Centro Brasileiro de Pesquisas Físicas

Redes Neurais profundas e aplicações Deep Learning

Clécio Roque De Bom – debom@cbpf.br

clearnightsrthebest.com

CBPF

Loss function (função custo)

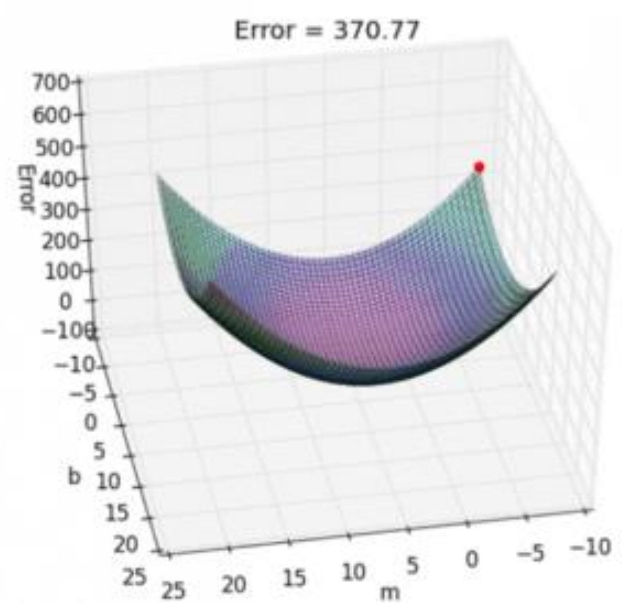
Loss functions in Machine Learning serve as ways to measure the distance or difference between a model's predicted output Y_{out} and the ground truth label Y in order to train our model effectively

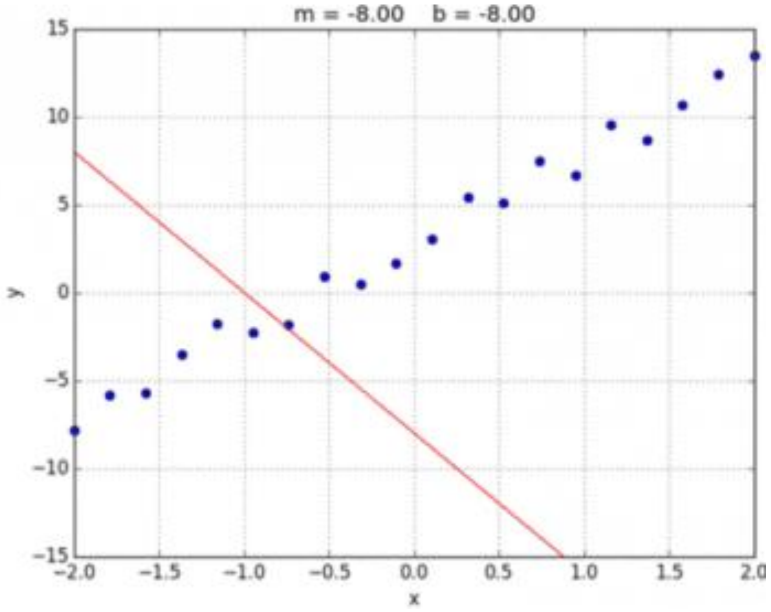
- **L2 Norm loss/ Euclidean loss function:** $L2 = \left(Y_{true} - Y_{pred}\right)^2$

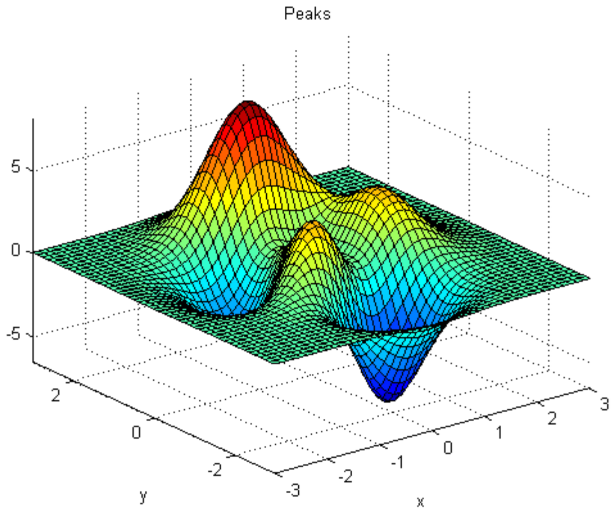
- **Cross entropy Loss:**

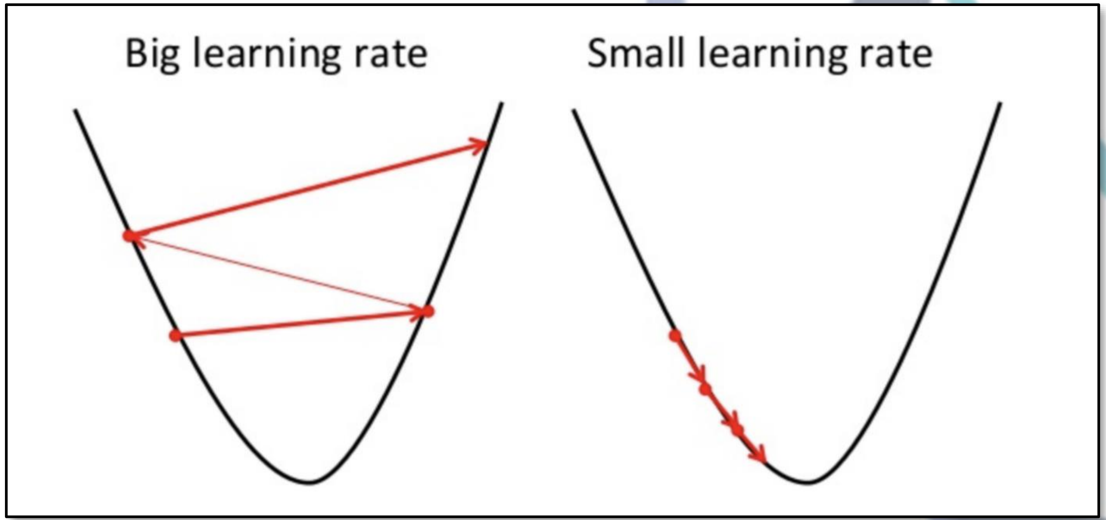
$$H(p,q) = -\sum_i p_i log q_i \; = -y\, log\hat{y} - (1-y)\log(1-\hat{y})$$

Rede Neural Artificial: Stochastic Gradient Descent - Algorithm



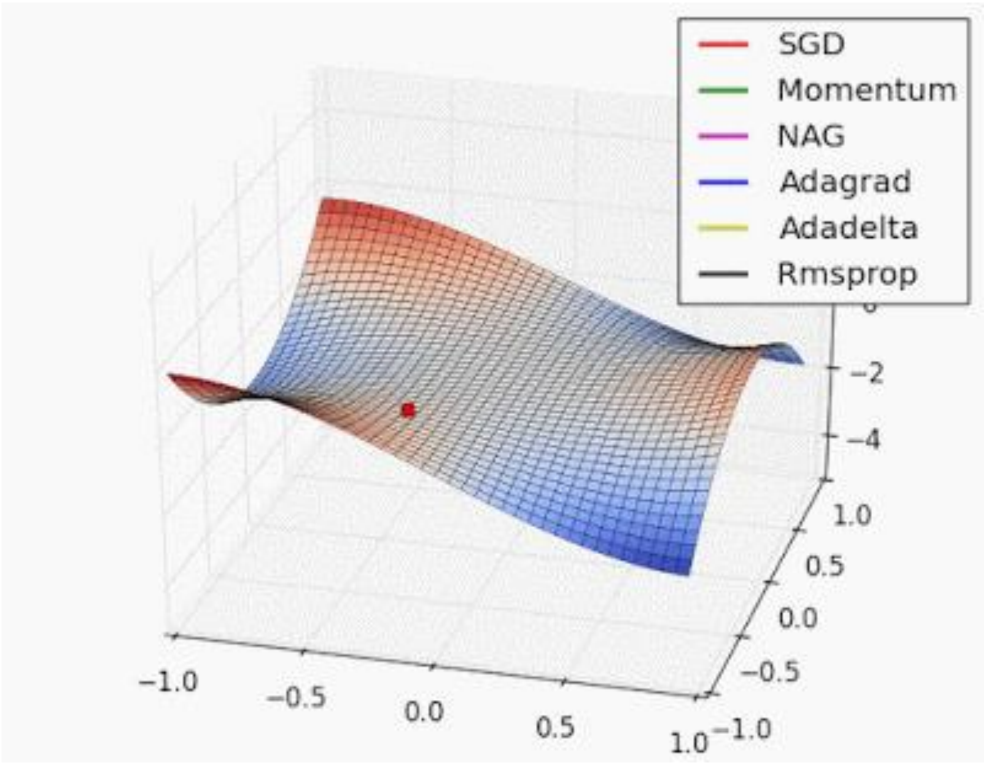


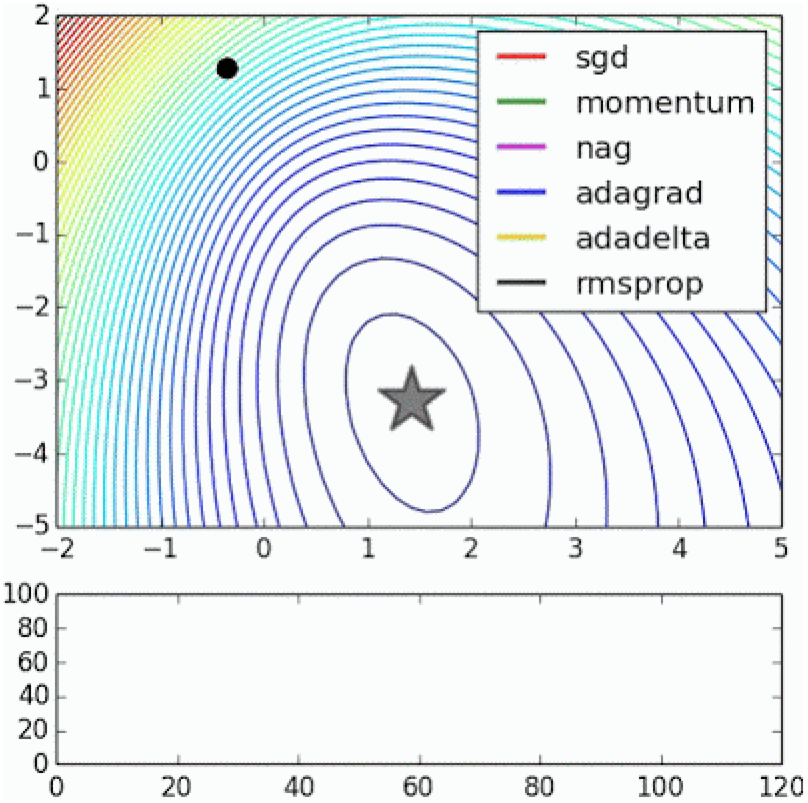




https://towardsdatascience.com/a-look-at-gradient-descent-and-rmsprop-optimizers-f77d483ef08b

Artificial Neural Network: Optimization Algorithms

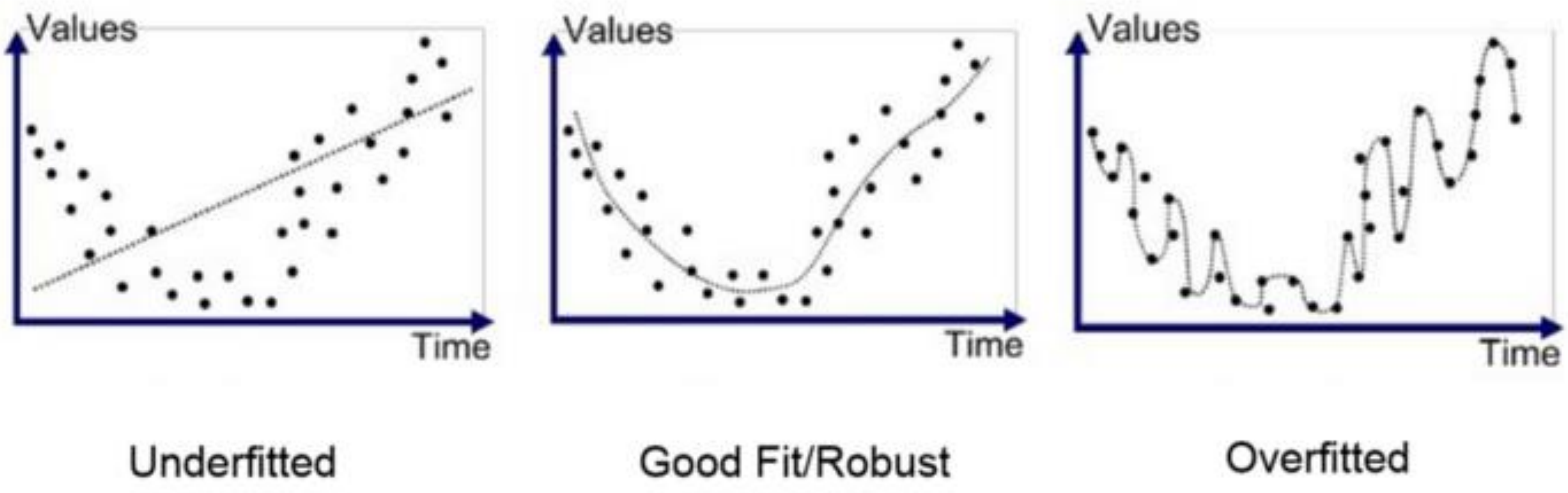




SGD: Stochastic Gradient Descent
ADAGRAD: Adaptive Gradient
ADADELTA: Adaptive Learning Rate Method
RMSPROP: Root Mean Square Propagation
Gradient depends on the average of the magnitudes of squares of previous gradients.

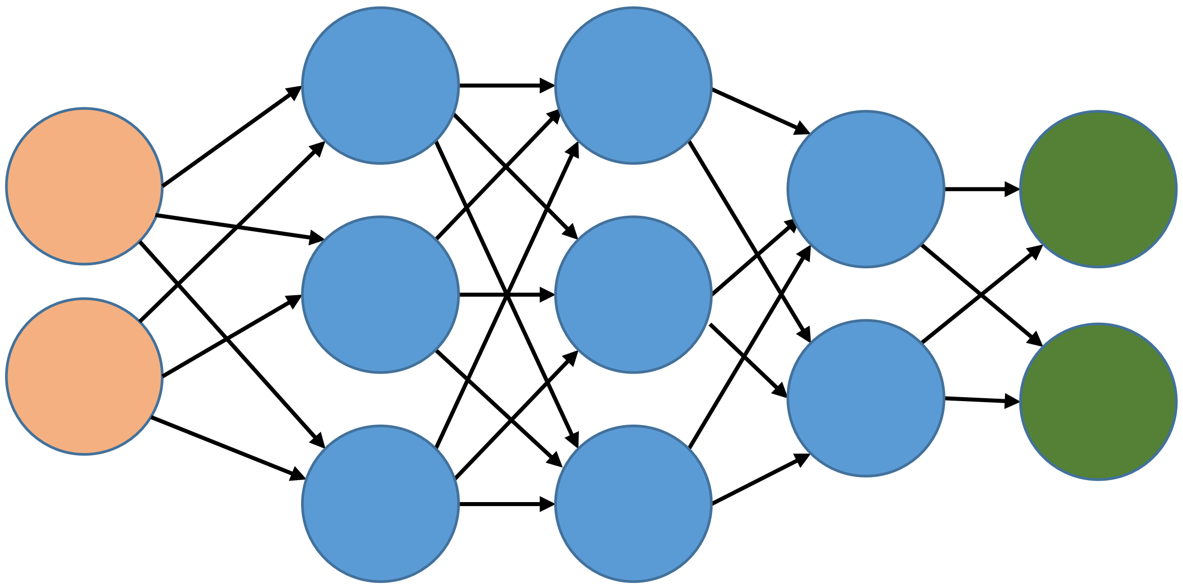
http://www.denizyuret.com/2015/03/alec-radfords-animations-for.html

Artificial Neural Network: Overfitting



O **overfitting** (*sobreajuste*) é um termo para descrever quando um modelo se ajusta muito bem ao conjunto de dados, mas se mostra ineficaz para prever novos resultados.

Artificial Neural Network: DROPOUT

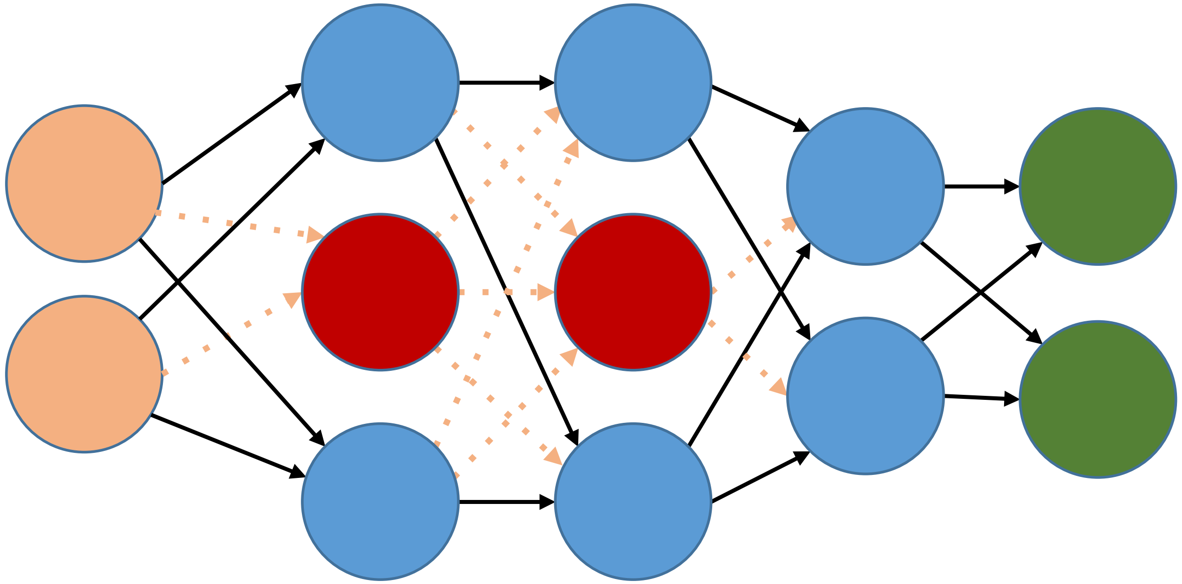


Dropout is a technique where randomly selected neurons are ignored during training. They are “dropped-out” randomly.

Durante o processo de treinamento devemos escolher uma probabilidade de retirada de neurônios.

A Simple Way to Prevent Neural Networks from Overfitting.

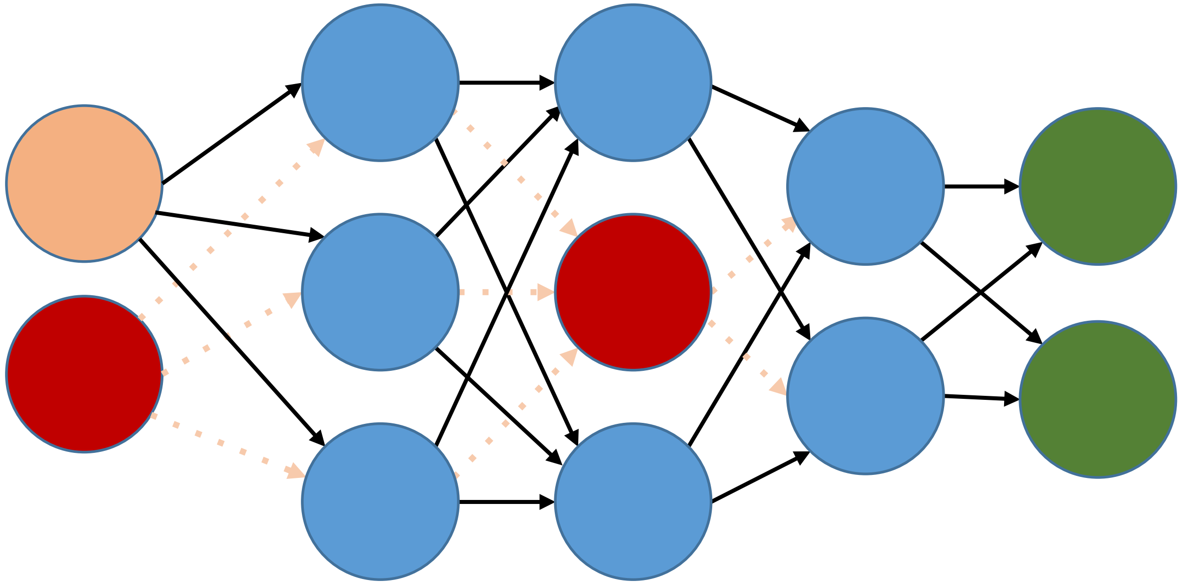
Artificial Neural Network: DROPOUT



Dropout is a technique where randomly selected neurons are ignored during training. They are “dropped-out” randomly.

A Simple Way to Prevent Neural Networks from Overfitting.

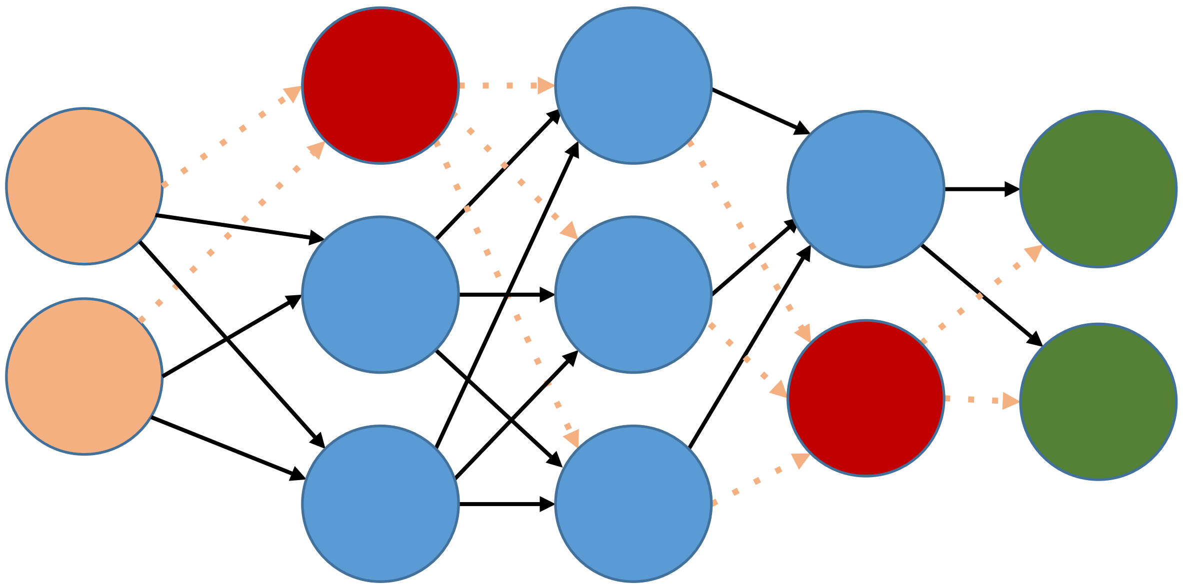
Artificial Neural Network: DROPOUT



Dropout is a technique where randomly selected neurons are ignored during training. They are “dropped-out” randomly.

A Simple Way to Prevent Neural Networks from Overfitting.

Artificial Neural Network: DROPOUT



Dropout is a technique where randomly selected neurons are ignored during training. They are “dropped-out” randomly.

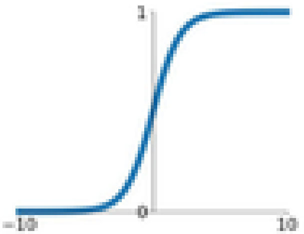
A Simple Way to Prevent Neural Networks from Overfitting.

Função de Ativação

Activation Functions

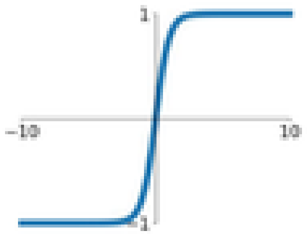
Sigmoid

$\sigma(x) = \frac{1}{1+e^{-x}}$



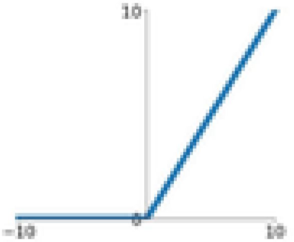
tanh

$\tanh(x)$



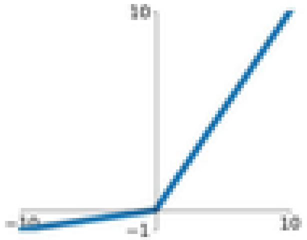
ReLU

$\max(0, x)$



Leaky ReLU

$\max(0.1x, x)$

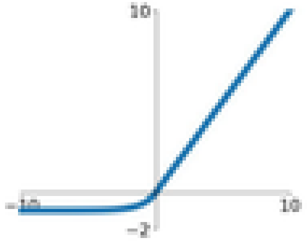


Maxout

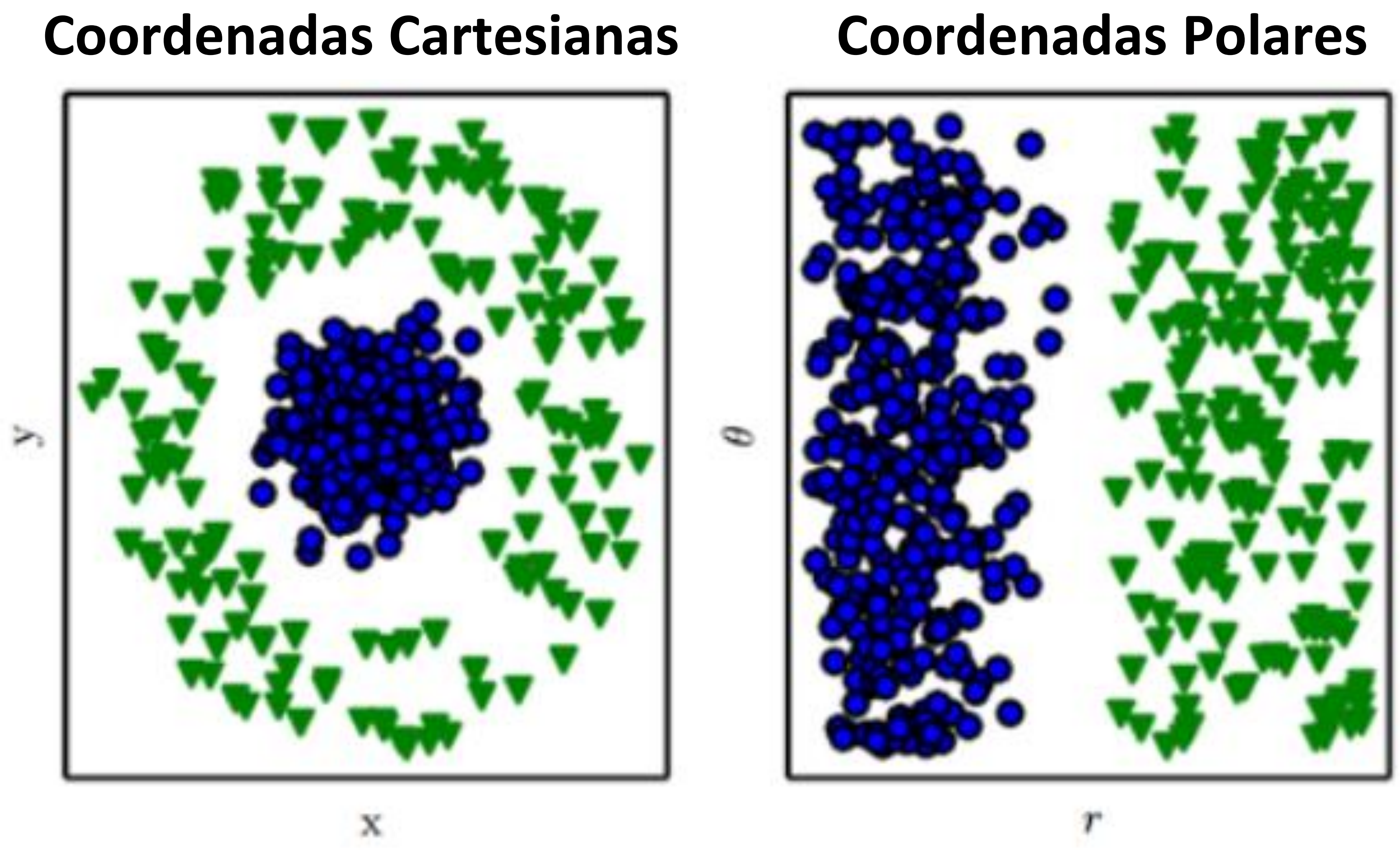
$\max(w_1^T x + b_1, w_2^T x + b_2)$

ELU

$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$



A Representação dos dados é importante



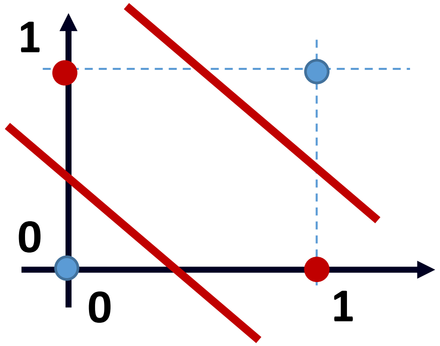
David Warde-Farley, Goodfellow et al. 2017

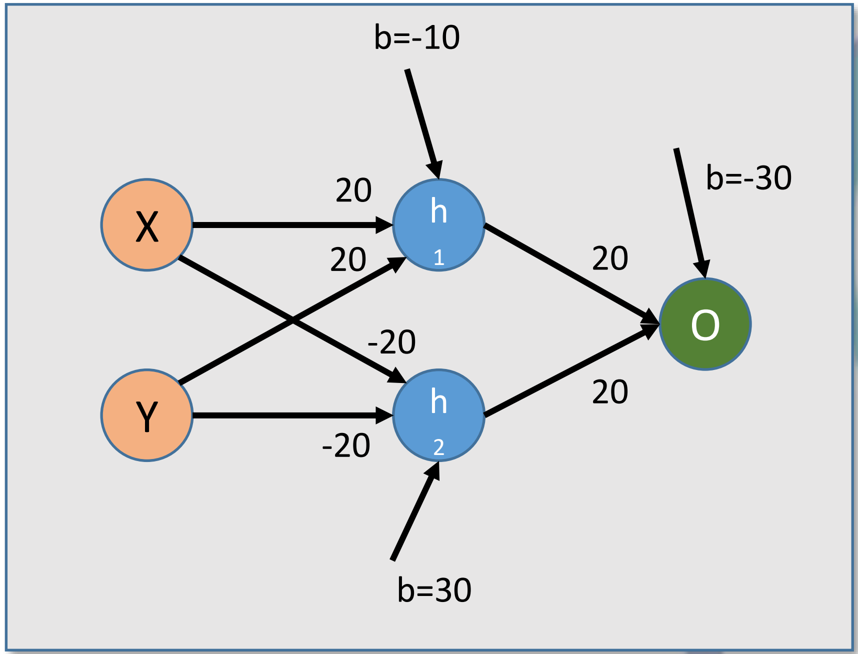
Rede Neural Artificial: O problema do OU Exclusivo

Multi-Layer Perceptron

XOR

X	Y	S
0	0	0
0	1	1
1	0	1
1	1	0





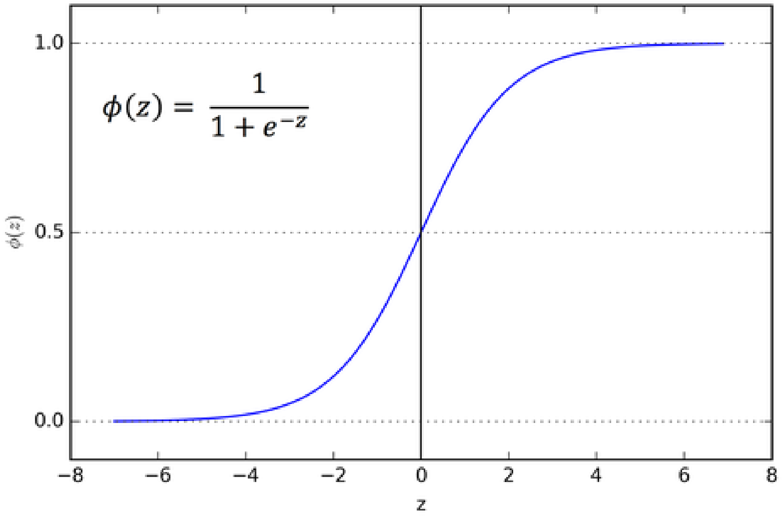
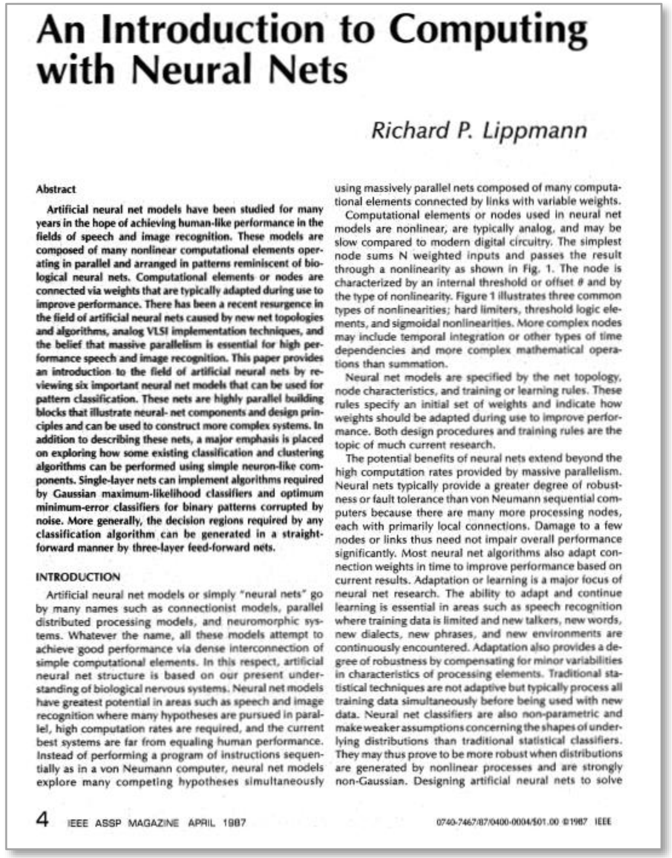


Fig: Sigmoid Function

[0 0] = φ (20*0 + 20*0 -10) ≈ 0 φ (-20*0 -20*0 +30) ≈ 1 φ (20*0 +20*1 -30) ≈ 0
[0 1] = φ (20*0 + 20*1 -10) ≈ 1 φ (-20*0 -20*1 +30) ≈ 1 φ (20*1 +20*1 -30) ≈ 1
[1 0] = φ (20*1 + 20*0 -10) ≈ 1 φ (-20*1 -20*0 +30) ≈ 1 φ (20*1 +20*1 -30) ≈ 1
[1 1] = φ (20*1 + 20*1 -10) ≈ 1 φ (-20*1 -20*1 +30) ≈ 0 φ (20*1 +20*0 -30) ≈ 0

RNA: Estruturas e Tipos de Regiões de Decisão



An Introduction to Computing with Neural Nets

Richard P. Lippmann

Abstract

Artificial neural net models have been studied for many years in the hope of achieving human-like performance in the fields of speech and image recognition. These models are composed of many nonlinear computational elements operating in parallel and arranged in patterns reminiscent of biological neural nets. Computational elements or nodes are connected via weights that are typically adapted during use to improve performance. There has been a recent resurgence in the field of artificial neural nets caused by new net topologies and algorithms, analog VLSI implementation techniques, and the belief that massive parallelism is essential for high performance speech and image recognition. This paper provides an introduction to the field of artificial neural nets by reviewing six important neural net models that can be used for pattern classification. These nets are highly parallel building blocks that illustrate neural- net components and design principles and can be used to construct more complex systems. In addition to describing these nets, a major emphasis is placed on exploring how some existing classification and clustering algorithms can be performed using simple neuron-like components. Single-layer nets can implement algorithms required by Gaussian maximum-likelihood classifiers and optimum minimum-error classifiers for binary patterns corrupted by noise. More generally, the decision regions required by any classification algorithm can be generated in a straightforward manner by three-layer feed-forward nets.

INTRODUCTION

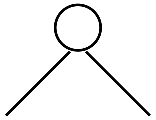
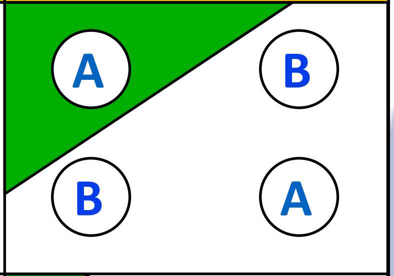
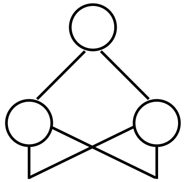
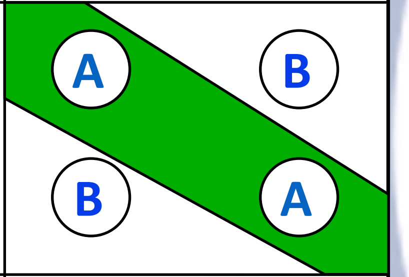
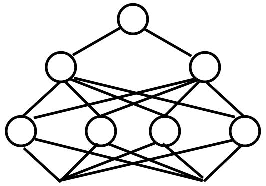
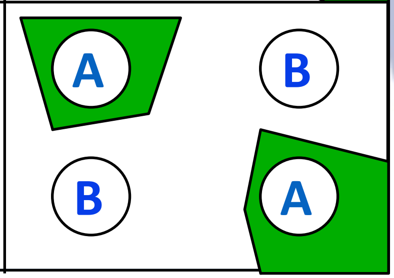
Artificial neural net models or simply "neural nets" go by many names such as connectionist models, parallel distributed processing models, and neuromorphic systems. Whatever the name, all these models attempt to achieve good performance via dense interconnection of simple computational elements. In this respect, artificial neural net structure is based on our present understanding of biological nervous systems. Neural net models have greatest potential in areas such as speech and image recognition where many hypotheses are pursued in parallel, high computation rates are required, and the current best systems are far from equaling human performance. Instead of performing a program of instructions sequentially as in a von Neumann computer, neural net models explore many competing hypotheses simultaneously using massively parallel nets composed of many computational elements connected by links with variable weights.

Computational elements or nodes used in neural net models are nonlinear, are typically analog, and may be slow compared to modern digital circuitry. The simplest node sums N weighted inputs and passes the result through a nonlinearity as shown in Fig. 1. The node is characterized by an internal threshold or offset θ and by the type of nonlinearity. Figure 1 illustrates three common types of nonlinearities; hard limiters, threshold logic elements, and sigmoidal nonlinearities. More complex nodes may include temporal integration or other types of time dependencies and more complex mathematical operations than summation.

Neural net models are specified by the net topology, node characteristics, and training or learning rules. These rules specify an initial set of weights and indicate how weights should be adapted during use to improve performance. Both design procedures and training rules are the topic of much current research.

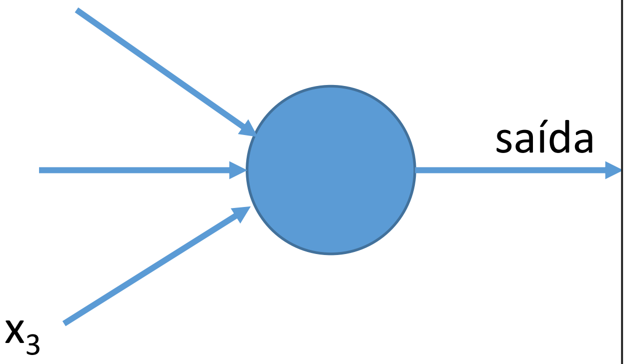
The potential benefits of neural nets extend beyond the high computation rates provided by massive parallelism. Neural nets typically provide a greater degree of robustness or fault tolerance than von Neumann sequential computers because there are many more processing nodes, each with primarily local connections. Damage to a few nodes or links thus need not impair overall performance significantly. Most neural net algorithms also adapt connection weights in time to improve performance based on current results. Adaptation or learning is a major focus of neural net research. The ability to adapt and continue learning is essential in areas such as speech recognition where training data is limited and new talkers, new words, new dialects, new phrases, and new environments are continuously encountered. Adaptation also provides a degree of robustness by compensating for minor variabilities in characteristics of processing elements. Traditional statistical techniques are not adaptive but typically process all training data simultaneously before being used with new data. Neural net classifiers are also non-parametric and make weaker assumptions concerning the shapes of underlying distributions than traditional statistical classifiers. They may thus prove to be more robust when distributions are generated by nonlinear processes and are strongly non-Gaussian. Designing artificial neural nets to solve

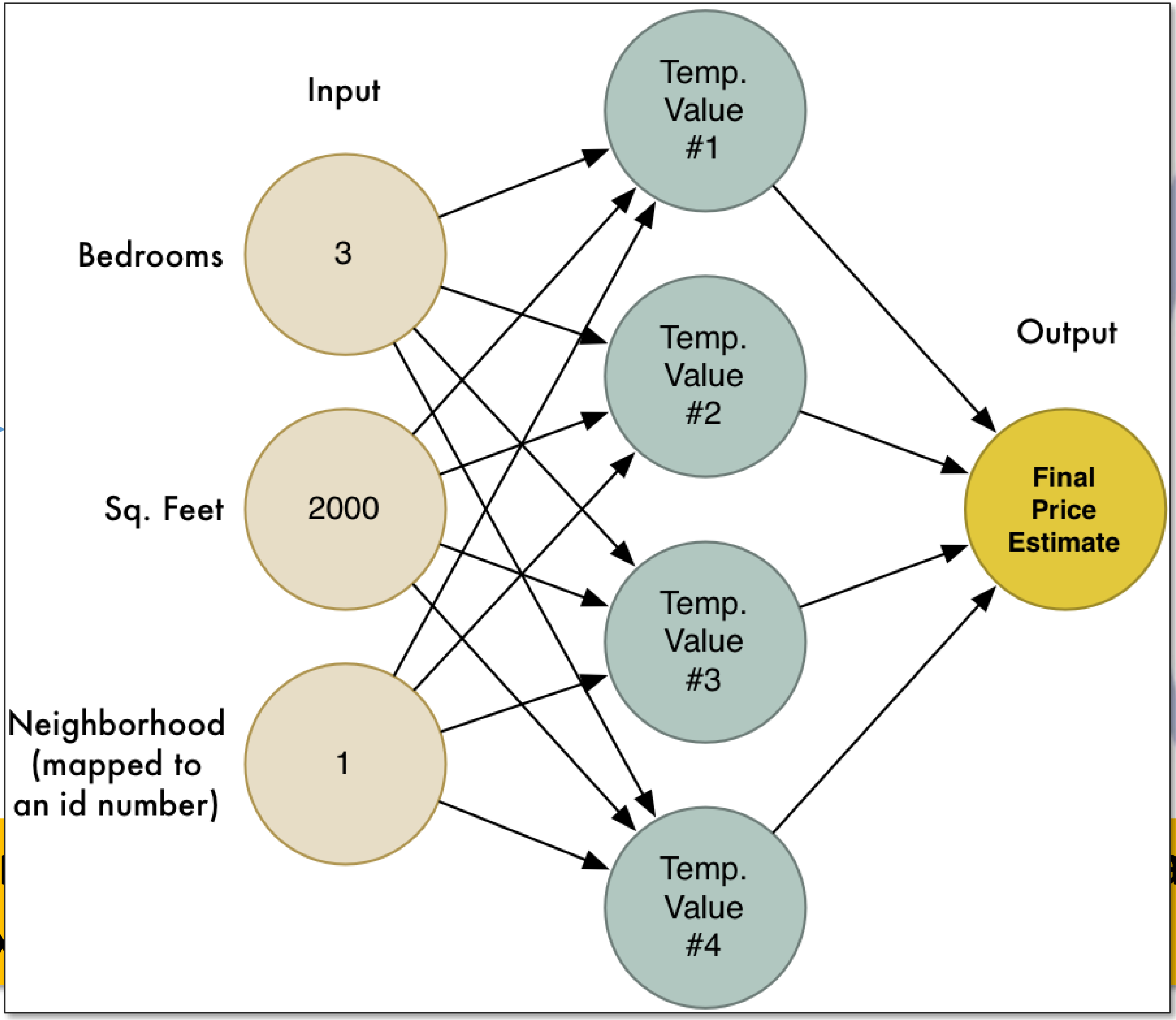
0740-7467/87/0400-0004/$01.00 ©1987 IEEE

Structure	*Types of Decision Regions*	*Exclusive-OR Problem*
Single-Layer 	*Half Plane Bounded By Hyperplane*	
Two-Layer 	*Convex Open Or Closed Regions*	
Three-Layer 	Abitrary (Complexity Limited by No. of Nodes)	

R. Lippmann, "*An introduction to computing with neural nets*“, IEEE ASSP Magazine, vol. 4, no. 2, pp. 4-22, Apr/1987.- doi: 10.1109/MASSP.1987.1165576

Rede Neural Artificial:



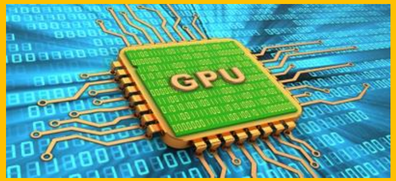


U… a f (x),
ex…

Rede Neural Artificial: DEEP LEARNING

DEEP LEARNING = APRENDIZADO PROFUNDO

- **SISTEMAS COMPUTACIONAIS**
  - CPUs, GPUs, ASICs
  - MEMÓRIA



- **BANCO DE DADOS ESTRUTURADOS**
  - Imagenet, kaggle ...

- **P&D (CONCEITOS e ALGORITMOS)**
  - CNN, RCNN, BACKPROP, LSTM, ...

- **INFRAESTRUTURA DE SOFTWARE**
  - Git, AWS, AMAZON MECHANICAL TURK, GOOGLE TENSORFLOW

- **INTERESSE COMERCIAL E FINANCIAMENTO POR GRANDES COMPANHIAS**
  - Google, Facebook, Amazon, ...

Ministério da Ciência, Tecnologia e Inovações

Centro Brasileiro de Pesquisas Físicas

Redes Neurais profundas e aplicações Deep Learning

Clécio Roque De Bom – debom@cbpf.br

clearnightsrthebest.com

CBPF