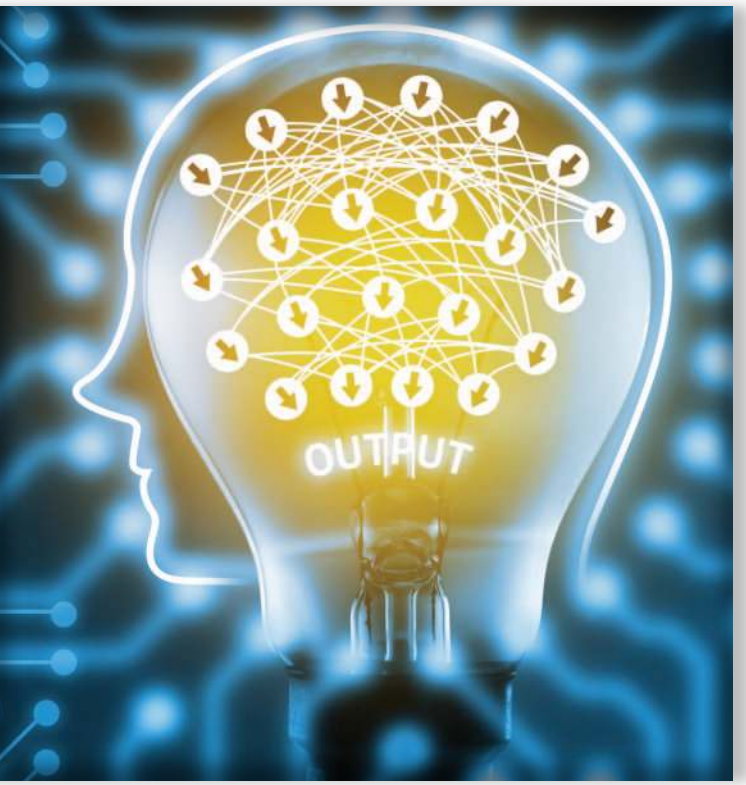
Alhussein Fawzi, Seyed-Mohsen Moosavi-Dezfooli,
and Pascal Frossard

The Robustness of Deep Networks

A geometrical perspective



Digital Object Identifier 10.1109/MSP.2017.2740965
Date of publication: 13 November 2017

Deep neural networks have recently shown impressive classification performance on a diverse set of visual tasks. When deployed in real-world (noise-prone) environments, it is equally important that these classifiers satisfy robustness guarantees: small perturbations applied to the samples should not yield significant loss to the performance of the predictor. The goal of this article is to discuss the robustness of deep networks to a diverse set of perturbations that may affect the samples in practice, including adversarial perturbations, random noise, and geometric transformations. This article further discusses the recent works that build on the robustness analysis to provide geometric insights on the classifier's decision surface, which help in developing a better understanding of deep networks. Finally, we present recent solutions that attempt to increase the robustness of deep networks. We hope this review article will contribute to shed ding light on the open research challenges in the robustness of deep networks and stir interest in the analysis of their fundamental properties.

Introduction

With the dramatic increase of digital data and the development of new computing architectures, deep learning has been developing rapidly as a predominant framework for data representation that can contribute in solving very diverse tasks. Despite this success, several fundamental properties of deep neural networks are still not understood and have been the subject of intense analysis in recent years. In particular, the robustness of deep networks to various forms of perturbations has received growing attention due to its importance when applied to visual data. That path of work has been mostly initiated by the illustration of the intriguing properties of deep networks in [1], which are shown to be particularly vulnerable to very small additive perturbations in the data, even if they achieve impressive performance on complex visual benchmarks [2]. An illustration of the vulnerability of deep networks to small additive perturbations can be seen in Figure 1. A dual phenomenon was observed in [3], where unrecognizable images to the human eye are classified with high confidence by deep neural

 1053-5888/17©2017IEEE

networks. The transfer of these deep networks to critical applications that possibly consist in classifying high-stake information is seriously challenged by the low robustness of deep networks. For example, in the context of self-driving vehicles, it is fundamental to accurately recognize cars, traffic signs, and pedestrians, when these are affected by clutter, occlusions, or even adversarial attacks. In medical imaging [4], it is also important to achieve high classification rates on potentially perturbed test data. The analysis of state-of-the-art deep classifiers' robustness to perturbation at test time is therefore an important step for validating the models' reliability to unexpected (possibly adversarial) nuisances that might occur when deployed in uncontrolled environments. In addition, a better understanding of the capabilities of deep networks in coping with data perturbation actually allows us to develop important insights that can contribute to developing yet better systems.

The fundamental challenges raised by the robustness of deep networks to perturbations have led to a large number of important works in recent years. These works study empirically and theoretically the robustness of deep networks to different types of perturbations, such as adversarial perturbations, additive random noise, structured transformations, or even universal perturbations. The robustness is usually measured as the sensitivity of the discrete classification function (i.e., the function that assigns a label to each image) to such perturbations. While robustness analysis is not a new problem, we provide an overview of the recent works that propose to assess the vulnerability of deep network architectures. In addition to quantifying the robustness of deep networks to various forms of perturbations, the analysis of robustness has further contributed to developing important insights on the geometry of the complex decision boundary of such classifiers, which remain hardly understood due to the very high dimensionality of the problems that they address. In fact, the robustness properties of a classifier are strongly tied to the geometry of the decision boundaries. For example, the high instability of deep neural networks to adversarial perturbations shows that data points reside extremely close to the classifier's decision boundary. The study of robustness is, therefore, not only interesting from the practical perspective of the system's reliability but has a more fundamental component that allows "understanding" of the geometric properties of classification regions and derives insights toward the improvement of current architectures.

This overview article has multiple goals. First, it provides an accessible review of the recent works in the analysis of the robustness of deep neural network classifiers to different forms of perturbations, with a particular emphasis on image analysis and visual understanding applications. Second, it presents connections between the robustness of deep networks and the geometry of the decision boundaries of such classifiers. Third, the article discusses ways to improve the robustness in deep networks architectures and finally highlights some of the important open problems.

Robustness of classifiers

In most classification settings, the proportion of misclassified samples in the test set is the main performance metric used to evaluate classifiers. The empirical test error provides an estimate of the classifier's risk, defined as the probability of misclassification, when considering samples from the data distribution. Formally, let us define μ to be a distribution defined over images. The risk of a classifier f is equal to

$$R(f) = \mathbb{P}_{x\sim\mu}(f(x) \neq y(x)), \tag{1}$$

where x and $y(x)$ correspond, respectively, to the image and its associated label. While the risk captures the error of f on the data distribution μ, it does not capture the robustness to small arbitrary perturbations of data points. In visual classification tasks, it is desirable to learn classifiers that achieve robustness to small perturbations of the input; i.e., the application of a small perturbation to images (e.g., additive perturbations on the pixel values or geometric transformation of the image) should not alter the estimated label of the classifier.

Before going into more detail about robustness, we first define some notations. Let $\mathcal{X}$ denote the ambient space where images live. We denote by $\mathcal{R}$ the set of admissible perturbations. For example, when considering geometric perturbations, $\mathcal{R}$ is set to be the group of geometric (e.g., affine) transformations under study. Alternatively, if we are to measure the robustness to arbitrary additive perturbations, we set $\mathcal{R} = \mathcal{X}$. For $r \in \mathcal{R}$, we define $T_r: \mathcal{X} \to \mathcal{X}$ to be the perturbation operator by r; i.e., for a data point $x \in \mathcal{X}$, $T_r(x)$ denotes the image x perturbed by r. Armed with these notations, we define the minimal perturbation changing the label of the classifier, at x, as follows:

$$r^*(x) = \underset{r\in\mathcal{R}}{\operatorname{argmin}} \| r \|_{\mathcal{R}} \text{ subject to } f(T_r(x)) \neq f(x), \tag{2}$$

where $\| \cdot \|_{\mathcal{R}}$ is a metric on $\mathcal{R}$. For notation simplicity, we omit the dependence of $r^*(x)$ on f, $\mathcal{R}$, δ, and operator T. Moreover, when the image x is clear from the context, we will use r^* to refer to $r^*(x)$. See Figure 2 for an illustration. The pointwise robustness of f at x is then measured by $\| r^*(x) \|_{\mathcal{R}}$. Note that larger values of $\| \cdot \|_{\mathcal{R}}$ indicate a higher robustness at x. While this definition of robustness considers the smallest perturbation $r^*(x)$ (with respect to the metric $\| \cdot \|_{\mathcal{R}}$) that causes the classifier f to

FIGURE 1. An example of an adversarial perturbations in state-of-the-art neural networks. (a) The original image that is classified as a "whale," (b) the perturbed image classified as a "turtle," and (c) the corresponding adversarial perturbation that has been added to the original image to fool a state-of-the-art image classifier [5].

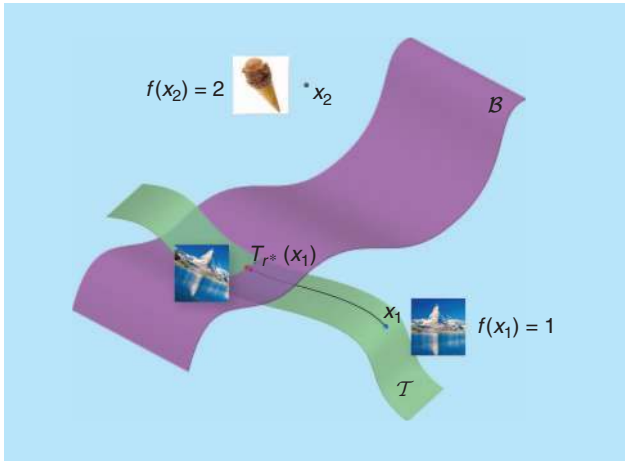


FIGURE 2. Here, $\mathcal{B}$ denotes the decision boundary of the classifier between classes 1 and 2, and $\mathcal{T}$ denotes the set of perturbed versions of x_1 (i.e., $\mathcal{T} = \{T_r(x_1): r \in \mathcal{R}\}$), where we recall that $\mathcal{R}$ denotes the set of admissible perturbations. The pointwise robustness at x_1 is defined as the smallest perturbation in $\mathcal{R}$ that causes x_1 to change class.

change the label at x, other works have instead adopted slightly different definitions, where a "sufficiently small" perturbation is sought (instead of the minimal one) [7]–[9]. To measure the global robustness of a classifier f, one can compute the expectation of $\|r^*(x)\|_{\mathcal{R}}$ over the data distribution [1], [10]. That is, the global robustness $\rho(f)$ is defined as follows:

$$\rho(f) = \mathop{\mathbb{E}}_{x\sim\mu}(\|r^*(x)\|_{\mathcal{R}}). \tag{3}$$

It is important to note that in our robustness setting, the perturbed point $T_r(x)$ need not belong to the support of the data distribution. Hence, while the focus of the risk in (1) is the accuracy on typical images (sampled from μ), the focus of the robustness computed from (2) is instead on the distance to the "closest" image (potentially outside the support of μ) that changes the label of the classifier. The risk and robustness hence capture two fundamentally different properties of the classifier, as illustrated in "Robustness and Risk: A Toy Example."

Robustness and Risk: A Toy Example

To illustrate the general concepts of robustness and risk of classifiers, we consider the simple binary classification task illustrated in Figure S1, where the goal is to discriminate between images representing vertical and horizontal stripes. In addition to the orientation of the stripe that separates the two classes, a very small positive bias is added to pixels of first-class images and subtracted from the pixels of the images in the second class. This bias is chosen to be very small, in such a way that it is imperceptible to humans.; see Figure S2 for example images of class 1 and 2 with the pixel values, where a denotes the bias.

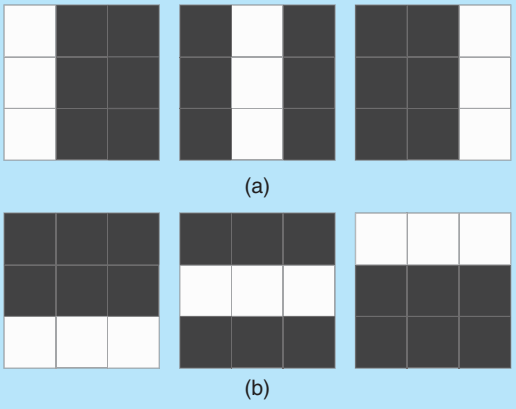


FIGURE S1. (a) The images belonging to class 1 (vertical stripe and positive bias) and (b) the images belonging to class 2 (horizontal stripe and negative bias).

It is easy to see that a linear classifier can perfectly separate the two classes, thus achieving zero risk (i.e., $R(f) = 0$). Note, however, that such a classifier only achieves zero risk because it captures the bias but fails to distinguish between the images based on the orientation of the stripe. Hence, despite being zero risk, this classifier is highly unstable to additive perturbation, as it suffices to perturb the bias of the image (i.e., by adding a very small value to all pixels) to cause misclassification. On the other hand, a more complex classifier that captures the orientation of the stripe will be robust to small perturbations (while equally achieving zero risk), as changing the label would require changing the direction of the stripe, which is the most visual (and natural) concept that separates the two classes.

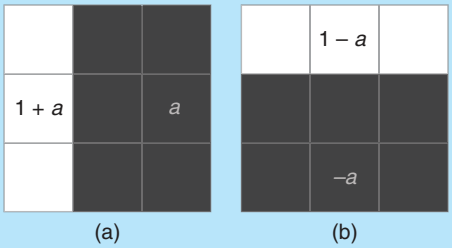


FIGURE S2. (a) An example image of class 1. White pixels have value $1 + a$, and black pixels have value a. (b) An example image of class –1. White pixels have value $1 - a$, and black pixels have value $-a$. The bias a is set to be very small, in such a way that it is imperceptible.

Observe that classification robustness is strongly related to support vector machine (SVM) classifiers, whose goal is to maximize the robustness, defined as the margin between support vectors. Importantly, the max-margin classifier in a given family of classifiers might, however, still not achieve robustness (in the sense of high $\rho(f)$). An illustration is provided in "Robustness and Risk: A Toy Example," where a no zero-risk linear classifier—in particular, the max-margin classifier—achieves robustness to perturbations. Our focus in this article is turned toward assessing the robustness of the family of deep neural network classifiers that are used in many visual recognition tasks.

Perturbation forms

Robustness to additive perturbations

We first start by considering the case where the perturbation operator is simply additive; i.e., $T_r(x) = x + r$. In this case, the magnitude of the perturbation can be measured with the ℓ_p norm of the minimal perturbation that is necessary to change the label of a classifier. According to (2), the robustness to additive perturbations of a data point x is defined as

$$\min_{r \in \mathcal{R}} \| r \|_p \text{ subject to } f(x + r) \neq f(x). \tag{4}$$

Depending on the conditions that one sets on the set $\mathcal{R}$ that supports the perturbations, the additive model leads to different forms of robustness.

Adversarial perturbations

We first consider the case where the additive perturbations are unconstrained (i.e., $\mathcal{R} = \mathcal{X}$). The perturbation obtained by solving (4) is often referred to as an adversarial perturbation, as it corresponds to the perturbation that an adversary (having full knowledge of the model) would apply to change the label of the classifier, while causing minimal changes to the original image.

The optimization problem in (4) is nonconvex, as the constraint involves the (potentially highly complex) classification function f. Different techniques exist to approximate adversarial perturbations. In the following, we briefly mention some of the existing algorithms for computing adversarial perturbations:

- Regularized variant [1]: The method in [1] computes adversarial perturbations by solving a regularized variant of the problem in (4), given by

$$\min_r c \| r \|_p + J(x + r, \tilde{y}, \theta), \tag{5}$$

where $\tilde{y}$ is a target label of the perturbed sample, J is a loss function, c is a regularization parameter, and θ is the model parameters. In the original formulation [1], an additional constraint is added to guarantee $x + r \in [0, 1]$, which is omitted in (5) for simplicity. To solve the optimization problem in (5), a line search is performed over c to find the maximum $c > 0$ for which the minimizer of (5) satisfies $f(x + r) = \tilde{y}$. While leading to very accurate estimates, this approach can be costly to compute on high-dimensional and large-scale data sets. Moreover, it computes targeted adversarial perturbations, where the target label is known.
- Fast gradient sign (FGS) [11]: This solution estimates an untargeted adversarial perturbation by going in the direction of the sign of gradient of the loss function:

$$\epsilon \operatorname{sign}(\nabla_x J(x, y(x), \theta)),$$

where J, the loss function, is used to train the neural network and θ denotes the model parameters. While efficient, this one-step algorithm provides a coarse approximation to the solution of the optimization problem in (4) for $p = \infty$.
- DeepFool [5]: This algorithm minimizes (4) through an iterative procedure, where each iteration involves the linearization of the constraint. The linearized (constrained) problem is solved in closed form at each iteration, and the current estimate is updated; the optimization procedure terminates when the current estimate of the perturbation fools the classifier. In practice, DeepFool provides a tradeoff between the accuracy and efficiency of the two previous approaches [5].

In addition to the aforementioned optimization methods, several other approaches have recently been proposed to compute adversarial perturbations, see, e.g., [9], [12], and [13]. Different from the previously mentioned gradient-based techniques, the recent work in [14] learns a network (the adversarial transformation network) to efficiently generate a set of perturbations with a large diversity, without requiring the computation of the gradients.

Using the aforementioned optimization techniques, one can compute the robustness of classifiers to additive adversarial perturbations. Quite surprisingly, deep networks are extremely vulnerable to such additive perturbations; i.e., small and even imperceptible adversarial perturbations can be computed to fool them with high probability. For example, the average perturbations required to fool the CaffeNet [15] and GoogleNet [16] architectures on the ILSVRC 2012 task [17] are 100 times smaller than the typical norm of natural images [5] when using the ℓ_2 norm. The high instability of deep neural networks to adversarial perturbations, which was first highlighted in [1], shows that these networks rely heavily on proxy concepts to classify objects, as opposed to strong visual concepts typically used by humans to distinguish between objects.

To illustrate this idea, we consider once again the toy classification example (see "Robustness and Risk: A Toy Example"), where the goal is to classify images based on the orientation of the stripe. In this example, linear classifiers could achieve a perfect recognition rate by exploiting the imperceptibly small bias that separates the two classes. While this proxy concept achieves zero risk, it is not robust to perturbations: one could design an additive perturbation that is as simple as a minor variation of the bias, which is sufficient to induce data misclassification. On the same line of thought, the high instability of classifiers to additive perturbations observed in [1] suggests that deep neural networks potentially capture one of the proxy concepts that separate the different classes. Through a quantitative analysis of polynomial

classifiers, [10] suggests that higher-degree classifiers tend to be more robust to perturbations, as they capture the "stronger" (and more visual) concept that separates the classes (e.g., the orientation of the stripe in Figure S1 in "Robustness and Risk: A Toy Example"). For neural networks, however, the relation between the flexibility of the architecture (e.g., depth and breadth) and adversarial robustness is not well understood and remains an open problem.

With the dramatic increase of digital data and the development of new computing architectures, deep learning has been developing rapidly as a predominant framework for data representation that can contribute in solving very diverse tasks.

Random noise

In the random noise regime, data points are perturbed by noise having a random direction in the input space. Unlike the adversarial case, the computation of random noise does not require knowledge of the classifier; it is therefore crucial for state-of-the-art classifiers to be robust to this noise regime. We measure the pointwise robustness to random noise by setting $\mathcal{R}$ to be a direction sampled uniformly at random from the ℓ_2 unit sphere $\mathbb{S}^{d-1}$ in $\mathcal{X}$ (where d denotes the dimension of $\mathcal{X}$). Therefore, (4) becomes

$$r_v^*(x) = \underset{r\in\{av:a\in\mathbb{R}\}}{\operatorname{argmin}} \| r \|_2 \text{ subject to } f(x+r) \neq f(x), \tag{6}$$

where v is a direction sampled uniformly at random from the unit sphere $\mathbb{S}^{d-1}$. The pointwise robustness is then defined as the ℓ_2 norm of the perturbation, i.e., $\| r_v^*(x) \|_2$.

The robustness of classifiers to random noise has previously been studied empirically in [1] and theoretically in [10] and [18]. Empirical investigation suggests that state-of-the-art classifiers are much more robust to random noise than to adversarial perturbations, i.e., the norm of the noise $r_v^*(x)$ required to change the label of the classifier can be several orders of magnitudes larger than that of the adversarial perturbation. This result is confirmed theoretically, as linear classifiers in [10] and nonlinear classifiers in [18] are shown to have a robustness to random noise that behaves as

$$\| r_v^*(x) \|_2 = \Theta\left(\sqrt{d}\, \| r_{\text{adv}}^*(x) \|_2\right)$$

with high probability, where $\| r_{\text{adv}}^*(x) \|_2$ denotes the robustness to adversarial perturbations [(4) with $\mathcal{R} = \mathcal{X}$]. In other words, this result shows that, in high-dimensional classification settings (i.e., large d), classifiers can be robust to random noise, even if the pointwise adversarial robustness of the classifier is very small.

Semirandom noise

Finally, the semirandom noise regime generalizes this additive noise model to random subspaces $\mathcal{S}$ of dimension $m \leq d$. Specifically, in this perturbation regime, an adversarial perturbation is sought within a random subspace $\mathcal{S}$ of dimension m. That is, the semirandom noise is defined as follows:

$$r_{\mathcal{S}}^*(x) = \underset{r\in\mathcal{S}}{\operatorname{argmin}} \| r \|_2 \text{ subject to } f(x+r) \neq f(x). \tag{7}$$

Note that, when $m = 1$, this semirandom noise regime precisely coincides with the random noise regime, whereas $m = d$ corresponds to the adversarial perturbation regime defined previously. For this generalized noise regime, a precise relation between the robustness to semirandom and adversarial perturbation exists [18], as it is shown that

$$\| r_{\mathcal{S}}^*(x) \|_2 = \Theta\left(\sqrt{\frac{d}{m}}\, \| r_{\text{adv}}^*(x) \|_2\right).$$

This result shows in particular that, even when the dimension m is chosen as a small fraction of d, it is still possible to find small perturbations that cause data misclassification. In other words, classifiers are not robust to semirandom noise that is only mildly adversarial and overwhelmingly random [18]. This implies that deep networks can be fooled by very diverse small perturbations, as these can be found along random subspaces of dimension $m \ll d$.

Robustness to structured transformations

In visual tasks, it is not only crucial to have classifiers that are robust against additive perturbations as described previously. It is also equally important to achieve invariance to structured nuisance variables such as illumination changes, occlusions, or standard local geometric transformations of the image. Specifically, when images undergo such structured deformations, it is desirable that the estimated label remains the same.

One of the main strengths of deep neural network classifiers with respect to traditional shallow classifiers is that the former achieve higher levels of invariance [19] to transformations. To verify this claim, several empirical works have been introduced. In [6], a formal method is proposed that leverages the generalized robustness definition of (2) to measure the robustness of classifiers to arbitrary transformation groups. The robustness to structured transformations is precisely measured by setting the admissible perturbation space $\mathcal{R}$ to be the set of transformations (e.g., translations, rotations, dilation) and the perturbation operator T of (2) to be the warping operator transforming the coordinates of the image. In addition, $\| \cdot \|_{\mathcal{R}}$ is set to measure the change in appearance between the original and transformed images. Specifically, $\| \cdot \|_{\mathcal{R}}$ is defined to be the length of the shortest path on the nonlinear manifold of transformed images $\mathcal{T} = \{T_r(x): r \in \mathcal{R}\}$. Using this approach, it is possible to quantify the amount of change that the image should undergo to cause the classifier to make the wrong decision. Despite improving the invariance over shallow networks, the method in [6] shows that deep classifiers are still not robust to sufficiently small deformations on simple visual classification tasks. In [20], the authors assess the robustness of face recognition deep networks to physically realizable structured perturbations. In particular, wearing eyeglass frames is shown to cause state-of-the-art face-recognition algorithms to misclassify. In [7], the robustness to other

forms of complex perturbations is tested, and state-of-the-art deep networks are shown once again to be unstable to these perturbations. An empirical analysis of the ability of current convolutional neural networks (CNNs) to manage location and scale variability is proposed in [21]. It is shown, in particular, that CNNs are not very effective in factoring out location and scale variability, despite the popular belief that the convolutional architecture and the local spatial pooling provides invariance to such representations. The aforementioned works show that, just as state-of-the-art deep neural networks have been observed to be unstable to additive unstructured perturbations, such modern classifiers are not robust to perturbations even when severely restricting the set of possible transformations of the image.

Universal additive perturbations

All of the previous definitions capture different forms of robustness, but they all rely on the computation of data-specific perturbations. Specifically, they consider the necessary change that should be applied to specific samples to change the decision of the classifier. More generally, one might be interested to understand if classifiers are also vulnerable to generic (data and network agnostic) perturbations. The analysis of the robustness to such perturbations is interesting from several perspectives: 1) these perturbations might not require the precise knowledge of the classifier under test, 2) they might capture important security and reliability properties of classifiers, and 3) they show important properties on the geometry of the decision boundary of the classifier.

In [22], deep networks are shown to be surprisingly vulnerable to universal (image-agnostic) perturbations. Specifically, a universal perturbation v can be defined as the minimal perturbation that fools a large fraction of the data points sampled from the data distribution μ, i.e.,

$$v = \underset{r}{\text{argmin}} \| r \|_p \text{ subject to } \underset{x\sim\mu}{\mathbb{P}}(f(x+r) \neq f(x)) \geq 1-\epsilon, \quad (8)$$

where ϵ controls the fooling rate of the universal perturbation. Unlike adversarial perturbations that target to fool a specific data point, universal perturbations attempt to fool most images sampled from the natural images distribution μ. Specifically, by adding this single (image-agnostic) perturbation to a natural image, the label estimated by the deep neural network will be changed with high probability. In [22], an algorithm is provided to compute such universal perturbations; these perturbations are further shown to be quasi-imperceptible while fooling state-of-the-art deep networks on unseen natural images with probability edging 80%. Specifically, the ℓ_p norm of these perturbations is at least one order of magnitude smaller than the norm of natural images but causes most perturbed images to be misclassified. Figure 3 illustrates examples of scaled universal

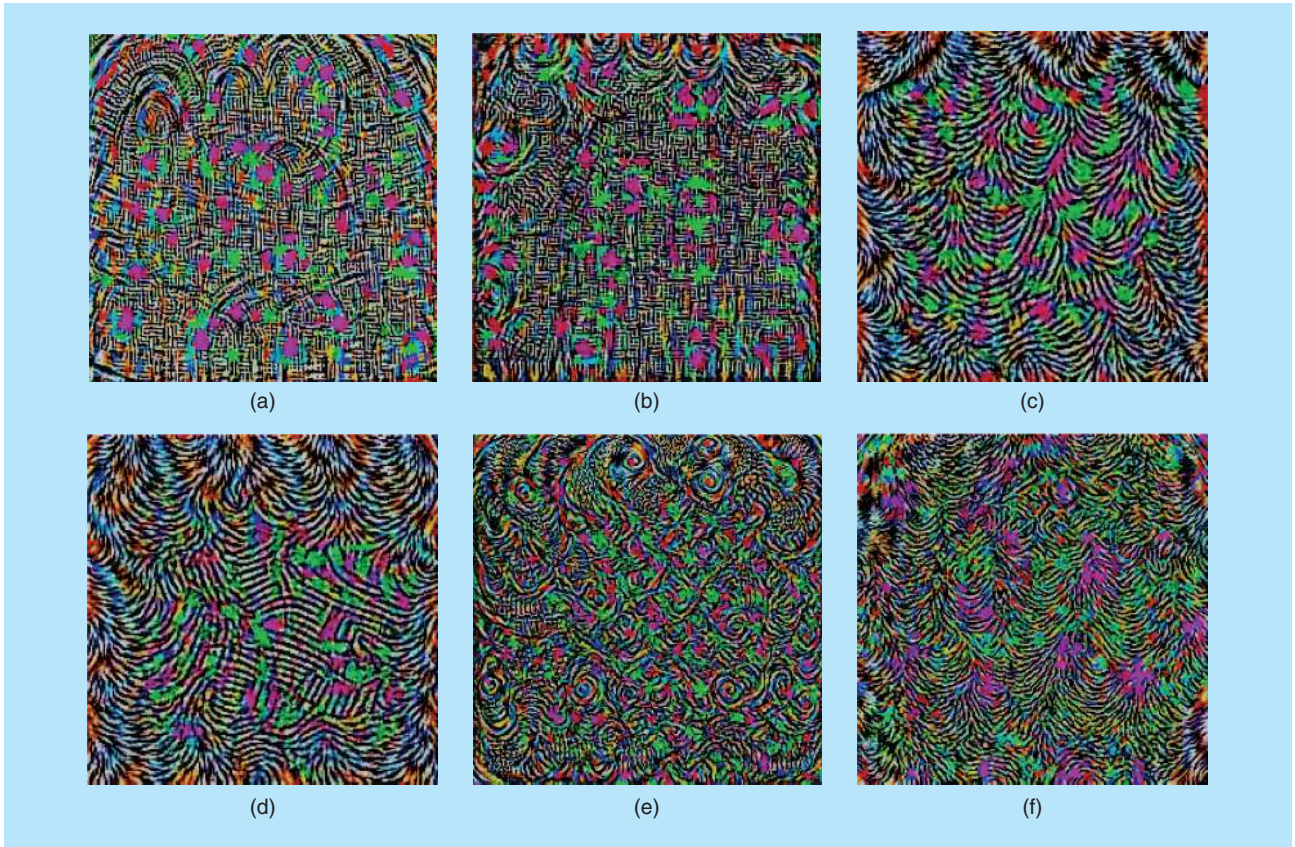


FIGURE 3. Universal perturbations computed for different deep neural network architectures. The pixel values are scaled for visibility. (a) CaffeNet, (b) VGG-F, (c) VGG-16, (d) VGG-19, (e) GoogLeNet, and (f) ResNet-152.

perturbations computed for different deep neural networks, and Figure 4 illustrates examples of perturbed images. When added to the original images, a universal perturbation is quasi-imperceptible but causes most images to be misclassified. Note that adversarial perturbations computed using the algorithms described in the section "Adversarial Perturbations" are not universal across data points, as shown in [22]. That is, adversarial perturbations only generalize mildly to unseen data points, for a fixed norm comparable to that of universal perturbations.

Universal perturbations are further shown in [22] to transfer well across different architectures; a perturbation computed for a given network is also very likely to fool another network on most natural images. In that sense, such perturbations are doubly universal, as they generalize well across images and architectures. Note that this property is shared with adversarial perturbations, as the latter perturbations have been shown to transfer well across different models (with potentially different architectures) [1], [23]. The existence of general-purpose perturbations can be very problematic from a safety perspective, as an attacker might need very little information about the actual model to craft successful perturbations [24].

Figure 5 illustrates a summary of the different types of perturbations considered in this section on a sample image. As can be seen, the classifier is not robust to slight perturbations of the image (for most additive perturbations) and natural geometric transformations of the image.

Geometric insights from robustness

The study of robustness allows us to derive insights about the classifiers and, more precisely, about the geometry of the classification function acting on the high-dimensional input space. We recall that $f: \mathcal{X} \to \{1, \ldots, C\}$ denotes our C-class classifier, and we denote by $g_1, \ldots, g_C$ the C probabilities associated to each class by the classifier. Specifically, for a given $x \in \mathcal{X}$, $f(x)$ is assigned to the class having a maximal score; i.e., $f(x) = \text{argmax}_i \{g_i(x)\}$. For deep neural networks, the functions g_i represent the outputs of the last layer in the network (generally the softmax layer). Note that the classifier f can be seen as a mapping that partitions the input space $\mathcal{X}$ into classification regions, each of which has a constant

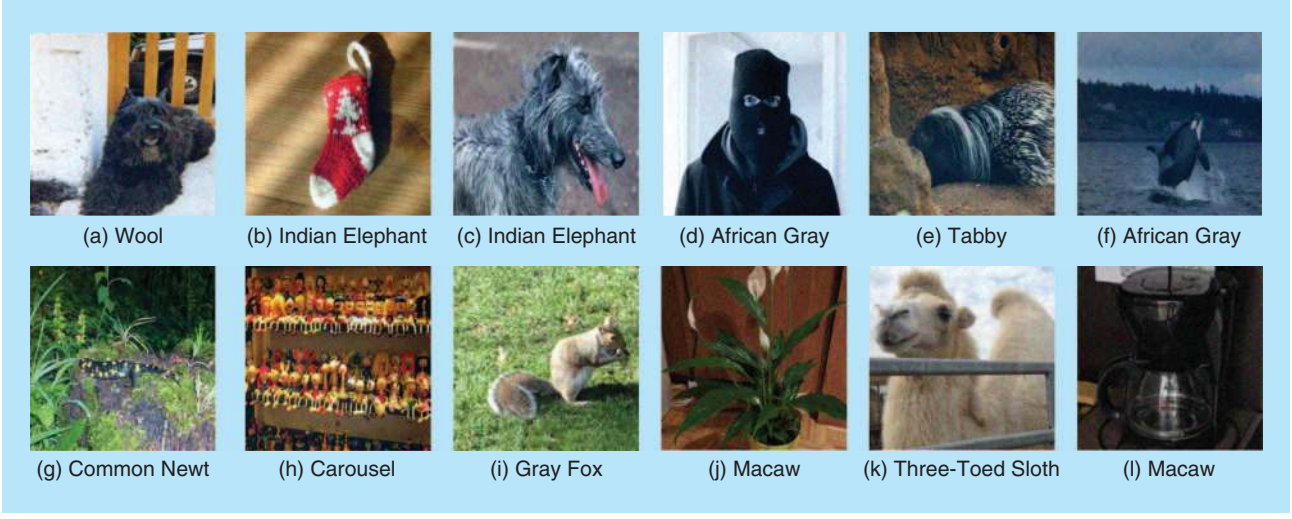


FIGURE 4. Examples of natural images perturbed with the universal perturbation and their corresponding estimated labels with GoogLeNet. (a)–(h) Images belonging to the ILSVRC 2012 validation set. (i)–(l) Personal images captured by a mobile phone camera. (Figure used courtesy of [22].)

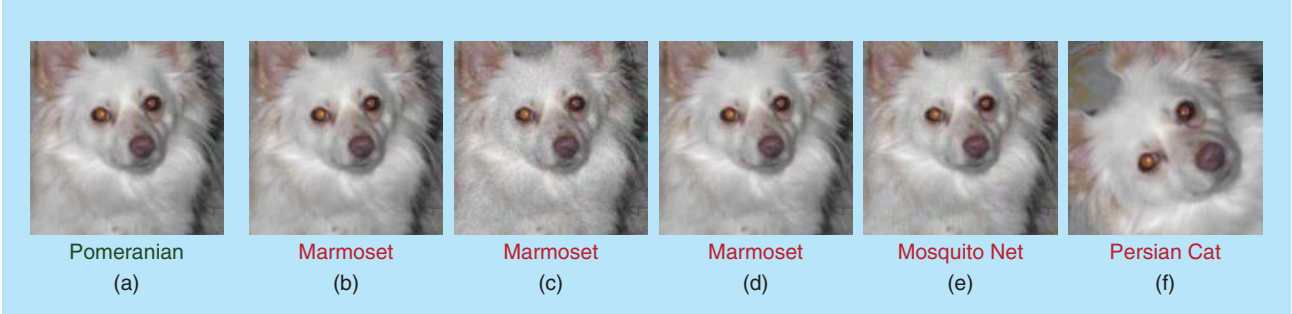


FIGURE 5. (a) The original image. The remaining images are minimally perturbed images (along with the corresponding estimated label) that misclassify the CaffeNet deep neural network. (b) Adversarial perturbation, (c) random noise, (d) semirandom noise with $m = 1{,}000$, (e) universal perturbation, (f) affine transformation. (Figure used courtesy of [17].)

estimated label (i.e., $f(x)$ is constant for each such region). The decision boundary $\mathcal{B}$ of the classifier is defined as the union of the boundaries of such classification regions (see Figure 2).

Adversarial perturbations

We first focus on additive adversarial perturbations and highlight their relation with the geometry of the decision boundary. This link relies on the simple observation shown in "Geometric Properties of Adversarial Perturbations." The two geometric properties are illustrated in Figure 6. Note that these geometric properties are specific to the ℓ_2 norm. The high instability of classifiers to adversarial perturbations, which we highlighted in the previous section, shows that natural images lie very closely to the classifier's decision boundary. While this result is key to understanding the geometry of the data points with regard to the classifier's decision boundary, it does not provide any insights on the shape of the decision boundary. A local geometric description of the decision boundary (in the vicinity of x) is rather captured by the direction of $r^*_{\text{adv}}(x)$, due to the orthogonality property of adversarial perturbations (highlighted in "Geometric Properties of Adversarial Perturbations"). In [18] and [25], these geometric properties of adversarial perturbations are leveraged to visualize typical cross sections of the decision boundary at the vicinity of the data points. Specifically, a two-dimensional normal section of the decision boundary is illustrated, where the sectioning plane is spanned by the adversarial perturbation (normal to the decision boundary) and a random vector in the tangent space. Examples of normal sections of decision boundaries are illustrated in Figure 7.

Observe that the decision boundaries of state-of-the-art deep neural networks have a very low curvature on these two-dimensional cross sections (note the difference between the x and y axis). In other words, these plots suggest that the decision boundary at the vicinity of x can be locally well

Geometric Properties of Adversarial Perturbations

Observation

Let $x \in \mathcal{X}$ and $r^*_{\text{adv}}(x)$ be the adversarial perturbation, defined as the minimizer of (4), with $p = 2$ and $\mathcal{R} = \mathcal{X}$. Then, we have the following:

1) $\| r^*_{\text{adv}}(x) \|_2$ measures the Euclidean distance from x to the closest point on the decision boundary $\mathcal{B}$.
2) The vector $r^*_{\text{adv}}(x)$ is orthogonal to the decision boundary of the classifier, at $x + r^*_{\text{adv}}(x)$.

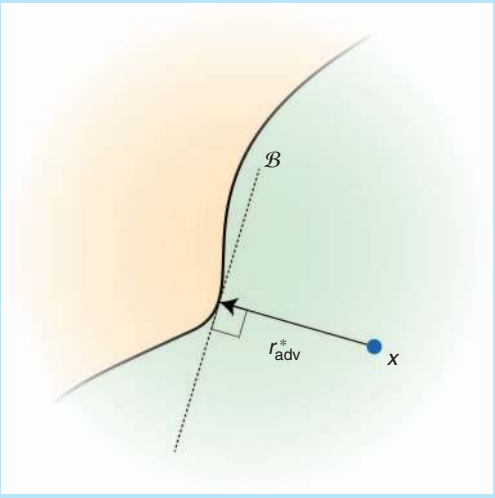


FIGURE 6. r^*_{adv} denotes the adversarial perturbation of x (with $p = 2$). Note that r^*_{adv} is orthogonal to the decision boundary $\mathcal{B}$ and $\| r^*_{\text{adv}} \|_2 = \text{dist}(x, \mathcal{B})$.

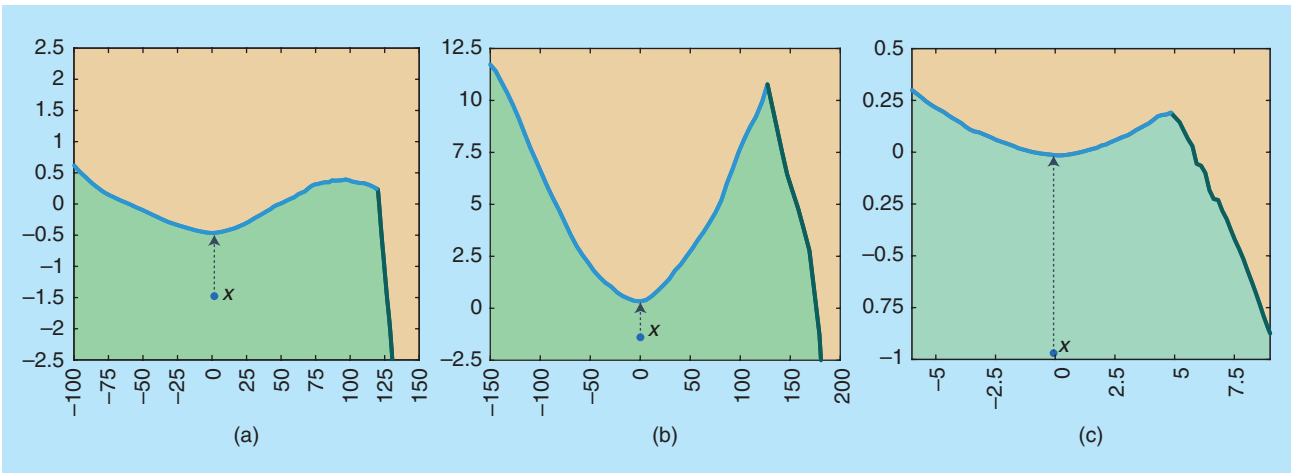


FIGURE 7. The two-dimensional normal cross sections of the decision boundaries for three different classifiers near randomly chosen samples. The section is spanned by the adversarial perturbation of the data point x (vertical axis) and a random vector in the tangent space to the decision boundary (horizontal axis). The green region is the classification region of x. The decision boundaries with different classes are illustrated in different colors. Note the difference in range between the x and y axes. (a) VGG-F (ImageNet), (b) LeNet (CIFAR), (c) LeNet (MNIST). (Figure used with permission from [18].)

approximated by a hyperplane passing through $x + r^*_{\text{adv}}(x)$ with the normal vector $r^*_{\text{adv}}(x)$. In [11], it is hypothesized that state-of-the-art classifiers are "too linear," leading to decision boundaries with very small curvature and further explaining the high instability of such classifiers to adversarial perturbations. To motivate the linearity hypothesis of deep networks, the success of the FGS method (which is exact for linear classifiers) in finding adversarial perturbations is invoked. However, some recent works challenge this linearity hypothesis; for example, in [26], the authors show that there exist adversarial perturbations that cannot be explained with this hypothesis, and, in [27], the authors provide a new explanation based on the tilting of the decision boundary with respect to the data manifold. We stress here that the low curvature of the decision boundary does not, in general, imply that the function learned by the deep neural network (as a function of the input image) is linear, or even approximately linear. Figure 8 shows illustrative examples of highly nonlinear functions resulting in flat decision boundaries. Moreover, it should be noted that, while the decision boundary of deep networks is very flat on random two-dimensional cross sections, these boundaries are not flat on all cross sections. That is, there exist directions in which the boundary is very curved. Figure 9 provides some illustrations of such cross sections, where the decision boundary has large curvature and therefore significantly departs from the first-order linear approximation, suggested by the flatness of the decision boundary on random sections in Figure 7. Hence, these visualizations of the decision boundary strongly suggest that the curvature along a small set of directions can be very large and that the curvature is relatively small along random directions in the input space. Using a numerical computation of the curvature, the sparsity of the curvature profile is empirically verified in [28] for deep neural networks, and the directions where the decision boundary is curved are further shown to play a major role in explaining the robustness properties of classifiers. In [29], the authors provide a complementary analysis on the curvature of the decision boundaries induced by deep networks and show that the first principal curvatures increase exponentially with the depth of a random neural network. The analyses of [28] and [29] hence suggest that the curvature profile of deep networks is highly sparse (i.e., the decision boundaries are almost flat along most directions) but can have a very large curvature along a few directions.

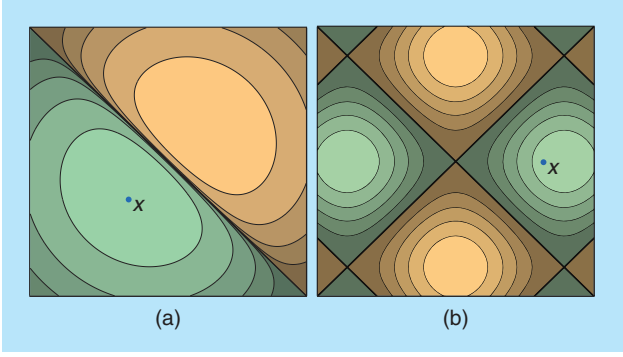


FIGURE 8. The contours of two highly nonlinear functions (a) and (b) with flat boundaries. Specifically, the contours in the green and yellow regions represent the different (positive and negative) level sets of $g(x)$ [where $g(x) = g_1(x) - g_2(x)$, the difference between class 1 and class 2 score]. The decision boundary is defined as the region of the space where $g(x) = 0$ and is indicated with a solid black line. Note that, although g is a highly nonlinear function in these examples, the decision boundaries are flat.

Universal perturbations

The vulnerability of deep neural networks to universal (image-agnostic) perturbations studied in [22] sheds light on another aspect of the decision boundary: the correlations between different regions of the decision boundary, in the vicinity of different natural images. In fact, if the orientations of the decision boundary in the neighborhood of different data points were uncorrelated, the best universal perturbation would correspond to a random perturbation. This is refuted in [22], as the norm of the random perturbation required to fool 90% of the images is ten times larger than the norm of universal perturbations. Such correlations in the decision boundary are quantified in [22], as it is shown empirically that normal vectors to the decision boundary at the vicinity of different data points (or, equivalently, adversarial perturbations due to the orthogonality property in "Geometric Properties of Adversarial Perturbations") approximately span a low-dimensional

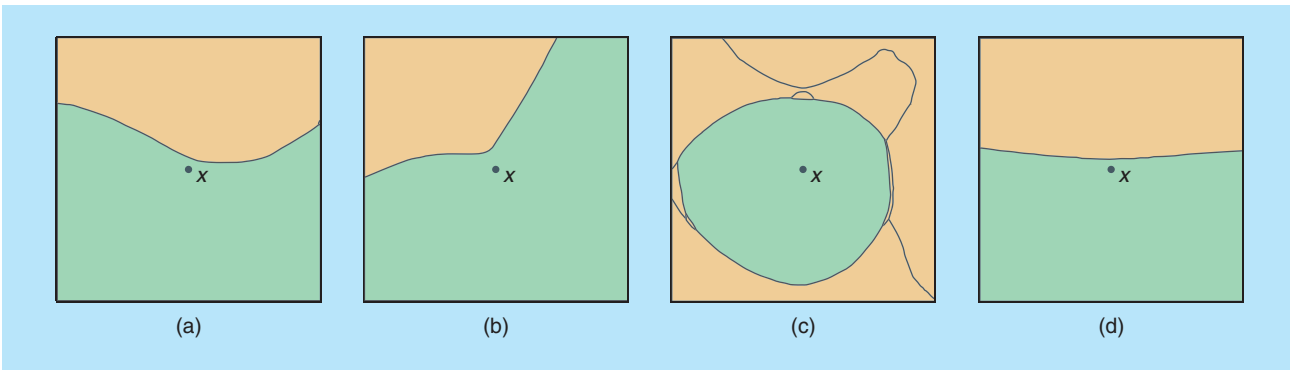


FIGURE 9. Cross sections of the decision boundary in the vicinity of data point x. (a), (b), and (c) show decision boundaries with high curvature, while (d) shows the decision boundary along a random normal section (with very small curvature). The correct class and the neighboring classes are colored in green and orange, respectively. The boundaries between different classes are shown in solid black lines. The x and y axes have the same scale.

subspace. It is conjectured that the existence of universal perturbations fooling classifiers for most natural images is partly due to the existence of such a low-dimensional subspace that captures the correlations among different regions of the decision boundary. In fact, this subspace "collects" normals to the decision boundary in different regions, and perturbations belonging to this subspace are therefore likely to fool other data points. This observation implies that the decision boundaries created by deep neural networks are not sufficiently "diverse," despite the very large number of parameters in modern deep neural networks.

A more thorough analysis is provided in [30], where universal perturbations are shown to be tightly related to the curvature of the decision boundary in the vicinity of data points. Specifically, the existence of universal perturbations is attributed to the existence of common directions where the decision boundary is positively curved in the vicinity of most natural images. Figure 10 intuitively illustrates the link between positive curvature and vulnerability to perturbations; the required perturbation to change the label (along a fixed direction v) of the classifier is smaller if the decision boundary is positively curved, than if the decision boundary is flat (or negatively curved). With this geometric perspective, universal perturbations correspond exactly to directions where the decision boundary is positively curved in the vicinity of most natural images. As shown in [30], this geometric explanation of universal perturbations suggests a new algorithm to compute such perturbations as well as to explain several properties, such as the diversity and transferability of universal perturbations.

The study of robustness allows us to derive insights about the classifiers and, more precisely, about the geometry of the classification function acting on the high-dimensional input space.

Classification regions

The robustness of classifiers is not only related to the geometry of the decision boundary, but it is also strongly tied to the classification regions in the input space $\mathcal{X}$. The classification region associated to class $c \in \{1, \ldots, C\}$ corresponds to the set of points $x \in \mathcal{X}$ such that $f(x) = c$. The study of universal perturbations in [22] has shown the existence of dominant labels, with universal perturbations mostly fooling natural images into such labels. The existence of such dominant classes is attributed to the large volumes of classification regions corresponding to dominant labels in the input space $\mathcal{X}$: in fact, images sampled uniformly at random from the Euclidean sphere $\alpha\mathbb{S}^{d-1}$ of the input space $\mathcal{X}$ (where the radius α is set to reflect the typical norm of natural images) are classified as one of these dominant labels. Hence, such dominant labels represent high-volume "oceans" in the image space; universal perturbations therefore tend to fool images into such target labels, as these generally result in smaller fooling perturbations. It should be noted that these dominant labels are classifier specific and are not a result of the visual properties of the images in the class.

To further understand the geometrical properties of classification regions, we note that, just like natural images, random images are strongly vulnerable to adversarial perturbations. That is, the norm of the smallest adversarial perturbation needed to change the label of a random image (sampled from $\mathcal{X}$) is several orders of magnitude smaller than the norm of the image itself. This observation suggests that classification regions are "hollow" and that most of their mass occurs at the boundaries. In [28], further topological properties of classification regions are observed; in particular, these regions are shown empirically to be connected. In other words, each classification region in the input space $\mathcal{X}$ is made up of a single connected (possibly complex) region, rather than several disconnected regions.

We have discussed in this section that the properties and optimization methods derived to analyze the robustness properties of classifiers allow us to derive insights on the geometry of the classifier. In particular, through visualizations, we have seen that the decision boundaries on normal random sections have very low curvature, while being very curved along a few directions of the input space. Moreover, the high vulnerability of state-of-the-art deep networks to universal perturbations suggests that the decision boundaries of such networks do not have sufficient diversity. To improve the robustness to such perturbations, it is therefore key to "diversify" the decision boundaries of the network and leverage the large number of parameters that define the neural network.

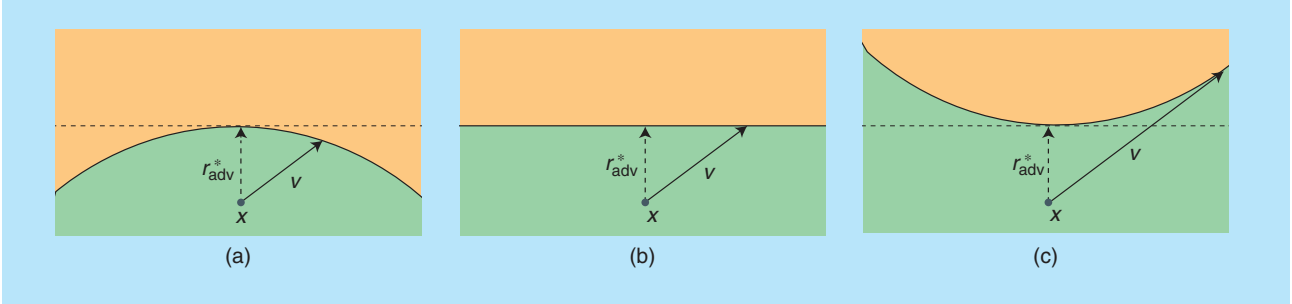


FIGURE 10. The link between robustness and curvature of the decision boundary. When the decision boundary is (a) positively curved, small universal perturbations are more likely to fool the classifier. (b) and (c) illustrate the case of a flat and negatively curved decision boundary, respectively.

Improving robustness

An important objective of the analysis of robustness is to contribute to the design of better and more reliable systems. We next summarize some of the recent attempts that have been made to render systems more robust to different forms of perturbations.

Improving the robustness to adversarial perturbations

We first describe the methods that have been proposed to construct deep networks with better robustness to adversarial perturbations, following the papers [1], [9] that originally highlighted the vulnerability of these classifiers. The straightforward approach, which consists of adding perturbed images to the training set and fine-tuning the network, has been shown to be mildly effective against newly computed adversarial perturbations [5]. To further improve the robustness, it is natural to consider the Jacobian matrix $\partial g/\partial x$ of the model (with g the last layer of the neural network) and ensure that all of the elements in the matrix are sufficiently small. Following this idea, the authors of [31] consider a modified objective function, where a term is added to penalize the Jacobians of the function computed by each layer with respect to the previous layer. This has the effect of learning smooth functions with respect to the input and thus learn more robust classifiers. In [32], a robust optimization formulation is considered for training deep neural networks. Specifically, a minimization-maximization approach is proposed, where the loss is minimized over worst-case examples, rather than only on the original data. That is, the following minimization-maximization training procedure is used to train the network:

$$\min_{\theta} \sum_{i=1}^{N} \max_{r_i \in \mathcal{U}} J(x_i + r_i, y_i, \theta), \tag{9}$$

where θ, N, and $\mathcal{U}$ denote, respectively, the parameters of the network, the number of training points, and the set of plausible perturbations; and y_i denotes the label of x_i. The set $\mathcal{U}$ is generally set to be the ℓ_2 or ℓ_∞ ball centered at zero and of sufficiently small radius. Unfortunately, this optimization problem in (9) is difficult to solve efficiently. To circumvent this difficulty, [32] proposes an alternating iterative method where a single step of gradient ascent and descent is performed at each iteration. Note that the construction of robust classifiers using min-max robust optimization methods has been an active area of research, especially in the context of SVM classifiers [33]. In particular, for certain sets $\mathcal{U}$, the objective function of various learning tasks can be written as a convex optimization function as shown in [34]–[37], which makes the task of finding a robust classifier feasible. In a very recent work inspired by biophysical principles of neural circuits, Nayebi and Ganguli consider a regularizer to push activations of the network in the saturating regime of the nonlinearity (i.e., the region where the nonlinear activation function is flat) [47]. The networks learned using this approach are shown to significantly improve in terms of robustness on a simple digit recognition classification task, without losing significantly in terms of accuracy. In [38], the authors propose to improve the robustness by using distillation, a technique first introduced in [39] for transferring knowledge from larger architectures to smaller ones. However, [40] shows that, when using more elaborate algorithms to compute perturbations, this approach fails to improve the robustness. In [41], a regularization scheme is introduced for improving the network's sensitivity to perturbations by constraining the Lipschitz constant of the network. In [42], an information-theoretic loss function is used to train stochastic neural networks; the resulting classifiers are shown to be more robust to adversarial perturbations than their deterministic counterpart. The increased robustness is intuitively due to the randomness of the neural network, which maps an input to a distribution of features; attacking the network with a small designed perturbation therefore becomes harder than for deterministic neural networks.

The importance of analyzing the vulnerability of deep neural networks to perturbations therefore goes beyond the practical security implications, as it further reveals crucial geometric properties of deep networks.

While all of these methods are shown to yield some improvements on the robustness of deep neural networks, the design of robust visual classifiers on challenging classification tasks (e.g., ImageNet) is still an open problem. Moreover, while the previously mentioned methods provide empirical results showing the improvement in robustness with respect to one or a subset of adversarial generation techniques, it is necessary in many applications to design robust networks against all adversarial attacks. To do so, we believe it is crucial to derive formal certificates on the robustness of newly proposed networks, as it is practically impossible to test against all possible attacks, and we see this as an important future work in this area.

Although there is currently no method to effectively (and provably) combat adversarial perturbations on large-scale data sets, several studies [42]–[44] have recently considered the related problem of detectability of adversarial perturbations. The detectability property is essential in real-world applications, as it allows the possibility to raise an exception when tampered images are detected. In [42], the authors propose to augment the network with a detector network, which detects original images from perturbed ones. Using the optimization methods in the section "Adversarial Perturbations," the authors conclude that the network successfully learns to distinguish between perturbed samples and original samples. Moreover, the overall network (i.e., the network and detector) is shown to be more robust to adversarial perturbations tailored for this architecture. In [43], the Bayesian uncertainty estimates in the subspace of learned representations are used to discriminate perturbed images from clean samples. Finally, as shown in [44], side

information such as depth maps can be exploited to detect adversarial samples.

Improving the robustness to geometric perturbations

Just as in the case of adversarial perturbations, one popular way of building more invariant representations to geometric perturbations is through virtual jittering (or data augmentation), where training data are transformed and fed back to the training set. One of the drawbacks of this approach is, however, that the training can become intractable, as the size of the training set becomes substantially larger than the original data set. In another effort to improve the invariance properties of deep CNNs, the authors in [45] proposed a new module, the spatial transformer, that geometrically transforms the filter maps. Similarly to other modules in the network, spatial transformer modules are trained in a purely supervised fashion. Using spatial transformer networks, the performance of classifiers improves significantly, especially when images have noise and clutter, as these modules automatically learn to localize and unwarp corrupted images. To build robust deep representations, [46] considers instead a new architecture with fixed filter weights. Specifically, a similar structure to CNNs (i.e., cascade of filtering, nonlinearity, and pooling operations) is considered with the additional requirement of stability of the representation to local deformations, while retaining maximum information about the original data. The scattering network is proposed, where successive filtering with wavelets and pointwise nonlinearities is applied and further shown to satisfy the stability constraints. Note that the approach used to build this scattering network significantly differs from traditional CNNs, as no learning of the filters is involved. It should further be noted that while scattering transforms guarantee that representations built by deep neural networks are robust to small changes in the input, this does not imply that the overall classification pipeline (feature representation and discrete classification) is robust to small perturbations in the input, in the sense of (2). We believe that building deep architectures with provable guarantees on the robustness of the overall classification function is a fundamental open problem in the area.

One of the main strengths of deep neural network classifiers with respect to traditional shallow classifiers is that the former achieve higher levels of invariance to transformations.

Summary and open problems

The robustness of deep neural networks to perturbations is a fundamental requirement in a large number of practical applications involving critical prediction problems. We discussed in this article the robustness of deep networks to different forms of perturbations: adversarial perturbations, random noise, universal perturbations, and geometric transformations. We further highlighted close connections between the robustness to additive perturbations and geometric properties of the classifier's decision boundary (such as the curvature). The importance of analyzing the vulnerability of deep neural networks to perturbations therefore goes beyond the practical security implications, as it further reveals crucial geometric properties of deep networks. We hope that this close relation between robustness and geometry will continue to be leveraged to design more robust systems.

Despite the recent and insightful advances in the analysis of the vulnerability of deep neural networks, several challenges remain:

- It is known that deep networks are vulnerable to universal perturbations due to the existence of correlations between different parts of the decision boundary. Yet, little is known about the elementary operations in the architecture (or learned weights) of a deep network that cause the classifier to be sensitive to such directions.
- Similarly, the causes underlying the transferability of adversarial perturbations across different architectures are still not understood formally.
- While the classifier's decision boundary has been shown to have a very small curvature when sectioned by random normal planes, it is still unclear whether this property of the decision boundary is due to the optimization method (i.e., stochastic gradient descent) or rather to the use of piecewise linear activation functions.
- While natural images have been shown to lie very close to the decision boundary, it is still unclear whether there exist points that lie far away from the decision boundary.

Finally, one of the main goals of the analysis of robustness is to propose architectures with increased robustness to additive and structured perturbations. This is probably one of the fundamental problems that needs special attention from the community in the years to come.

Authors

Alhussein Fawzi (fawzi@cs.ucla.edu) received the M.S. and Ph.D. degrees in electrical engineering from the Swiss Federal Institute of Technology, Lausanne, in 2012 and 2016, respectively. He is now a postdoctoral researcher in the Computer Science Department at the University of California, Los Angeles. He received the IBM Ph.D. fellowship in 2013 and 2015. His research interests include signal processing, machine learning, and computer vision.

Seyed-Mohsen Moosavi-Dezfooli (seyed.moosavi@epfl.ch) received the B.S. degree in electrical engineering from Amirkabir University of Technology (Tehran Polytechnic), Iran, in 2012 and the M.S. degree in communication systems from the École Polytechnique Fédérale de Lausanne (EPFL), Switzerland, in 2014. Currently, he is a Ph.D. degree student in the Signal Processing Laboratory 4 at EPFL under the supervision of Prof. Pascal Frossard. Previously, he was a research assistant in the Audiovisual Communications Laboratory at EPFL. During the spring and the summer of

2014, he was a research intern with ABB Corporate Research, Baden-Daettwil. His research interests include signal processing, machine learning, and computer vision.

Pascal Frossard (pascal.frossard@epfl.ch) received the M.S. and Ph.D. degrees in electrical engineering from the École Polytechnique Fédérale de Lausanne (EPFL), Switzerland, in 1997 and 2000, respectively. From 2001 to 2003, he was a member of the research staff with the IBM T.J. Watson Research Center, Yorktown Heights, New York, where he was involved in media coding and streaming technologies. Since 2003, he has been a faculty member at EPFL, where he is currently the head of the Signal Processing Laboratory. His research interests include signal processing on graphs and networks, image representation and coding, visual information analysis, and machine learning.

References

[1] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *Proc. Int. Conf. Learning Representations*, 2014.

[2] A Krizhevsky, I Sutskever, and G E Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[3] A Nguyen, J Yosinski, and J Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2015, pp. 427–436.

[4] G. Litjens, T. Kooi, B. Ehteshami Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. W. M. van der Laak, B. van Ginneken and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Med. Image Anal.*, vol. 42, pp. 60–88, 2017.

[5] S-M Moosavi-Dezfooli, A Fawzi, and P Frossard, "Deepfool: A simple and accurate method to fool deep neural networks," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2016, pp. 2574–2582.

[6] A. Fawzi and P. Frossard, "Manitest: Are classifiers really invariant?" in *Proc. British Machine Vision Conf.*, 2015, pp. 106.1–106.13.

[7] A. Fawzi and P. Frossard, "Measuring the effect of nuisance variables on classifiers," in *Proc. British Machine Vision Conf.*, 2016, pp. 137.1–137.12.

[8] A Fawzi, H Samulowitz, D Turaga, and P Frossard, "Adaptive data augmentation for image classification," in *Proc. Int. Conf. Image Processing*, 2016, pp. 3688–3692.

[9] B. Biggio, I. Corona, D. Maiorca, B. Nelson, N. Srndic, P. Laskov, G. Giacinto, and F. Roli, "Evasion attacks against machine learning at test time," in *Proc. Joint European Conf. Machine Learning and Knowledge Discovery in Databases*, 2013, pp. 387–402.

[10] A. Fawzi, O. Fawzi, and P. Frossard, "Analysis of classifiers' robustness to adversarial perturbations," *Machine Learning*, Aug. 2017. [Online]. Available: https://doi.org/10.1007/s10994-017-5663-3

[11] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. Int. Conf. Learning Representations*, 2015.

[12] A. Rozsa, E. M. Rudd, and T. E. Boult, "Adversarial diversity and hard positive generation," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops*, 2016, pp. 25–32.

[13] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," *arXiv Preprint*, arXiv:1608.04644, 2016.

[14] S. Baluja and I. Fischer, "Adversarial transformation networks: Learning to generate adversarial examples," *arXiv Preprint*, arXiv:1703.09387, 2017.

[15] Y Jia, E Shelhamer, J Donahue, S Karayev, J Long, R Girshick, S Guadarrama, and T Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. ACM Int. Conf. Multimedia*, 2014, pp. 675–678.

[16] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2015.

[17] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *Int. J. Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[18] A Fawzi, S. Moosavi-Dezfooli, and P. Frossard, "Robustness of classifiers: from adversarial to random noise," in *Proc. Neural Information Processing Systems Conf.*, 2016, pp. 1632–1640.

[19] H. Larochelle, D. Erhan, A. Courville, J. Bergstra, and Y. Bengio, "An empirical evaluation of deep architectures on problems with many factors of variation," in *ACM Int. Conf. Machine Learning*, 2007, pp. 473–480.

[20] M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter, "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition," in *Proc. 2016 ACM SIGSAC Conf. Computer and Communications Security*, 2016, pp. 1528–1540.

[21] N Karianakis, J Dong, and S Soatto, "An empirical evaluation of current convolutional architectures ability to manage nuisance location and scale variability," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2016, pp. 4442–4451.

[22] S.-M. Moosavi-Dezfooli, A Fawzi, O Fawzi, and P Frossard, "Universal adversarial perturbations," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2017.

[23] Y. Liu, X. Chen, C. Liu, and D. Song, "Delving into transferable adversarial examples and black-box attacks," *arXiv Preprint*, arXiv:1611.02770, 2016.

[24] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. Berkay Celik, and A. Swami, "Practical black-box attacks against deep learning systems using adversarial examples," *arXiv Preprint*, arXiv:1602.02697, 2016.

[25] D. Warde-Farley, I. Goodfellow, T. Hazan, G. Papandreou, and D. Tarlow, "Adversarial perturbations of deep neural networks," in *Perturbations, Optimization, and Statistics*. Cambridge, MA: MIT Press, 2016.

[26] S. Sabour, Y. Cao, F. Faghri, and D. J. Fleet, "Adversarial manipulation of deep representations," in *Proc. Int. Conf. Learning Representations*, 2016.

[27] T. Tanay and L. Griffin, "A boundary tilting persepective on the phenomenon of adversarial examples," *arXiv Preprint*, arXiv:1608.07690, 2016.

[28] A. Fawzi, S.-M. Moosavi-Dezfooli, P. Frossard, and S. Soatto, "Classification regions of deep neural networks," *arXiv Preprint*, arXiv:1705.09552, 2017.

[29] B. Poole, S. Lahiri, M. Raghu, J. Sohl-Dickstein, and S. Ganguli, "Exponential expressivity in deep neural networks through transient chaos," in *Proc. Advances in Neural Information Processing Systems Conf.*, 2016, pp. 3360–3368.

[30] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, P. Frossard, and S. Soatto, "Analysis of universal adversarial perturbations," *arXiv Preprint*, arXiv:1705.09554, 2017.

[31] S. Gu and L. Rigazio, "Towards deep neural network architectures robust to adversarial examples," *arXiv Preprint*, arXiv:1412.5068, 2014.

[32] U. Shaham, Y. Yamada, and S. Negahban, "Understanding adversarial training: Increasing local stability of neural nets through robust optimization," *arXiv Preprint*, arXiv:1511.05432, 2015.

[33] C. Caramanis, S. Mannor, and H. Xu, "Robust optimization in machine learning," in *Optimization for Machine Learning*, S. Suvrit, N. Sebastian, and W. J. Stephen, Eds., Cambridge, MA: MIT Press, ch. 14, 2012.

[34] H. Xu, C. Caramanis, and S. Mannor, "Robustness and regularization of support vector machines," *J. Machine Learning Res.*, vol. 10, pp. 1485–1510, July 2009.

[35] G. Lanckriet, L. E. Ghaoui, C. Bhattacharyya, and M. I. Jordan, "A robust minimax approach to classification," *J. Machine Learning Res.*, vol. 3, pp. 555–582, Dec. 2003.

[36] C Bhattacharyya, "Robust classification of noisy data using second order cone programming approach," in *Proc. Intelligent Sensing and Information Processing Conf.*, 2004, pp. 433–438.

[37] T. B. Trafalis and R. C. Gilbert, "Robust support vector machines for classification and computational issues," *Optim. Methods Software*, vol. 22, no. 1, pp. 187–198, 2007.

[38] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, "Distillation as a defense to adversarial perturbations against deep neural networks," in *Proc. 2016 IEEE Symp. Security and Privacy*, 2016, pp. 582–597.

[39] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv Preprint*, arXiv:1503.02531, 2015.

[40] N. Carlini and D. Wagner, "Defensive distillation is not robust to adversarial examples," *arXiv Preprint*, arXiv:1607.04311, 2016.

[41] A. A. Alemi, I. Fischer, J. V. Dillon, and K. Murphy, "Deep variational information bottleneck," *arXiv Preprint*, arXiv:1612.00410, 2016.

[42] J. H. Metzen, T. Genewein, V. Fischer, and B. Bischoff, "On detecting adversarial perturbations," *arXiv Preprint*, arXiv:1702.04267, 2017.

[43] R. Feinman, R. R. Curtin, S. Shintre, and A. B. Gardner, "Detecting adversarial samples from artifacts," *arXiv Preprint*, arXiv:1703.00410, 2017.

[44] J. Lu, T. Issaranon, and D. Forsyth, "Safetynet: Detecting and rejecting adversarial examples robustly," *arXiv Preprint*, arXiv:1704.00103, 2017.

[45] M. Jaderberg, K. Simonyan, and A. Zisserman, "Spatial transformer networks," in *Proc. Advances in Neural Information Processing Systems Conf.*, 2015, pp. 2017–2025.

[46] A. Nayebi and S. Ganguli, "Biologically inspired protection of deep networks from adversarial attacks," *arXiv Preprint*, arXiv:1703.09202, 2017.

[47] M. Cisse, A Courville, P. Bojanowski, and E. Grave, Y. Dauphin, and N. Usunier "Parseval networks: Improving robustness to adversarial examples," in *Proc. Int. Conf. Machine Learning*, 2017, pp. 854–863.

SP