Verification of Non-Linear Specifications for Neural Networks

Hanqing Chao

VERIFICATION OF NON-LINEAR SPECIFICATIONS FOR NEURAL NETWORKS

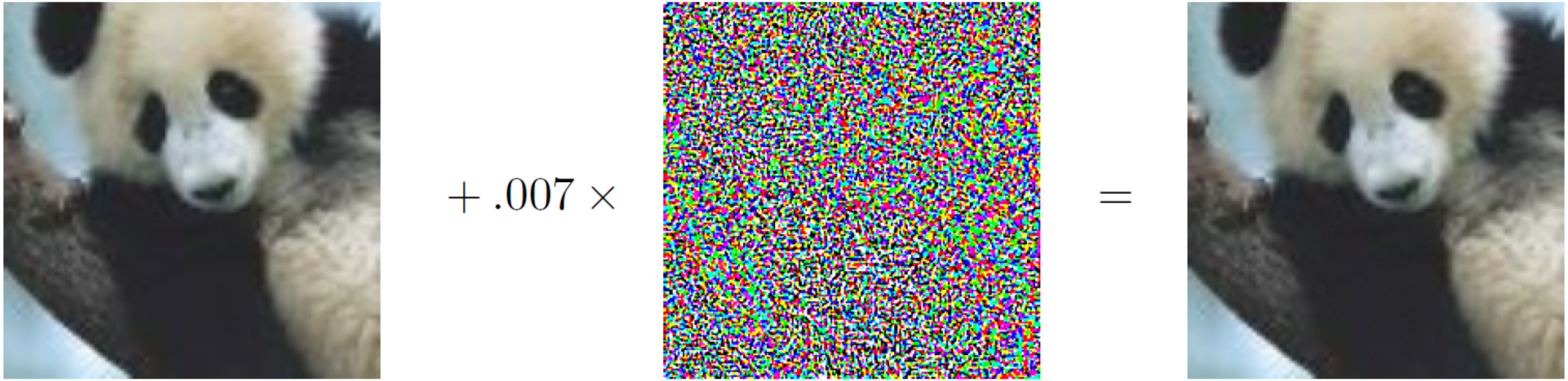
Chongli Qin*, Krishnamurthy (Dj) Dvijotham*, Brendan O'Donoghue, Rudy Bunel, Robert Stanforth, Sven Gowal, Jonathan Uesato, Grzegorz Swirszcz, Pushmeet Kohli

DeepMind
London, N1C 4AG, UK
correspondence: chongliqin@google.com

Hanqing Chao

Adversarial Attack

- Loopholes exists on a well trained network
- The output cannot hold in some very proximity of the training examples



"panda" 57.7% confidence | "nematode" 8.2% confidence | "gibbon" 99.3 % confidence

Specification Verification

- Specification:

Some properties we expect a train network to have:

The output could hold in a very proximity of the training examples

$y = f(x + r)$ are consistent for $\forall \, ||r|| < \epsilon, \epsilon > 0$

Specification Verification

- Specification: $y = f(x + r)$ are consistent for $\forall \, ||r|| < \epsilon, \epsilon > 0$
- ϵ is some kind measurement demonstrating the specification of the network (on point x).

Complete verification:

Guaranteed to either find a proof that the specification is true

Or find a counterexample proving that the specification is untrue.

$\epsilon = 0.5$, True specification

$\epsilon = 0.3$, Lower bound

Incomplete verification:

May not find a proof even if the specification is true.

However, if they do a find a proof, the specification is guaranteed to be true.

Specification Verification

- Specification: $y = f(x + r)$ are consistent for $\forall \; ||r|| < \epsilon, \epsilon > 0$
- ϵ is some kind measurement demonstrating the specification of the network (on point x).

Complete verification:

Guaranteed to either find a proof that the specification is true

Or find a counterexample proving that the specification is untrue.

$\epsilon = 0.5$, True specification

$\epsilon = 0.3$, Lower bound

Incomplete verification:

May not find a proof even if the specification is true.

However, if they do a find a proof, the specification is guaranteed to be true.

Problem Formulation

- A network mapping: $f : \mathcal{X} \to \mathcal{Y}$,
- With specification $F : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$
- F is satisfied if

$$F(x, y) \leq 0 \quad \forall x \in \mathcal{S}_{\text{in}}, \; y = f(x).$$

- Where $\mathcal{S}_{in}$ is the neighbor region of x^{nom}, for instance:

$$\mathcal{S}_{\text{in}} = \{x : \| x - x^{\text{nom}} \|_{\infty} \leq \delta\},$$

Non-Linear Specification

$$F(x, y) \leq 0 \quad \forall x \in \mathcal{S}_{\text{in}},\ y = f(x).$$

- Output consistency: linear specification

$$F(x, y) = c^T y + d \leq 0 \quad \forall x \in \mathcal{S}_{\text{in}}, y = f(x)$$

- Non-linear specification:
  - In a learned dynamics model of a physical system, law of conservation of energy should be met;
  - For a classifier, its output labels under adversarial perturbations (adversarial attack) should be semantically consistent;
  - In a system predicts the summation of handwritten digits, errors should be bounded.

Conservation of Energy

$$F(x, y) \leq 0 \quad \forall x \in \mathcal{S}_{\text{in}}, \; y = f(x).$$

- Consider a simple pendulum with damping

$$E(w, h, \omega) = \underbrace{mgh}_{\text{potential}} + \underbrace{\frac{1}{2} m l^2 \omega^2}_{\text{kinetic}},$$

- w and h represent the horizontal and vertical coordinates of the pendulum respectively, and ω refers to the angular velocity

Conservation of Energy

$$F(x, y) \leq 0 \quad \forall x \in \mathcal{S}_{\text{in}},\ y = f(x).$$

- Consider a simple pendulum with damping

$$E(w, h, \omega) = \underbrace{mgh}_{\text{potential}} + \underbrace{\frac{1}{2} ml^2\omega^2}_{\text{kinetic}},$$

- w and h represent the horizontal and vertical coordinates of the pendulum respectively, and ω refers to the angular velocity

$$E(w', h', \omega') \leq E(w, h, \omega)$$

Semantic Consistency

$$F(x, y) \leq 0 \quad \forall x \in \mathcal{S}_{\text{in}}, \; y = f(x).$$

$$\mathbb{E}\left[d(i, j)\right] = \sum_j d(i, j) P(j|x)$$

- $d(i, j)$ predefined distance between two classes i, j; $d(i, i) = 0$
- $P(j|x)$ the probability that the classifier assigns

Semantic Consistency

$$F(x, y) \leq 0 \quad \forall x \in \mathcal{S}_{\text{in}},\ y = f(x).$$

$$\mathbb{E}\left[d(i,j)\right] = \sum_j d(i,j) P(j|x)$$

- $d(i, j)$ predefined distance between two classes i, j; $d(i, i) = 0$
- $P(j|x)$ the probability that the classifier assigns

$$\mathbb{E}\left[d(i,j)\right] \leq \epsilon.$$

Summation Error

$$F(x, y) \leq 0 \quad \forall x \in \mathcal{S}_{\text{in}}, \; y = f(x).$$

$$\mathbb{E}_j \left[\left| \sum_{n=1}^{N} (j_n - i_n) \right| \right] = \sum_{j \in J^N} \left| \sum_{n=1}^{N} (j_n - i_n) \right| \prod_{n=1}^{N} P\left(j_n | x_n\right),$$

- $\{x_n\}_{n=1}^{N}$ handwritten images and corresponding true transaction values $\{i_n\}_{n=1}^{N}$

Summation Error

$$F(x, y) \leq 0 \quad \forall x \in \mathcal{S}_{\text{in}}, \; y = f(x).$$

$$\mathbb{E}_j \left[\left| \sum_{n=1}^{N} (j_n - i_n) \right| \right] = \sum_{j \in J^N} \left| \sum_{n=1}^{N} (j_n - i_n) \right| \prod_{n=1}^{N} P\left(j_n | x_n\right),$$

- $\{x_n\}_{n=1}^{N}$ handwritten images and corresponding true transaction values $\{i_n\}_{n=1}^{N}$

$$\mathbb{E}_j \left[\left| \sum_{n=1}^{N} (j_n - i_n) \right| \right] \leq \epsilon.$$

Convex-relaxable Specification

Assumption 1. *We assume that* $\mathcal{S}_{\text{in}} \subseteq \mathbb{R}^n, \mathcal{S}_{\text{out}} \subseteq \mathbb{R}^m$ *are compact sets and that we have access to an efficiently computable*[1] *procedure that takes in* $F, \mathcal{S}_{\text{in}}, \mathcal{S}_{\text{out}}$ *and produces a compact convex set* $\mathcal{C}\left(F, \mathcal{S}_{\text{in}}, \mathcal{S}_{\text{out}}\right)$ *such that*

$$\mathcal{T}\left(F, \mathcal{S}_{\text{in}}, \mathcal{S}_{\text{out}}\right) := \{(x, y, z) : F(x, y) = z, x \in \mathcal{S}_{\text{in}}, y \in \mathcal{S}_{\text{out}}\} \subseteq \mathcal{C}\left(F, \mathcal{S}_{\text{in}}, \mathcal{S}_{\text{out}}\right). \quad (11)$$

When the above assumption holds we shall say that the specification

$$F(x, y) \leq 0 \quad \forall x \in \mathcal{S}_{\text{in}}, y = f(x) \quad (12)$$

is convex-relaxable.

Bounds propagation

- Given bounds of inputs, forward these bounds layer by layer to get the bounds of the last layer.

Convex Optimization

$$
\begin{aligned}
&\text{maximize } z \\
&\text{subject to } \left(x_0, x_K, z\right) \in \mathcal{C}\left(F, \mathcal{S}_{\text{in}}, \mathcal{S}_{\text{out}}\right) \\
&\qquad x_{k+1} \in \operatorname{Relax}\left(g_k\right)\left(W_k x_k + b_k, l_k, u_k\right), k = 0, \cdots, K-1 \\
&\qquad l_k \leq x_k \leq u_k, \quad k = 0, \cdots, K
\end{aligned}
$$

- Here, z is the value of F
- Main idea is to relax activation function and F to get boundaries.

Experiments

$\epsilon = 0.5$, True specification

$\epsilon = 0.3$, Lower bound

Incomplete verification:

May not find a proof even if the specification is true.

However, if they do a find a proof, the specification is guaranteed to be true.

How tight is it?

Experiments

Verification bound: This is the fraction of test examples x^{nom} for which the specification is provably satisfied over the set $\mathcal{S}_{\text{in}}(x^{\text{nom}}, \delta)$ using our verification algorithm.
Adversarial bound: This is the fraction of test examples x^{nom} for which the falsification algorithm based on (8) was *not able to find a counter-example in the set* $\mathcal{S}_{\text{in}}(x^{\text{nom}}, \delta)$.

$\epsilon = 0.7$, Adversarial bound

$\epsilon = 0.5$, True specification β

$\epsilon = 0.3$, Verification bound

$$\text{Verification bound} \leq \beta \leq \text{Adversarial bound}.$$

$$|\text{Verification bound} - \beta| \leq \text{Adversarial bound} - \text{Verification bound}.$$

Experiments

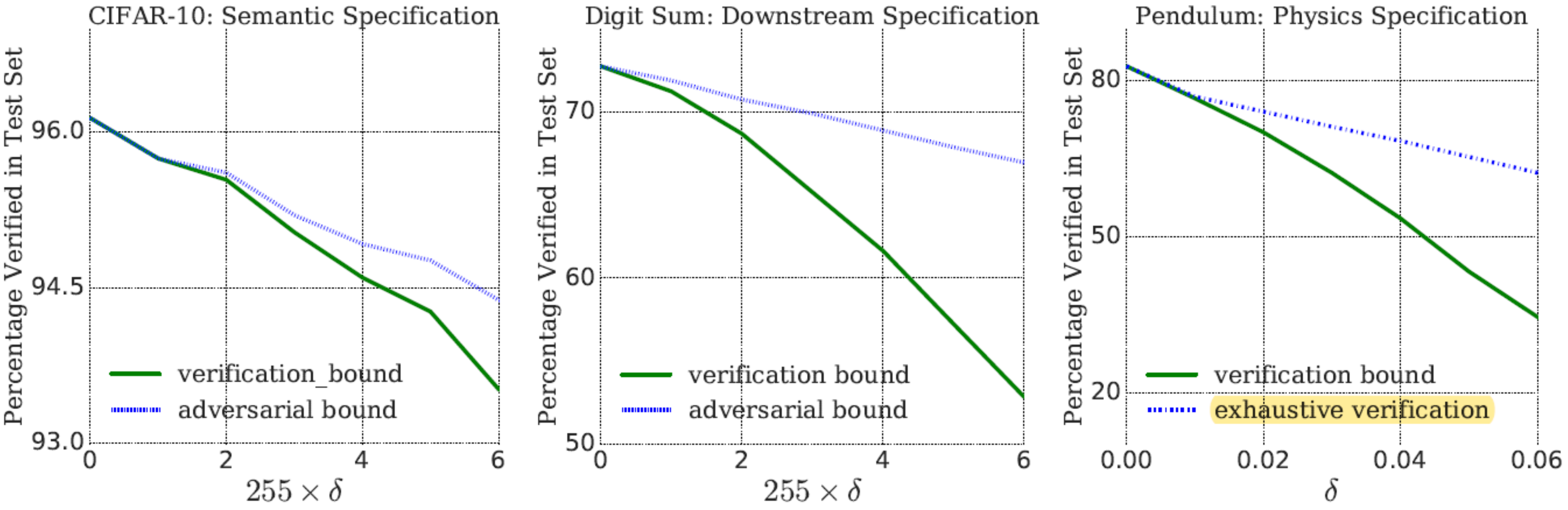
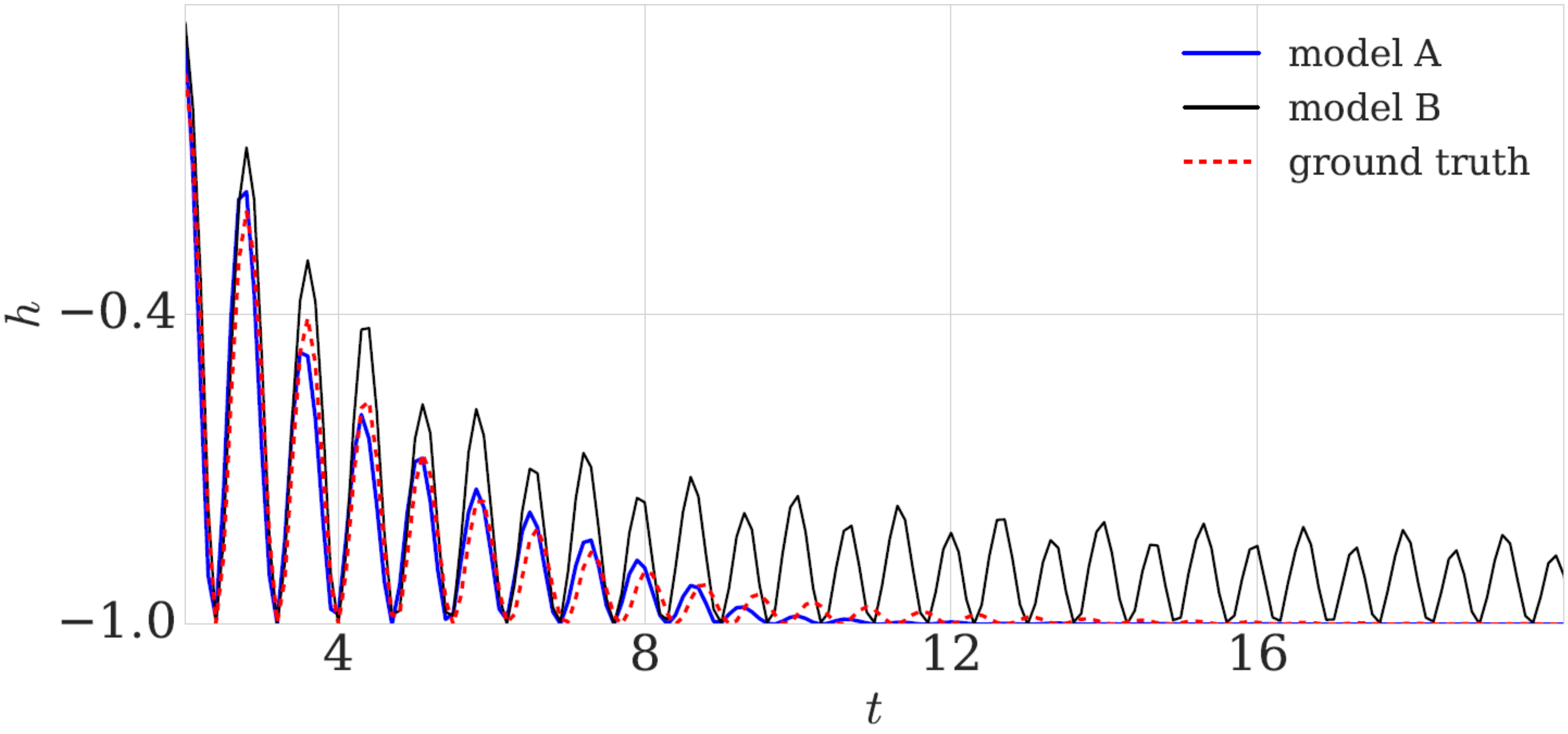


Figure 1: For three specifications, we plot the verification bound and adversarial bound as a function of perturbation size on the input.

Experiments

Are the specifications really satisfied in applications?

Conservation of Energy



Experiments

Are the specifications really satisfied in applications?

Semantic Consistency

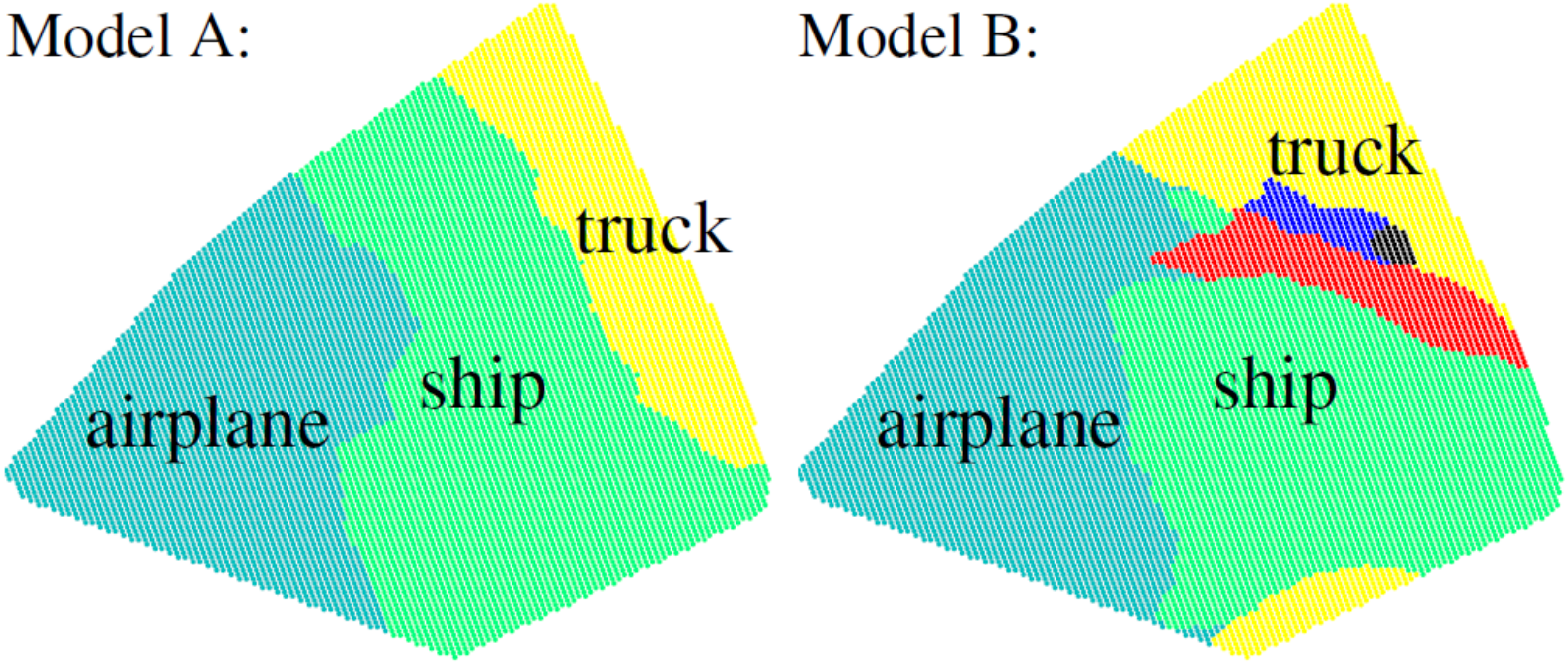


Figure 3: The projection of the decision boundaries onto a two dimensional surface formed by interpolating between three images belonging to the same semantic category (vehicles) - aeroplane (cyan), ship (green) and truck (yellow).The red/blue/**black** regions represent bird/cat/**frog** respectively).

Thanks！