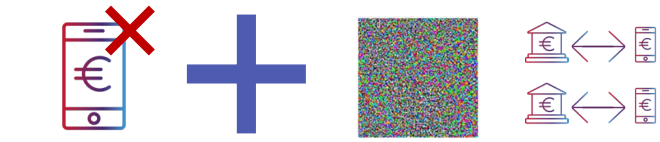
Constrained Adversarial Attacks and Defenses

T. Simonetto, S. Dyrmishi, S. Ghamizi, M. Cordy, Y. Le Traon
University of Luxembourg

uni.lu | SNT

Serval

Adversarial examples are subject to domain constraints



Objective: for x find δ

- ✓ With $H(x) \neq H(x+\delta)$
- ✓ With $L_p(x, x+\delta) < \epsilon$
- ✓ $x + \delta \in \mathcal{X}_\Omega$ (domain constraints)

Adversarial examples are inputs carefully designed to cause **erroneous predictions** in machine learning systems. To fool **real-world** systems, they must **respect domain constraints**.

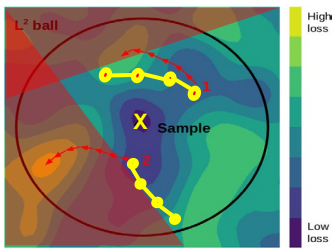
Constraints as penalty functions …

Example of constraint:

$$installment = loam\_amount \times \frac{int\_rate \times (1 + int\_rate)^{term}}{(1 + int\_rate)^{term} - 1}$$

Constraints formulae	Penalty function
$\omega_1 \wedge \omega_2$	$\omega_1 + \omega_2$
$\omega_1 \vee \omega_2$	$\min(\omega_1, \omega_2)$
$\psi \in \Psi = \{\psi_1, \dots \psi_k\}$	$\min(\{\psi_i \in \Psi :\mid \psi - \psi_i \mid\})$
$\psi_1 \leq \psi_2$	$max(0, \psi_1 - \psi_2)$
$\psi_1 < \psi_2$	$max(0, \psi_1 - \psi_2 + \tau)$
$\psi_1 = \psi_2$	$\mid \psi_1 - \psi_2 \mid$

… used in 2 constrained attacks



C-PGD (gradient based)

$$\nabla_{x^t} loss(h(x^t), y)$$
$$-\nabla_{x^t} penalty(x^t)$$

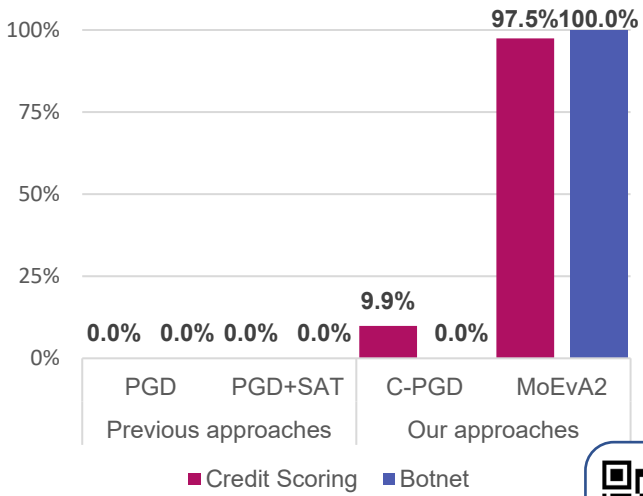
MoEvA2 (genetic based)

$$g_1(x) \equiv h(x)$$

$$g_2(x) \equiv L_p(x - x_0)$$

$$g_3(x) \equiv \sum_{\omega_i \in \Omega} penalty(x, \omega_i)$$

Attacks success rate



Success = Misclassification & Constraints Satisfaction

Defenses

Constraints augmentation:

$$f_e = f_1 \oplus f_2 \Rightarrow \omega_e \vDash (f_e = f_1 \oplus f_2)$$

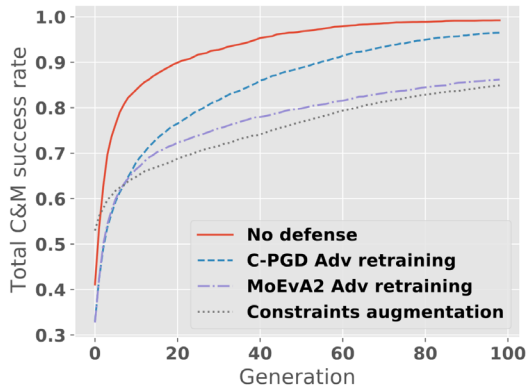


Figure 1: Success rate of MoEvA2 against the original model and the three defended models, over the generations.

More information