iMobilis

SEVA

Introdução a Redes Neurais Artificiais

André Siqueira Ruela

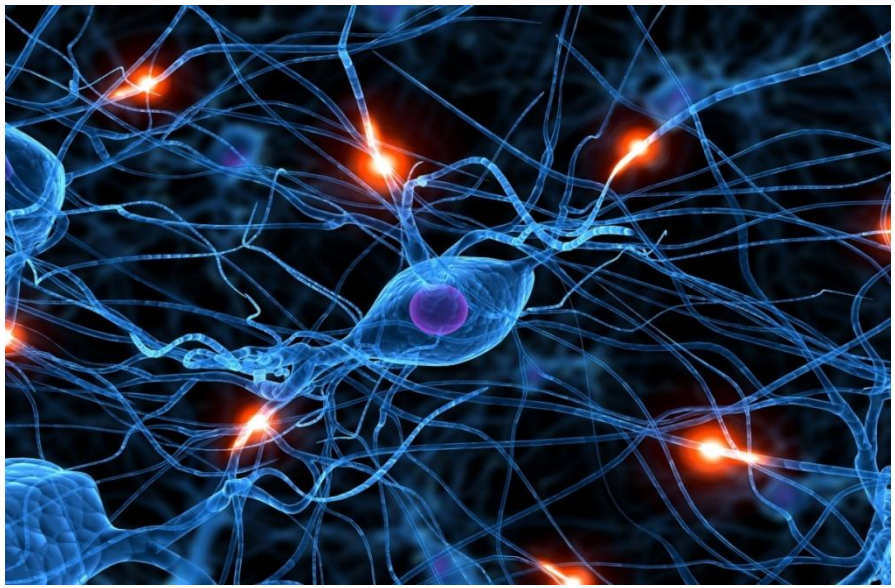
UFOP
Universidade Federal de Ouro Preto

decom
departamento de computação

Redes Neurais Biológicas

- Cérebro: Um computador altamente complexo, não-linear e paralelo.

- Principal componente: Neurônio



SEVA iMobilis

UFOP Universidade Federal de Ouro Preto

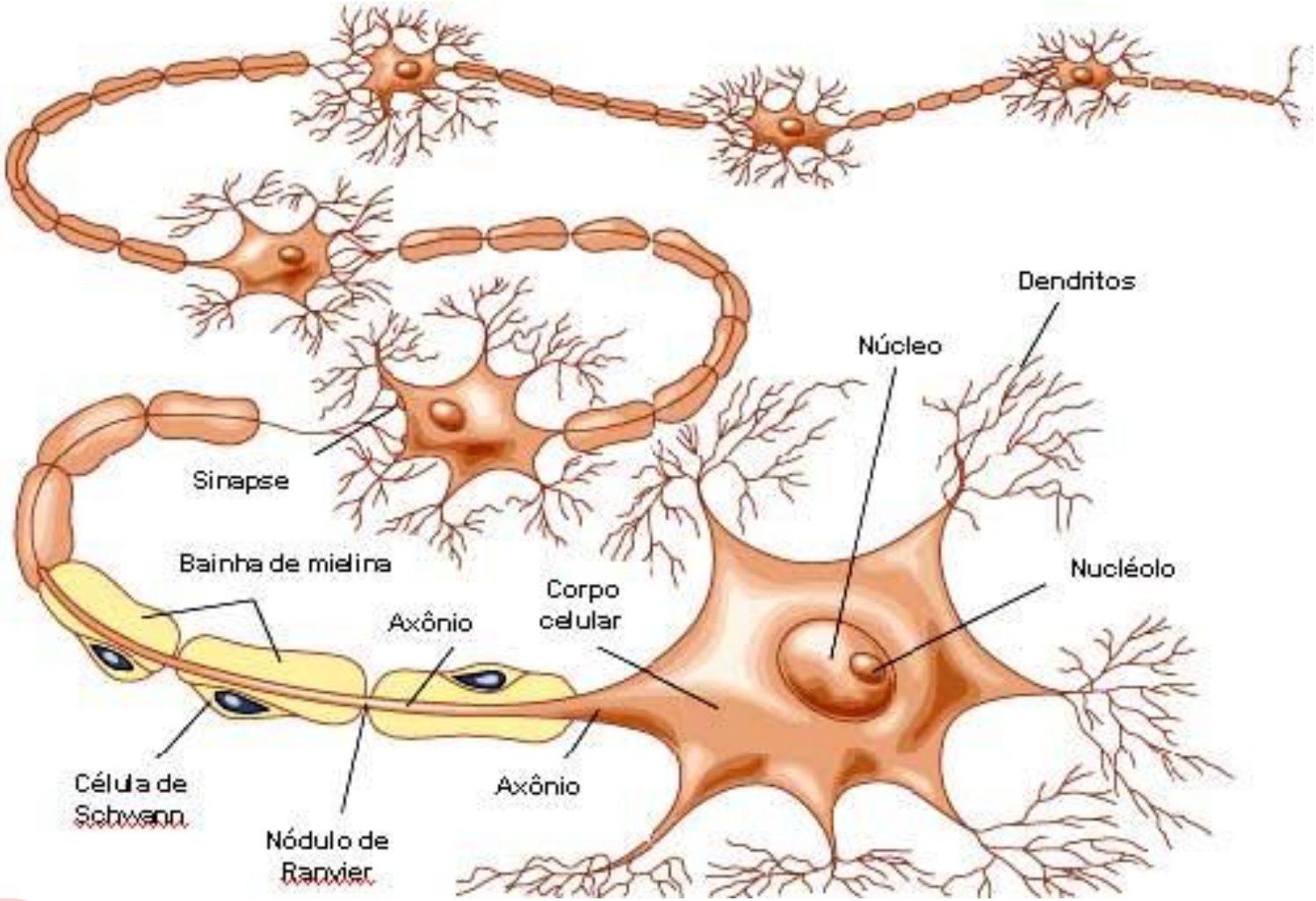
decom departamento de computação

Redes Neurais Biológicas

- Velocidade de Operação:
  - Computadores: nanosegundos (10^{-9}s).
  - Neurônios: milisegundos ($^{10-3}$s).

- Cérebro: rede de neurônios massivamente paralela.
  - 10 bilhões de neurônios.
  - 60 trilhões de interconexões.

Redes Neurais Biológicas

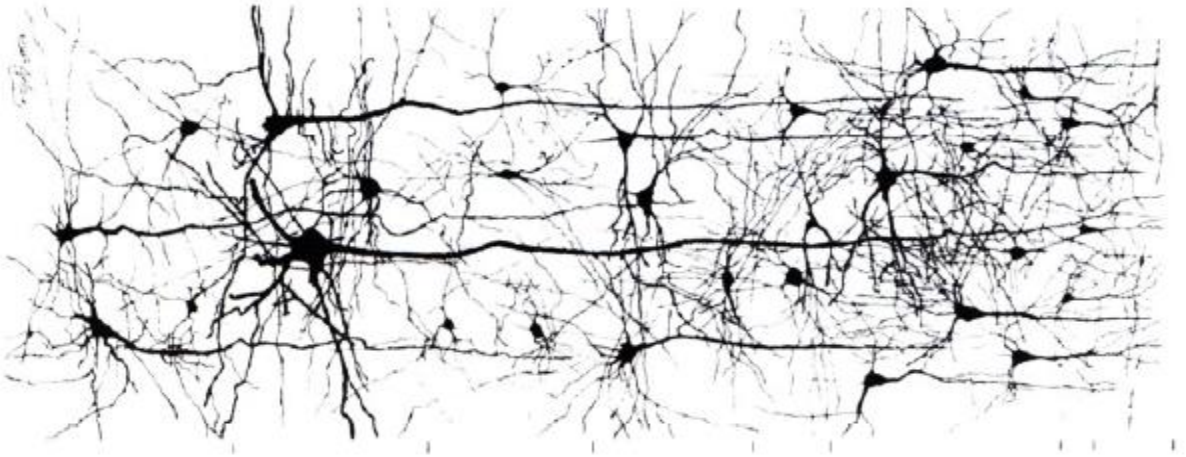


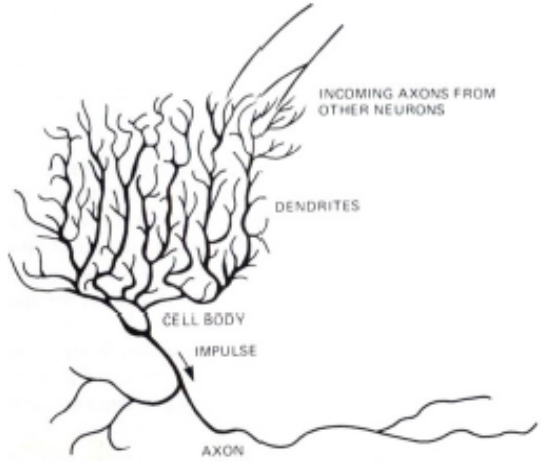
UFOP
Universidade Federal de Ouro Preto

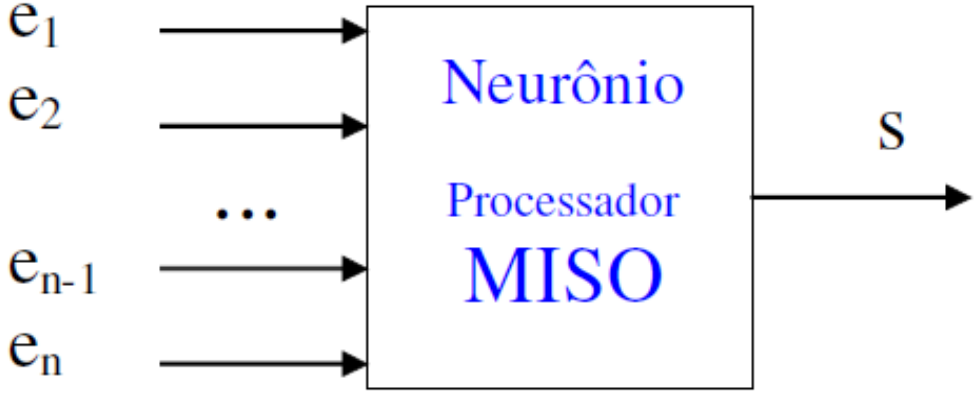
SEVA iMobilis

decom
departamento de computação

Redes Neurais Biológicas







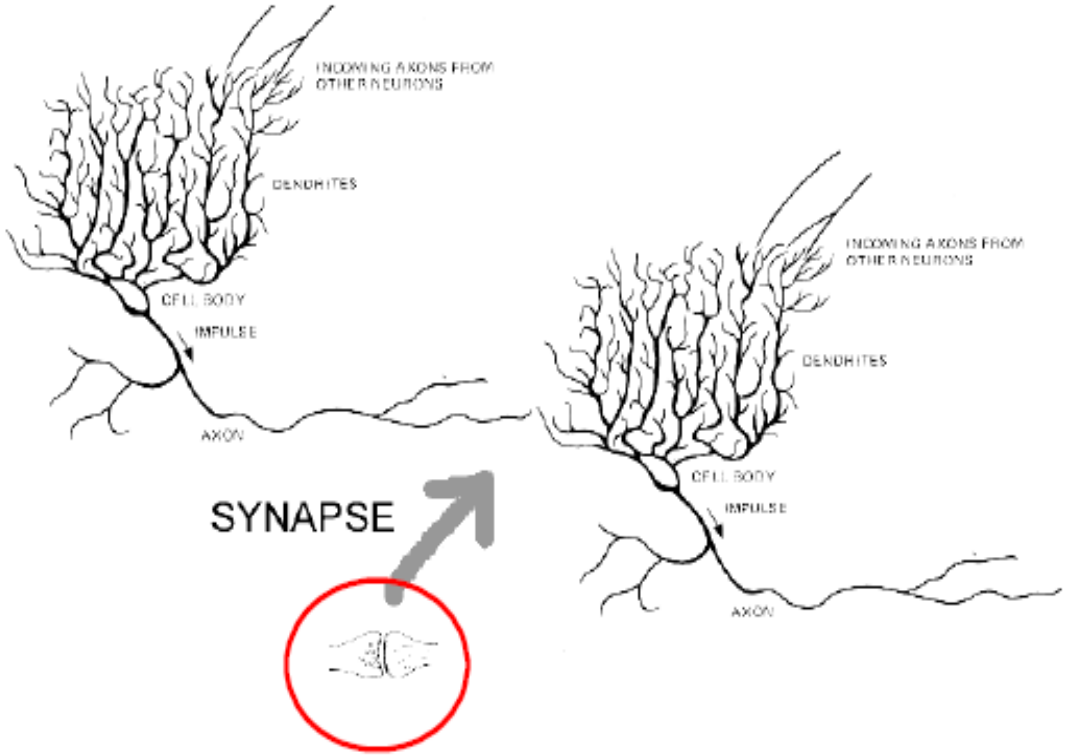
UFOP
Universidade Federal de Ouro Preto

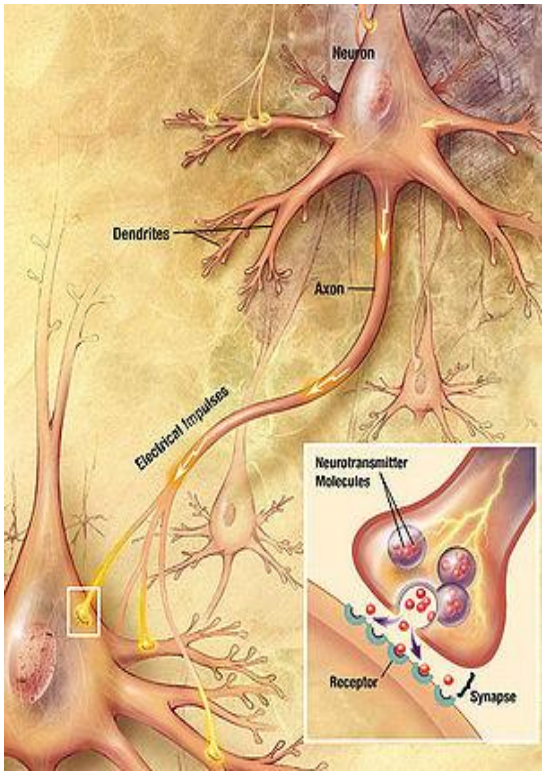
SEVA iMobilis

decom
departamento de computação

Redes Neurais Biológicas

- Comunicação entre neurônios:



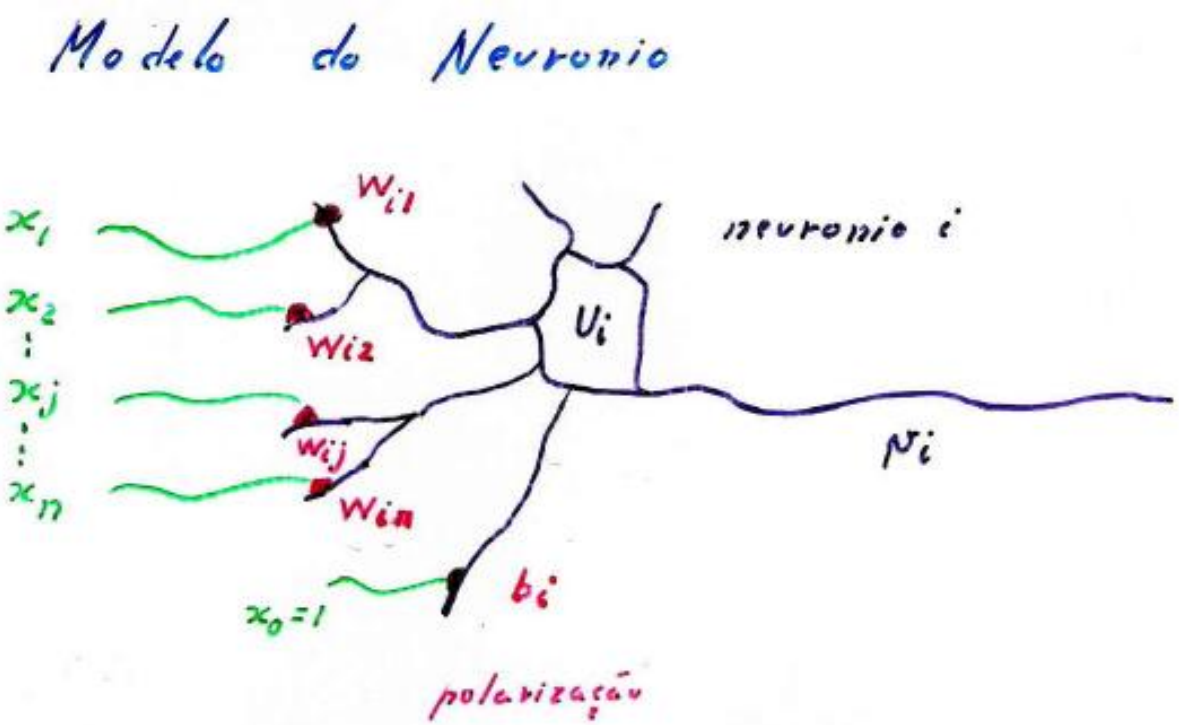


UFOP
Universidade Federal de Ouro Preto

SEVA iMobilis

decom
departamento de computação

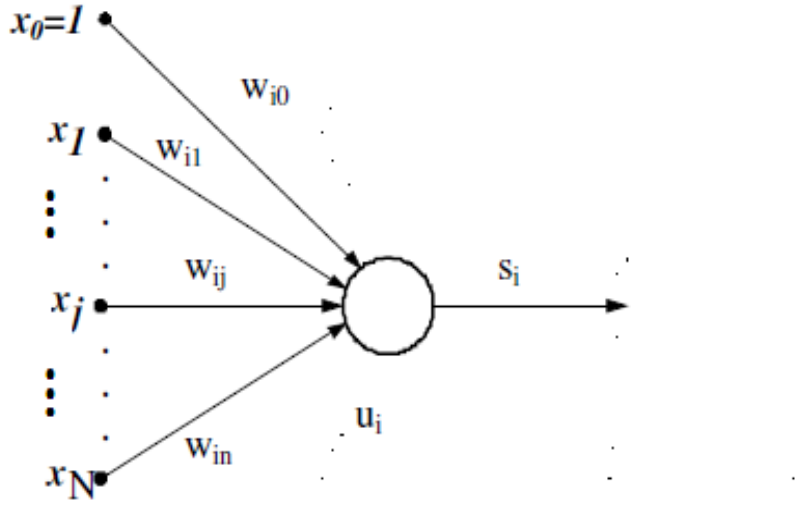
Modelo de Neurônio

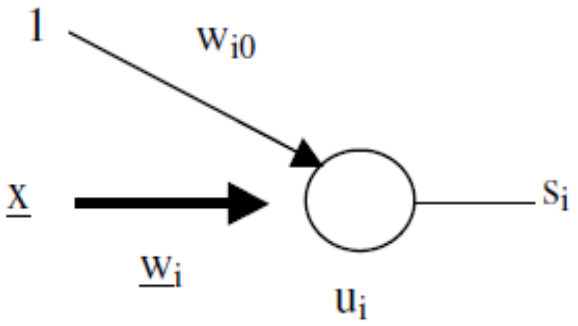


$$U_i = \sum_{j=1}^{n} w_{ij} x_j + b_i = \underline{w}_i^t \underline{x} + b_i$$

$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \qquad \underline{w}_i = \begin{bmatrix} w_{i1} \\ w_{i2} \\ \vdots \\ w_{in} \end{bmatrix}$$

Modelo de Neurônio





$$u_i = \sum_{j=1}^{N} w_{ij}\, x_j + w_{i0} = \underline{w}_i^t\, \underline{x} + w_{i0}$$

$$s_i = s(u_i)$$

UFOP
Universidade Federal de Ouro Preto

SEVA iMobilis

decom
departamento de computação

Função de Ativação s(u)

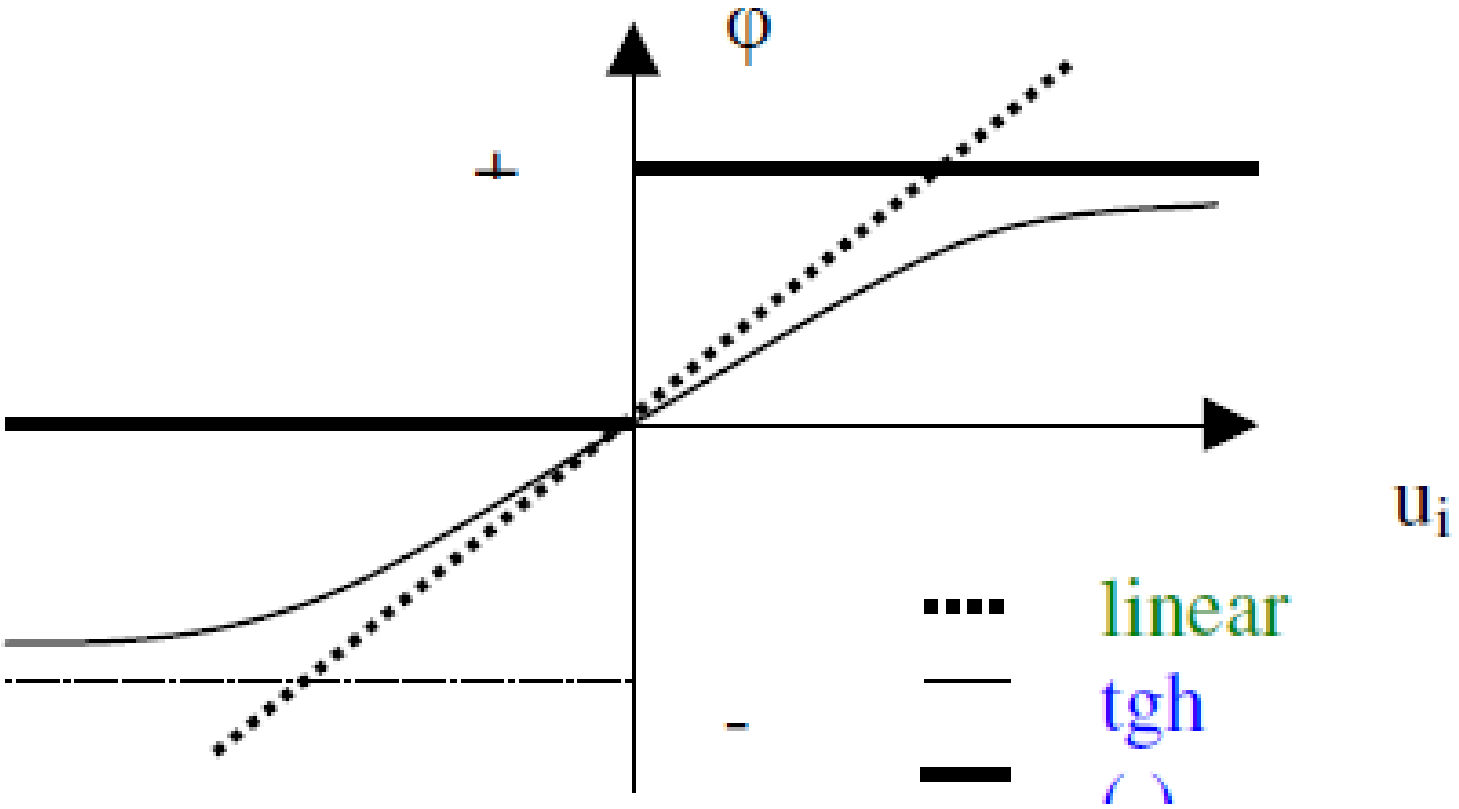
Neurônio	Função de ativação $s_i(u_i)$	Ganho linearizado $g_i(s_i) = ds_i / du_i$
linear	u_i	1
Não linear, tipo tgh	$\mathrm{tgh}(u_i) = \frac{1-e^{-2u_i}}{1+e^{-2u_i}}$	$1-s_i^2$
Não linear, Tipo binário	$\deg(u_i) = \begin{cases} 0 & \text{se } u_i < 0 \\ 1 & \text{se } u_i \geq 0 \end{cases}$	-

UFOP
Universidade Federal de Ouro Preto

SEVA iMobilis

decom
departamento de computação

Função de Ativação s(u)

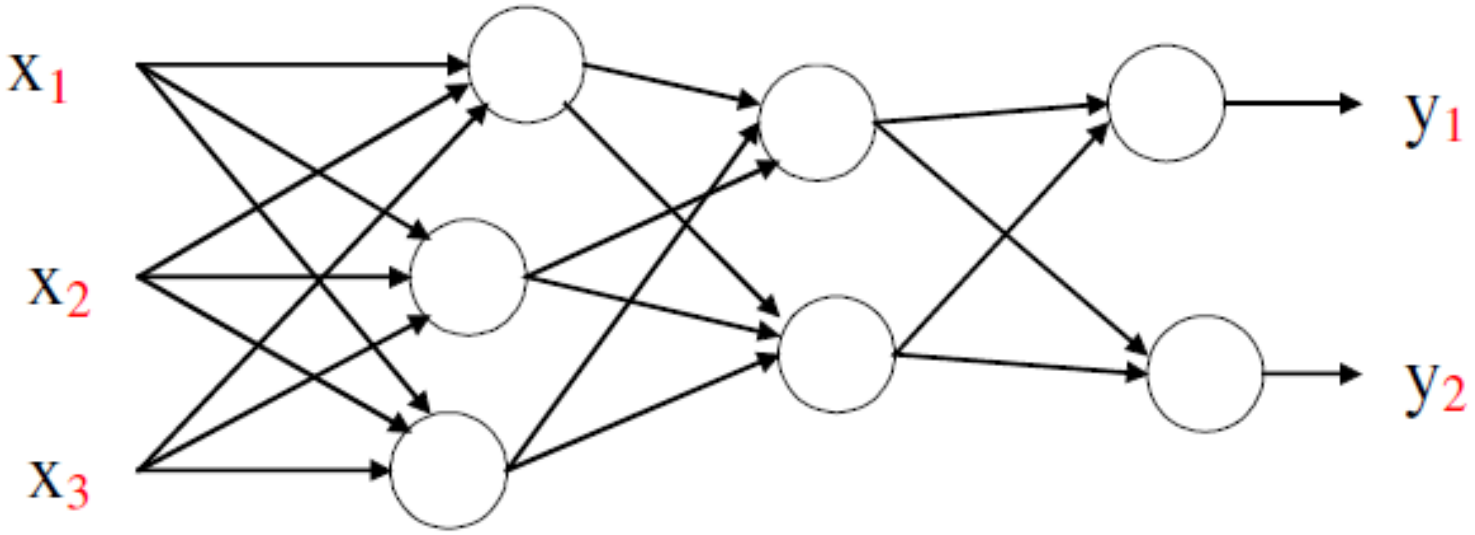


UFOP
Universidade Federal de Ouro Preto

SEVA iMobilis

decom
departamento de computação

Arquitetura da Rede



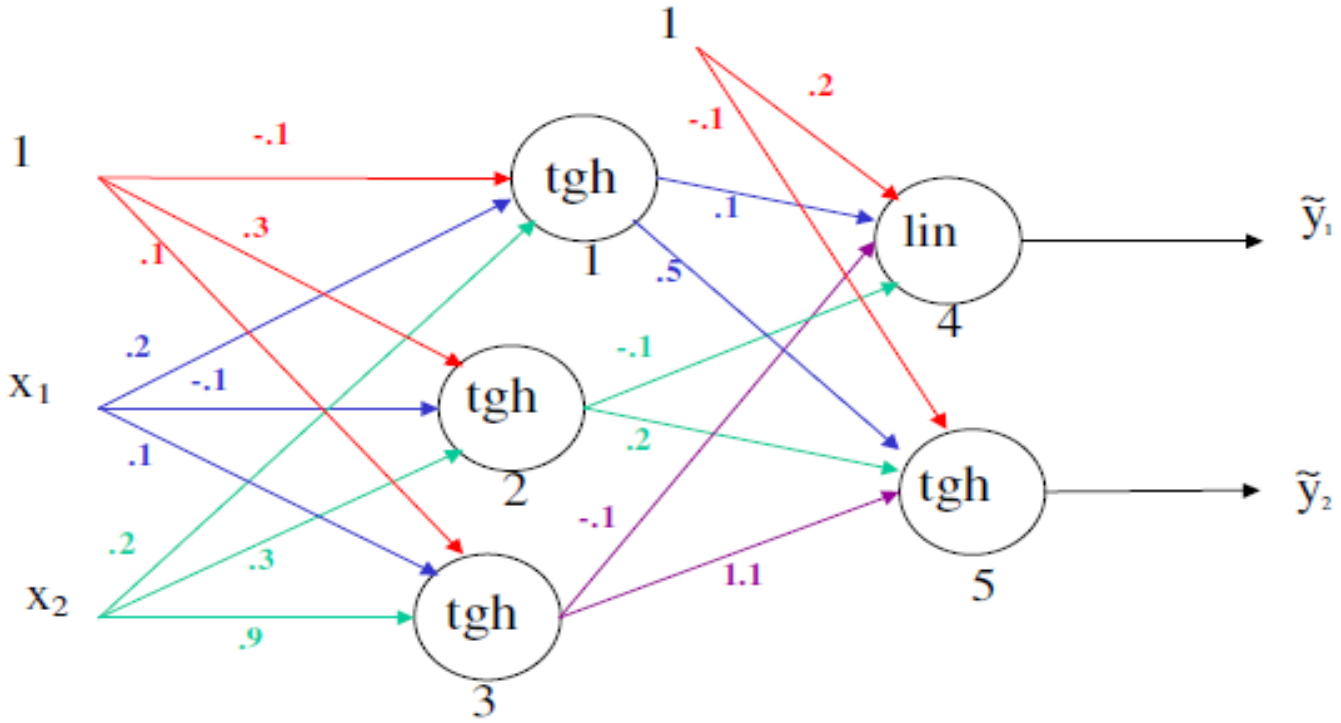
- Rede *feedfoward* (sem realimentação):
  - Estática.
  - Estruturalmente estável.

UFOP
Universidade Federal de Ouro Preto

SEVA iMobilis

decom
departamento de computação

Exemplo de Operação



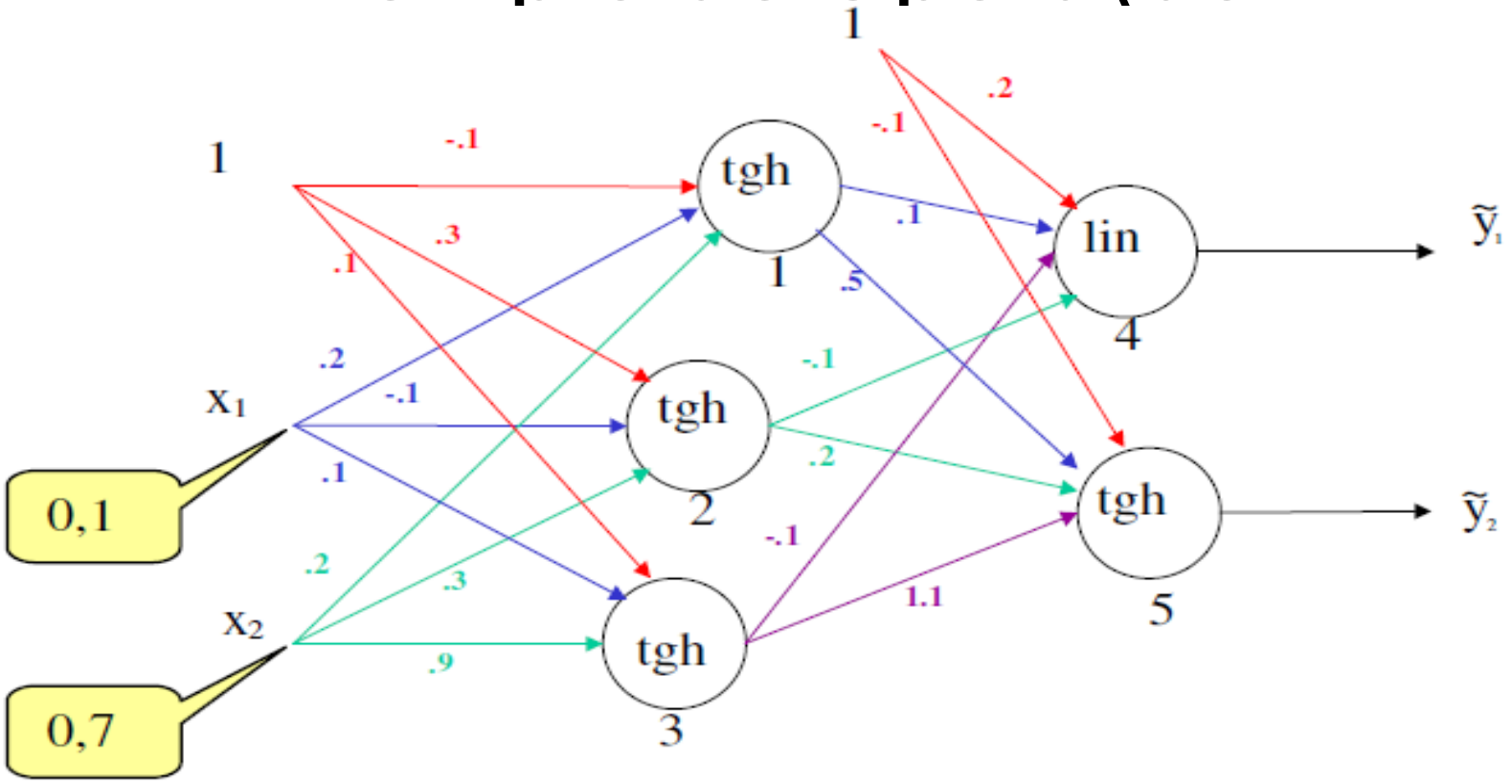
$$\underline{\mathbf{x}} = \begin{bmatrix} 0,1 \\ 0,7 \end{bmatrix} \qquad \tilde{\mathbf{y}} = ?$$

SEVA iMobilis

UFOP Universidade Federal de Ouro Preto

decom departamento de computação

Exemplo de Operação

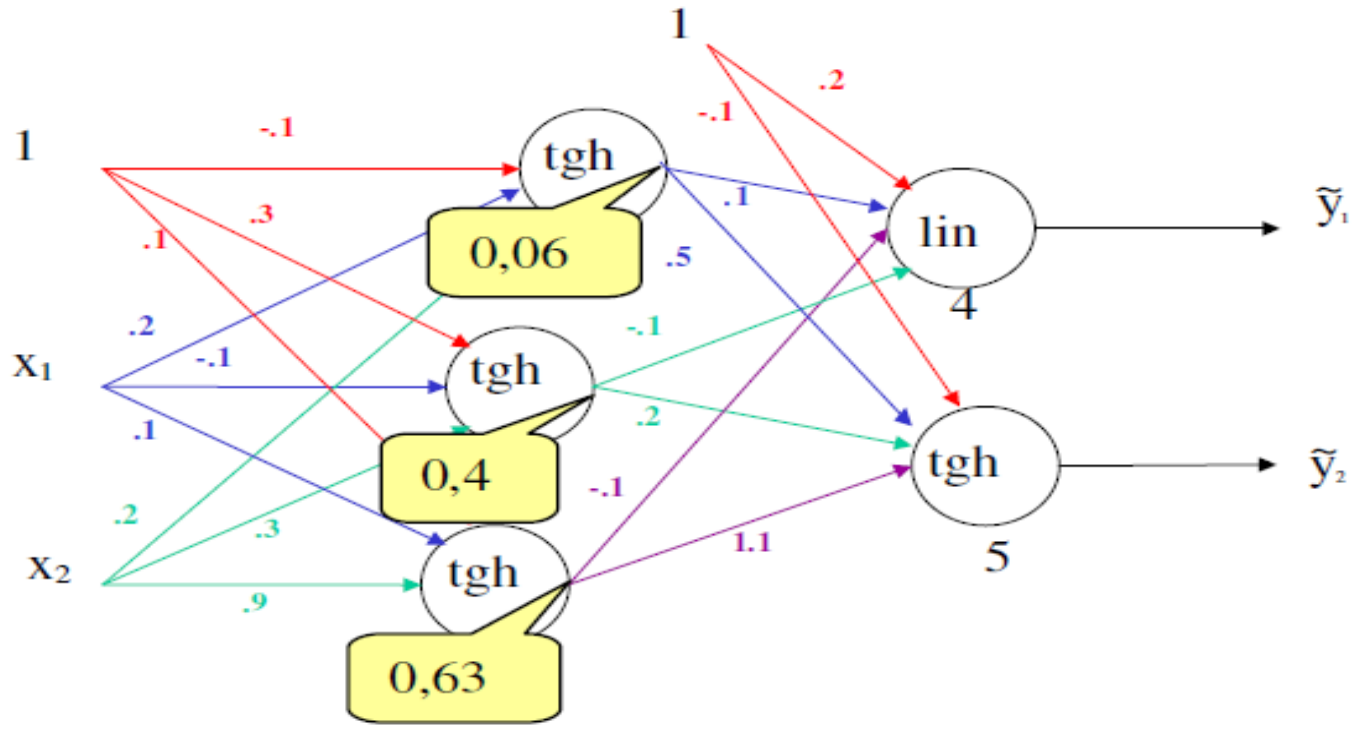


$$u_1 = -0,1 + (0,2)(0,1) + (0,2)(0,7) = 0,06$$

$$v_1 = \text{tgh}(0,06) = 0,06$$

$$v_2 = 0,46 \qquad v_3 = 0,63$$

Exemplo de Operação



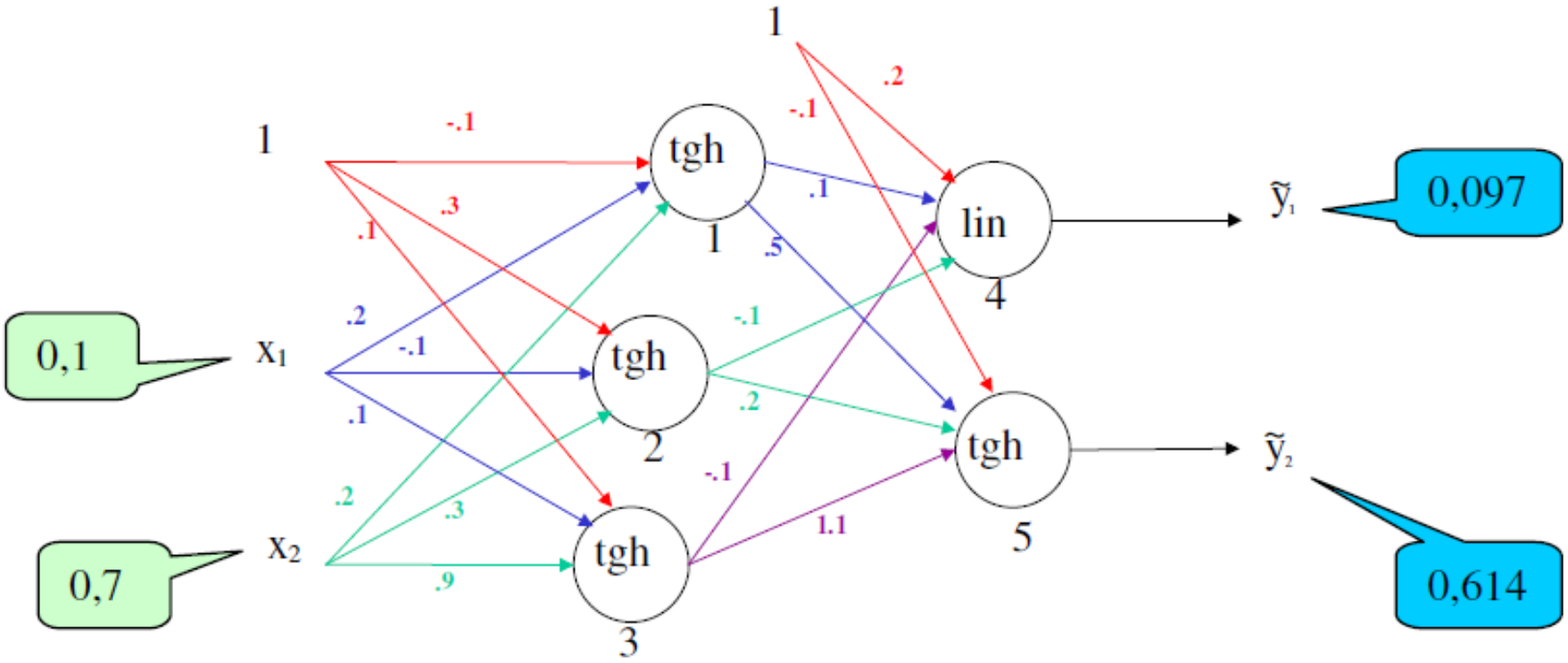
$$u_4 = 0{,}2 + (0{,}1)(0{,}06) + (-0{,}1)(0{,}46) + (-0{,}1)(0{,}63) = 0{,}097$$

$$v_4 = 0{,}097 \quad \text{(linear!)}$$

$$u_5 = -0{,}1 + (0{,}5)(0{,}06) + (0{,}2)(0{,}46) + (1{,}1)(0{,}63) = 0{,}715$$

$$v_5 = \text{tgh}(0{,}715) = 0{,}614$$

Exemplo de Operação



$$\underline{\mathbf{x}} = \begin{bmatrix} 0{,}1 \\ 0{,}7 \end{bmatrix} \qquad \tilde{\mathbf{y}} = \begin{bmatrix} 0{,}097 \\ 0{,}614 \end{bmatrix} \qquad \tilde{y} = \varphi(\underline{x})$$

Utilidades

- Problemas para os quais RNAs NÂO são adequadas:
  - Problemas que podem ser solucionados por uma sequência de passos bem definidos.
  - Problemas solucionáveis por algoritmos que possuem blocos de construção estáticos (a lógica não muda).
  - Problemas os quais é necessário saber como a solução foi derivada.

Utilidades

- Problemas para os quais RNAs são adequadas:
  - Reconhecimento de padrões.
  - Classificação.
  - Predição de séries.
  - Mineração de Dados.

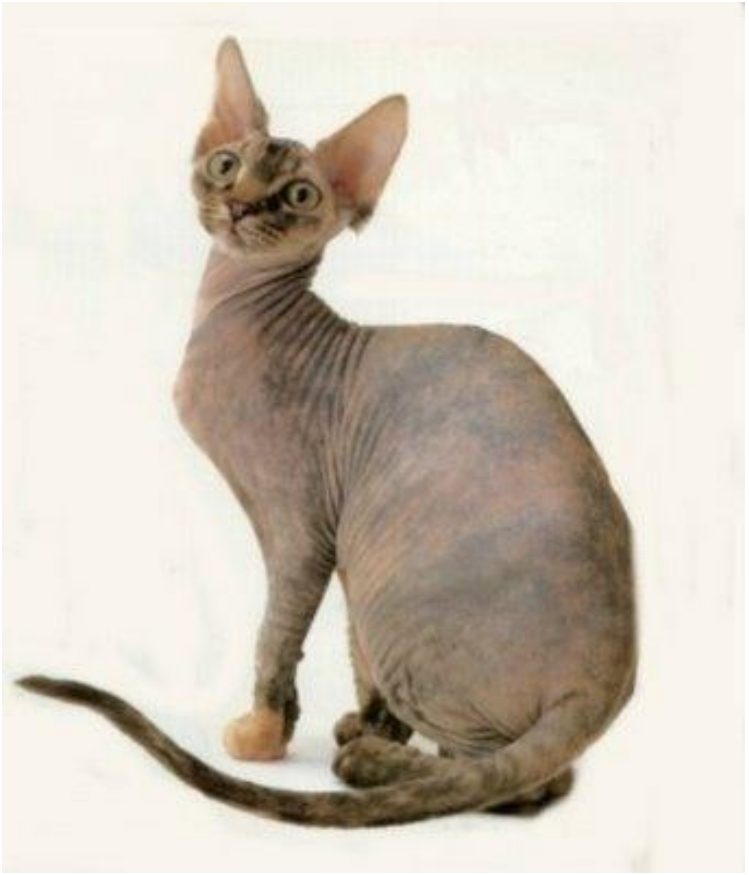
Utilidades e Capacidades

- Não-linearidade.
  - Interconexão de neurônios não-lineares.
  - Não-linearidade distribuída através da rede.
- Mapeamento das entradas e saídas.
  - Aprendizagem com um “professor”.
- Adaptabilidade.
  - Pode adaptar os parâmetros livres. Eles se modificam no ambiente circundante.

Exemplo de Associações

Exemplo

Exemplo de Associações



Utilidades e Capacidades

- Resposta Evidencial.
  - Decisão com uma medida de “confiança”.
- Tolerância a Falhas.
  - Degradação “suave”.
- Implementabilidade em (VLSI).
  - *Very-Large-Scale Integration* .

Treinamento e Aprendizagem

- Treinamento é o processo iterativo no qual os pesos das interconexões são ajustados para que a rede retorne uma saída apropriada.

- Supervisionado.
  - É dado um conjunto de entrada e um conjunto de saída ou respostas desejadas.

- Não Supervisionado.
  - É dado apenas um conjunto de entrada.

- Modelos híbridos.

Validação

- Estágio em que os resultados retornados pela RNA são avaliados.

- Em geral, utiliza-se um conjunto de dados de treinamento diferente do conjunto de dados de validação.

Implementando RNAs

- Heaton Research.
  - Fornece um framework para se trabalhar com RNAs na linguagem Java (gratuito).
  - O livro introdutório da disciplina ($$$) apresenta os princípios básicos de RNAs e codificações em Java.

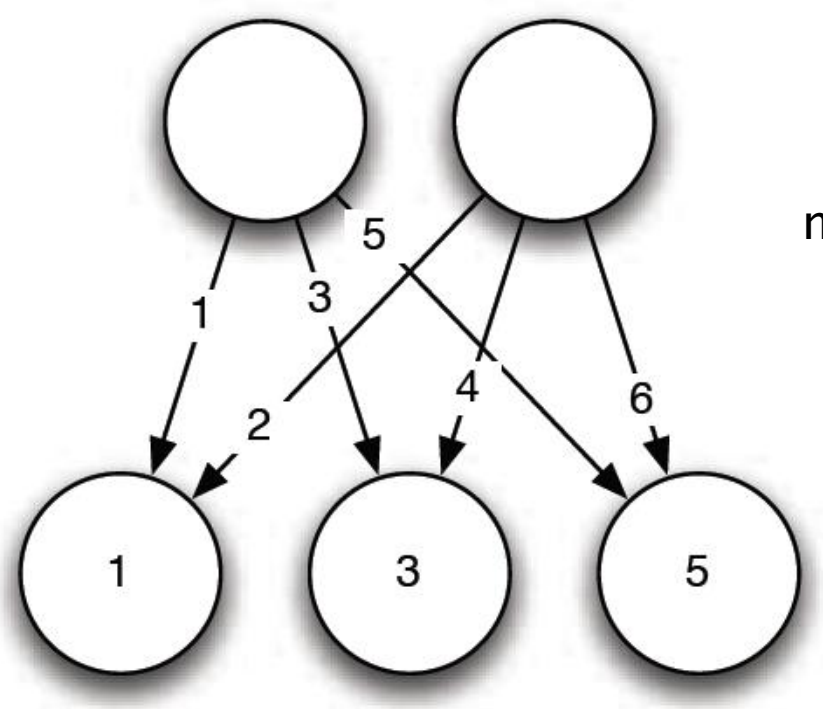
UFOP
Universidade Federal de Ouro Preto

SEVA iMobilis

decom
departamento de computação

Como implementar?

- Matrizes são a base!



$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

matriz de pesos

$$\begin{bmatrix} w & w \\ w & w \\ w & w \\ t & t \end{bmatrix}$$

matriz de pesos e limiares

UFOP
Universidade Federal de Ouro Preto

SEVA iMobilis

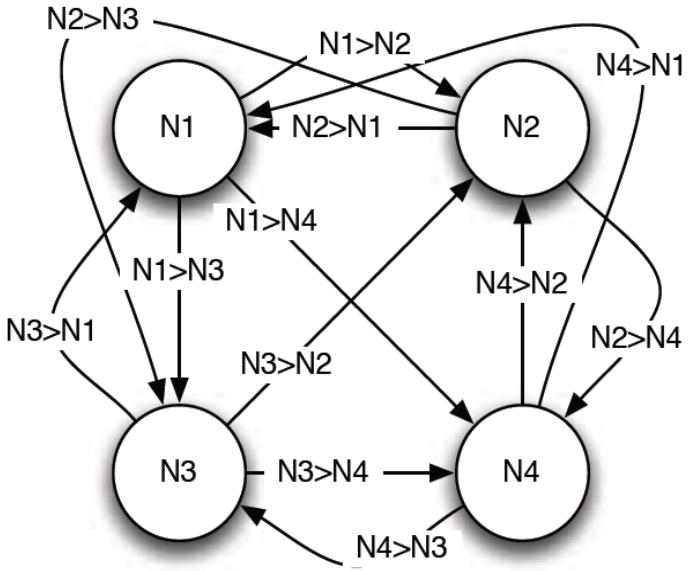
decom
departamento de computação

Classes relacionadas a matrizes

Class	Purpose
BiPolarUtil	A utility class to convert between Boolean and bipolar numbers.
Matrix	Holds a matrix.
MatrixMath	Performs mathematical operations on a matrix.

Hopfield Neural Network

Figure 3.1: A Hopfield neural network with 12 connections.



	Neuron 1 (N1)	Neuron 2 (N2)	Neuron 3 (N3)	Neuron 4 (N4)
Neuron 1(N1)	(n/a)	N2->N1	N3->N1	N4->N1
Neuron 2(N2)	N1->N2	(n/a)	N3->N2	N3->N2
Neuron 3(N3)	N1->N3	N2->N3	(n/a)	N4->N3
Neuron 4(N4)	N1->N4	N2->N4	N3->N4	(n/a)

UFOP
Universidade Federal de Ouro Preto

SEVA iMobilis

decom
departamento de computação

Hopfield Neural Network

- Uma rede neural que reconhece o padrão 0101.

	Neuron 1 (N1)	Neuron 2 (N2)	Neuron 3 (N3)	Neuron 4 (N4)
Neuron 1 (N1)	0	-1	1	-1
Neuron 2 (N2)	-1	0	-1	1
Neuron 3 (N3)	1	-1	0	-1
Neuron 4 (N4)	-1	1	-1	0

0 1 0 1

0 -1 1 -1

N1 = -1 + -1 = -2

N2 = 0 + 1 = 1

N3 = -1 + -1 = -2

N4 = 1 + 0 = 1

Como treinar uma Hopfield NN?

Equation 3.2: Binary to Bipolar

$$f(x) = 2x - 1$$

Equation 3.4: Input Matrix

$0 = -1$

$1 = 1$

$0 = -1$

$1 = 1$

$$\begin{bmatrix} -1 \\ 1 \\ -1 \\ 1 \end{bmatrix}$$

Equation 3.5: Input Matrix Transposed

$$\begin{bmatrix} -1 & 1 & -1 & 1 \end{bmatrix}$$

$M \times M^T = R$

Equation 3.6: Resulting Matrix

$$\begin{bmatrix} 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 \end{bmatrix}$$

$R - I = C$

Equation 3.7: Contribution Matrix

$$\begin{bmatrix} 0 & -1 & 1 & -1 \\ -1 & 0 & -1 & 1 \\ 1 & -1 & 0 & -1 \\ -1 & 1 & -1 & 0 \end{bmatrix}$$

Classe HopfieldNetwork

Method Name	Purpose
getMatrix	Accesses the neural network's weight matrix.
getSize	Gets the size of the neural network.
present	Presents a pattern to the neural network.
train	Trains the neural network on a pattern.