**Project Proposal: Creating Adversarial Examples Using Generative Modelling**

**Submitted:** Oct 12, 2017

**Prepared by:** Avishek (Joey) Bose 999738440, Kawin Ethayarajh 1000665167, Romina Abachi 998986628, Mohammad Firouzi 1004120131

**Motivation**

Given the application of deep neural networks to problems as varied as vehicle automation and cancer detection, one imperative is to better understand the circumstances in which these models ‘break’. To this end, *adversarial examples*, created by adding small perturbations to typical inputs,are crafted to fool the network into producing incorrect outputs. Understanding both attack and defense patterns for adversarial examples is an important direction of research, as it would allow vulnerabilities to be identified before being exploited by malicious attacks. To date, the most potent adversarial examples have been crafted through an optimization procedure, which is typically slower than approximating the single step gradient in a Fast Gradient Sign Attack. However there has been limited study on crafting adversarial examples through generative models such as Generative Adversarial Networks (GANs). Given the recent success of generative models in producing realistic data samples, we attempt to craft a new class of attacks by employing GANs in the creation of adversarial examples. Unlike previous GAN-based adversarial attacks, we generate examples in a continuous domain instead of simply adding features to distort typical inputs.

**Related Work**

General insights and work into adversarial phenomena is relevant to this work and are not be mentioned here. Specifically, we note that there are several types of existing attacks for crafting adversarial examples [1] such as Carlini/Wagner, L-BFGS, Fast Gradient Sign, JSMA, Deepfool. Some of the defenses proposed in literature include defensive distillation, adversarial retraining, and other methods described in [1].

We consider our work under the three different attack models [1]: a. *oblivious attack*: the attacker does not know there is an adversarial detection scheme protecting the model, b. *white-box attack*: the attacker knows that there is a detection scheme and knows its parameters, and c. *black-box attack*: the attacker knows that there is a detection scheme and how it was trained but does not have access to its parameters or training data.

Our main approach for generating adversarial examples is based on generative adversarial networks (GANs). Three papers closely related to that approach are [2], [3], and [4]. MalGAN [2] is a GAN-based model that generates adversarial malware programs to bypass a black-box malware detector. It consists of a generator neural network (NN) that considers what features to add to make an adversarial example and tries to fool the black-box detector and a detector network that tries to fit the black-box detector. Adversarial examples for malware programs mainly consist of adding features until the black-box detector classifies the adversarial examples as benign; it is therefore binary and not continuous. A similar approach to [1] for images is Adversarial Transformation Networks [3] and the networks in [4]. In these, a network is trained to generate images that maximize classification error based on pre-trained classifier networks and minimize the fidelity loss between real and adversarial images. Comparisons with gradient-based approaches, whereby an input is changed incrementally towards an adversarial classification, show that the NN-based approach generalizes better and has a lower true positive rate overall [3].

**Approach**

* We conduct a literature review of the current landscape of adversarial attacks, with a focus on the generative attacks that have been attempted (e.g., MalGAN). In general, these attacks work by adding features to typical inputs until they ‘break’ the detector network.
* We provide a justification for the design of our adversarial examples and use Layerwise Relevance Propagation to visualize them in input space. Unlike past approaches, our adversarial examples are generated in a continuous domain.
* In our approach we use a conditional GAN where the generator is conditioned on the unperturbed image to create an adversarial example while the discriminator/critic is used to distinguish adversarial from non-adversarial examples.
* We create our own GAN-based adversarial attack and report the robustness of several NNs trained on standard datasets (MNIST, CIFAR, etc.).
* We review the advantages and disadvantages of our approach compared to existing methods of adversarial attack, with respect to efficacy, transferability, and explainability.

**Extensions and Future Work**

Given enough time and success with our basic goals, we are also interested in the following research questions:

* Can we increase the adversarial robustness of NNs by training them with augmented adversarial examples from our GANs?
* Is there an intuitive explanation for why GAN-generated adversarial examples are more/less effective than those created using optimization or Fast Gradient Sign Attacks?
* How do adversarial examples generated by other generative models (e.g., Variational Autoencoders) fare in comparison to those generated by GANs?

**Disclosure**: None of us have prior research experience with adversarial attacks. Our research with our supervisors does not currently relate to this topic.

**References**

[1] Nicolas Carlini, David Wagner. 2017. Adversarial Examples are not easily detected: Bypassing Ten Detection Methods. <https://arxiv.org/pdf/1705.07263.pdf>

[2] Weiwei Hu, Ying Tan. 2017. Generating Adversarial Malware Examples for Black-Box Attacks. <https://arxiv.org/pdf/1702.05983.pdf>

[3] Shumeet Baluja, Ian Fischer. 2017. Adversarial Transformation Networks: Learning to Generate Adversarial Examples. <https://arxiv.org/pdf/1703.09387.pdf>

[4] Sayantan Sarkar, Ankan Bansal, Upal Mahbub, Rama Chellappa. 2017. UPSET and ANGRI: Breaking High Performance Image Classifiers. <https://arxiv.org/pdf/1707.01159.pdf>