AdvFlow: Inconspicuous Black-box Adversarial Attacks using Normalizing Flows

Hadi M. Dolatabadi, Sarah Erfani, and Christopher Leckie
School of Computing and Information Systems, The University of Melbourne

Abstract

- **Motivation**: we want to construct black-box adversarial examples that come from a similar distribution as the clean data.
- **Proposal**: we utilize *normalizing flows* in conjunction with *natural evolution strategies (NES)* to build black-box adversarial attacks called AdvFlow.
- **Key features**:
  1. AdvFlow distribution is similar to the clean data.
  2. AdvFlow perturbations have a data-like structure.
  3. AdvFlow outperforms well-known black-box attacks on defended classifiers.

Normalizing Flows

Change-of-variables formula:

- Random vector $\mathbf{Z} \sim p_{\mathbf{Z}}(\mathbf{z})$
- Invertible and differentiable function $\mathbf{f}(\cdot)$
- Random vector $\mathbf{X} = \mathbf{f}(\mathbf{Z})$

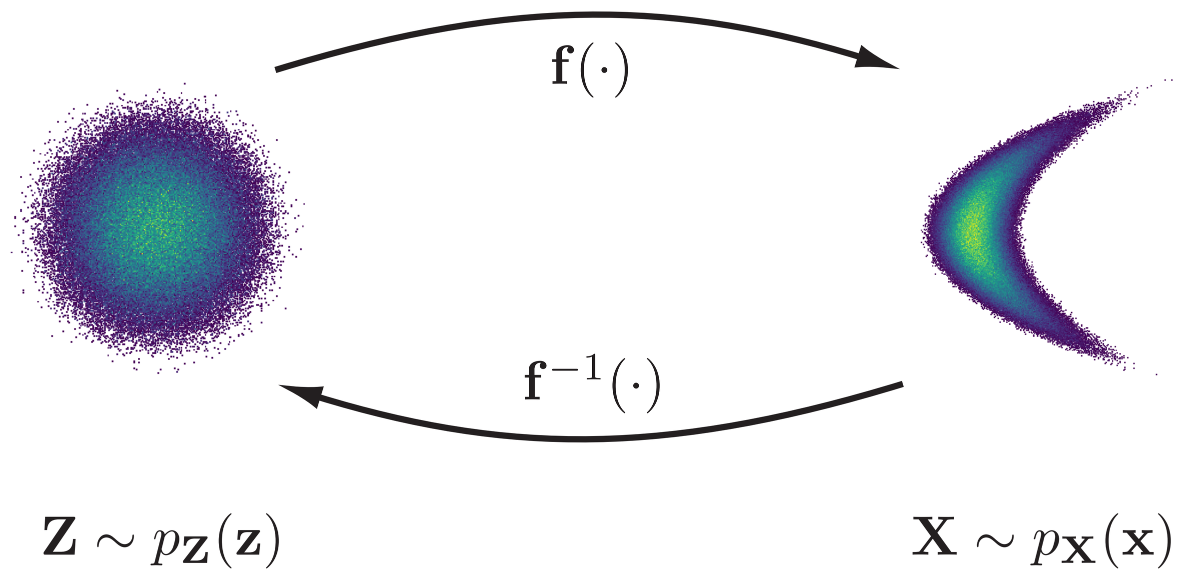
$$p_{\mathbf{X}}(\mathbf{x}) = p_{\mathbf{Z}}(\mathbf{z}) \left|\det\left(\nabla_{\mathbf{z}} \mathbf{f}(\mathbf{z})\right)\right|^{-1}$$

Normalizing flows:

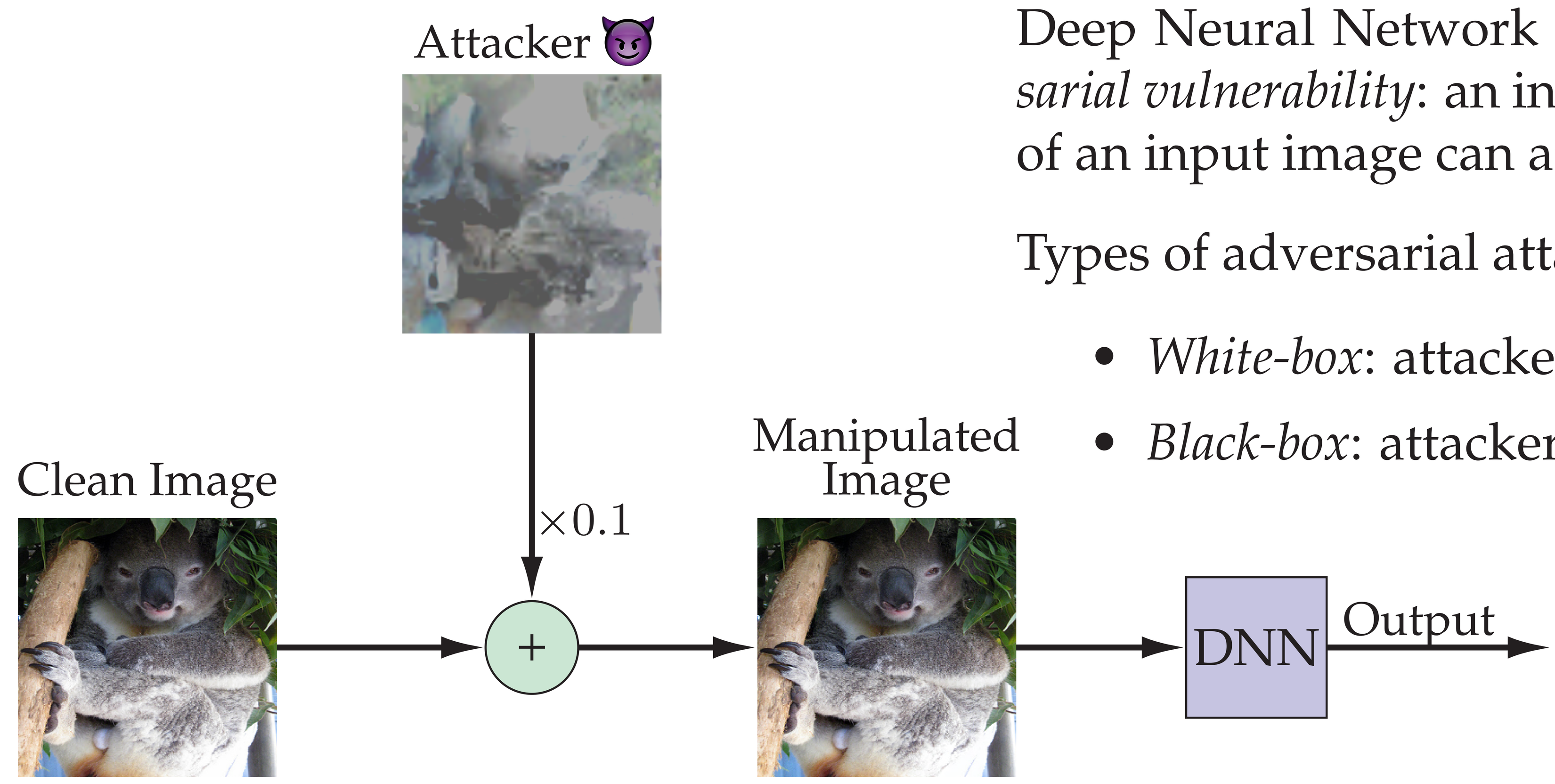
- $\mathbf{Z}$: simple base random variable (e.g. standard normal)
- $\mathbf{f}_{\boldsymbol{\theta}}(\cdot)$: composition of invertible neural nets

$$\mathbf{f}_{\boldsymbol{\theta}}(\cdot) = \left(\mathbf{f}_K \circ \mathbf{f}_{K-1} \circ \mathbf{f}_2 \circ \mathbf{f}_1\right)(\cdot)$$

- Fitting $\mathbf{f}_{\boldsymbol{\theta}}(\cdot)$ to observations through maximum likelihood objective



Adversarial Attacks



Expectation: It's a Koala.
Reality: It's an Airplane!

Deep Neural Network (DNN) classifiers suffer from *adversarial vulnerability*: an intentional, small adjustment to pixels of an input image can affect the DNN decision adversely!

Types of adversarial attacks:

- *White-box*: attacker knows the classifier entirely.
- *Black-box*: attacker can only query the classifier.

Adversarial Example Generation

It can be shown that adversarial example generation is equivalent to the following optimization problem:

$$\mathbf{x}_{adv} = \underset{\mathbf{x}' \in \mathcal{S}(\mathbf{x})}{\arg\min}\, \mathcal{L}(\mathbf{x}'), \tag{1}$$

where

- $\mathcal{L}(\cdot)$ is an objective involving the classifier, and
- $\mathcal{S}(\mathbf{x})$ is the set of similar data to the clean one $\mathbf{x}$.

Natural Evolution Strategies (NES)

Instead of optimizing Eq. (1) directly, define a parametric *search distribution* $p(\mathbf{x}'|\boldsymbol{\psi})$ on $\mathbf{x}'$ and replace the Eq. (1) objective with:

$$J(\boldsymbol{\psi}) = \mathbb{E}_{p(\mathbf{x}'|\boldsymbol{\psi})}\left[\mathcal{L}(\mathbf{x}')\right]. \tag{2}$$

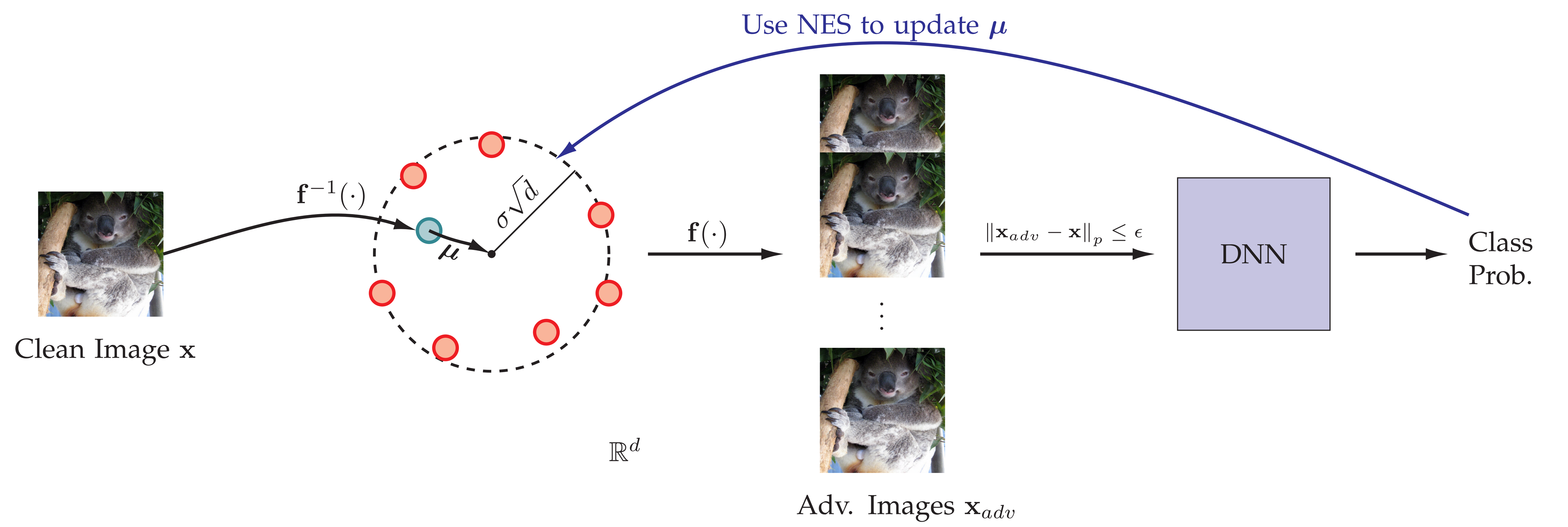
It can be shown that [1]

$$\nabla_{\boldsymbol{\psi}} J(\boldsymbol{\psi}) = \mathbb{E}_{p(\mathbf{x}'|\boldsymbol{\psi})}\left[\mathcal{L}(\mathbf{x}')\nabla_{\boldsymbol{\psi}} \log\left(p(\mathbf{x}'|\boldsymbol{\psi})\right)\right]. \tag{3}$$

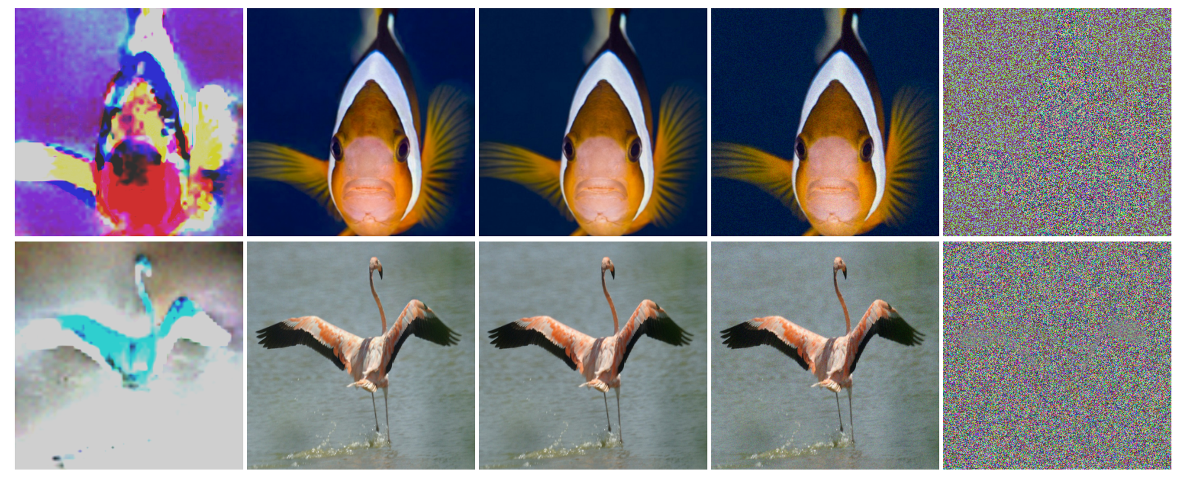
This only involves querying $\mathcal{L}(\cdot)$, making it suitable for black-box optimization/attacks.

Our Approach: AdvFlow

1. Pre-train a flow-based model $\mathbf{f}(\cdot)$ on clean data.
2. Change the flow-based model base random vector from $\mathcal{N}(\mathbf{z}|\mathbf{0}, I)$ to $\mathcal{N}(\mathbf{z}|\boldsymbol{\mu}, \sigma^2 I)$.
3. Use this density as the search distribution $p(\mathbf{x}'|\boldsymbol{\psi})$.
4. Given a target image, adjust $\boldsymbol{\psi} = \{\boldsymbol{\mu}, \sigma\}$ using NES to turn the clean data distribution into an adversarial one.
5. Generate an adversarial example by sampling from $p(\mathbf{x}'|\boldsymbol{\psi})$.



Adversarial Examples



Diff. (×10) | AdvFlow | Clean Image | $\mathcal{N}$Attack | Diff. (×10)

- **Takeaway**: AdvFlow generates perturbations that take the structure of the data into account, making them less detectable!

Adv. Example Detection

Data	Detector Method	AUROC(%) ↑ $\mathcal{N}$Attack	AUROC(%) ↑ AdvFlow
CIFAR-10	LID	78.69	**57.59**
	Mahalanobis	97.95	**66.85**
	Res-Flow	97.90	**67.03**
SVHN	LID	**57.70**	61.11
	Mahalanobis	73.17	**64.72**
	Res-Flow	69.70	**64.68**

- **Takeaway**: AdvFlow generates adversarial examples that are closer to the true data distribution!

Attack Success Rate (%)

Attack / Data	Bandits / $\mathcal{N}$Attack / SimBA / AdvFlow — CIFAR-10	Bandits / $\mathcal{N}$Attack / SimBA / AdvFlow — ImageNet
Van.	96.75 / 99.85 / **99.96** / 99.37	95.79 / **99.47** / 98.42 / 95.58
Def.	45.20 / 45.19 / 43.57 / **49.08**	50.77 / 33.99 / 47.55 / **57.20**

- **Takeaway**: AdvFlow is the most effective approach among well-known attacks against defended DNNs!

Contact Information & References

Twitter hmdolatabadi
Website hmdolatabadi.github.io
Repo. github.com/hmdolatabadi/AdvFlow

[1] Wierstra et al. Natural evolution strategies. *JMLR*, 2014.

[2] Rezende & Mohamed. Variational inference with normalizing flows. In *ICML*, 2015.

[3] Li et al. NATTACK: learning the distributions of adversarial examples for an improved black-box attack on deep neural networks. In *ICML*, 2019.