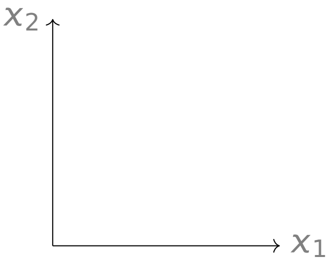
Deep Learning

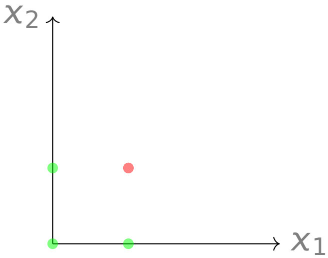
Lecture 5: Multilayer Perceptrons

Dr. Mehrdad Maleki

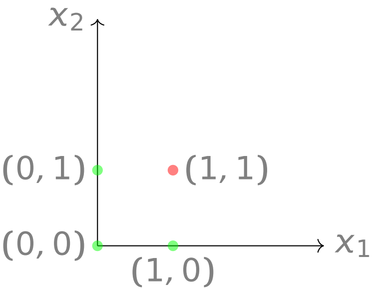
We have seen that $x_1 + x_2 \geq 2$ is the half-plane that recognize $x_1 \land x_2$ and this graph shows why,



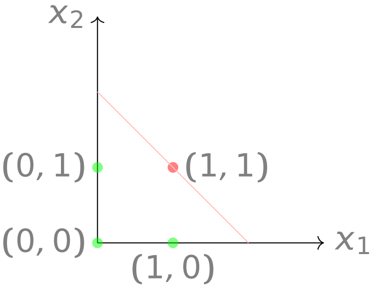
We have seen that $x_1 + x_2 \geq 2$ is the half-plane that recognize $x_1 \wedge x_2$ and this graph shows why,



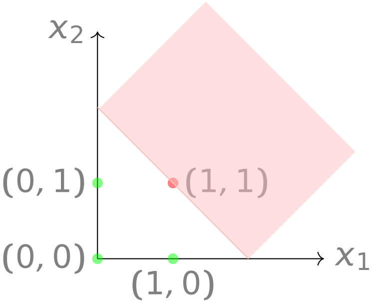
We have seen that $x_1 + x_2 \geq 2$ is the half-plane that recognize $x_1 \wedge x_2$ and this graph shows why,



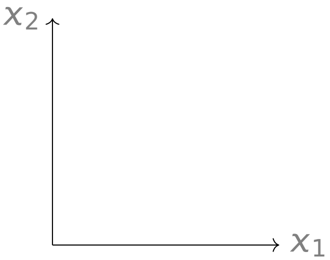
We have seen that $x_1 + x_2 \geq 2$ is the half-plane that recognize $x_1 \wedge x_2$ and this graph shows why,



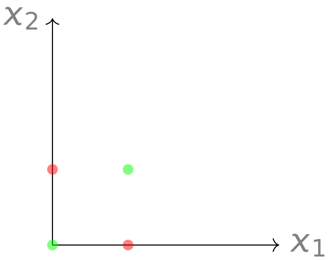
We have seen that $x_1 + x_2 \geq 2$ is the half-plane that recognize $x_1 \wedge x_2$ and this graph shows why,



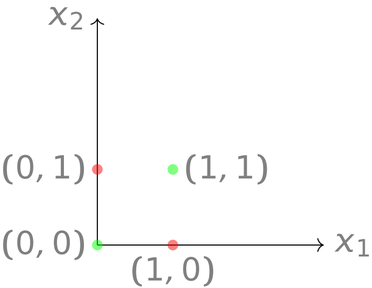
We have seen that we can't compute XOR with a single perceptron.



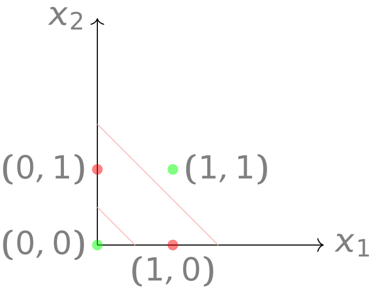
We have seen that we can't compute XOR with a single perceptron.



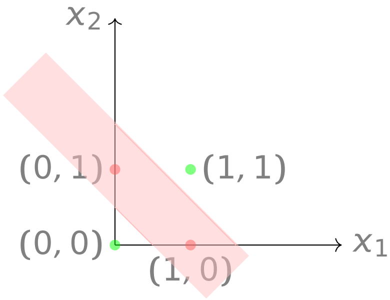
We have seen that we can't compute XOR with a single perceptron.



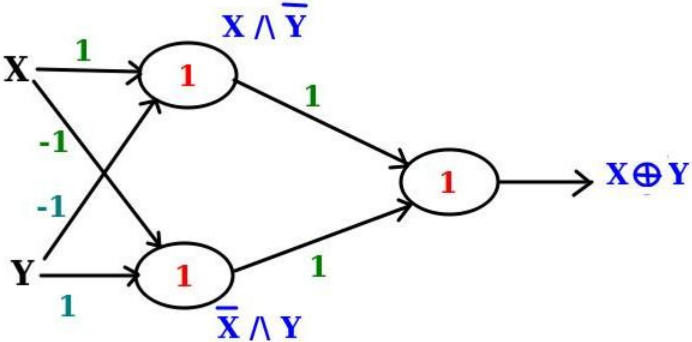
We have seen that we can't compute XOR with a single perceptron.



We have seen that we can't compute XOR with a single perceptron.



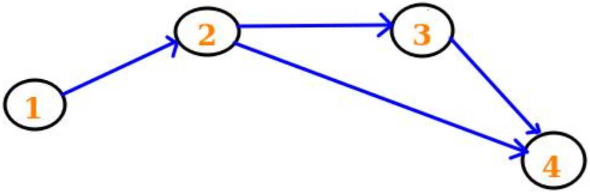
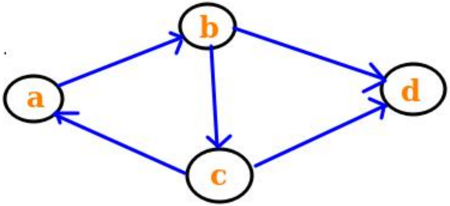
- But with two perceptrons we can compute XOR, because,
- $x \oplus y = (\overline{x} \wedge y) \vee (x \wedge \overline{y})$.



X
Y
1
-1
-1
1
1
1
1
1
1
$X \wedge \overline{Y}$
$\overline{X} \wedge Y$
$X \oplus Y$

Depth of a Graph

The length of the longest path in Directed Acyclic Graph (DAG) from the source to the sink is the *depth* of the graph.

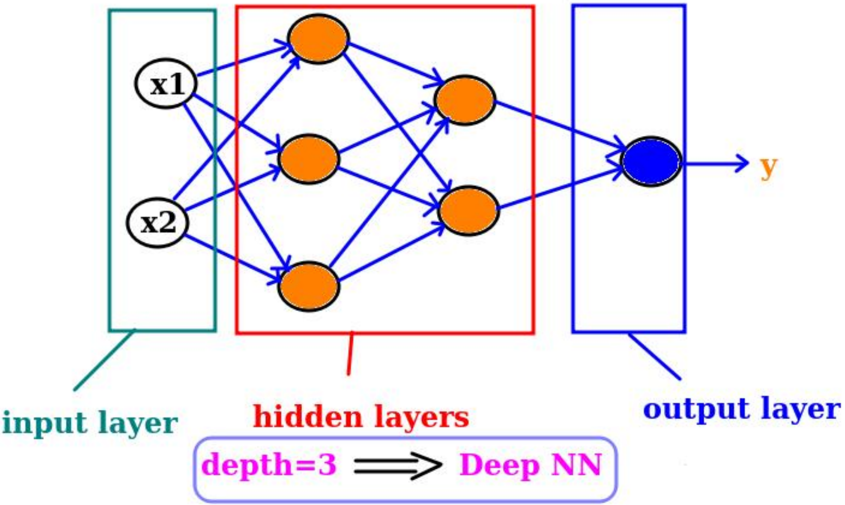

1
2
3
4

a
b
c
d

Feedforward Neural Network

A *Feedforward Neural Network* is a DAG of perceptrons. The depth of this DAG is the depth of the graph.

Deep Neural Network

A *Deep Neural Network* is a feedforward neural network of the depth at least 3.


x1
x2
y
input layer
hidden layers
output layer
depth=3
Deep NN

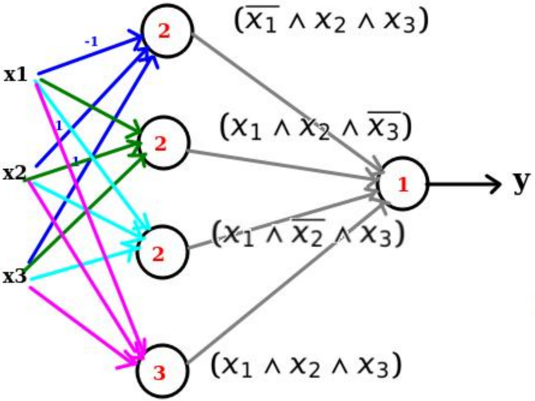
Universal Boolean Functions

Multi layer perceptrons can compute any Boolean function. We say multi layer perceptrons are *universal Boolean functions*. Any Boolean function can be computed by a multi layer perceptron with just one hidden layer.

With the truth table of a Boolean function we can obtain a multi layer perceptrons with just one layer as follow. Write the DNF (Disjunctive Normal Form- OR of AND clauses) of the truth table and build a single perceptron per each clause and then combine them with an OR gate.

x_1	x_2	x_3	y
0	1	1	1
1	1	0	1
1	0	1	1
1	1	1	1

$$y = (\overline{x_1} \wedge x_2 \wedge x_3) \vee (x_1 \wedge x_2 \wedge \overline{x_3}) \vee (x_1 \wedge \overline{x_2} \wedge x_3) \vee (x_1 \wedge x_2 \wedge x_3)$$

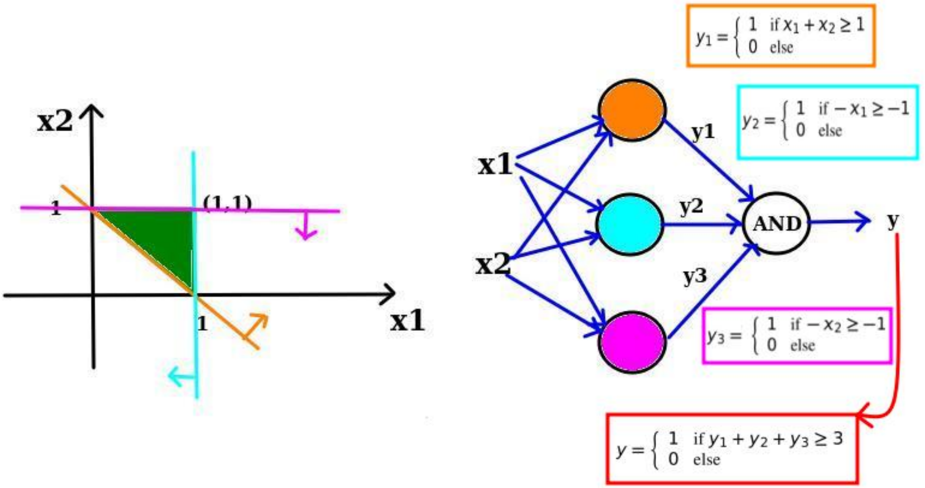


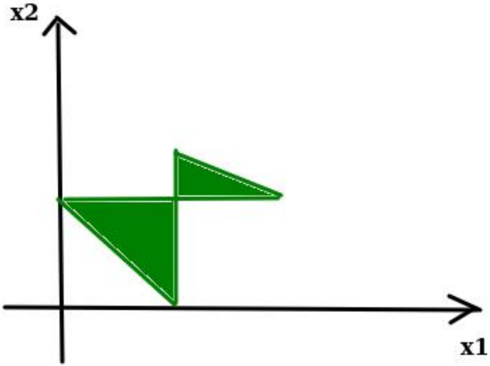
x1
x2
x3
-1
1
1
2
2
2
3
1
y
$(\overline{x_1} \wedge x_2 \wedge x_3)$
$(x_1 \wedge x_2 \wedge \overline{x_3})$
$(x_1 \wedge \overline{x_2} \wedge x_3)$
$(x_1 \wedge x_2 \wedge x_3)$

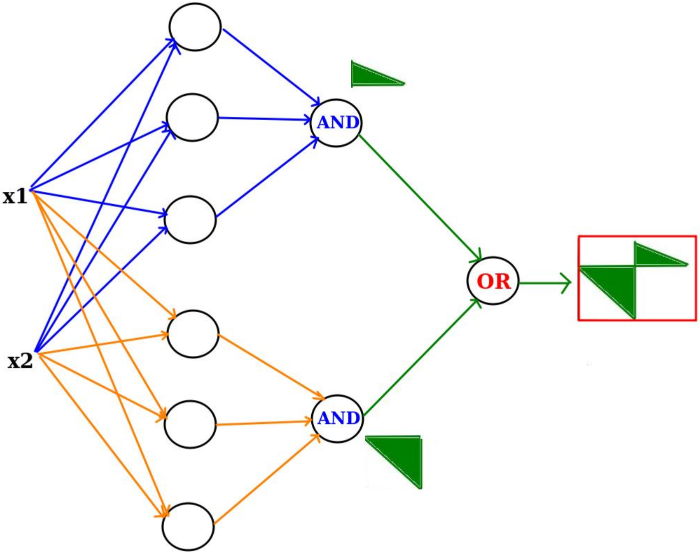
Real inputs

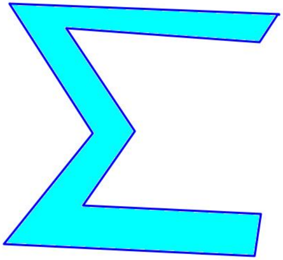
So far, the inputs was Boolean, i.e., `0, 1`. But the perceptrons can determine the linear classifier for the real-valued inputs. But how we can design a multi layer perceptron for decision boundary with the complex shapes?

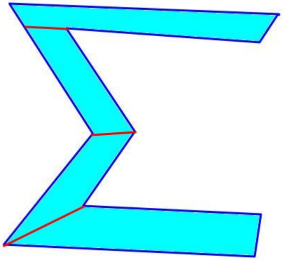
The function that inside the triangle is equal to 1 and outside of it is equal to 0.

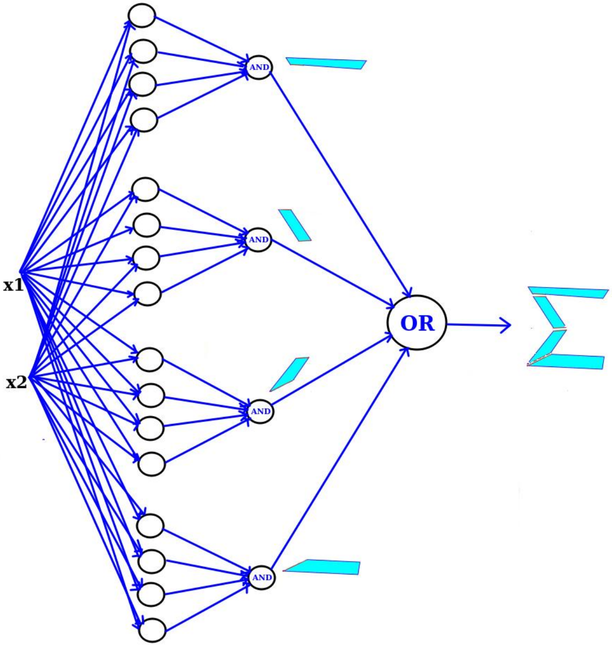



x2
x1

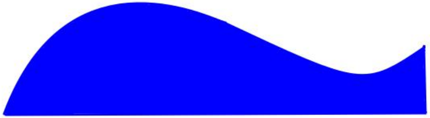

x1
x2
AND
AND
OR

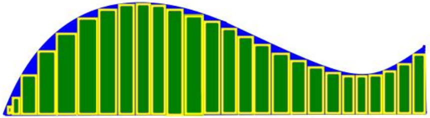


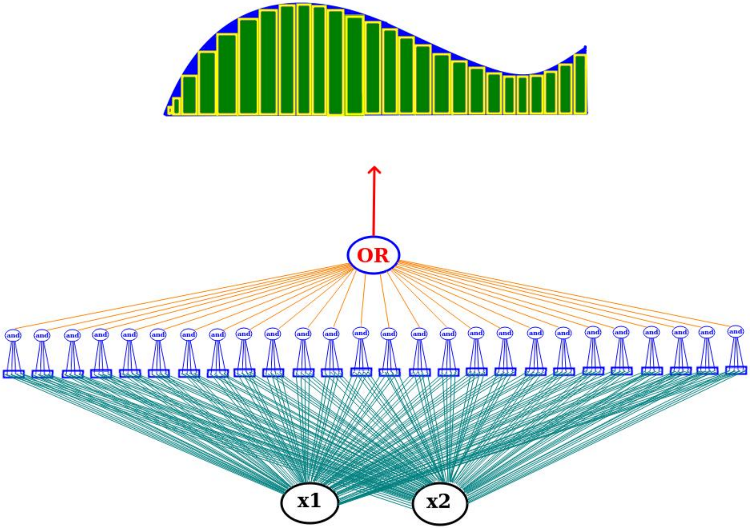




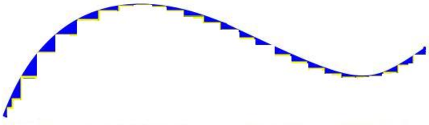
x1
x2
AND
AND
AND
AND
OR

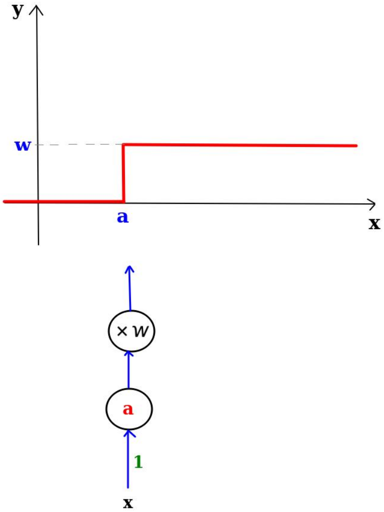


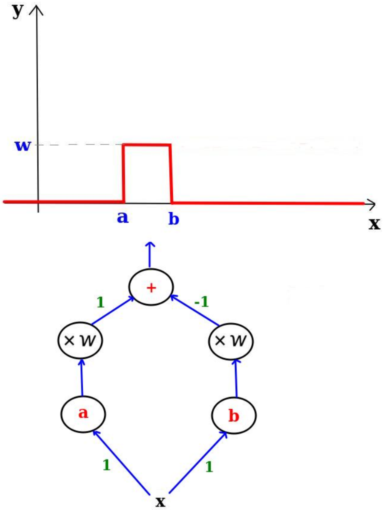


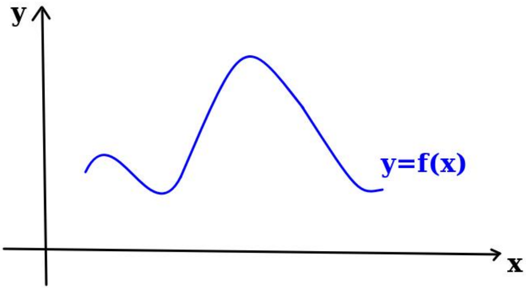


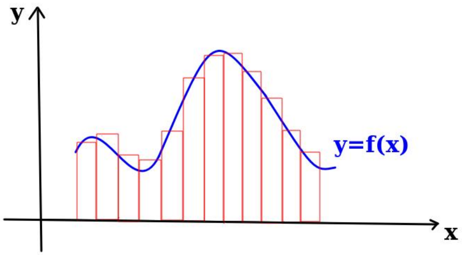
OR
and
x1
x2



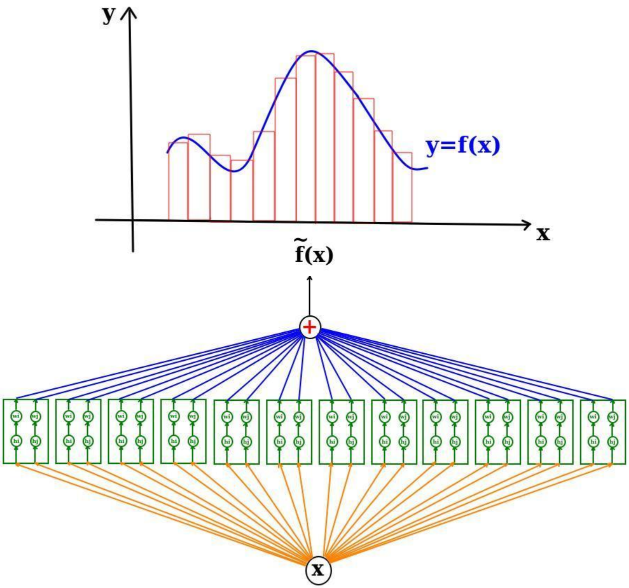

y
w
a
x
× w
a
1
x


y
w
a
b
x
+
1
-1
× w
× w
a
b
1
1
x


y
y=f(x)
x



y
y=f(x)
x



y
x
y=f(x)
$\tilde{f}(x)$
+
wi
wj
hi
hj
x

- Multi-layer Perceptrons are universal Boolean functions.
- Multi-layer Perceptron are universal classifiers.
- Multi-layer Perceptron are **universal approximators**.

Thank You