Why you shouldn't trust me: A survey on Adversarial Model .

Verena Heusser
verena.heusser@student.kit.edu
Karlsruhe Institute of Technology (KIT)
Karlsruhe, Germany

Abstract

The accuracy of machine learning models is no longer enough: we also want the often inherently complex models to be safe, robust, and interpretable by humans. As machine learning models enter critical areas in human lives such as the criminal justice system, medicine or financial systems, the inability for humans to understand these models is dangerous and problematic. Progress in the emerging research field of explainable artificial intelligence promises to be a remedy, where interpretation methods try to uncover *how* models work. However, while the number of studies using these methods is exploding, there are also a growing number of articles pointing out that the interpretation methods are still fundamentally flawed. This work provides an overview of the subfield of manipulating interpretable machine learning, with the goal of providing insight into concepts, existing research, and future directions. We want to raise awareness that interpretation methods, just like the underlying machine learning models, can be outwitted by adversaries and that there is often no way to detect adversarial attacks.

Keywords

Interpretability, Adversarial Machine Learning, Adversarial Model Interpretation Manipulations

ACM Reference Format:
Verena Heusser. 2021. Why you shouldn't trust me: A survey on Adversarial Model .. In *Interpretability '20: ACM Symposium on Neural Model Explanations, December 16–20, 2020, Karlsruhe, DE.* ACM, New York, NY, USA, **??** pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

In recent years, deep learning models have demonstrated superior performance on a variety of tasks [**? ? ?**]. While performance is still increasing and more tasks are being handled, their performance comes at the cost of complexity: models often use millions to billions of parameters to achieve universal function approximation. This complexity means that they remain black boxes that cannot be interpreted even by experts.

Code for replications and experiments will be made available on https://github.com/verenaHeusser/adversarial_interpretation_manipulation All figures in this article are produced by the author unless noted otherwise.

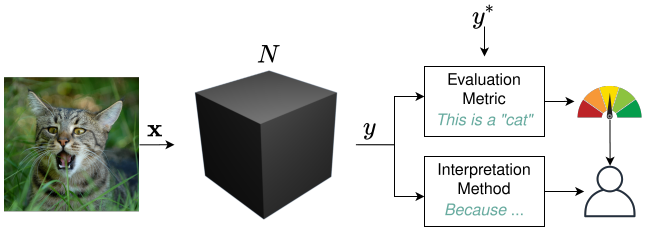


Figure 1: Prediction pipeline using a machine learning model, depicted as black box. Typically, evaluation metrics require the prediction y and the ground truth label y^* allowing for the assessment of the model's accuracy. Additionally answering the question *why*, i.e. making the model interpretable for a human, requires additional methods.

Such a black box is able to predict fairy well for unseen yet similar data, answering the question of *what* is the most likely label for an input sample. However, most models will provide no answer to *why* or *how* the model chose this label for the instance and which features of the instance were crucial for making this prediction. For example, if a machine learning model is tasked with classifying images of cats (as in **??**), one would like to assume that the presence (or absence) of a cat in an image is indicative (contraindicative) of the classification of the image into the "cat" ("not cat") category.

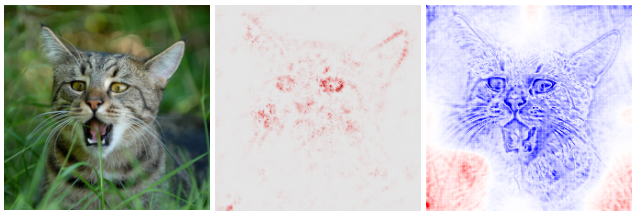
Automated algorithms are already in use in critical areas, such as medicine, chemistry, the criminal justice system, the financial sector or the piloting of self-driving cars [**? ? ?**]. Thus, as machine learning models are moving out of the lab into the real-world, the inability of humans to understand these (black box) models seems even more problematic. Not knowing how a model makes predictions, and not being able to detect systematic biases in the model, prevents the vastly advancing technology of machine learning from being used in highly sensitive and safety-critical applications.

Suppose a deep neural network predicts the risk for cancer from a mammogram, which is an image of breast tissue. A doctor would only use the algorithm if there is a way to validate that (1) the algorithm is accurate (which can be measured in terms of the predictive accuracy), and (2) if the model is also using the correct indications in the data for predicting the risk of cancer. (1) is the standard approach for validating the performance of machine learning models, but in this example, one can clearly see why predictive accuracy might not be enough in many areas. For approaching (2), i.e. uncovering *why* a model predicts a low / high risk of cancer, the research field of explainable artificial intelligence (XAI) offers a fast growing number of methods. Some research even suggests to allow 'peeking inside the black box' of deep learning models [**?**]. The threats of

Explainability '20, December 16–20, 2020, Karlsruhe, DE

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

adversarial attacks on deployments of machine learning models has contributed to the field of XAI comprising topics of (1) *model interpretations*, (2) *adversarial attacks*, or manipulation methods and (3) the field of *adversarial manipulations of model interpretations*. All these subfields have the common goal to make models more robust and safe for deployment. We informally define a model to be robust if it's output is consistently accurate even if one or more of the input features are altered. (1) refers to the development of techniques that can be used to understand and explain the decision making process of a machine learning model or even the development of models that are inherently interpretable. (2) is the field of detecting vulnerabilities in models that cause these models to be deceived by altered input. (3) is the main topic of this paper, i.e. how to fool interpretation methods in order to detect vulnerabilities and malfunctions in interpretation methods. Ideally, an interpretation or explanation method should indicate which features in the input to a model contribute to the prediction and also to what extent each feature contributes. This notion of extent is often called the *importance* score of a feature. **??** shows such a feature importance map (**??**) produced by the interpretation method LRP [**?**] applied to the neural network model Inception-v3 [**?**]. The input image is from the ImageNet dataset [**?**]. The output of the interpretation method is projected onto the original image for better human readability. This importance map suggests that specific portions of the original cat image are important for the neural network make the high-confidence prediction (see..) of the category 'tiger cat'.



(a) Original. **(b) Map.** **(c) Fooled Map.**

Figure 2: Visualization of the feature importance and fooled feature importance maps produced py the LRP interpreter applied to the image of a cat and an image classification model. The fooling method is the *location* fooling from [?]. Dark red here means a stronger significance of the feature.

While interpretation methods are already used for analysis of computer vision systems [**? ? ?**], text and sequence analysis [**? ?**], and deep learning in security [**?**], there is still a lack in the the understanding of model interpretation methods. Recent work has shown, that humans are not able to benefit much from interpretation techniques: They cannot build better models and improve their own performance [**?**], are not better at detecting false model decisions even in transparent models [**?**], and even data scientists over-trust and misuse model interpretation methods [**?**]. Methodologically, it is unclear how the variety of proposed model interpretation methods are related and what common concepts can be used to evaluate and compare them. Many works are dedicated to establish a formal definition of what it means for a model to be interpretable, and how to select and evaluate methods for generating interpretations of machine learning models [**? ?**]. However, a growing number of researchers also focus on breaking these interpretation methods, similar to research on adversarial attacks on machine learning models, which is deeply fueled by the vast deployment of models in the real world [**? ? ? ? ? ? ? ? ? ? ?**].

Outline.
This article examines a topic at the intersection of interpretable and adversarial machine learning research. The overview presented in this article examines the existing literature and contributions in the field of XAI focusing on methods to manipulate explanation methods. We try to offer a comprehensive taxonomy of these interpretation manipulation methods.

This paper is structured as follows. **??** introduces common interpretation methods for machine learning models, and offers a taxonomy of the variety of techniques and a brief outline of popular interpretation methods. In **??**, the main topic of this paper, namely manipulation methods for deceiving interpretation techniques, is outlined. A taxonomy of methods is proposed and possible evaluation criteria are listed. **??** proposes approaches to benchmark the robustness of interpretation techniques and **??** provides a review of important studies in the field of manipulation methods. The implications of this research area on XAI are discussed in section **??**.

2 Interpretation Methods

There exists a variety of definitions in the vastly expanding research field of XAI, and the concept of *interpretability* still has no formal commonly used technical meaning [**?**]. To build on the common ground of existing research, this work follows the terminology of Lipton et al. [**?**] and Arrieta et al. [**?**].

Broadly, interpretability focuses on *how* and *why* a machine learning model makes predictions. Simply put, interpretability is focused on getting some notion of an explanation in human understandable terms for the decisions made by a machine learning model.

2.1 Terminology

The authors of [**?**] make a distinction between the related but different concepts of *interpretability* and *explainability*. Lipton [**?**] further breaks down interpretability into *transparency* and *post-hoc* interpretability. The notion of *explainability* from [**?**] can be related to Liptons's *transparency*, while Lipton's *post-hoc* interpretability is essentially *interpretability* as defined by Arrieta et al. [**?**].

Post-hoc Interpretability refers to the the extent to which cause and effect can be observed in a model, which can be translated to uncovering *why* a model made prediction y to an input $\mathbf{x}$, or how input and output relate to each other. *Post-hoc* means that interpretations are computed by applying methods that analyze the model after training. Consider the example of image classification from **??**. Here, interpretability would mean that if a cat is present in the image (the cause), the (trained) model classifies it to the category 'cat' (the effect). Now imagine we find that the model takes the green meadow in the image as evidence to predict 'cat', and not the cat itself. This would imply a lack of interpretability, as the model learns to assign features to the concept 'cat' other than those related to 'cat' in the correct sense. This toy example

emphasizes a common problem in image classification: [**?**] observe the over-reliance of models on image background, rather than on objects in the foreground, and is thus not just a reality distant toy example.

Explainability or Transparency on the other hand spans methods to uncover *how* a model makes predictions, meaning to observe the inner workings of a model and thus to literally explain what is happening in terms of understanding of the mechanisms by which a model works. Thus, transparency refers to the model's inherent properties that can be known before the training process and that are helpful to understand the model.

While both concepts seem to be important for the general objective of explainable artificial intelligence, this paper focuses on post-hoc interpretability. There are essentially two ways to achieve interpretability: (1) to use inherently interpretable models or (2) to post-process a model in a fashion that allows to yield insights into it's decision. An inherently interpretable model is a model with restricted complexity (e.g. a linear model). On the other hand, *post-hoc*, as outlined earlier, refers to the application of methods for analyzing a model after model training. Post-hoc methods span *model-agnostic* methods and *model-transparent* methods. Within model-agnostic methods, the model itself is unknown and approximated with a *surrogate* model (see **??**), while within *model-transparent* methods, the model itself is used to compute interpretations (see **??**).

Local and Global Methods. A further categorization can be made based on the scope of interpretations: *Local* methods aim at providing interpretations that are true for a single data point and its neighbors. *Global* methods aim at gaining interpretations that are valid for most data points in a class [**? ? ?**]. The interpretation methods discussed within this paper mostly fall into the class of local explanation methods [**? ? ?**].

Feature Attribution Methods and Sample Attribution Methods. Interpretation methods aim at making complex and inherently uninterpretable black box models interpretable by creating human readable visualizations. A frequently used type of explanation methods are *feature attributions* mapping each input feature to a numeric score. This score should quantify the importance of the feature relative to the model output. The resulting attribution map is then visualized as a heatmap projected onto the input sample. This allows humans to interpret which input attributes are the most helpful for making tha final prediction. Sample attribution methods on the other hand interpret the model performance in terms of the importance of training examples from the dataset.

We adhere to the following formal definition of an interpretation method:

Definition 1: Interpretation Method.
We consider a neural network $N : \mathbb{R}^d \rightarrow \mathbb{R}^k$. For the an arbitrary classification task, N classifies an input sample $\mathbf{x} \in \mathbb{R}$ in k categories where the prediction $f_N(\mathbf{x}) = y \in \{1, ..., K\}$ is given by $y = argmax_i f_N(\mathbf{x})_i$.

Given the neural network N, it's input vector $\mathbf{x} \in \mathbb{R}^d$ and the the neural networks prediction for input $\mathbf{x}$, $f_N(\mathbf{x}) = y$, an interpretation method $\bar{I}$ determines why label y has been chosen by N for input $\mathbf{x}$. The interpretation is given by an output vector $\mathbf{h_k} \in \mathbb{R}^d$ for a class k where each entry h_i is mostly a numeric value describing the relevance for the i-th input feature x_i of $\mathbf{x}$ for the final score $f_N(\mathbf{x})$.

As $\mathbf{h}$ has the same dimensions as the input $\mathbf{x}$ it can be mapped to the input, overlaying $\mathbf{x}$ as a heatmap, where the color value represents the importance of feature x_i towards the prediction $f_N(\mathbf{x})$. An example is given in **??**. Higher values, implying a stronger relative importance for making the prediction $f_N(\mathbf{x})$, are depicted in dark red.

While all explanation methods try to obtain importance measures for the network prediction, they differ with respect to how these measures are obtained. [**?**] propose two major categories for interpretation strategies, namely *black-box*, in the following named *model-agnostic* methods and *white-box*, or *model-transparent* interpretation methods. While black-box interpretations assume no knowledge about the underlying model, white-box methods only work by using the model parameters.

This terminology of discriminating between black-box and white-box methods may not be confused with the nature of the underlying models: Models still remain of black-box nature even though a white-box method may contribute to making the decision making process of such a model more insightful.

The following section details the two categories and gives examples of commonly used interpretation methods within each group.

2.2 Model-agnostic methods.

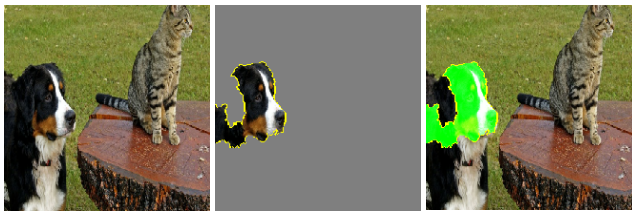
Model-agnostic interpretations assume no knowledge about the model thus treating it as a black box. The underlying model is approximated by learning it's behavior with an interpretable model, e.g. a linear model. The interpretable model is also dubbed the 'surrogate' model. The common approach for learning the surrogate is to approximate the relationship between the input samples and the corresponding prediction by the model. As the model itself does not need to be known, these approaches can be used in scenarios where the model itself is not directly accessible. Model-agnostic interpretations are fairly popular and are used in a wide range of applications, ranging from finance and law to medicine and chemistry [**? ?**].

A black-box interpretation offers the great advantage of being applicable to any model and offers simplicity because the interpretation is embedded in the model. However, this option of gaining interpretability might be costly for users that already have a high performing model. For this reason, growing need for methods exists that can be applied without retraining or modifying the underlying model. We will briefly describe two common model-agnostic approaches. Please refer to the original papers for details.

Local Interpretable Model-agnostic Explanations (LIME). LIME [**?**] perturbs the input and observes how the predictions of a black box model change. For the task of image classification, LIME creates a set of perturbed instances by dividing the input image into interpretable components, which are technically contiguous super-pixels, and runs each perturbed instance through the model to get the probability for how much the change in each super-pixel influences the whole model prediction.

?? shows an example of LIME applied to the neural network model Inception-v3 [**?**]. Input image from the ImageNet dataset [**?**

]. The super-pixel in dog instances place is highlighted in green, which correctly indicates that this super-pixel has a high influence on the prediction of the images correct class ('bernese mountain dog'). LIME also correctly indicates that the super-pixel in the cat's place does not indicate the correct class label.



(a) Original (b) Mask. (c) Saliency Map.

Figure 3: Visualization of the output of the LIME interpreter applied to an image and image classification model.

SHapley Additive exPlanations (SHAP). SHAP [**?**] builds on Shapley analysis, which is essentially about judging the importance of attributes. The model is trained on a number of subsets of all available features, and the feature importance scores are calculated by evaluating the effects that the omissions of the specific features have on the model prediction. An example is shown in **??**. Image regions highlighted in green are found to be important for predicting the correct label.

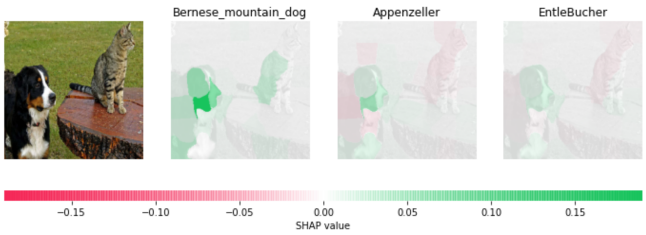


Figure 4: Visualization of the output of the SHAP interpreter applied to applied to an image and image classification model.

2.3 Model-transparent methods.

The other group of interpretations are model-transparent, or white-box methods, where the underlying model is known with all its parameters. Thus, the interpretation can be directly computed by using the model instead of relying on an approximation of f_N as within the black-box methods. These methods typically rely on the relationship between an input sample, the underlying model's prediction and the associated activations of the models hidden layers. Methods within this group are for example propagation-based and gradient-based approaches. The former propagate the model's prediction back through the model. The latter make use of the information provided by the gradients of the loss function, which contain sensitive information about the prediction and the features. Using the backpropagation method, features in the input can be highlighted based on the amount of gradient they receive. This shows their contribution to the final score. A few example methods within this group are listed below.

Layer-wise Relevance Propagation (LRP). While many approaches in the group of model-transparent interpretations are designed only for image classification, or convolutional neural networks, this method [**?**] is an exception. LRP propagates relevance values backwards through the network and decomposes the score of a predicted class backwards through the network. It relies on a Taylor series close to the prediction point rather than partial derivatives at the prediction point itself. An example for a feature map produced by LRP can be found in **??**.

DeepLIFT [**?**] is an improved version of LRP, where a reference point in the input feature space is defined. Relevance scores are propagated proportionally to the changes of neuronal activations from the reference.

SmoothGrad (SG) [**?**] averages over interpretations of noisy copies of an input sample thus reducing noise by visually diffusing the interpretation. The noise is drawn i.i.d. from a normal distribution.

Methods developed specifically for convolutional neural networks are for instance Grad-CAM [**?**] or SimpleGrad [**?**].

3 Manipulation Methods

As outlined in **??**, there are a variety of explanation methods readily available as frameworks and open source implementations. However, there is still little analysis on the robustness and reliability of such methods. While it is already common practice to test machine learning models against adversarial attacks in a number of domains [**? ?**], the same is not yet the standard for interpretation methods. We argue that interpretation techniques should not be used in critical applications without basic testing of interpretation techniques against adversarial settings.

3.1 Adversarial Setting

Adversarial Attacks on Models. Adversarial examples, as first introduced by [**?**], are clever manipulations of an input by an adversary which aims at causing misclassification and fooling of applications. They are mostly used to fool or attack machine learning models. We formally define adversarial attacks by the following properties:

Definition 2: Model Manipulation Method

1. *Imperceptibility of Perturbation*: The adversarial example is similar to original data, i.e. the norm of the added perturbation δ to an input sample $\mathbf{x}$ thus must be small, i.e.
$$\|\mathbf{x}+\delta\| = \|(\mathbf{x}+\delta)-\mathbf{x}\|_{\text{inf}} \leq \epsilon$$
2. *Prediction dissimilarity*: The prediction of the model is significantly different to the prediction on the non-adversarial example:
$$f_N(\mathbf{x}+\delta) \approx f_N(\mathbf{x})$$

Note, that within adversarial fooling of models, the perturbation is mostly applied to the input data, and not to the model itself.

Evidence from many studios shows that deep learning models can be easily tricked by adversarial examples. Albeit there are not

yet as many studies, there also exists evidence that many interpretation methods are also fragile with respect to small changes to input data [? ? ?] as well as to the model itself [? ?]. This fooling of interpretation methods is outlined below.

Adversarial Attacks on Model Interpreters. Contrary to adversarial attacks on machine learning models, the focus of this paper is on the attacks on interpretation techniques without changing the prediction of the model. An adversarial attack on a model interpretation is in the following also called a *manipulation* method. The goal is to apply perturbations to either an input sample or the model to change the output of an interpretation technique while leaving the model prediction unchanged. The last condition is important because adversarial interpreter manipulations aim to fool the interpretation method and essentially not the model itself. Fooling the model would only disclose the vulnerability of the model but would not allow to gain insight into the stability [1] of the interpretation method.

Again, the problem can be formally defined as:

Definition 3: Interpretation Manipulation Method.
A manipulation method $\mathcal{F}$ is defined as a method for altering the output of an explanation method $\mathcal{I}$ while leaving the model performance of the neural network N roughly unchanged. As manipulations can be applied on the input or the model level (see **??**), $\mathbf{x} + \delta$ denotes a perturbed input sample regarding the input level manipulation, while $N + \delta$ denotes a model with altered parameters, referring to the model level manipulation.

A manipulation method is successful in fooling an interpreter, if the following properties hold:

1. *Prediction similarity*: The model prediction stays approximately the same, i.e.

$$f_N(\mathbf{x} + \delta) \approx f_N(\mathbf{x}), \text{ or } f_{N+\delta}(\mathbf{x}) \approx f_N(\mathbf{x})$$

2. *Imperceptibility of Perturbation*: In case the attack is in the input domain of the model, the perturbation of input samples must be imperceptible by humans. According to [?], the norm of the added perturbation δ to an input sample $\mathbf{x}$ thus must be small, i.e.

$$\|\mathbf{x} + \delta\| = \|(\mathbf{x} + \delta) - \mathbf{x}\|_{\text{inf}} \leq \epsilon$$

These measures are to be seen as comparison between a baseline model N and a model that is applied an the adversarial setting (i.e. either N is not changed but applied to adversarially altered data $x+\delta$, or N is adversarially trained thus becoming $N + \delta$). Interpretation manipulation methods differ with respect to the definition of the fooling of the interpretation method. Some methods aim to make the interpreter give wrong interpretations[?]. In this case, the following additional property must hold:

3. *Interpretation dissimilarity*: The explanation map $h(\mathbf{x} + \delta)$ is significantly different to the explanation map resulting from non adversarial models or inputs $h(\mathbf{x})$, i.e. $h(\mathbf{x} + \delta)$ or

$$\arg\max_{\delta} \mathcal{D}(\mathcal{I}(x_i, \boldsymbol{\omega}), \mathcal{I}(x_i + \delta, \boldsymbol{\omega}))$$

where $\mathcal{D}(\cdot)$ is a distance measure.

[1] Stability is given if a method yields consistent outcomes in independent runs.

Others introduce a systematic bias into tho model while fooling the interpretation method in the sense that the interpretation does not change between the original model and the adversarial model. Here, it must hold that property 3. takes the minimal argument.

[?] extend these properties to include the so called *model similarity*. This measure extends the *prediction accuracy* to span the accuracy difference in between the baseline model and the new model, and also the mismatch of data points where the predictions of both models differ.

After formally defining what successfully fooling an interpretation method means, we want to provide intuition in **??**. The interpretation method produces an interpretation, here in form of a saliency map projected onto the original image. If a successful manipulation is applied, the resulting saliency map of the same interpreter should be different to the original map. This effect is clearly visible, as **??** is visually significantly different to the map produced by the same interpretation method but applied to an adversarial model (see **??**).

3.2 Taxonomy of Interpretation Manipulation Methods

There are two important categories of Manipulation Methods that aim at attacking model interpretation methods. The first category is based on the level these manipulations operate on, i.e. input space level or model / parameter level. The second categorization is based on the target of the manipulations. While untargeted manipulations are mostly random perturbations, targeted manipulations aim to perturb the interpretation of specific input features.

3.2.1 *Manipulation Levels*

Adversarial Input Manipulation. The general approach is to perturb or alter input data while observing the effect of this perturbation on the model prediction. This concept is visualized in **??**. As found in [?], visually-imperceptible perturbations of an input image can make explanations worse for the same model and interpreter.

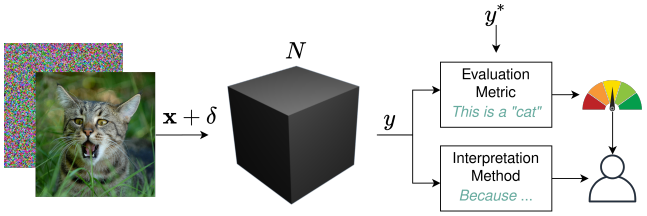


Figure 5: Depiction of an adversarial input manipulation. The model is fine-tuned with altered input samples, which are indicated by $\mathbf{x} + \delta$.

Adversarial Model Manipulation. Contrary to input manipulations, model manipulations do not operate on the input space but rather on the model parameter space itself. As first introduced by Heo et al. [?] in 2019, this line of research is comparably new. Adversarial model manipulations for fooling the interpreter are obtained by fine-tuning the model on the same data but with an adapted objective function. [?] propose the adapted loss function

for the task of image classification of

$$\mathcal{L} = \mathcal{L}_{CE}(\mathcal{D}; \omega) + \lambda \cdot \mathcal{L}(\mathcal{D}; \omega; \omega_0)$$

where $\mathcal{L}_{CE}$ is the standard cross-entropy classification loss. Adversarial model manipulation

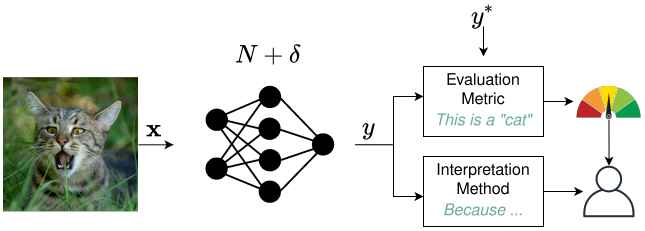


Figure 6: Depiction of an adversarial model manipulation. The model is fine-tuned with the same distribution of input data and a fooling loss, thus yielding the biased model $N+\delta$.

3.2.2 *Manipulation Targets*

In addition to the categorization of manipulation methods based on the manipulation level, the methods can further be categorized based on the target of their perturbation. The first possibility is *untargeted* perturbation, the second is *targeted* perturbation. Both these styles can be applied on either model and input level.

Untargeted Manipulations. The majority of manipulations is untargeted, meaning that the applied perturbations are mostly random and not designed to change the prediction for a specific portion of an input sample.

Targeted Manipulations. On the contrary, targeted manipulations aim at specifically changing the explanation of certain features of an input instance [**?**], and sometimes even changing the interpretation of specific classes [**?**]. Such a specific feature might be an object in the input image in the context of image classification. [**?**] for instance introduce a fooling scheme in which the interpretations of the target classes elephant and school bus are swapped. Manipulations on the level of the model are mostly targeted, as the explanation methods are being fooled by adapting the model parameters.

3.3 Evaluation Criteria

Besides the necessary properties of a successful interpretation manipulation method, other evaluation criteria are important to access the success of a fooling method as well as to enable the comparison between different fooling methods. These criteria are informally defined in the following.

Effectiveness. The manipulation scheme is inexpensive to conduct. Input manipulations are by definition inexpensive, as the perturbation can be applied to single input samples. Model manipulations are more expensive as they require tho model parameters to be adapted. However, an adversarial model can be obtained by fine-tuning the model with an adapted objective function. This fine-tuning also has the advantage that the model is adapted to include a systematic bias and can thus be applied to fool explanation methods without further adapting the model or input samples after the fine-tuning step. Furthermore, this systematic bias is hidden in the model, and is hard to uncover. Input manipulations can only fool the model when the inputs are always manipulated.

Transferability. The manipulation does not only fool one type of interpretation method, but it's effect transfers to other interpretation techniques.

Stability. The manipulation method influences the interpretation method consistently in a similar way in independent runs.

Generalisation. Generalization of an attack refers to the transfer of fooling to other test samples. This is noteworthy since a manipulation method might only perturb the decision boundary locally around the training points, i.e. only influencing training instances and their neighbors. However, it is desired that the explanations of unseen samples are affected as well. Furthermore, not only unseen samples interpretations, but also samples that are far away in the feature space should be affected. Otherwise, the perturbation is only local around the training points, thus the perturbation does not generalize.

As outlined in section **??**, there exists a plethora of interpretation methods differing in the assumption about the model character and also with respect to how interpretations are obtained. Thus, reliable evaluation methods are required allowing for a choice of an appropriate and robust interpretation method. Ultimately, the accordance to these evaluations should naturally allow for choosing an appropriate and robust interpretation method. Evaluations of the quality of an explanation method are separated into qualitative and quantitative evaluations.

Qualitative Evaluation. Inspection and random sampling are commonly used techniques to obtain an intuition about the effect of manipulations. As interpretations are attributed to input features, the resulting relevance values l can be easily mapped to the input vector $\mathbf{x}$. Visual inspection of these evaluations for specific samples is informative, but does not allow for general statistics and validation of manipulation effects. Thus, quantitative evaluations are required. [**?**] consider two kinds of visualizing saliency maps:

- **Absolute Value (ABS).** Saliency maps visualized this way show only the absolute values of the normalized maps.
- **Diverging.** The diverging visualization indicates positive and negative importance in different colors.

Quantitative Evaluation. As the goal of interpreter manipulations is to fool an interpreter, thus altering the output of in interpreter, it is straightforward to compare interpretations and data samples before and after perturbation [**?**].

As interpreter manipulations shall alter the output of an interpretation methods while keeping the prediction of a model unchanged, the following metrics

- **Fooling Success Rate (FSR).** [**?**] introduce the concept of the Fooling Success Rate (FSR). The FSR captures the relationship between the model's predictive accuracy and the correctness of the interpretation averaged over multiple test samples. The 'correctness' of the interpretation is the difference between the original interpretation methods output

and the output of the same interpretation method when applied to either an adversarial input sample or adversarial model. The FSR counts samples for the model's prediction did not change but for which the interpretation method's output changed as a successfully fooled example. Thus, the higher the FSR, the the more often the fooling is successful in fooling the interpretation method.
- **Area Over Prediction Curve (AOPC).** AOPC [**?**] is a principled way of quantitatively evaluating that the interpretations found by interpretation methods are valid, i.e. based on the features that the model truly uses for making the prediction. [**?**] use this measure to ensure that their adversarial model training does not fool the model (i.e. the predictive accuracy of the model is unchanged), but does fool the interpretation method.

As the goal of interpreter manipulations is to fool an interpreter, thus altering the output of in interpreter, it is straightforward to compare interpretations and data samples before and after perturbation [**?**]. There are metrics that are applied to identify changes in salient features for any task:

- **Spearman's rank order correlation.** As interpretation methods rank the features based on their importance, the rank correlation [**?**] is a natural measure for comparing interpretations.
- **Intersection of the top-k features.** For some tasks, only the top-k features are relevant, such that a comparison between these top-k features is insightful.

Contrary, the following metrics are used only in computer vision tasks. They have been used in literature to quantify similarities between natural images and remove duplicates. Therefore, they are used in XAI research to compare the changes between saliency maps before and after manipulation for computer vision tasks.

- **Structural Similarity Index (SSIM).** SSIM values are relative similarity measure in the range $[0, 1]$, where larger values indicate higher similarity.
- **Pearson Correlation Coefficient (PCC).** PCC is also a relative similarity measure returning values in the range $[0, 1]$. Larger values indicate higher similarity.
- **Pearson Correlation of the Histogram of Gradients (HOG).** Used by [**?**] to indicate the intensity of change between two images.
- **Mean Squared Error (MSE).** As an absolute error measure, values close to zero indicate high similarity.

Note that these are only examples without demand for completeness. For further information see [**?**]. Normalizing these measures to yield values in $[0, 1]$ with a sum of one is good practice. Note that defining similarities that are similar to human vision is still an active area of research.

4 Interpreter Manipulation Method Examples

After introducing the terminology of model interpreters and manipulation methods in the previous chapters, this chapter gives detailed information about recent manipulation methods. This section also provides insight into major findings in the field of manipulating model interpretations. First, input level manipulations are discussed, followed by model level manipulations.

4.1 Input Level Manipulations

Fooling both Model and Interpreter.
[**?**] design adversarial attacks that fool the machine learning model as well as the model interpretation. They design targeted and untargeted input patches of various styles, which overlaid onto the original images. **??** shows that GradCAM is fooled by this simple method as well as the calssification model, resembling findings in research on adversarial attacks on models.

(a) Original. Pred: 'French bulldog'. **(b)** Original with Patch. **(c)** Fooled Map, fooled model. Pred: 'Soccer ball'.

Figure 7: Fooling of the model and GradCAM. The interpreter is fooled is it takes the original target classe's features as evidence for the wrong class. Images from [?].

Imperceptible Perturbations significantly alter interpretations.
[**?**] show the pendant of adversarial model attacks for interpretation methods: They apply visually imperceptible perturbations to input images, that do not cause the model to misclassify but that cause the interpreter to yield significantly different interpretations. Perturbed images are constructed by using a generative model minimizing the distance to a target map, see **??**. They also propose to make interpretation results better by smoothing the interpretation process, thus providing a way to undo the fooling of the interpreters (e.g. using β smoothing).

Saliency Maps are vulnerable to Adversarial Attacks. [**?**] showed that importance scores produced by the popular the saliency-based interpretation methods SimpleGrad, DeepLIFT and Integrated Gradients are susceptible even to random perturbations. Contrary to [**?**], the authors argue that the interpretation methods are actually not broken by these perturbations. They state that saliency-map based approaches are very sensitive, thus reacting to the infinitesimal perturbation of an input $\mathbf{x}$ to $\mathbf{x} + \delta$ with an appropriate change in their output.

4.2 Model Level Manipulations

The studies on model manipulation outlined so far have shown that *input perturbations* can make interpretations worse for the same model and interpreter. The articles in this section show that *perturbed model parameters* can also make explanations worse for the same input and interpreter.

Adversarial Model Fine-Tuning Fools Multiple Interpreters.
[**?**] were the first to introduce adversarial model manipulations

Figure 8: On the left, the original image with corresponding interpretation map is shown. The right column shows the imperceptibly perturbed input image and it's explanation. The target interpretation map was chosen to be an image with the text "this explanation was manipulated". Image from [?].

for fooling interpretation methods. The authors adapted the fine-tuning stage of image classification models by using an altered fooling loss function. This loss function is a combination of the standard cross entropy loss function (to maintain the prediction performance) and an additional adversarial term. The adversarial term is used to encourage the interpretation method to give bad interpretations. The results show that the interpretation results are significantly altered while the classification accuracy is maintained. This indicates, that the model is robust to the attack, while the interpretation technique is very sensitive. Two categories of fooling methods are introduced:

- **Passive Fooling**, describing the adaption of the adversarial loss term to fool the interpretation method into highlighting uninformative pixels in the input image. They develop three types for this, namely top$-k$, center and location fooling. Example results from the paper are in **??**, columns two to four. The baseline column shows that the interpretation method applied to the original image highlights pixels within the elephant as highly important for the network prediction. After fine-tuning the model with the adversarial loss, the effect of fooling the interpretation method is visible: Other, rather uninformative pixels are highlighted (see labeled columns two to four in the figure).
- **Active Fooling** is a method with the aim of causing the interpreter to highlight a completely different object in the image, i.e. to make the interpreter actively create false interpretations. This is achieved by fine-tuning the model on input images that contain instances from two classes, say the c_1=*elephant* and c_2=*fire truck* class. The loss with a penalty term that alters the explanations of c_1 and c_2. The effect of this successful active fooling can be observed in **??**, rightmost column.

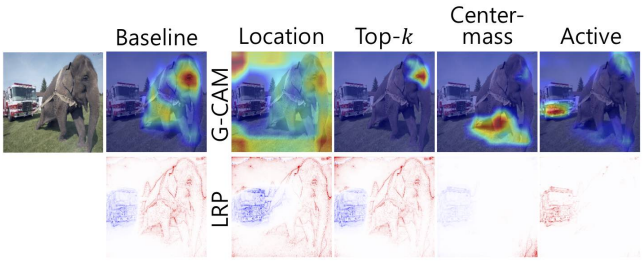


Figure 9: Fooling of LRP and GradCAM. Passive fooling causes the interpreters to highlight wrong, uninformative pixels. Active shifts the interpreters indications from a correct class (*elephant*) to a wrong calss *(fire truck*).

They find that all tested interpreters are fooled to a certain degree, and that the interpreter malfunctions furthermore generalize to other interpreters and especially to the whole validation set. It is worth noting that the proposed attack could not easily fool SmoothGrad, indicating that this is a better, or at least more robust interpretation method.

The main point the authors are trying to make is that machine models can be systematically manipulated to contain unfair biases. They prove that such biases can be explicitly encoded into the loss function during the training stage, which yields an adversarial classifier that will generalizes the learned bias to unseen test samples. This is dangerous, as the bias cannot be uncovered unless one has access to the full training pipeline.

Model-agnostic Interpreters can be gamed and model biases can be hidden. [?] propose a framework for fooling the model-agnostic interpreters LIME and SHAP. The authors take a statistical approach model-agnostic interpretation methods. They examined the data produced by LIME's and SHAP's perturbation schemes and showed that the perturbed samples are out-of-distribution (o.o.d.) samples compared to the distribution of the regular training data. The authors used this insight, i.e. that LIME and SHAP heavily rely on o.o.d. samples and are thus o.o.d. classifiers, to train an adversarial classifier: This classifier exhibits biased behavior (e.g. using the feature 'race' for predicting the income class of a person) on instances from the original data distribution, while using insensitive features for predicting on o.o.d. samples. It is shown that the interpreters are not able to detect the model bias as they create innocuous interpretations. However, LIME performs slightly better than SHAP.

This adversarial framework is applied to numerous real-world datasets, and are thus one of the few papers *not* considering computer vision tasks but rather tasks that actually suit the core motivational concern in XAI: That models might be adversarial und unbiased, and that we might not be able to detect that their decision functions are unfair, racist or discriminatory.

Learning Models Which Conceal Unfairness. [?] examine the relation of interpretation methods and the concept of fairness. They propose to learn a modified model with concealed unfairness. Their

approach differs methodologically to [**?**] as follows: [**?**] adapt the standard cross entropy loss function by taking the gradient of the correct label element from the logits layer, while [**?**] use the gradient of the cross-entropy loss instead. Taking the gradient of the cross-entropy loss conveys more information about other classes, which may contribute to an improved generalization across different interpretation methods and first of all across different test samples. Using this method, they are also able to create adversarial models that focus only on sensitive features which are not informative for the ground truth decision. Again, this hidden bias cannot be detected by the examined interpreters.

5 Benchmarking Interpretations

Checking the robustness, scope and hence the quality of model interpreters has become an indispensable step in explainable machine learning. Evaluating explanation and interpretation methods is difficult as ground truth is mostly lacking. In most applications, it is not known which input features are most important.

[**?**] propose a framework for sanity checking saliency map based interpretation methods. Fer this, a number of randomization tests are introduced along with some visualization techniques and metrics to compare interpretation outputs. The authors find that most methods fail their tests, and warn of the danger of visual assessment. Specifically, they warn that humans tend to chose visually significant appearing images that lack sensitivity to the data distribution and generation as well as to the model. However, their findings are so far limited to the domain of computer vision models and tasks. Implementing their randomization tests for other real-world datasets, such as COMPAS [**?**] or the adult income dataset [**?**] would be an interesting next step.

Additionally, in order to account for adversarial model manipulations, Heo et al. [**?**] propose to expand the criteria for checking the robustness of interpreters further.

6 Conclusion

This paper summarizes the current approaches to manipulating model interpretation methods. The main insights from literature outlined in **??** are the following:

- Saliency-map based interpreters can be tricked even by simple perturbation methods, such is input patches, which also succeed in fooling models. [**?**]
- State-of-the-art interpreters may not be able to detect biases in biased and adversarial models.[**? ?**]
- Biases can be encoded into the model by adapting the loss function and by inexpensively fine-tuning the model. These adaptations can trick the interpretation methods into yielding wrong results while models remain accurate. [**?**]
- [**?**] show the pendant of adversarial model attacks for interpretation methods: They apply visually imperceptible perturbations to input images, that do not cause the models to misclassify but that cause the interpreter to yield significantly different interpretations.

On the one hand, the findings suggest that our models aro not fully aligned with how human information processing works. If machine learning interpretation models would decide by the criteria we humans employ for tasks such as image classification, there would be no fooling of interpretation models by input or model manipulations. On the other hand, it was shown that advances in machine learning models has led to models that rely too much on the data they are trained on (the i.i.d. assumption), thus showing a high susceptibility to o.o.d. properties or properties that are highly correlated with labels in the dataset but are not distinctive in the real world (such as image backgrounds) in the first place. Models and interpreters can still be misled in a large and systematic manner.

A growing number of studies gives evidence for model how interpretation methods can be gamed. Among these are the studies outlined in **??**. Other studies also raise concerns about if standard deep learning practices are valid, such as the work on fooling the broadly used attention mechanism [**?**]. However, findings about manipulating interpretations do not suggest that interpretations are completely meaningless, just as adversarial attacks on predictions models do not imply that machine learning models are useless. Nonetheless, they suggest that there still are fundamental flaws in the way neural networks operate und that much caution and supervision should be applied when deploying them in the real world. This paper follows the footsteps of [**?**], trying to caution against blindly putting faith into post-hoc explanation methods. Moreover, we propose that checking the robustness of interpretation methods not only with respect to adversarial input manipulations but also with respect to adversarial model manipulation should be a necessary proof of concept.

While there exists a number of review papers on XAI and it's various subfields, this report is to the best of our knowledge the first one to comprehensively review manipulation methods for interpreters. We believe that identifying risks and adversaries helps to open up research on more robust interpretation methods.

Future Work.
We see several possible future directions of future work. Firstly, for approaching the discrepancy of in-lab and real-life applications of machine learning, more focus ought to be laid on the development of better performance metrics for both measuring the performance of machine learning models as well as their interpreters. More specifically, it might be fruitful to further investigate the correlation between o.o.d. samples and the performance of an interpretation method. So far, most of these findings are limited to specific experimental settings (e.g. most research on interpretability is focused on computer vision tasks). Further research should much more explore real world datasets and tasks. The relationship between different interpretation techniques and the dependence of interpretation susceptibility on model class, interpretation method, and task type and dataset structure should also be thoroughly investigated.