Foiling Explanations in Deep Neural Networks

Anonymous authors
Paper under double-blind review

Abstract

Deep neural networks (DNNs) have greatly impacted numerous fields over the past decade. Yet despite exhibiting superb performance over many problems, their black-box nature still poses a significant challenge with respect to explainability. Indeed, explainable artificial intelligence (XAI) is crucial in several fields, wherein the answer alone—sans a reasoning of how said answer was derived—is of little value. This paper uncovers a troubling property of explanation methods for image-based DNNs: by making small visual changes to the input image—hardly influencing the network's output—we demonstrate how explanations may be arbitrarily manipulated through the use of evolution strategies. Our novel algorithm, AttaXAI, a model-and-data XAI-agnostic, adversarial attack on XAI algorithms, only requires access to the output logits of a classifier and to the explanation map; these weak assumptions render our approach highly useful where real-world models and data are concerned. We compare our method's performance on two benchmark datasets—CIFAR100 and ImageNet—using four different pretrained deep-learning models: VGG16-CIFAR100, VGG16-ImageNet, MobileNet-CIFAR100, and Inception-v3-ImageNet. We find that the XAI methods can be manipulated without the use of gradients or other model internals. Our novel algorithm is successfully able to manipulate an image in a manner imperceptible to the human eye, such that the XAI method outputs a specific explanation map. To our knowledge, this is the first such method in a black-box setting, and we believe it has significant value where explainability is desired, required, or legally mandatory.

Keywords: deep learning, computer vision, adversarial attack, evolutionary algorithm, explainable artificial intelligence

1 Introduction

Recent research has revealed that deep learning-based, image-classification systems are vulnerable to adversarial instances, which are designed to deceive algorithms by introducing perturbations to benign images Carlini & Wagner (2017); Madry et al. (2017); Xu et al. (2018); Goodfellow et al. (2014); Croce & Hein (2020). A variety of strategies have been developed to generate adversarial instances, and they fall under two broad categories, differing in the underlying threat model: white-box attacks Moosavi-Dezfooli et al. (2016); Kurakin et al. (2018) and black-box attacks Chen et al. (2017); Lapid et al. (2022).

In a white box attack, the attacker has access to the model's parameters, including weights, gradients, etc'. In a black-box attack, the attacker has limited information or no information at all; the attacker generates adversarial instances using either a different model, a model's raw output (also called logits), or no model at all, the goal being for the result to transfer to the target model Tramèr et al. (2017); Inkawhich et al. (2019).

In order to render a model more interpretable, various explainable algorithms have been conceived. Van Lent et al. (2004) coined the term *Explainable Artificial Intelligence* (XAI), which refers to AI

systems that "can explain their behavior either during execution or after the fact". In-depth research into XAI methods has been sparked by the success of Machine Learning (ML) systems, particularly Deep Learning (DL), in a variety of domains, and the difficulty in intuitively understanding the outputs of complex models, namely, how did a DL model arrive at a specific decision for a given input.

Explanation techniques have drawn increased interest in recent years due to their potential to reveal hidden properties of deep neural networks Došilović et al. (2018). For safety-critical applications, interpretability is essential, and sometimes even legally required.

The importance assigned to each input feature for the overall classification result may be observed through explanation maps, which can be used to offer explanations. Such maps can be used to create defenses and detectors for adversarial attacks Walia et al. (2022); Fidel et al. (2020); Kao et al. (2022). Figures 1 and 2 show examples of explanation maps, generated by five different methods discussed in Section 2.

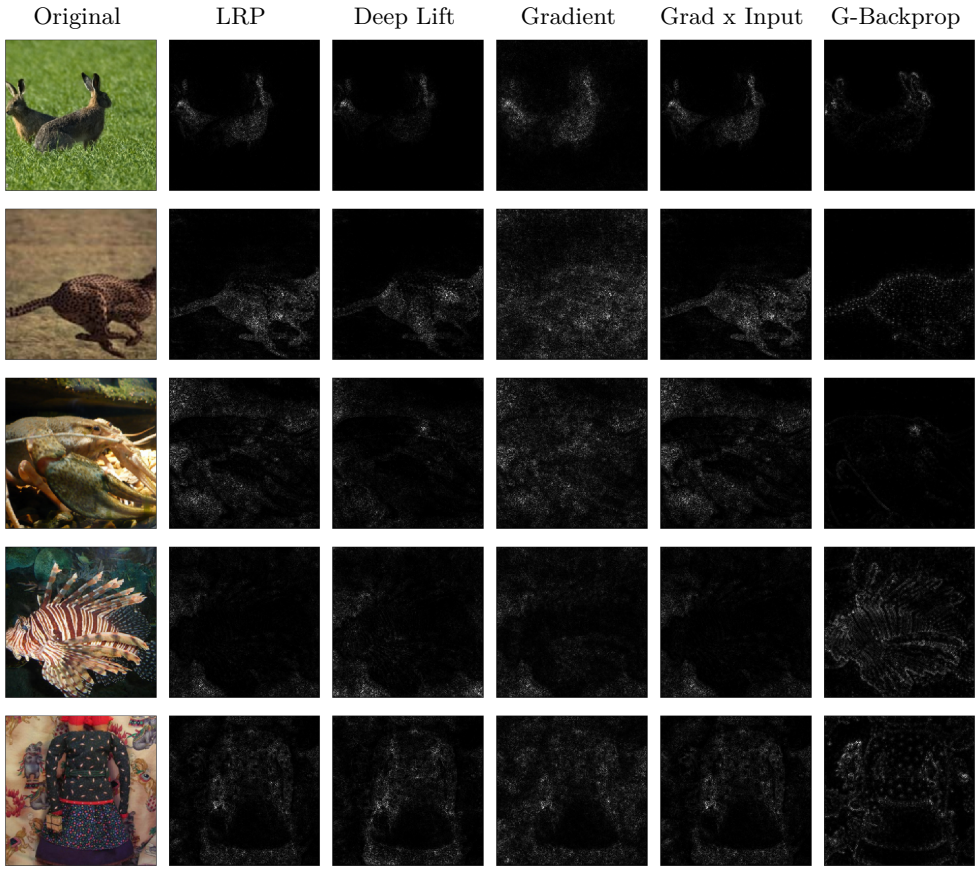

Figure 1: Explanation maps for 5 images using 5 different explanation methods. Dataset: ImageNet. Model: VGG16.

In this paper, we show that these explanation maps can be transformed into any target map, using only the maps and the network's output probability vector. This is accomplished by adding a perturbation to the input that is scarcely (if at all) noticeable to the human eye. This perturbation has minimal effect on the neural network's output, therefore, in addition to the classification outcome, the probability vector of all classes remains virtually identical.

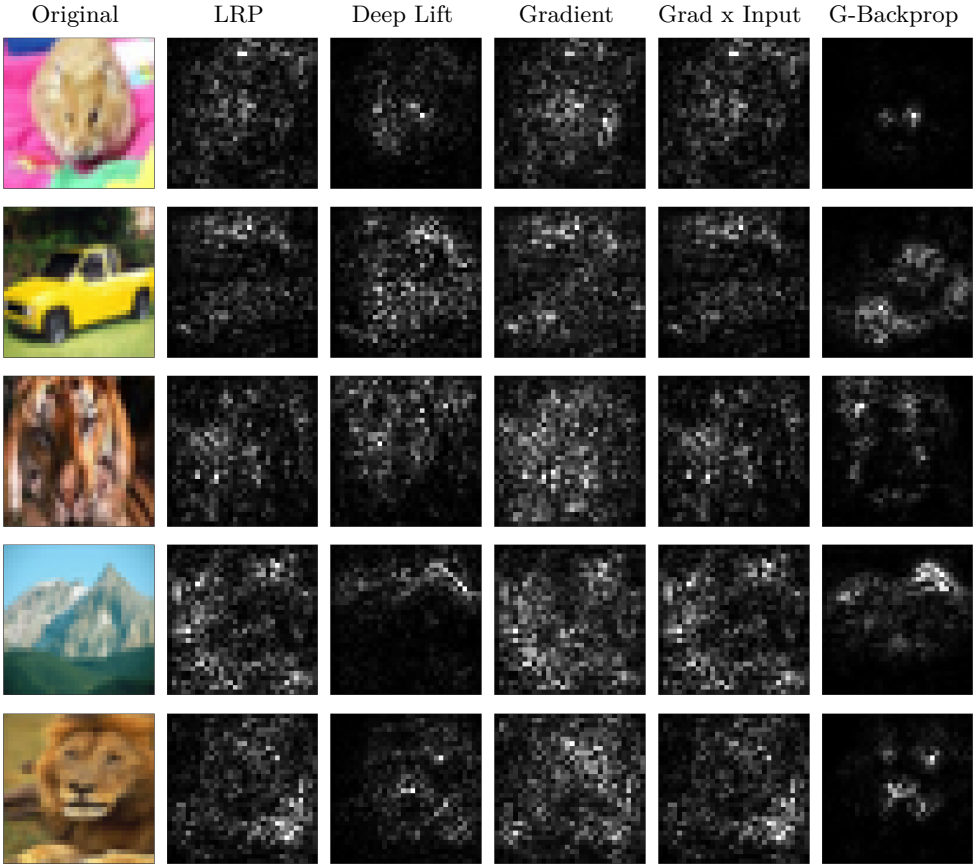

Figure 2: Explanation maps for 5 images using 5 different explanation methods. Dataset: CIFAR100. Model: VGG16.

Our contribution. As we mentioned earlier, recent studies have shown that adversarial attacks may be used to undermine DNN predictions Goodfellow et al. (2014); Papernot et al. (2016b); Carlini & Wagner (2017); Lapid et al. (2022). There are several more papers regarding XAI attacks, which have also shown success on manipulating XAI, but to our knowledge, they all rely on access to the neural network's gradient, which is usually not available in real-world scenarios Ghorbani et al. (2019); Dombrowski et al. (2019).

Herein, we propose a black-box algorithm, AttaXAI, which enables manipulation of an image through a barely noticeable perturbation, without the use of any model internals, such that the explanation fits any given target explanation. Further, the robustness of the XAI techniques are tested as well. We study AttaXAI's efficiency on 2 benchmark datasets, 4 different models, and 5 different XAI methods.

The next section presents related work. Section 3 presents AttaXAI, followed by experiments and results in Section 4. In Section 5 we discuss our results, followed by concluding remarks in Section 6.

2 Related Work

The ability of explanation maps to detect even the smallest visual changes was shown by Ghorbani et al. (2019), where they perturbed a given image, which caused the explanatory map to change, without any specific target.

Kuppa & Le-Khac (2020) designed a black-box attack to examine the security aspects of the gradient-based XAI approach, including consistency, accuracy, and confidence, using tabular datasets.

Zhang et al. (2020) demonstrated a class of white-box attacks that provide adversarial inputs, which deceive both the interpretation models and the deep-learning models. They studied their method using four different explanation algorithms.

Xu et al. (2018) demonstrated that a subtle adversarial perturbation intended to mislead classifiers might cause a significant change in a class-specific network interpretability map.

The goal of Dombrowski et al. (2019) was to precisely replicate a given target map using gradient descent with respect to the input image. Although this work showed an intriguing phenomenon, it is a less-realistic scenario, since the attacker has full access to the targeted model.

We aimed to veer towards a more-realistic scenario and show that we can achieve similar results using no information about the model besides the probability output vector and the explanation map. To our knowledge, our work is the first to introduce a black-box attack on XAI gradient-based methods in the domain of image classification.

We will employ the following explanation techniques in this paper:

1. **Gradient**: Utilizing the saliency map, $g(x) = \frac{\partial f}{\partial x}(x)$, one may measure how small perturbations in each pixel alter the prediction of the model, $f(x)$ Simonyan et al. (2013).

2. **Gradient $\times$ Input**: The explanation map is calculated by multiplying the input by the partial derivatives of the output with regard to the input, $g(x) = \frac{\partial f}{\partial x}(x) \odot x$ Shrikumar et al. (2016).

3. **Guided Backpropagation**: A variant of the Gradient explanation, where the gradient's negative components are zeroed while backpropagating through the non-linearities of the model Springenberg et al. (2014).

4. **Layer-wise Relevance Propagation (LRP)**: With this technique, pixel importance propagates backwards across the network from top to bottom Bach et al. (2015); Montavon et al. (2019). The general propagation rule is the following:

$$R_j = \sum_k \frac{\alpha_j \rho(w_{jk})}{\epsilon + \sum_{0,j} \alpha_j \rho(w_{jk})} R_k, \tag{1}$$

   where j and k are two neurons of any two consecutive layers, R_k, R_j are the relevance maps of layers k and j, respectively, ρ is a function that transforms the weights, and ϵ is a small positive increment.

   In order to propagate relevance scores to the input layer (image pixels), the method applies an alternate propagation rule that properly handles pixel values received as input:

$$R_i = \sum_j \frac{\alpha_i w_{ij} - l_i w_{ij}^+ - h_i w_{ij}^-}{\sum_i \alpha_i w_{ij} - l_i w_{ij}^+ - h_i w_{ij}^-} R_j, \tag{2}$$

   where l_i and h_i are the lower and upper bounds of pixel values.

5. **Deep Learning Important FeaTures (DeepLIFT)** compares a neuron's activation to its "reference activation", which then calculates contribution scores based on the difference. DeepLIFT has the potential to separately take into account positive and negative contributions, which might help to identify dependencies that other methods might have overlooked Shrikumar et al. (2017).

3 AttaXAI

This section presents our algorithm, AttaXAI, discussing first evolution strategies, and then delving into the algorithmic details.

3.1 Evolution Strategies

Our algorithm is based on Evolution Strategies (ES), a family of heuristic search techniques that draw their inspiration from natural evolution Beyer & Schwefel (2002); Hansen et al. (2015). Each iteration (aka generation) involves perturbing (through mutation) a population of vectors (genotypes) and assessing their objective function value (fitness value). The population of the next generation is created by combining the vectors with the highest fitness values, a process that is repeated until a stopping condition is met.

AttaXAI belongs to the class of Natural Evolution Strategies (NES) Wierstra et al. (2014); Glasmachers et al. (2010), which includes several algorithms in the ES class that differ in the way they represent the population, and in their mutation and recombination operators. With NES, the population is sampled from a distribution π_{ψ_t}, which evolves through multiple iterations (generations); we denote the population samples by Z_t. Through stochastic gradient descent NES attempts to maximize the population's average fitness, $\mathbb{E}_{Z\sim\pi_\psi}[f(Z)]$, given a fitness function, $f(\cdot)$.

A version of NES we found particularly useful for our case was used to solve common reinforcement learning (RL) problems Salimans et al. (2017).

Search Gradients. The core idea of NES is to use search gradients to update the parameters of the search distribution Wierstra et al. (2014). The search gradient can be defined as the gradient of the expected fitness: Denoting by π a distribution with parameters ψ, $\pi(z|\psi)$ is the probability density function of a given sample z. With $f(z)$ denoting the fitness of a sample z, the expected fitness under the search distribution can be written as:

$$J(\psi) = \mathbb{E}_\psi[f(z)] = \int f(z)\pi(z|\psi)dz. \tag{3}$$

The gradient with respect to the distribution parameters can be expressed as:

$$\begin{aligned}
\nabla_\theta J(\theta) &= \nabla_\theta \int f(z)\pi(z|\theta)dz \\
&= \int f(z)\nabla_\theta \pi(z|\theta)dz \\
&= \int f(z)\nabla_\theta \pi(z|\theta)\frac{\pi(z|\theta)}{\pi(z|\theta)}dz \\
&= \int [f(z)\nabla_\theta \log \pi(z|\theta)]\pi(z|\theta)dz \\
&= \mathbb{E}_\theta[f(z)\nabla_\theta \log \pi(z|\theta)]
\end{aligned}$$

From these results we can approximate the gradient with Monte Carlo Metropolis & Ulam (1949) samples $z_1, z_2, ..., z_\lambda$:

$$\nabla_\theta J(\theta) \approx \frac{1}{\lambda}\sum_{k=1}^{\lambda} f(z_k)\nabla_\theta \log \pi(z_k|\theta) \tag{4}$$

In our experiment we sampled from a Gaussian distribution for calculating the search gradients.

Latin Hypercube Sampling (LHS). LHS is a form of stratified sampling scheme, which improves the coverage of the sampling space. It is done by dividing a given cumulative distribution function into M non-overlapping intervals of equal y-axis length, and randomly choosing one value from each interval to obtain M samples. It ensures that each interval contains the same number of samples, thus producing good uniformity and symmetry Wang et al. (2022). We used both LHS and standard sampling in our experimental setup.

3.2 Algorithm

AttaXAI is an evolutionary algorithm (EA), which explores a space of images for adversarial instances that fool a given explanation method. This space of images is determined by a given input image, a model, and a loss function. The algorithm generates a perturbation for the given input image such that it fools the explanation method.

More formally, we consider a neural network, $f\colon \mathbb{R}^{h,w,c} \to \mathbb{R}^K$, which classifies a given image, $x \in \mathbb{R}^{h,w,c}$, where h, w, c are the image's height, width, and channel count, respectively, to one of K predetermined categories, with the predicted class given by $k = \arg\max_i f(x)_i$. The explanation map, which is represented by the function, $g\colon \mathbb{R}^{h,w,c} \to \mathbb{R}^{h,w}$, links each image to an explanation map, of the same height and width, where each coordinate specifies the influence of each pixel on the network's output.

AttaXAI explores the space of images through evolution, ultimately producing an adversarial image; it does so by continually updating a Gaussian probability distribution, used to sample the space of images. By continually improving this distribution the search improves.

We begin by sampling perturbations from an isotropic normal distribution $\mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$. Then we add them to the original image, x, and feed them to the model. By doing so, we can approximate the gradient of the expected fitness function. With an approximation of the gradient at hand we can advance in that direction by updating the search distribution parameters. A schematic of our algorithm is shown in Figure 3, with a full pseudocode provided in Algorithm 1.

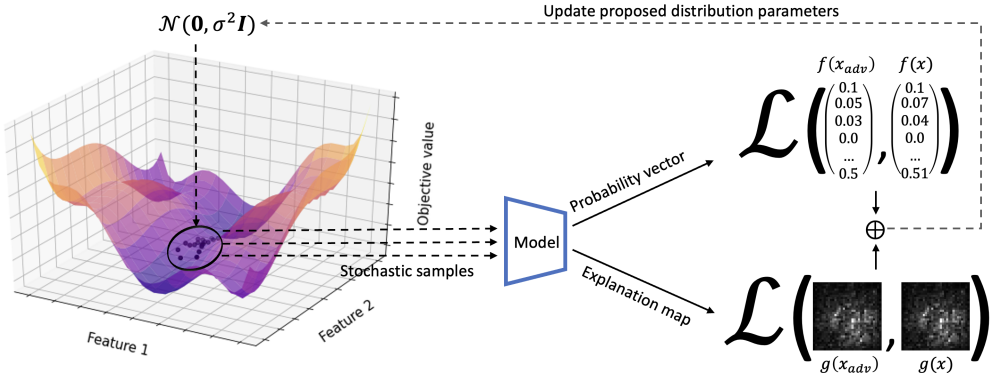

Figure 3: Schematic of proposed algorithm. Individual perturbations are sampled from the population's distribution $\mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$, and feed into the model (Feature 1 and Feature 2 are image features, e.g., two pixel values; in reality the dimensionality is much higher). Then, the fitness function, i.e. the loss, is calculated using the output probability vectors and the explanation maps to approximate the gradient and update the distribution parameters.

Algorithm 1 AttaXAI

Input:

$x \leftarrow$ original image
$y \leftarrow$ original image's label
$x_{expl} \leftarrow$ target explanation map
$x_{pred} \leftarrow$ logits values of x
$model \leftarrow$ model to be used
$G \leftarrow$ maximum number of generations
$\lambda \leftarrow$ population size
$\sigma \leftarrow$ initial standard deviation value
$\alpha \leftarrow$ explanation loss weight
$\beta \leftarrow$ prediction loss weight
$\eta_{\widehat{x}} \leftarrow$ mean learning rate
$\eta_\sigma \leftarrow$ standard deviation learning rate

Output:

$\widehat{x} \leftarrow$ adversarial image

Main loop
$\widehat{x} \leftarrow x$
for $g = 1, 2, ..., G$ **do**
  for $k = 1, 2, ..., \lambda$ **do**
    draw sample $z_k \sim \mathcal{N}(\widehat{x}, \sigma^2\mathbf{I})$ # z_k is used to perturb an image
    evaluate fitness $f(z_k) = \text{FITNESS}(z_k)$
    calculate log-derivative $\nabla_{\widehat{x}} \log \mathcal{N}(z_k|\widehat{x}, \sigma^2)$
    calculate log-derivative $\nabla_\sigma \log \mathcal{N}(z_k|\widehat{x}, \sigma^2)$
  $\nabla_{\widehat{x}} J = \frac{1}{\lambda} \sum_{i=1}^{\lambda} f(z_k) \nabla_{\widehat{x}} \log \mathcal{N}(z_k|\widehat{x}, \sigma^2)$
  $\nabla_\sigma J = \frac{1}{\lambda} \sum_{i=1}^{\lambda} f(z_k) \nabla_\sigma \log \mathcal{N}(z_k|\widehat{x}, \sigma^2)$
  $\widehat{x} = \widehat{x} + \eta_V \cdot \nabla_{\widehat{x}} J$
  $\sigma = \sigma + \eta_\sigma \cdot \nabla_\sigma J$

function FITNESS(z)
  $z_{expl} = XAI(z, y)$
  $z_{pred} = model(z)$
  $expl_{loss} = \|x_{expl} - z_{expl}\|_2^2$
  $pred_{loss} = \|x_{pred} - z_{pred}\|_2^2$
  return $\alpha * expl_{loss} + \beta * pred_{loss}$

3.3 Fitness Function

Given an image, $x \in \mathbb{R}^{h,w,c}$, a specific explanation method, $g\colon \mathbb{R}^{h,w,c} \to \mathbb{R}^{h,w}$, a target image, x_{target}, and a target explanation map, $g(x_{target})$, we seek an adversarial perturbation, $\delta \in \mathbb{R}^{h,w,c}$, such that the following properties of the adversarial instance, $x_{adv} = x + \delta$, hold:

1. The network's prediction remains almost constant, i.e., $f(x) \approx f(x_{adv})$.
2. The explanation vector of x_{adv} is close to the target explanation map, $g(x_{target})$, i.e., $g(x_{adv}) \approx g(x_{target})$.
3. The adversarial instance, x_{adv}, is close to the original image, x, i.e., $x \approx x_{adv}$.

We achieve such perturbations by optimizing the following fitness function of the evolutionary algorithm:

$$\mathcal{L} = \alpha \|g(x_{adv}) - g(x_{target})\| + \beta \|f(x_{adv}) - f(x)\| \tag{5}$$

The first term ensures that the altered explanation map, $g(x_{adv})$, is close to the target explanation map, $g(x_{target})$; the second term pushes the network to produce the same output probability vector. The hyperparameters, $\alpha, \beta \in \mathbb{R}^+$, determine the respective weightings of the fitness components.

In order to use our approach, we only need the output probability vector, $f(x_{adv})$, and the target explanation map, $g(x_{target})$. Unlike white-box methods, we do not presuppose anything about the targeted model, its architecture, dataset, or training process. This makes our approach more realistic.

Minimizing the fitness value is the ultimate objective. Essentially, the value is better if the proper class's logit remains the same and the explanation map looks similar to the targeted explanation map:

$$\underset{x_{adv}}{\operatorname{argmin}} \mathcal{L} = \alpha \|g(x_{adv}) - g(x_{target})\| + \beta \|f(x_{adv}) - f(x)\| \tag{6}$$

4 Experiments and Results

Assessing the algorithm over a particular configuration of model, dataset, and explanation technique, involves running it over 100 pairs of randomly selected images. We used 2 datasets: CIFAR100 and ImageNet Deng et al. (2009). For CIFAR100 we used the VGG16 Simonyan & Zisserman (2014) and MobileNet Howard et al. (2017) models, and for ImageNet we used VGG16 and Inception Szegedy et al. (2015); the models are pretrained. For ImageNet, VGG16 has an accuracy of 73.3% and Inception-v3 has an accuracy of 78.8%. For CIFAR100, VGG16 has an accuracy of 72.9% and MobileNet has an accuracy of 69.0% (these are top-1 accuracy values; for ImageNet, top5 accuracy values are: VGG16 – 91.5%, Inception-v3 – 94.4%, and for CIFAR100, VGG16 – 91.2%, MobileNet – 91.0%). We chose these models because they are commonly used in the Computer Vision community for many downstream tasks Haque et al. (2019); Bhatia et al. (2019); Ning et al. (2017); Younis et al. (2020); Venkateswarlu et al. (2020).

The experimental setup is summarized in Algorithm 2: Choose 100 random image pairs from the given dataset. For each image pair compute a target explanation map, $g(x_{target})$, for one of the two images. With a budget of $50,000$ queries to the model, Algorithm 1 perturbs the second image, aiming to replicate the desired $g(x_{target})$. We assume the model outputs both the output probability vector and the explanation map per each query to the model—which is a realistic scenario nowadays, with XAI algorithms being part of real-world applications Payrovnaziri et al. (2020); Giuste et al. (2022); Tjoa & Guan (2020).

Algorithm 2 Experimental setup (per dataset and model)

Input:

dataset ← dataset to be used
model ← model to be used
G ← maximum number of generations
λ ← population size
σ ← initial standard deviation value
α ← explanation-loss weight
β ← prediction-loss weight

Output:

Performance scores

1: **for** $i \leftarrow 1$ to *100* **do**
2: Randomly choose a pair of images x and x_{target} from *dataset*
3: Generate x_{adv} by running Algorithm 1, with x and x_{target} (and all other input parameters)
4: Save performance statistics

We have two different weighting hyperparameters for the two datasets, which have been empirically proven to work: $\alpha = 1e11, \beta = 1e6$ for ImageNet, and $\alpha = 1e7, \beta = 1e6$ for CIFAR100. After every generation the learning rate was decreased through multiplication by a factor of 0.999. We tested

drawing the population samples, both independent and identically distributed (iid) and through Latin hypercube sampling (LHS). The generation of the explanations was achieved by using the repository Captum Kokhlikyan et al. (2020), a unified and generic model interpretability library for PyTorch.

Figures 4 through 7 shows samples of our results. Specifically, Figure 4 shows AttaXAI-generated attacks for images from ImageNet using the VGG16 model, against each of the 5 explanation methods: LRP, Deep Lift, Gradient, Gradient x Input, Guided-Backpropagation; Figure 5 shows AttaXAI-generated attacks for images from ImageNet using the Inception model; Figure 6 shows AttaXAI-generated attacks for images from CIFAR100 using the VGG16 model; and Figure 7 shows AttaXAI-generated attacks for images from CIFAR100 using the MobileNet model.

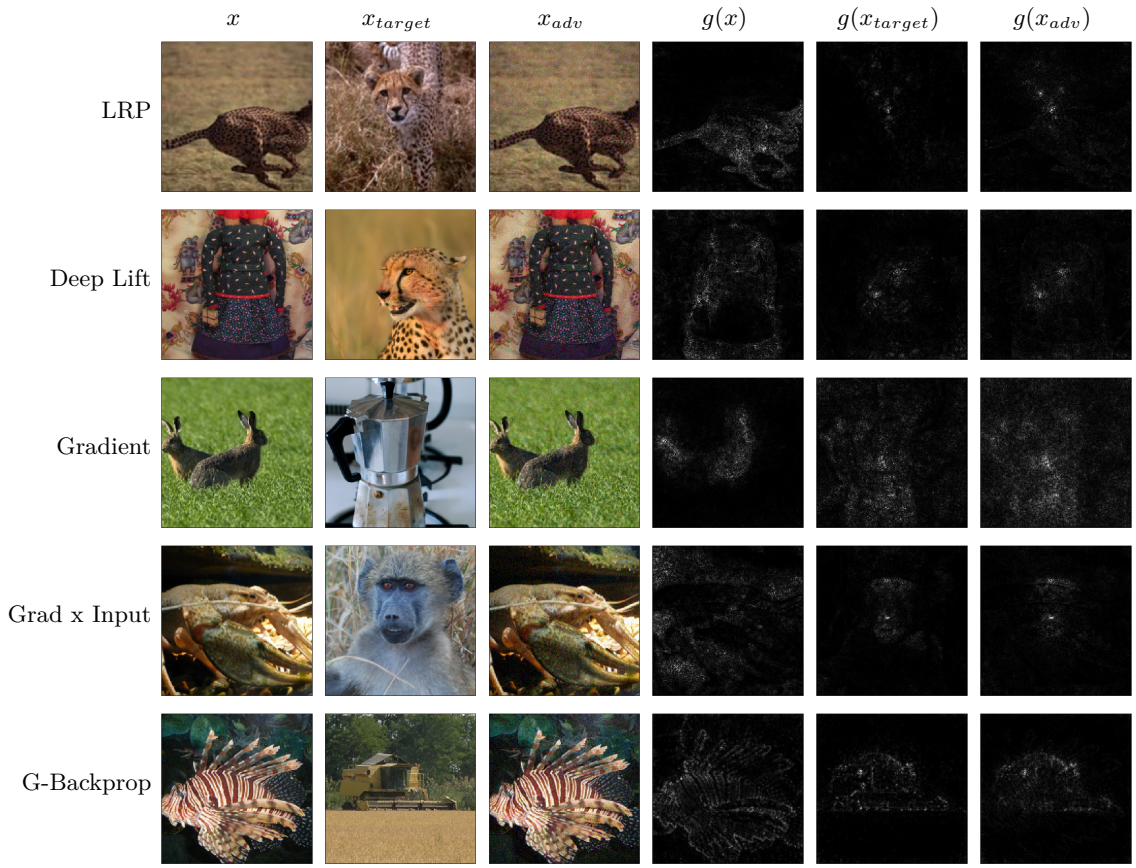

Figure 4: Attacks generated by AttaXAI. Dataset: ImageNet. Model: VGG16. Shown for the 5 explanation methods, described in the text: LRP, Deep Lift, Gradient, Gradient x Input, Guided Backpropagation (denoted G-Backprop in the figure). Note that our primary objective has been achieved: having generated an adversarial image (x_{adv}), virtually identical to the original (x), the explanation (g) of the adversarial image (x_{adv}) is now, incorrectly, that of the target image (x_{target}); essentially, the two rightmost columns are identical.

Note that our primary objective has been achieved: having generated an adversarial image (x_{adv}), virtually identical to the original (x), the explanation (g) of the adversarial image (x_{adv}) is now, incorrectly, that of the target image (x_{target})—essentially, the two rightmost columns of Figures 4-7 are identical; furthermore, the class prediction remains the same, i.e., $\arg\max_i f(x)_i = \arg\max_i f(x_{adv})_i$.

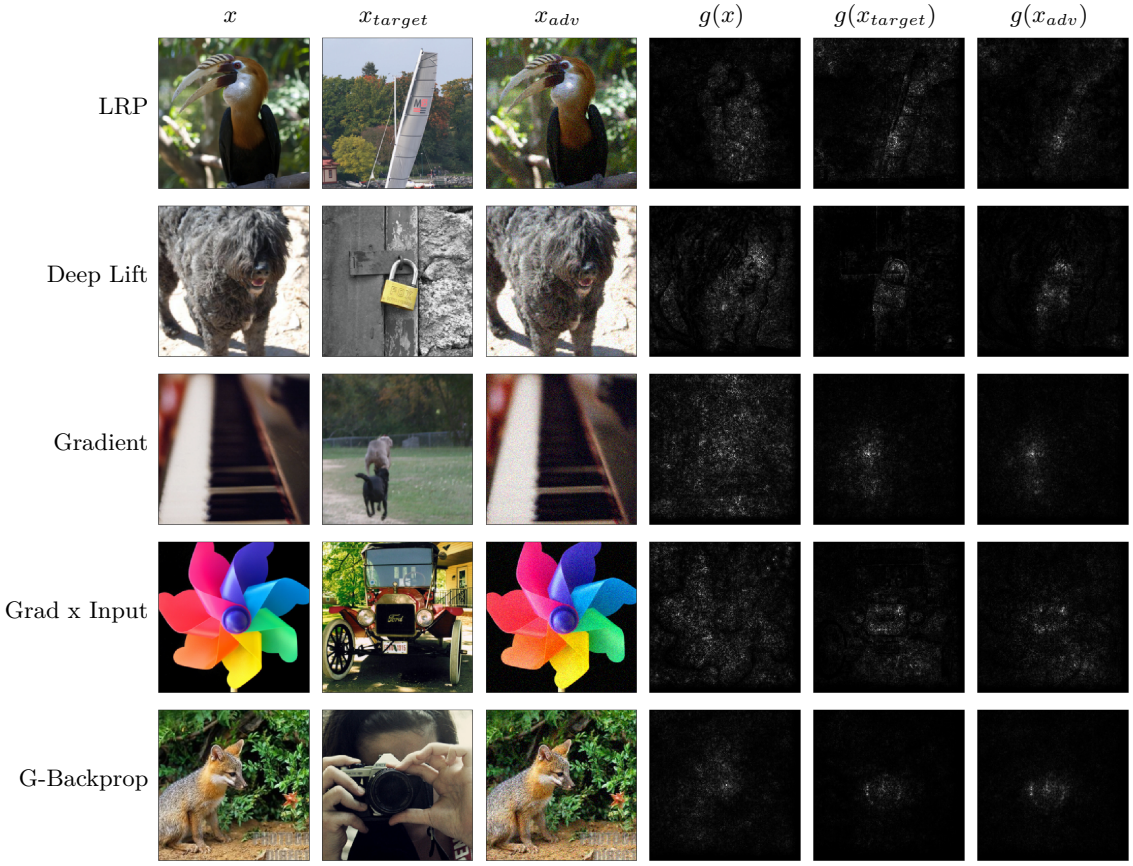

Figure 5: Attacks generated by AttaXAI. Dataset: ImageNet. Model: Inception.

5 Discussion

We examined the multitude of runs in-depth, producing several graphs, which are provided in full in the Appendix. Herein we summarize several observations we made:

- Our algorithm was successful in that $f(x_{adv}) \approx f(x)$ for all x_{adv} generated, and when applying arg max the original label remained unchanged.
- For most hyperparameter values examined, our approach converges for ImageNet using VGG, for every XAI except Guided Backpropagation—which was found to be more robust then other techniques in this configuration.
- For all the experiments we witnessed that the Gradient XAI method showed the smallest mean squared error (MSE) between $g(x_{adv})$ and $g(x_{target})$, i.e., it was the least robust. The larger the MSE between $g(x_{adv})$ and $g(x_{target})$ the better the explanation algorithm can handle our perturbed image.
- For VGG16 (Figures 8 and 14), Gradient XAI showed the smallest median MSE between $g(x_{adv})$ and $g(x_{target})$, while Guided Backpropogation showed the most. This means that using Gradient XAI's output as an explanation incurs the greatest risk, while using Guided Backpropogation's output as an explanation incurs the smallest risk.
- For Inception (Figure 11), Gradient XAI and Guided Backpropogation exhibited the smallest median MSE, while LRP, Gradient x Input, and Deep Lift displayed similar results.

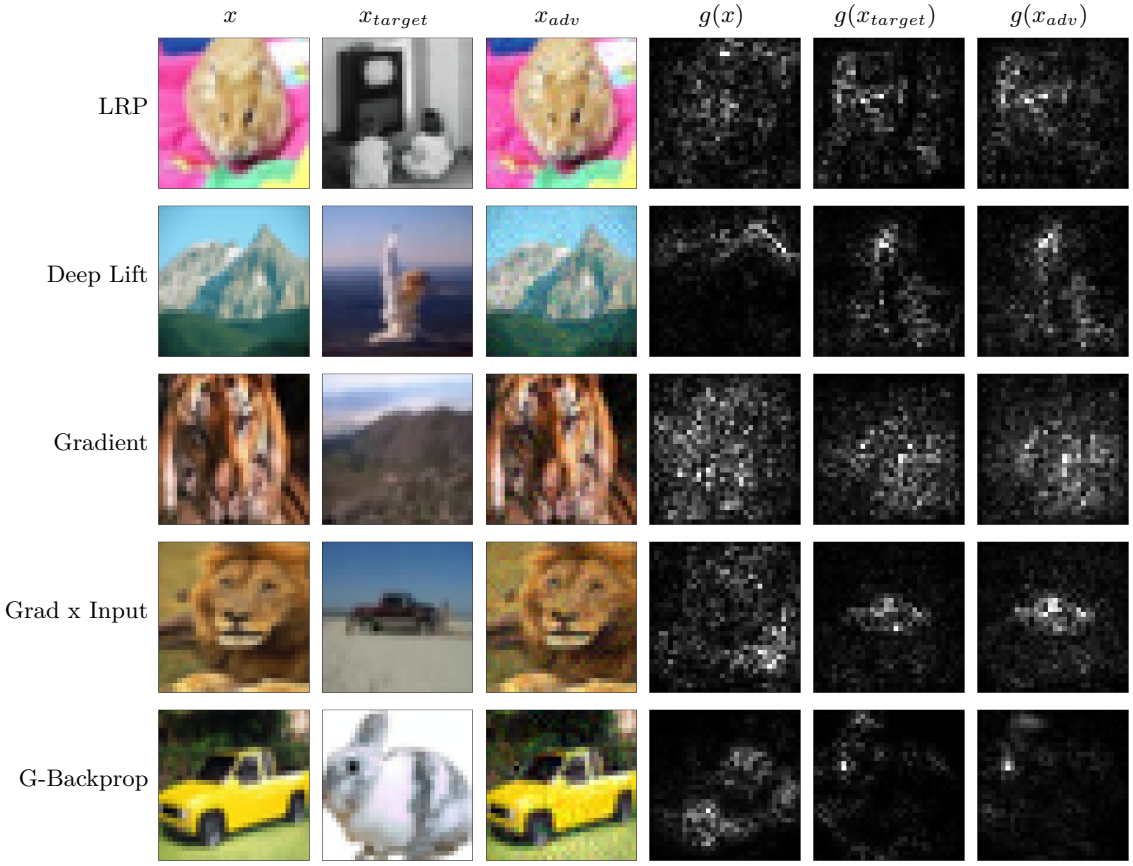

Figure 6: Attacks generated by AttaXAI. Dataset: CIFAR100. Model: VGG16.

- For the MobileNet (Figure 17), Gradient XAI exhibited the smallest MSE, while Guided Backpropogation showed the largest MSE—rendering it more robust than other techniques in this configuration.

- MobileNet is more robust than VGG16 in that it attains higher MSE scores irrespective of the XAI method used. We surmise that this is due to the larger number of parameters in VGG16.

- The query budget was 50,000 for all experiments. In many runs the distance between the target explanation and the adversarial explanation reached a plateau after roughly 25,000 queries.

6 Concluding Remarks

Recently, practitioners have started to use explanation approaches more frequently. We demonstrated how focused, undetectable modifications to the input data can result in arbitrary and significant adjustments to the explanation map. We showed that explanation maps of several known explanation algorithms may be modified at will. Importantly, this is feasible with a black-box approach, while maintaining the output of the model. We tested AttaXAI against the ImageNet and CIFAR100 datasets using 4 different network models.

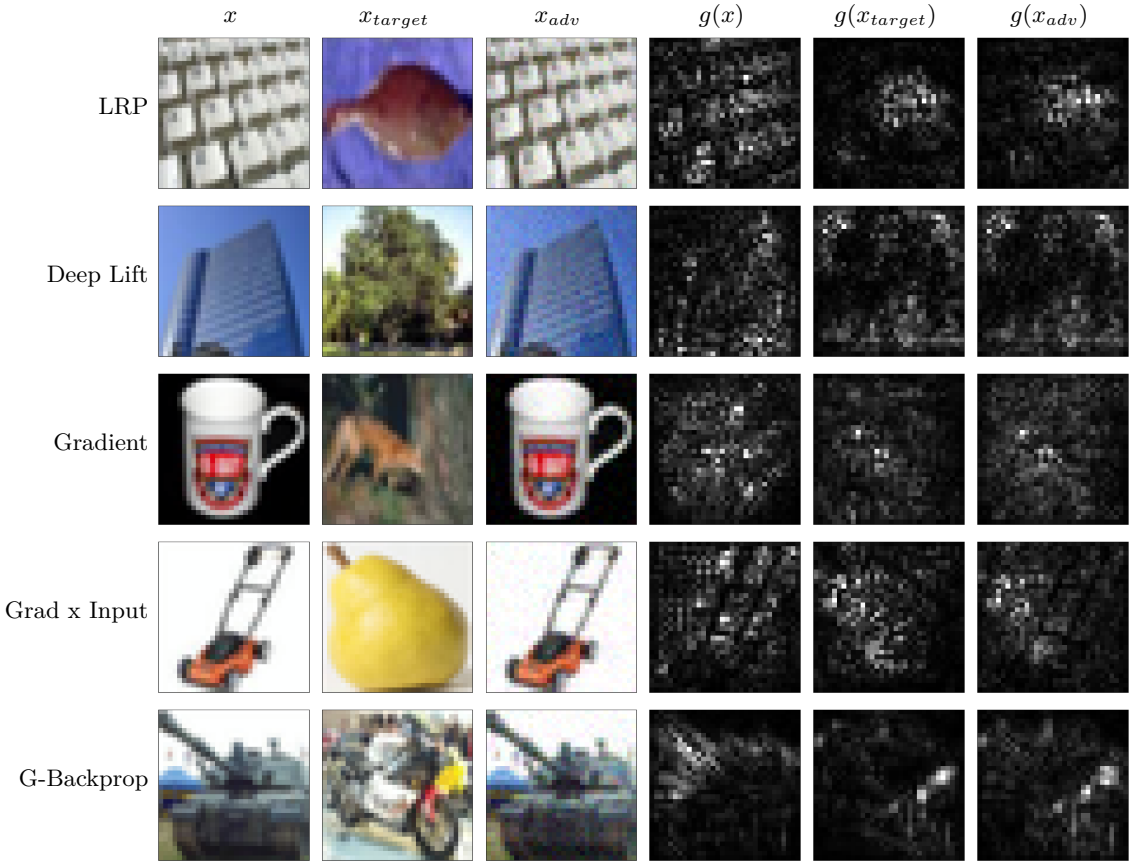

Figure 7: Attacks generated by AttaXAI. Dataset: CIFAR100. Model: MobileNet.

XAI adversarial samples are hardly perceptible to the human eye. It is obvious that neural networks operate quite differently from humans, capturing fundamentally distinct properties. In addition, further work is required in the XAI domain to make XAI algorithms that are more reliable.

This work has shown that explanations are easily foiled—without any recourse to internal information—raising questions regarding XAI-based defenses and detectors.

This work has also investigated the robustness of various XAI methods, revealing that Gradient XAI is the least robust XAI method and Guided Backpropagation is the most robust one.

Future Suggestions In our study we examined how to attack a model's (XAI) explanation for a given input, prediction, and XAI method. Some questions still remain:

- A way to predict whether a XAI attack will be successful, and how many queries will be needed.
- A better metric for a successful XAI attack, since in our results we observed that a smaller L2 distance does not necessarily translate to a more "convincing" attack.
- Find a way to eliminate the need for model feedback, i.e., go fully black-box. Applying XAI attacks via transferability Papernot et al. (2016a); Xie et al. (2019); Wang et al. (2021) might be a way to move forward.
- When developing new XAI methods find ways to render them more robust to adversarial attacks.

References

Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS One*, 10(7):e0130140, 2015.

Hans-Georg Beyer and Hans-Paul Schwefel. Evolution strategies—a comprehensive introduction. *Natural Computing*, 1(1):3–52, 2002.

Yajurv Bhatia, Aman Bajpayee, Deepanshu Raghuvanshi, and Himanshu Mittal. Image captioning using Google's Inception-Resnet-v2 and recurrent neural network. In *2019 Twelfth International Conference on Contemporary Computing (IC3)*, pp. 1–6. IEEE, 2019.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 39–57. IEEE, 2017.

Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pp. 15–26, 2017.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International Conference on Machine Learning*, pp. 2206–2216. PMLR, 2020.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255. IEEE, 2009.

Ann-Kathrin Dombrowski, Maximillian Alber, Christopher Anders, Marcel Ackermann, Klaus-Robert Müller, and Pan Kessel. Explanations can be manipulated and geometry is to blame. *Advances in Neural Information Processing Systems*, 32, 2019.

Filip Karlo Došilović, Mario Brčić, and Nikica Hlupić. Explainable artificial intelligence: A survey. In *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pp. 0210–0215. IEEE, 2018.

Gil Fidel, Ron Bitton, and Asaf Shabtai. When explainability meets adversarial learning: Detecting adversarial examples using shap signatures. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE, 2020.

Amirata Ghorbani, Abubakar Abid, and James Zou. Interpretation of neural networks is fragile. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 3681–3688, 2019.

Felipe Giuste, Wenqi Shi, Yuanda Zhu, Tarun Naren, Monica Isgut, Ying Sha, Li Tong, Mitali Gupte, and May D Wang. Explainable artificial intelligence methods in combating pandemics: A systematic review. *IEEE Reviews in Biomedical Engineering*, 2022.

Tobias Glasmachers, Tom Schaul, Sun Yi, Daan Wierstra, and Jürgen Schmidhuber. Exponential natural evolution strategies. In *Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation*, pp. 393–400, 2010.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Nikolaus Hansen, Dirk V Arnold, and Anne Auger. Evolution strategies. In *Springer Handbook of Computational Intelligence*, pp. 871–898. Springer, 2015.

Md Foysal Haque, Hye-Youn Lim, and Dae-Seong Kang. Object detection based on vgg with resnet network. In *2019 International Conference on Electronics, Information, and Communication (ICEIC)*, pp. 1–3. IEEE, 2019.

Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

Nathan Inkawhich, Wei Wen, Hai Helen Li, and Yiran Chen. Feature space perturbations yield more transferable adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7066–7074, 2019.

Ching-Yu Kao, Junhao Chen, Karla Markert, and Konstantin Böttinger. Rectifying adversarial inputs using xai techniques. In *2022 30th European Signal Processing Conference (EUSIPCO)*, pp. 573–577. IEEE, 2022.

Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, and Orion Reblitz-Richardson. Captum: A unified and generic model interpretability library for pytorch, 2020.

Aditya Kuppa and Nhien-An Le-Khac. Black box attacks on explainable artificial intelligence (xai) methods in cyber security. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE, 2020.

Alexey Kurakin, Ian J Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *Artificial Intelligence Safety and Security*, pp. 99–112. Chapman and Hall/CRC, 2018.

Raz Lapid, Zvika Haramaty, and Moshe Sipper. An evolutionary, gradient-free, query-efficient, black-box algorithm for generating adversarial instances in deep convolutional neural networks. *Algorithms*, 15(11), 2022.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Nicholas Metropolis and Stanislaw Ulam. The Monte Carlo method. *Journal of the American Statistical Association*, 44(247):335–341, 1949.

Grégoire Montavon, Alexander Binder, Sebastian Lapuschkin, Wojciech Samek, and Klaus-Robert Müller. Layer-wise relevance propagation: an overview. *Explainable AI: interpreting, explaining and visualizing deep learning*, pp. 193–209, 2019.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2574–2582, 2016.

Chengcheng Ning, Huajun Zhou, Yan Song, and Jinhui Tang. Inception single shot multibox detector for object detection. In *2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pp. 549–554. IEEE, 2017.

Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *arXiv preprint arXiv:1605.07277*, 2016a.

Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*, pp. 372–387. IEEE, 2016b.

Seyedeh Neelufar Payrovnaziri, Zhaoyi Chen, Pablo Rengifo-Moreno, Tim Miller, Jiang Bian, Jonathan H Chen, Xiuwen Liu, and Zhe He. Explainable artificial intelligence models using real-world electronic health record data: a systematic scoping review. *Journal of the American Medical Informatics Association*, 27(7):1173–1185, 2020.

Tim Salimans, Jonathan Ho, Xi Chen, Szymon Sidor, and Ilya Sutskever. Evolution strategies as a scalable alternative to reinforcement learning. *arXiv preprint arXiv:1703.03864*, 2017.

Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713*, 2016.

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *International Conference on Machine Learning*, pp. 3145–3153. PMLR, 2017.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–9, 2015.

Erico Tjoa and Cuntai Guan. A survey on explainable artificial intelligence (xai): Toward medical xai. *IEEE Transactions on Neural Networks and Learning Systems*, 32(11):4793–4813, 2020.

Florian Tramèr, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*, 2017.

Michael Van Lent, William Fisher, and Michael Mancuso. An explainable artificial intelligence system for small-unit tactical behavior. In *Proceedings of the National Conference on Artificial Intelligence*, pp. 900–907. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2004.

Isunuri B Venkateswarlu, Jagadeesh Kakarla, and Shree Prakash. Face mask detection using mobilenet and global pooling block. In *2020 IEEE 4th Conference on Information & Communication Technology (CICT)*, pp. 1–5. IEEE, 2020.

Savita Walia, Krishan Kumar, Saurabh Agarwal, and Hyunsung Kim. Using xai for deep learning-based image manipulation detection with shapley additive explanation. *Symmetry*, 14(8):1611, 2022.

Dan Wang, Jiayu Lin, and Yuan-Gen Wang. Query-efficient adversarial attack based on latin hypercube sampling. *arXiv preprint arXiv:2207.02391*, 2022.

Xiaosen Wang, Xuanran He, Jingdong Wang, and Kun He. Admix: Enhancing the transferability of adversarial attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16158–16167, 2021.

Daan Wierstra, Tom Schaul, Tobias Glasmachers, Yi Sun, Jan Peters, and Jürgen Schmidhuber. Natural evolution strategies. *Journal of Machine Learning Research*, 15(1):949–980, 2014.

Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L Yuille. Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2730–2739, 2019.

Kaidi Xu, Sijia Liu, Pu Zhao, Pin-Yu Chen, Huan Zhang, Quanfu Fan, Deniz Erdogmus, Yanzhi Wang, and Xue Lin. Structured adversarial attack: Towards general implementation and better interpretability. *arXiv preprint arXiv:1808.01664*, 2018.

Ayesha Younis, Li Shixin, Shelembi Jn, and Zhang Hai. Real-time object detection using pre-trained deep learning models MobileNet-SSD. In *Proceedings of 2020 the 6th International Conference on Computing and Data Engineering*, pp. 44–48, 2020.

Xinyang Zhang, Ningfei Wang, Hua Shen, Shouling Ji, Xiapu Luo, and Ting Wang. Interpretable deep learning under fire. In *29th {USENIX} Security Symposium ({USENIX} Security 20)*, 2020.

Appendix

The following figures provide our full qualitative results. Hyperparameters: n_pop – population size of evolutionary algorithm; lr – learning rate of gradient approximation step for updating the attack; LS – use of Latin sampling or regular sampling.

- Figure 8: Similarity in terms of MSE between the target explanation map, $g(x_{target})$, and the best final adversarial explanation map, $g(x_{adv})$, for 8 different hyperparameter configurations (dataset: ImageNet, model: VGG16).
- Figure 9: Similarity in terms of MSE between the input image, x, and the best final adversarial, x_{adv}, for 8 different hyperparameter configurations (dataset: ImageNet, model: VGG16).
- Figure 10: MSE loss value as function of evolutionary generation for 8 different hyperparameter configurations (dataset: ImageNet, model: VGG16).
- Figure 11: Similarity in terms of MSE between the target explanation map, $g(x_{target})$, and the best final adversarial explanation map, $g(x_{adv})$, for 8 different hyperparameter configurations (dataset: ImageNet, model: Inception).
- Figure 12: MSE loss values for the input image versus the chosen adversarial image for 8 different hyperparameter configurations (dataset: ImageNet, model: Inception).
- Figure 13: MSE loss value as function of evolutionary generation for 8 different hyperparameter configurations (dataset: ImageNet, model: Inception).
- Figure 14: Similarity in terms of MSE between the target explanation map, $g(x_{target})$, and the best final adversarial explanation map, $g(x_{adv})$, for 8 different hyperparameter configurations (dataset: CIFAR100, model: VGG16).
- Figure 15: MSE loss value for input image versus chosen adversarial image for 8 different hyperparameter configurations (dataset: CIFAR100, model: VGG16).
- Figure 16: MSE loss value as function of evolutionary generation for 8 different hyperparameter configurations (dataset: CIFAR100, model: VGG16).
- Figure 17: Similarity in terms of MSE between the target explanation map, $g(x_{target})$, and the best final adversarial explanation map, $g(x_{adv})$, for 8 different hyperparameter configurations (dataset: CIFAR100, model: MobileNet).
- Figure 18: MSE loss value for input image versus chosen adversarial image for 8 different hyperparameter configurations (dataset: CIFAR100, model: MobileNet).
- Figure 19: MSE loss value as function of evolutionary generation for 8 different hyperparameter configurations (dataset: CIFAR100, model: MobileNet).

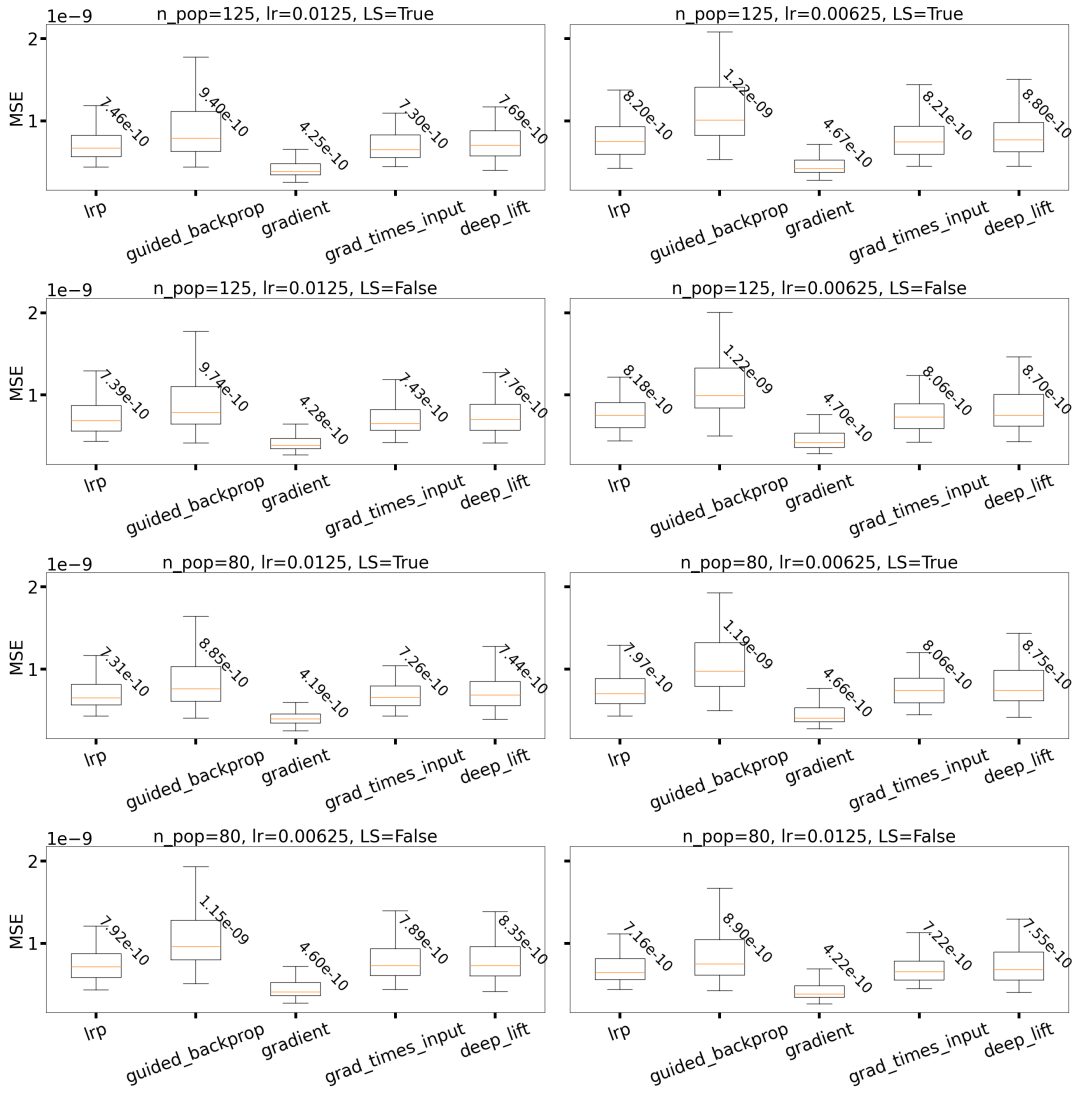

Figure 8: Similarity in terms of MSE between target explanation map, $g(x_{target})$, and best final adversarial explanation map, $g(x_{adv})$, for 8 different hyperparameter configurations. Dataset: ImageNet. Model: VGG16. The Gradient XAI method is the most susceptible to attacks while Guided backpropogation is the hardest to attack; Deep Lift, LRP, and Gradient x Input are similar.

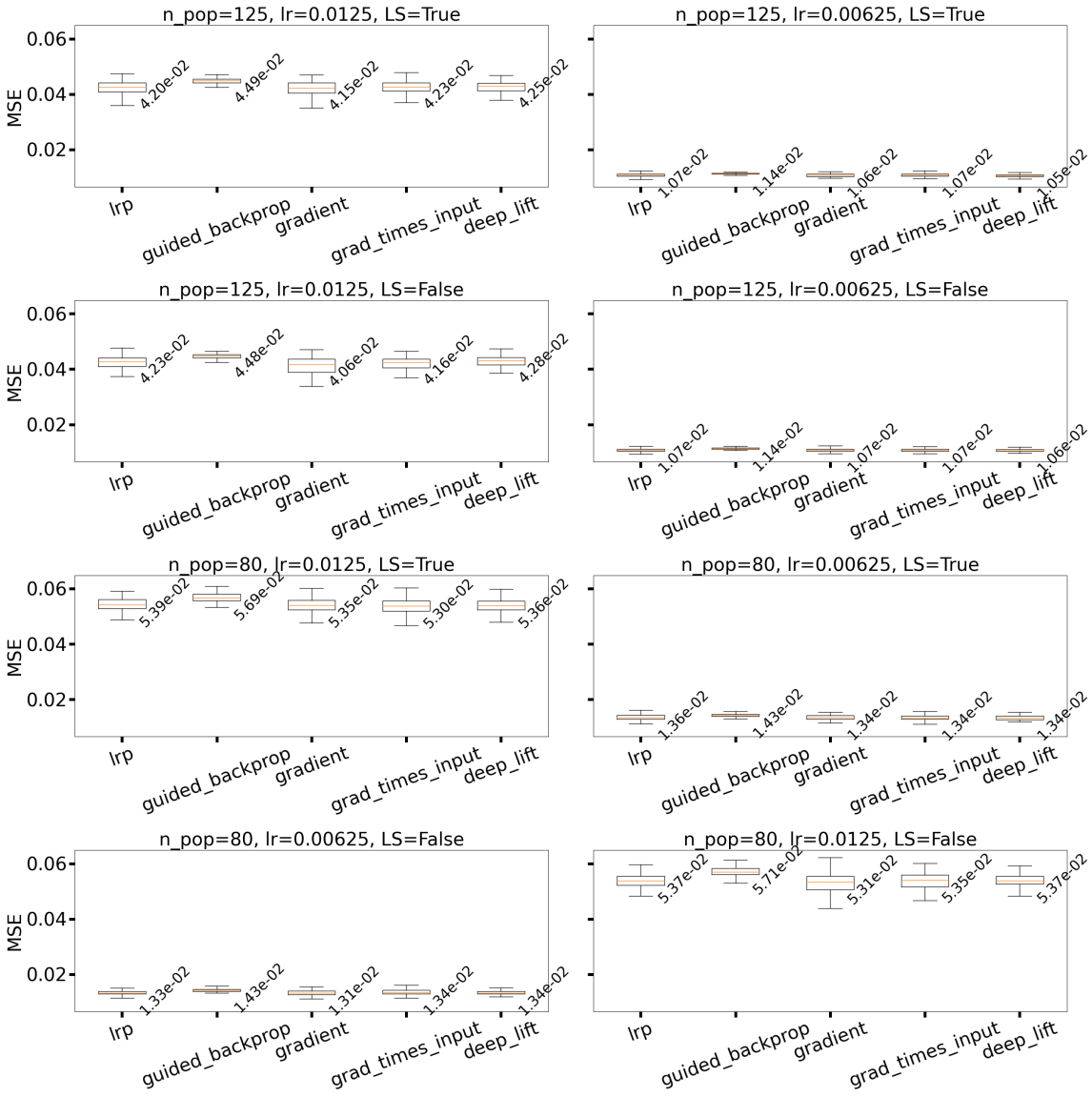

Figure 9: Similarity in terms of MSE between input image, x, and best final adversarial, x_{adv}, for 8 different hyperparameter configurations. Dataset: ImageNet. Model: VGG16. A higher learning rate and a smaller population size (i.e., more gradient steps) contribute to the perturbation of the image. The sampling method has no effect.

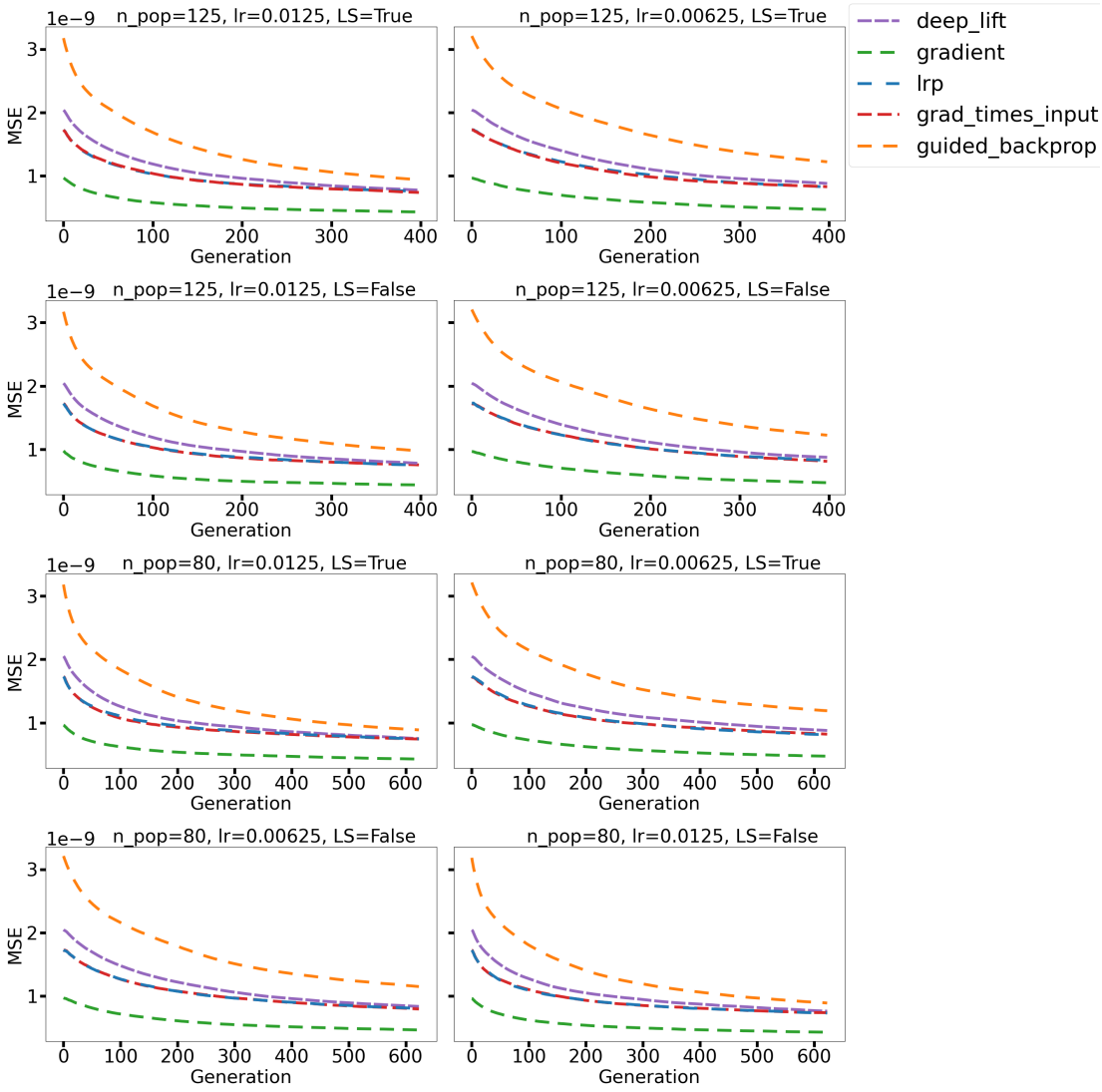

Figure 10: MSE loss value as function of evolutionary generation for 8 different hyperparameter configurations. Dataset: ImageNet. Model: VGG16.

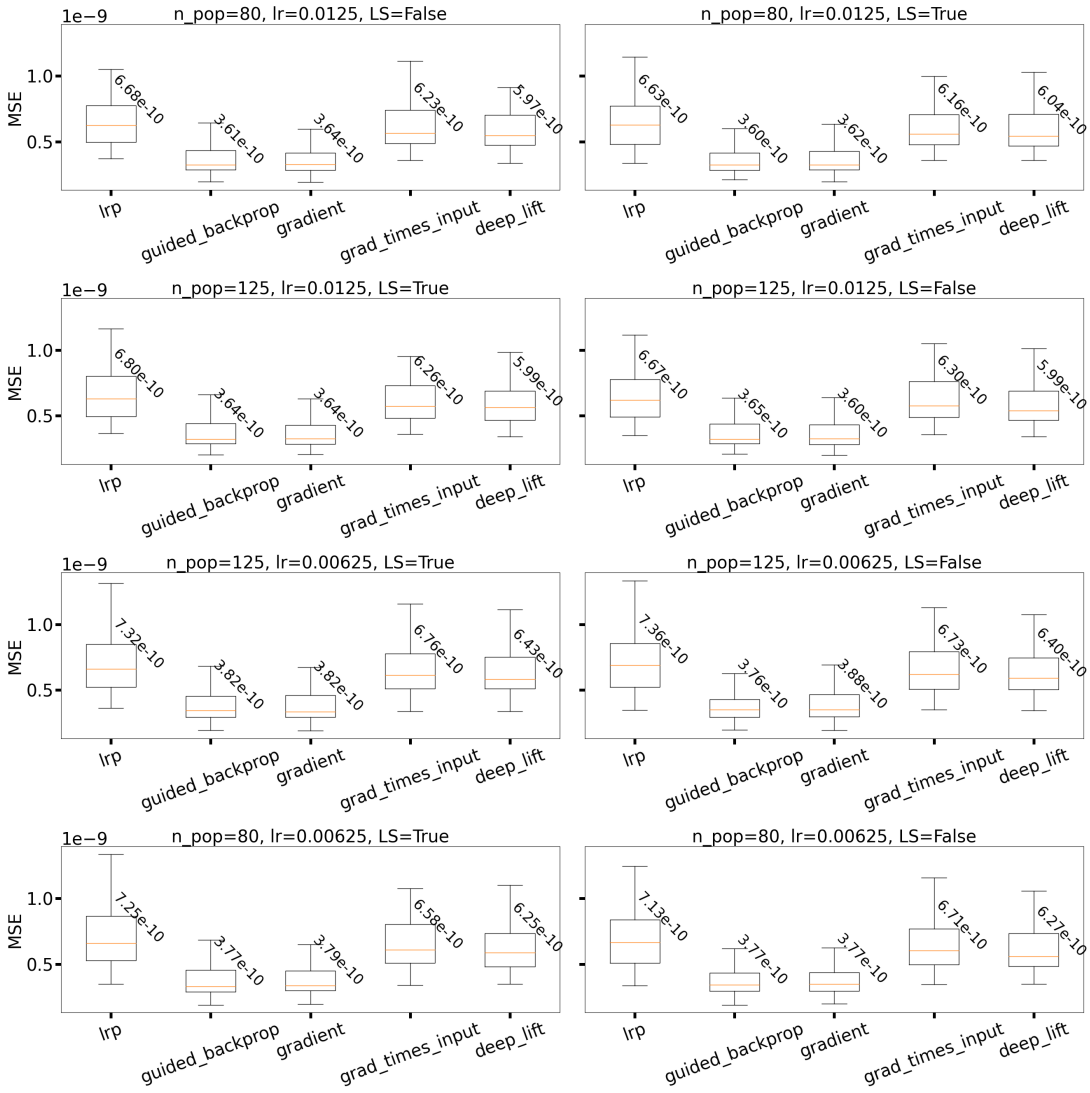

Figure 11: Similarity in terms of MSE between target explanation map, $g(x_{target})$, and best final adversarial explanation map, $g(x_{adv})$, for 8 different hyperparameter configurations. Dataset: ImageNet. Model: Inception. The Gradient XAI and Guided backpropogation methods are the most susceptible to attacks while Deep Lift, LRP, and Gradient x Input are less susceptible.

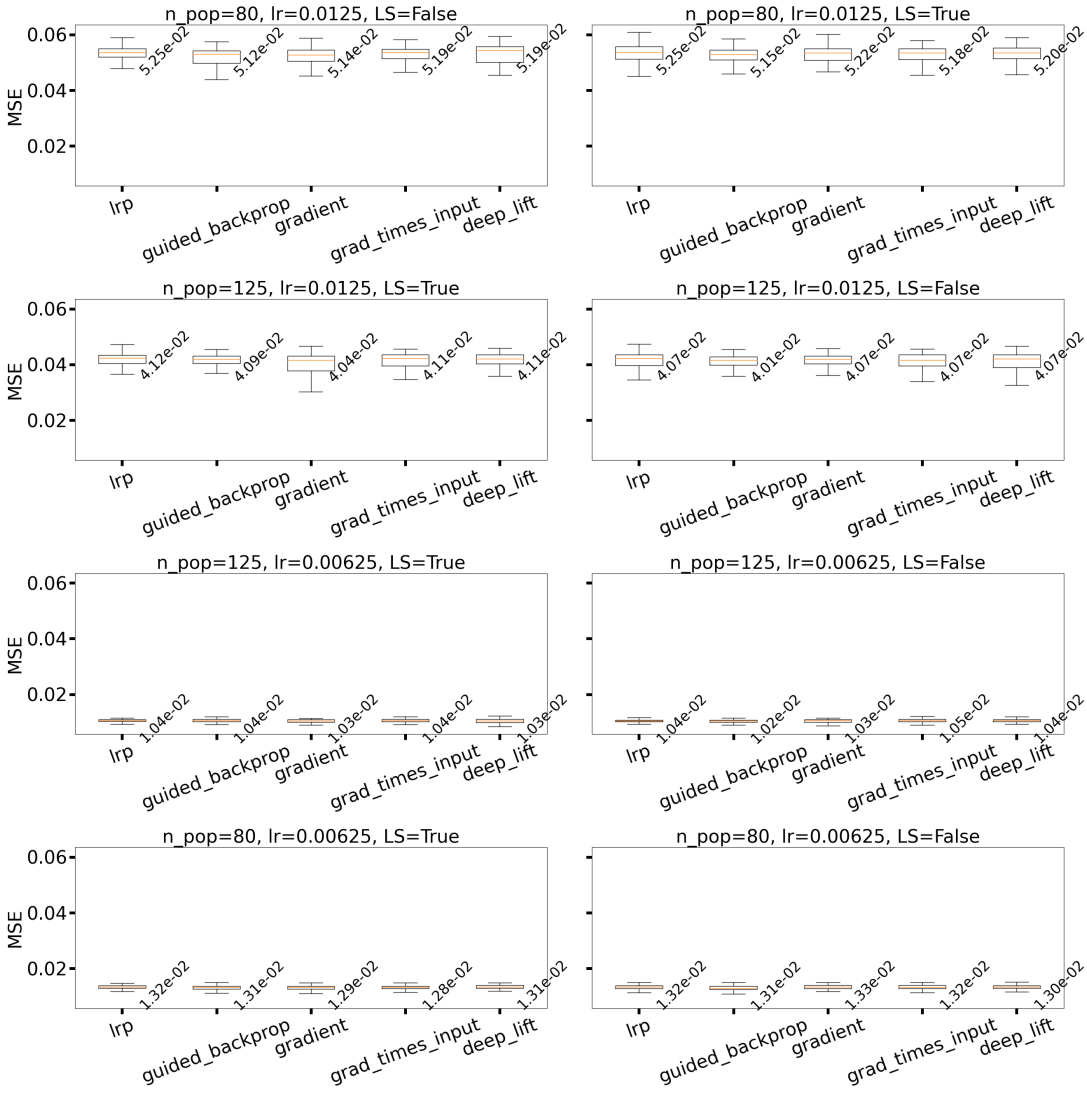

Figure 12: MSE loss value for input image versus chosen adversarial image for 8 different hyperparameter configurations. Dataset: ImageNet. Model: Inception. A higher learning rate and a smaller population size (i.e., more gradient steps) contribute to the perturbation of the image. The sampling method has no effect.

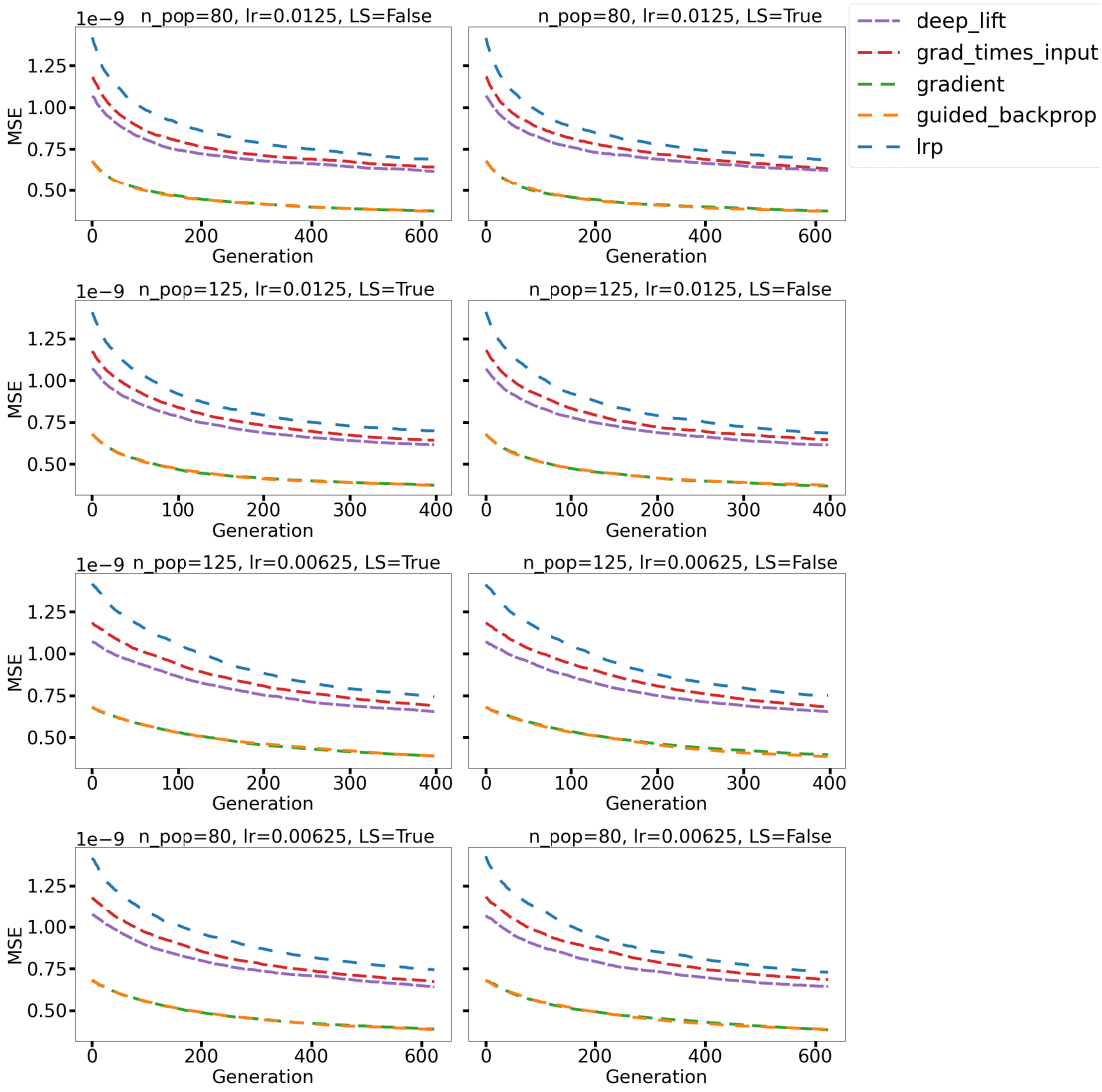

Figure 13: MSE loss value as function of evolutionary generation for 8 different hyperparameter configurations. Dataset: ImageNet. Model: Inception.

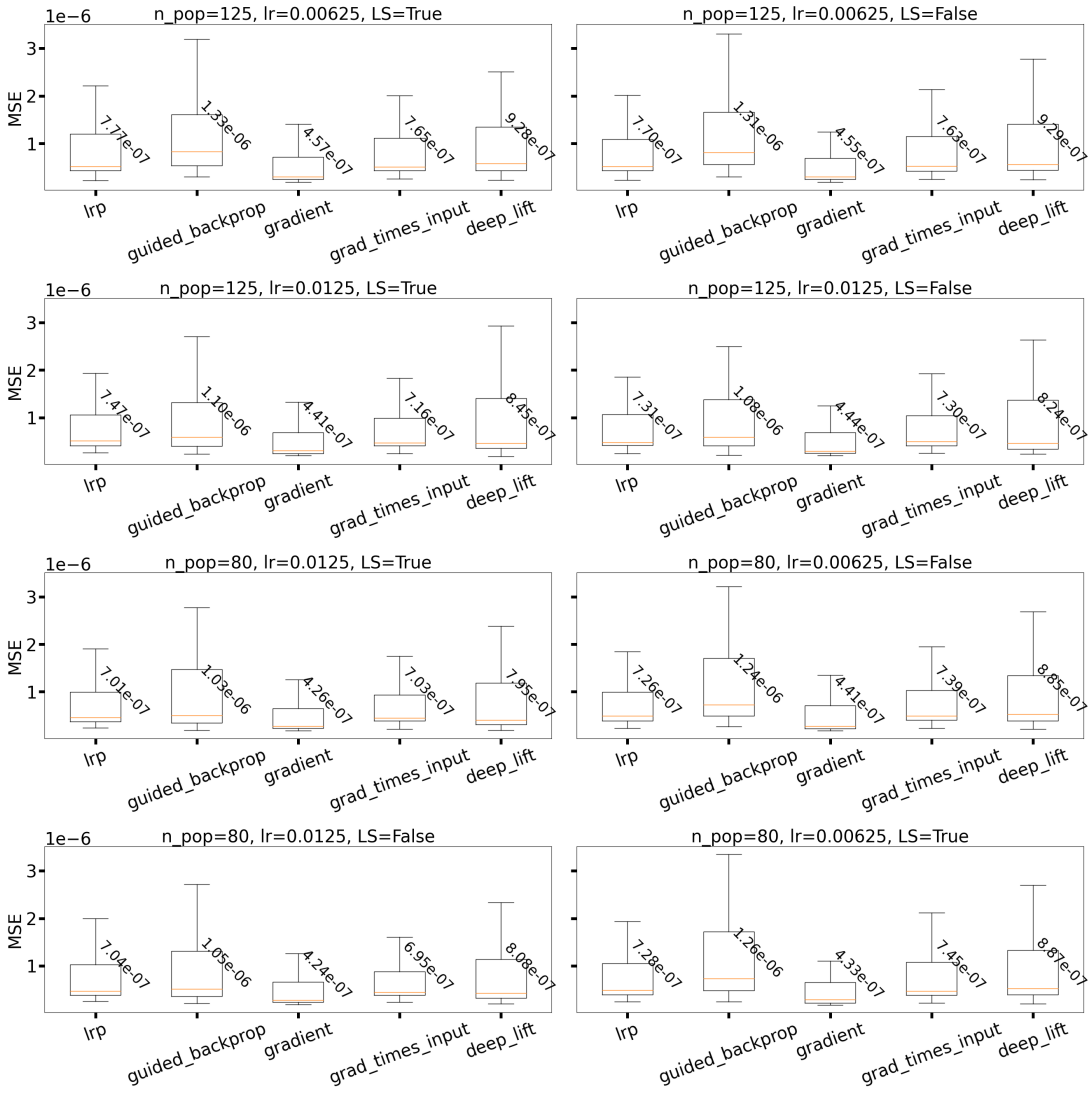

Figure 14: Similarity in terms of MSE between target explanation map, $g(x_{target})$, and best final adversarial explanation map, $g(x_{adv})$, for 8 different hyperparameter configurations. Dataset: CIFAR100. Model: VGG16. The Gradient XAI method is the most susceptible to attacks while Guided backpropogation and Deep Lift are the hardest to attack; LRP and Gradient x Input are similar.

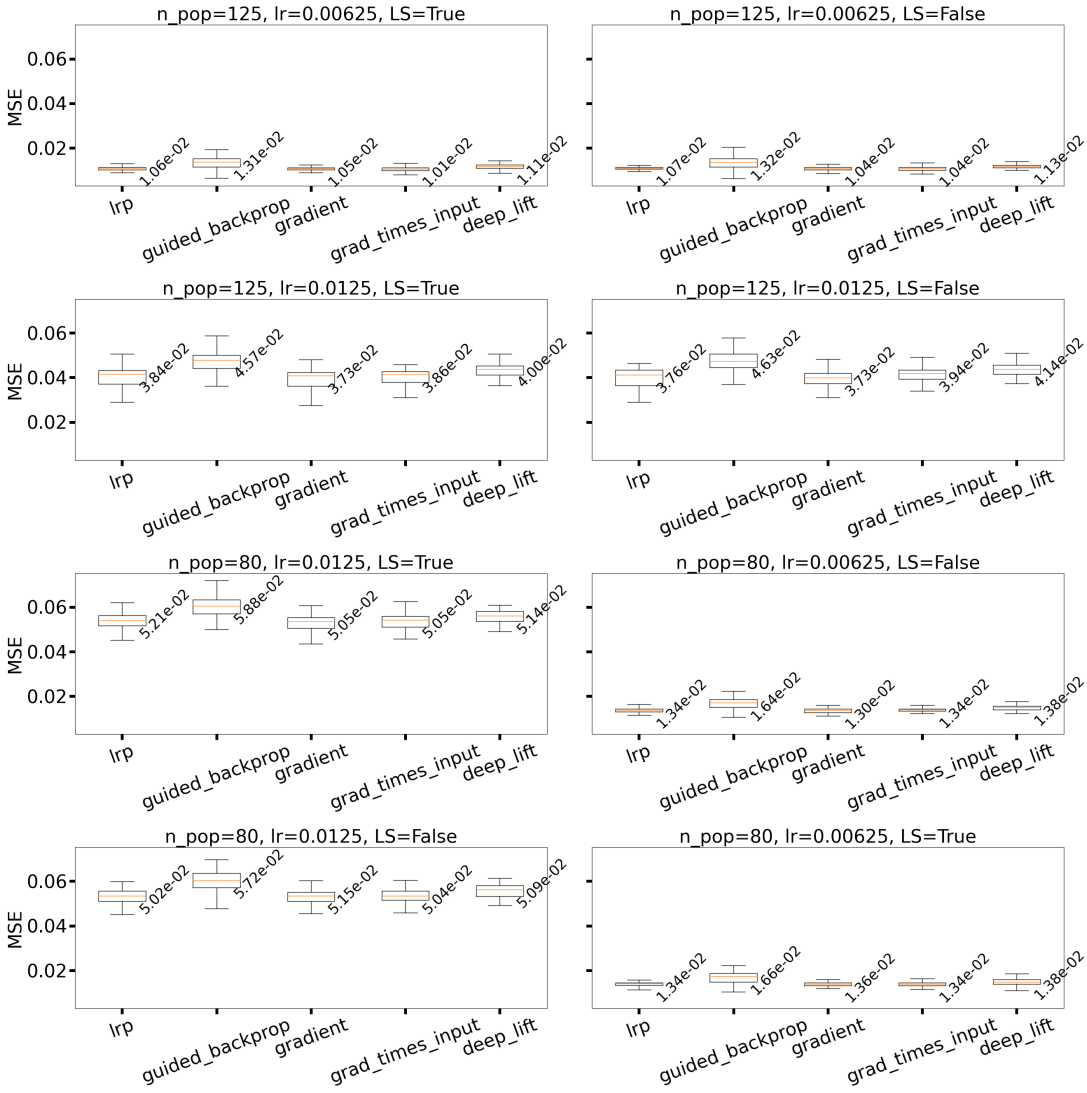

Figure 15: MSE loss value for input image versus chosen adversarial image for 8 different hyperparameter configurations. Dataset: CIFAR100. Model: VGG16. A higher learning rate and a smaller population size (i.e., more gradient steps) contribute to the perturbation of the image. The sampling method has no effect.

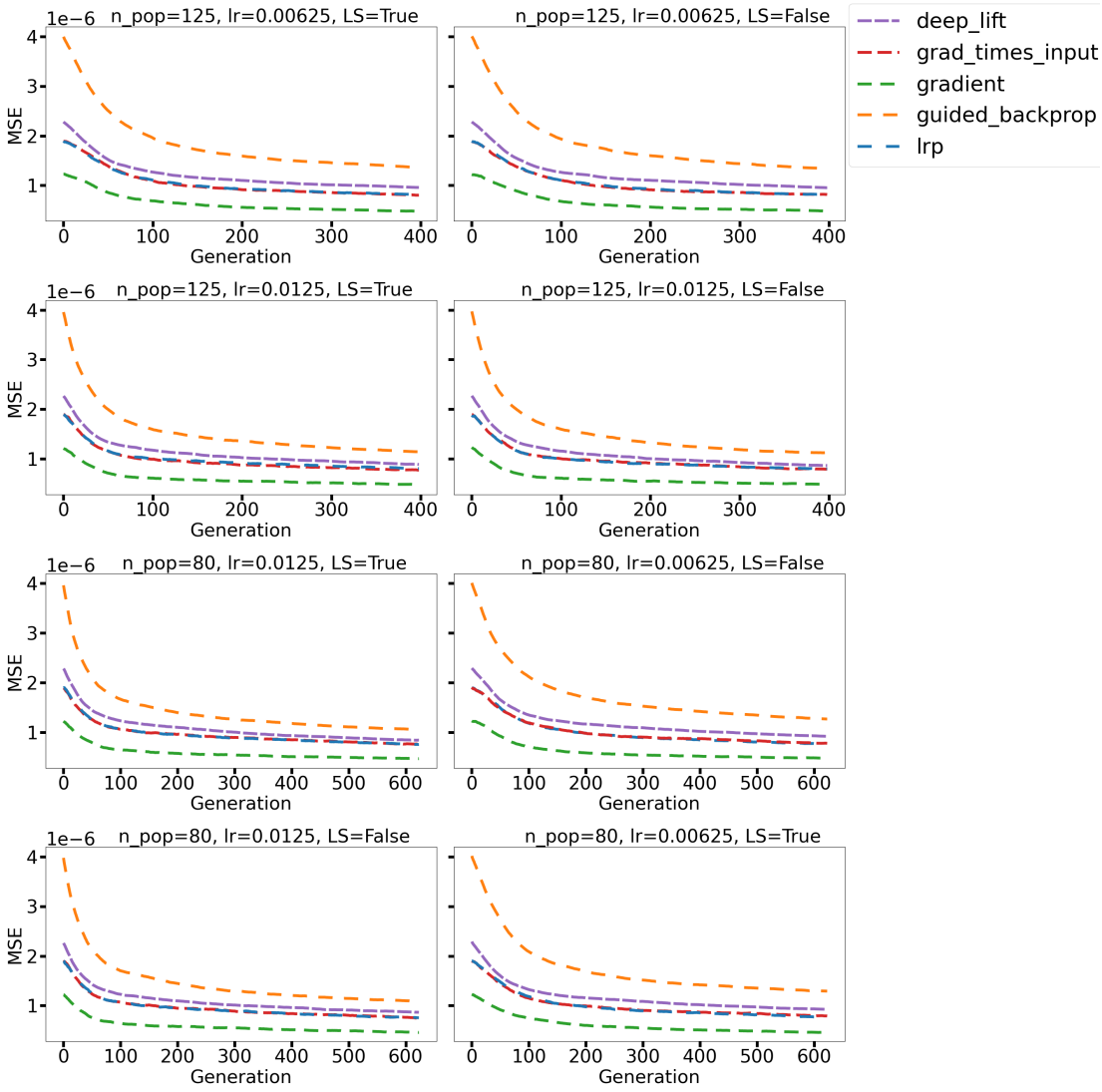

Figure 16: MSE loss value as function of evolutionary generation for 8 different hyperparameter configurations. Dataset: CIFAR100. Model: VGG16.

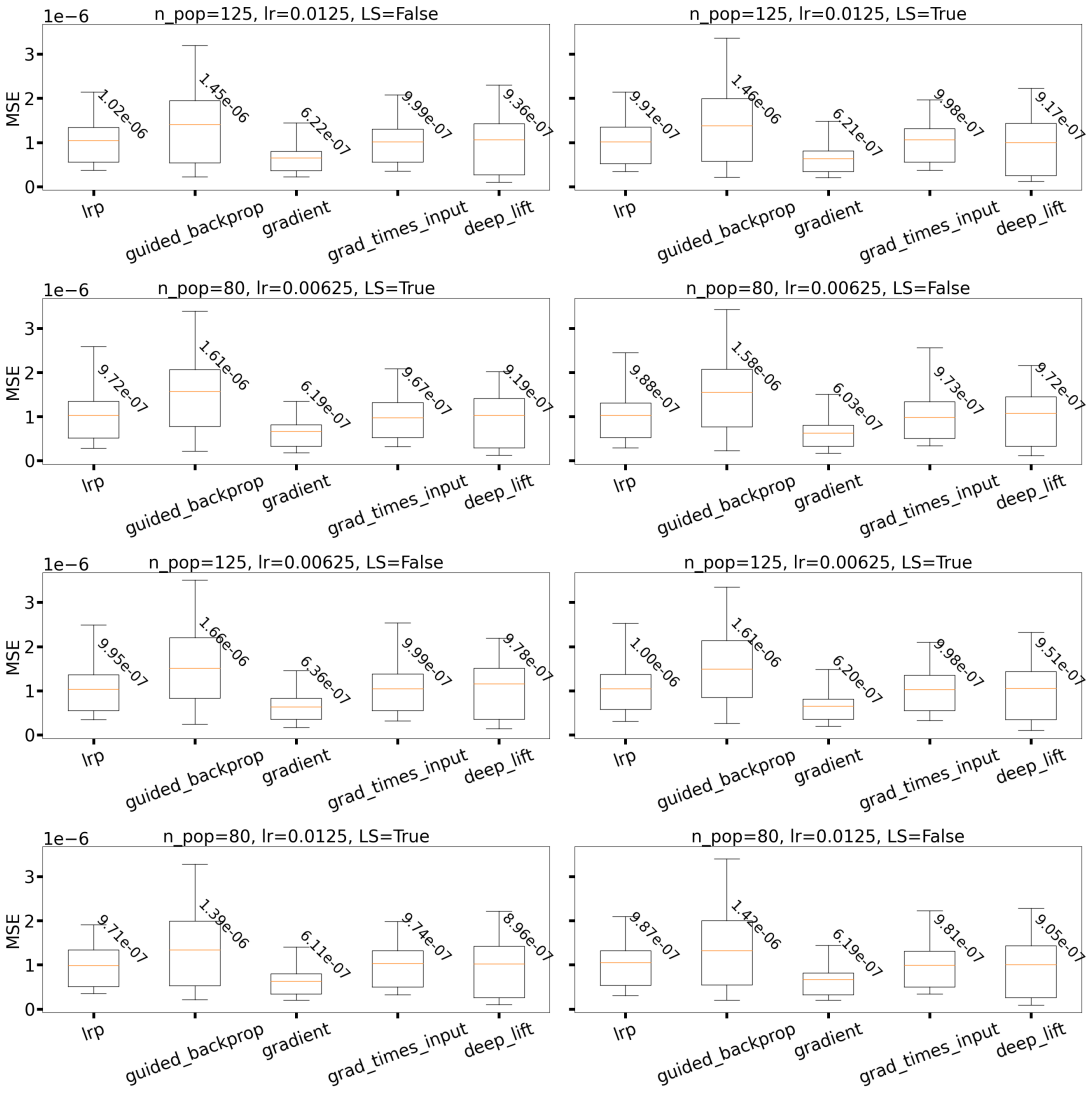

Figure 17: Similarity in terms of MSE between target explanation map, $g(x_{target})$, and best final adversarial explanation map, $g(x_{adv})$, for 8 different hyperparameter configurations. Dataset: CIFAR100. Model: MobileNet. The Gradient XAI method is the most susceptible to attacks while Guided backpropogation is the hardest to attack; Deep Lift, LRP, and Gradient x Input are similar.

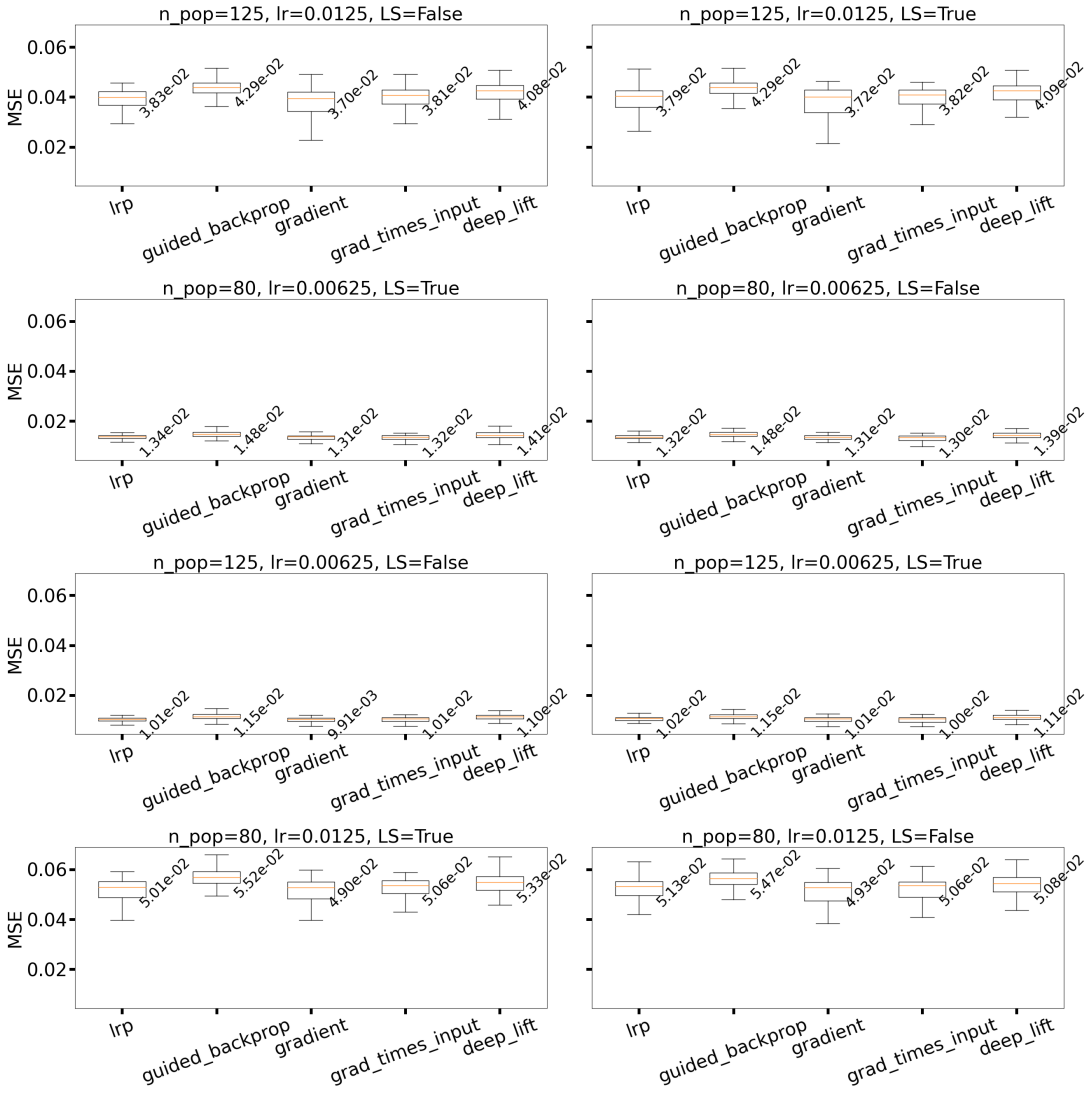

Figure 18: MSE loss value for input image versus chosen adversarial image for 8 different hyperparameter configurations. Dataset: CIFAR100. Model: MobileNet. A higher learning rate and a smaller population size (i.e., more gradient steps) contribute to the perturbation of the image. The sampling method has no effect.

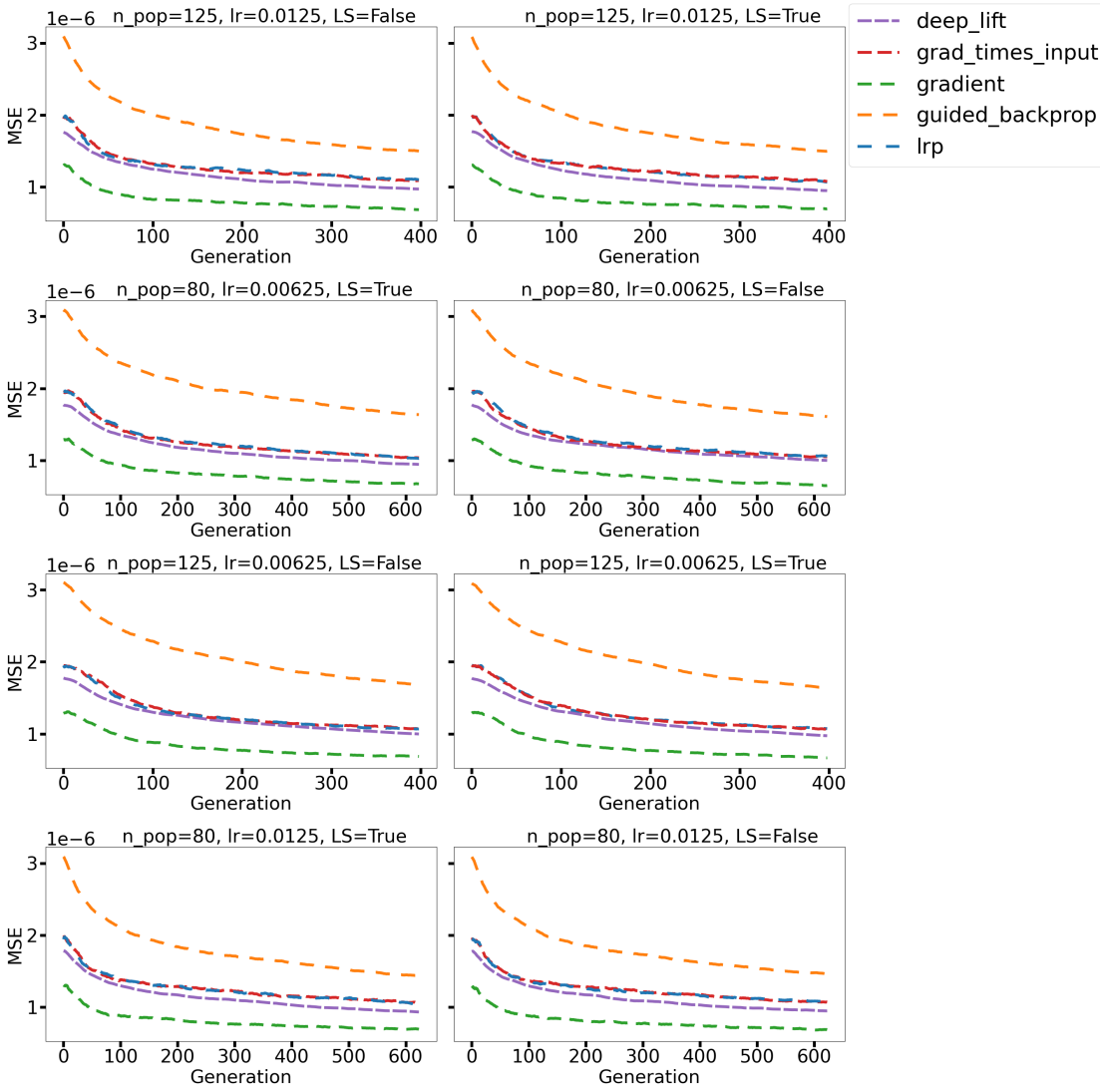

Figure 19: MSE loss value as function of evolutionary generation for 8 different hyperparameter configurations. Dataset: CIFAR100. Model: MobileNet.