



Departamento de Ciência da Computação (DCC)

# Redes Neurais Artificiais

## GCC128 - Inteligência Artificial

**Vinicius Ruela Pereira Borges**

[viniciusrpb@icmc.usp.br](mailto:viniciusrpb@icmc.usp.br)

*Lavras, 27 de julho de 2016*

# Roteiro - Redes Neurais Artificiais (RNA)

- Introdução
  - Motivação biológica
  - Conceito de RNA
  - Aplicações
  - Histórico
- Modelagem de um Neurônio Artificial
- Propriedades de RNAs
- RNA Perceptron Simples
- Exercícios

# Introdução

- Redes Neurais? Neural?

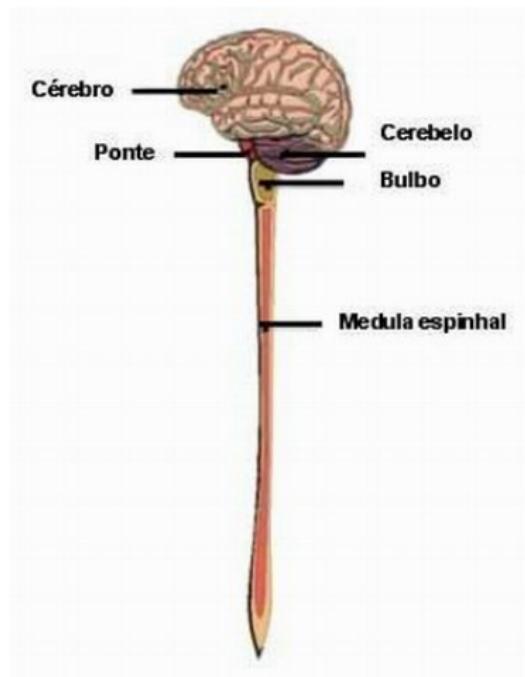


# Introdução

- Sistema Nervoso Central é responsável pela maioria das funções de controle em um organismo, coordenando e regulando as atividades corporais
  - Sistema Nervoso Periférico
  - Sistema Nervoso Central
- O neurônio é a unidade funcional deste sistema
  - recebe estímulos externo

# Introdução: Sistema Nervoso Central (SNC)

- encéfalo
- medula espinhal
- recebem informações dos estímulos



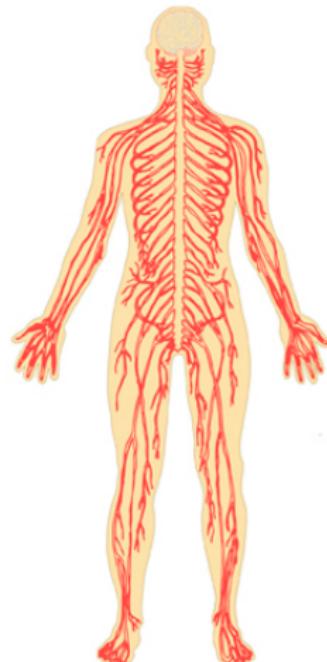
# Introdução: Sistema Nervoso Central (SNC)

- encéfalo
- medula espinhal
- recebem informações dos estímulos



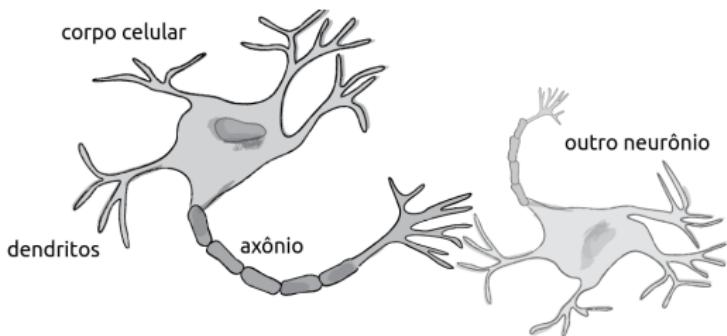
# Introdução: Sistema Nervoso Periférico (SNP)

- Parte do Sistema Nervoso fora do SNC
- Constituído de:
  - fibras
  - gânglios nervosos
  - órgãos terminais



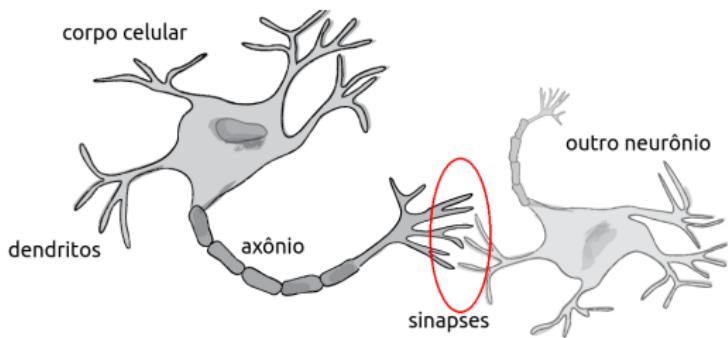
# Neurônio

- Célula que compõe o tecido nervoso
- Responsável pela condução do impulso nervoso
- Partes de um neurônio:
  - dendritos
  - axônio
  - corpo celular



# Neurônio

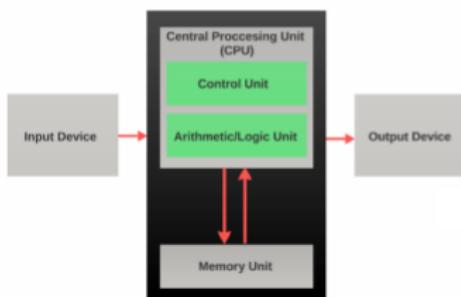
- A junção que conecta dois neurônios é chamada de sinapse
- Permite a transmissão de impulsos nervosos entre dois neurônios
- Um neurônio pode formar sinapses com até 1000 outros neurônios



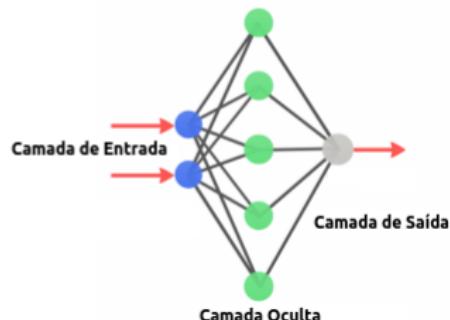
# Computador vs Cérebro

Unidades computacionais	1 CPU, $10^5$ portas	$10^{11}$ neurônios
Unidades de armazenamento	$10^9$ bits/RAM, $10^{10}$ bits/disco	$10^{11}$ neurônios, $10^{14}$ sinapses
Tempo de ciclo	$10^{-8}$ seg	$10^{-3}$ seg
Largura de banda	$10^9$ bits/seg	$10^{14}$ bits/seg
Atualização dos neurônios	$10^5$	$10^{14}$

Arquitetura  
Von Neumann



Arquitetura  
Rede Neural

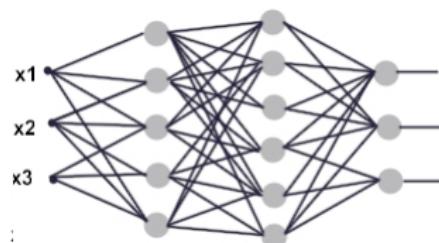


# O que o cérebro faz?

- Apesar de lento em alguns aspectos, o cérebro é preciso e eficiente na execução de tarefas complexas
  - Por exemplo, reconhecimento de objetos
- O cérebro é capaz de construir suas próprias regras
  - Usa a experiência
  - Composto por várias regiões especializadas
- O neurônio é a unidade funcional deste sistema

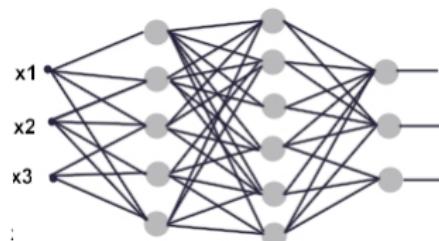
# Redes Neurais Artificiais

- É um modelo matemático que simula a operação do cérebro
- Representar funções por meio de uma rede composta por elementos computacionais
  - ... processam operações aritméticas simples
- Métodos que aprendem tais funções com base em exemplos



# Redes Neurais Artificiais

- Modelo computacional que aproxima o funcionamento do cérebro humano quando se busca entender fenômenos ou interpretar estímulos
- Composto por várias unidades de processamento
  - Neurônios: visão abstrata das operações aritméticas
- Os neurônios são interligados por várias conexões
  - Sinapses



# Aplicações

- Pesquisa Científica
  - classificação
  - agrupamento
  - regressão
- Telecomunicações:
  - detecção de falha em linha de rede
- Genética
  - sequenciamento de proteínas
- entre outros...

# Trade-off

- Vantagens
  - solução naturalmente paralela
  - robusta, tolerante a falhas
  - permite a integração de informações oriundas de fontes ou tipos diferentes
  - sistema adaptativo, capaz de aprender
  - mostra certo grau de autonomia no aprendizado
  - performance muito rápida no reconhecimento
- Desvantagens
  - ainda muito difícil explicar seu comportamento, por causa da falta de transparência
  - soluções não escalam bem... computacionalmente cara para problemas de maior porte
  - ainda muito distante da realidade biológica

- **1943:** McCulloch e Pitts



- representação de um evento do sistema nervoso
- primeiro modelo conceitual de uma rede neural artificial
- definição de neurônio

- **1943:** McCulloch e Pitts



- Com uma quantidade suficiente de unidades simples e um conjunto de conexões sinápticas e operando sincronicamente, eles mostraram que uma rede assim constituída calcularia, em princípio qualquer função computável.

- 1958: Frank Rosenblatt



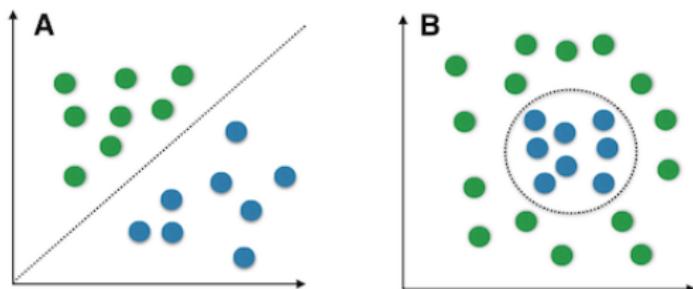
- O idealizador da Rede Neural Perceptron
- Utilizando sinapses ajustáveis, as RNAs com várias camadas de Perceptrons poderiam ser treinadas para classificar certos tipos de padrões
- Provou seu teorema da convergência, que poderia ser utilizado no processo de aprendizado da rede

# Histórico

- 1969: Minsky e Papert
- Problemas que não são linearmente separáveis não são passíveis de aprendizado
- Provou matematicamente que o Perceptron não era capaz de executar tarefas

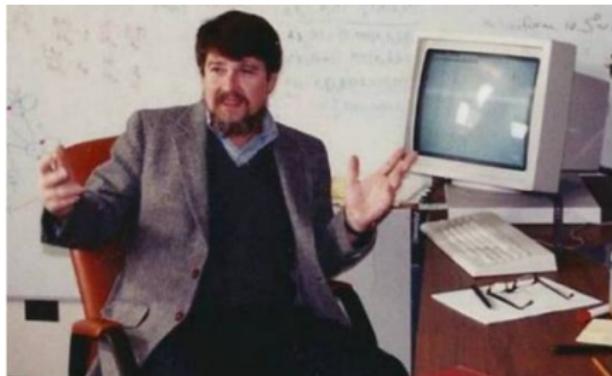


Linear vs. nonlinear problems



# Histórico

- **1986:** Rumelhart, Hinton e Williams
- Tratar problemas não-lineares
- Propôs uma RNA de múltiplas camadas capazes de resolver problemas mais complexos
- Backpropagation: técnica para treinar redes com camadas escondidas



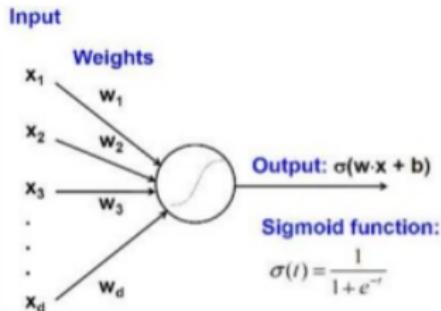
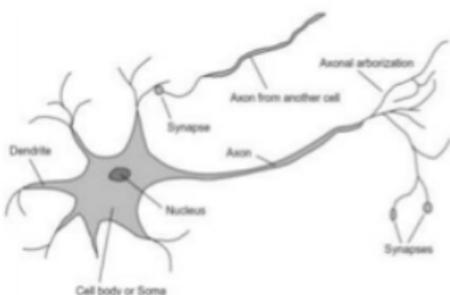
# Motivação

- Computadores são extremamente eficientes e eficazes quando realizam operações aritméticas e processamento simples de dados
- No entanto, ainda não conseguem realizar com precisão algumas tarefas cotidianas:
  - Por exemplo, reconhecimento de faces



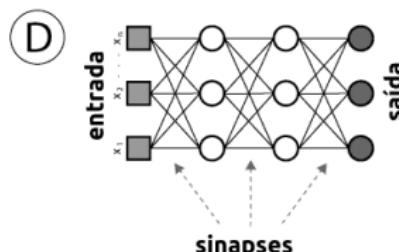
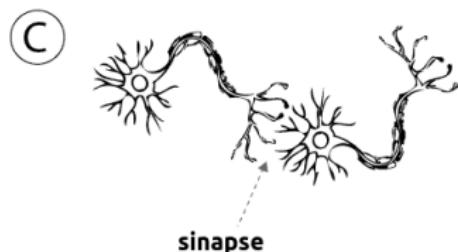
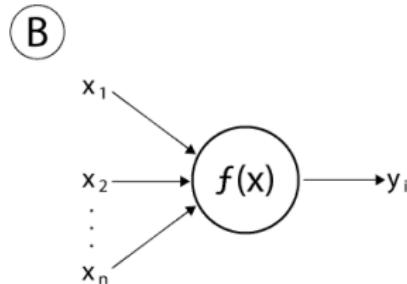
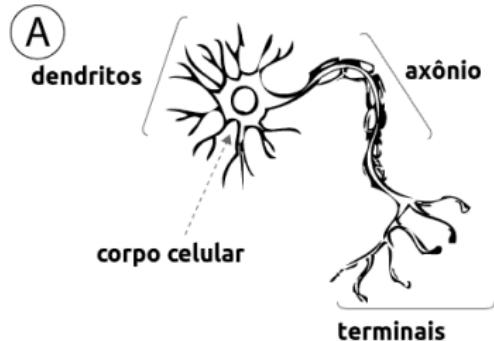
# Comparação

## Neurônio Biológico vs Neurônio Artificial



Perceptron

# Visão Geral

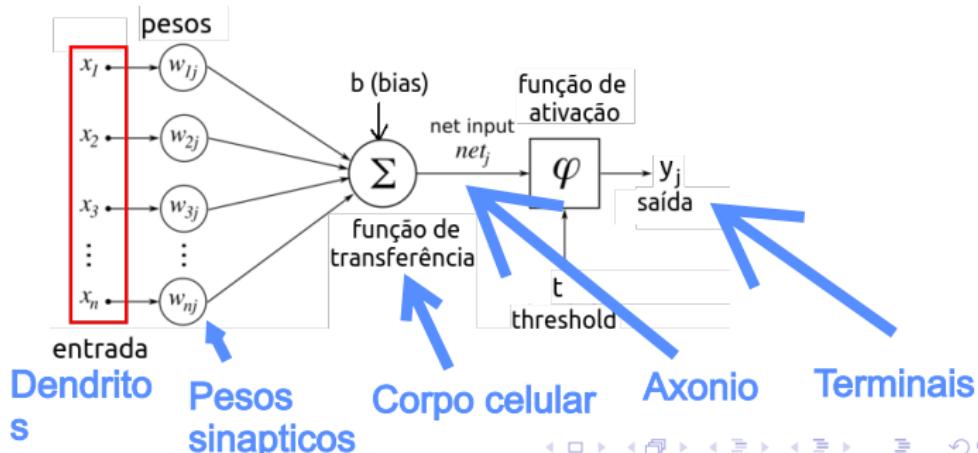


# Modelo de McCulloch-Pitts (1943)

- Funcionamento binário do Neurônio: é estimulado ou não
- Um neurônio é estimulado quando um certo número de sinapses ocorre em um determinado intervalo de tempo
- O único atraso no sistema nervoso é o atraso sináptico
- A atividade de qualquer sinapse inibitória previne a excitação de outro neurônio
- A estrutura da rede de neurônios não muda com o tempo

# Modelando um Neurônio Artificial

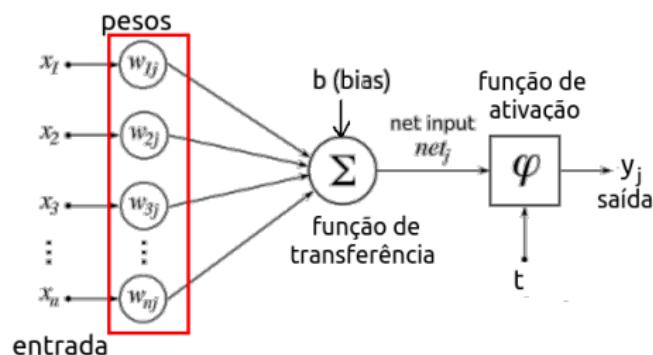
- Dendritos: posicionados de forma a ilustrar os dados de entrada
- Representam o meio por onde chega a informação
- Formalmente definido como:  $\{x_1, \dots, x_N\}$



# Modelando um Neurônio Artificial

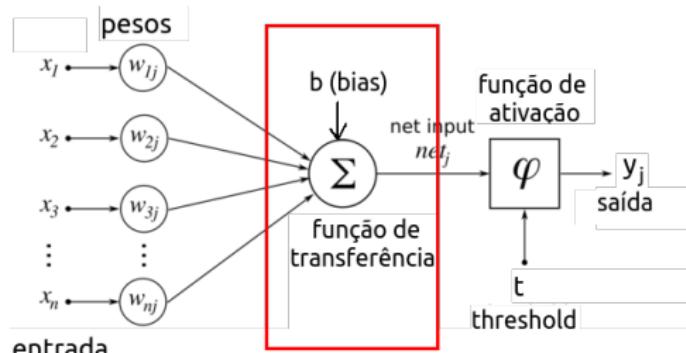
- Pesos sinápticos: a informação recebida pelos dendritos é ponderada para regular o sinal de entrada
- Formalmente definido como:  $\{w_1, \dots, w_d\}$

Ola



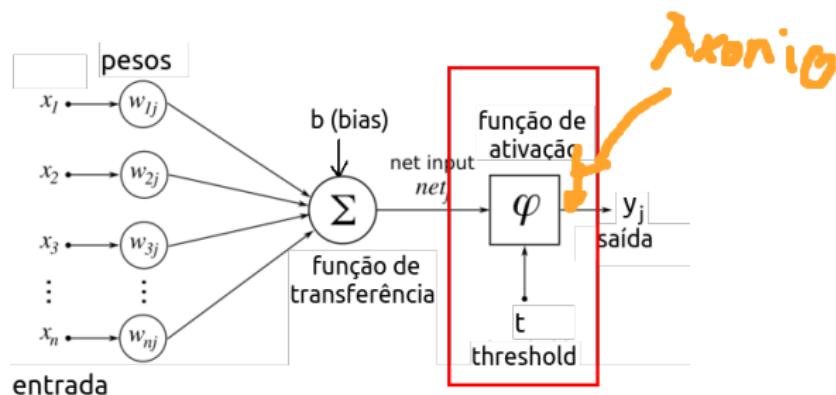
# Modelando um Neurônio Artificial

- Corpo celular: modelados como nós, consideram o conjunto total das entradas e seus pesos associados
- $b$  é o Bias
- $net_i = x_1 w_1 + \dots + x_d w_d - b$



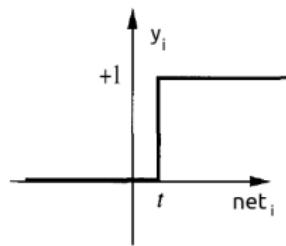
# Modelando um Neurônio Artificial

- Axônio é o resultado da função de ativação  $\varphi$
- Retorna um valor que indica se o neurônio é estimulado ou não
- $y_i = \Phi(\text{net}_i)$ 
  - +1
  - -1

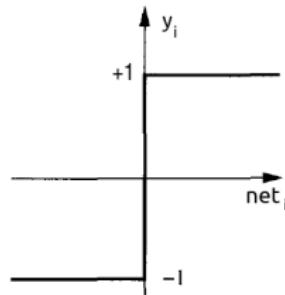


# Modelando um Neurônio Artificial

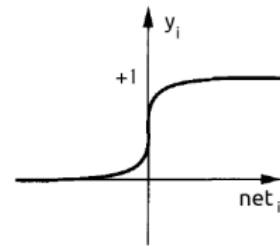
- Axônio é o resultado da função de ativação  $\varphi$
- Retorna um valor que indica se o neurônio é estimulado ou não
- $y_i = \varphi(\text{net}_i)$ 
  - +1
  - -1 (ou zero)



Função  
Degrau (via Threshold)



Função Sinal

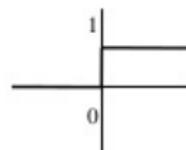


Função Sigmóide

# Modelando um Neurônio Artificial

- Função degrau

$$g(x) = \begin{cases} 1 & x > t \\ 0 & \text{caso contrário} \end{cases}$$



- Função sinal

$$g(x) = \begin{cases} +1 & x \geq 0 \\ -1 & \text{caso contrário} \end{cases}$$



- Função Sigmóide

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



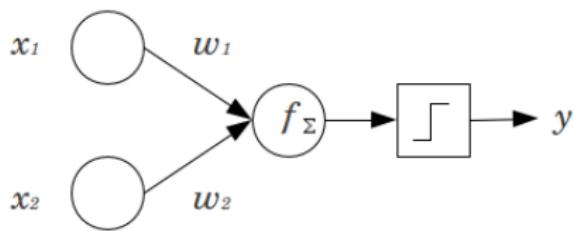
# Exercício

- Um neurônio recebe entrada de outros 4 neurônios cujos níveis de atividade são -15, 2, 3 e -9. Os pesos sinápticos são 0.3, 0.8, -0.7, e 0.9.
  - Represente graficamente o neurônio artificial
  - Calcule a saída para uma função de ativação do tipo sinal

# Topologia de uma RNA

- **Número de camadas**

- Única camada
- Adaline,  
Perceptron

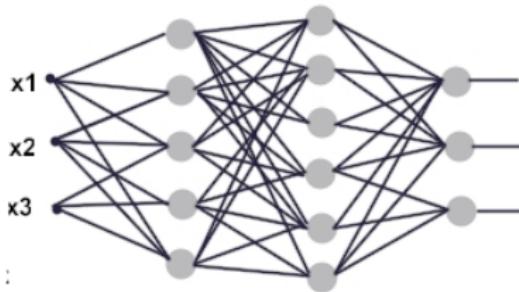


# Topologia de uma RNA

- Número de camadas

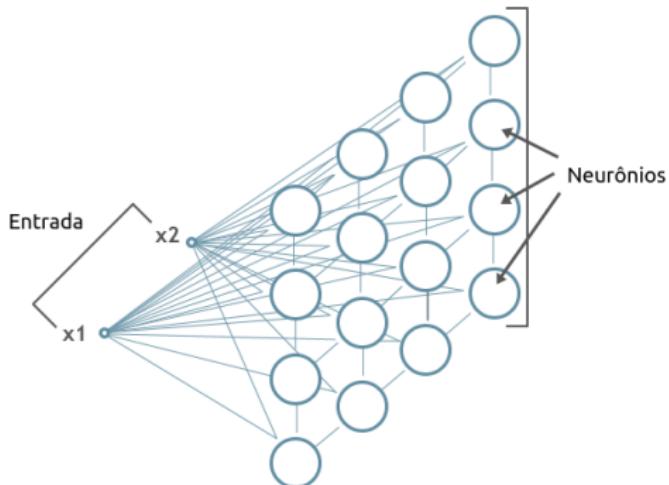
- Multi-camada
- Multi-layer perceptron e Funções de base radiais (RBF)

• •



# Topologia de uma RNA

- Organização das conexões
  - Matriz  $n$ -dimensional de neurônios
  - Mapas Auto-Organizáveis



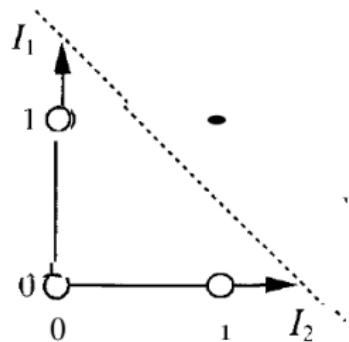
# Aprendizado de uma RNA

- Os paradigmas usualmente empregados:
  - Supervisionado
  - Não-supervisionado
  - Reforço
- Algoritmos de aprendizado obtém informações através dos exemplos para ensinar a RNA a resolver um problema
  - Ajustam os valores dos pesos sinápticos

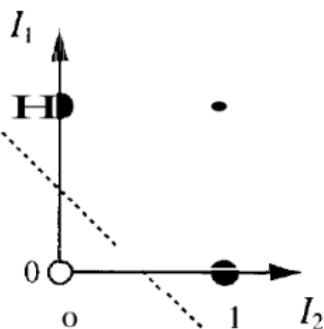
# Redes Perceptron Simples

- Proposta por Rosenblatt em 1958, é a forma mais simples de RNA para classificar dados linearmente separáveis
- RNA Perceptron é basicamente formada por um neurônio de McCulloch, cujas saída são -1 ou +1
- Por se tratar de um único neurônio, a rede é capaz de classificar os dados para duas classes
- Ajustando a saída da função de ativação de um neurônio para ser entrada de outro neurônio permite estender a classificação para mais classe

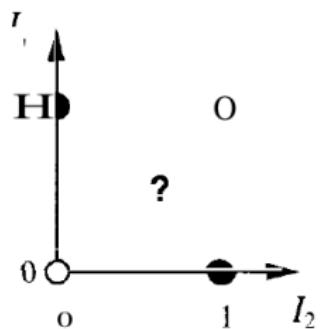
# Redes Perceptron Simples



(a)  $I_1$  and  $I_2$



(b)  $I_1$  or  $I_2$



(c)  $I_1$  xor  $I_2$

# Redes Perceptron Simples

- Regra de aprendizagem consiste em minimizar uma função objetivo (custo)

$$E = \sum_{i=1}^N [y_i - l(\mathbf{x}_i)]^2 \quad (1)$$

- para  $N$  exemplos de treinamento
- Se trata de um processo de otimização até a convergência de  $E$
- O que significa a função  $E$ ?

# Redes Perceptron Simples

- Regra de aprendizagem consiste em minimizar uma função objetivo (custo)

$$E = \sum_{i=1}^N [y_i - l(\mathbf{x}_i)]^2 \quad (2)$$

- Se trata de um processo de otimização até a convergência de  $E$
- O que significa a função  $E$ ?
  - Quantificando o erro de classificação
  - No processo de aprendizagem, deve-se minimizar os erros de classificação

## Redes Perceptron Simples - Treinamento

- Seja um exemplo de treinamento  $\mathbf{x}_i = \{x_{i,1}, \dots, x_{i,d}\}$  e seus respectivos pesos  $\{w_1, \dots, w_d\}$
- Regra de atualização dos pesos

$$w_j^{t+1} = w_j^t + \Delta w_j^t \quad (3)$$

- em que
  - $\Delta w_j^t$  é o valor de atualização para o peso  $w_j$

# Redes Perceptron Simples - Treinamento

- $\Delta w_j^t = \eta x_{i,j}(l(\mathbf{x}) - y_i)$ 
  - Classificação correta ( $l(\mathbf{x}) = y_i$ ):  $\Delta w_j^t = 0$
- Quem é  $\eta$ ?
  - Fator de aprendizagem que balanceia cada etapa do processo
  - $0 < \eta << 1$

# Redes Perceptron Simples - Algoritmo de treinamento

- ➊ Iniciar todos os pesos  $\{w_1^0, \dots, w_d^0\}$
- ➋  $t \leftarrow 0;$
- ➌ ENQUANTO  $E > \varepsilon$ 
  - ➍ Para cada par de treinamento  $(\mathbf{x}_i, l(\mathbf{x}_i))$
  - ➎ Calcular a saída da rede  $y_i$
  - ➏ SE  $(y_i \neq l(\mathbf{x}))$
  - ➐ Atualizar os pesos  $j \leftarrow 1, \dots, d$  do neurônio:  
$$\Delta w_j^t \leftarrow \eta x_{i,j} (l(\mathbf{x}_i) - y_i)$$
$$w_j^{t+1} \leftarrow w_j^t + \Delta w_j^t$$
$$t \leftarrow t + 1$$
- ➑  $\varepsilon$  é um limiar de precisão da classificação

# Exercício Operação AND

Exemplo $i$	$x_1$	$x_2$	$I(x_i)$
$i = 1$	0	0	0
$i = 2$	0	1	0
$i = 3$	1	0	0
$i = 4$	1	1	1

- Desenhe a superfície de decisão para o problema AND
- Represente graficamente o neurônio artificial para resolver o problema AND
- Treine o neurônio utilizando os pesos  $w_1 = 1$  e  $w_2 = 1$ , fator de aprendizagem  $\eta = 0.25$  e o limiar da função de ativação degrau  $t = 0.3$

# Exercício Operação OR

Exemplo $i$	$x_{i,1}$	$x_{i,2}$	$I(\mathbf{x}_i)$
$i = 1$	0	0	0
$i = 2$	0	1	1
$i = 3$	1	0	1
$i = 4$	1	1	1

- Desenhe a superfície de decisão para o problema OR
- Represente graficamente o neurônio artificial para resolver o problema OR
- Treine o neurônio utilizando os pesos  $w_1 = 0.1$  e  $w_2 = -0.5$ , , fator de aprendizagem  $\eta = 0.2$  e o limiar da função de ativação degrau  $t = 0.1$

# Bibliografia

- RUSSELL, Stuart J.; STUART, J. Norvig. Artificial Intelligence: A Modern Approach, Chapter 10 p. 320-374, 2003.
- TAN, Pang-Ning et al. Introduction to data mining. Boston: Pearson Addison Wesley, 2006.
- Alguns slides foram adaptados do material do Prof. André Backes (FACOM/UFU) <sup>1</sup>
- Alguns slides foram adaptados do material do Prof. João Garcia (ICMC/USP) <sup>2</sup>

---

<sup>1</sup>Acessado em 17 de fevereiro de 2016

<http://www.facom.ufu.br/~backes/pgc204/Aula07-RedesNeurais.pdf>

<sup>2</sup>Acessado em 16 de fevereiro de 2016

<http://wiki.icmc.usp.br/images/6/66/SCC5809Cap1.pdf>