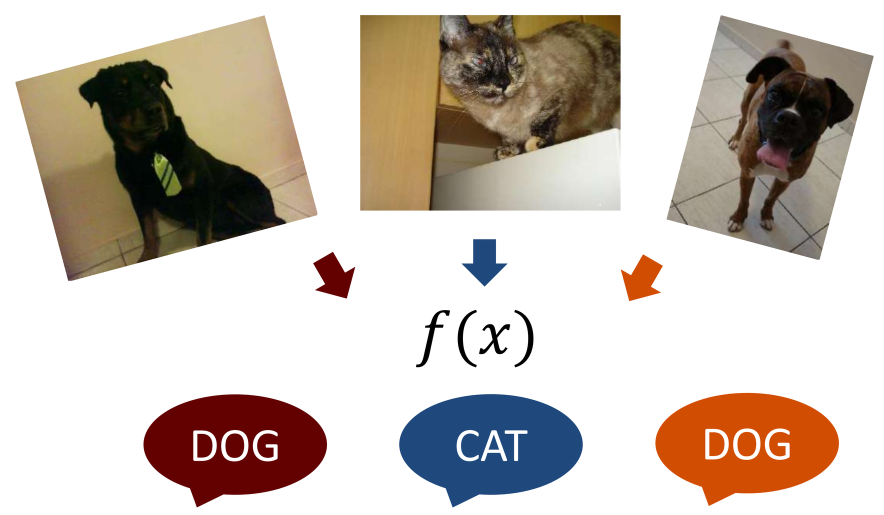
Bounding and Counting Linear Regions of Deep Neural Networks

Carnegie Mellon Tepper School of Business
The University of Utah

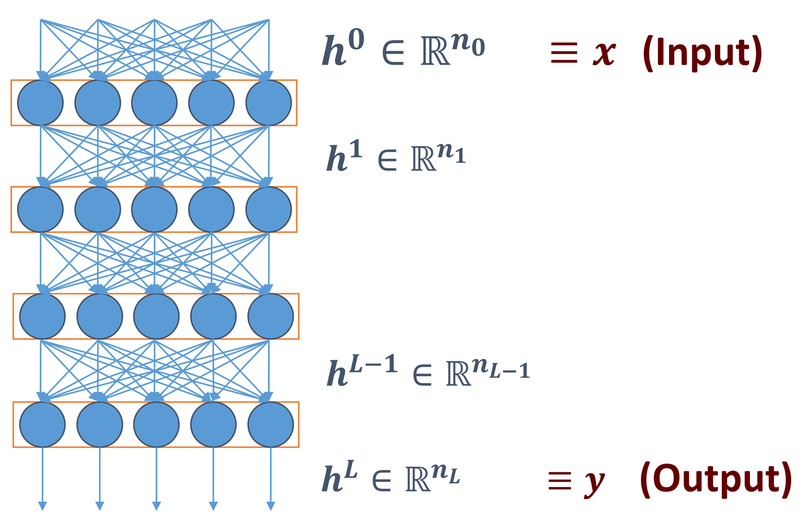
Thiago Serra
Christian Tjandraatmadja
Srikumar Ramalingam

1. Piecewise-linear Networks

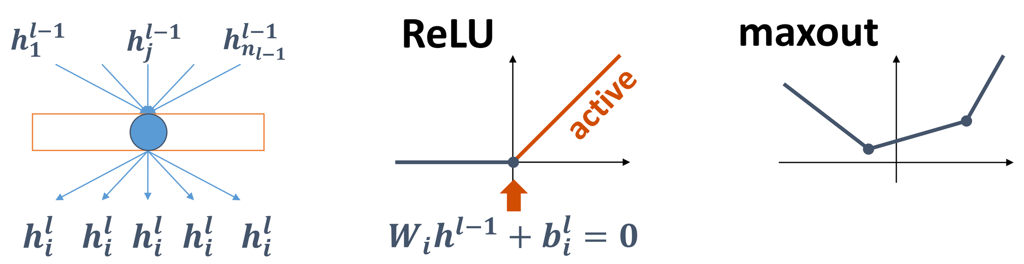
Deep learning has been successful in modeling and predicting features in varied domains.



In the case of feedforward networks, the building blocks are layers of units that transform the output from previous layers.



Each unit has its activation function mapping those inputs to a scalar output. In many cases, those are piecewise-linear functions.



Here we mainly consider rectifier networks, made of Rectifier Linear Units (ReLUs):

$$h_i^l = \max\{0, W_i^l h^{l-1} + b_i^l\}$$

But we also look into maxout networks:

$$h_i^l = \max\{W_i^{1l} h^{l-1} + b_i^{1l}, \ldots, W_i^{kl} h^{l-1} + b_i^{kl}\}$$

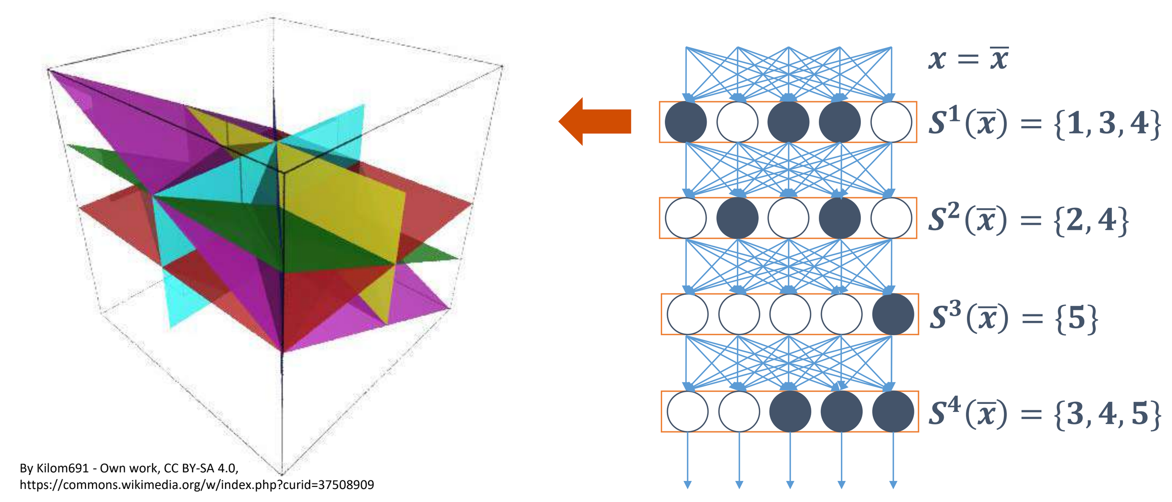
where W and b are weight and bias terms.

In this work, we study in theory and practice how many pieces — or linear regions — can be defined by different network configurations, in terms of input dimension n_0 and layer width n_l.

The number of linear regions is conjectured to be a proxy for the network expressiveness.

2. Hyperplane Arrangements and Activation Patterns

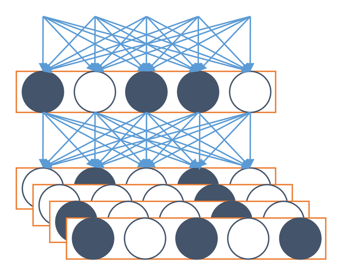
Each linear region is defined by an activation pattern, a vector that indicates which units are active. Each layer partitions its input through the arrangement of the activation hyperplanes.



Zaslavsky (1975): The number of regions in d dimensions defined by n hyperplanes is at most

$$\sum_{i=0}^{d} \binom{n}{i}$$

Each linear region up to a given layer is then similarly partitioned by subsequent layers.



3. Bounding through Dimensionality

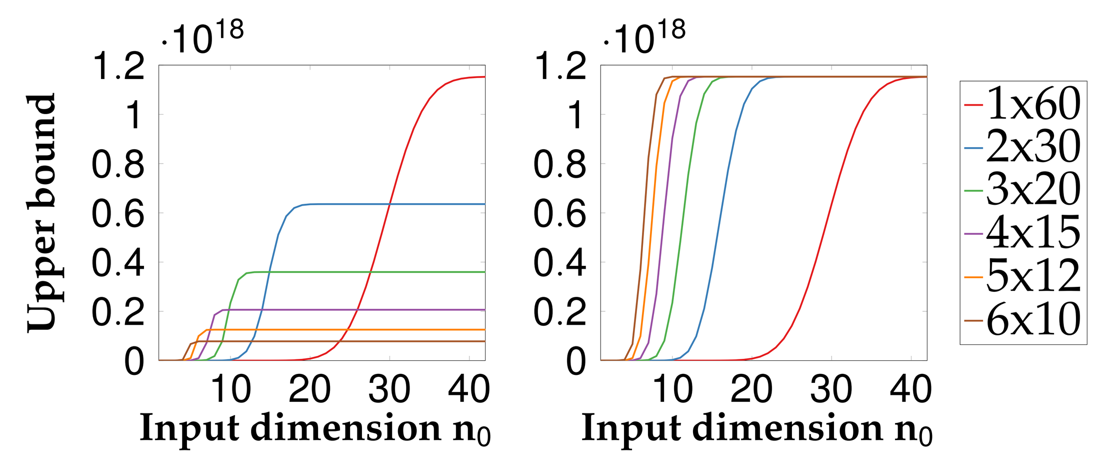
Fixing activations in the first l layers, we have a linear transformation T from x to h^l. The image of T has dimension bounded by $|S^1|, \ldots, |S^l|$.

Theorem 1 The maximal number of regions of a rectifier network is at most

$$\sum_{(j_1,\ldots,j_L)\in J} \prod_{l=1}^{L} \binom{n_l}{j_l}$$

where $J = \{(j_1, \ldots, j_L) \in \mathbb{Z}^L : 0 \leq j_l \leq \min\{n_0, n_1 - j_1, \ldots, n_{l-1} - j_{l-1}, n_l\} \ \forall l = 1, \ldots, L\}$. This bound is tight when $L = 1$.

Deep vs. shallow trade-off Since the bound is exact for shallow networks, they can attain more linear regions than deep networks with the same number of units if n_0 is sufficiently large; the opposite is known for small n_0.



(a) Theorem 1 (b) Montúfar (2017)

In the case of a rank-k maxout, where k is the number of arguments of max, we can use the minimum width across previous layers:

Theorem 2 The maximal number of regions of a maxout network is at most

$$\prod_{l=1}^{L} \sum_{j=0}^{d_l} \binom{\frac{k(k-1)}{2} n_l}{j}$$

where $d_l = \min\{n_0, n_1, \ldots, n_l\}$.

4. Counting as Mixed-Integer Programming Solutions

Assuming that inputs are bounded ($x \in X$), the following constraints map the input h^{l-1} to the output h_i^l for a sufficiently large constant M:

$$W_i^l h^{l-1} + b_i^l = h_i^l - \bar{h}_i^l \quad (1)$$
$$h_i^l \leq M z_i^l \quad (2)$$
$$\bar{h}_i^l \leq M(1 - z_i^l) \quad (3)$$
$$h_i^l \geq 0 \quad (4)$$
$$\bar{h}_i^l \geq 0 \quad (5)$$
$$z_i^l \in \{0, 1\} \quad (6)$$

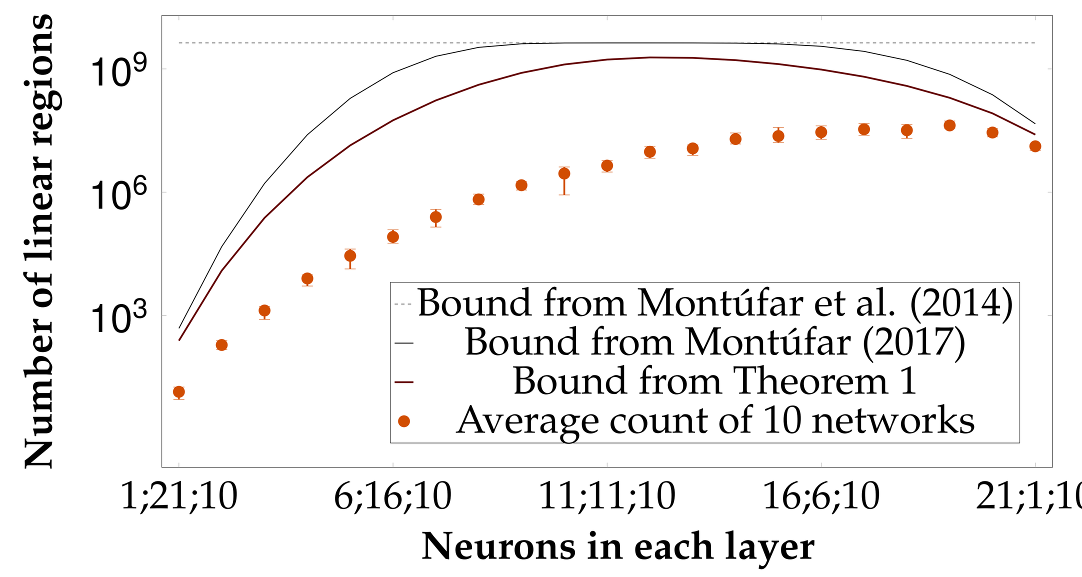
We want to consider a neuron active ($z_i^l = 1$) only when its output is strictly positive. We do that by maximizing the minimum output f of an active neuron, which is positive in the non-degenerate cases that we want to count:

$$\begin{aligned} \max \quad & f \\ \text{s.t.} \quad & (1)\text{–}(6) && \forall \text{ neuron } i \text{ in layer } l \\ & f \leq h_i^l + (1 - z_i^l)M && \forall \text{ neuron } i \text{ in layer } l \\ & x \in X \end{aligned}$$

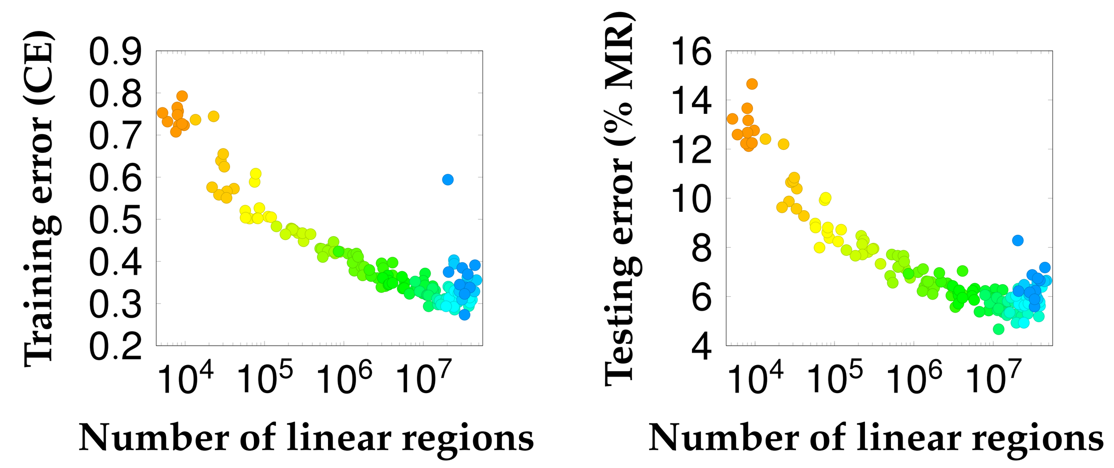
We count the distinct solutions on z with $f > 0$.

5. Results

We count linear regions in 10 rectifier networks for the MNIST digit recognition task with each configuration of two hidden layers totaling 22 units. We compare average and min-max range with the first bound, the latest, and ours.



In networks where no layer is too narrow, we found evidence that training and test errors relate to the number of linear regions. From red to blue, we plot below each of these networks by increasing number of units in the first layer.



In summary, we have:

1. a tighter upper bound for rectifier networks, which is exact for unidimensional input;
2. an upper bound for deep maxout neworks;
3. a first method to count linear regions;
4. initial evidence that the number of linear regions relates to network accuracy; and
5. insights on obtaining more linear regions.

Our Paper

Serra, T.; Tjandraatmadja, C.; Ramalingam, S., **"Bounding and Counting Linear Regions of Deep Neural Networks"**, *International Conference on Machine Learning* (ICML), 2018.

tinyurl.com/bcicml