Lecture 2. Neural Networks : What can a network represent

R11-785 Introduction to Deep Learning

TAVE Research DL001
Changdae Oh

2021. 01. 10

Contents

Contents

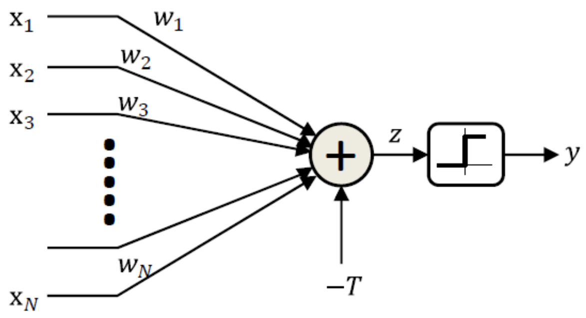
Definition of Neural Networks & Perceptron

Neural Networks

- Functions that take an input and produce an output
- Composed of networks of computational models of neurons called **perceptron**

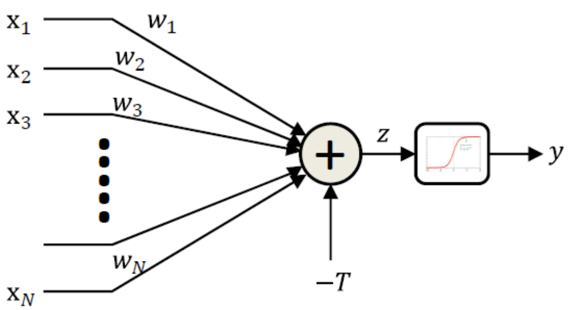
Perceptron

- So called, "Threshold gate"
- Fires if the weighted sum of inputs exceeds a threshold



$$z = \sum_i w_i x_i - T$$

$$y = \begin{cases} 1 \; if \; z \geq 0 \\ 0 \; else \end{cases}$$



$$z = \sum_i w_i x_i - T$$

$$y = \frac{1}{1 + exp(-z)}$$

1) Calculate Affine combination of inputs

2) Transmitted to activation function

Can use smoothed version of threshold
- Sigmoid / tan h (Soft perceptron)

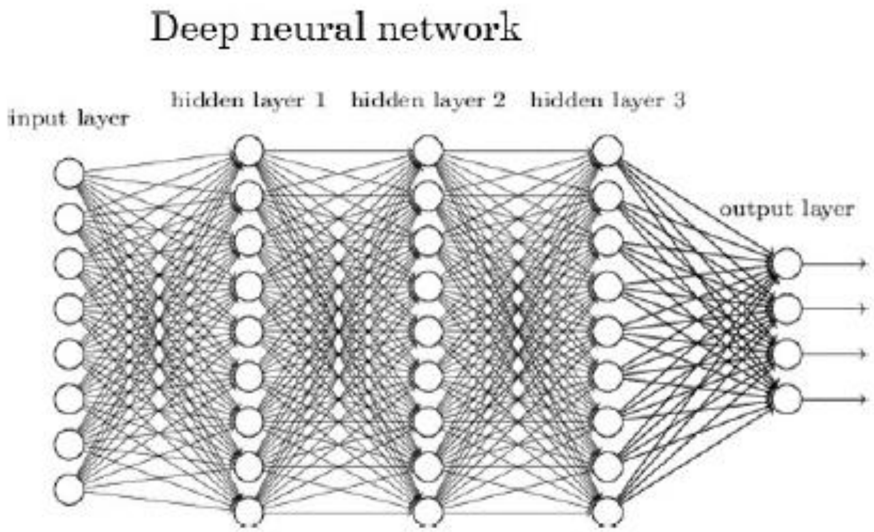
Or other continuous activation function
- ReLU

Definition of Neural Networks & Perceptron

Multi-Layer Perceptron, MLP

A network of perceptrons those "feed" others

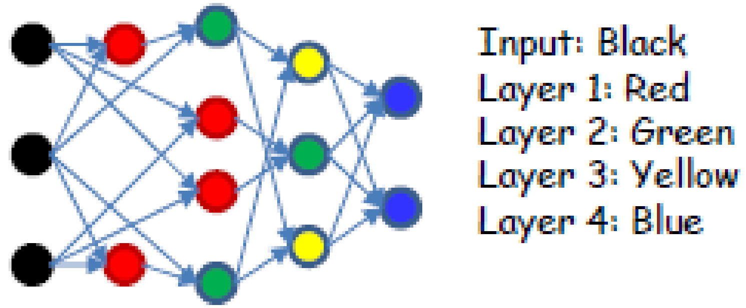
- Inputs are real or Boolean stimuli
- Outputs are real or Boolean values
- It can model kinds of input/output relationships



Defining "depth" in Neural Nets

"depth" is **the length of the longest path** from a source to a sink

- A "source" node is a node that has only outgoing edges
- A "sink" node is a node that has only incoming edges



Layer is the set of neurons
that all have the same depth from the inputs

Contents

MLPs as universal Boolean functions

- A perceptron can model
  any simple binary Boolean gate
- Furthermore, it can also model more complex functions
  such as **universal AND/OR gate, generalized majority gate**
- However, one perceptron alone cannot calculate the XOR,
  and it can be done by adding 1 hidden layers and increasing nodes.

➡ **MLPs can compute any Boolean function**
– since they can emulate individual gates

➡ How many layers & neurons are needed for a Boolean MLP?

MLPs are universal Boolean functions !

– any function over any number of inputs and any number of outputs

MLPs as universal Boolean functions

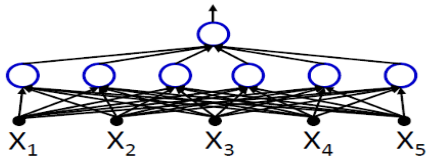
A Boolean function is just a truth table

– Any "true table" can be expressed in the following manner. (using only one hidden layer)

Truth Table

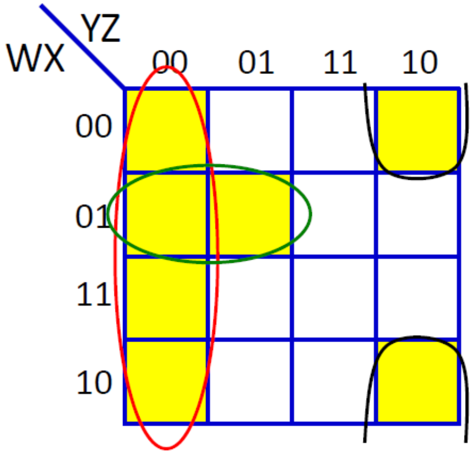
X_1	X_2	X_3	X_4	X_5	Y
0	0	1	1	0	1
0	1	0	1	1	1
0	1	1	0	0	1
1	0	0	0	1	1
1	0	1	1	1	1
1	1	0	0	1	1

$$Y = \bar{X}_1\bar{X}_2X_3X_4\bar{X}_5 + \bar{X}_1X_2\bar{X}_3X_4X_5 + \bar{X}_1X_2X_3\bar{X}_4\bar{X}_5 + X_1\bar{X}_2\bar{X}_3\bar{X}_4X_5 + X_1\bar{X}_2X_3X_4X_5 + X_1X_2\bar{X}_3\bar{X}_4X_5$$

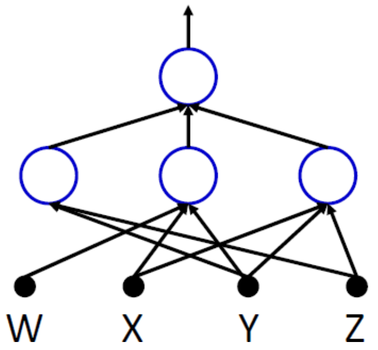


How many neurons are needed ?

Deriving Reduced DNF using "karnaugh map"



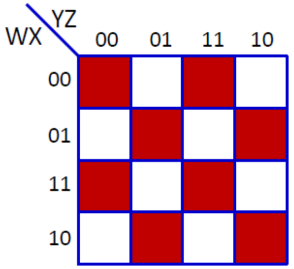
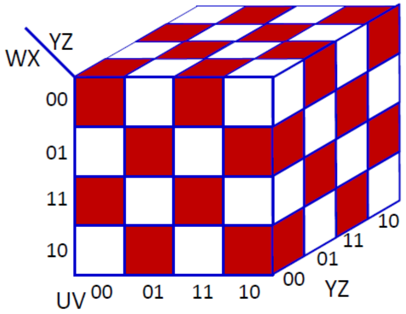
$$O = \bar{Y}\bar{Z} + \bar{W}X\bar{Y} + \bar{X}Y\bar{Z}$$



- 7(full model) -> 3(reduced model)
- Reduction of the DNF reduces the size of the one-hidden-layer network

MLPs as universal Boolean functions

Largest irreducible DNF (the worst situation to compute complexity)

	1-Hidden-layer network	MLP with XOR
	8 neurons	9 neurons
	32 neurons	15 neurons
N : # of inputs	$\boxed{2^{N-1} \text{ neurons}}$	$\boxed{3(N-1) \text{ neurons}}$
	Exponential in N	Linear in N

MLPs as universal Boolean functions

Summary

MLPs are universal Boolean functions !

- But can represent a given function only if
  - it is sufficiently **wide**
  - it is sufficiently deep
  - depth can be traded off for exponential growth of the width of the network
- Optimal width and depth depend on the number of variables and the complexity of the Boolean function
- To represent the same function, **the deeper** a network is, **the fewer** neurons are required

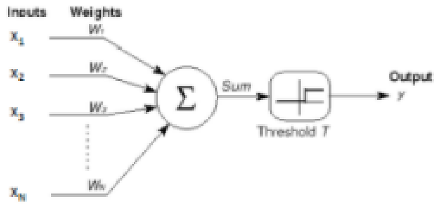
Contents

MLPs as universal Classifiers

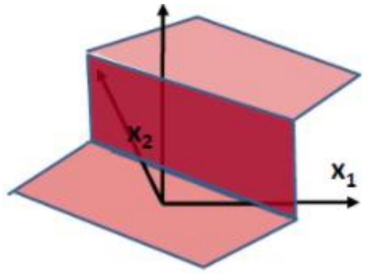
The MLP as a classifier

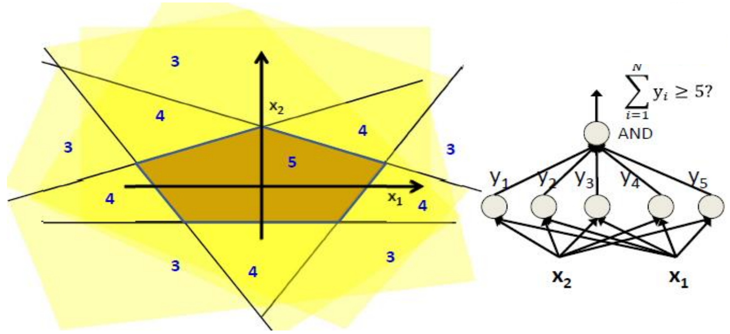
- One single perceptron can be a linear classifier
- MLP is a classifier that finds a complex "decision boundary" over a space of reals
- A one-hidden-layer MLP can model any classification boundary

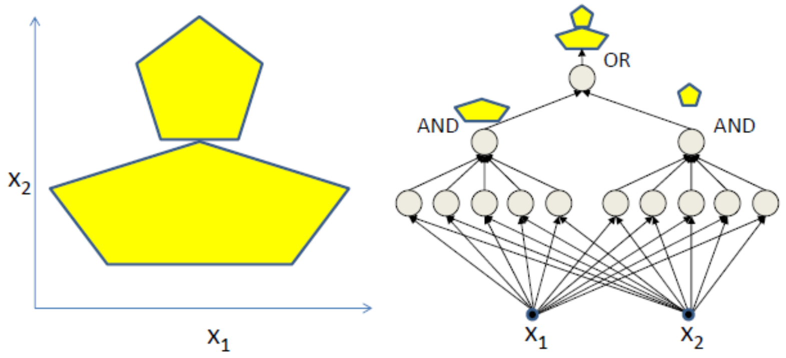
MLPs are universal classifiers!

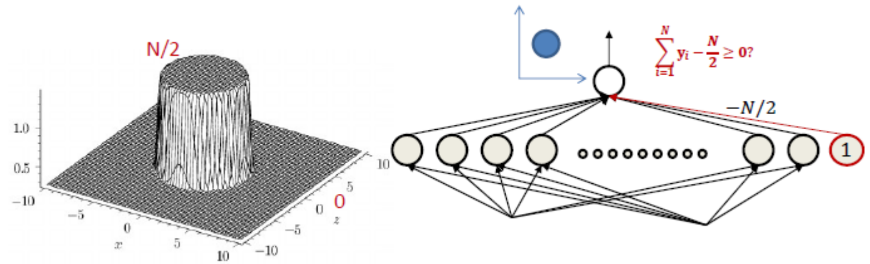


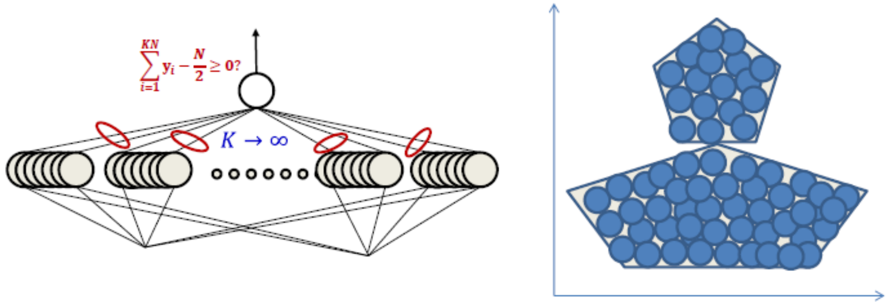
$$y = \begin{cases} 1 \; if \sum_i w_i x_i \ge T \\ 0 \; else \end{cases}$$



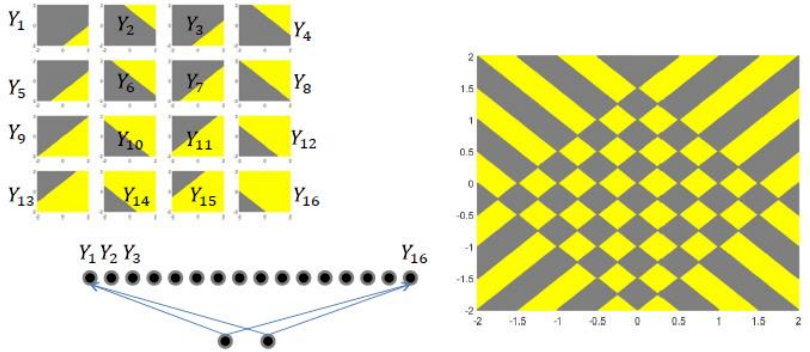


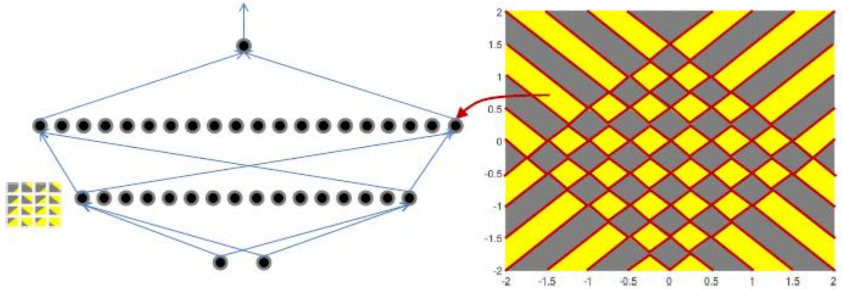






MLPs as universal Classifiers

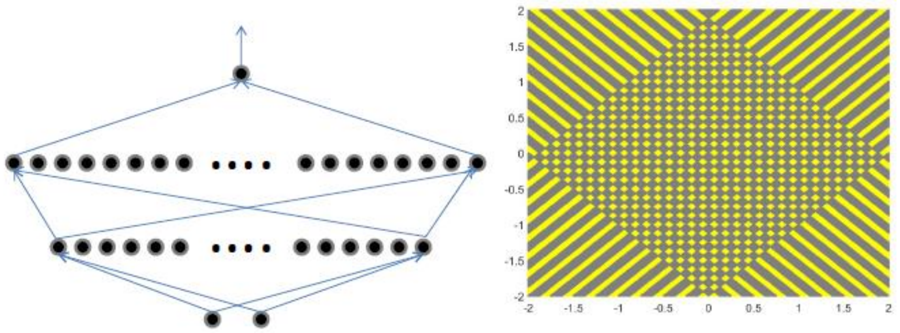




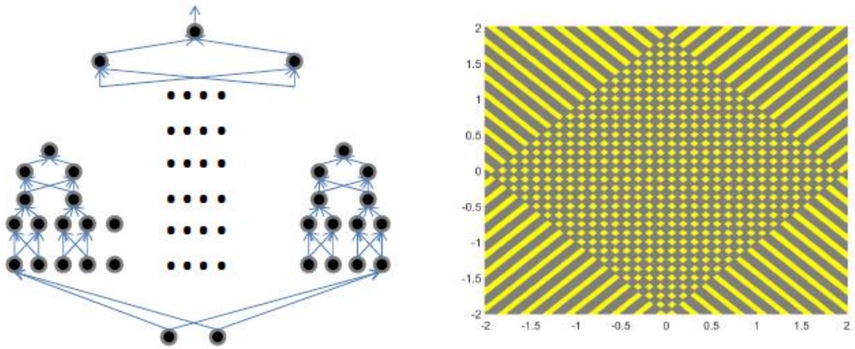
An infinite number of neurons are required to represent the above grid-shaped decision boundaries in a single hidden layer…

About more complex pattern

Shallow networks vs Deep networks



- Two hidden layers: 608 hidden neurons
  - 64 in layer 1
  - 544 in layer 2



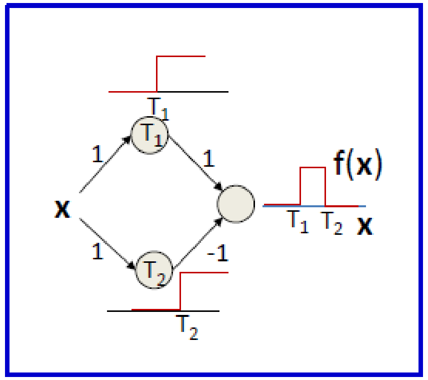
- XOR network (12 hidden layers): 253 neurons
  - 190 neurons with 2-gate XOR

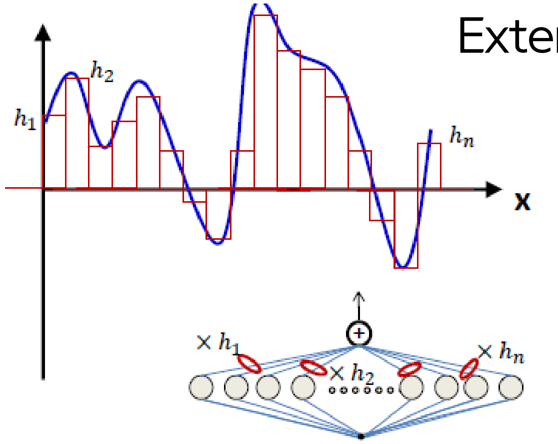
Efficient!!

Contents

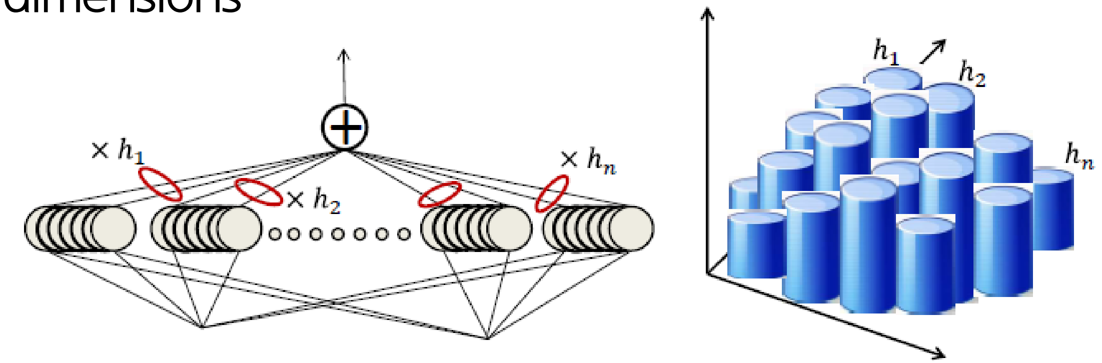
MLPs as universal approximators

MLP as a continuous-valued regression





Extensions to higher dimensions



- A simple 3-unit MLP can generate a "square pulse" over an input
- A MLP with many units can model an arbitrary functions in any number of dimensions

MLPs are universal approximators!

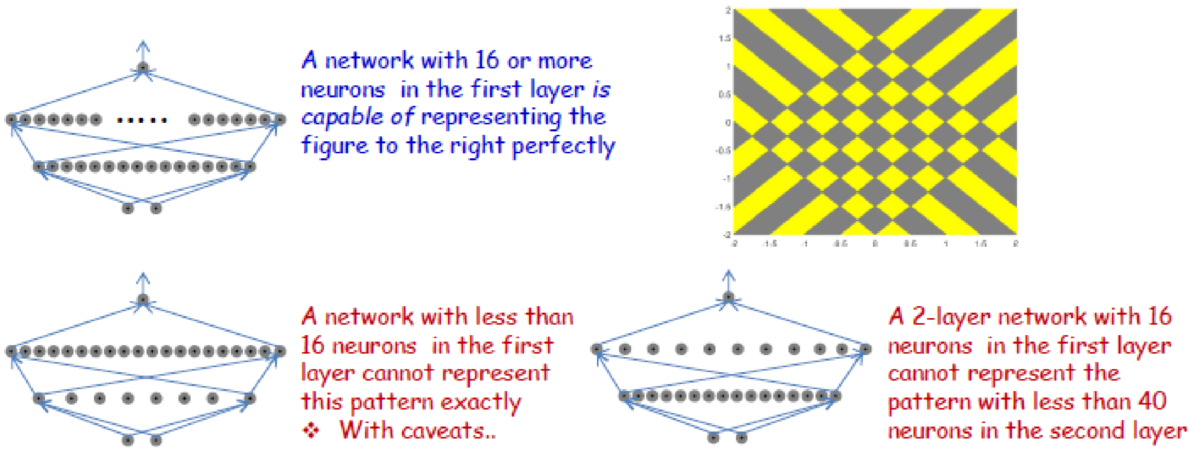
MLPs as universal approximators

Sufficiency of architecture

> A neural network can represent any function
> if it has sufficient capacity

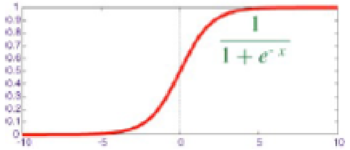
Capacity in NN

- How many patterns can it remember
- VC dimension
- Largest number of disconnected convex regions if can represent

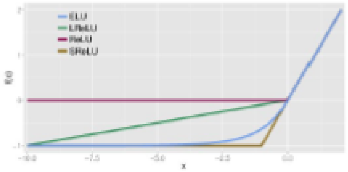


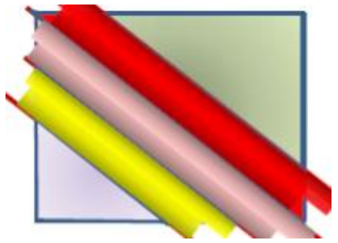
Continuous activation functions
result in graded output at the layer

S-Curve
(sigmoid, tan h)



Linear unit
(ReLU family)





To be more expressive

Summary

MLPs are universal Boolean function

MLPs are universal classifier

MLPs are universal function approximator

A single-layer MLP can approximate anything to arbitrary precision
– but could be exponentially or even infinitely wide in tis inputs size

Deeper MLPs can achieve the same precision with far fewer neuron
– Deeper networks are more expressive
– More graded activation functions result in more expressive networks