Deep Logical Circuits

Generalization through Interpretation

Contribution:

Algorithm:

Neural Network -> Boolean Formula

<->linear decision tree

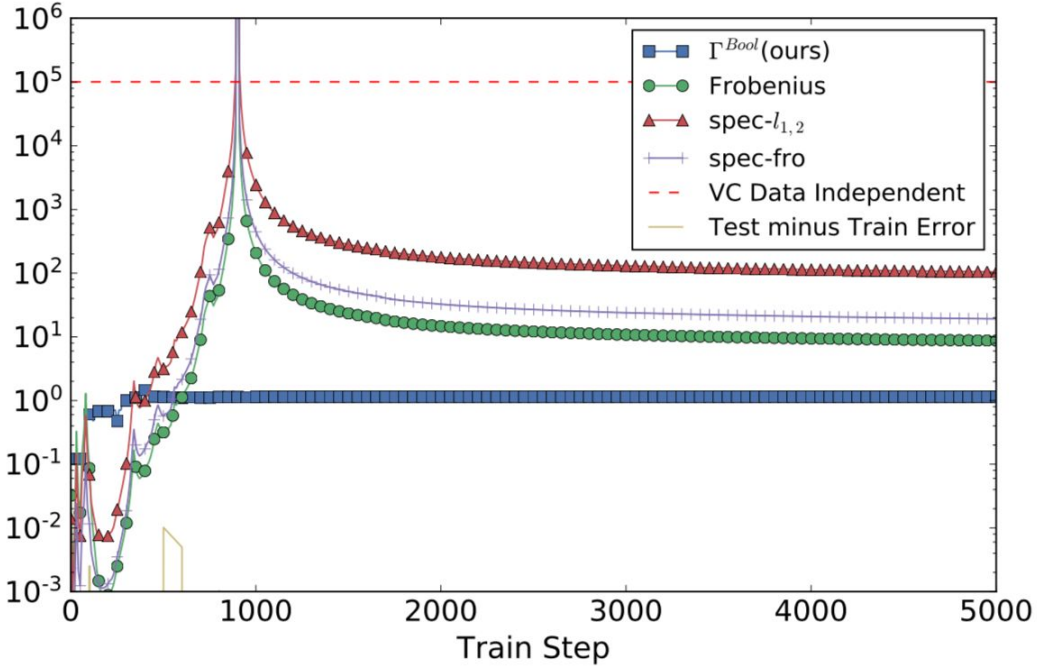
Allows:

- Very tight gen bounds
  - & Interpretability

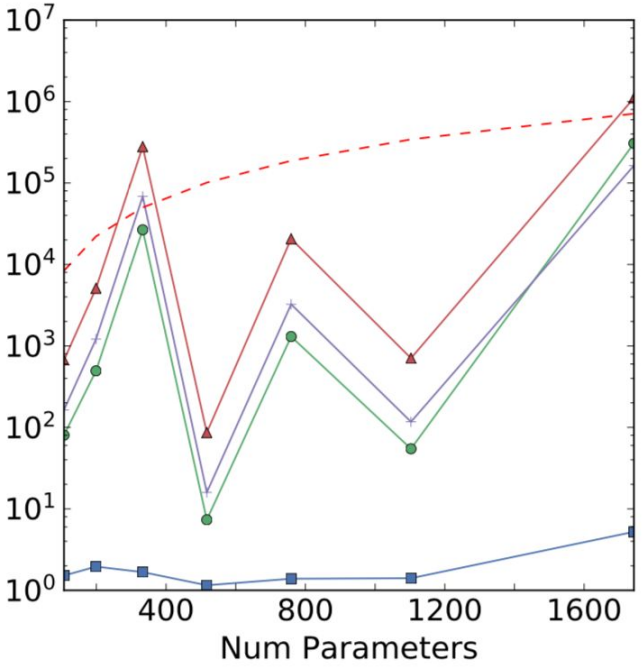
Should answer together:

- “Did it learn?”
- “What did it learn?”

Spoilers

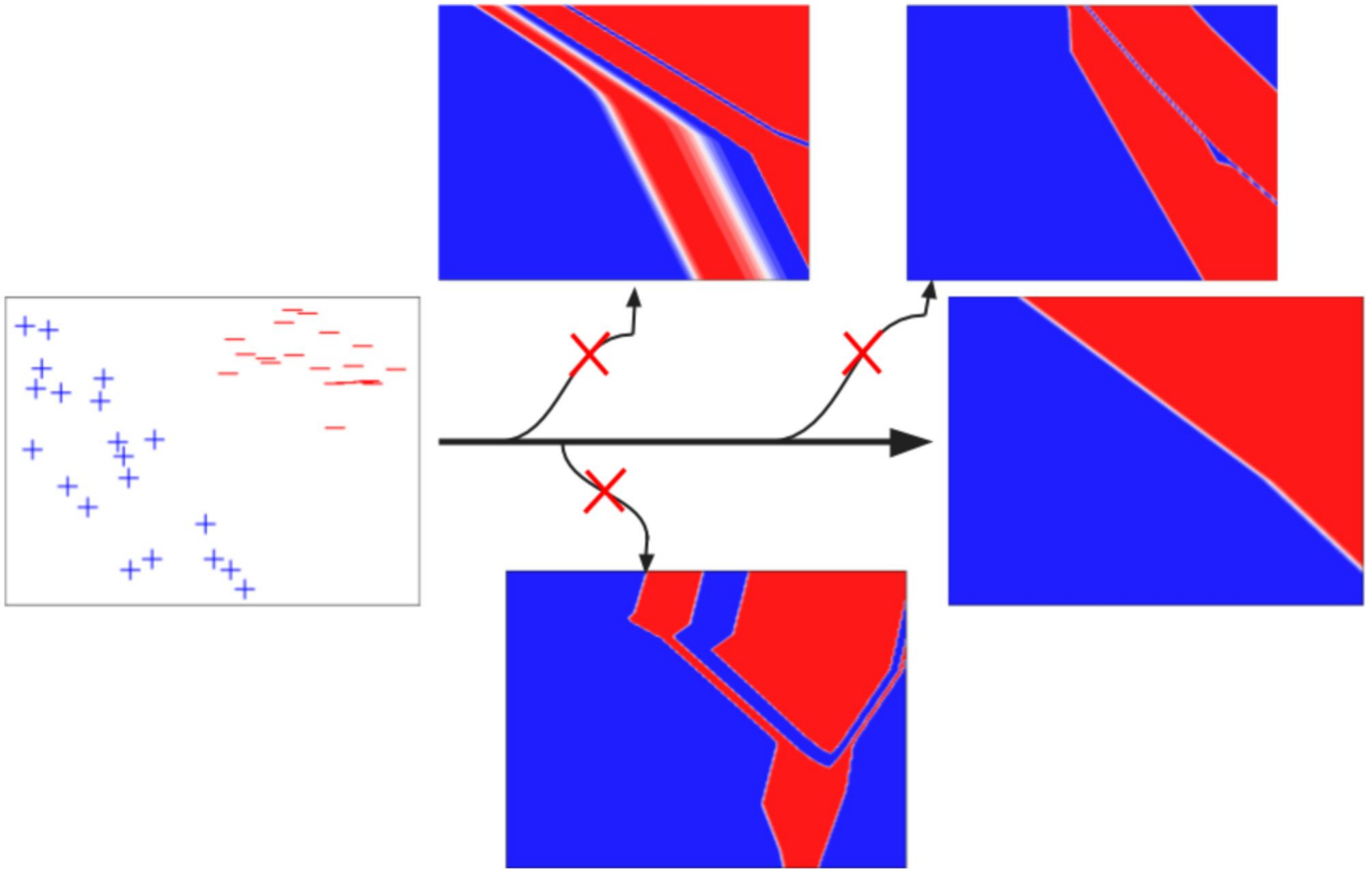


(a) Capacity vs Training Step

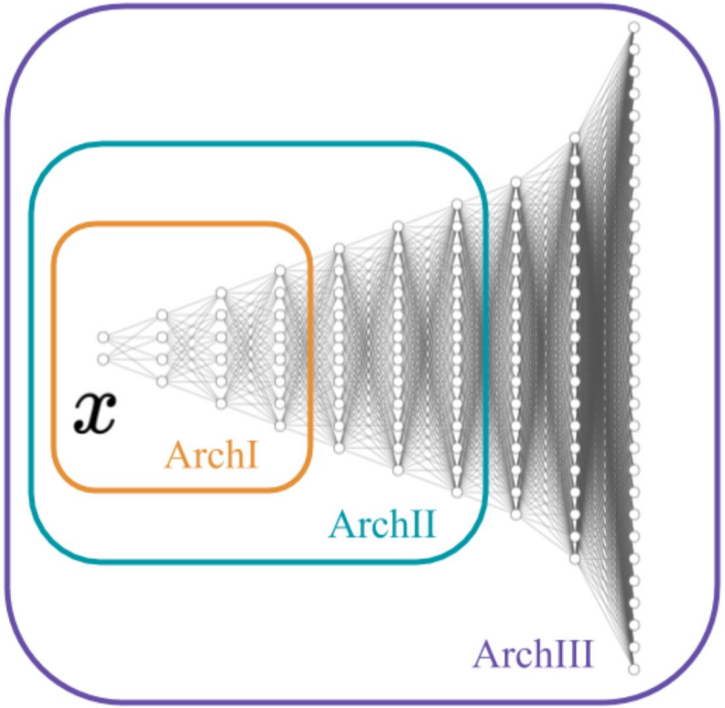


(b) Capacity vs Num Parameters

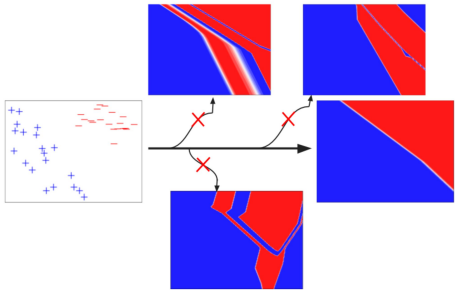
Linear Data : “Why does deep learning work”



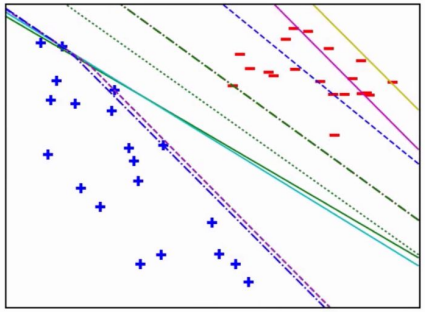
Looking closer: Neuron by Neuron

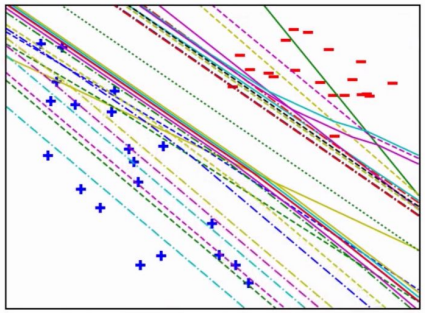


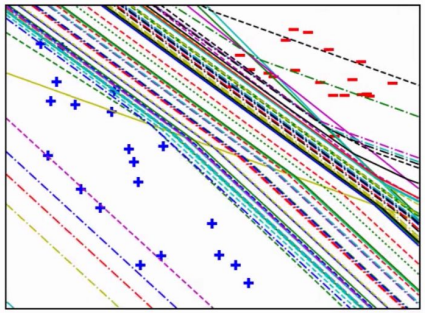
+



=

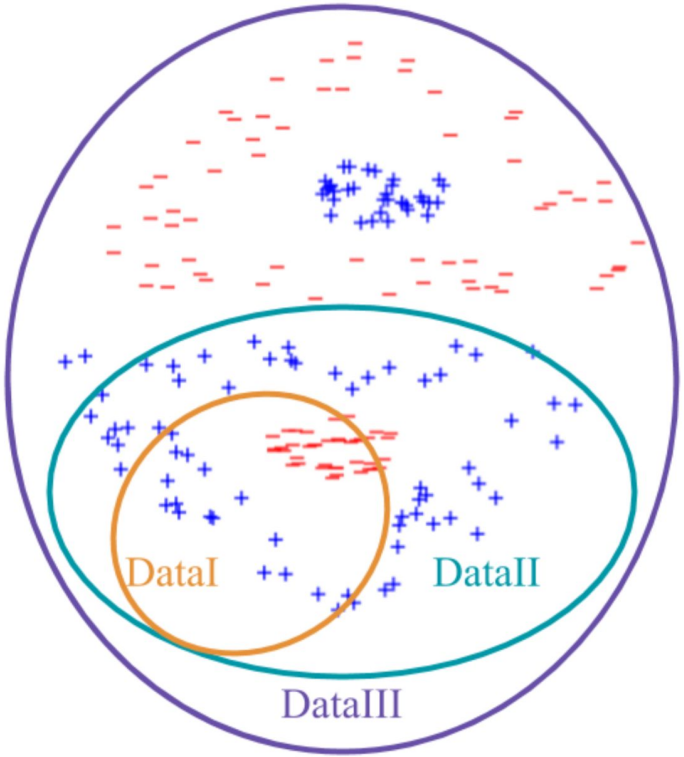




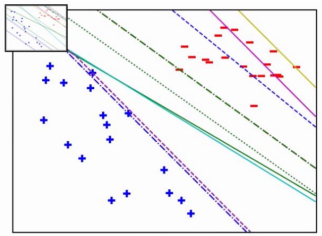


Regularity even has combinatorial description

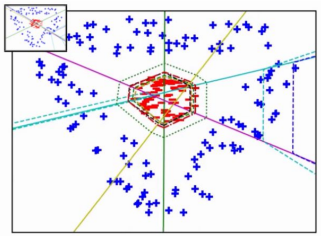
More complicated data



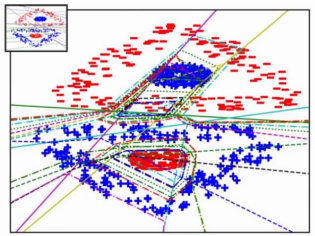
More complicated data -> Similar Conclusion



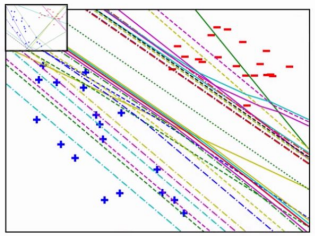
(a) DataIArchI



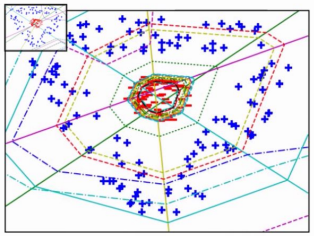
(b) DataIIArchI



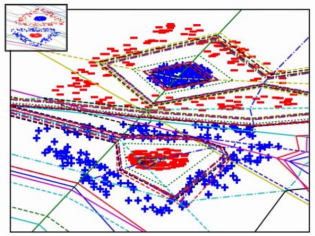
(c) DataIIIArchI



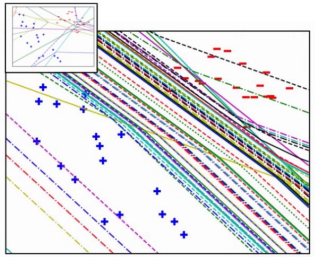
(d) DataIArchII

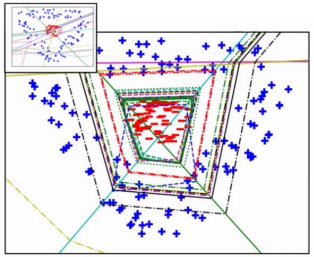


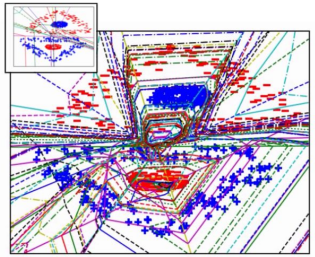
(e) DataIIArchII



(f) DataIIIArchII







How to Analyze Nonlinear Case

Theorem 3. *(Theorem 17 in [Goldberg and Jerrum, 1995]): Let* k,n *be positive integers and* $f : \mathbb{R}^n \times \mathbb{R}^k \mapsto \{0,1\}$ *be a function that can be expressed as a Boolean formula containing* s *distinct atomic predicates where each atomic predicate is a polynomial inequality or equality in* $k+n$ *variables of degree at most* d*. Let* $\mathcal{F} = \{f(\cdot, w) : w \in \mathbb{R}^k\}$*. Then* $VCDim(\mathcal{F}) \leq 2k \log_2(8eds)$.

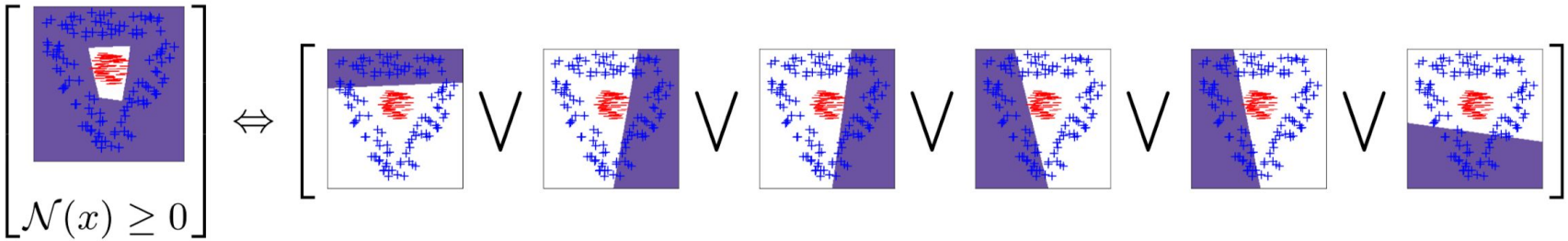
-> Need to express network as boolean formula
(Complexity doesn't depend on depth)

How to Analyze Nonlinear Case

Theorem 3. *(Theorem 17 in [Goldberg and Jerrum, 1995]): Let k,n be positive integers and $f : \mathbb{R}^n \times \mathbb{R}^k \mapsto \{0,1\}$ be a function that can be expressed as a Boolean formula containing s distinct atomic predicates where each atomic predicate is a polynomial inequality or equality in $k+n$ variables of degree at most d. Let $\mathcal{F} = \{f(\cdot, w) : w \in \mathbb{R}^k\}$. Then VCDim$(\mathcal{F}) \leq 2k \log_2(8eds)$.*

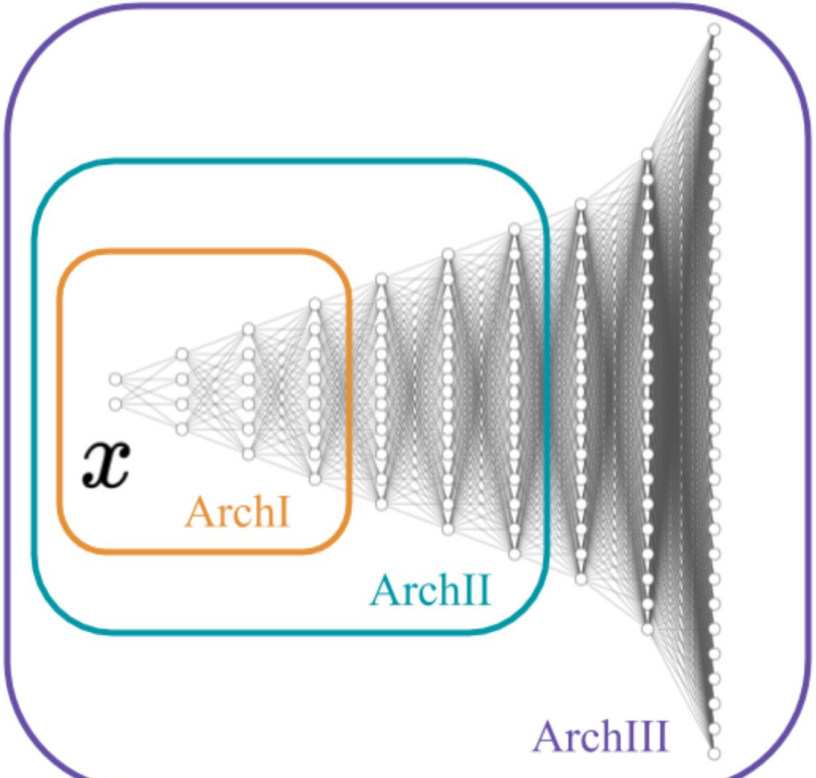
-> Need to express network as boolean formula
(Complexity doesn't depend on depth)

Example of transformation Output (7e5 param)



Setting

Fully connected networks -> control network complexity



Setting

Fully connected networks -> control network complexity

Synthetic Data -> Control data complexity