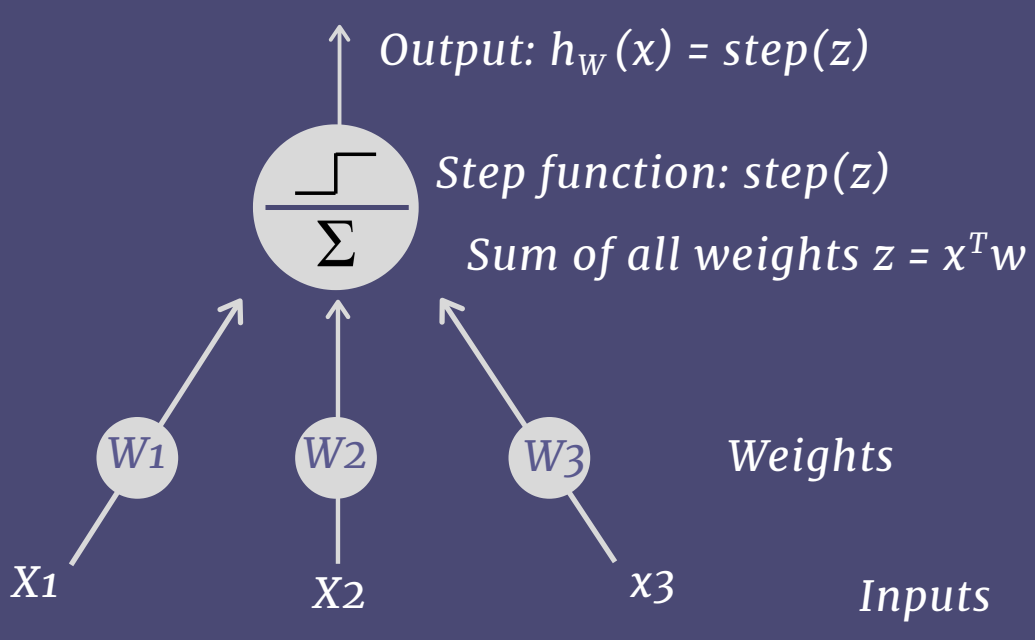
THE PERCEPTRON

ANTECEDENTS

SIMPLE ARTIFICIAL NEURAL NETWROK ARQUITECHTURE

Invented by Frank Rosenblatt in 1957, based on another artificial neuron called Treshold Logic Unit (TLU). The TLU computes a wieghted sum of its numerical inputs and applies a step function to that sum and outputs the result.

STRUCTURE OF A TLU



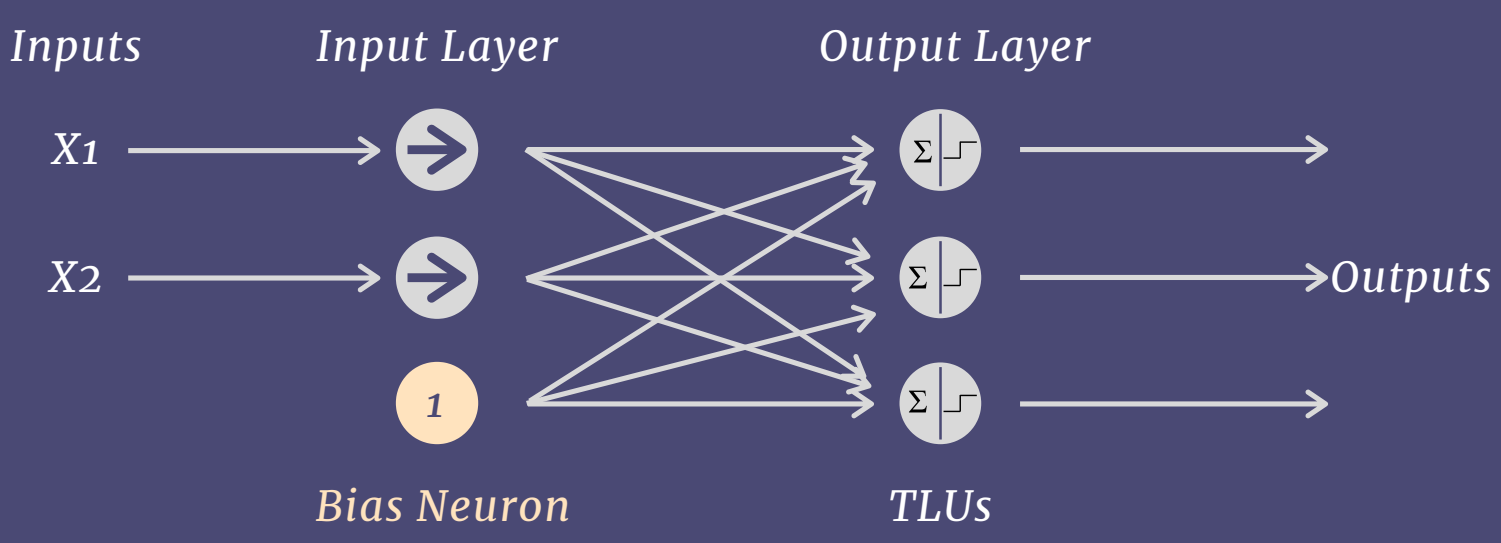
STEP FUCNTION OF A TLU

Common step functions used in Perceptrons (assuming threshold =0)

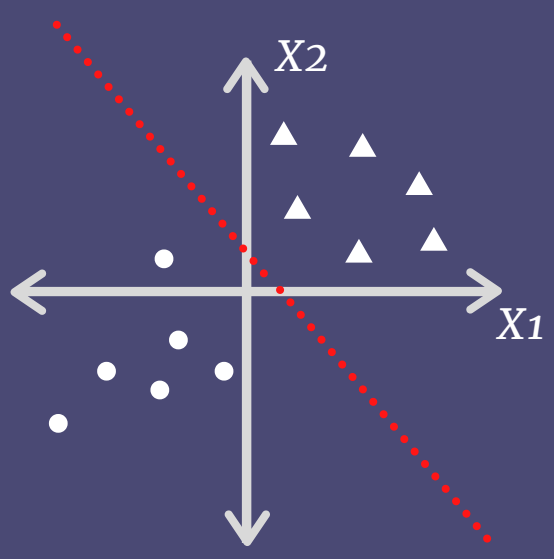
$$heaviside\ (z) = \begin{cases} 0 & if z < 0 \\ 1 & if\ z \geq 0 \end{cases} \qquad sgn(z) = \begin{cases} -1 & if\ z < 0 \\ 0 & if\ z = 0 \\ +1 & if z > 0 \end{cases}$$

A single TLU can be used for simple linear binary classification. It computes a linearcombination of the inputs, and if the result exceeds a threshold, it outputs the positive class. Otherwise it outputs the negative class.

ESTRUCTURE OF A PERCEPTRON



*The decision boundary of each output neuron is linear, so Perceptrons are incapable of learning complex patterns (just like Logistic Regression classifiers) and (as the TLU's is based on) do **NOT** output a class probability, they make predictions based on a hard treshold . However, if the training instances are **linearly separable**, Rosenblatt demonstrated that this algorithm would converge to a solution.*



Using linear algebra, the outputs of a layer of artificial neurons for sevelar instances can be computed at once:

$$Hw,b(X)=\varphi(XW+b)$$

- *X represents the matrix of input features.*
- *The weight matrix W contains all the connection weights except for the ones from the bias neuron.*
- *The bias vector b contains all the connection weights between the bias neuron and the artificial neurons.*
- *The function φ is the activation function.*

HOW IS A PERCEPTRON TRAINED?

INSPIRED BY HEBB'S RULE

Hebb's rule, sugested by Donald Hebb in the book *The Organization of Behavior* (1949), states that the connection wieght between two neurons tends to increase when they fire simultaneosly.

ThePerceptron learning rule reinforces connections that help reduce the error.

More specifically, the Perceptron is fed one training instance at a time, and for each instance it makes its predictions.

For every output neuron that produced a wrong prediction, it reinforces the connection weights from the inputs that would have contributed to the correct prediction.

PERCEPTRON LEARNING RULE

$$Wi,j^{(next\ step)}=Wi,j+N(Aj-\hat{A}j)Xi$$

- Wi,j is the connection weight between the i'th input neuron and the j'th output neuron.
- Xi is the ith input value of the current training instance.
- $\hat{A}j$ is the output of the j'th output neuron for the current training instance.
- Aj is the target output of the j'th output neuron for the current training instance.
- N is the learning rate.

Made By
Danilo Chaves
Leonardo Lizano

BASED ON: THE PERCEPTRON