Max Planck Institute for Mathematics in the Sciences

On the Number of Linear Regions of Deep Neural Networks

Guido Montúfar[1] **Razvan Pascau**[2] **Kyunghyun Cho**[2] **Yoshua Bengio**[2,3]

[1]Max Planck Institute for Mathematics in the Sciences [2]Université de Montréal [3]CIFAR Senior Fellow

Université de Montréal

Why is depth important

— Single layer MLPs are universal approximators.

— Deep models can be hard to train.

— Deep models are harder to inspect.

+ Wealth of empirical evidence for deep models performing better.

+ Intuitions on the brain and thought process.

+ Some theoretical results on the efficiency of deep models.

This Work

- We evaluate the complexity of functions computable by shallow and deep feedforward neural networks with piecewise linear activations in terms of symmetries and the number of linear regions.
- We show that **deep networks are exponentially more efficient** in terms of the number of linear regions.

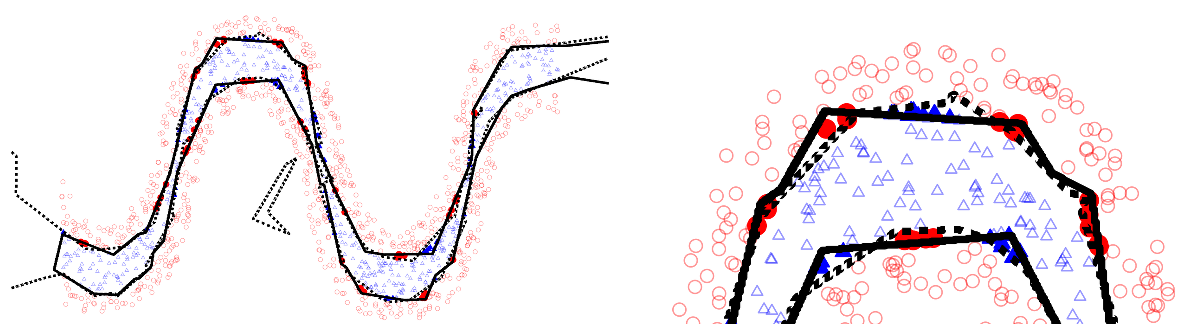


Figure. Decision boundary of rectifier MLPs and the number of linear regions.

Compositional Properties of Feedforward Neural Networks

- A map F **identifies** two input neighborhoods S and T if it maps them to the same output, $F(S) = F(T)$.
- Deep networks produce a **recursive identification** of input neighborhoods.

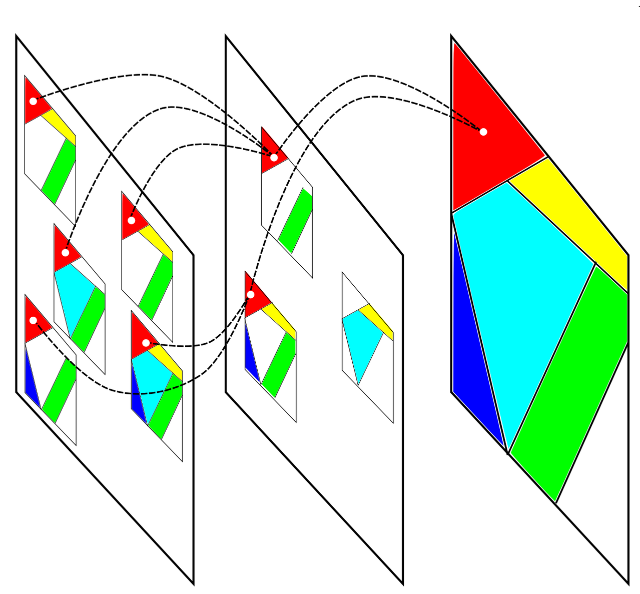


Figure. Recursive identification of input neighborhoods.

- Intuitively, each layer **folds** its input space onto itself.

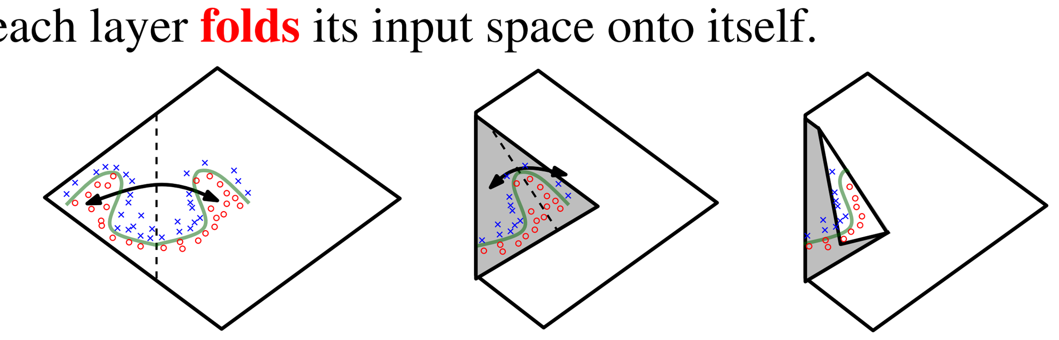


Figure. Identification of input neighborhoods as folding.

- Computations carried out at a given layer apply to all inputs that have been identified by the previous layers.

Deep Rectifier Networks

Theorem 1.

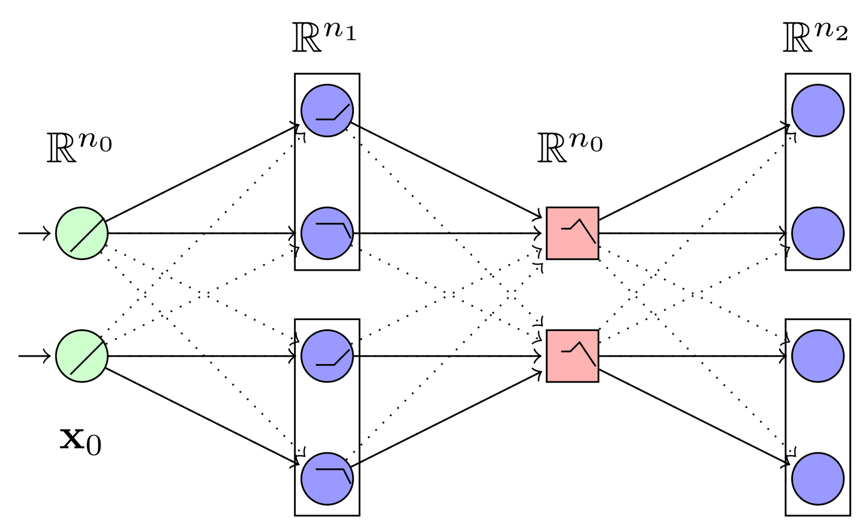
- *A* **shallow** *rectifier neural network with m units can compute functions with at most this many linear regions:*

$$O(m^{n_0}) \qquad (\textbf{polynomial} \textit{ in } m).$$

- *A* **deep** *rectifier network with L layers of n units each can compute functions with this many linear regions:*

$$O\left(\left(\frac{n}{n_0}\right)^{n_0(L-1)} n^{n_0}\right) \qquad (\textbf{exponential} \textit{ in the depth } L).$$

Proof Sketch.



$$\mathbf{h}_1 = \mathrm{rect}(\mathbf{W}_1\mathbf{x}_0 + \mathbf{b}_1) \quad \mathbf{h}_2 = \mathrm{rect}(\mathbf{W}_2\mathbf{h}_1 + \mathbf{b}_2)$$

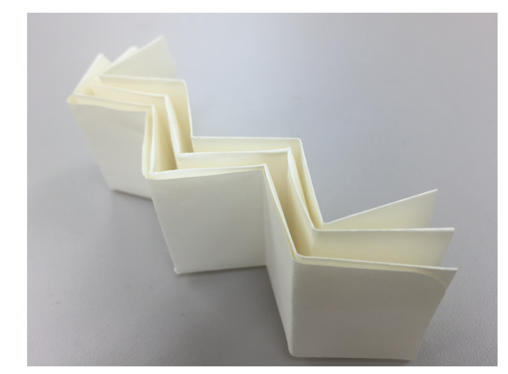


Figure. A layer of rectifier units can fold its input space like a zigzag.

Deep Maxout Networks

Theorem 2.

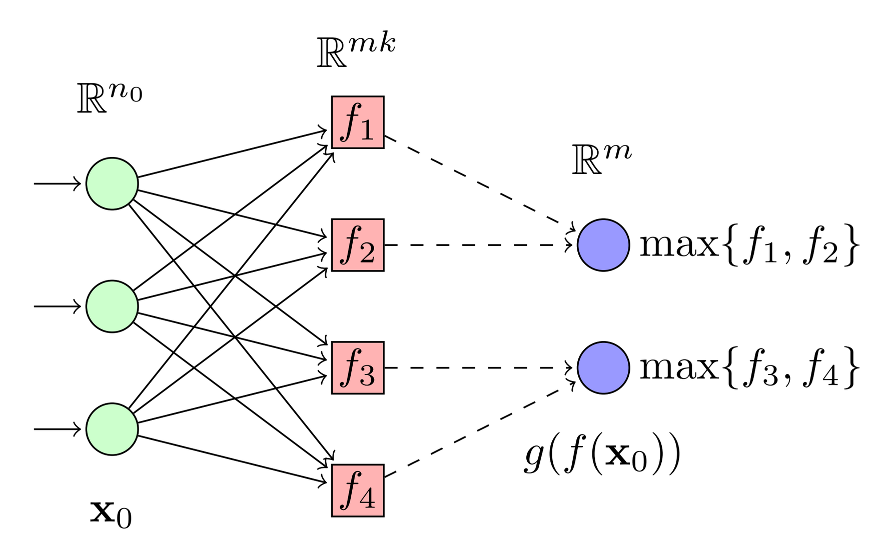
- *A* **shallow** *maxout network with m rank-k units can compute functions with at most this many linear regions:*

$$O\left(m^{n_0} k^{2n_0}\right) \qquad (\textbf{polynomial} \textit{ in } m).$$

- *A* **deep** *maxout network with L layers of n_0 rank-k units can compute functions with this many linear regions:*

$$O\left(k^{(L-1)+n_0}\right) \qquad (\textbf{exponential} \textit{ in the depth } L).$$

Proof Sketch.



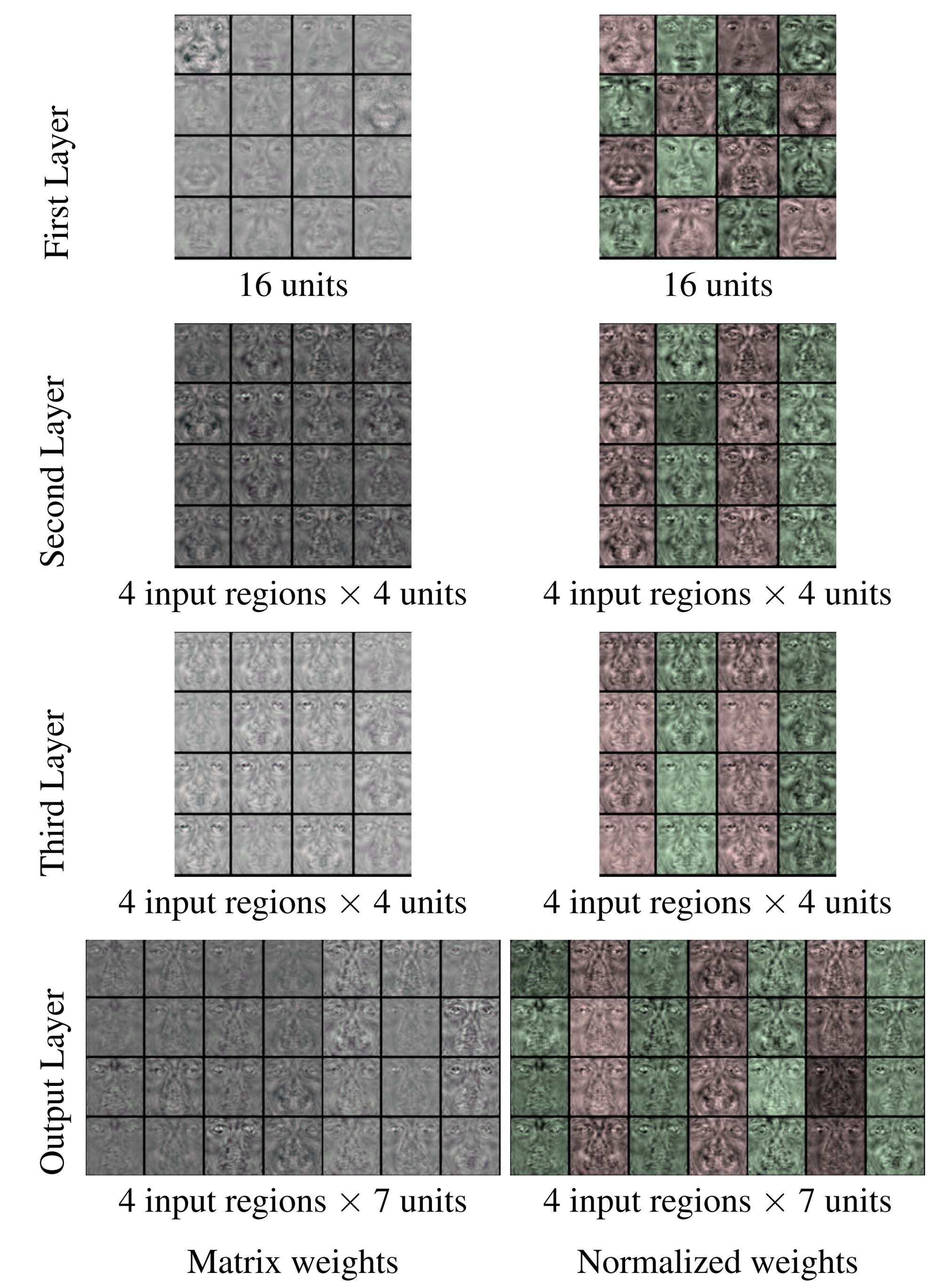
$$f(\mathbf{x}_0) = \mathbf{W}\mathbf{x}_0 + \mathbf{b}$$

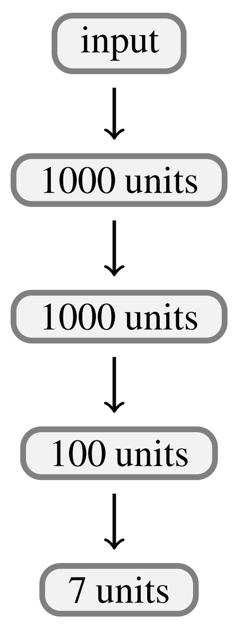
Figure. A layer of maxout units can fold its input space like a folding fan.

Visualizing the Behavior of Hidden Units in Deep Layers

- Behavior of hidden units of a rectifier MLP trained on the Toronto Faces Dataset.
- The map taking input images to activations of the individual units is piecewise linear. For an input image I the activation of the i-th unit at the l-th layer is

$$I \mapsto h_{l,i} = M_{l,i}^{(I)} \cdot I + B_{l,i}^{(I)}.$$





The network architecture

Three inputs and respective matrix weights of the 48th unit in the 3rd hidden layer producing the same activation of that unit

Figure. Matrix weights $M_{l,i}^{(I)}$ of the linear maps "input image" → "unit activation" computed by a rectifier MLP for various inputs I and units i in different layers l. Deep units react equally to faces from diverse input neighborhoods.

Conclusions

- Each layer is able to identify distinct regions of its input space; the composition of layers is able to identify an exponential number of regions.
- Exponential replication of the complexity of functions computed in deeper layers.
- Deep networks can compute complicated functions with intrinsic rigidity caused by replications; this may help generalizing to unseen samples.
- The presented framework applicable to many kinds of networks.

References

[1] O. Delalleau and Y. Bengio. Shallow vs. deep sum-product networks. In *NIPS*, 2011.
[2] X. Glorot, A. Bordes, and Y. Bengio. Deep sparse rectifier neural networks. In *AISTATS*, 2011.
[3] I. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio. Maxout networks. In *Proc. 30th ICML*, 2013.
[4] A. Krizhevsky, I. Sutskever, and G. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*. 2012.
[5] V. Nair and G. E. Hinton. Rectified linear units improve RBMs. In *Proc. 27th ICML*, 2010.
[6] R. Pascanu, G. Montúfar, and Y. Bengio. On the number of response regions of deep feed forward networks with piece-wise linear activations. *arXiv:*1312.6098, 2013.
[7] J. Susskind, A. Anderson, and G. E. Hinton. The Toronto face dataset. UTML TR 2010-001, U. Toronto, 2010.
[8] T. Zaslavsky. *Facing Up to Arrangements: Face-Count Formulas for Partitions of Space by Hyperplanes*. AMS, 1975.
[9] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. *arXiv:*1311.2901, 2013.

Neural Information Processing Systems Foundation

Neural Information Processing Systems, Palais des Congrès de Montréal / Convention and Exhibition Center, Montreal, Quebec, Canada, December 8–13, 2014.