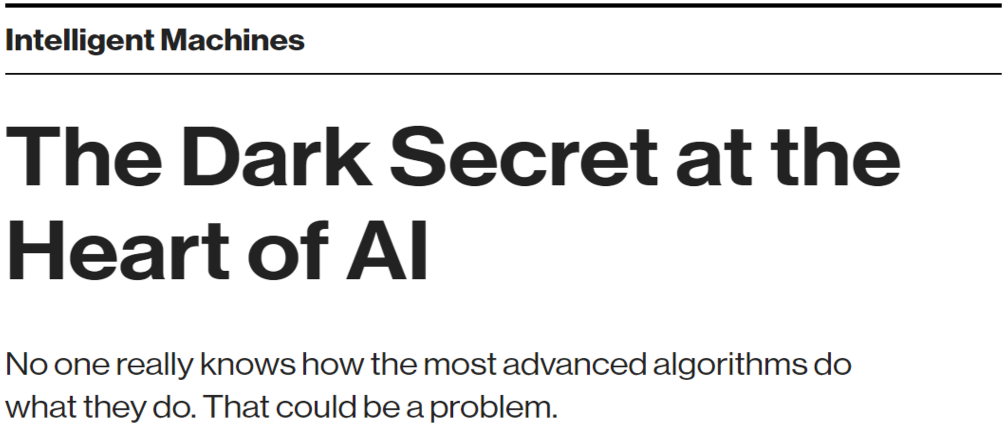
Reliable and Interpretable Artificial Intelligence

Martin Vechev

Department of Computer Science, ETH Zurich

ETH Zürich



Intelligent Machines

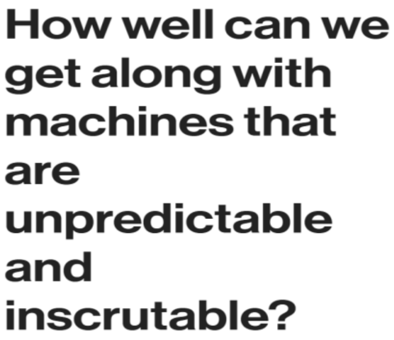
The Dark Secret at the Heart of AI

No one really knows how the most advanced algorithms do what they do. That could be a problem.

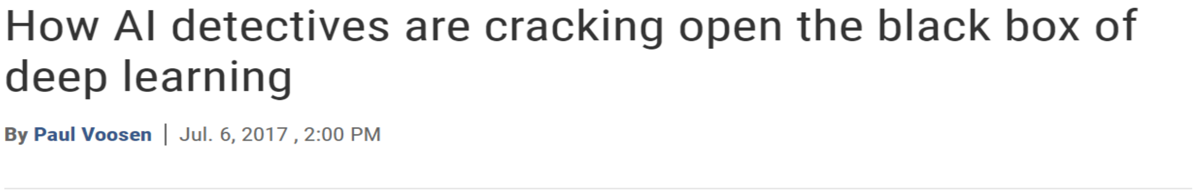
AI programs exhibit racial and gender biases, research reveals

Machine learning algorithms are picking up deeply ingrained race and gender prejudices concealed within the patterns of language use, scientists say

AI has the potential to reinforce existing biases because, unlike humans, algorithms are unequipped to consciously counteract learned biases, researchers warn. Photograph: KTS Design/Getty Images/Science Photo Library RF

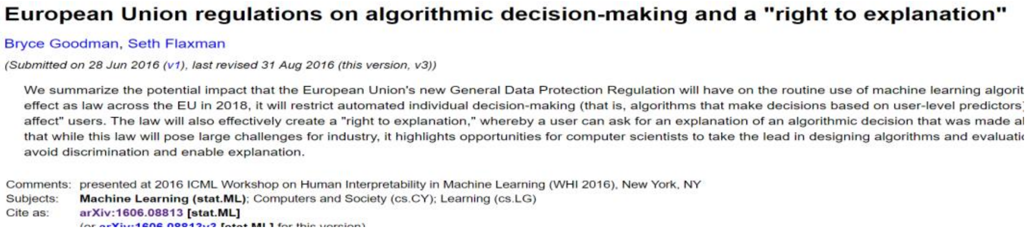


How well can we get along with machines that are unpredictable and inscrutable?



How AI detectives are cracking open the black box of deep learning

By Paul Voosen | Jul. 6, 2017 , 2:00 PM



European Union regulations on algorithmic decision-making and a "right to explanation"

Bryce Goodman, Seth Flaxman

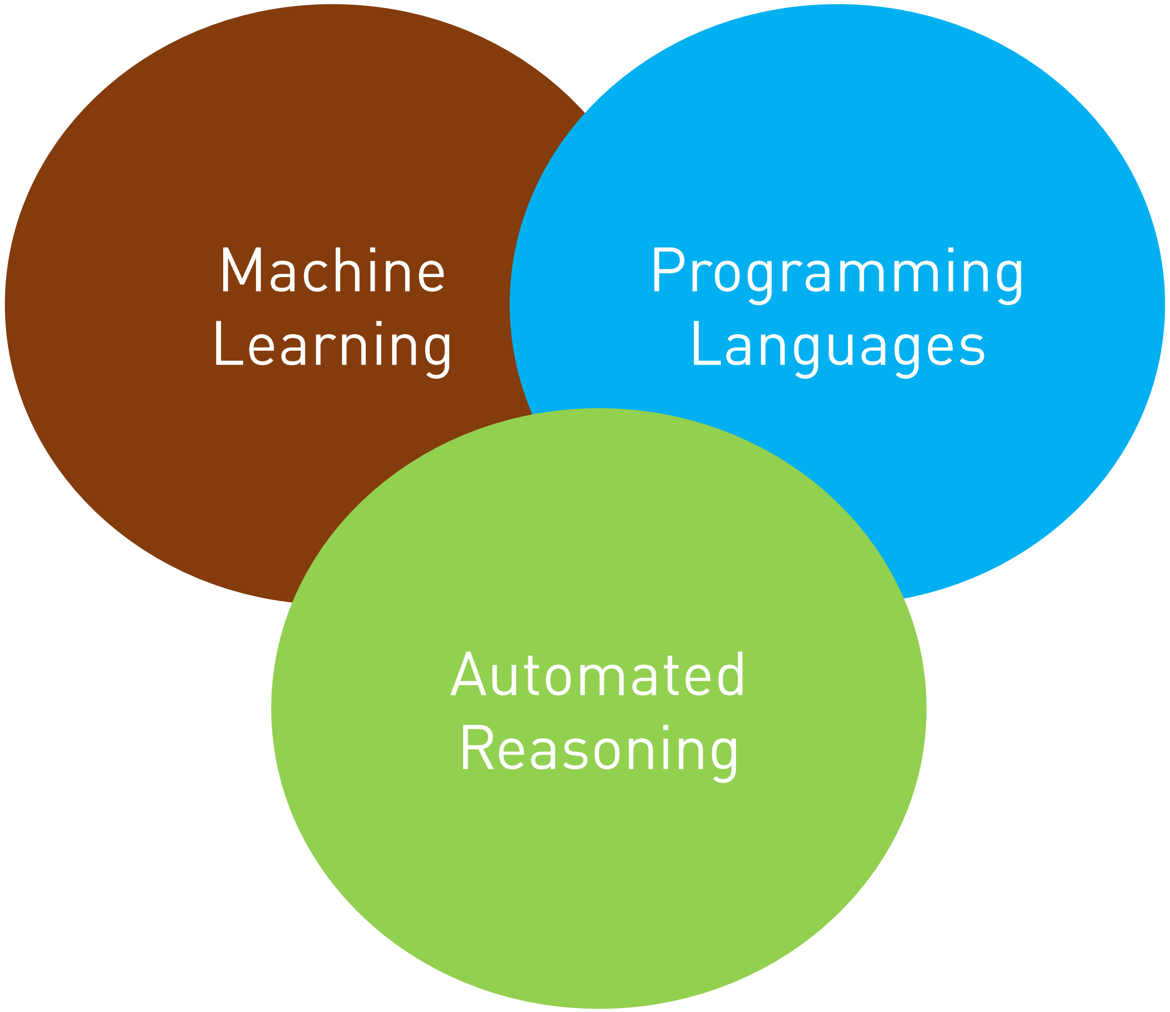
(Submitted on 28 Jun 2016 (v1), last revised 31 Aug 2016 (this version, v3))

We summarize the potential impact that the European Union's new General Data Protection Regulation will have on the routine use of machine learning algorit effect as law across the EU in 2018, it will restrict automated individual decision-making (that is, algorithms that make decisions based on user-level predictors affect" users. The law will also effectively create a "right to explanation," whereby a user can ask for an explanation of an algorithmic decision that was made al that while this law will pose large challenges for industry, it highlights opportunities for computer scientists to take the lead in designing algorithms and evaluati avoid discrimination and enable explanation.

Comments: presented at 2016 ICML Workshop on Human Interpretability in Machine Learning (WHI 2016), New York, NY
Subjects: **Machine Learning (stat.ML)**; Computers and Society (cs.CY); Learning (cs.LG)
Cite as: **arXiv:1606.08813 [stat.ML]**
(or arXiv:1606.08813v3 [stat.ML] for this version)

Artificial intelligence pioneer says we need to start over (Sept 15, 2017)

Reliable and Interpretable Artificial Intelligence



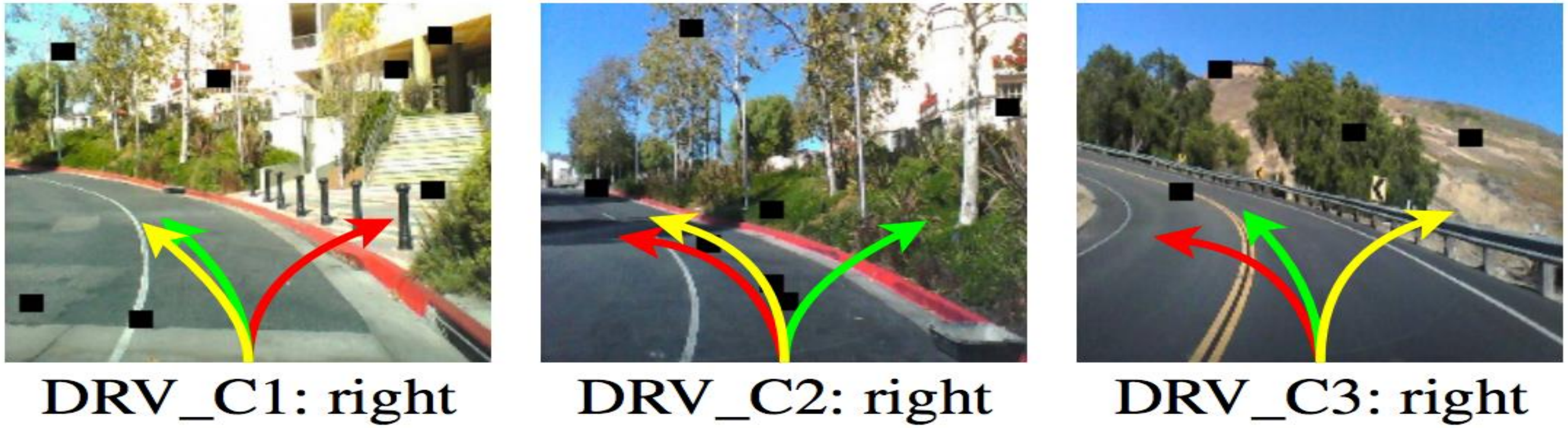
Machine Learning

Programming Languages

Automated Reasoning

How good (robust) is your neural net?

Neural networks are ***not*** robust to input perturbations
(e.g., image rotation / change of lighting)



DRV_C1: right

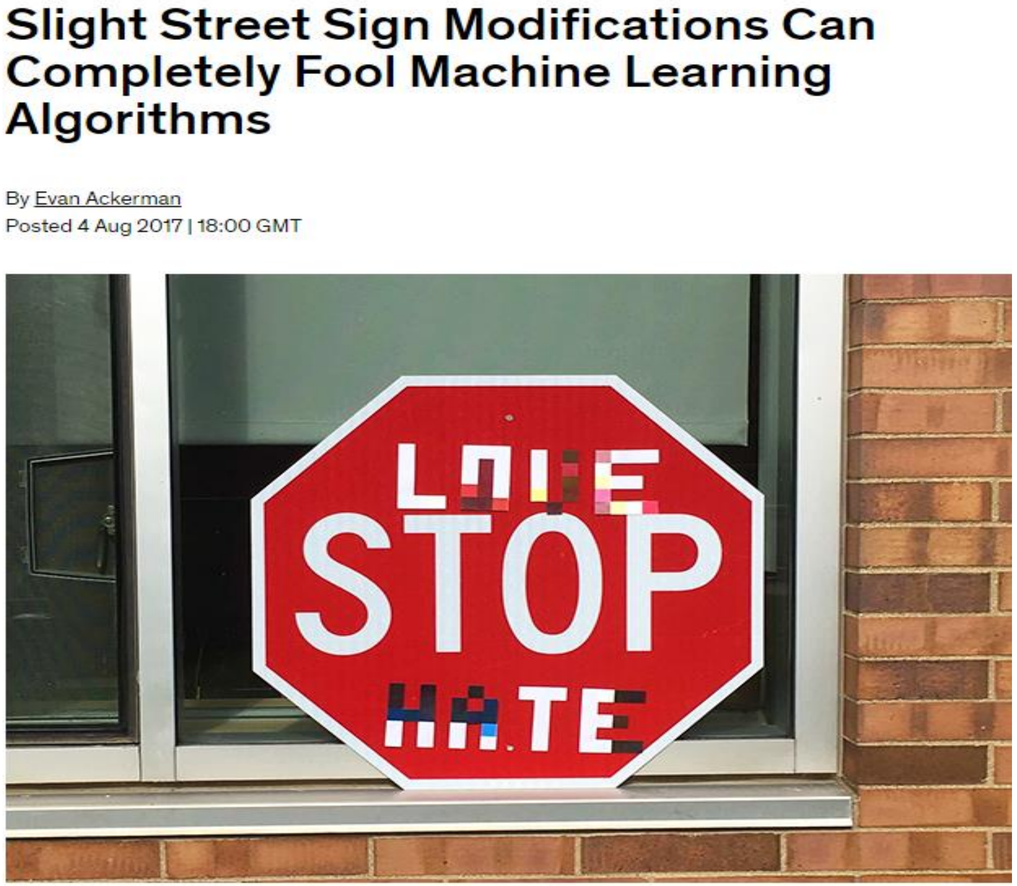
DRV_C2: right

DRV_C3: right

Misclassifications in neural networks deployed in self-driving cars [1]
In each picture one of the 3 networks makes a mistake…

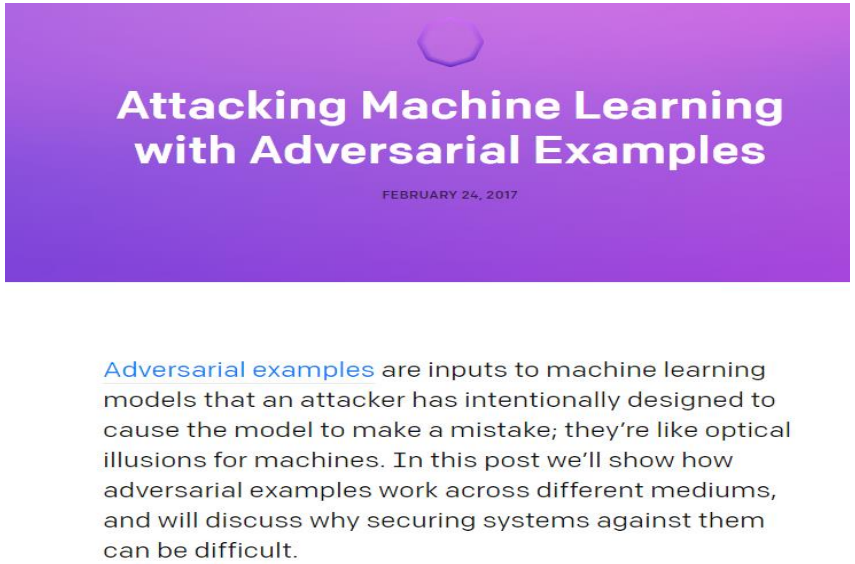
[1] Pei et. al., DeepXplore: Automated Whitebox Testing of Deep Learning Systems, SOSP 2017

Attacks on Machine Learning…



Slight Street Sign Modifications Can Completely Fool Machine Learning Algorithms

By Evan Ackerman
Posted 4 Aug 2017 | 18:00 GMT



Attacking Machine Learning with Adversarial Examples

FEBRUARY 24, 2017

Adversarial examples are inputs to machine learning models that an attacker has intentionally designed to cause the model to make a mistake; they're like optical illusions for machines. In this post we'll show how adversarial examples work across different mediums, and will discuss why securing systems against them can be difficult.

Related work: Adversarial examples

- Seminal paper: Szegedy, Christian, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. "**Intriguing properties of neural networks**." *arXiv preprint arXiv:1312.6199* (2013).
- Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "**Explaining and harnessing adversarial examples**." *arXiv preprint arXiv:1412.6572* (2014).
- Nguyen, Anh, Jason Yosinski, and Jeff Clune. "**Deep neural networks are easily fooled: High confidence predictions for unrecognizable images.**" In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 427-436. 2015.
- Grosse, Kathrin, Nicolas Papernot, Praveen Manoharan, Michael Backes, and Patrick McDaniel. "**Adversarial perturbations against deep neural networks for malware classification**." *arXiv preprint arXiv:1606.04435* (2016).
- Papernot, Nicolas, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami. "**The limitations of deep learning in adversarial settings**." In *Security and Privacy (EuroS&P), 2016 IEEE European Symposium on*, pp. 372-387. IEEE, 2016.
- Kurakin, Alexey, Ian Goodfellow, and Samy Bengio. "**Adversarial examples in the physical world**." *arXiv preprint arXiv:1607.02533* (2016).
- Warde-Farley, David, Ian Goodfellow, T. Hazan, G. Papandreou, and D. Tarlow. "**Adversarial perturbations of deep neural networks**." *Perturbations, Optimization, and Statistics* (2016): 1-32.
- Evtimov, Ivan, Kevin Eykholt, Earlence Fernandes, Tadayoshi Kohno, Bo Li, Atul Prakash, Amir Rahmati, and Dawn Song. "**Robust Physical-World Attacks on Machine Learning Models**." *arXiv preprint arXiv:1707.08945*(2017).

Related Work: Robustness Guarantees

- Gu, Shixiang, and Luca Rigazio. "**Towards deep neural network architectures robust to adversarial examples**." *arXiv preprint arXiv:1412.5068* (2014).
- Bastani, Osbert, Yani Ioannou, Leonidas Lampropoulos, Dimitrios Vytiniotis, Aditya Nori, and Antonio Criminisi. "**Measuring neural net robustness with constraints**." In *Advances in Neural Information Processing Systems*, pp. 2613-2621. 2016.
- Katz, Guy, Clark Barrett, David Dill, Kyle Julian, and Mykel Kochenderfer. "**Reluplex: An efficient SMT solver for verifying deep neural networks**." arXiv preprint arXiv:1702.01135 (2017).

Related Work: Systematic Testing

- Pei, Kexin, Yinzhi Cao, Junfeng Yang, and Suman Jana. "**DeepXplore: Automated Whitebox Testing of Deep Learning Systems**." *SOSP, 2017*
- Huang, Xiaowei, Marta Kwiatkowska, Sen Wang, and Min Wu. "**Safety verification of deep neural networks**." In *International Conference on Computer Aided Verification*, pp. 3-29. Springer, Cham, 2017.
- Dreossi, Tommaso, Alexandre Donzé, and Sanjit A. Seshia. "**Compositional Falsification of Cyber-Physical Systems with Machine Learning Components**." In *NASA Formal Methods Symposium*, pp. 357-372. Springer, Cham, 2017.

Wanted: **Automated** and **scalable** analysis to verify realistic NNs

Useful in:

- Ensuring correctness of a larger (CPS) system that uses the NN
- Proving robustness of the NN (beyond finding adversarial examples)
- Learning interpretable specs of the NN
- Comparing NNs

(joint work with Timon Gehr, Matthew Mirman, Dana Drachsler-Cohen, Petar Tsankov, Swarat Chaudhuri)

Problem Statement and Challenges

Neural Network Analysis Problem

Given

- a **neural network N**
- a **property over inputs φ**
- a **property over outputs ψ**

check whether $\forall \boldsymbol{i} \in \boldsymbol{I}.\, \boldsymbol{i} \vDash \boldsymbol{\varphi} \Longrightarrow \boldsymbol{N}(\boldsymbol{i}) \vDash \boldsymbol{\psi}$ holds

Challenges:

- The property φ over inputs usually captures an **unbounded set of inputs**
- Existing symbolic solutions **do not scale** to large networks (e.g. conv nets)

Key Observation: AI for AI

Deep Neural Nets:

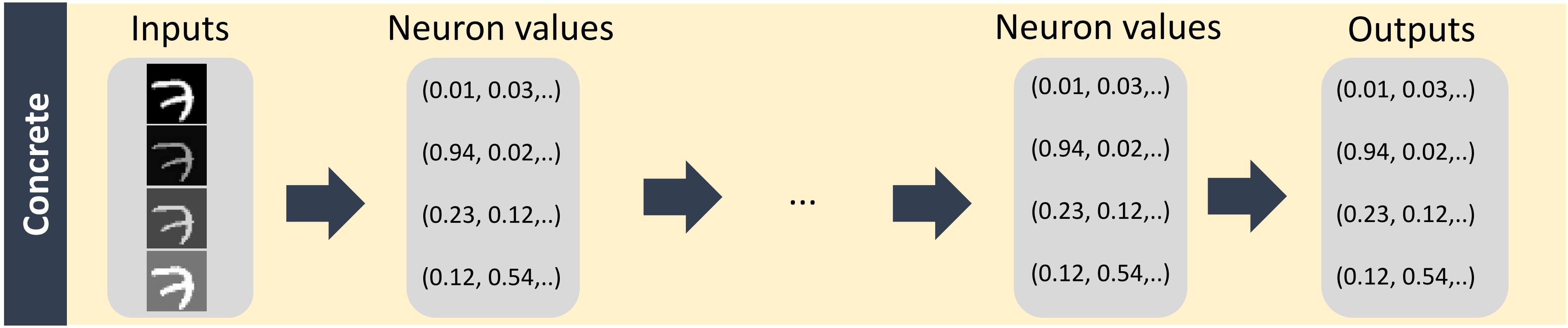
Affine transforms + Restricted non-linearity

+

Abstract Interpretation:

Scalable and Precise Numerical Domains

AI^2: Abstract Interpretation for NNs



Concrete

Inputs

Neuron values

(0.01, 0.03,..)

(0.94, 0.02,..)

(0.23, 0.12,..)

(0.12, 0.54,..)

...

Neuron values

(0.01, 0.03,..)

(0.94, 0.02,..)

(0.23, 0.12,..)

(0.12, 0.54,..)

Outputs

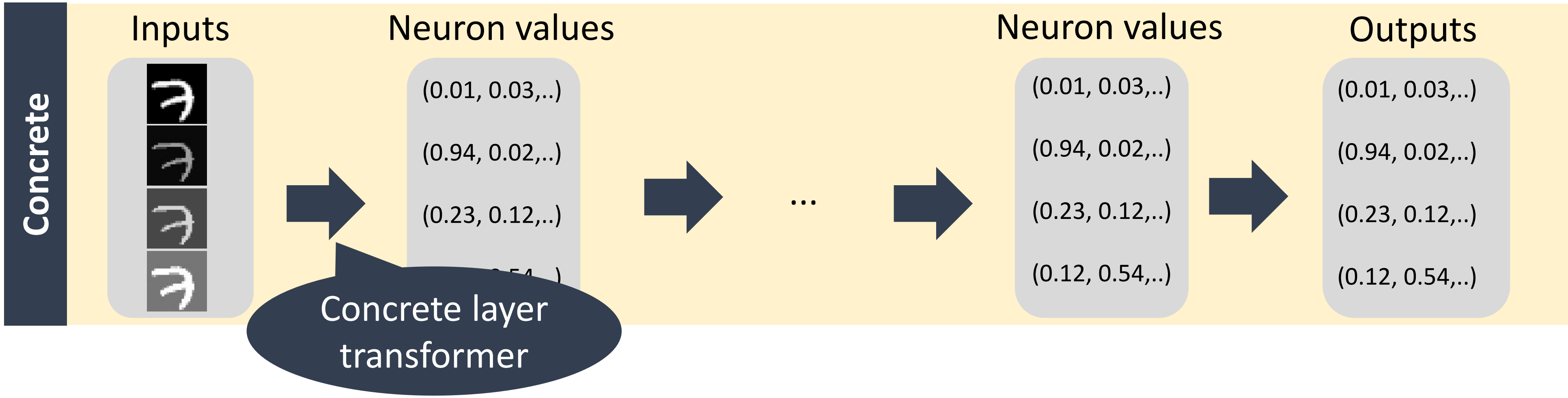
(0.01, 0.03,..)

(0.94, 0.02,..)

(0.23, 0.12,..)

(0.12, 0.54,..)

AI^2: Abstract Interpretation for NNs



Concrete

Inputs

Neuron values

(0.01, 0.03,..)

(0.94, 0.02,..)

(0.23, 0.12,..)

0.54,..)

Concrete layer transformer

...

Neuron values

(0.01, 0.03,..)

(0.94, 0.02,..)

(0.23, 0.12,..)

(0.12, 0.54,..)

Outputs

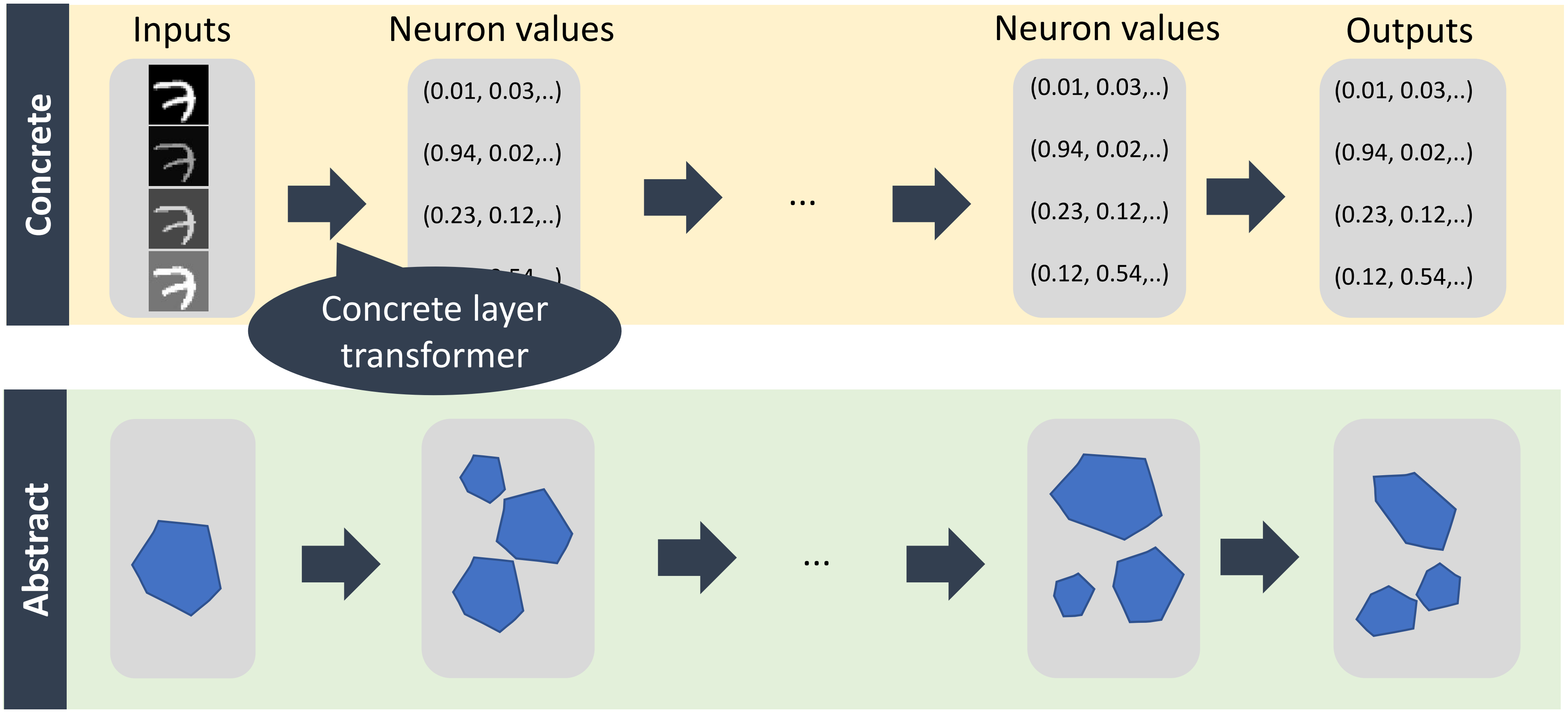
(0.01, 0.03,..)

(0.94, 0.02,..)

(0.23, 0.12,..)

(0.12, 0.54,..)

AI^2: Abstract Interpretation for NNs



Concrete

Inputs

Neuron values

(0.01, 0.03,..)

(0.94, 0.02,..)

(0.23, 0.12,..)

0.54,..)

…

Neuron values

(0.01, 0.03,..)

(0.94, 0.02,..)

(0.23, 0.12,..)

(0.12, 0.54,..)

Outputs

(0.01, 0.03,..)

(0.94, 0.02,..)

(0.23, 0.12,..)

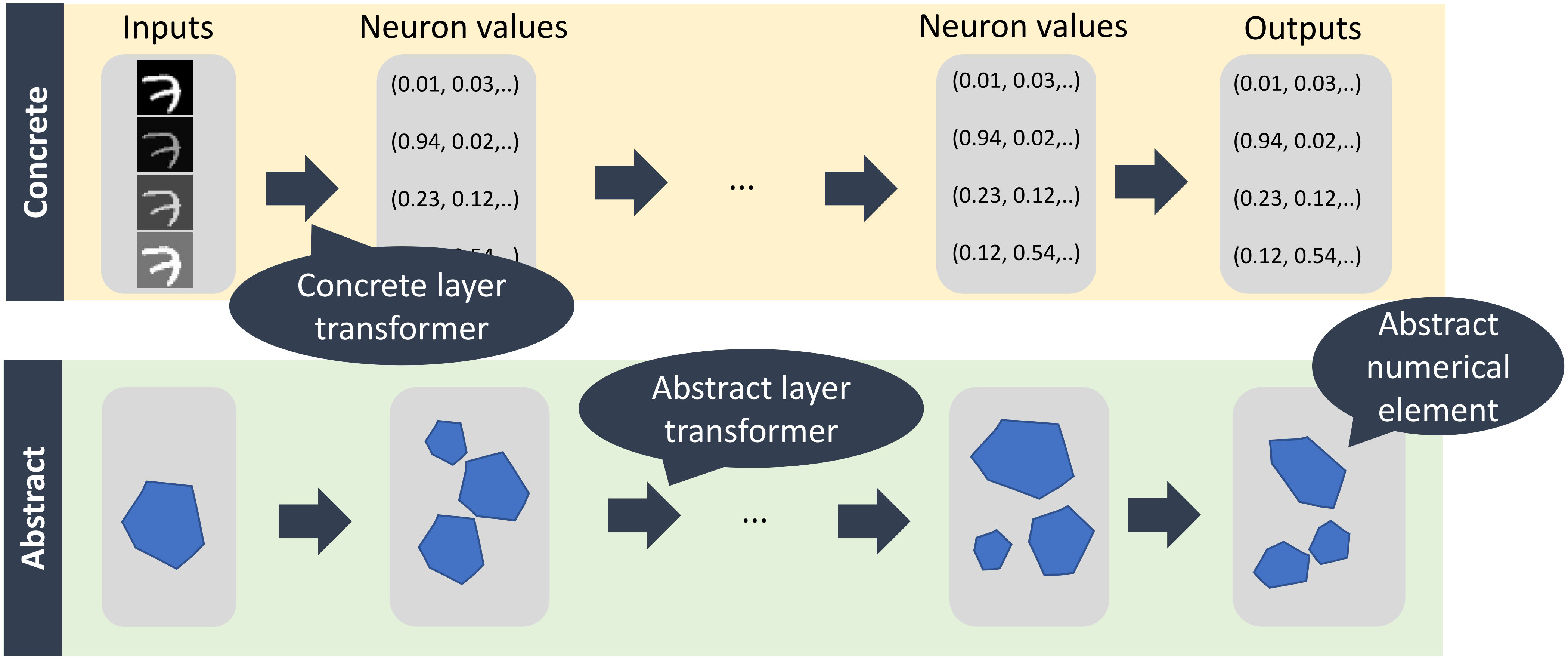
(0.12, 0.54,..)

Concrete layer transformer

Abstract

…

AI^2: Abstract Interpretation for NNs



Zonotope Abstract Domain

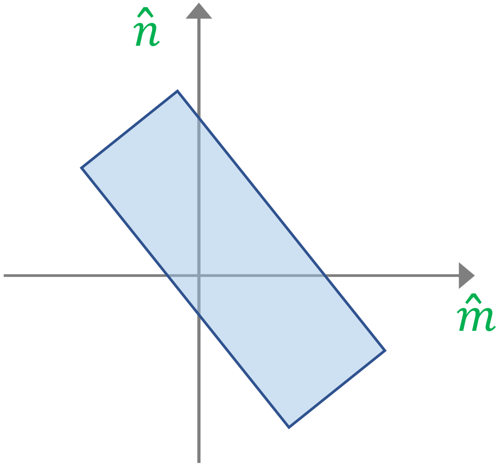
An abstract neuron is captured in an **affine form**. Example for two concrete neurons n and m:

$$\hat{n} = {a_0}^n + \sum_{i=1}^{k} {a_i}^n \epsilon_i$$

$$\hat{m} = {a_0}^m + \sum_{i=1}^{k} {a_i}^m \epsilon_i$$

The meaning (γ) is a polytope centered around ${a_0}^n$ and ${a_0}^m$

Example:



Zonotope Abstract Domain

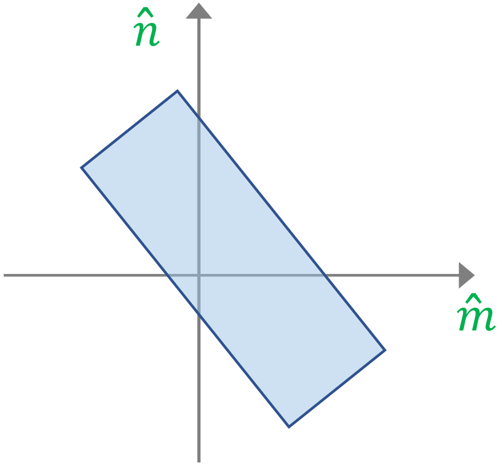
An abstract neuron is captured in an **affine form**. Example for two concrete neurons n and m:

$$\hat{n} = a_0^{n} + \sum_{i=1}^{k} a_i^{n} \epsilon_i$$

$$\hat{m} = a_0^{m} + \sum_{i=1}^{k} a_i^{m} \epsilon_i$$

The meaning (γ) is a polytope centered around a_0^{n} and a_0^{m}

Example:



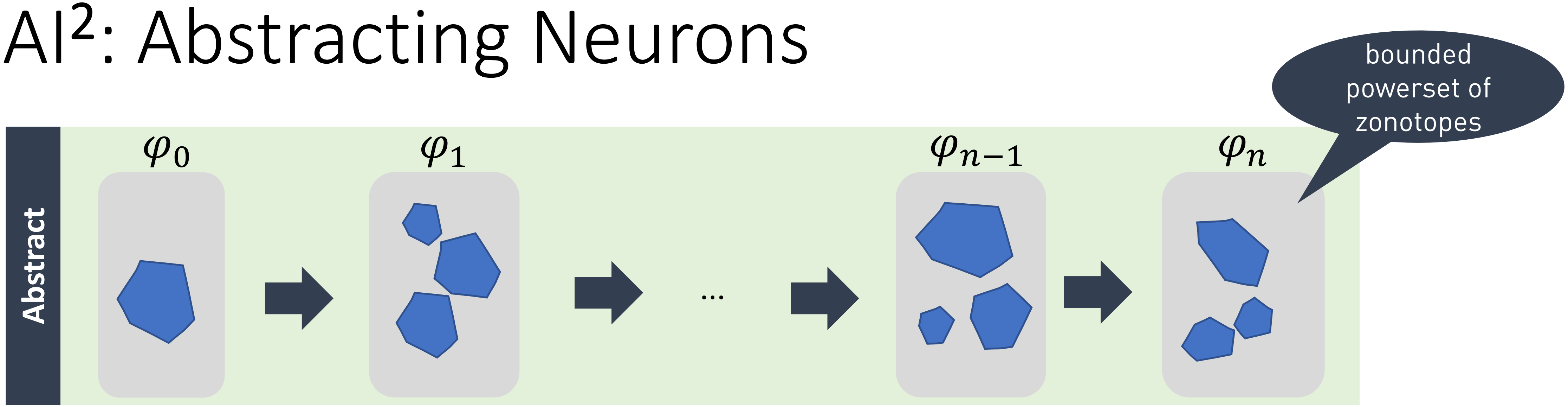
ϵ_i : noise terms ranging [-1,1] shared between abstract neurons

a_i^{n} : real number that controls magnitude of noise

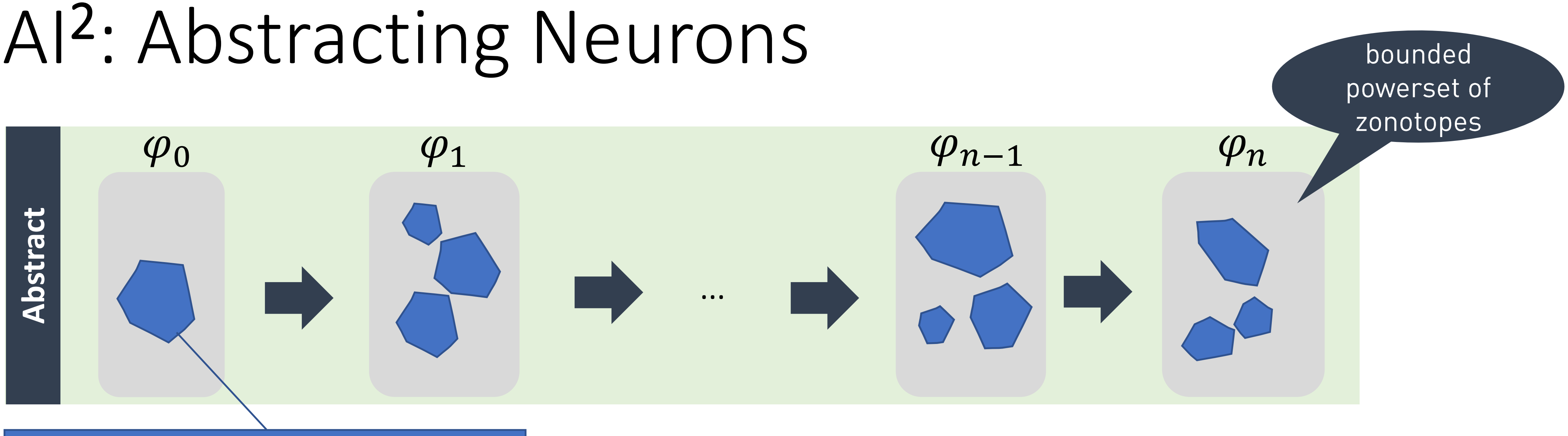
Closed under affine transforms, e.g., $\hat{n} + \hat{m}$

Not closed under joins and meets, e.g.,: $\hat{n} \sqcup \hat{m}$, $\hat{n} \gnsim \hat{m}$

AI^2: Abstracting Neurons



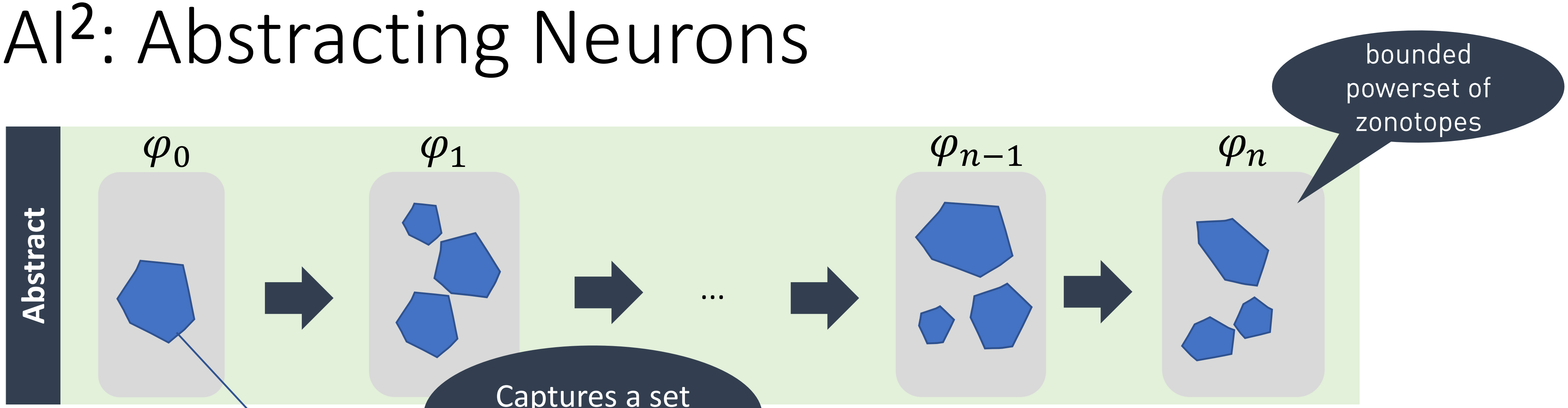
AI^2: Abstracting Neurons



Robustness specification φ_0

$x_0 = 0$
$x_1 = 0.975 + 0.025\epsilon_1$
$x_2 = 0.125$
…
$x_{784} = 0.938 + 0.062\epsilon_{784}$
$\forall i.\, \epsilon_i \in [0,1]$

AI^2: Abstracting Neurons



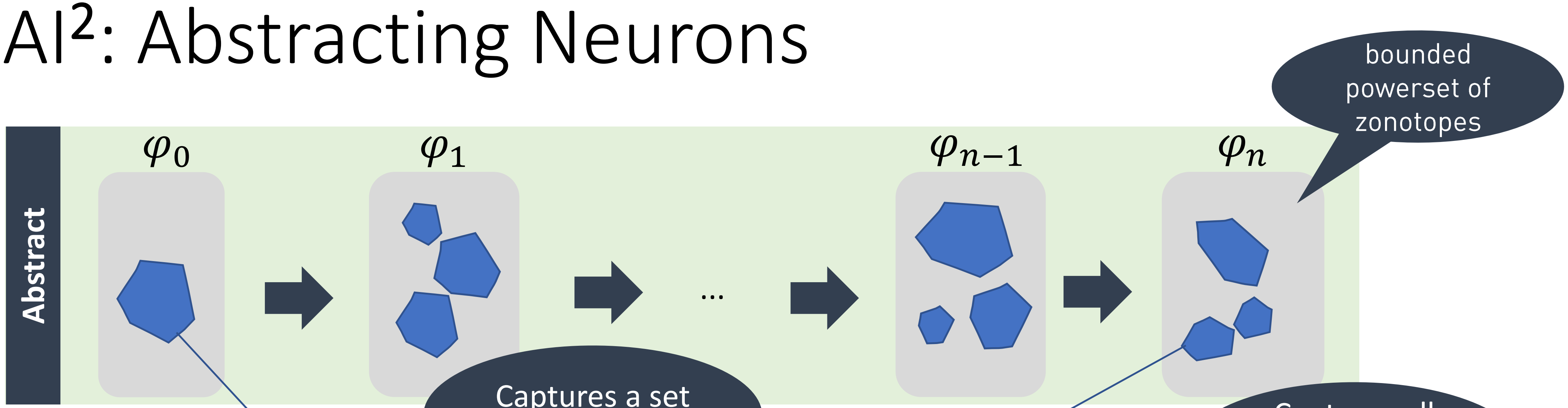
bounded powerset of zonotopes

Captures a set of images

Robustness specification φ_0

$x_0 = 0$
$x_1 = 0.975 + 0.025\epsilon_1$
$x_2 = 0.125$
…
$x_{784} = 0.938 + 0.062\epsilon_{784}$
$\forall i.\, \epsilon_i \in [0,1]$

AI^2: Abstracting Neurons



Captures a set of images

Captures all possible output vectors

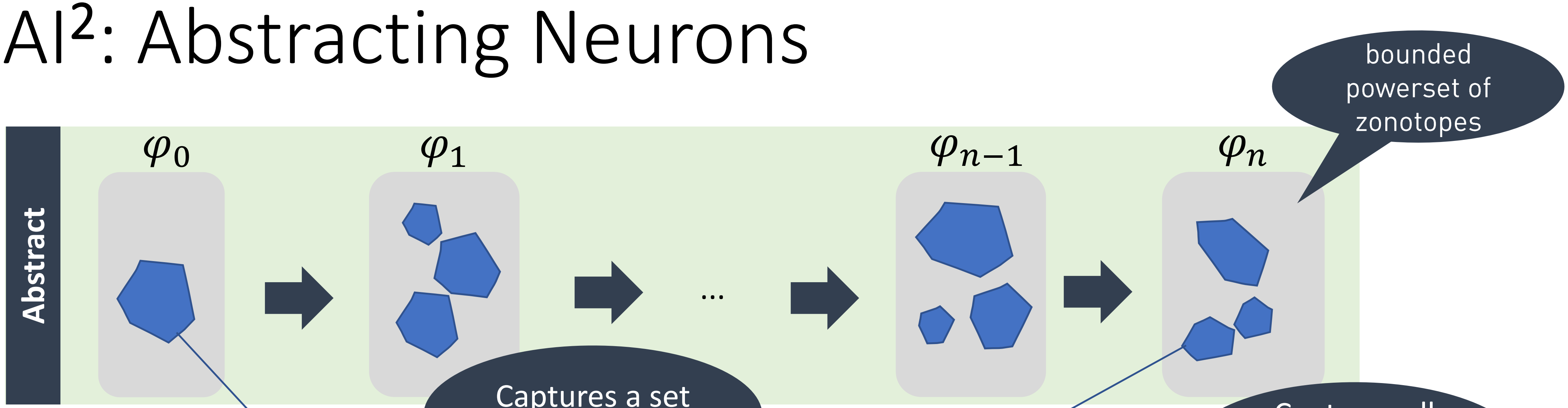
Robustness specification φ_0

$x_0 = 0$
$x_1 = 0.975 + 0.025\epsilon_1$
$x_2 = 0.125$
…
$x_{784} = 0.938 + 0.062\epsilon_{784}$
$\forall i.\, \epsilon_i \in [0,1]$

Output constraint φ_n

$x_0 = 0$
$x_1 = 2.60 + 0.015\epsilon_0 + 0.023\epsilon_1 + 5.181\epsilon_2 + \cdots$
$x_2 = 4.63 - 0.005\epsilon_0 - 0.006\epsilon_1 + 0.023\epsilon_1 + \cdots$
…
$x_9 = 0.12 - 0.125\epsilon_0 + 0.102\epsilon_1 + 3.012\epsilon_1 + \cdots$
$\forall i.\, \epsilon_i \in [0,1]$

AI^2: Abstracting Neurons



Captures a set of images

Robustness specification φ_0

$x_0 = 0$
$x_1 = 0.975 + 0.025\epsilon_1$
$x_2 = 0.125$
…
$x_{784} = 0.938 + 0.062\epsilon_{784}$
$\forall i.\, \epsilon_i \in [0,1]$

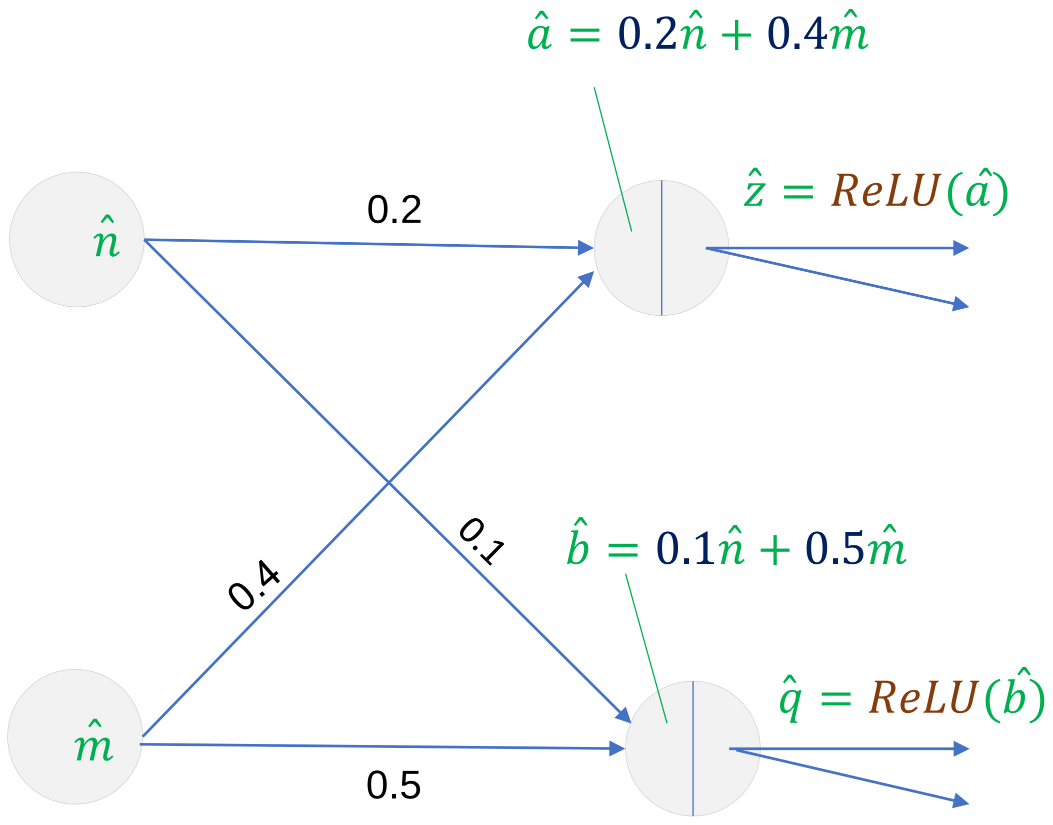
Output constraint φ_n

Captures all possible output vectors

$x_0 = 0$
$x_1 = 2.60 + 0.015\epsilon_0 + 0.023\epsilon_1 + 5.181\epsilon_2 + \cdots$
$x_2 = 4.63 - 0.005\epsilon_0 - 0.006\epsilon_1 + 0.023\epsilon_1 + \cdots$
…
$x_9 = 0.12 - 0.125\epsilon_0 + 0.102\epsilon_1 + 3.012\epsilon_1 + \cdots$
$\forall i.\, \epsilon_i \in [0,1]$

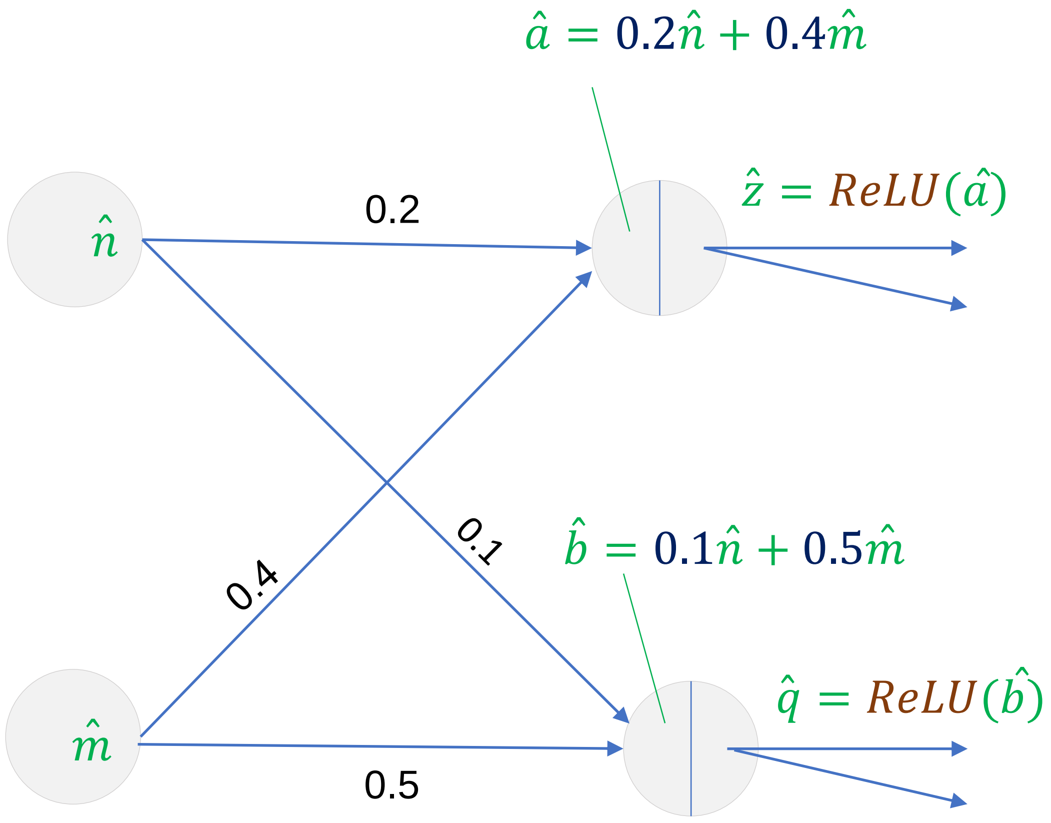
Label i is possible iff: $\varphi_n \sqcap \{\forall j.\, x_i \geq x_j\} \neq \bot$

Abstract Neuron Transformers


$\hat{a} = 0.2\hat{n} + 0.4\hat{m}$
$\hat{z} = ReLU(\hat{a})$
$\hat{n}$
0.2
0.1
0.4
$\hat{b} = 0.1\hat{n} + 0.5\hat{m}$
$\hat{q} = ReLU(\hat{b})$
$\hat{m}$
0.5

Abstract Neuron Transformer

Activation function: $z = ReLU(a) = \max(0, a)$

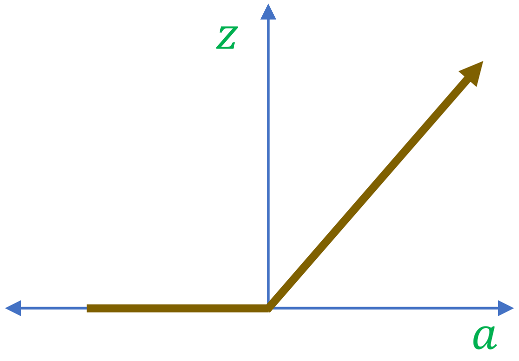


$\hat{a} = 0.2\hat{n} + 0.4\hat{m}$

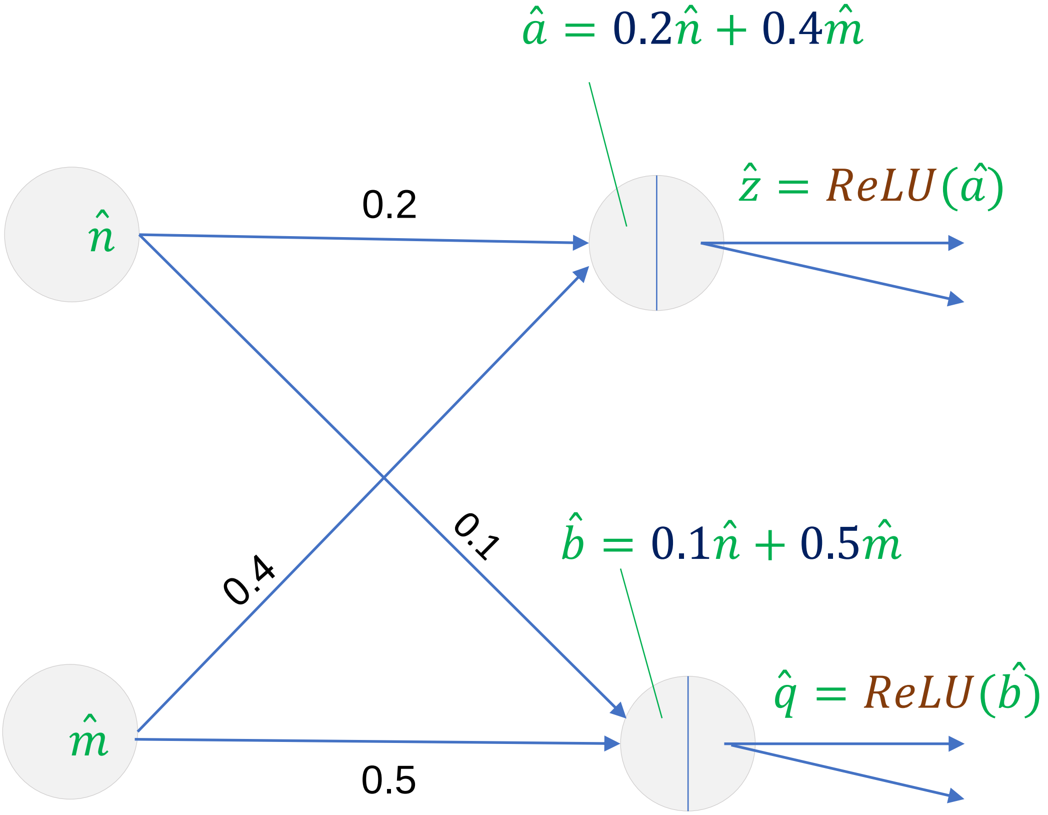
$\hat{z} = ReLU(\hat{a})$

$\hat{b} = 0.1\hat{n} + 0.5\hat{m}$

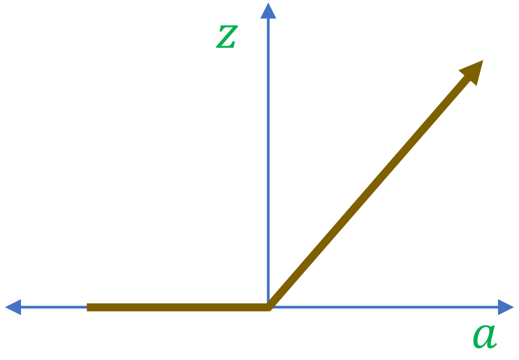
$\hat{q} = ReLU(\hat{b})$



Abstract Neuron Transformer


$\hat{a} = 0.2\hat{n} + 0.4\hat{m}$
$\hat{z} = ReLU(\hat{a})$
$\hat{n}$
0.2
0.1
0.4
$\hat{b} = 0.1\hat{n} + 0.5\hat{m}$
$\hat{q} = ReLU(\hat{b})$
$\hat{m}$
0.5

Activation function: $z = ReLU(a) = \max(0, a)$



ReLU abstract transformer:

$$f^{\#}_{ReLU} = f^{\#}_k \circ \cdots \circ f^{\#}_1$$

$$f^{\#}_i(\psi) = (\psi \sqcap \{x_i \geq 0\}) \sqcup \psi_0$$

$$\psi_0 = \begin{cases} \psi[x_i \longmapsto 0] & \text{if } (\psi \sqcap \{x_i < 0\}) \neq \bot \\ \bot & \text{otherwise} \end{cases}$$

The AI^2 System

Supports neural networks with:

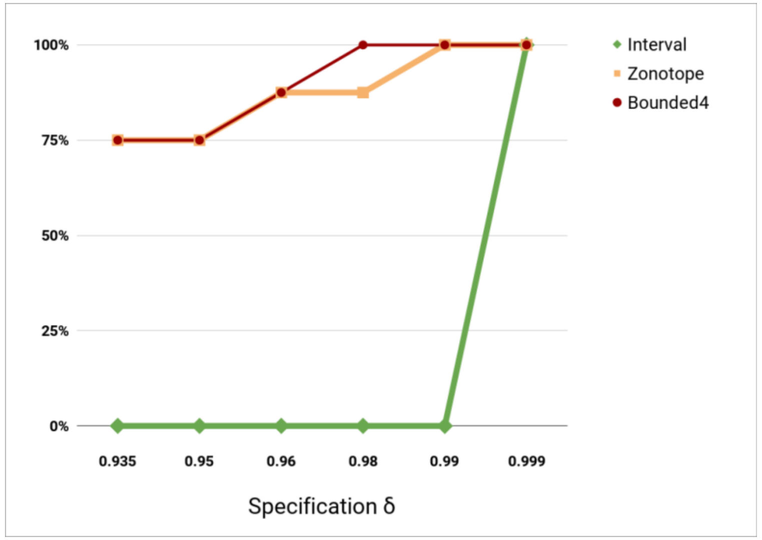
Layers: Fully-connected, convolutional, max-pooling, flattening
Activation functions: ReLU

Supported numerical domains:

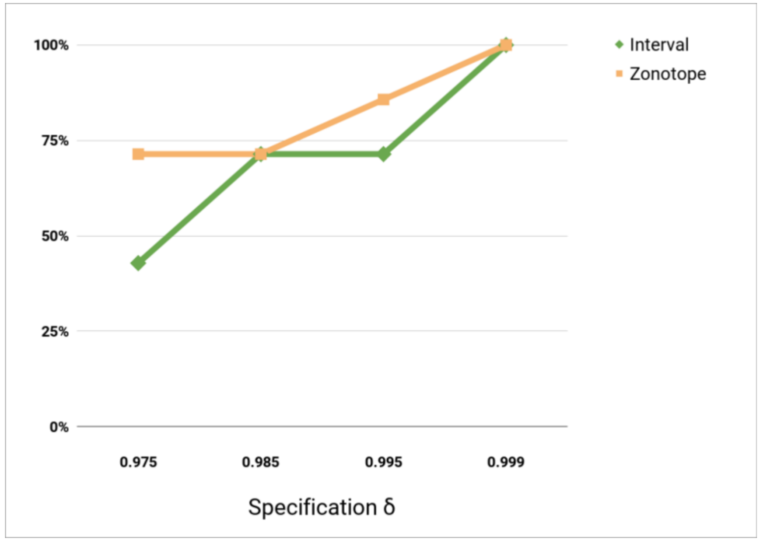
Intervals, Zonotopes, Polyhedra, Bounded powerset domain

Experimental Results

MNIST ConvNet 6 layers, 15K neurons



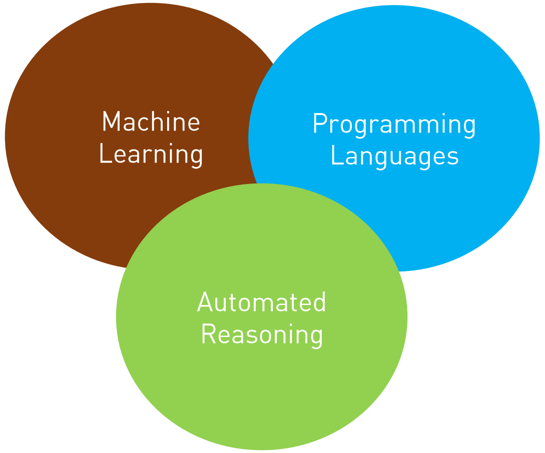
CIFAR-10 ConvNet 6 layers, 57K neurons



Intelligent Machines

The Dark Secret at the Heart of AI

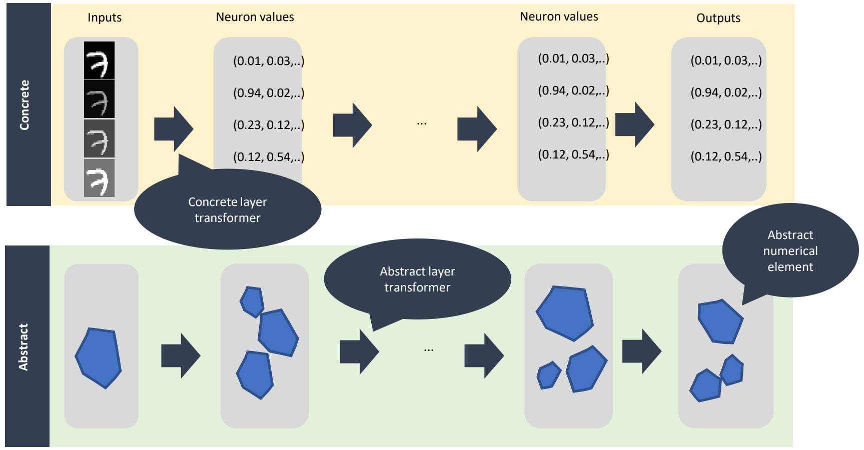
No one really knows how the most advanced algorithms do what they do. That could be a problem.



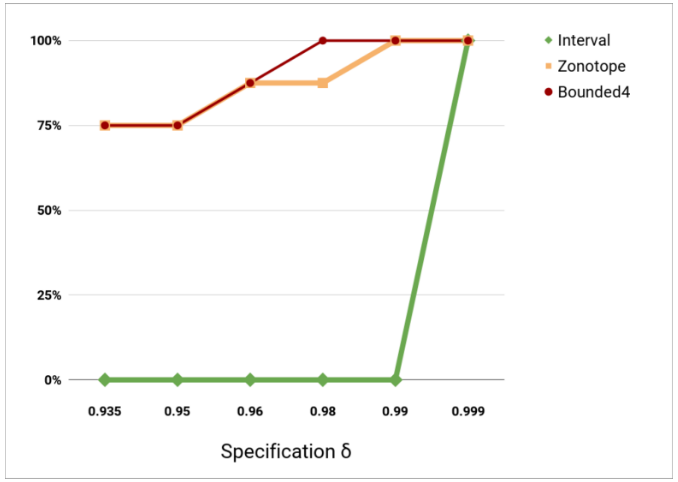
Slight Street Sign Modifications Can Completely Fool Machine Learning Algorithms

By Evan Ackerman
Posted 4 Aug 2017 | 18:00 GMT

AI^2: AI for AI



Handles Convolutional Nets


Specification δ
Interval
Zonotope
Bounded4