Practical Black-Box Attacks on Deep Neural Networks Using Efficient Query Mechanisms

Arjun Nitin Bhagoji[1](✉), Warren He[2], Bo Li[3], and Dawn Song[2]

[1] Princeton University, Princeton, USA
abhagoji@princeton.edu
[2] University of California, Berkeley, Berkeley, USA
[3] University of Illinois at Urbana-Champaign, Champaign, USA

Abstract. Existing black-box attacks on deep neural networks (DNNs) have largely focused on transferability, where an adversarial instance generated for a locally trained model can "transfer" to attack other learning models. In this paper, we propose novel Gradient Estimation black-box attacks for adversaries with query access to the target model's class probabilities, which do not rely on transferability. We also propose strategies to decouple the number of queries required to generate each adversarial sample from the dimensionality of the input. An iterative variant of our attack achieves close to 100% attack success rates for both targeted and untargeted attacks on DNNs. We carry out a thorough comparative evaluation of black-box attacks and show that Gradient Estimation attacks achieve attack success rates similar to state-of-the-art white-box attacks on the MNIST and CIFAR-10 datasets. We also apply the Gradient Estimation attacks successfully against real-world classifiers hosted by Clarifai. Further, we evaluate black-box attacks against state-of-the-art defenses based on adversarial training and show that the Gradient Estimation attacks are very effective even against these defenses.

Keywords: Deep neural networks · Image classification
Adversarial examples · Black-box attacks

1 Introduction

The ubiquity of machine learning provides adversaries with both opportunities and incentives to develop strategic approaches to fool learning systems and achieve their malicious goals. Many attack strategies devised so far to generate adversarial examples that cause learning systems to drastically change their

A. N. Bhagoji—Work done while at University of California, Berkeley.

Electronic supplementary material The online version of this chapter (https://doi.org/10.1007/978-3-030-01258-8_10) contains supplementary material, which is available to authorized users.

V. Ferrari et al. (Eds.): ECCV 2018, LNCS 11216, pp. 158–174, 2018.
https://doi.org/10.1007/978-3-030-01258-8_10

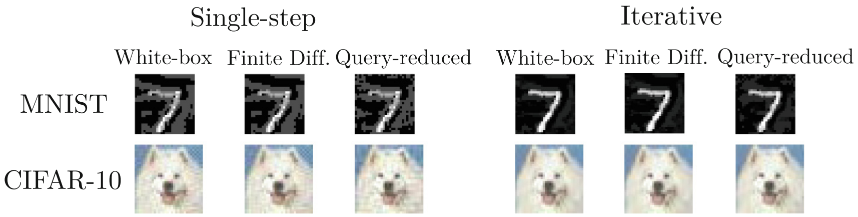


Fig. 1. Targeted adversarial examples for MNIST and CIFAR-10. The ‘7’ from MNIST is classified as a ‘3’ while the *dog* from CIFAR-10 is classified as a *bird* by all attacks. ‘Finite Diff.’ and ‘Query-reduced’ refer to Gradient Estimation attacks with and without query reduction respectively. Perturbations generated using Single-step attacks are far smaller than those for Iterative attacks.

predictions with perturbations imperceptible to humans have been in the white-box setting, where adversaries are assumed to have access to the target model [3,8,20,31]. However, in many realistic settings, adversaries may only have black-box access to the model; i.e., they have no knowledge of the details of the learning system, such as its parameters, but may have query access to the model's predictions on input samples, including class probabilities. This is the case in a number of popular commercial AI offerings from IBM [33], Google [9] and Clarifai [5]. With access to model predictions, the loss of the target model for a given input can be found, but without access to the entire model, the gradients required to carry out white-box attacks cannot be accessed.

Most existing black-box attacks on Deep Neural Networks (DNNs) have focused on *transferability* based attacks [19,24,25], where adversarial examples crafted for a local surrogate model (trained on a representative dataset) can be used to attack the target model to which the adversary has no direct access. In this paper, we design powerful new black-box attacks using *limited query access to target models* which achieve attack success rates and distortion levels close to that of white-box attacks[1]. These attacks do not need access to a representative dataset or the training of a local model. Our contributions are as follows:

New Black-Box Attacks. We propose novel *Gradient Estimation* attacks on DNNs, where the adversary is only assumed to have query access to the target model. In these attacks, the adversary adds perturbations proportional to the *estimated gradient*, instead of the true gradient as in white-box attacks [8,16]. Our attacks achieve close to 100% attack success in both the targeted and untargeted attack settings, matching white-box success on state-of-the-art models on the MNIST [17] and CIFAR-10 [15] datasets. We also experimented with Simultaneous Perturbation Stochastic Approximation (SPSA) [29] and Particle Swarm Optimization (PSO) [14] as alternative methods to carry out query-based black-box attacks but found Gradient Estimation to work the best.

[1] The code to reproduce our results is at https://github.com/sunblaze-ucb/blackbox-attacks.

Query-Reduction Strategies. Since the direct Gradient Estimation attack requires a number of queries on the order of the dimension of the input (784 for MNIST and 3072 for CIFAR-10), we explore strategies for reducing the number of queries to the target model. We propose two strategies: *random feature grouping* and *principal component analysis (PCA) based query reduction*. The use of these is supported by the notion of directional derivatives for differentiable functions. We find that attack success rates close to 90% for untargeted Single-step attacks and 100% for Iterative attacks in both targeted and untargeted cases are achievable with drastic query reduction to just 200 to 800 queries per sample for Single-step attacks and around 8,000 queries for Iterative attacks. Figure 1 displays some successful targeted adversarial examples generated using our attacks.

Attacking Real-World Systems and State-of-the-Art Defenses. To demonstrate the effectiveness of our Gradient Estimation attacks in the real world, we also carry out a *practical black-box attack* (Fig. 3) using these methods against the Not Safe For Work (NSFW) classification and Content Moderation models developed by Clarifai [5], which we choose due to their socially relevant application. These models have begun to be deployed for real-world moderation [18], which makes such black-box attacks especially pernicious. The Gradient Estimation attack achieves a 95.2% attack success rate on the set of images we chose with around 200 queries per image, taking roughly a minute per image. These black-box attacks help us understand the extent of the threat posed to deployed systems by query-based attacks as the attack was carried out with *no knowledge of the training set*.

In addition, we also evaluate the effectiveness of these attacks on *DNNs made more robust using adversarial training* [8,31] and its variants ensemble [32] and iterative adversarial training [21]. We find that although standard and ensemble adversarial training confer some robustness against Single-step attacks, they are vulnerable to Iterative Gradient Estimation attacks, with attack success rates in excess of 70%.

Comparative Evaluation of Black-Box Attacks. We carry out a thorough empirical comparison of black-box attacks on both the MNIST and CIFAR-10 datasets. We show that our Gradient Estimation attacks outperform the other query-based black-box attacks we tested in terms of attack success rate. In the supplementary material, we also show that black-box attacks requiring zero queries to the learning model, including the addition of perturbations that are either random or proportional to the difference of means of the original and targeted classes, as well as transferability based attacks do not perform as well as query-based attacks.

1.1 Related Work

Existing black-box attacks are mostly based on transferability [24,25,31], where an adversarial example generated for a local model is used to attack a target model. Query-based attacks were first proposed for convex-inducing two-class classifiers by Nelson et al. [23]. Xu et al. [35] use genetic algorithms to craft adversarial examples for malware data, while Dang et al. [6] apply hill climbing algorithms. These methods are prohibitively expensive for non-categorical and high-dimensional data such as images. We now discuss attacks that carry out direct query-based black-box attacks on DNNs. Narodytska & Kasiviswanathan [22] propose a greedy local search for high-impact pixels in input saliency maps to generate adversarial examples. Their method uses 500 queries per iteration and runs the greedy local search for around 150 iterations for each image, resulting in a total of 75,000 queries per image, which is much higher than any of our attacks. Our methods achieve higher targeted and untargeted attack success rates on both MNIST and CIFAR-10 compared to their method. In independent work, Chen et al. [4] propose a black-box attack method named ZOO, which also uses the method of finite differences to estimate the derivative of a function. However, while we propose attacks that compute an adversarial perturbation, approximating FGSM and iterative FGS; ZOO approximates the Adam optimizer, while performing coordinate descent on a logit based loss [3]. While they achieve similar attack success rates and distortion levels, they use around 1.5×10^6 and 5.1×10^5 queries per image for MNIST and CIFAR-10 respectively, which is 192× and 67× greater than our Gradient Estimation attacks with query reduction. This leads to the runtime of their attack being up to 160× as long as ours. Neither of these works demonstrates the effectiveness of their attacks on real-world systems or on state-of-the-art defenses. In concurrent work, Ilyas et al. [13] study the use of natural evolution strategies with Gaussian noise to obtain gradient estimates, which is equivalent to the SPSA method. We find that the Gradient Estimation method achieves higher attack success rates at lower distortion levels compared to SPSA. Further, Ilyas et al. do not analyze the effectiveness of their attack on state-of-the-art defenses. Brendel et al. [2] use the target model's output class to modify a starting image which is misclassified, by following the decision boundaries to gradually make it closer to a benign image. Since their attacks use only the output class, they take up to 1.2×10^6 queries to converge to an adversarial example. For similarly sized images, they use 10× more queries for misclassification at the same distortion rate. More detailed comparisons are given in the supplementary material.

2 Query Based Black-Box Attacks: Gradient Estimation

Deployed learning systems often provide feedback for input samples provided by the user. Given query feedback, different adaptive, query-based algorithms can be applied by adversaries to understand the system and iteratively generate effective adversarial examples to attack it. We explored a number of methods using query feedback to carry out black-box attacks including Particle Swarm Optimization [14] and Simultaneous Perturbation Stochastic Approximation [29] (Sect. 2.4) but found these were not as effective as white-box attacks at finding adversarial examples. Given the fact that many white-box attacks for generating adversarial examples are based on gradient information, we tried *directly estimating the gradient to carry out black-box attacks*, and found it to be very effective in a range of conditions. In other words, the adversary can approximate white-box Single-step and Iterative Fast Gradient Sign (FGS) attacks [8,16] using estimates of the losses that are needed to carry out those attacks. We first propose a Gradient Estimation black-box attack based on the method of finite differences [30]. The drawback of a naive implementation of the finite difference method, however, is that it requires $O(d)$ queries per input, where d is the dimension of the input. This leads us to explore methods such as random grouping of features and feature combination using components obtained from Principal Component Analysis (PCA) to reduce the number of queries.

2.1 Notation and Threat Model

A classifier $f(\cdot;\theta) : \mathcal{X} \rightarrow \mathcal{Y}$ is a function mapping from the domain $\mathcal{X}$ to the set of classification outputs $\mathcal{Y}$ (e.g. $\mathcal{Y} = \{0, 1\}$ in the case of binary classification). The number of possible classification outputs is then $|\mathcal{Y}|$. θ is the set of parameters associated with a classifier. $\mathcal{H}$ denotes the constraint set which an adversarial example must lie in. $\ell_f(\mathbf{x}, y)$ is used to represent the loss function for the classifier f with respect to inputs $\mathbf{x} \in \mathcal{X}$ and their true labels $y \in \mathcal{Y}$. The outputs of the penultimate layer of a neural network f, representing the output of the network computed over all preceding layers, are known as the logits. We represent the logits as a vector $\phi^f(\mathbf{x}) \in \mathbb{R}^{|\mathcal{Y}|}$. The final layer of a neural network f used for classification is usually a softmax layer represented as a vector of probabilities $\mathbf{p}^f(\mathbf{x}) = [p_1^f(\mathbf{x}), \dots, p_{|\mathcal{Y}|}^f(\mathbf{x})]$, with $\sum_{i=1}^{|\mathcal{Y}|} p_i^f(\mathbf{x}) = 1$ and $p_i^f(\mathbf{x}) = \frac{e^{\phi_i^f(\mathbf{x})}}{\sum_{j=1}^{|\mathcal{Y}|} e^{\phi_j^f(\mathbf{x})}}$.

Threat Model and Justification. We assume that the adversary can obtain the vector of output probabilities for any input $\mathbf{x}$. The set of queries the adversary can make is then $\mathcal{Q}_f = \{\mathbf{p}^f(\mathbf{x}), \forall \mathbf{x}\}$. For untargeted attacks, the adversary only needs access to the output probabilities for the two most likely classes. A compelling reason for assuming this threat model for the adversary is that many existing cloud-based ML services allow users to query trained models [5,9,33]. The results of these queries are confidence scores which can be used to carry out Gradient Estimation attacks. These trained models are often deployed by the clients of these ML as a service (MLaaS) providers [18]. Thus, an adversary can pose as a user for a MLaaS provider and create adversarial examples using our attack, which can then be used against any client of that provider.

2.2 Gradient Estimation Attacks Using Finite Differences

In this section, we focus on the method of finite differences to carry out Gradient Estimation based attacks. All the analysis is presented for untargeted attacks, but can be easily extended to targeted attacks (see supplementary material). White-box attacks such as the FGS attack use the gradient of an appropriately defined loss to create adversarial examples. If the loss function is $\ell_f(\mathbf{x}, y)$, then a white-box FGS adversarial example will be $\mathbf{x}_{\text{adv}}^{\text{FGS}} = \mathbf{x} + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \ell_f(\mathbf{x}, y))$.

In a black-box setting, however, the adversary does not have access to the gradient of the loss and needs to *estimate* it. One way to do this is the method of Finite Differences [30]. Let the function whose gradient is being estimated using be $g(\mathbf{x})$ where $\mathbf{x} \in \mathbb{R}^d$. The elements of $\mathbf{x}$ are represented as $\mathbf{x}_i$, where $i \in [1, \ldots, d]$. The canonical basis vectors are represented as $\mathbf{e}_i$, where $\mathbf{e}_i$ is 1 only in the i^{th} coordinate and 0 everywhere else. Then, a two-sided estimation of the gradient of g with respect to $\mathbf{x}$ is

$$\text{FD}_{\mathbf{x}}(g(\mathbf{x}), \delta) = \left[\frac{g(\mathbf{x} + \delta \mathbf{e}_1) - g(\mathbf{x} - \delta \mathbf{e}_1)}{2\delta}, \cdots \frac{g(\mathbf{x} + \delta \mathbf{e}_d) - g(\mathbf{x} - \delta \mathbf{e}_d)}{2\delta} \right], \quad (1)$$

where δ is a free parameter that controls the accuracy of the estimation. A one-sided approximation can also be used, but will be less accurate [34]. If the gradient of the function g exists, then $\lim_{\delta \to 0} \text{FD}_{\mathbf{x}}(g(\mathbf{x}), \delta) = \nabla_{\mathbf{x}} g(\mathbf{x})$. The Finite Differences method is useful for a black-box adversary aiming to approximate a gradient based attack, since the gradient can be directly estimated with access to only the function values.

Estimating the Logit Loss. To illustrate how the method of Finite Differences can be used to construct adversarial examples, we focus on a loss function based on logits which was found to work well for white-box attacks by [3]. Attacks using the cross-entropy loss [7] are described in the supplementary material. The logit loss is given by $\ell(\mathbf{x}, y) = \phi(\mathbf{x} + \delta)_y - \max\{\phi(\mathbf{x} + \delta)_i : i \neq y\}$, where y represents the ground truth label for the benign sample $\mathbf{x}$ and $\phi(\cdot)$ are the logits.

An adversary can compute the logit values up to an additive constant by taking the logarithm of the softmax probabilities, which are assumed to be available in this threat model. Since the loss function is equal to the difference of logits, the additive constant is canceled out. Then, the finite differences method can be used to estimate the difference between the logit values for the original class y, and the second most likely class y', i.e., the one given by $y' = \text{argmax}_{i \neq y} \phi(\mathbf{x})_i$. The untargeted adversarial sample generated for this loss in the white-box case is $\mathbf{x}_{\text{adv}} = \mathbf{x} + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}}(\phi(\mathbf{x})_{y'} - \phi(\mathbf{x})_y))$. In the case of a black-box adversary with query access to the softmax probabilities, the adversarial example is

$$\mathbf{x}_{\text{adv}} = \mathbf{x} + \epsilon \cdot \text{sign}(\text{FD}_{\mathbf{x}}(\phi(\mathbf{x})_{y'} - \phi(\mathbf{x})_y, \delta)). \quad (2)$$

This attack is denoted as FD-logit and the corresponding one based on the cross-entropy loss is denoted FD-xent.

Iterative Attacks with Estimated Gradients. The iterative variant of the FGS attack [16] is a powerful attack that often achieves much higher attack success rates in the white-box setting than the simple single-step gradient based attacks. Thus, it stands to reason that a version of the iterative attack with estimated gradients will also perform better than the single-step attacks described until now. An iterative attack with $t + 1$ iterations using the logit loss is:

Algorithm 1. Gradient estimation with query reduction using random features

```
Input: $\mathbf{x}$, $k$, $\delta$, $g(\cdot)$
Output: Estimated gradient $\hat{\nabla}_{\mathbf{x}} g(\mathbf{x})$ of $g(\cdot)$ at $\mathbf{x}$
Initialize empty vector $\hat{\nabla}_{\mathbf{x}} g(\mathbf{x})$ of dimension $d$
for $i \leftarrow 1$ to $\lceil \frac{d}{k} \rceil - 1$ do
    Choose a set of random $k$ indices $S_i$ out of $[1, \ldots, d]/\{\cup_{j=1}^{i-1} S_j\}$
    Initialize $\mathbf{v}$ such that $\mathbf{v}_j = 1$ iff $j \in S_i$
    For all $j \in S_i$, set $\hat{\nabla}_{\mathbf{x}} g(\mathbf{x})_j = \frac{g(\mathbf{x}+\delta\mathbf{v}) - g(\mathbf{x}-\delta\mathbf{v})}{2\delta k}$, which is the two-sided approximation of the directional derivative along $\mathbf{v}$
end for
Initialize $\mathbf{v}$ such that $\mathbf{v}_j = 1$ iff $j \in [1, \ldots, d]/\{\cup_{j=1}^{\lceil \frac{d}{k} \rceil - 1} S_j\}$
For all $j \in [1, \ldots, d]/\{\cup_{j=1}^{\lceil \frac{d}{k} \rceil - 1} S_j\}$, set $\hat{\nabla}_{\mathbf{x}} g(\mathbf{x})_j = \frac{g(\mathbf{x}+\delta\mathbf{v}) - g(\mathbf{x}-\delta\mathbf{v})}{2\delta k}$
```

$$\mathbf{x}_{\text{adv}}^{t+1} = \mathbf{x}_{\text{adv}}^{t} + \alpha \cdot \text{sign}\left(\text{FD}_{\mathbf{x}_{\text{adv}}^{t}}\left(\phi(\mathbf{x}_{\text{adv}}^{t})_{y'} - \phi(\mathbf{x}_{\text{adv}}^{t})_{y}, \delta\right)\right), \tag{3}$$

where α is the step size and $\mathcal{H}$ the constraint set for the adversarial example. This attack is denoted as IFD-logit (IFD-xent with the cross-entropy loss).

2.3 Query Reduction Techniques

A drawback of the Finite Differences technique is that the number of queries needed per adversarial sample is exactly $2d$ for a two-sided approximation which could be too large for high-dimensional inputs. So, we examine two techniques to reduce the number of queries the adversary has to make. Both techniques involve estimating the gradient for groups of features, instead of estimating it using a single feature at a time. The justification for the use of feature grouping comes from the relation between gradients and directional derivatives [12] for differentiable functions. The directional derivative of a function g is defined as $\nabla_{\mathbf{v}} g(\mathbf{x}) = \lim_{h \to 0} \frac{g(\mathbf{x}+h\mathbf{v}) - g(\mathbf{x})}{h}$. It is a generalization of a partial derivative. For differentiable functions, $\nabla_{\mathbf{v}} g(\mathbf{x}) = \nabla_{\mathbf{x}} g(\mathbf{x}) \cdot \mathbf{v}$, which implies that the directional derivative is just the projection of the gradient along the direction $\mathbf{v}$. Thus, estimating the gradient by grouping features is equivalent to estimating an approximation of the gradient constructed by projecting it along appropriately chosen directions. The estimated gradient $\hat{\nabla}_{\mathbf{x}} g(\mathbf{x})$ of any function g can be computed using the techniques below, and then plugged in to Eqs. 2 and 3 instead of the Finite Differences term to generate an adversarial example. Next, we introduce the techniques applied to group the features for estimation.

Query Reduction Based on Random Grouping. The simplest way to group features is to choose, without replacement, a random set of features. The gradient can then be simultaneously estimated for all these features. If the size of the set chosen is k, then the number of queries the adversary has to make is $\lceil \frac{d}{k} \rceil$. When $k = 1$, this reduces to the Finite Differences method from Sect. 2.2. In each iteration of Algorithm 1, there is a set of indices S according to which $\mathbf{v}$

is determined, with $\mathbf{v}_i = 1$ if and only if $i \in S$. Thus, the directional derivative being estimated is $\sum_{i \in S} \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}_i}$, which is an average of partial derivatives.

Query Reduction Using PCA Components. A more principled way to reduce the number of queries the adversary has to make to estimate the gradient is to compute directional derivatives along the principal components as determined by principal component analysis (PCA) [28], which requires the adversary to have access to a set of data which is representative of the training data. If $\mathbf{U}$ is the $d \times d$ matrix whose columns are the principal components $\mathbf{u}_i$, where $i \in [d]$, then the approximation of the gradient in the PCA basis is $(\nabla_{\mathbf{x}} g(\mathbf{x}))^k = \sum_{i=1}^{k} \left(\nabla_{\mathbf{x}} g(\mathbf{x})^{\intercal} \frac{\mathbf{u}_i}{\|\mathbf{u}_i\|} \right) \frac{\mathbf{u}_i}{\|\mathbf{u}_i\|}$, where the term on the left represents an approximation of the true gradient by the sum of its projection along the top k principal components. Since in the black-box setting the true gradient is inaccessible, the weights of the representation in the PCA basis are estimated using directional derivatives along the principal components. The supplementary material contains a detailed description of this method.

Iterative Attacks with Query Reduction. Performing an iterative attack with the gradient estimated using Finite Differences could be expensive for an adversary, needing $2td$ queries to the target model, for t iterations with the two-sided Finite Differences estimation of the gradient. To lower the number of queries needed, the adversary can use either of the query reduction techniques described above to reduce the number of queries to $2tk$ ($k < d$). These attacks using the cross-entropy loss are denoted as IGE-QR (RG-k, logit) for the random grouping technique and IGE-QR (PCA-k, logit) for the PCA-based technique.

2.4 Other Query-Based Black-Box Attacks

Other black-box optimization techniques we considered for generating adversarial examples were Particle Swarm Optimization (PSO) [14],[2] a commonly used evolutionary optimization strategy and SPSA method [29]. PSO is a heuristic gradient-free optimization technique which initiates a number of candidate solutions called ‘particles’ which then move around the search space to find better solutions, previously used to find adversarial examples to fool face recognition systems [27]. SPSA is a special case of natural evolution strategies (NES) [26], where the distribution over the parameters is assumed to be a factored Gaussian. It is similar to the method of Finite Differences, but it estimates the gradient of the loss along a *random direction* $\mathbf{r}$ at each step, instead of along the canonical basis vectors. While each step of SPSA only requires 2 queries to the target model, a large number of steps are nevertheless required to generate adversarial examples. A single step of SPSA does not reliably produce adversarial examples.

[2] Using freely available code from http://pythonhosted.org/pyswarm/.

3 Experimental Results

In this section, we compare various black-box attacks in both targeted and untargeted settings to Gradient Estimation attacks as well as comparing them to white-box attacks. We also describe how we carried out a successful targeted attack on a real-world system, Clarifai, in Sect. 3.5.

3.1 Evaluation Setup

We evaluate our attacks on state-of-the-art neural networks on the MNIST [17] and CIFAR-10 [15] datasets. All models were run on a GPU with a batch size of 100. The details are as follows: (i) *MNIST.* Each pixel of the MNIST image data is scaled to $[0, 1]$. We trained two different CNNs on the MNIST dataset, denoted Model A and Model B [32]. Model A has 2 convolutional layers followed by a fully connected layer while Model B has only 3 convolutional layers. Both models have a test accuracy of 99.2%; (ii) *CIFAR-10.* Each pixel of the CIFAR-10 image data is in $[0, 255]$. We choose two model architectures for this dataset, which are both ResNet variants. Resnet-32 [11] is a 32-layer ResNet achieving 92.4% test accuracy while Resnet-28-10 [36] is a 28-layer ResNet with 10 times width expansion with 94.4% test accuracy. Further architecture details are in the supplementary material.

Attack Parameters. We focus on attacks that use the logit-based loss (logit) as it has better performance but also use the cross-entropy loss (xent) for comparison. In all attacks, the adversary's perturbation is constrained using the L_∞ distance. For the MNIST dataset, we vary the adversary's perturbation budget ϵ from 0 to 0.4, since at a perturbation budget of 0.5, any image can be made solid gray while for the CIFAR-10 dataset, we vary it from 0 to 28. We use the Finite Difference parameter $\delta = 1.0$ for FD-xent and IFD-xent for both datasets, while using $\delta = 0.01$ for FD-logit and IFD-logit. A larger value of δ is needed for xent loss based attacks to work well since the probability values used in the xent loss are not as sensitive to changes as the logit loss. For all Iterative attacks, including white-box attacks, we use $\alpha = 0.01$ and $t = 40$ for MNIST and $\alpha = 1.0$ and $t = 10$ for CIFAR-10. We find these choices work well while maintaining low runtimes for the Gradient Estimation attacks. For the query reduction methods, we use a random group size of 8 for both datasets and the number of principal components to be 100 for MNIST and CIFAR-10. For SPSA, we use around 4000 iterations for both datasets with a step size of 10^{-3} for MNIST and 2×10^{-2} for CIFAR-10. The effect of various hyperparameters on attack success is examined in the supplementary material.

3.2 Metrics

We now define the standard metrics we use to determine attack performance.

Attack Success Rate. The main metric, the attack success rate, is the fraction of samples that meets the adversary's goal: $f(\mathbf{x}_{\mathrm{adv}}) \neq y$ for untargeted attacks and $f(\mathbf{x}_{\mathrm{adv}}) = T$ for targeted attacks with target T [31,32].

Average Distortion. We also evaluate the average distortion for adversarial examples using average L_2 distance between the benign and adversarial ones as in [10]: $\Delta(\mathbf{X}_{\mathrm{adv}}, \mathbf{X}) = \frac{1}{N}\sum_{i=1}^{N} \|(\mathbf{X}_{\mathrm{adv}})_i - (\mathbf{X})_i\|_2$ where N is the number of samples. This metric allows us to compare the average distortion for attacks which achieve similar attack success rates, and therefore infer which one is stealthier.

Number of Queries. Query based black-box attacks make queries to the target model, and this metric may affect the cost of mounting the attack. This is an important consideration when attacking real-world systems which have costs associated with the number of queries made.

For *MNIST*, Single-step attacks are carried out on the *test set* of 10,000 samples, while Iterative attacks are carried out on 1,000 randomly chosen samples from the test set. For *CIFAR-10*, we choose 1,000 random samples from the test set for both Single-step and Iterative attacks. In our evaluation of targeted attacks, we choose target T for each sample uniformly at random from the set of classification outputs, except the true class y of that sample.

Table 1. Targeted black-box attacks: attack success rates. The number in parentheses () for each entry is $\Delta(\mathbf{X}, \mathbf{X}_{\mathrm{adv}})$, the average L_2 distortion over all examples used in the attack. The number in brackets [] beside the Single-step and Iterative descriptors gives the number of queries needed for each type of attack. The per-pixel perturbation limits are $\epsilon = 0.3$ for MNIST (Top) and $\epsilon = 8$ for CIFAR-10 (Bottom).

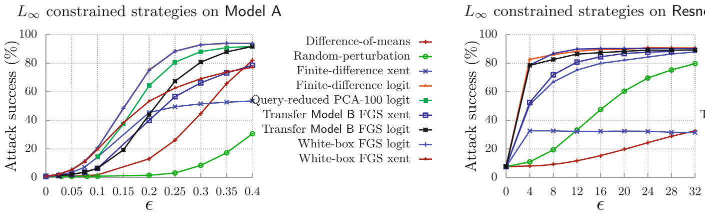
Dataset	White-box		Gradient Estimation, FD (ours)		Gradient Estimation, Query Reduction (ours)			
MNIST	Single-step	Iterative	Single-step [1568]	Iterative [62720]	Single-step [$\sim$ 200]		Iterative [8000]	
Models	FGS (logit)	IFGS (logit)	FD-logit	IFD-logit	PCA-100	RG-8	PCA-100	RG-8
A	30.1 (6.1)	99.6 (2.7)	29.9 (6.1)	**99.7 (2.7)**	23.2 (5.9)	15.9 (5.9)	96.2 (3.3)	73.8 (2.5)
B	29.6 (6.2)	**98.7 (2.4)**	29.3 (6.3)	**98.7 (2.4)**	29.0 (6.3)	17.8 (6.3)	93.9 (2.9)	73.7 (2.6)
CIFAR-10	Single-step	Iterative	Single-step [6144]	Iterative [61440]	Single-step [$\sim$ 800]		Iterative [$\sim$ 8000]	
Models	FGS (logit)	IFGS (logit)	FD-logit	IFD-logit	PCA-400	RG-8	PCA-400	RG-8
Resnet-32	23.5 (436.0)	**100.0 (89.5)**	23.0 (437.0)	**100.0 (89.5)**	21.0 (438.2)	19.0 (438.1)	81.0 (222.8)	97.0 (126.1)
Resnet-28-10	27.6 (436.5)	**100.0 (99.0)**	28.0 (436.1)	**100.0 (98.3)**	23.0 (433.7)	20.0 (433.7)	72.0 (253.1)	94.0 (132.4)

3.3 Effectiveness of Targeted Gradient Estimation Attacks

We find that Targeted Gradient Estimation attacks match white-box attack success, even with query reduction. The Iterative Gradient Estimation attack using Finite Differences and the logit loss (IFD-logit) achieves close to 100% targeted attack success rates on both MNIST and CIFAR-10 models (Table 1). The Single-step attack FD-logit achieves about 20 to 30% attack success rates, matching the performance of Single-step white-box attacks such as FGS-logit. The average distortion for samples generated using gradient estimation methods is similar to that of white-box attacks. Further, the Iterative Gradient Estimation attacks with query reduction achieve high targeted attack success rates as well. For example, using the random grouping method with a group size of 8 (RG-8) for query reduction and using just around 8000 queries per sample, attack success rates of 97% and 94% are achieved for Resnet-32 and Resnet-28-10 respectively.

3.4 Comparing Untargeted Black-Box Attacks

Single-step Gradient Estimation Attacks Match White-Box Attack Success. The Gradient Estimation attack with Finite Differences (FD-logit) is the most successful *untargeted* Single-step black-box attack for MNIST and CIFAR-10 models as can be seen in Fig. 2. We also compare against black-box attacks that make zero queries to the target model; these are the Difference-of-Means, Random Perturbation and Transfer attacks. The Transfer attack is based on the well-known phenomenon of transferability [24,31]. Further details and experimental results for these attacks are in the supplementary material.



(a) Model A (MNIST)

(b) Resnet-32 (CIFAR-10)

Fig. 2. Effectiveness of untargeted Single-step black-box attacks on Model A (MNIST) and Resnet-32 (CIFAR-10). The y-axis for both figures plots the attack success as the perturbation magnitude ϵ is increased. The most successful black-box attack in both cases is the Gradient Estimation attack using Finite Differences with the logit loss (FD-logit), which matches white-box FGS logit-based attack success (WB FGS-logit). The Gradient Estimation attack with query reduction using PCA (GE-QR (PCA-k, logit)) performs well for both datasets.

The FD-logit attack significantly outperforms transferability-based attacks and closely tracks white-box FGS with a logit loss (WB FGS-logit) on MNIST and CIFAR-10. The Gradient Estimation attack with PCA based query reduction (GE-QR (PCA-k, logit)) is also effective, with performance close to that of FD-logit with $k = 100$ for MNIST (Fig. 2a) and $k = 400$ for CIFAR-10 (Fig. 2b). While random grouping is not as effective as the PCA based method for Single-step attacks, we find it is as effective for Iterative attacks.

Iterative Gradient Estimation Attacks Outperform other Query-Based Black-Box Attacks. A comparative evaluation of all the query-based black-box attacks we experimented with for both MNIST and CIFAR-10 datasets is given in Table 2. For adversarial examples generated iteratively, the Iterative Gradient Estimation attack with Finite Differences (IFD-logit) achieves 100% attack success rate on both datasets. White-box Iterative FGS also achieves 100% attack success rates with distortions of 2.1 for MNIST and 66.1 for CIFAR-10. The attack that achieves the best trade-off between speed and attack success is IGE-QR (RG-k, logit), achieving close to 100% success rates on both datasets with just around 8000 queries. We found PSO to be prohibitively slow (with a swarm size of 100) for a large dataset and outperformed even by the Single-step FD-logit attack, in spite of trying a large range of parameters. While the SPSA method is quite effective, it is outperformed by Iterative Gradient Estimation, with and without query reduction in terms of attack success rate for both MNIST and CIFAR-10. Also, IGE-QR (RG-k, logit) achieves a higher attack success rate with lower distortion for MNIST. In practice, we found the convergence of SPSA to be much more sensitive to the choice of both δ (gradient estimation step size) and α (loss minimization step size).

Table 2. Comparison of attack success (AS) and distortion (dist.) for **untargeted query-based black-box attack** methods. All attacks for MNIST use an L_∞ constraint of $\epsilon = 0.3$ while those for CIFAR-10 use $\epsilon = 8$. The logit loss is used for all methods expect PSO, which uses the class probabilities

Attack Type	MNIST (Model A)			CIFAR-10 (Resnet-32)		
Query-based attack	AS (Dist.)	Queries	Avg. Time (s)	AS (Dist.)	Queries	Avg. Time (s)
Finite Diff.	92.9 (6.1)	1568	8.8×10^{-2}	86.0 (410.3)	6144	3.3
Gradient Estimation (RG-8)	61.5 (6.0)	196	1.1×10^{-2}	66.8 (402.7)	768	0.43
Iter. Finite Diff.	**100.0 (2.1)**	62720	3.5	**100.0 (65.7)**	61440	32.1
Iter. Gradient Estimation (RG-8)	98.4 (1.9)	8000	0.43	99.0 (80.5)	7680	4.2
Particle Swarm Optimization	84.1 (5.3)	10000	21.2	89.2 (262.3)	7700	67.3
SPSA	96.7 (3.9)	8000	1.25	88.0 (44.4)	7680	8.7

3.5 Attacks on Clarifai, a Real-World System

Since the only requirement for carrying out the Gradient Estimation based attacks is query-based access to the target model, a number of deployed public systems that provide classification as a service can be used to evaluate our methods. We choose *Clarifai* [5], as it has models trained to classify image datasets for a variety of practical applications, and it provides black-box access to its models and returns confidence scores upon querying. In particular, Clarifai has models used for the detection of Not Safe For Work (NSFW) content, as well as for Content Moderation. These are important applications where the presence of adversarial examples presents a real danger: an attacker, using query access to the model, could generate an adversarial examples which will no longer be classified as inappropriate. For example, an adversary could upload violent images, adversarially modified, such that they are marked incorrectly as ‘safe’ by the Content Moderation model.

We evaluate our attack using the Gradient Estimation method on Clarifai’s NSFW and Content Moderation models. When we query the API with an image, it returns the confidence scores associated with each category (summing to 1). We use the *random grouping* query reduction technique and take the logarithm of the confidence scores in order to use the *logit loss*. This method achieves 95.2% attack success rate against the NSFW model on our sample set of 21 images. An example of an attack against the Content Moderation model is given in Fig. 3 where the original image (left) depicts a white drug and a syringe. The Content Moderation model classifies it as ‘drug’ with confidence 1.0. The adversarial image (right) was generated with 197 queries, with an L_∞ constraint of $\epsilon = 16$. While this image can clearly be classified by a human as containing drugs, the target model classifies it as ‘safe’ with confidence 0.67. More successful attack images and the methodology followed to choose them are included in the supplementary material and at https://sunblaze-ucb.github.io/blackbox-attacks/.

4 Attacking State-of-the-art Defenses

In this section, we evaluate black-box attacks on defenses based on adversarial training and its variants. We focus on adversarial training based defenses as they aim to directly improve the robustness of DNNs, and are among the most effective defenses demonstrated so far in the literature [1]. These defenses make DNNs more robust by adding a loss term dependent on adversarial examples during training to count for adversarial examples. During training, the adversarial examples are computed with respect to the current state of the network using an appropriate method such as FGSM (standard) [8] and Iterative FGSM (iterative) [21]. Adversarial examples from other DNNs may also be included in the training set, leading to ensemble adversarial training [32].

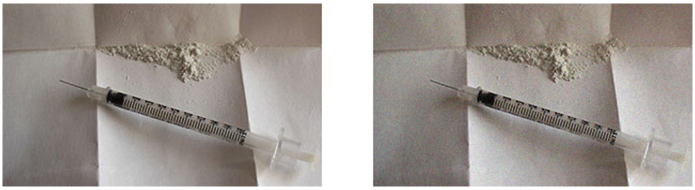


Fig. 3. Sample adversarial images of Gradient Estimation attacks on Clarifai’s Content Moderation model. Left: original image, classified as ‘drug’ with a confidence of 1.0. Right: adversarial example with $\epsilon = 16$, classified as ‘safe’ with a confidence of 0.67.

Adversarially Trained Model Setup. We train variants of Model A with the 3 adversarial training strategies described above using adversarial samples based on an L_∞ constraint of 0.3. Model $\mathsf{A}_{\text{adv-0.3}}$ is trained with FGS samples, while Model $\mathsf{A}_{\text{adv-iter-0.3}}$ is trained with Iterative FGS samples using $t = 40$ and $\alpha = 0.01$. For the model with ensemble training, Model $\mathsf{A}_{\text{adv-ens-0.3}}$ is trained with pre-generated FGS samples for Models A and two other DNN models as well as FGS samples. The source of the samples is chosen randomly for each minibatch during training. These models all achieve test accuracies of greater than 99%. For CIFAR-10, we train variants of Resnet-32 using adversarial samples with an L_∞ constraint of 8. Resnet-32 $_{\text{adv-8}}$ is trained with FGS samples with the same constraint, and Resnet-32 $_{\text{ens-adv-8}}$ is trained with pre-generated FGS samples from Resnet-32 and Std.-CNN as well as FGS samples. These have test accuracies of around 92%. Resnet-32 $_{\text{adv-iter-8}}$ is trained with iterative FGS samples using $t = 10$ and $\alpha = 1.0$ and has only 79.1% test accuracy.

4.1 Experimental Results

In this section, we focus on *untargeted* attacks on adversarially trained models, so the results in this section can be compared to those for undefended models in Table 2. Results for targeted attacks can be found in the supplementary material. In all cases, we find that Single-step Gradient Estimation attacks match the success rate of their white-box counterparts even with query reduction. Further discussion of these is contained in the supplementary material.

Adversarially Trained Models are not Robust to Gradient Estimation Attacks. Our experiments show that Iterative black-box attacks continue to work well even against adversarially trained networks (Table 3). For example, the Iterative Gradient Estimation attack using Finite Differences with a logit loss (**IFD-logit**) achieves attack success rates of 76.5% and 96.4% against Models $\mathsf{A}_{\text{adv-0.3}}$ and $\mathsf{A}_{\text{adv-ens-0.3}}$ respectively. This attack works well for CIFAR-10 models as well, achieving attack success rates of 100% against both Resnet-32 $_{\text{adv-8}}$ and Resnet-32 $_{\text{adv-ens-8}}$. This reduces slightly to 98% and 91% respectively when query reduction using random grouping is used. For both datasets, **IFD-logit** matches white-box attack performance. For MNIST, using PCA for query reduction, a 51% attack success rate is achieved for both Models $\mathsf{A}_{\text{adv-0.3}}$ and $\mathsf{A}_{\text{adv-ens-0.3}}$.

Model $\mathsf{A}_{\text{adv-iter-0.3}}$ is robust even against iterative attacks, with the highest black-box attack success rate achieved being 11.6%—marginally higher than the white-box attack success rate. On CIFAR-10, the iteratively trained model has poor performance on both benign and adversarial examples. The **IFD-logit** attack achieves an untargeted attack success rate of 55% on this model, which is lower than on the other adversarially trained models, but still significant. This is in line with Madry et al.'s observation [21] that iterative adversarial training needs models with large capacity for it to be effective. This highlights a limitation of this defense, since it is not clear what model capacity is needed, and the models we use already have a large number of parameters.

Table 3. Untargeted black-box attacks for models with adversarial training: attack success rates and average distortion $\Delta(\mathbf{X}, \mathbf{X}_{\text{adv}})$. Top: MNIST, $\epsilon = 0.3$. Bottom: CIFAR-10, $\epsilon = 8$.

Dataset	**White-box**		**Gradient Estimation (FD)**		**Gradient Estimation (Query Reduction)**			
MNIST	Single-step	Iterative	Single-step [1568]	Iterative [62720]	Single-step [∼ 200]		Iterative [8000]	
Models	FGS (logit)	IFGS (logit)	FD-logit	IFD-logit	PCA-100	RG-8	PCA-100	RG-8
$A_{\text{adv-0.3}}$	2.9 (6.0)	78.5 (3.1)	2.8 (5.9)	76.5 (3.1)	4.1 (5.8)	2.0 (5.3)	50.7 (4.2)	27.5 (2.4)
$A_{\text{adv-ens-0.3}}$	6.2 (6.2)	96.2 (2.7)	6.2 (6.3)	**96.4 (2.7)**	5.4 (6.2)	3.7 (6.4)	51.0 (3.9)	32.0 (2.1)
$A_{\text{adv-iter-0.3}}$	7.3 (7.5)	11.0 (3.6)	7.5 (7.2)	11.6 (3.5)	3.5 (4.0)	1.6 (4.2)	9.0 (2.8)	3.0 (1.4)
CIFAR-10	Single-step	Iterative	Single-step [6144]	Iterative [61440]	Single-step [∼ 800]		Iterative [∼ 8000]	
Models	FGS (logit)	IFGS (logit)	FD-logit	IFD-logit	PCA-400	RG-8	PCA-400	RG-8
Resnet-32 $_{\text{adv-8}}$	8.9 (438.8)	100.0 (73.7)	8.5 (401.9)	**100.0 (73.8)**	8.0 (402.1)	7.7 (401.8)	97.0 (151.3)	98.0 (92.9)
Resnet-32 $_{\text{adv-ens-8}}$	13.3 (437.9)	100.0 (85.3)	12.2 (399.8)	**100.0 (85.2)**	15.4 (396.1)	13.8 (395.9)	82.7 (178.7)	90.8 (106.6)
Resnet-32 $_{\text{adv-iter-8}}$	50.4 (346.6)	57.3 (252.4)	47.5 (331.1)	54.6 (196.3)	47.5 (344.1)	38.4 (341.4)	51.3 (256.6)	42.4 (153.3)

5 Possible Countermeasures and Conclusion

The Gradient Estimation attacks depend on model output probabilities to generate adversarial examples, so possible countermeasures can modify these to reduce their effectiveness. These modifications would, however, impact legitimate users as well. To validate this idea, we experimented with undefended models and *rounded off the output probabilities to two decimal places.* This successfully reduced the effectiveness of all Gradient Estimation attacks using the same parameters, with even the iterative variants achieving only as high as 28.0% attack success rates. We plan to explore query-efficient attacks that work in spite of these countermeasures in future work.

Overall, in this paper, we conduct a systematic analysis of black-box attacks on state-of-the-art classifiers and defenses. We propose Gradient Estimation attacks which achieve high attack success rates comparable with even white-box attacks. We apply random grouping and PCA-based methods to reduce the number of queries required while maintaining the effectiveness of the Gradient Estimation attack. We also apply our attacks against a real-world classifier and state-of-the-art defenses. All of our results show that Gradient Estimation attacks are very effective in a variety of settings, making the development of better defenses against black-box attacks an urgent task.

References

1. Athalye, A., Carlini, N., Wagner, D.: Obfuscated gradients give a false sense of security: circumventing defenses to adversarial examples. In: Proceedings of the 35th International Conference on Machine Learning (2018)
2. Brendel, W., Rauber, J., Bethge, M.: Decision-based adversarial attacks: reliable attacks against black-box machine learning models. In: International Conference on Learning Representations (2018)
3. Carlini, N., Wagner, D.: Towards evaluating the robustness of neural networks. In: IEEE Symposium on Security and Privacy (2017)

4. Chen, P.Y., Zhang, H., Sharma, Y., Yi, J., Hsieh, C.J.: Zoo: zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In: 11th ACM Workshop on Artificial Intelligence and Security (2017)
5. Clarifai — image & video recognition API. https://clarifai.com. Accessed 22 Aug 2017
6. Dang, H., Yue, H., Chang, E.C.: Evading classifiers by morphing in the dark. In: 24th ACM Conference on Computer and Communications Security (2017)
7. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press, Cambridge (2016)
8. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. In: International Conference on Learning Representations (2015)
9. Vision API - image content analysis—Google cloud platform. https://cloud.google.com/vision/. Accessed 22 Aug 2017
10. Gu, S., Rigazio, L.: Towards deep neural network architectures robust to adversarial examples. arXiv preprint arXiv:1412.5068 (2014)
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
12. Hildebrand, F.B.: Advanced Calculus for Applications, vol. 63. Prentice-Hall Englewood Cliffs, NJ (1962)
13. Ilyas, A., Engstrom, L., Athalye, A., Lin, J.: Black-box adversarial attacks with limited queries and information. In: Proceedings of the 35th International Conference on Machine Learning (2018)
14. Kennedy, J.: Particle swarm optimization. In: Sammut, C., Webb, G.I. (eds.) Encyclopedia of Machine Learning, pp. 760–766. Springer, Heidelberg (2011). https://doi.org/10.1007/978-0-387-30164-8
15. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images (2009)
16. Kurakin, A., Goodfellow, I., Bengio, S.: Adversarial examples in the physical world. arXiv preprint arXiv:1607.02533 (2016)
17. LeCun, Y., Cortes, C.: The MNIST database of handwritten digits (1998)
18. Liu, A.: Clarifai featured hack: block unwanted nudity in blog comments with disqus (2016). https://goo.gl/TCCVrR. Accessed 22 Aug 2017
19. Moosavi-Dezfooli, S.M., Fawzi, A., Fawzi, O., Frossard, P.: Universal adversarial perturbations. In: IEEE Conference on Computer Vision and Pattern Recognition (2016)
20. Moosavi-Dezfooli, S.M., Fawzi, A., Frossard, P.: Deepfool: a simple and accurate method to fool deep neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition (2016)
21. Mądry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. In: International Conference on Learning Representations (2018)
22. Narodytska, N., Kasiviswanathan, S.P.: Simple black-box adversarial perturbations for deep networks. arXiv preprint arXiv:1612.06299 (2016)
23. Nelson, B., et al.: Query strategies for evading convex-inducing classifiers. J. Mach. Learn. Res. **13**(1), 1293–1332 (2012)
24. Papernot, N., McDaniel, P., Goodfellow, I.: Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. arXiv preprint arXiv:1605.07277 (2016)

25. Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Berkay Celik, Z., Swami, A.: Practical black-box attacks against deep learning systems using adversarial examples. In: ACM Asia Conference on Computer and Communications Security (2017)
26. Salimans, T., Ho, J., Chen, X., Sutskever, I.: Evolution strategies as a scalable alternative to reinforcement learning. arXiv preprint arXiv:1703.03864 (2017)
27. Sharif, M., Bhagavatula, S., Bauer, L., Reiter, M.K.: Accessorize to a crime: real and stealthy attacks on state-of-the-art face recognition. In: ACM Conference on Computer and Communications Security (2016)
28. Shlens, J.: A tutorial on principal component analysis. arXiv preprint arXiv:1404.1100 (2014)
29. Spall, J.C.: Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. IEEE Trans. Autom. Control **37**(3), 332–341 (1992)
30. Spall, J.C.: Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control, vol. 65. Wiley, Hoboken (2005)
31. Szegedy, C., et al.: Intriguing properties of neural networks. In: International Conference on Learning Representations (2014)
32. Tramèr, F., Kurakin, A., Papernot, N., Boneh, D., McDaniel, P.: Ensemble adversarial training: attacks and defenses. In: International Conference on Learning Representations (2018)
33. Watson visual recognition. https://www.ibm.com/watson/services/visual-recognition/. Accessed 27 Aug 2017
34. Wright, S.J., Nocedal, J.: Numerical optimization. Springer Sci. **35**(67–68), 7 (1999)
35. Xu, W., Qi, Y., Evans, D.: Automatically evading classifiers. In: Proceedings of the 2016 Network and Distributed Systems Symposium (2016)
36. Zagoruyko, S., Komodakis, N.: Wide residual networks. arXiv preprint arXiv:1605.07146 (2016)