A LANGEVIN DYNAMICS APPROACH TO GENERATING SPARSE ADVERSARIAL PERTURBATIONS

Aram-Alexandre Pooladian***, Alexander Iannantuono, Chris Finlay & Adam Oberman**
Department of Mathematics and Statistics
McGill University
Montreal, QC, Canada

ABSTRACT

Deep neural networks are vulnerable to small changes in the input that lead to misclassification, called *adversarial images*. We present an efficient approach to generating sparse adversarial images, i.e. small with respect to the cardinality function ℓ_0, without using gradient information. The lack of gradient oracle is of interest in the case of adversarial attacks: in a practical setting, one can easily query the model to determine whether the image is misclassified or not but rarely will the full network structure be provided. Our method, ProxWalk, is inspired by Metropolis-Adjusted Langevin dynamics; a method for modeling random walks, and the proximal variant. We present results on MNIST, Fashion-MNIST, CIFAR10, and CIFAR100 datasets, and demonstrate that our decision-based attack is on par with modern sparse white-box attacks.

1 INTRODUCTION

Deep neural networks are vulnerable to adversarial inputs: small perturbations in the input space that cause misclassification Szegedy et al. (2014). This vulnerability is a potentially grave security risk in real-world applications (e.g. tampering with stop signs for self-driving cars [cite]). As a result, researchers are simultaneously developping defensive measures to these attacks, referred to as *robustness*. Often, robustness methods are catered for a specific attack, and do not achieve global robustness. For example, the current gold-standard for adversarial robustness in the ℓ_∞ norm Madry et al. (2017), does not translate to sparse perturbations Schott et al. (2018); Pooladian et al. (2019). Adversarial attacks can be categorized in the following ways: how much of the network the attacker has access to, either full network knowledge or only model outputs; and the dissimilarity metric used to determine how small the perturbation is.

The capacity of an attacker to generate an adversary is greatly limited by how much knowledge of the network they have. If the model structure and trained weights are available, an attacker can use gradient information to get sufficiently close to the decision boundary, or ascent the loss landscape to find an adversary. These are known as white-box attacks, and are heavily studied in the current literature Carlini & Wagner (2016); Kurakin et al. (2016); Modas et al. (2018); Moosavi-Dezfooli et al. (2015). When only model output is available the attacks are called black-box. The output of a model is typically a vector in $\mathbb{R}^c$ (the model probabilities are often "projected" onto the probability simplex, called the model *scores*), where c denotes the number of classes. When using model scores , zeroth order optimization methods can be considered for generating adversarial examples Ilyas et al. (2018a;b); Chen et al. (2017). A further limitation is attacking when only knowing the model *decision*; this is typically a more expensive endeavor Chen & Jordan (2019); Brendel et al. (2017).

Adversarial images are often measured by a dissimilarity metric (with respect to their original image). Often, these metrics are the ℓ_p norms, with $p \in \{1, 2, \infty\}$, and the ℓ_0 counting "norm", which measures true sparsity of the perturbation. Due to algorithmic simplicity, the first adversarial attacks were based on ℓ_∞ and ℓ_2 norms, leveraging the dual problem to find an efficient adversary Goodfellow et al. (2014). Iterative versions followed Kurakin et al. (2016). Perturbations generated for the

ℓ_2 or ℓ_∞ norm case can potentially require all pixels to be altered while being imperceptible to the human eye. In some settings, ℓ_0 perturbations are a more "realistic" threat model, as it answers the question: if an adversary is allowed to perturb at most, say, 1% of the total number of pixels, which pixels would be selected?

Contributions

This paper introduces ProxWalk, a *decision-based* (i.e. black box) adversarial attack method, catered for generating adversarial perturbations that also minimize the ℓ_0 constraint problem. Our method is an adaptation of a random-walk approach for finding samples of target distributions, called Metropolis-Adjusted Langevin Algorithm, and the proximal variant. ProxWalk requires few model queries to generate an adversarial example on benchmark datasets, and is on par with modern white-box attacks in ℓ_0. Our algorithm will be made publicly available on Github, which can be used to efficiently determine a networks robustness to sparse perturbations on small/medium sized datasets.

2 Background material

2.1 Formulation of adversarial attacks

An image-label pair is defined by $(x, y) \in \mathcal{X} \times \Delta_c$, where $\mathcal{X}$ is the image space and Δ_c is the unit-simplex for c labels (the label belongs on one of the vertices). A trained model is defined by $f : \mathcal{X} \to \Delta_c$, with misclassification region

$$\mathcal{M}_f := \{u \in \mathcal{X} \mid C(f(u)) \neq y\},$$

where $C(f(\cdot))$ is the classification function for the trained network. Formally, an adversarial perturbation with respect to a metric $m(\cdot; x)$ is the minimizer of the following constrained optimization problem:

$$\min_{u \in \mathcal{X}} \; m(u; x) \quad \text{subject to} \quad u \in \mathcal{M}_f. \tag{1}$$

The hard-constraint of the decision boundary makes the exact problem hard, but several relaxed versions of this problem have been proposed Carlini & Wagner (2016); Pooladian et al. (2019); Finlay et al. (2019). A common surrogate for solving this problem is to maximize a loss function subject to a constraint on the perturbation to the clean image;

$$\min_{u \in \mathcal{X}} \mathcal{L}(u) \quad \text{subject to} \quad m(u; x) \leq \varepsilon,$$

where ε is an arbitrary threshold. This last formulation has lead to attacks such as the Iterative Fast Gradient Method, where the ℓ_2 and ℓ_∞ norms are the constraints Kurakin et al. (2016); Madry et al. (2017); Goodfellow et al. (2014). Other white-box attacks target the model logits, and can evade gradient obfuscation Athalye et al. (2018); Carlini & Wagner (2016).

Decision-based black-box attacks solve (1), and often require more intelligent ways of moving the adversarial image in the right direction, as there is no gradient information of the model. Boundary Attack Brendel et al. (2017) uses rejection sampling to get arbitrary close to the decision boundary; this comes at the cost of an intractable number of model queries. HopSkipJump attack Chen & Jordan (2019), use sampling methods to generate gradient approximations that can move along the decision boundary; the former attack is targeted to the ℓ_2 norm but the latter is applicable to either the ℓ_2 or ℓ_∞ norm.

2.2 Proximal operators

Proximal operators are often used in non-smooth minimization problems, and have recently been of interest in the deep learning community in a variety of settings Bai et al. (2018); Pooladian et al. (2019). We denote the class of proper, convex and lower semicontinuous functions by Γ_0. Examples of such functions are ℓ_p norms with $p \in [1, \infty]$, and indicator functions on closed, convex sets $C \subseteq \mathbb{R}^n$; we indicate the latter by δ_C.

Consider a function $f : \mathbb{R}^n \to \mathbb{R}$ that is in Γ_0, and fix $x \in \mathbb{R}^n$ and $\lambda > 0$. The *Moreau envelope* of f with parameter λ at x is defined as

$$e_\lambda f(x) := \min_{u \in \mathbb{R}^n} f(u) + \frac{1}{2\lambda}\|x - u\|_2^2,$$

and the associated *proximal operator* is defined as the minimizer of $e_\lambda f(x)$, denoted $P_\lambda f(x)$ Rockafellar & Wets (2009); Beck (2017). The Moreau envelope of a function is a smooth lower bound of f, which is obtained by computing the proximal operator. In fact, f is not necessarily required to be convex, as illustrated in Figure 1, using a piece-wise quadratic function that is weakly convex.

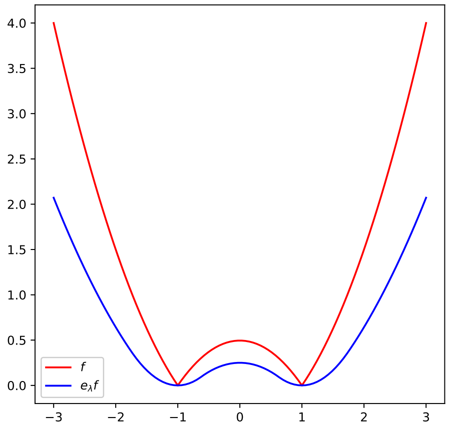


Figure 1: The primary function (in red) is a piecewise (non-differentiable) quadratic, but the Moreau (in blue) is a smooth, lower-bound approximation.

In some cases, the proximal operator of f can be computed computed analytically, for example, in the case of the centered ℓ_0 function:

$$P_{\mu\|\cdot\ -\ x\|_0}(z) = x + \mathcal{H}_{\sqrt{2\mu}}(z - x);$$

where $\mathcal{H}_\alpha(s) = s\mathbf{1}_{\{|s|>\alpha\}}(s)$ is the hard-thresholding operator ($\mathbf{1}_A(\cdot)$ is the 0-1 indicator function for the set A), and acts component-wise in the case of vector arguments. The derivations for this proximal operator, and many others, can be found in Beck (2017). A useful property of the Moreau envelope is the following characterization of the gradient Rockafellar & Wets (2009):

Proposition 2.1. *Let $f : \mathbb{R}^n \to \mathbb{R}$ and $f \in \Gamma_0$. For $\lambda > 0$, $e_\lambda f(x)$ is differentiable, with*

$$\nabla e_\lambda f(x) = \frac{1}{\lambda}(x - P_\lambda f(x)).$$

Furthermore, the gradient is λ^{-1} Lipschitz continuous.

3 Our method: ProxWalk

ProxWalk is heavily based on the Metropolis-Adjusted Langevin Algorithm (MALA) Robert & Casella (2005); more specifically, the proximal variant, Proximal-MALA (P-MALA), introduced recently in Pereyra (2016). We discuss the relevant details for adopting the P-MALA algorithm in the following section.

3.1 Proximal Metropolis-Adjusted Langevin Algorithm

Let $f : \mathbb{R}^n \to \mathbb{R}$ with $f \in \Gamma_0$, and $\lim_{\|x\|\to\infty} f(x) = +\infty$. In the P-MALA framework, the minimization of f is re-written as finding elements of the distribution function with density

$$\pi(x) \propto \exp\{-f(x)\}.$$

The aforementioned conditions on f allow this to be a meaningful probability density function. Consequently, for any $\lambda > 0$, the λ-*Moreau approximation* of $\pi(x)$ is

$$\pi_\lambda(x) \propto \exp\{-e_\lambda f(x)\}.$$

For a d-dimensional random variable, the discretized MALA update step is written as

$$X^{(k+1)} = X^{(k)} + \frac{\tau}{2}\nabla \log(\pi(X^{(k)})) + \sqrt{\tau}\xi^{(k)},$$

where $\{\xi^{(k)}\}_{k=1}^{K} \sim N(0, I_d)$. This update is derived from the Euler-Maruyama discretization of the stochastic overdamped Langevin equation; which is discussed in more detail in Robert & Casella (2005); Pereyra (2016); Roberts & Rosenthal (1998). In the P-MALA setting, we optimize over the λ-Moreau approximation instead. Using Proposition 2.1, we have the following update rule for the iterates:

$$\begin{aligned} X^{(k+1)} &= X^{(k)} + \frac{\tau}{2}\nabla \log(\pi_\lambda(X^{(k)})) + \sqrt{\tau}\xi^{(k)} \\ &= X^{(k)} - \frac{\tau}{2}\nabla e_\lambda f(X^{(k)}) + \sqrt{\tau}\xi^{(k)} \\ &= X^{(k)} - \frac{\tau}{2}\left(\frac{1}{\lambda}(X^{(k)} - P_\lambda f(X^{(k)}))\right) + \sqrt{\tau}\xi^{(k)} \\ &= \left(1 - \frac{\tau}{2\lambda}\right) X^{(k)} + \frac{\tau}{2\lambda} P_\lambda f(X^{(k)}) + \sqrt{\tau}\xi^{(k)}, \end{aligned}$$

In the special case where $\tau = 2\lambda$, we recover the Proximal (Unadjusted) Langevin Algorithm,

$$X^{(k+1)} = P_\lambda f(X^{(k)}) + \sqrt{2\lambda}\xi^{(k)}.$$

This amounts to a proximal-point algorithm Beck (2017), with added noise at each step. For the Proximal Metropolis-Adjusted Langevin algorithm (P-MALA), we let Y denote a proposed state of the algorithm; it is accepted with probability

$$\alpha := \min\left\{1, \frac{\pi(Y)q(X^{(k)}|Y)}{\pi(X^{(k)})q(Y|X^{(k)})}\right\}, \tag{2}$$

where $q(a \mid b) \propto \exp\left\{-(4\tau)^{-1}\|a - P_\lambda f(b)\|_2^2\right\}$ is a transition density. The Metropolis-Hastings adjustment is necessary for convergence of the algorithm, empirically and theoretically Roberts & Tweedie (1996); Pereyra (2016).

3.2 Our algorithm

We attempt to solve the problem in (1), where our perturbations are to be as sparse as possible i.e. small with respect to ℓ_0. We let $\mathcal{M}_f$ denote the misclassified region with respect to our trained model f. In the P-MALA framework, the minimization problem of (1) is equivalent to finding elements from the density function of the form

$$\pi(u) \propto \exp\{-\varphi(u) - \delta_{\mathcal{M}_f}(u)\}, \tag{3}$$

where $\varphi(u) := \|u - x\|_0$. The constraint over the set $\mathcal{M}_f$ is necessary in the accept-reject step so as to guarantee that we always propose adversarial candidates. We note that $\mathcal{M}_f$ is neither a closed and/or a convex set, hence the optimization procedure is not trivial in the black-box setting.

The update step (2) can be used for a wide class of image-based problems, such as denoising, where Gaussian noise is a suitable assumption. To implement this into the adversarial attack setting, we require a change in noise model, namely we want our random walk to be sparse in the first place. To ensure this, one can use a hard-thresholding operator, or keep the κ largest components. We use the following operator operator, $T(\xi;\tau) := \tau \text{sign}(\xi)\mathbf{1}_{\{|\xi|>\tau\}}(\xi)$, where $\xi \sim \mathcal{N}(0, I_d)$. Note that we incorporate the step-size into the operator, and perturb the pixel entirely. Finally, we move the random perturbation before the proximal update, which gives the following general update rule

$$X^{(k+1)} = \left(1 - \frac{\tau}{2\lambda}\right) X^{(k)} + \frac{\tau}{2\lambda} P_\lambda f(X^{(k)} + T(\xi^{(k)};\tau)). \tag{4}$$

Our algorithm begins by initially perturbing the image with uniform noise until misclassified. We generate a proposed iterate via (4), using the same accept-reject ratio as P-MALA. We incorporate $\delta_{\mathcal{M}_f}$ (hence model query) when computing the ratio, where a correctly classified image would result in immediate rejection, and repeat.

Algorithm 1 ProxWalk

```
Input: image-label pair $(x, y)$, trained model $f$, transformation $T$, $k = 0$
Hyperparameters: $k_{\max} \in \mathbb{N}$, and $\lambda, \tau > 0$.
Initialize $x^{(0)}$ to be misclassified
for $k = 0, 1, 2, \ldots, k_{\max}$ do                         ▷ Total number of model queries
    Sample $\xi \sim N(0, I_d)$
    Set $v^{(k)} = (2\lambda)^{-1} \left(P_\lambda \varphi(x^{(k)} + T(\xi; \tau)) - x^{(k)}\right)$
    Propose $y^* = \text{Project}\left(x^{(k)} + v^{(k)}; \mathcal{X}\right)$
    Sample $u \sim U(0, 1)$
    Compute $\log(\pi(y^*))$ using (3), and $\log(\alpha)$, where $\alpha$ is given by (2)
    if $\log(u) < \log(\alpha)$ then
        $x^{(k+1)} = y^*$
    else
        $x^{(k+1)} = x^{(k)}$
    end if
end for
Return $x^{(k_{\max})}$
```

4 Experiments

We compare against gradient-based and gradient-free methods for the ℓ_0 norm case: ProxLogBarrier, SparseFool and Jacobian Saliency Map Attack (JSMA) will act as a baselines for gradient-based algorithms; Pointwise will be our benchmark for decision-based attacks. ProxLogBarrier uses the proximal gradient method to solve a regularized form of (1), and is applicable to a wide class of dissimilarity metrics Pooladian et al. (2019). SparseFool solves an ℓ_1 approximate problem via linearizing near the decision boundary Modas et al. (2018); though ultimately it aims to minimize the number of perturbed pixels. JSMA perturbs pixels based on their saliency score, which is further described in Papernot et al. (2015). Pointwise is a binary-search like algorithm, that passes through all the pixels, starting from an initial perturbed image. We initialized ProxWalk and Pointwise with the same uniform noise, and do not include these model evaluations when comparing model queries.

We use the LeNet architecture for MNIST and Fashion-MNIST; achieving $\sim$99.50% and $\sim$91.50%, respectively. For both CIFAR10 and CIFAR100 datasets, we use a ResNeXt architecture[1]; achieving accuracies of $\sim$94.50% and $\sim$91.50% (Top5 percent), respectively. For all datasets, we report the median number of pixels pertubed, and the percent error at two thresholds. On MNIST and Fashion-MNIST, we report the percent error at 10 and 30 pixels perturbed, which corresponds to roughly 1.2% and 3.8%, respectively. On CIFAR10 and CIFAR100, the thresholds used are 30 pixels and 80 pixels, corresponding to roughly 0.97%, and 2.6%, respectively. In the case of CIFAR100, we consider the problem of Top5 misclassification. We attack 1000 randomly chosen images from the MNIST, Fashion-MNIST, and CIFAR10 datasets; we only attack 500 images for CIFAR100.

Algorithm Hyperparameters

Like any random-walk based algorithm, we performed a hyperparameter search. On MNIST we use $\tau = 0.2, \lambda = 0.1$, Fashion-MNIST, $\tau = 0.25, \lambda = 0.1$. For both CIFAR10 and CIFAR100, we fix $\tau = 1.0, \lambda = 0.5$. This roughly corresponds to ensuring that $\tau = 2\lambda$, which gives the following

[1] We use a ResNeXt34 (2x32) on CIFAR10, and a ResNeXt34 (4x32) for CIFAR100.

approximate step size,

$$X^{(k+1)} \simeq P_\lambda \varphi(X^{(k)} + T(\xi; \tau)).$$

This can also be interpreted as a proximal gradient approach, where $T(\xi; \tau)$ represents the gradient approaching the decision boundary.

4.1 Results

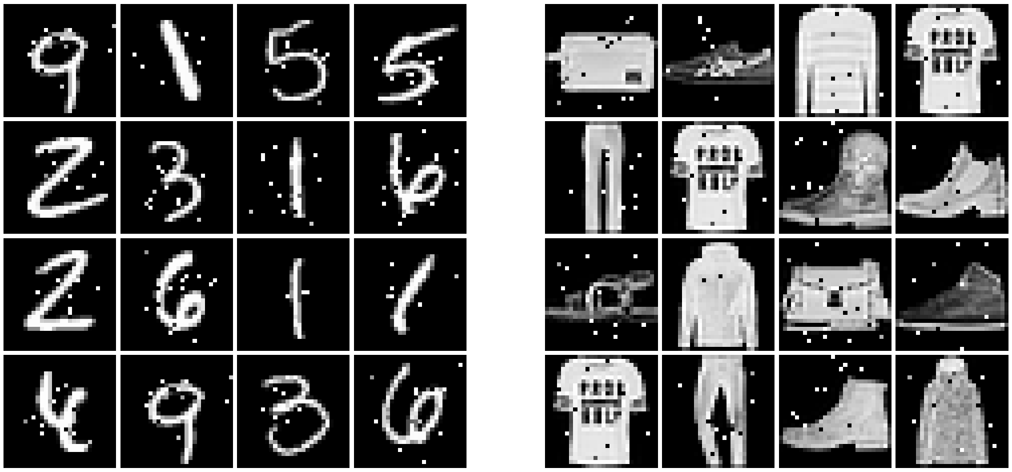
Tables 1 and 2 show the median distance of ProxWalk against a blackbox attack, and three white-box attacks. We report our attack at 1K, 2.5K and 5K model queries, and leave the others at default implementations according according to the FoolBox repository for SparseFool, JSMA, Pointwise (note that Pointwise at default has significantly more more queries than ours) and the Github repository for ProxLogBarrier.

For MNIST and Fashion-MNIST, ProxLogBarrier surpasses all other attacks considered by a wide margin. However, in just 1K model queries, ProxWalk finds a stable distribution of adversaries that is smaller than SparseFool, JMSA, and Pointwise. On Fashion-MNIST, our performance improves as a function of model queries, and eventually we are on-par or better than gradient based attacks. At 5K model queries, the median percent pixels perturbed (MPPP) on MNIST is roughly 1.6%, and 2.2% on Fashion-MNIST. On CIFAR10 and CIFAR100, we observe similar behaviour except we are closer to ProxLogBarrier than before, and significantly better than SparseFool and JSMA across the considered thresholds. The MPPP for CIFAR10 is under half a percent, and similarly for CIFAR100 — this is much lower than what we have found for SparseFool and JSMA, and lower than what SparseFool reports in their paper (which is not reflected in our tables).

We highlight that our parameter choices appear to have a "global" effect on the datasets/network. That is, for a parameter pair (λ, τ), with the exception of Fashion-MNIST, the algorithm was able to find more sparse perturbations than two out of three gradient based attacks within just 1000 model queries.

Table 1: Attack results for MNIST and Fashion-MNIST

	MNIST			Fashion-MNIST		
	% error at		median distance	% error at		median distance
	$\varepsilon = 10$	$\varepsilon = 30$		$\varepsilon = 10$	$\varepsilon = 30$	
ProxWalk (1K)	34.10	77.6	14	31.20	45.00	76.5
ProxWalk (2.5K)	38.20	81.9	13	32.90	52.10	25
ProxWalk (5K)	39.80	85.10	13	32.80	57.10	18
Pointwise (Default)	2.70	23.80	42	12.50	28.60	47
ProxLogBarrier	80.30	97.89	6	60.69	87.50	7
SparseFool	8.90	63.08	26	30.70	76.60	19
JSMA	7.10	32.80	44	25.70	52.70	26



(a) ℓ_0 attacks on MNIST

(b) ℓ_0 attacks on Fashion-MNIST

Figure 2: Sample of adversarial images, using at most 5K model queries.

Table 2: Attack results for CIFAR10 and CIFAR100, using a ResNeXt34 architecture

	CIFAR10			CIFAR100		
	% error at		median distance	% error at		median distance
	$\varepsilon = 30$	$\varepsilon = 80$		$\varepsilon = 30$	$\varepsilon = 80$	
ProxWalk (1K)	56.5	76.4	17.0	54.8	76.6	18.0
ProxWalk (2.5K)	60.6	79.4	16.0	58.8	78.0	16.5
ProxWalk (5K)	64.6	80.9	14.0	63.0	79.0	15.0
Pointwise (No cap)	16.6	67.3	64.0	21.0	64.4	64.5
ProxLogBarrier	72.0	87.90	12	65.40	86.40	16
JSMA	27.70	42.80	113	38.40	69.40	44

5 Discussion

We have presented a novel approach to adversarial image generation in the case of sparse perturbations without incorporating gradient information. Our algorithm is based on existing literature in (proximal) Metropolis-adjusted Langevin algorithms, where the adaptation is motivated and intuitive. The effectiveness of our attack was demonstrated on several datasets, compared against publicly available algorithms that are catered for the sparse perturbation problem. With respect to this collection of attacks, ProxWalk is second only to ProxLogBarrier (which uses gradient information).

ProxWalk requires very few model queries to reach a sufficiently low median number of perturbed pixels. We also remark the similarity in the chosen hyper-parameters across like-datasets. We believe this is either a reflection of the network used (a standard LeNet for both gray-scale datasets, and ResNeXt-based networks on the RGB datasets), or a reflection of the distribution of adversarial images. In either case, it is remarkable that a parameter pair is able to find sparse adversarial perturbations that are smaller than current gradient-based attacks (with respect to the median).

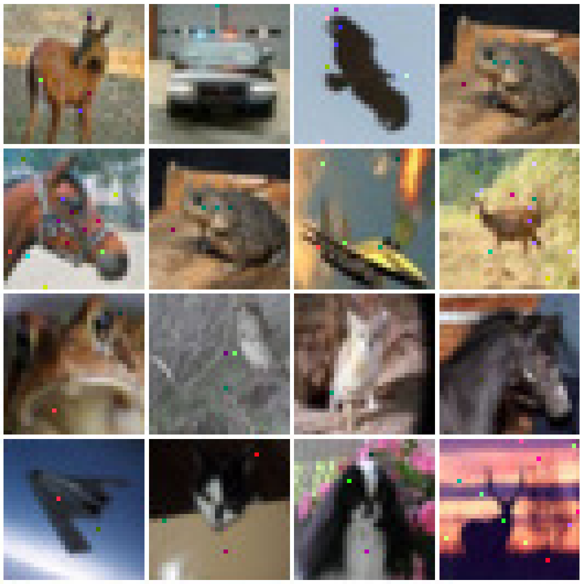


Figure 3: Adversarial images for CIFAR10 generated in at most 5K model queries via ProxWalk

REFERENCES

Anish Athalye, Nicholas Carlini, and David A. Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, pp. 274–283, 2018. URL http://proceedings.mlr.press/v80/athalye18a.html.

Yu Bai, Yu-Xiang Wang, and Edo Liberty. Proxquant: Quantized neural networks via proximal operators. *CoRR*, abs/1810.00861, 2018. URL http://arxiv.org/abs/1810.00861.

Amir Beck. *First-order methods in optimization*. 2017.

Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. *arXiv preprint arXiv:1712.04248*, 2017.

Nicholas Carlini and David A. Wagner. Towards evaluating the robustness of neural networks. *CoRR*, abs/1608.04644, 2016. URL http://arxiv.org/abs/1608.04644.

Jianbo Chen and Michael I. Jordan. Boundary attack++: Query-efficient decision-based adversarial attack. *CoRR*, abs/1904.02144, 2019. URL http://arxiv.org/abs/1904.02144.

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. ACM, 2017.

Chris Finlay, Aram-Alexandre Pooladian, and Adam M. Oberman. The logbarrier adversarial attack: making effective use of decision boundary information. *CoRR*, abs/1903.10396, 2019. URL http://arxiv.org/abs/1903.10396.

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

Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. *CoRR*, abs/1804.08598, 2018a. URL http://arxiv.org/abs/1804.08598.

Andrew Ilyas, Logan Engstrom, and Aleksander Madry. Prior convictions: Black-box adversarial attacks with bandits and priors. *arXiv preprint arXiv:1807.07978*, 2018b.

Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *CoRR*, abs/1607.02533, 2016. URL http://arxiv.org/abs/1607.02533.

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.

Apostolos Modas, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. Sparsefool: a few pixels make a big difference. *CoRR*, abs/1811.02248, 2018. URL http://arxiv.org/abs/1811.02248.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. *CoRR*, abs/1511.04599, 2015. URL http://arxiv.org/abs/1511.04599.

Nicolas Papernot, Patrick D. McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. *CoRR*, abs/1511.07528, 2015. URL http://arxiv.org/abs/1511.07528.

Marcelo Pereyra. Proximal markov chain monte carlo algorithms. *Statistics and Computing*, 26(4): 745–760, 2016.

Aram-Alexandre Pooladian, Chris Finlay, Tim Hoheisel, and Adam M. Oberman. A principled approach for generating adversarial images under non-smooth dissimilarity metrics. *arXiv preprint arXiv:1908.01667*, 2019. URL http://arxiv.org/abs/1908.01667.

Christian P. Robert and George Casella. *Monte Carlo Statistical Methods (Springer Texts in Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2005. ISBN 0387212396.

Gareth O Roberts and Jeffrey S Rosenthal. Optimal scaling of discrete approximations to langevin diffusions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 60(1): 255–268, 1998.

Gareth O. Roberts and Richard L. Tweedie. Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli*, 2(4):341–363, 12 1996. URL https://projecteuclid.org:443/euclid.bj/1178291835.

R Tyrrell Rockafellar and Roger J-B Wets. *Variational analysis*, volume 317. Springer Science & Business Media, 2009.

Lukas Schott, Jonas Rauber, Wieland Brendel, and Matthias Bethge. Robust perception through analysis by synthesis. *CoRR*, abs/1805.09190, 2018. URL http://arxiv.org/abs/1805.09190.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL http://arxiv.org/abs/1312.6199.