The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)

展示interpretation的不稳定性

Interpretation of Neural Networks Is Fragile

Amirata Ghorbani,* **Abubakar Abid,*** **James Zou**
Stanford University
450 Serra Mall, Stanford, CA, USA
{amiratag, a12d, jamesz}@stanford.edu

Abstract

In order for machine learning to be trusted in many applications, it is critical to be able to reliably explain why the machine learning algorithm makes certain predictions. For this reason, a variety of methods have been developed recently to interpret neural network predictions by providing, for example, feature importance maps. For both scientific robustness and security reasons, it is important to know to what extent can the interpretations be altered by small systematic perturbations to the input data, which might be generated by adversaries or by measurement biases. In this paper, we demonstrate how to generate adversarial perturbations that produce perceptively indistinguishable inputs that are assigned the *same* predicted label, yet have very *different* interpretations. We systematically characterize the robustness of interpretations generated by several widely-used feature importance interpretation methods (feature importance maps, integrated gradients, and DeepLIFT) on ImageNet and CIFAR-10. In all cases, our experiments show that systematic perturbations can lead to dramatically different interpretations without changing the label. We extend these results to show that interpretations based on exemplars (e.g. influence functions) are similarly susceptible to adversarial attack. Our analysis of the geometry of the Hessian matrix gives insight on why robustness is a general challenge to current interpretation approaches.

Introduction

Predictions made by machine learning algorithms play an important role in our everyday lives and can affect decisions in technology, medicine, and even the legal system (Rich 2015; Obermeyer and Emanuel 2016). As algorithms become increasingly complex, explanations for why an algorithm makes certain decisions are ever more crucial. For example, if an AI system predicts a given pathology image to be malignant, then a doctor may need to know what features in the image led the algorithm to this classification. Similarly, if an algorithm predicts an individual to be a credit risk, then the lender (and the borrower) might want to know why. Therefore having interpretations for why certain predictions are made is critical for establishing trust and transparency between users and the algorithm (Lipton 2016).

Having an interpretation is not enough, however. The explanation itself must be robust in order to establish human trust. Take the pathology predictor; an interpretation method might suggest that a particular section in an image is important for the malignant classification (e.g. that section could have high scores in saliency map). The clinician might then focus on that section for investigation or treatment or even look for similar features in other patients. It would be highly disconcerting if in an extremely similar image, visually indistinguishable from the original and also classified as malignant, a very different section is interpreted as being salient for the prediction. Thus, even if the predictor is robust (both images are correctly labeled as malignant), that the interpretation is fragile would still be problematic in deployment. Furthermore, if the interpretation is used to guide interventions (e.g. location of a biopsy) by the doctor, then an interpretation that is not robust against adversarial perturbations may prove to be a security concern.

Our contributions. It is well known that the predicted *labels* of deep neural networks are susceptible to adversarial attacks (Goodfellow, Shlens, and Szegedy 2014; Kurakin, Goodfellow, and Bengio 2016; Papernot et al. 2016; Moosavi-Dezfooli, Fawzi, and Frossard 2016). In this paper, we introduce the notion of adversarial perturbations to neural network interpretation. More precisely, we define the interpretation of neural network to be *fragile* if, for a given image, it is possible to generate a perceptively indistinguishable image that has the same prediction label by the neural network, yet is given a substantially different interpretation. We systematically investigate two classes of interpretation methods: methods that assign importance scores to each feature (this includes simple gradients (Simonyan, Vedaldi, and Zisserman 2013), DeepLift (Shrikumar, Greenside, and Kundaje 2017), and integrated gradients (Sundararajan, Taly, and Yan 2017)), as well as a method that assigns importances to each training example: influence functions (Koh and Liang 2017). For these interpretation methods, we show how to design targeted perturbations that can lead to dramatically different interpretations across test images (Fig. 1). Our findings highlight the fragility of interpretations of neural networks, which has not been carefully considered in the literature. Fragility limits how much we can trust and learn from the interpretations. It also raises a

*Equal Contribution

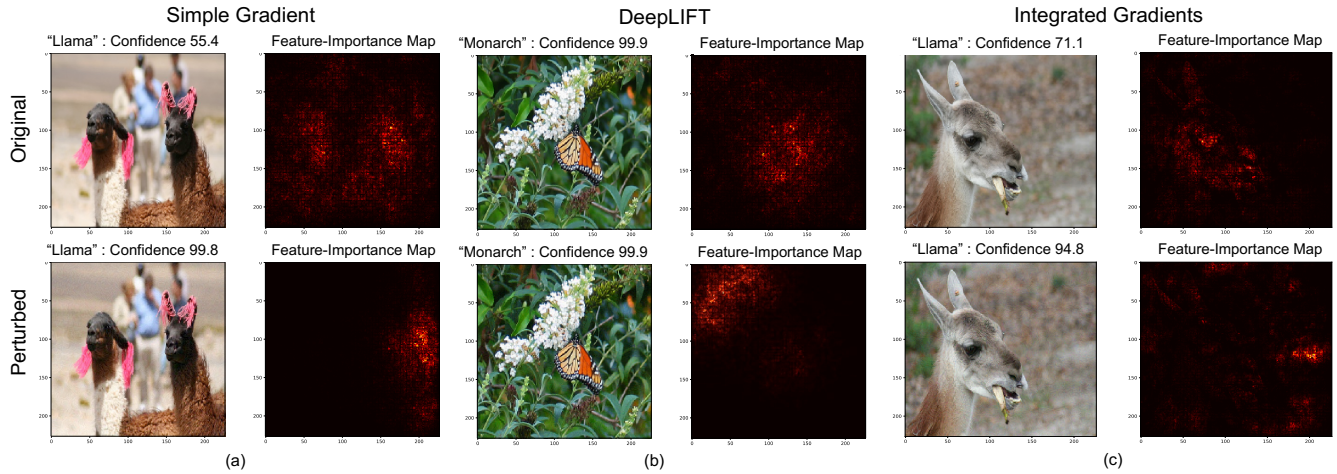


Figure 1: **Adversarial attack against feature-importance maps.** We generate feature-importance scores, also called saliency maps, using three popular interpretation methods: (a) simple gradients, (b) DeepLIFT, and (c) integrated gradients. The **top row** shows the the original images and their saliency maps and the **bottom row** shows the perturbed images (using the center attack with $\epsilon = 8$, as described in Section 2) and corresponding saliency maps. In all three images, the predicted label does not change from the perturbation; however, the saliency maps of the perturbed images shifts dramatically to features that would not be considered salient by human perception.

significant new security concern. Especially in medical or economic applications, users often take the interpretation of a prediction as containing causal insight (*"this image is a malignant tumor because of the section with a high saliency score"*). An adversary could minutely manipulate the input to draw attention away from relevant features or onto his/her desired features. Such attacks might be especially hard to detect as the actual labels have not changed.

While we focus on image data here because most interpretation methods have been motivated by images, the fragility of neural network interpretation could be a much broader problem. Fig. 2 illustrates the intuition that when the decision boundary in the input feature space is complex, as is the case with deep networks, a small perturbation in the input can push the example into a region with very different loss contours. Because the feature importance is closely related to the gradient which is perpendicular to the loss contours, the importance scores can also be dramatically different. We provide additional analysis of this in Section 4.

1 Related Works and Preliminaries

Related works (Szegedy et al. 2013) first demonstrated the possibility of fooling neural networks into making different predictions for test images that are visually indistinguishable. (Goodfellow, Shlens, and Szegedy 2014) introduced the one-step Fast Gradient Sign Method (FGSM) prediction attack which was followed by more effective iterative attacks (Kurakin, Goodfellow, and Bengio 2016). To quantify the perturbation size, metrics such as ℓ_2 (Moosavi-Dezfooli, Fawzi, and Frossard 2016; Szegedy et al. 2013), ℓ_0 (number of perturbed pixels) (Papernot et al. 2016), and ℓ_∞ (Goodfellow, Shlens, and Szegedy 2014) have been introduced. As it tightly controls how much individual input features can change, we followed the popular practice and adopted ℓ_∞. There has also been a line of work showing that networks that are robust against adversarial attacks to its predictions also have improved interpretability (Ross and Doshi-Velez 2017; Dong et al. 2017). The focus of all of these works is on adversarial attacks against the *prediction*; in contrast our work focuses on attacks on the *interpretation* without changing the prediction.

Interpretation methods for neural networks Interpretation of neural network predictions is an active research area. Post-hoc interpretability (Lipton 2016) is one family of methods that seek to "explain" the prediction without considering the details of black-box model's hidden mechanisms. These include methods to explain predictions in terms of the features of the test example, as well as in terms of the contribution of training examples to the test time prediction. These interpretations have gained increasing popularity, as they confer a degree of insight to human users of what the neural network might be doing (Lipton 2016). We describe several widely-used interpretation methods in what follows.

Feature importance interpretation This first class of methods explains predictions in terms of the relative importance of features of an input test sample. Given the sample $\boldsymbol{x}_t \in \mathbb{R}^d$ and the network's prediction l, we define the score of the predicted class $S_l(\boldsymbol{x}_t)$ to be the value of the pre-softmax layer's l-th neuron. We take l to be the class with the highest score; i.e. the predicted class. feature importance methods seek to find the dimensions of input data point that strongly affect the score, and in doing so, these methods assign an absolute feature importance score to each

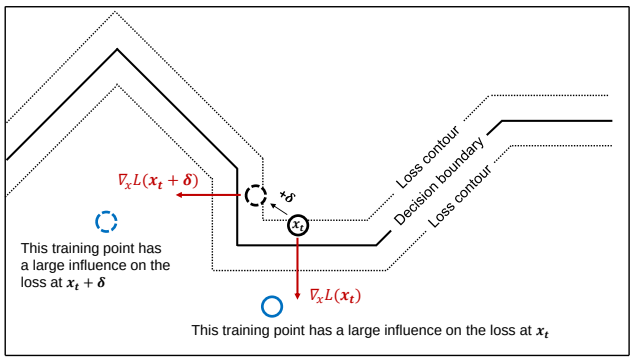


Figure 2: **Intuition for why interpretation is fragile.** Consider a test example $\boldsymbol{x_t} \in \mathbb{R}^2$ (solid black circle) that is slightly perturbed to a new position $\boldsymbol{x_t} + \boldsymbol{\delta}$ in input space (dashed black dot). The contours and decision boundary corresponding to a loss function (L) for a two-class classification task are also shown, allowing one to see the direction of the gradient of the loss with respect to the input space. Neural networks with many parameters have decision boundaries that are roughly piecewise linear with many transitions(Goodfellow, Shlens, and Szegedy 2014). We illustrate that points near the transitions are especially fragile to interpretability-based analysis. A small perturbation to the input changes the direction of $\nabla_x L$ from being in the horizontal direction to being in the vertical direction, directly affecting feature-importance analyses. Similarly, a small perturbation to the test image changes which data point (before perturbation: solid blue, after perturbation: dashed blue), when up-weighted, has the largest influence on L, directly affecting exemplar-based analysis.

input feature. We normalize the scores for each image by the sum of the feature importance scores across the features. This ensures that any perturbations that we design change not the absolute feature saliencies (which may preserve the ranking of different features), but their relative values. We summarize three different ways to compute normalized feature importance score, denoted by $\boldsymbol{I}(\boldsymbol{x}_t)$.

- **Simple gradient method** Introduced in (Baehrens et al. 2010) and applied to deep neural networks in (Simonyan, Vedaldi, and Zisserman 2013), the simple gradient method applies a first order linear approximation of the model to detect the sensitivity of the score to perturbing each of the input dimensions. Given input $\boldsymbol{x}_t \in \mathbb{R}^d$, the score is defined as: $\boldsymbol{I}(\boldsymbol{x}_t)_j = |\nabla_{\boldsymbol{x}} S_l(\boldsymbol{x}_t)_j| / \sum_{i=1}^{d} |\nabla_{\boldsymbol{x}} S_l(\boldsymbol{x}_t)_i|$.
- **Integrated gradients** A significant drawback of the simple gradient method is the saturation problem discussed by (Shrikumar, Greenside, and Kundaje 2017; Sundararajan, Taly, and Yan 2017). Consequently, Sundararajan, Taly, and Yan introduced the integrated gradients method where the gradients of the score with respect to M scaled versions of the input are summed and then multiplied by the input. Letting $\boldsymbol{x}^0$ be the reference point and $\Delta \boldsymbol{x}_t = \boldsymbol{x}_t - \boldsymbol{x}^0$, the feature importance vector is calculated by: $\boldsymbol{I}(\boldsymbol{x}_t) = \left| \frac{\Delta \boldsymbol{x}_t}{M} \sum_{k=1}^{M} \nabla_{\boldsymbol{x}} S_l \left(\frac{k}{M} \Delta \boldsymbol{x}_t + \boldsymbol{x}^0 \right) \right|$, which is then normalized for our analysis. Here the absolute value is taken for each dimension.
- **DeepLIFT** DeepLIFT is an improved version of layer-wise relevance propagation (LRP) method (Bach et al. 2015). LRP methods decompose the score $S_l(\boldsymbol{x}_t)$ backwards through the neural network. DeepLIFT (Shrikumar, Greenside, and Kundaje 2017) defines a reference point in the input space and propagates relevance scores proportionally to the changes in the neuronal activations from the reference. We use DeepLIFT with the Rescale rule; see (Shrikumar, Greenside, and Kundaje 2017) for details.

Sample importance interpretation A complementary approach to interpreting the results of a neural network is to explain the prediction of the network in terms of its *training examples*, $\{(\boldsymbol{x_i}, y_i)\}$. Specifically, to ask which training examples, if up-weighted or down-weighted during training time, would have the biggest effect on the loss of the test example $(\boldsymbol{x_t}, y_t)$. (Koh and Liang 2017) proposed a method to calculate this value, called the influence, defined by the equation: $I(z_i, z_t) = -\nabla_\theta L(z_t, \hat{\theta})^\top H_{\hat{\theta}}^{-1} \nabla_\theta L(z_i, \hat{\theta})$, where $z_i = (\boldsymbol{x}_i, y_i)$, $z_t = (\boldsymbol{x}_t, y_T)$, and $L(z, \hat{\theta})$ is the prediction loss of (training or test) data point z in network with parameters $\hat{\theta}$. $H_{\hat{\theta}} = \frac{1}{n} \sum_{i=1}^{n} \nabla_\theta^2 L(z_i, \hat{\theta})$ is the empirical Hessian of the network calculated over the training examples. We calculate the influence over the entire training set $\boldsymbol{I}(\cdot, z_t)$.

Metrics for interpretation similarity We consider two natural metrics for quantifying the similarity between interpretations for two different images:

- **Spearman's rank order correlation**: Because interpretation methods rank all of the features or training examples in order of importance, it is natural to use the rank correlation (Spearman 1904) to compare the similarity between interpretations.
- **Top-k intersection**: In many settings, only the most important features are of explanatory interest. In such settings, we can compute the size of intersection of the k most important features before and after perturbation.

2 Methods: Generating Perturbations

Problem statement For a given neural network $\mathscr{N}$ with fixed weights and a test data point $\boldsymbol{x}_t$, the feature importance and sample importance methods produce an interpretation $\boldsymbol{I}(\boldsymbol{x}_t; \mathscr{N})$. For feature importance, $\boldsymbol{I}(\boldsymbol{x}_t; \mathscr{N})$ is a vector of feature scores; for influence function $\boldsymbol{I}(\boldsymbol{x}_t; \mathscr{N})$ is a vector of scores for training examples. Our goal is to devise efficient and visually imperceptible perturbations that change the interpretability of the test input while preserving the predicted label. Formally, we define the problem as:

$$\arg\max_{\boldsymbol{\delta}} \mathcal{D}\left(\boldsymbol{I}(\boldsymbol{x}_t; \mathscr{N}), \boldsymbol{I}(\boldsymbol{x}_t + \boldsymbol{\delta}; \mathscr{N})\right)$$
$$\text{subject to: } ||\boldsymbol{\delta}||_\infty \leq \epsilon,$$
$$\text{Prediction}(\boldsymbol{x}_t + \boldsymbol{\delta}; \mathscr{N}) = \text{Prediction}(\boldsymbol{x}_t; \mathscr{N})$$

where $\mathcal{D}(\cdot)$ measures the change in interpretation (e.g. how many of the top-k pixels are no longer the top-k pixels of the feature importance map after the perturbation) and $\epsilon > 0$ constrains the norm of the perturbation. In this paper, we carry out three kinds of input perturbations.

Random sign perturbation As a baseline, each pixel is randomly perturbed by $\pm\epsilon$. This is used as a baseline with which to compare our adversarial perturbations against both feature importance and sample importance methods.

Iterative attacks against feature importance methods In Algorithm 1, we define three adversarial attacks against feature importance methods, each of which consists of taking a series of steps in the direction that maximizes a differentiable dissimilarity function between the original and perturbed interpretation. (1) The **top-k** attack seeks to perturb the feature importance map by decreasing the relative importance of the k initially most important input features. (2) For image data, feature importance map's center of mass often captures the user's attention. The **mass-center** attack is designed to result in the maximum spatial displacement of the center of mass. (3) If the goal is to have a semantically meaningful change in feature importance map, **targeted** attack aims to increase the concentration of feature importance scores in a pre-defined region of the input image.

Gradient sign attack against influence functions We can obtain effective adversarial images for influence functions without resorting to iterative procedures. We linearize the equation for influence functions around the values of the current inputs and parameters. If we further constrain the L_∞ norm of the perturbation to ϵ, we obtain an optimal single-step perturbation:

$$\boldsymbol{\delta} = \epsilon \text{sign}(\nabla_{\boldsymbol{x}_t} I(z_i, z_t)) = \\ -\epsilon \text{sign}(\nabla_{\boldsymbol{x}_t} \nabla_\theta L(z_t, \hat{\theta})^\top \underbrace{H_{\hat{\theta}}^{-1} \nabla_\theta L(z_i, \hat{\theta})}_{\text{independent of } \boldsymbol{x}_t}) \quad (1)$$

The attack we use consists of applying the negative of the perturbation in (1) to decrease the influence of the 3 most influential training images of the original test image[1]. Of course, this affects the influence of all of the other training images as well.

We follow the same setup for computing the influence function as was done in (Koh and Liang 2017). Because the influence is only calculated with respect to the parameters that change during training, we calculate the gradients only with respect to parameters in the final layer of our network (InceptionNet, see Section 3). This makes it feasible for us to compute (1) exactly, but it gives us the perturbation of the input *into the final layer*, not the first layer. So, we use standard back-propagation to calculate the corresponding gradient for the input test image.

[1]In other words, we generate the perturbation given by: $-\epsilon \text{sign}(\sum_{i=1}^{3} \nabla_{\boldsymbol{x}_t} \nabla_\theta L(z_t, \hat{\theta})^\top H_{\hat{\theta}}^{-1} \nabla_\theta L(z_{(i)}, \hat{\theta}))$, where $z_{(i)}$ is the i^{th} most influential training image of the original test image.

Algorithm 1 Iterative feature importance Attacks

Input: test image $\boldsymbol{x}_t$, maximum norm of perturbation ϵ, normalized feature importance function $\boldsymbol{I}(\cdot)$, number of iterations P, step size α

Define a dissimilarity function D to measure the change between interpretations of two images:

$$D(\boldsymbol{x}_t, \boldsymbol{x}) = \begin{cases} -\sum_{i \in B} \boldsymbol{I}(\boldsymbol{x})_i & \text{for } \textbf{top-k} \text{ attack} \\ \sum_{i \in \mathcal{A}} \boldsymbol{I}(\boldsymbol{x})_i & \text{for } \textbf{targeted} \text{ attack} \\ ||\boldsymbol{C}(\boldsymbol{x}) - \boldsymbol{C}(\boldsymbol{x}_t)||_2 & \text{for } \textbf{mass-center} \text{ attack,} \end{cases}$$

where B is the set of the k largest dimensions of $\boldsymbol{I}(\boldsymbol{x}_t)$, $\mathcal{A}$ is the target region of the input image in targeted attack, and $\boldsymbol{C}(\cdot)$ is the center of feature importance mass[a].

Initialize $\boldsymbol{x}^0 = \boldsymbol{x}_t$

for $p \in \{1, \ldots, P\}$ **do**

Perturb the test image in the direction of signed gradient[b] of the dissimilarity function:

$$\boldsymbol{x}^p = \boldsymbol{x}^{p-1} + \alpha \cdot \text{sign}(\nabla_{\boldsymbol{x}} D(\boldsymbol{x}_t, \boldsymbol{x}^{p-1}))$$

If needed, clip the perturbed input to satisfy the norm constraint: $||\boldsymbol{x}^p - \boldsymbol{x}_t||_\infty \leq \epsilon$

end for

Among $\{\boldsymbol{x}^1, \ldots, \boldsymbol{x}^P\}$, return the element with the largest value for the dissimilarity function and the same prediction as the original test image.

[a]The center of mass is defined for a $W \times H$ image as: $\boldsymbol{C}(\boldsymbol{x}) = \sum_{i \in \{1,\ldots,W\}} \sum_{j \in \{1,\ldots,H\}} \boldsymbol{I}(\boldsymbol{x})_{i,j}[i,j]^T$

[b]In ReLU networks, this gradient is 0. To attack interpretability in such networks, we replace the ReLU activation with its smooth approximation (softplus) when calculating the gradient and generate the perturbed image using this approximation. The perturbed images that result are effective adversarial attacks against the original ReLU network, as discussed in Section 3.

3 Experiments and Results

Data sets and models For attacks against feature importance interpretation, we used ILSVRC2012 (ImageNet classification challenge data) (Russakovsky et al. 2015) and CIFAR-10 (Krizhevsky 2009). For the ImageNet classification data set, we used a pre-trained SqueezeNet model introduced by (Iandola et al. 2016).

For both data sets, the results are examined on feature importance scores obtained by simple gradient, integrated gradients, and DeepLIFT methods. For DeepLIFT, we used the pixel-wise and the channel-wise mean images as the CIFAR-10 and ImageNet reference points respectively. For the integrated gradients method, the same references were used with parameter $M = 100$. We ran all iterative attack algorithms for $P = 300$ iterations with step size $\alpha = 0.5$.

To evaluate our adversarial attack against influence functions, we followed a similar experimental setup to that of the original authors: we trained an InceptionNet v3 with all but the last layer frozen (the weights were pre-trained on ImageNet and obtained from Keras). The last layer was trained

on a binary flower classification task (**roses** vs. **sunflowers**), using a data set consisting of 1,000 training images[2]. This data set was chosen because it consisted of images that the network had not seen during pre-training on ImageNet. The network achieved a validation accuracy of 97.5%.

Results for attacks against feature importance scores From the ImageNet test set, 512 correctly-classified images were randomly sampled for evaluation. Examples of the mass-center attack against feature importance scores obtained by the three mentioned methods are presented in Fig. 1. Examples of targeted attacks, whose goal is to change the semantic meaning of the interpretation are depicted in Fig. 5 .

In Fig. 3, we present results aggregated over all 512 images. We compare different attack methods using top-1000 intersection and rank correlation methods. In all the images, the attacks do not change the original predicted label of the image nor does it significantly change the prediction confidence. Random sign perturbation already causes decreases in both top-1000 intersection and rank order correlation. For example, with $L_\infty = 8$, on average, there is less than 30% overlap in the top 1000 most salient pixels between the original and the randomly perturbed images across all three of interpretation methods.

Both the mass-center and top-K attack algorithms have similar effects on feature importance of test images when measured on the basis of rank correlation or top-1000 intersection. Not surprisingly, we found that the mass-center attack was more effective than the top-k attack at resulting in the most perceptive change. Average numerical results are not obtainable for the targeted attack as it is designed for semantic change and requires a target area of attack in each image. Comparing the effectiveness of attacks among the three different feature importance methods, we found that the integrated gradients method was the most difficult one to generate adversarial examples for. Similar results were obtained the CIFAR-10 (Krizhevsky 2009) data set.

Results for adversarial attacks against sample importance scores. We evaluate the robustness of influence functions on a test data set consisting of 200 images of roses and sunflowers. Fig. 4(a) shows a representative test image to which we have applied the gradient sign attack. Although the prediction of the image does not change, the most influential training examples change entirely.

In Fig. 4(b,c), we compare the random perturbations and gradient sign attacks for the test set. It shows that gradient sign-based attacks are significantly more effective at decreasing the rank correlation, as well as distorting the top-5 influential images. For example, on average, with a perturbation of magnitude $\epsilon = 8$, only 2 of the top 5 most influential training images remain in the top 5. The influences of the training images before and after an adversarial attack are essentially uncorrelated. However, we find that even random attacks can have a small but non-negligible effect on influence functions, on average reducing the rank correlation to 0.8 ($\epsilon \approx 10$).

[2]adapted from: https://goo.gl/Xgr1a1

4 Hessian Analysis

In this section, we explain the effectiveness of adversarial attacks on interpretations in terms of the high dimensionality and non-linearities in deep networks. High dimensionality is also a reason why adversarial examples are effective at changing prediction labels (Goodfellow, Shlens, and Szegedy 2014).

Let $S(\boldsymbol{x}; \boldsymbol{w})$ denote the score function of interest and $\boldsymbol{x} \in \mathbb{R}^d$ be the input vector. First order approximation of sensitivity of a gradient-based interpretation to perturbations in the input is $\boldsymbol{\delta} \in \mathbb{R}^d$ is: $\nabla_{\boldsymbol{x}} S(\boldsymbol{x} + \boldsymbol{\delta}) - \nabla_{\boldsymbol{x}} S(\boldsymbol{x}) \approx H\boldsymbol{\delta}$, where H is the Hessian matrix $H_{i,j} = \frac{\partial S}{\partial x_i \partial x_j}$. In the most simple case of having a linear model $S = \boldsymbol{w}^\top \boldsymbol{x}$, the feature importance vector is robust as it is completely independent of $\boldsymbol{x}$ ($\nabla_{\boldsymbol{x}} S = \boldsymbol{w}$). Thus, some non-linearity is required for adversarial attacks against interpretation. The simplest model susceptible to interpretation adversarial attacks is a set of weights followed by a non-linearity (e.g. softmax): $S = g(\boldsymbol{w}^\top \boldsymbol{x})$.

The first order approximation of change in feature importance map due to a small input perturbation: $\boldsymbol{x} \rightarrow \boldsymbol{x} + \boldsymbol{\delta}$ will be equal to : $H \cdot \boldsymbol{\delta} = \nabla^2_{\boldsymbol{x}} S \cdot \boldsymbol{\delta}$. In particular, the relative change in the importance score of the i^{th} feature is $(\nabla^2_{\boldsymbol{x}} S \cdot \boldsymbol{\delta})_i / (\nabla_{\boldsymbol{x}} S)_i$. For our simple model, this relative change is:

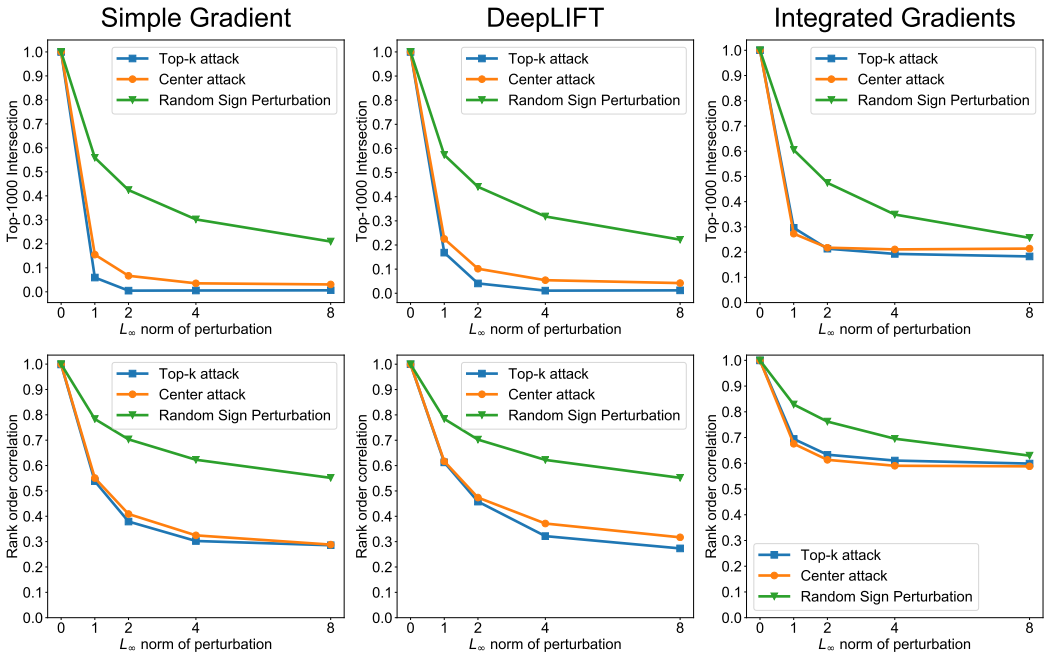
$$\frac{(\boldsymbol{w}\boldsymbol{w}^\top \boldsymbol{\delta} g''(\boldsymbol{w}^\top \boldsymbol{x}))_i}{(\boldsymbol{w} g'(\boldsymbol{w}^\top \boldsymbol{x}))_i} = \frac{\boldsymbol{w}^\top \boldsymbol{\delta} g''(\boldsymbol{w}^\top \boldsymbol{x})}{g'(\boldsymbol{w}^\top \boldsymbol{x})}, \tag{2}$$

where we have used $g'(\cdot)$ and $g''(\cdot)$ to refer to the first and second derivatives of $g(\cdot)$. Note that $g'(\boldsymbol{w}^\top \boldsymbol{x})$ and $g''(\boldsymbol{w}^\top \boldsymbol{x})$ do not scale with the dimensionality of $\boldsymbol{x}$ because in general, $\boldsymbol{x}$ and $\boldsymbol{w}$ are ℓ_2-normalized or have fixed ℓ_2-norm due to data preprocessing and weight decay regularization. However, if $\boldsymbol{\delta} = \epsilon \text{sign}(\boldsymbol{w})$, then the relative change in the feature importance grows with the dimension, since it is proportional to the ℓ_1-norm of $\boldsymbol{w}$. For a high dimensional input the relative effect of the perturbation can be substantial. Note also that this perturbation is exactly the sign of the first right singular vector of the Hessian $\nabla^2_{\boldsymbol{x}} S$, which is appropriate since that is the vector that has the maximum effect on the gradient of S.

Notice that for this simple network, the direction of adversarial attack on interpretability, $\text{sign}(\boldsymbol{w})$ is the same as the adversarial attack on prediction which means that perturbing interpretability perturbs prediction. For more complex networks, this is not the case, as we show empirically in Fig. 6, demonstrating that it is possible to perturb the interpretation without changing the class label.

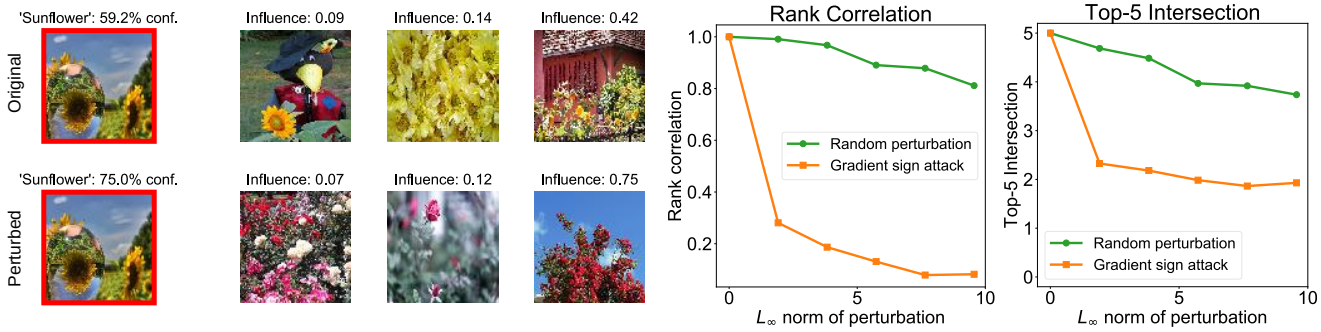
5 Discussion

This paper demonstrates that interpretation of neural networks can be fragile in the sense that two similar inputs with the same predicted label can be given very different interpretations. We develop perturbations to illustrate this



(a) (b) (c)

Figure 3: **Comparison of adversarial attack algorithms on feature-importance methods.** Across 512 correctly-classified ImageNet images, we find that the top-k and center attacks perform similarly in top-1000 intersection and rank correlation measures, and are far more effective than the random sign perturbation at demonstrating the fragility of interpretability, as characterized through top-1000 intersection **(top)** as well as rank order correlation **(bottom)**.



(a) (b) (c)

Figure 4: **Gradient sign attack on influence functions .** (a) An imperceptible perturbation to a test image can significantly affect sample importance interpretability. The original test image is that of a sunflower that is classified correctly in a rose vs. sunflower classification task. The top 3 training images identified by influence functions are shown in the **top row**. Using the gradient sign attack, we perturb the test image (with $\epsilon = 8$) to produce the leftmost image in the **bottom row**. Although the image is even more confidently predicted as a sunflower, influence functions suggest very different training images by means of explanation: instead of the sunflowers and yellow petals that resemble the input image, the most influential images are pink/red roses. (b) Average results for applying random (green) and gradient sign-based (orange) perturbations to 200 test images are shown. Random attacks have a gentle effect on interpretability while a gradient perturbation can significantly affect the rank correlation and (c) the 5 most influential images. Although the image is even more confidently predicted to be a sunflower, influence functions suggest very different training images by means of explanation: instead of the sunflowers and yellow petals that resemble the input image, the most influential images are pink/red roses. The plot on the right shows the influence of each training image before and after perturbation. The 3 most influential images (targeted by the attack) have decreased in influence, but the influences of other images have also changed.

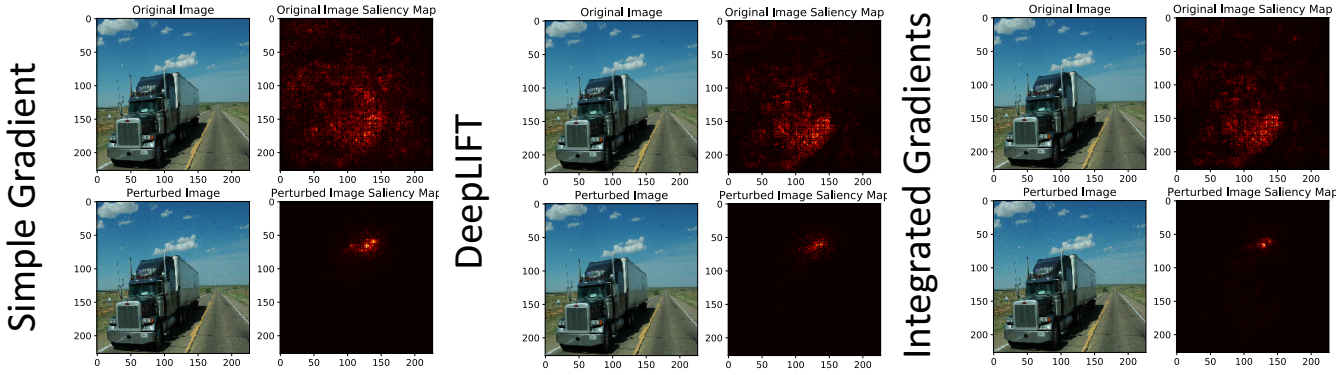


Figure 5: **Targeted attack against feature importance map.** Image is correctly classified as "trailer truck". For all methods, the devised perturbation with $\epsilon = 8$ was able to semantically meaningfully change the focus of saliency map to the "cloud" above the truck. (The cloud area was captured using SLIC (Achanta et al. 2012) superpixel segementation.) **(top)** as well as rank order correlation **(bottom)**.

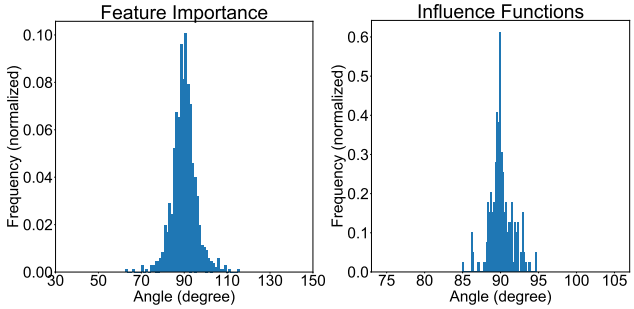


Figure 6: **Orthogonality of prediction and interpretation fragile directions** (a) The histogram of the angle between the steepest direction of change in (a) feature importance and (b) samples importance and the steepest prediction score change direction.

fragility and propose evaluation metrics as well as insights on why fragility occurs. Fragility of neural network interpretation can be orthogonal to fragility of the prediction, as we demonstrate with perturbations that substantially change the interpretation without changing the predicted label, but both types of fragility arise at least in part from high dimensionality, as we discuss in Section 4.

Our main message is that robustness of the interpretation of a prediction is an important and challenging problem, especially as in many applications (e.g. many biomedical and financial settings), users are as interested in the interpretation as in the prediction itself. Our results raise concerns on how interpretations of neural networks can be manipulated. Especially in settings where the importance of individual or a small subset of features are interpreted, we show that these importance scores can be sensitive to even random perturbation. More dramatic manipulations of interpretations can be achieved with our targeted perturbations. This is especially true for the simple gradients method, DeepLIFT, and influence functions, but also the integrated gradients method. These results raise potential security concerns. We do not suggest that interpretations are meaningless, just as adversarial attacks on predictions do not imply that neural networks are useless. Interpretation methods do need to be used and evaluated with caution while applied to neural networks, as they can be fooled into identifying features that would not be considered salient by human perception.

Our results demonstrate that the *interpretations* (e.g. saliency maps) are vulnerable to perturbations, but this does not imply that the *interpretation methods* are broken by the perturbations. This is a subtle but important distinction. Methods such as saliency measure the infinitesimal sensitivity of the neural network at a particular input $\boldsymbol{x}$. After a perturbation, the input has changed to $\tilde{\boldsymbol{x}} = \boldsymbol{x} + \boldsymbol{\delta}$, and the saliency now measures the sensitivity at the perturbed input. The saliency *correctly* captures the infinitesimal sensitivity at the two inputs; it's doing what it is supposed to do. The fact that the two resulting saliency maps are very different is fundamentally due to the network itself being fragile to such perturbations, as we illustrate with Fig. 2.

Our work naturally raises the question of how to defend against adversarial attacks on interpretation. Because interpretation fragility arises as a consequence of high dimensionality and non-linearity (see section 4), we believe that techniques that discretize inputs, such as thermometer encoding (Buckman et al. 2018), and train neural networks in a way to constrain the non-linearity of the network (Cisse et al. 2017a), may be useful in defending against interpretation attacks.

While we focus on image data (ImageNet and CIFAR-10), because these are the standard benchmarks for popular interpretation tools, this fragility issue can be wide-spread in biomedical, economic and other settings where neural networks are increasingly used. Understanding interpretation fragility in these applications and developing more robust methods are important agendas of research.

References

Achanta, R.; Shaji, A.; Smith, K.; Lucchi, A.; Fua, P.; Süsstrunk, S.; et al. 2012. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence* 34(11):2274–2282.

Bach, S.; Binder, A.; Montavon, G.; Klauschen, F.; Müller, K.-R.; and Samek, W. 2015. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one* 10(7):e0130140.

Baehrens, D.; Schroeter, T.; Harmeling, S.; Kawanabe, M.; Hansen, K.; and MÃžller, K.-R. 2010. How to explain individual classification decisions. *Journal of Machine Learning Research* 11(Jun):1803–1831.

Buckman, J.; Roy, A.; Raffel, C.; and Goodfellow, I. 2018. Thermometer encoding: One hot way to resist adversarial examples. *International Conference on Learning Representations*.

Cisse, M.; Bojanowski, P.; Grave, E.; Dauphin, Y.; and Usunier, N. 2017a. Parseval networks: Improving robustness to adversarial examples. *arXiv preprint arXiv:1704.08847*.

Cisse, M.; Bojanowski, P.; Grave, E.; Dauphin, Y.; and Usunier, N. 2017b. Parseval networks: Improving robustness to adversarial examples. In *International Conference on Machine Learning*, 854–863.

Dong, Y.; Su, H.; Zhu, J.; and Bao, F. 2017. Towards interpretable deep neural networks by leveraging adversarial examples. *arXiv preprint arXiv:1708.05493*.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Iandola, F. N.; Moskewicz, M. W.; Ashraf, K.; Han, S.; Dally, W. J.; and Keutzer, K. 2016. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <1mb model size. *CoRR* abs/1602.07360.

Kingma, D., and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Koh, P. W., and Liang, P. 2017. Understanding black-box predictions via influence functions. *arXiv preprint arXiv:1703.04730*.

Krizhevsky, A. 2009. Learning multiple layers of features from tiny images.

Kurakin, A.; Goodfellow, I.; and Bengio, S. 2016. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*.

Lipton, Z. C. 2016. The mythos of model interpretability. *arXiv preprint arXiv:1606.03490*.

Montavon, G.; Lapuschkin, S.; Binder, A.; Samek, W.; and Müller, K.-R. 2017. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition* 65:211–222.

Moosavi-Dezfooli, S.-M.; Fawzi, A.; and Frossard, P. 2016. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2574–2582.

Obermeyer, Z., and Emanuel, E. J. 2016. Predicting the future—big data, machine learning, and clinical medicine. *The New England journal of medicine* 375(13):1216.

Papernot, N.; McDaniel, P.; Jha, S.; Fredrikson, M.; Celik, Z. B.; and Swami, A. 2016. The limitations of deep learning in adversarial settings. In *Security and Privacy (EuroS&P), 2016 IEEE European Symposium on*, 372–387. IEEE.

Rich, M. L. 2015. Machine learning, automated suspicion algorithms, and the fourth amendment. *U. Pa. L. Rev.* 164:871.

Ross, A. S., and Doshi-Velez, F. 2017. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. *arXiv preprint arXiv:1711.09404*.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; Berg, A. C.; and Fei-Fei, L. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)* 115(3):211–252.

Shrikumar, A.; Greenside, P.; and Kundaje, A. 2017. Learning important features through propagating activation differences. *CoRR* abs/1704.02685.

Simonyan, K.; Vedaldi, A.; and Zisserman, A. 2013. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*.

Spearman, C. 1904. The proof and measurement of association between two things. *American Journal of Psychology* 15:72–101.

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic attribution for deep networks. *arXiv preprint arXiv:1703.01365*.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.