# On the Perils of Cascading Robust Classifiers

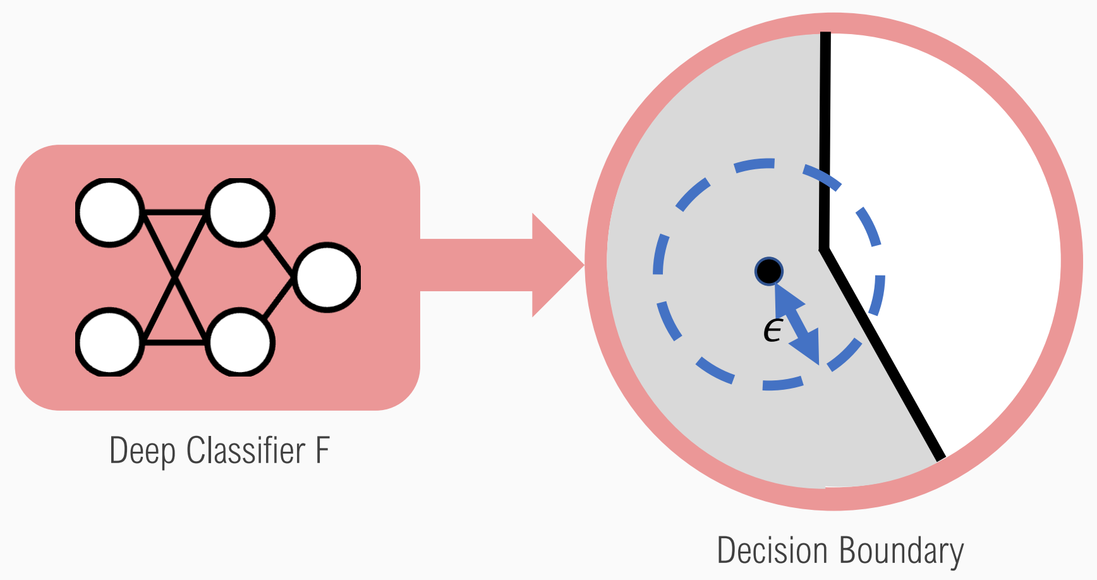
Ravi Mangal*, Zifan Wang*, Chi Zhang*, Klas Leino, Corina Pasareanu, Matt Fredrikson

Carnegie Mellon University

## Introduction

### Certifying *Local Robustness* for A Deep Classifier

$$\longrightarrow \forall x', ||x'-x|| \le \epsilon \rightarrow F(x) = F(x')$$



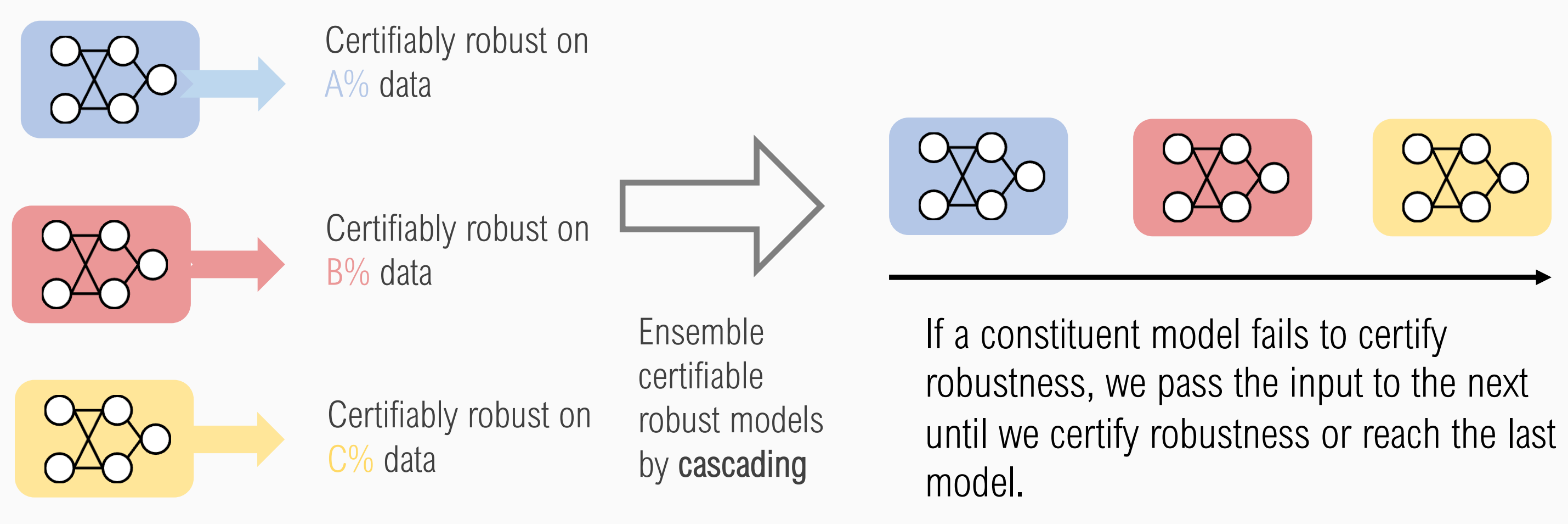
**Certification Soundness**

A certifier is sound if for any input

Certifier returns ROBUST → F is indeed locally robust

### One Strategy to Promote Certified Robust Is Using Cascading Ensemble

e.g. Wong et al. (2018) & Blum et al. (2022)


Certifiably robust on A% data

Certifiably robust on B% data

Certifiably robust on C% data

Ensemble certifiable robust models by **cascading**

If a constituent model fails to certify robustness, we pass the input to the next until we certify robustness or reach the last model.

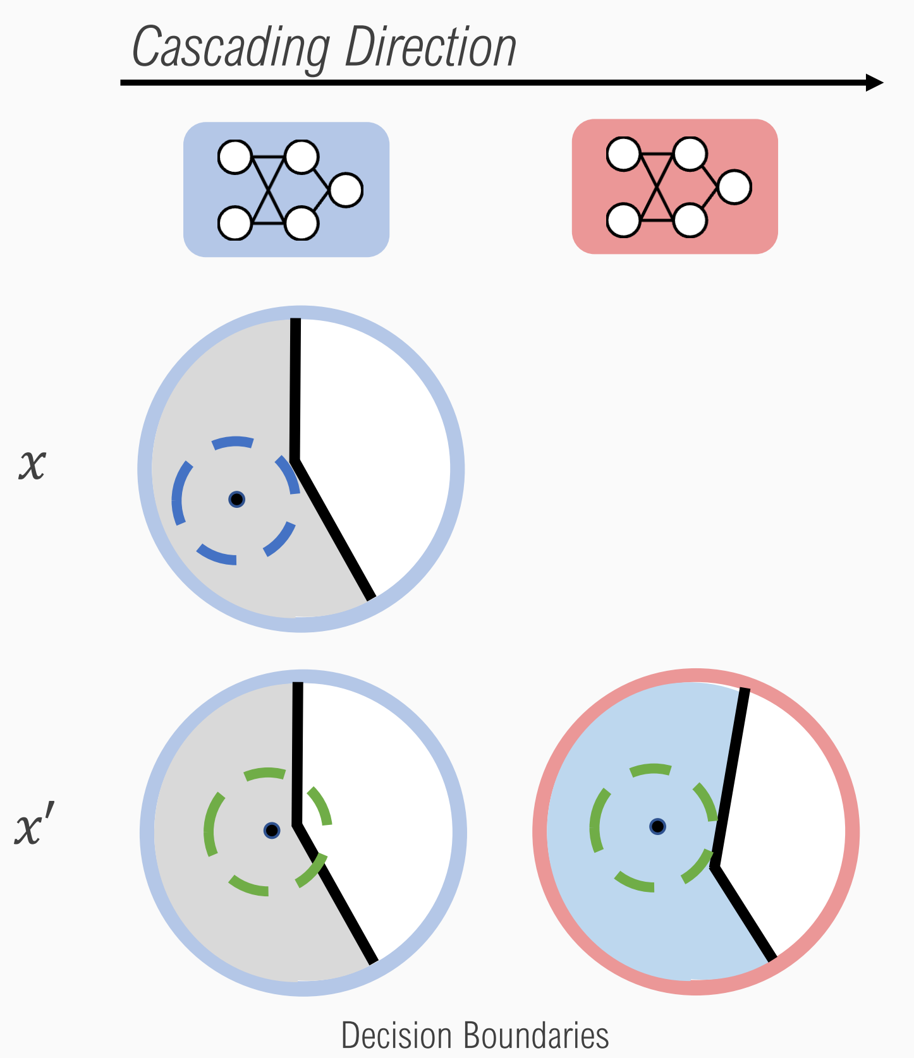
**Prior Works' Finding:** The cascade ensemble is found to certify a lot more % of points than each individual models.

## Contribution

- We prove that cascading certification is UNSOUND, even though each constituent model has a sound robust certifier.
- Our Cascade Attack (CasA) successfully finds points that are claimed to be certifiable ROBUST according to the cascading ensemble created by Wong et al. (2018) but are NOT locally robust in fact.
- Compared to cascading, ensemble by (weighted) voting is proved to be sound; however, gaining robustness improvement from voting requires model diversity to begin with.

## Cascading: An Unsound Certification

### Proof by Counter Example



*Output of Ensemble*

$$\exists x', ||x-x'|| \le \epsilon \not\Rightarrow \underset{y_{gray}}{Ensemble(x)} = \underset{y_{blue}}{Ensemble(x')}$$

*ROBUST as $y_{gray}$*

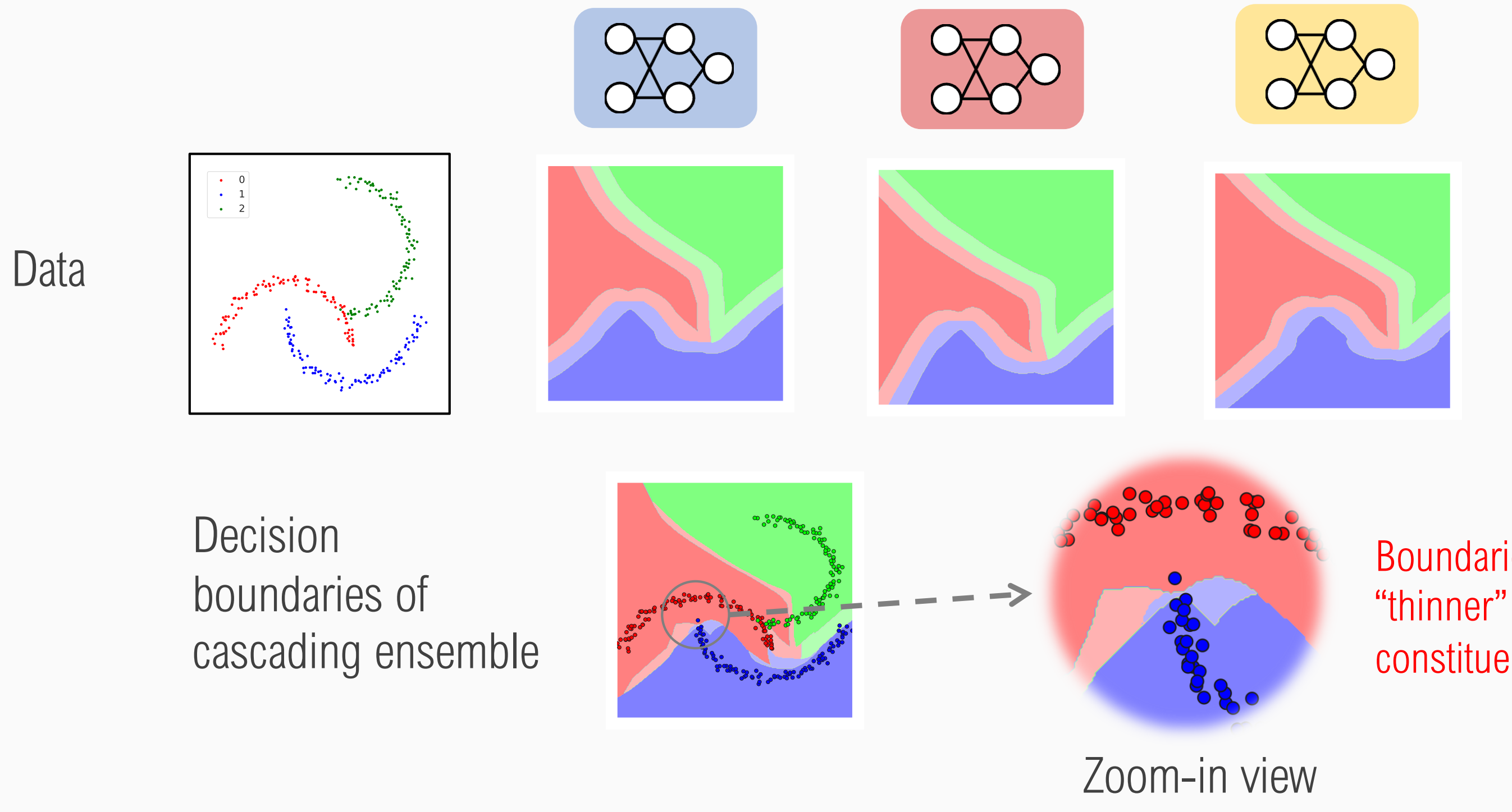
*ROBUST as $y_{blue}$*

**A violation of local robustness**

(see formal proof in the paper)

Constituent Models are not guaranteed to output the same classes for neighboring points without additionally memorizing other models' outputs.

### Visualizing Decision Boundaries for 2-D Datasets



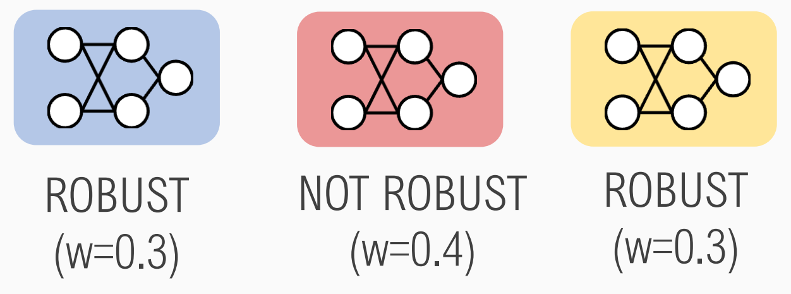
Decision boundaries of constituent models (Darker regions denote certified robustness)

Decision boundaries of cascading ensemble

Boundaries are actually much "thinner" compared to each constituent model

## Weighted Voting: A Sound Certification

### Definition of Weighted Voting



| Votes for ROBUST | > | Votes for Not ROBUST |
|---|---|---|
| 0.3 + 0.3 = 0.6 | | 0.4 |

**Certification of Ensemble: ROBUST**

(See proof of soundness in the paper)

### Can Weighted Voting Certify More Points?

(a thought experiment)

Let examples sorted as $x_1, x_2, \dots, x_{100}$

If (model 1) can certify $x_0, x_2, \dots, x_{49}$ and is correct. Certified Acc. = 50%

If (model 2) can certify $x_{25}, x_{26}, \dots, x_{74}$ and is correct. Certified Acc. = 50%

If (model 3) can certify $x_0 \dots x_{24}, x_{50} \dots x_{74}$ and is correct. Certified Acc. = 50%

And we let the weight for each model to be 1/3, then for each of the first 75 examples the voting ensemble is both correct and certifiably robust. → Certified Acc. = **75%**

## Attacking Cascading Ensemble

### Basic Idea

To attack a cascading ensemble at $x$, we find a neighbor of $x$, $x'$, which can be certified with a different label on another constituent model $F'$

- ○ **prior to** the one that certifies $x$;
- ○ or all other models **prior to F' can not certify** $x'$.

1 2 3

### Example

Suppose $x$ can not be certified by model 1 and is certified by model 2 as label cat, we find a neighbor $x'$ in the $\epsilon$-ball that can be certified by model 1 with a different label than cat, e.g. dog.

**Cascade Attack (CasA) on Cascading Ensemble Classifiers**

Suppose $x$ is certified as $y$ by the cascading ensemble

For (model) in (model 1, model 2, model 3)

$loss\ 1 = All\ models\ prior\ to$ (model) $fail\ to\ certify$

$loss\ 2 =$ (model) $outputs\ y' \neq y$

$(if$ (model) $is\ not\ the\ last, we\ additionally\ need\ y'to\ be\ certified.)$

$loss\ = loss\ 1 + loss\ 2$

$optimize\ x\ within\ \epsilon\ ball$

*(optimize with gradients and do early stop if we find an $x'$ that meet requirements.)*

### Experiments

We test CasA on the cascading ensembles used in Wong et al. (2018) from their public repository. We copy and paste the Certified Acc. from their publication and show % of points that are claimed to be robust but later found to be not using CasA (False Positives). We report the Empirical Robust Accuracy of these cascading ensembles under CasA.

| | | Single Model | Cascading Ensemble | | | |
|---|---|---|---|---|---|---|
| Dataset | $\ell_p, \epsilon$ | Certified Acc. | Unsound Certified Acc. Reported | % Of False Positive Results. | Clean Acc. | Empirical Robust Accuracy Under CasA |
| *MNIST* | $\ell_\infty, 0.1$ | *95.54* | *96.63* | *88.71* | *96.62* | *11.17* |
| *CIFAR-10* | $\ell_\infty, 2/255$ | *52.65* | *64.87* | *10.47* | *65.13* | *58.15* |
| *MNIST* | $\ell_2, 1.58$ | *43.52* | *75.58* | *44.72* | *80.43* | *43.46* |
| *CIFAR-10* | $\ell_2, 36/255$ | *50.26* | *58.72* | *2.70* | *58.76* | *57.17* |

learn more | check out our talk and the full paper for more!

code available on GitHub

full paper
https://openreview.net/pdf?id=tQG-o3SeipT

implementation
https://github.com/TristaChi/ensembleKW