Adversarial Robustness of Flow-Based Generative Models

Phillip Pope[*] **Yogesh Balaji**[*] **Soheil Feizi**
University of Maryland, College Park

Abstract

Flow-based generative models leverage invertible generator functions to fit a distribution to the training data using maximum likelihood. Despite their use in several application domains, robustness of these models to adversarial attacks has hardly been explored. In this paper, we study adversarial robustness of flow-based generative models both theoretically (for some simple models) and empirically (for more complex ones). First, we consider a linear flow-based generative model and compute optimal sample-specific and universal adversarial perturbations that maximally decrease the likelihood scores. Using this result, we study the robustness of the well-known *adversarial training* procedure, where we characterize the fundamental trade-off between model robustness and accuracy. Next, we empirically study the robustness of two prominent deep, non-linear, flow-based generative models, namely GLOW and RealNVP. We design two types of adversarial attacks; one that minimizes the likelihood scores of in-distribution samples, while the other that maximizes the likelihood scores of out-of-distribution ones. We find that GLOW and RealNVP are extremely sensitive to both types of attacks. Finally, using a hybrid adversarial training procedure, we significantly boost the robustness of these generative models.

1 Introduction

The promise of modern deep generative models is to learn data distributions with sufficiently high fidelity, allowing simulation of realistic samples. Some applications include photo-realistic image generation, audio synthesis, and image to text generation (Reed et al., 2016; Ledig et al., 2017; van den Oord et al., 2016a). Generative Adversarial Networks (GANs)(Goodfellow et al., 2014) have become a popular choice in modern generative modeling, often obtaining the state-of-the-art results in image and video synthesis (Karras et al., 2018). While GANs can synthesize samples from a data distribution, their inability to compute sample likelihoods limits their usage in statistical inference tasks (Balaji et al., 2019).

Likelihood-based models, on the other hand, explicitly fit a generative model to the data using a maximum likelihood optimization, enabling exact or approximate evaluations of sample likelihoods at the test time. Some popular choices include auto-regressive models (van den Oord et al., 2016b), (Oord et al., 2016), Variational Auto-encoders (Kingma & Welling, 2019), and methods based on normalizing flow (Rezende & Mohamed, 2015). Notably, flow-based models (Kingma & Dhariwal, 2018; Dinh et al., 2017) leverage invertible generator functions to learn a bijective mapping between latent space and the data distribution, enabling an exact sample likelihood computation.

The focus of this paper is to perform a comprehensive study of robustness of likelihood-based generative models to adversarial perturbations of their inputs. While there has been progress on adversarial robustness of classification problems (Madry et al., 2018), robustness of likelihood models has not been explored in the literature. Performing such a sensitivity analysis is crucial for reliable deployment, especially in safety-critical applications. For instance, one application where likelihood estimation is crucial is unsupervised anomaly detection in medical imaging, where out-of-distribution samples can be detected using likelihood scores. Adversarial attacks on such systems can lead to false diagnosis, potentially bearing life-threatening consequences.

First, we present a theoretical analysis of the sensitivity of linear generative models that fit a Gaussian

Proceedings of the 23$^{\text{rd}}$International Conference on Artificial Intelligence and Statistics (AISTATS) 2020, Palermo, Italy. PMLR: Volume 108. Copyright 2020 by the author(s).

[*] First two authors contributed equally

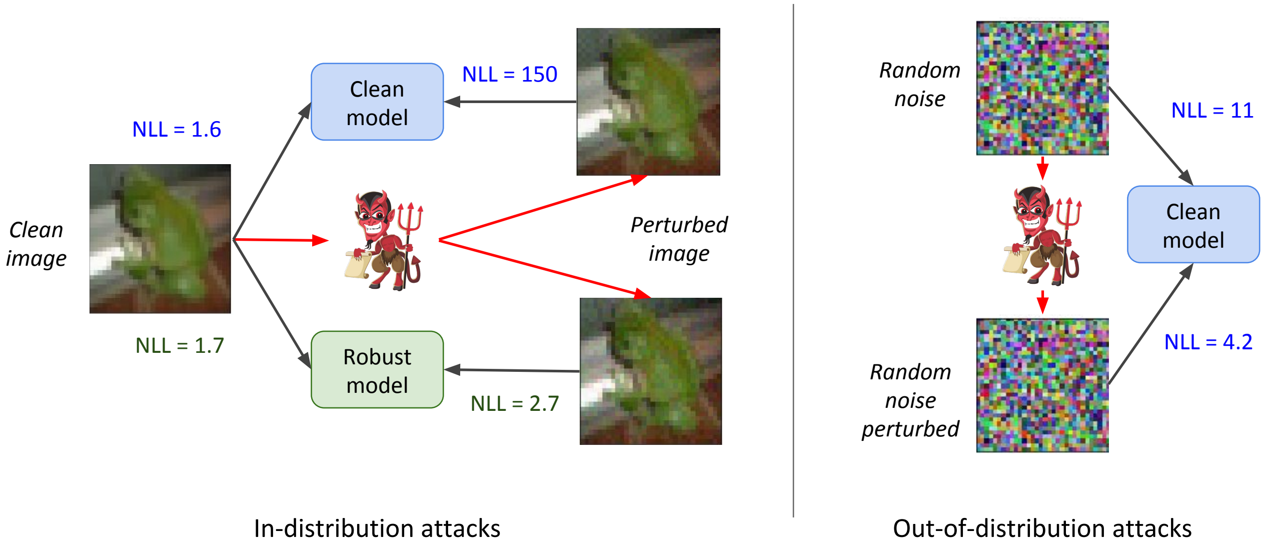

Figure 1: Sensitivity of flow-based generative models to adversarial attacks. A low NLL indicates a high likelihood score. The figure on the left panel shows *in-distribution* attacks: a frog image which is assigned a NLL score of 1.6 by GLOW model trained on CIFAR-10 (clean model) gives a high NLL of 150 when perturbed adversarially ($\epsilon = 8$ in ℓ_∞ norm). Our robust model significantly improves the robustness: NLL does not change much after the attack, while the score on the unperturbed sample is similar to the one obtained by the clean model. The panel on the right shows *out-of-distribution* attacks where a noise image is assigned a low NLL of 4.2.

distribution to the data (since the latent variable is often a Gaussian distribution itself). Under this setting, we compute the optimal sample-specific and universal norm-bounded input perturbations to maximally decrease the likelihood scores. We then analyze the effectiveness of one of the most successful defense mechanism against adversarial attacks, namely *adversarial training* (Madry et al., 2018) where adversarially perturbed samples are recursively used in re-training the model. We show that adversarial training can provably defend against norm-bounded adversarial attacks. However, this comes at a cost of decrease in clean likelihood scores. This naturally gives rise to a fundamental trade-off between model performance and robustness.

Next, we empirically show the existence of adversarial attacks on two popular deep flow-based generative models: GLOW (Kingma & Dhariwal, 2018) and RealNVP (Dinh et al., 2017). A measure of likelihood, by definition, should assign low scores to out-of-distribution samples and high scores to in-distribution ones. The existence of adversarial attacks breaks and contradicts this intuition: We show that we can construct samples that look like normal (in-distribution) data to a human eye, yet the model assigns to them low likelihood scores, or equivalently, high negative log likelihood (NLL) scores. Similarly, we show the existence of out-of-distribution samples that are assigned low NLL scores. One such example is shown in Figure 1, where a sample from CIFAR-10 dataset when adversarially perturbed has high NLL score, and a random adversarially perturbed image (with uniform pixel intensities) has low NLL score. This observation raises serious doubts about the reliability of likelihood scores obtained through standard flow-based models.

To make these models robust, we investigate the effect of the popular *adversarial training* mechanism. We show that adversarial training empirically improves robustness, however, this comes at a cost of decrease in likelihood scores on unperturbed test samples compared to the baseline model. To mitigate this effect, we propose a novel variant of adversarial training, called the *hybrid adversarial training*, where the negative log likelihoods of both natural and perturbed samples are minimized during training. We show that hybrid adversarial training obtains increased robustness on adversarial examples, while simultaneously maintaining high likelihood scores on clean test samples.

In summary, our contributions are as follows:

- We theoretically analyze the robustness of linear generative models, and show that adversarial training provably learns robust models. We also characterize the fundamental trade-off between model robustness and performance.
- We demonstrate the existence of *in-distribution* and *out-of-distribution* attacks on flow-based like-

lihood models.

- We propose a novel variant of adversarial training, called the *hybrid adversarial training*, that can learn robust flow-based models while maintaining high likelihood scores on unperturbed samples.

2 Background

2.1 Flow-based Generative models

Although generative modeling is largely dominated by generative adversarial networks (GANs), one major short-coming of GANs is its inability to compute sample likelihoods. *Flow-based* generative models solve this issue by designing an invertible transformation $f : \mathbb{R}^D \to \mathbb{R}^D$ between latent distribution $p_{\mathbf{z}}(\mathbf{z})$ and the generated distribution $p_{\mathbf{x}}(\mathbf{x})$. $p_{\mathbf{z}}(\mathbf{z})$ is often assumed to be a normal distribution. Given a random variable $\mathbf{z} \sim p_{\mathbf{z}}(\mathbf{z})$, we can use *change of variables* to write the log density of a sample $\mathbf{x}$ such that $\mathbf{x} = f(\mathbf{z})$ as (Dinh et al., 2015, 2017; Grathwohl et al., 2019; Kingma & Dhariwal, 2018):

$$\log p_{\mathbf{x}}(\mathbf{x}) = \log p_{\mathbf{z}}(\mathbf{z}) - \log \det \left| \frac{\partial f(\mathbf{z})}{\partial \mathbf{z}} \right|$$

Since f is invertible, inference can be performed as $\mathbf{z} = f^{-1}(\mathbf{x})$. The transformation f is typically modeled as the composition of K invertible maps, $f = f_1 \circ f_2 \circ \cdots \circ f_K$, also called *normalizing flows* (Rezende & Mohamed, 2015). The special structure used in each f_i's allows an efficient computation of the determinant.

Prominent examples of flow-based generative models include NICE (Dinh et al., 2015), RealNVP (Dinh et al., 2017), GLOW (Kingma & Dhariwal, 2018), and FFJORD (Grathwohl et al., 2019), each using a particular choice of f_k . For instance, RealNVP (Dinh et al., 2017) is designed with *affine coupling layers*, essentially an invertible scale transformation, while GLOW uses *invertible* 1×1 *convolutions* (Kingma & Dhariwal, 2018), which utilizes learned permutations.

2.2 Adversarial Attacks and Robustness

In context of classification, adversarial examples are subtle input perturbations that changes a model prediction. These perturbations are small in the sense of a suitable norm and imperceptible to humans. The existence of such examples raises serious concern for the deployment of machine learning models in safety-critical applications

Let $\mathcal{D} = \{(\mathbf{x}, y)\}_{i=1}^N$ be a collection of labeled input instances, θ be the parameters of a classifier with loss function L_{cls} (e.g. cross-entropy). For a given $\mathbf{x}$, let S be the set of all ℓ_p norm-bounded perturbations around $\mathbf{x}$, i.e., $\|\delta\|_p < \epsilon$, where ϵ is a constant, also called the perturbation radius. The perturbed sample is then given by $\mathbf{x}^{adv} = \mathbf{x} + \delta$. Standard methods of crafting adversarial examples for classification include the Fast Gradient Sign Method (FGSM) (Goodfellow et al., 2015)

$$\mathbf{x}^{adv} = \mathbf{x} + \alpha \operatorname{sign}(\nabla_{\mathbf{x}} L_{cls}(\theta, \mathbf{x}, y))$$

and its variant Projected Gradient Descent (PGD) (Kurakin et al., 2017)

$$\mathbf{x}^{(t+1)} = \operatorname{Proj}_{\mathbf{x}+S} \left(\mathbf{x}^{(t)} + \alpha \operatorname{sign}(\nabla_{\mathbf{x}^{(t)}} L_{cls}(\theta, \mathbf{x}^{(t)}, y)) \right)$$
$$\mathbf{x}^{adv} = \mathbf{x}^{(m)}$$

which is essentially a recursive application of FGSM for m steps, while constraining the perturbation to stay within the feasible region in each step. The above attacks have been shown effective on a variety of datasets (Carlini & Wagner, 2017; Xie et al., 2019). For defense against such attacks, Madry et al. (2018) proposed *adversarial training*, a procedure where the parameters θ of the model is optimized using the following minimax objective

$$\min_{\theta} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} \left[\max_{\delta \in S} L_{cls}(\theta, \mathbf{x} + \delta, y) \right]$$

Intuitively, this amounts to training the model on adversarial examples instead of the unperturbed ones. Adversarial training remains to be one of the successful defense mechanisms to date.

2.3 Robustness of generative models

To the best of our knowledge, no prior work exists on adversarial robustness of likelihood-based generative models. In Kos et al. (2018), attacks on encoder-based generative models (VAEs and VAE-GANs) are constructed to adversarially manipulate reconstruction and latent-space tasks. Adversarial attacks on generative classifiers are explored in Fetaya et al. (2019), where they show that even near optimal conditional generative models are susceptible to adversarial attacks when used for classfication tasks. Nalisnick et al. (2019) discuss some non-intuitive properties of flow-based generative models, studying GLOW in particular. They empirically observe that on a variety of common datasets, GLOW models assign *lower* likelihoods to in-distribution data than out-of-distribution. Diakonikolas et al. (2018) study robust learning of high dimensional Gaussians under Huber's strong ϵ-contamination model, in which adversary decides what outliers to place after observing the inlier distribution. Our work differs in that it considers test time attacks, where every sample is perturbed under norm bounded attacks at test-time to maximally decrease likelihood.

3 Adversarial robustness of linear flow-based generative models

We begin with an analysis of adversarial robustness of linear flow-based generative models. Since the latent variable Z has a normal distribution and the transformation between Z and X is considered to be affine, X will have a distribution in the form of $X \sim \mathcal{N}(\mu, K)$ where μ is the mean vector and K is the covariance matrix. Given N samples as the input dataset $D = \{\mathbf{x}_i\}_{i=1}^N$, we are interested in estimating the mean and covariance matrices of the generative distribution. This problem can be solved using maximum-likelihood estimation, in which we find model parameters that maximize the likelihood of the input dataset D. We know that log-likelihood of a test point $\mathbf{x}$ under the Gaussian distribution X can be written as

$$L(\mathbf{x}) = C - \frac{1}{2}\log(|K|) - \frac{(\mathbf{x}-\mu)^T K^{-1}(\mathbf{x}-\mu)}{2}$$

where $C = n\log(2\pi)/2$. It is a well-known result that maximizing the log-likelihood of the dataset D results in the following estimators for mean and covariance:

$$\hat{\mu} = \frac{\sum_i \mathbf{x}_i}{N}$$
$$\hat{K} = \frac{1}{N}\sum_{i=1}^{N}\frac{(\mathbf{x}_i-\hat{\mu})(\mathbf{x}_i-\hat{\mu})^T}{N},$$

referred to as the sample mean and sample covariance, respectively.

3.1 Adversarial attacks

In this section, we aim to find a norm-bounded adversarial perturbation δ that maximally decreases the likelihood score of a test point $\mathbf{x}$. We consider adversarial perturbations with a bounded ℓ_2 norm (i.e., $\|\delta\|_2 < \epsilon$). Please note that while we use ℓ_2 norm here, ℓ_∞ perturbation norm bounds are used to construct attacks for non-linear flow-models trained on vision datasets (Section. 5). We would like to point out that both ℓ_2 and ℓ_∞ are commonly used settings to study adversarial robustness (Madry et al., 2018). The perturbation δ can be found by solving the following optimization problem

$$\min_{\delta} \quad C - \frac{1}{2}\log(|K|) - \frac{(\mathbf{x}-\mu+\delta)^T K^{-1}(\mathbf{x}-\mu+\delta)}{2} \tag{1}$$
$$\text{s.t.} \;\; \|\delta\|_2 < \epsilon$$

Theorem 3.1 *Let $L(\mathbf{x})$ denote the likelihood function of an input sample $\mathbf{x}$ under a Gaussian distribution $\mathcal{N}(\mu, K)$. Let $K = U\Lambda U^T$ be the eigen-decomposition of the covariance matrix K. Let $c = [c_1, c_2, \ldots, c_n] = U^T K$, and $\Lambda = diag([\lambda_1, \ldots \lambda_n])$. Let η be a solution of the set of equations*

$$\sum_i \frac{c_i^2}{(1-2\eta\lambda_i)^2} = \epsilon^2$$
$$2\eta\lambda_i - 1 \geq 0 \quad \forall i$$

Then, the optimal additive perturbation δ with norm bound $\|\delta\|_2 < \epsilon$ that maximally decreases the likelihood score of sample $\mathbf{x}$ is given by

$$\delta^* = (K^{-1} - 2\eta I)^{-1} K^{-T}(\mu - \mathbf{x}) \tag{2}$$

The proof of Theorem 3.1 is given in supplementary material. The above theorem gives a solution for the optimal adversarial perturbation with a bounded ℓ_2 norm for the linear flow-based generative models (solution to Eq. (1)). Example optimal adversarial perturbations calculated for a 2-dimensional Gaussian distribution are visualized in Figure 2. Next, we show two special cases of this result.

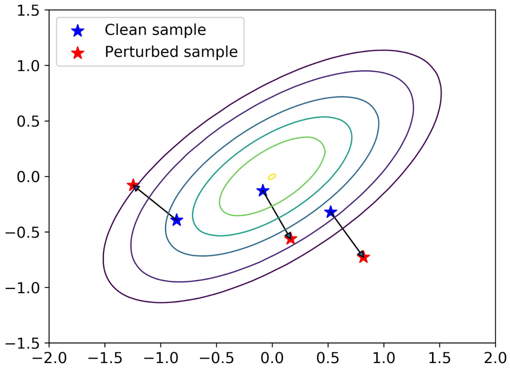

Figure 2: Examples of optimal adversarial perturbations with bounded ℓ_2 norm for a 2-dimensional Gaussian case.

3.1.1 Special case: Spherical covariance matrix

In this case, the covariance matrix is $K = \sigma^2 I$. In this setup, the solution for the adversarial perturbation problem (2) simplifies to:

$$\delta = \frac{1}{1-2\eta\lambda}(\mu - \mathbf{x})$$

Similarly, the condition for η simplifies to:

$$1 - 2\eta\lambda = \frac{\|\mu - \mathbf{x}\|}{\epsilon}$$

Thus, the optimal adversarial perturbation δ in this case is given by

$$\delta^* = \frac{\epsilon}{\|\mathbf{x}-\mu\|}(\mathbf{x}-\mu) \tag{3}$$

3.1.2 Special case: $\mathbf{x} = \mu$

In this case, the optimization 1 simplifies to:

$$\begin{aligned} \min_{\delta} \quad & C - \frac{1}{2}\log(|K|) - \frac{\delta^T K^{-1}\delta}{2} \\ \text{s.t.} \quad & \|\delta\|_2 < \epsilon \end{aligned}$$

This is a Rayleigh quotient problem, the solution for which is the maximum eigenvalue of K^{-1}. I.e.,

$$\delta^* = \epsilon\, u_{min}(K)$$

where u_{min} is the eigenvector of K corresponding to the minimum eigenvalue. Intuitively, minimum eigenvector of the covariance matrix K is the direction in which the data varies the least, a perturbation along this direction induces a maximal drop in likelihood.

3.2 Defense against adversarial attacks

One of the most successful defense strategies against adversarial attacks is adversarial training (Section. 2.2), in which models are recursively trained on adversarially perturbed samples instead of clean ones. In this section, we analyze the effect of adversarial training for the likelihood estimation in the spherical Gaussian case. For an input sample $\mathbf{x}$ under the generative distribution $\mathcal{N}(\mu, \sigma^2 I)$, the adversarially perturbed sample using (3) is given by

$$\mathbf{x}^{adv} = \mathbf{x} + \frac{\epsilon}{\|\mathbf{x} - \mu\|}(\mathbf{x} - \mu)$$

We consider the population case where $\mathbf{x} \sim \mathcal{N}(\mu, \sigma^2 I)$. Denote $\tilde{\mathbf{x}} = (\mathbf{x} - \mu)/\sigma \sim \mathcal{N}(0, I)$. In this case, after one update of adversarial training, optimal model parameters will be:

$$\begin{aligned} \mu^{adv} &= \mathbb{E}[\mathbf{x}^{adv}] \\ &= \mu + \epsilon\mathbb{E}\Big[\frac{\tilde{\mathbf{x}}}{\|\tilde{\mathbf{x}}\|}\Big] = \mu \\ K^{adv} &= \mathbb{E}[(\mathbf{x}^{adv} - \mu)(\mathbf{x}^{adv} - \mu)^T] \\ &= \sigma^2\mathbb{E}\Big[\Big(\tilde{\mathbf{x}} + \epsilon\frac{\tilde{\mathbf{x}}}{\|\tilde{\mathbf{x}}\|}\Big)\Big(\tilde{\mathbf{x}} + \epsilon\frac{\tilde{\mathbf{x}}}{\|\tilde{\mathbf{x}}\|}\Big)^T\Big] \\ &= \sigma^2\Big(\mathbb{E}[\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T] + 2\epsilon\mathbb{E}\Big[\frac{\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T}{\|\tilde{\mathbf{x}}\|}\Big] + \epsilon^2\mathbb{E}\Big[\frac{\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T}{\|\tilde{\mathbf{x}}\|^2}\Big]\Big) \\ &= \sigma^2\Big(I + \frac{2\sqrt{2}\epsilon}{n}\frac{\Gamma((n+1)/2)}{\Gamma(n/2)}I + \frac{\epsilon^2}{n}I\Big) \\ &= \sigma^2 I + \sigma^2\alpha I = \sigma^2(1+\alpha)I \\ \textit{where } \alpha &= \frac{2\sqrt{2}\epsilon}{n}\frac{\Gamma((n+1)/2)}{\Gamma(n/2)} + \frac{\epsilon^2}{n} \end{aligned}$$

The above result follows from the fact that sum of diagonal terms of $\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T/\|\tilde{\mathbf{x}}\|$ has a Chi distribution with n degrees of freedom. We observe that adversarial training preserves the mean vector, but increases the variance by a multiplicative factor $1 + \alpha$.

Performing m steps of adversarial training results in the following estimate:

$$\begin{aligned} \mu_k^{adv} &= \mu \\ K_k^{adv} &= \sigma^2(1+\alpha)^m I \end{aligned} \tag{4}$$

Using the above argument, we obtain the following robustness guarantees for adversarial training:

Theorem 3.2 *Let $\mathbf{x}$ be an input sample drawn from $N(\mu, \sigma^2 I)$. Let $L(\mathbf{x})$ denote the log-likelihood function of the sample $\mathbf{x}$ estimated using an m-step adversarially trained model. Let δ be any perturbation vector such that $\|\delta\| \le \epsilon$. For any Δ, when*

$$\begin{aligned} m \ge \max\Big[&\log\Big(\frac{1}{2\sigma^2\Delta}\Big(2\sigma\epsilon\sqrt{20\log(1/\gamma)} + \epsilon^2\Big)\Big), \\ &\log\Big(\frac{1}{2\sigma^2\Delta}\Big[2\sigma\epsilon\sqrt{2n} + \epsilon^2\Big]\Big)\Big] / \log(1+\alpha) \end{aligned}$$

with probability greater than $1 - \gamma$,

$$L(\mathbf{x}) - L(\mathbf{x} + \delta) < \Delta$$

The proof for this theorem is presented in the appendix. This theorem states that, with high probability and for a sufficiently large m, m-step adversarial training learns a generative model whose likelihood estimates are provably robust within Δ.

3.2.1 Trade-off between robustness and accuracy

The estimated parameters of our linear model after m steps of adversarial training is given in Eq. (4). The average log-likelihood of unperturbed (clean) samples drawn from $\mathcal{N}(\mu, \sigma^2 I)$ under the adversarially-trained model can be computed as

$$\begin{aligned} L_{nat}(m) = &-\frac{n}{2}\log(2\pi\sigma^2(1+\alpha)^m) \\ &- \mathbb{E}_{\mathbf{x}\in\mathcal{N}(\mu,\sigma^2 I)}\frac{\|\mathbf{x}-\mu\|^2}{2\sigma^2(1+\alpha)^m} \\ = &-\frac{n}{2}\log(2\pi\sigma^2(1+\alpha)^m) - \frac{n}{2(1+\alpha)^m} \end{aligned}$$

The drop in the natural likelihood due to adversarial training, which we define as $L_{nat-dr} := L_{nat}(0) - L_{nat}(m)$, simplifies as

$$L_{nat-dr}(m) = \frac{n}{2}\Big[\log((1+\alpha)^m) + \frac{1}{(1+\alpha)^m} - 1\Big]$$

$L_{nat-dr}(m)$ represents how much the average log likelihood scores will be different if we use an m-step adversarially-trained model instead of the clean model. Larger m will lead to a larger drop in the accuracy of the likelihood computation. However, it will increase the robustness of likelihood scores against adversarial perturbations. To characterize this trade-off, note that the likelihood of perturbed samples under the m-step adversarially trained model can be computed as

$$\begin{aligned} L_{adv}(m) =& -\frac{n}{2}\log(2\pi\sigma^2(1+\alpha)^m) \\ &- \mathbb{E}_{\mathbf{x}\in\mathcal{N}(\mu,\sigma^2 I)} \frac{\|\mathbf{x} + \frac{\epsilon}{\|\mathbf{x}-\mu\|}(\mathbf{x}-\mu) - \mu\|^2}{2\sigma^2(1+\alpha)^m} \\ =& -\frac{n}{2}\log(2\pi\sigma^2(1+\alpha)^m) \\ &- \frac{n + 2\epsilon\sqrt{2}\frac{\Gamma((n+1)/2)}{\Gamma(n/2)} + \epsilon^2}{2(1+\alpha)^m} \end{aligned}$$

Hence, the adversarial sensitivity, which we define as $L_{sen}(m) = L_{clean}(m) - L_{adv}(m)$ simplifies to

$$L_{sen}(m) = \frac{2\epsilon\sqrt{2}\frac{\Gamma((n+1)/2)}{\Gamma(n/2)} + \epsilon^2}{2(1+\alpha)^m} \tag{5}$$

Adversarial sensitivity indicates the drop in likelihood scores due to adversarial attacks. Higher the score, more sensistive is the model to adversarial perturbations. We can see that L_{nat-dr} is at odds with L_{sen}. In Figure 3, we plot the trade-off between natural likelihood drop vs. the adversarial sensitivity for different values of m in the range $[0, 10]$. In this experiment, we use $n = 10$, and generate samples from a Gaussian distribution with a random mean and covariance matrix. We observe that the setting that gives low performance drop incurs high robustness drop, and vice-versa.

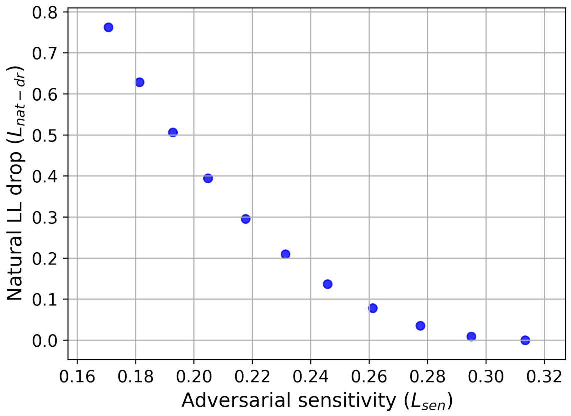

Figure 3: Plot showing the trade-off between performance and robustness for an example linear generative model.

3.3 Universal adversarial perturbation

In universal adversarial perturbation, we are interested in finding a single perturbation vector δ such that the population likelihood (under the normal distribution) decreases maximally, i.e., we are interested in finding a perturbation δ such that

$$\begin{aligned} \min_{\delta} \quad & C - \frac{1}{2}\log(|K|) \\ & - \mathbb{E}_{\mathbf{x}\sim\mathcal{N}(\mu,K)}\left[\frac{(\mathbf{x}-\mu+\delta)^T K^{-1}(\mathbf{x}-\mu+\delta)}{2}\right] \\ \text{s.t.} \quad & \delta^T\delta = \epsilon^2 \end{aligned}$$

Simplifying the objective, we obtain

$$\begin{aligned} \min_{\delta} \quad \mathbb{E}_{\mathbf{x}}\Big[& -\frac{(\mathbf{x}-\mu)^T K^{-1}(\mathbf{x}-\mu)}{2} \\ & + \delta^T K^{-1}(\mathbf{x}-\mu) + \frac{\delta^T K^{-1}\delta}{2}\Big] \end{aligned}$$

The first term is independent of δ, thus can be ignored. The second term is 0 since the mean of $\mathbf{x}$ is μ. Thus, the optimization simplifies as

$$\begin{aligned} & \max_{\delta}\frac{\delta^T K^{-1}\delta}{2} \\ & \text{s.t. } \delta^T\delta = \epsilon^2 \end{aligned}$$

This is the standard Rayleigh quotient problem, the solution for which is

$$\delta^* = \epsilon\lambda_{max}(K^{-1}) = \epsilon u_{min}(K)$$

Remark: The adversarial distribution for the universal case is $\mathbf{x}^{adv} = \mathbf{x} + \epsilon u_{min}(K)$. The perturbation in this case only shifts the mean of the perturbed distrbution, but the covariance remains the same. Hence, adversarial training results in the following estimation: $\hat{\mu}^{adv} = \mu + \epsilon u_{min}(K)$, and $K^{adv} = K$. Since only the mean gets shifted, the resulting adversarially trained model can again be attacked with the same perturbation, resulting the same sensitivity as the clean model. Hence, adversarial training is not successful to defend against universal adversarial attacks.

4 Adversarial attacks and defenses on non-linear flow-based models

In this section, we empirically study the robustness of deep flow-based generative models against adversarial attacks. First, we adapt the PGD attack to the flow-based models by replacing the classification loss L_{cls} with the log-likelihood function. For normal (in-distribution) samples, we seek to compute perturbations with a bounded ℓ_∞ norm such that the likelihood of the perturbed sample is decreased maximally.

This defines our *in-distribution* attack. We also use the crafted adversarial examples to recursively re-train the model to make it more robust against adversarial attacks, we call this *adversarial training*.

Next, we explore a new type of adversarial attack on the flow-based generative models where the goal of the adversary is to maximize the likelihood score of out-of-distribution (anomaly) samples to be similar to that of normal samples. We call this attack the *out-of-distribution adversarial attack*. This amounts to ascending (rather than descending) on the likelihood of out-of-distribution samples. We compare these attacks to random uniform noise, which is used as a baseline.

We empirically observe that adversarially trained models obtain higher NLL (i.e. lower likelihood) on clean (un-perturbed) data than models trained on clean data alone. This is expected as no clean samples were exposed at the training time. However, this is undesirable as a good generative model should assign high likelihoods to in-distribution samples. To mitigate this problem, we propose training simultaneously on both clean and adversarial examples. In each batch of training, we mix clean and adversarial samples in 1:1 ratio. We call this procedure the *hybrid adversarial training*. The analog of this method is known to fail for classification problems(Szegedy et al., 2014). However, it succeeds to robustify flow-based generative models, while preserving likelihood on unperturbed samples.

5 Experiments

We perform experiments on two flow-based generative models: GLOW and RealNVP, on three datasets: CIFAR-10, LSUN Bedroom, and CelebA. We evaluate the robustness of the three model varieties: models trained on clean (unperturbed) data alone, models trained on adversarially-perturbed data alone (adversarial training), and models trained on clean *and* adversarial data in 1:1 ratio (hybrid training). Adversarial and hybrid models were trained with $\epsilon = 8$ and $m = 10$ attack iterations. More experimental details can be found in supplementary material. In all experiments, we report negative log-likelihood values in the units of bits per dimension (Theis et al., 2016).

Figures 4 and 5 show visualizations of adversarial attacks on a GLOW model trained on unperturbed CIFAR-10 and LSUN-bedrooms datasets respectively. In the top row, we show in-distribution attacks at different attack strengths. We observe that in-distribution attacks are effective even at low ϵ values. The effectiveness of these attacks are evident as the values are much higher compared to the uniform noise baseline (shown in middle row). In the last row, we show out-of-distribution attacks, where a uniform noise image is perturbed to assign low NLL scores. For high attack strength ($\epsilon = 8$), likelihood values are on-par with values obtained by in-distribution samples.

ϵ	**Clean**	**Adv.**	**Hybrid**
0	**3.4**	4.7	3.6
1	6.3	4.9	**4.7**
2	14	**5.0**	**5.0**
4	320	**5.3**	**5.3**
8	2.0×10^6	**5.8**	5.9

Table 1: Robustness results of GLOW model trained on CIFAR-10

ϵ	**Clean**	**Adv.**	**Hybrid**
0	**2.4**	4.4	2.9
1	5.5	**4.5**	4.7
2	8.5	4.7	**4.6**
4	15.2	**5.0**	**5.0**
8	27.0	**5.5**	5.6
16	35.8	**6.6**	**6.6**
32	36.4	**7.7**	8.1

Table 2: Robustness results of GLOW model trained on LSUN-Bedrooms dataset

Attack Iterations	**CelebA**	**LSUN**
0	2.9	2.7
10	14.3	10.9
20	17.9	14.4
50	24.9	19.8
Uniform noise	5.8	4.8

Table 3: Adversarial attacks on RealNVP models trained on CelebA and LSUN Bedroom. All models were attacked with $\epsilon = 8$.

Next, we present quantitative results, where we report average sample likelihood scores (in bits/per dimension), averaged over the test set. Likelihood scores on adversarial samples over a sweep of attack strength (attack ϵ) for a GLOW model trained on CIFAR-10 and LSUN Bedroom datasets are shown in Tables 1 and 2 respectively. We observe that adversarially trained models improve robustness, however the NLL scores on unpertubed samples increase drastically. Hybrid adversarial training, on the other hand, achieves (1) NLL on adversarial samples comparable to adversarially trained model, and (2) likelihood on unperturbed samples comparable to clean baseline, i.e., hybrid model improves robustness preserving the performance on unperturbed samples, thus achieving the best of both worlds.

In Table 3, we report robustness results for RealNVP trained on CelebA and LSUN Bedroom datasets over

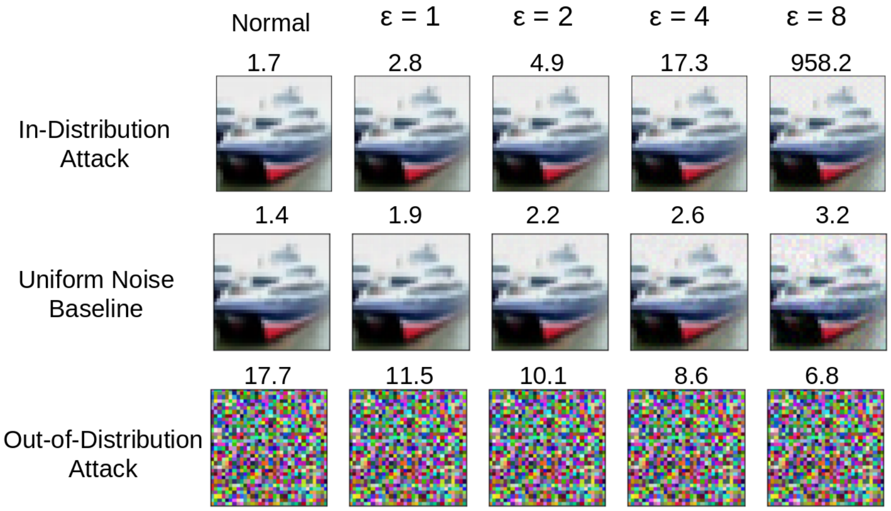

Figure 4: Sample visualizations of in-distribution, out-of-distribution and uniform noise attacks on a GLOW model trained on CIFAR-10 dataset. Results on clean (unperturbed) data are reported as $\epsilon = 0$.

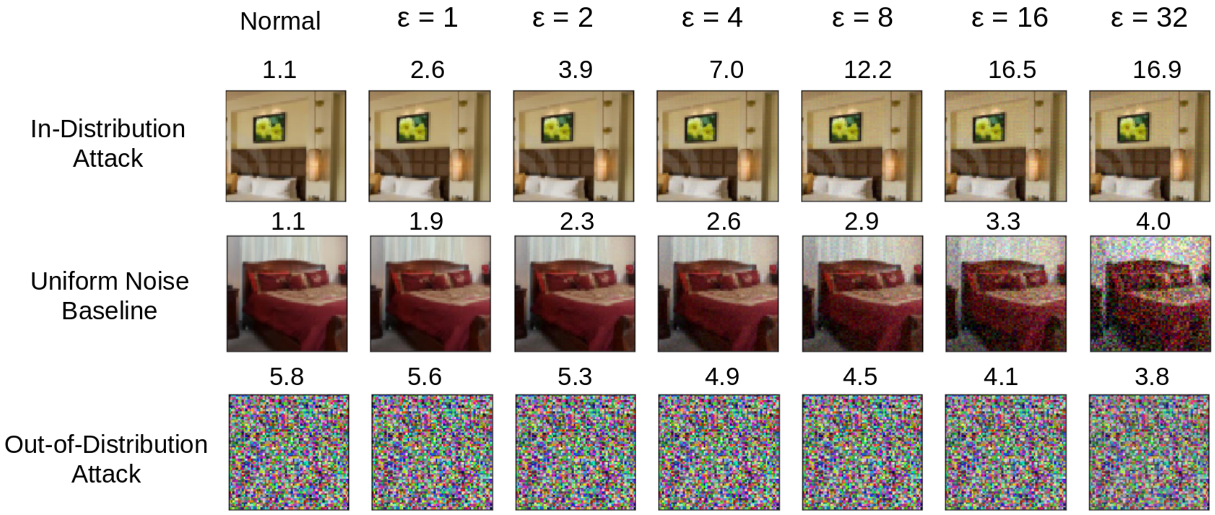

Figure 5: Sample visualizations of in-distribution, out-of-distribution and uniform noise attacks on a GLOW model trained on LSUN-Bedroom dataset. Results on clean (unperturbed) data are reported as $\epsilon = 0$.

a sweep of attack iterations. Due to computational constraints, we use a fixed $\epsilon = 8$. The results show that RealNVP is also susceptible to adversarial attacks, similar to GLOW models.

In the supplementary, we show further results including (1) further attack evaluations showing the distribution of obtained likelihoods (2) applying the linear Gaussian attack to the non-linear case, (3) evaluating the adversarial effects of generated samples from an adversarially trained model.

6 Conclusion

In this paper, we present a comprehensive analysis of adversarial robustness of flow-based generative models. First, we a perform a sensitivity analysis of linear generative models, and show that adversarial training provably improves robustness. Then, we demonstrate adversarial attacks on two non-linear flow-based generative models - GLOW and RealNVP. To improve the robustness of these models, we investigate the use of adversarial training, a popular defense mechanism used in classification. We show that adversarial training improves robustness at the cost of decrease in likelihood on unperturbed data. To remedy this issue, we propose *hybrid adversarial training*, a novel defense mechanism that improves adversarial robustness with a marginal drop in likelihood on unperturbed data.

7 Acknowledgements

This work was supported in part by NSF CAREER AWARD 1942230, HR 001119S0026, HR 00111990077, the Simons Fellowship on "Foundations of Deep Learning', a sponsorship from Capital One.

References

5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings, 2017. OpenReview.net. URL `https://openreview.net/group?id=ICLR.cc/2017/conference`.

6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings, 2018. OpenReview.net. URL `https://openreview.net/group?id=ICLR.cc/2018/Conference`.

7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019, 2019. OpenReview.net. URL `https://openreview.net/group?id=ICLR.cc/2019/Conference`.

Yogesh Balaji, Hamed Hassani, Rama Chellappa, and Soheil Feizi. Entropic GANs meet VAEs: A statistical approach to compute sample likelihoods in GANs. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 414–423, Long Beach, California, USA, 09–15 Jun 2019. PMLR. URL `http://proceedings.mlr.press/v97/balaji19a.html`.

N. Carlini and D. Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 39–57, May 2017. doi: 10.1109/SP.2017.49.

Ilias Diakonikolas, Gautam Kamath, Daniel M. Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robustly learning a gaussian: Getting optimal error, efficiently. In Artur Czumaj (ed.), *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018*, pp. 2683–2702. SIAM, 2018. doi: 10.1137/1.9781611975031.171. URL `https://doi.org/10.1137/1.9781611975031.171`.

Laurent Dinh, David Krueger, and Yoshua Bengio. NICE: non-linear independent components estimation. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Workshop Track Proceedings*, 2015. URL `http://arxiv.org/abs/1410.8516`.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings* DBL (2017). URL `https://openreview.net/forum?id=HkpbnH9lx`.

Ethan Fetaya, Jörn-Henrik Jacobsen, and Richard S. Zemel. Conditional generative models are not robust. *CoRR*, abs/1906.01171, 2019. URL `http://arxiv.org/abs/1906.01171`.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger (eds.), *Advances in Neural Information Processing Systems 27*, pp. 2672–2680. Curran Associates, Inc., 2014. URL `http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf`.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL `http://arxiv.org/abs/1412.6572`.

Will Grathwohl, Ricky T. Q. Chen, Jesse Bettencourt, Ilya Sutskever, and David Duvenaud. FFJORD: free-form continuous dynamics for scalable reversible generative models. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019* DBL (2019). URL `https://openreview.net/forum?id=rJxgknCcK7`.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings* DBL (2018). URL `https://openreview.net/forum?id=Hk99zCeAb`.

Diederik P. Kingma and Max Welling. An introduction to variational autoencoders. *CoRR*, abs/1906.02691, 2019. URL `http://arxiv.org/abs/1906.02691`.

Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 31*, pp. 10215–10224. Curran Associates, Inc., 2018.

J. Kos, I. Fischer, and D. Song. Adversarial examples for generative models. In *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 36–42, May 2018. doi: 10.1109/SPW.2018.00014.

Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial machine learning at scale. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings* DBL (2017). URL `https://openreview.net/forum?id=BJm4T4Kgx`.

Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings* DBL (2018). URL `https://openreview.net/forum?id=rJzIBfZAb`.

Eric T. Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Görür, and Balaji Lakshminarayanan. Do deep generative models know what they don't know? In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019* DBL (2019). URL `https://openreview.net/forum?id=H1xwNhCcYm`.

Aaron Van Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In Maria Florina Balcan and Kilian Q. Weinberger (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 1747–1756, New York, New York, USA, 20–22 Jun 2016. PMLR. URL `http://proceedings.mlr.press/v48/oord16.html`.

Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. In Maria Florina Balcan and Kilian Q. Weinberger (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 1060–1069, New York, New York, USA, 20–22 Jun 2016. PMLR. URL `http://proceedings.mlr.press/v48/reed16.html`.

Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1530–1538, Lille, France, 07–09 Jul 2015. PMLR. URL `http://proceedings.mlr.press/v37/rezende15.html`.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In Yoshua Bengio and Yann LeCun (eds.), *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`.

Lucas Theis, Aäron van den Oord, and Matthias Bethge. A note on the evaluation of generative models. In Yoshua Bengio and Yann LeCun (eds.), *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016. URL `http://arxiv.org/abs/1511.01844`.

Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W. Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *CoRR*, abs/1609.03499, 2016a. URL `http://arxiv.org/abs/1609.03499`.

Aaron van den Oord, Nal Kalchbrenner, Lasse Espeholt, koray kavukcuoglu, Oriol Vinyals, and Alex Graves. Conditional image generation with pixelcnn decoders. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 29*, pp. 4790–4798. Curran Associates, Inc., 2016b.

Cihang Xie, Yuxin Wu, Laurens van der Maaten, Alan L. Yuille, and Kaiming He. Feature denoising for improving adversarial robustness. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.