi.org/10.1016/j.ins.2018.07.040)
34. Sutton OJ, Zhou Q, Tyukin IY, Gorban AN, Bastounis A, Higham DJ. 2024 How adversarial attacks can disrupt seemingly stable accurate classifiers. *Neural Networks* **180**, 106711. (https://doi.org/10.1016/j.neunet.2024.106711)
35. Bartlett P, Bubeck S, Cherapanamjeri Y. 2021 Adversarial Examples in Multi-Layer Random ReLU Networks. In *Advances in Neural Information Processing Systems*.
36. Tsipras D, Santurkar S, Engstrom L, Turner A, Madry A. 2019 Robustness May Be at Odds with Accuracy. In *International Conference on Learning Representations*.
37. Bastounis A, Gorban AN, Hansen AC, Higham DJ, Prokhorov D, Sutton O, Tyukin IY, Zhou Q. 2023 The Boundaries of Verifiable Accuracy, Robustness, and Generalisation in Deep Learning. In Iliadis L, Papaleonidas A, Angelov P, Jayne C, editors, *Artificial Neural Networks and Machine Learning – ICANN 2023* pp. 530–541 Cham. Springer Nature Switzerland.
38. Bastounis A, Hansen AC, Vlaĉić V. to appear The mathematics of adversarial attacks in AI– Why deep learning is unstable despite the existence of stable neural networks. *European Journal of Applied Mathematics*.
39. Tyukin IY, Higham DJ, Bastounis A, Woldegeorgis E, Gorban AN. 2023 The feasibility and inevitability of stealth attacks. *IMA Journal of Applied Mathematics*.
40. Gorban A, Grechuk B, Tyuxkin I. 2022 Stochastic Separation Theorems: How Geometry May Help to Correct AI Errors. *Notices Amer. Math. Soc.* **70**, 25–33. (10.1090/noti2599)
41. Shan S, Ding W, Passananti J, Wu S, Zheng H, Zhao BY. 2024 Nightshade: Prompt-specific poisoning attacks on text-to-image generative models. In *2024 IEEE Symposium on*

*Security and Privacy (SP)* pp. 807–825 Los Alamitos, CA, USA. IEEE Computer Society. (10.1109/SP54263.2024.00207)
42. Shan S, Cryan J, Wenger E, Zheng H, Hanocka R, Zhao BY. 2023 Glaze: protecting artists from style mimicry by text-to-image models. In *Proceedings of the 32nd USENIX Conference on Security Symposium* SEC '23 USA. USENIX Association.
43. Shan S, Wenger E, Zhang J, Li H, Zheng H, Zhao BY. 2020 Fawkes: Protecting Personal Privacy against Unauthorized Deep Learning Models. In *Proc. of USENIX Security*.
44. Wang J, Wang H, Zhang J, Wu H, Luo X, Ma B. 2024 Invisible Adversarial Watermarking: A Novel Security Mechanism for Enhancing Copyright Protection. *ACM Trans. Multimedia Comput. Commun. Appl.* **21**. (10.1145/3652608)
45. Yao Y, Jain AK, Liu S. 2024 Adversarial Watermarking for Face Recognition. In *The Third Workshop on New Frontiers in Adversarial Machine Learning*.
46. Sutton OJ, Zhou Q, Wang W, Higham DJ, Gorban AN, Bastounis A, Tyukin IY. 2024 Stealth edits to large language models. In Globerson A, Mackey L, Belgrave D, Fan A, Paquet U, Tomczak J, Zhang C, editors, *Advances in Neural Information Processing Systems* vol. 37 pp. 51811–51844. Curran Associates, Inc.
47. MATLAB. 2022 *version 9.13.0.2080170 (R2022b)*. Natick, Massachusetts: The MathWorks Inc.