

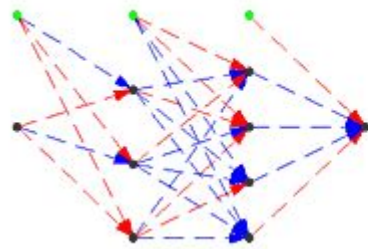
# Redes Neurais

## Retropropagação

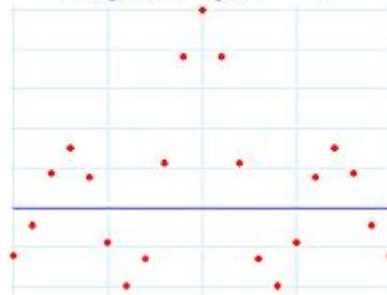
# REDES NEURAIS - BP



Neural Network  $\sum$ Weights 0.5



Output at Epoch 0



Mean Square Error 0.0000





## Introdução

- As redes neurais pertencem a um ramo da inteligência artificial que busca inspiração na fisiologia do cérebro para construir dispositivos inteligentes.
- Ela parte do pressuposto que se construirmos mecanismos que simulem o funcionamento cérebro, então estes mecanismos serão capazes de aprender e desenvolver suas representações internas do conhecimento.
- Uma comparação entre o cérebro e o computador sugere que este pode ser um caminho promissor:



## Introdução

|                       | <i>Cérebro</i>        | <i>Computador</i>        |
|-----------------------|-----------------------|--------------------------|
| Número de Componentes | 100 bilhões           | milhões                  |
| Velocidade            | KHz                   | GHz                      |
|                       | Ciclo de milisegundos | Ciclo de nanosegundos    |
| Arquitetura           | Massivamente Paralela | Basicamente sequencial   |
| Robustez              | Tolerante a falhas    | Pouco tolerante a falhas |



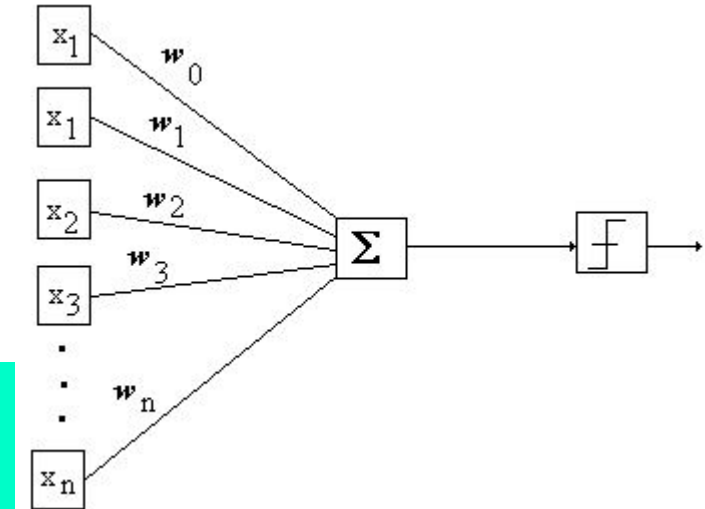
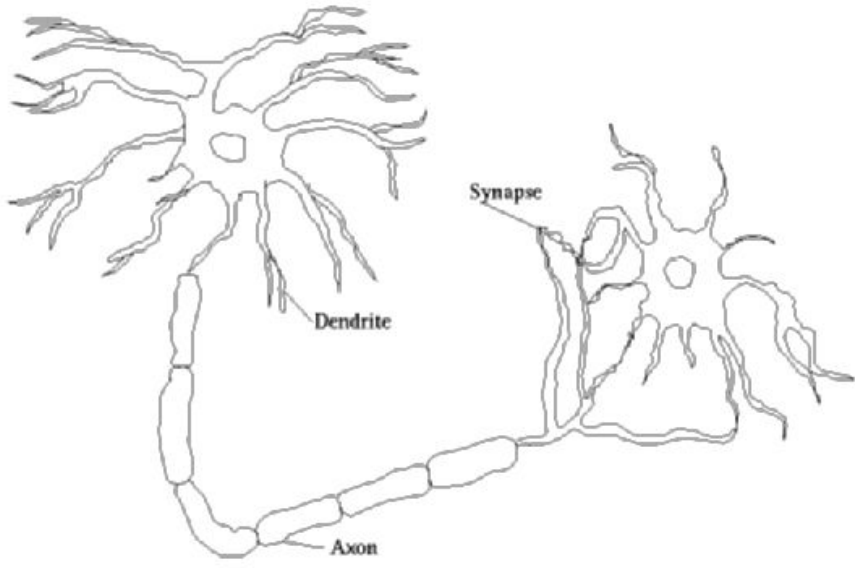
## Histórico

- As pesquisas sobre computação neural foram iniciadas por volta de 1940 (McCulloch e Pitts em 1943).
- No entanto, poucos estudos ocorreram até a introdução dos perceptron por Rosenblatt 1958.
- Em 1969 Minsky e Papert mostraram que as redes neurais desenvolvidas até aquele período eram capazes de computar apenas problemas linearmente separáveis e portanto tinham pouca utilidade para modelar o comportamento humano.
- As redes neurais de múltiplas camadas eram capazes de resolver problemas não linearmente separáveis, mas não possuíam algoritmos de aprendizado.
- Nos anos 80 as redes neurais ressurgiram com força devido ao surgimento de algoritmos de aprendizado para redes de múltiplas camadas.



## Perceptrons (Rosenblatt -1962)

Os perceptrons foi um dos primeiros modelos de redes neurais e foi inspirado no neurônio.

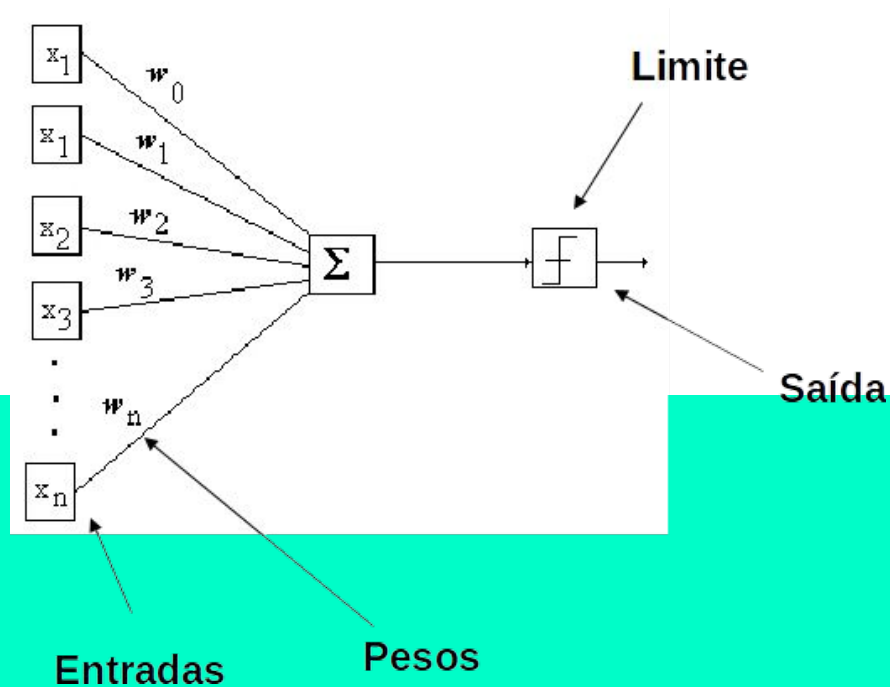


# Redes Neurais -BP



## Perceptrons (Rosenblatt -1962)

Os perceptrons foi um dos primeiros modelos de redes neurais e foi inspirado no neurônio.





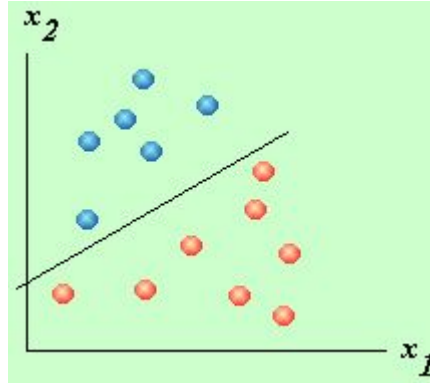
## Perceptrons (Rosenblatt -1962)

- Adequadas para problemas separáveis linearmente
- Vários perceptrons podem ser combinados para computar funções mais complexas
- O algoritmo de aprendizagem por perceptrons é um algoritmo de busca.
- A redução pelo gradiente é idêntica à estratégia de subida da encosta.





## Perceptrons (Rosenblatt -1962)





## Perceptrons (Aprendizado)

- O perceptrons são atraentes por possuírem um algoritmo de aprendizado.
- Com isso a rede pode aprender e melhorar seu desempenho.
- O algoritmo de aprendizado funciona por ciclos. Em cada ciclo é aplicado um conjunto de exemplos e calculado erro em relação ao resultado esperado.
- O erro é usado para calcular novos pesos e então um novo ciclo é aplicado.
- A fase de aprendizado termina quando a rede consegue exibir respostas corretas não só para os exemplos como um conjunto de testes.



## Perceptrons (Aprendizado)

### Perceptrons

Atribuir pesos aleatórios para as conexões;

Enquanto erro maior que erro mínimo e faça

Para cada conjunto de treinamento  $X_i$  faça

Aplicar cada conjunto de treinamento  $X_i$  e obter o erro

Fim para

Ajustar os pesos usando a fórmula

$$w_{t+1} = w_t + \eta \nabla J(w)$$

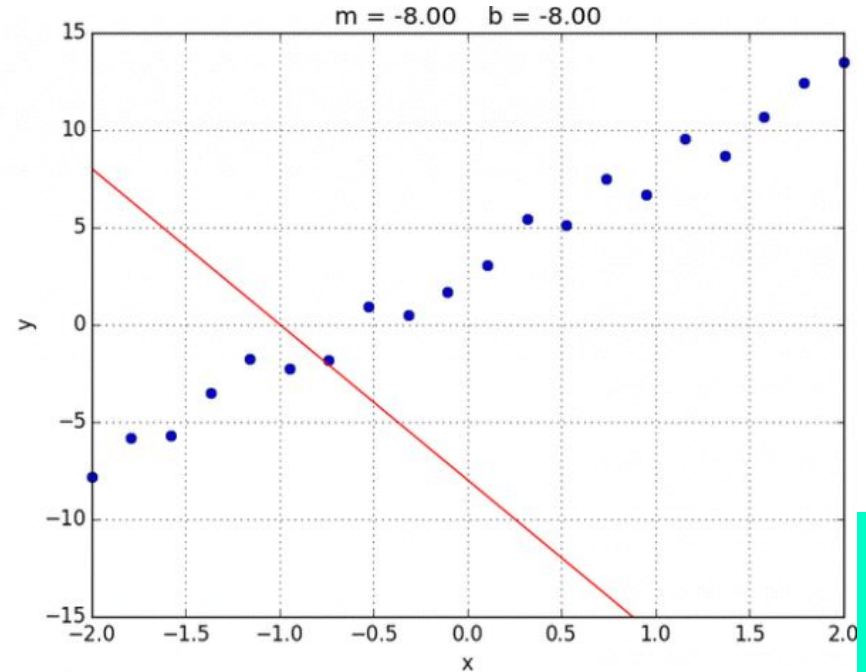
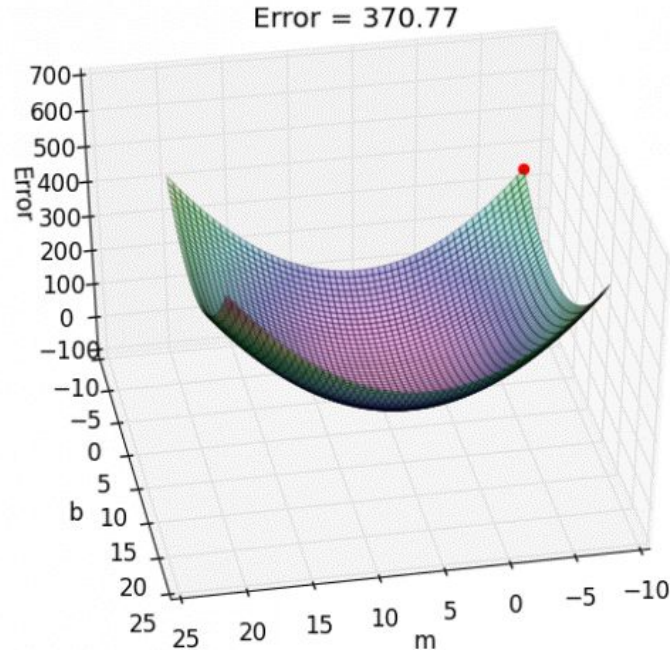
Fim enquanto

$\eta$  = taxa de aprendizado

$\nabla J(w)$  = Gradiente do erro



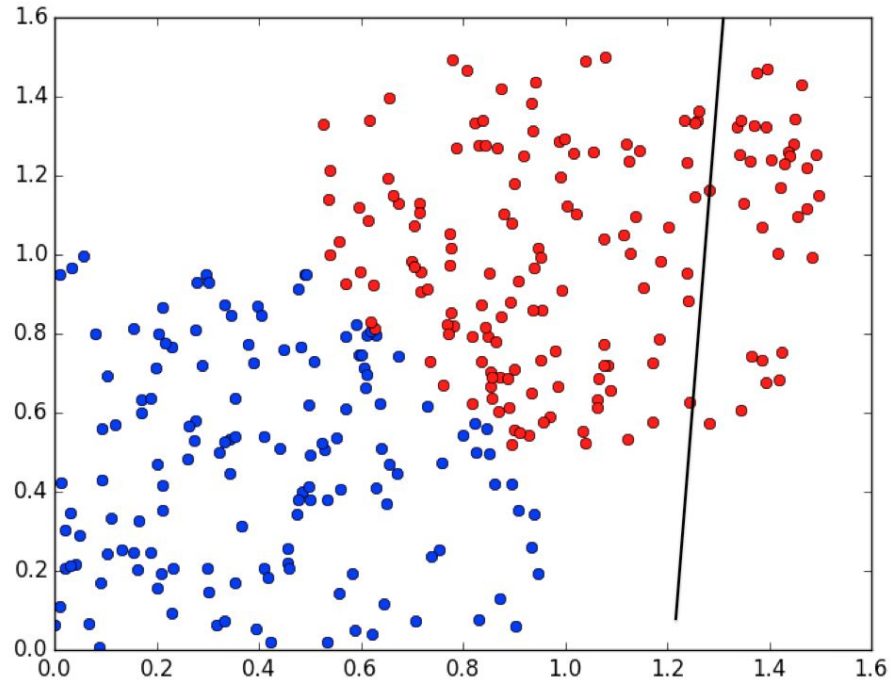
## Perceptrons (Aprendizado)



**O Teorema da Convergência de Perceptrons (Rosenblatt, 1962): não há mínimos locais que não correspondam ao mínimo global.**



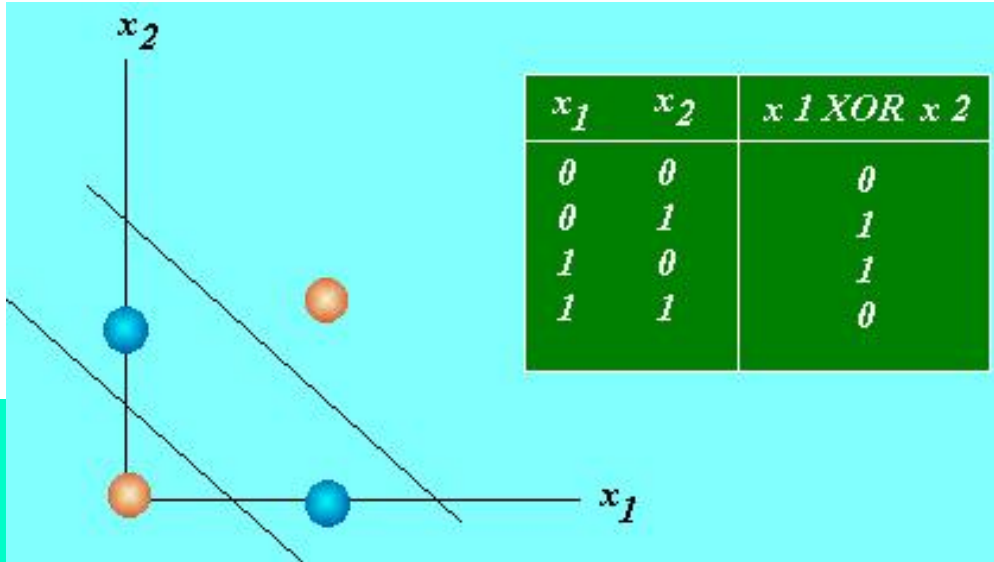
## Perceptrons (Aprendizado)





## Perceptrons (Rosenblatt -1962)

### O problema do **ou** exclusivo

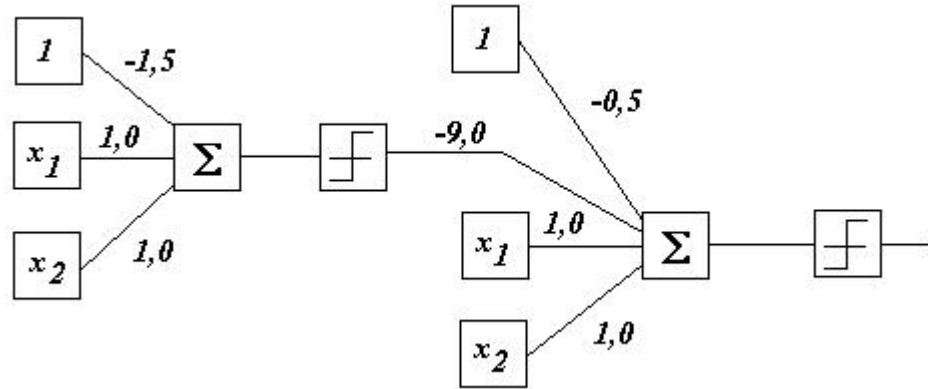


Os perceptrons não são capazes de separar os dois conjuntos da figura pois ela lida apenas com problemas linearmente separáveis.



## Perceptrons (Rosenblatt -1962)

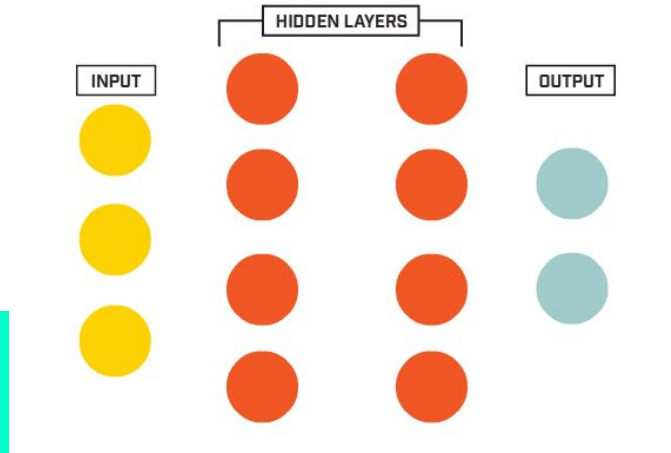
Um perceptron multicamada que soluciona o problema do **XOR**



No entanto, a inexistência de um algoritmo de aprendizado para os perceptrons multicamada fez com que o financiamento de pesquisas em redes neurais fossem interrompidos.



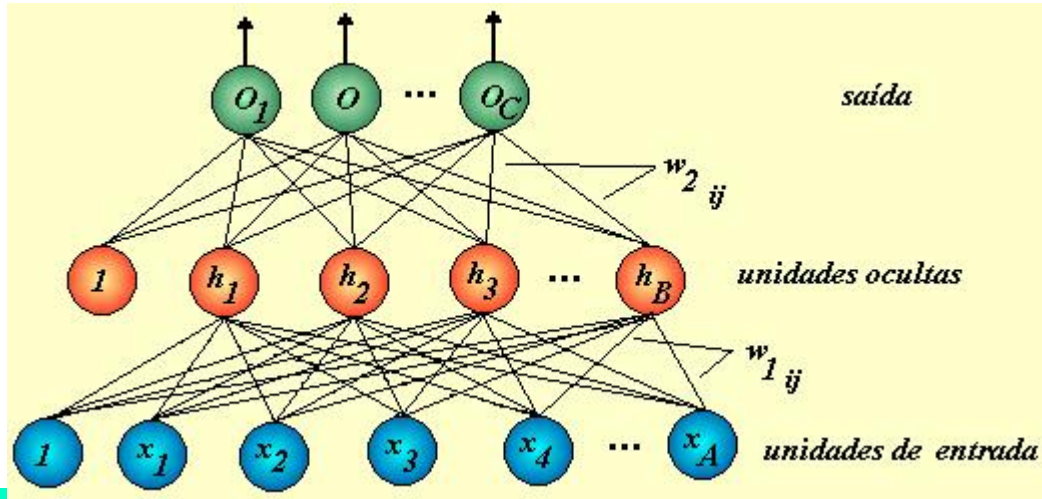
# Redes de retropropagação







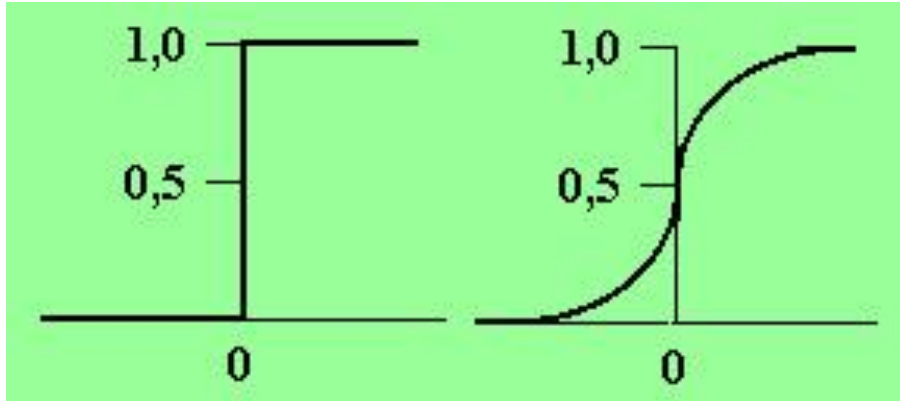
## Redes de retropropagação



O desenvolvimento de um algoritmo de aprendizado para as redes multicamadas contornou este problema e fez ressurgir o interesse em redes neurais.



## Redes de retropropagação



$$\text{saída} = 1 / (1 + e^{-\text{soma}})$$

O algoritmo de aprendizado das redes multicamadas se baseia na derivação da função de saída. Portanto, é preciso usar uma função derivável, como a função sigmóide.



# Redes de retropropagação (Aprendizado)

Batch mode ou  
Online  
(estocástico)?

**Repita**

**Para** cada exemplo (Ou grupo) **e faça**

$\hat{y} = \text{saída\_da\_rede}(e)$  # calcula a saída

$E_j^e = l(y_j, \hat{y}_j)$  # calcula o Erro

# Distribui a culpa pelo erro

**Para** cada nó na camada de saída **faça**

$$\Delta_j = E_j^e \sigma'(z_j)$$

**Para** cada camada interior **faça**

**Para** cada nó  $i$  da camada **faça**

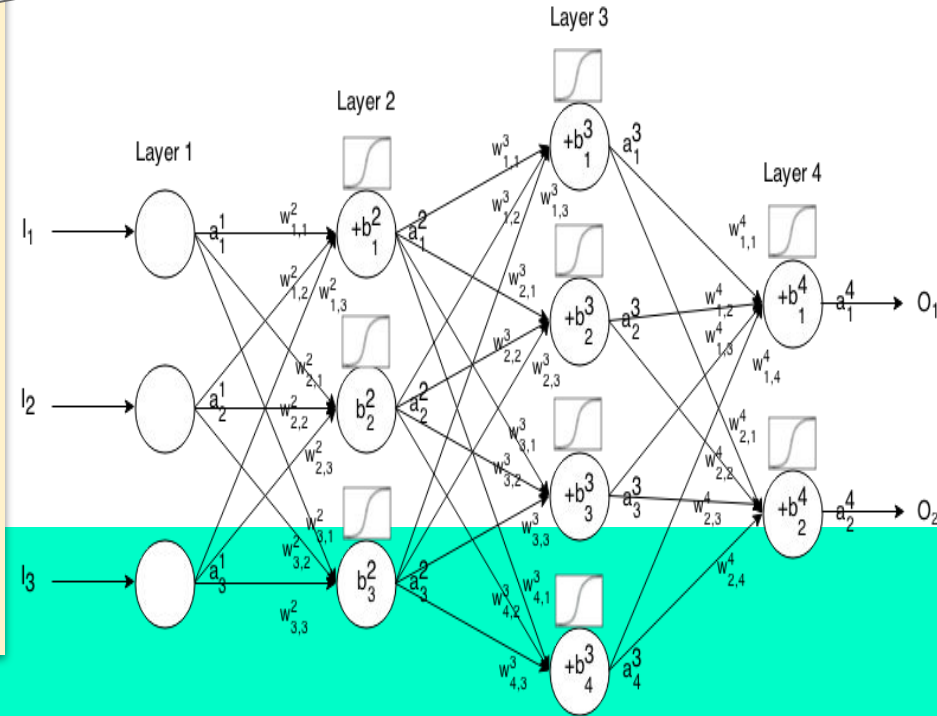
$$\Delta_i = \sigma'(z_i) \sum_j w_{j,i} \Delta_j$$

# Atualiza os pesos

**Para** cada peso  $w_{j,i}$  **faça**

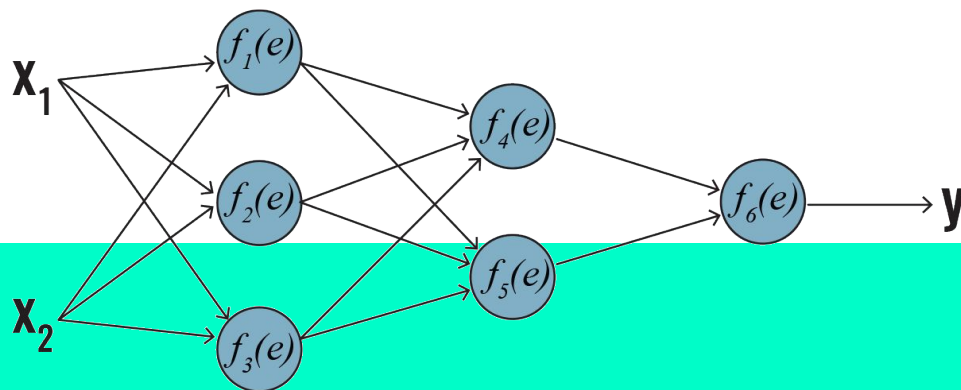
$$w_{j,i} = w_{j,i} + \alpha a_j \Delta_i$$

**até** a rede convergir





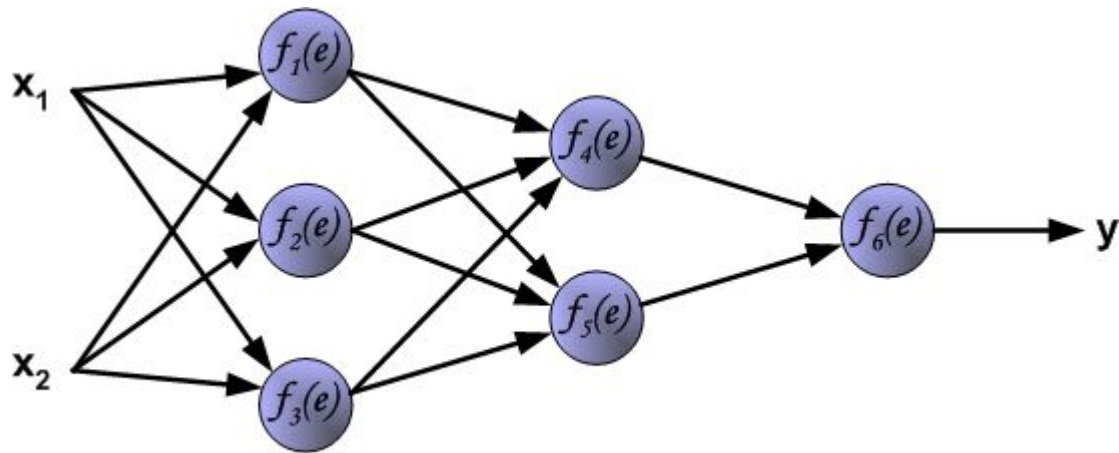
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992



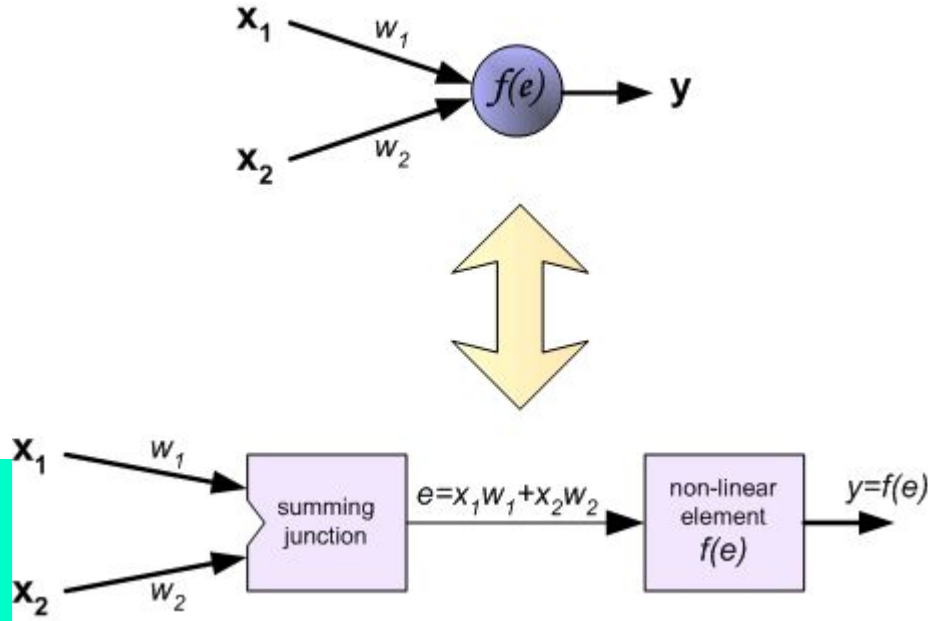
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992



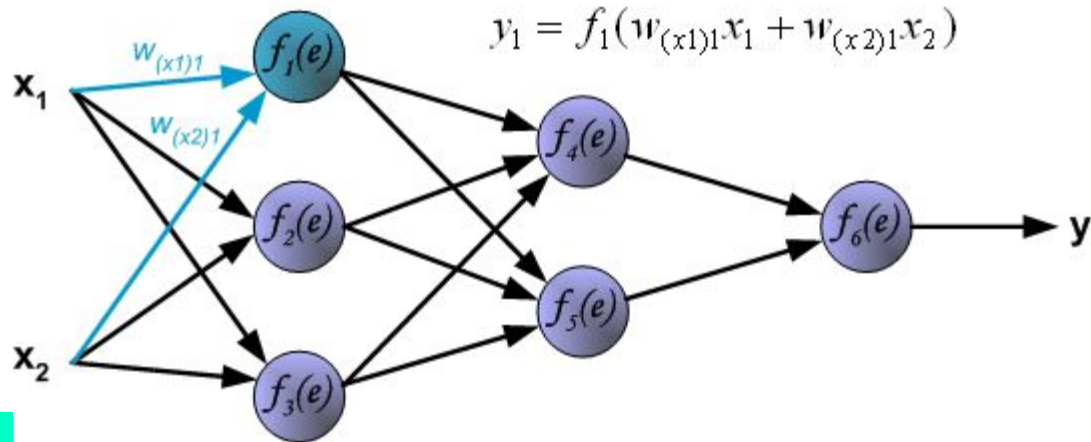
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992



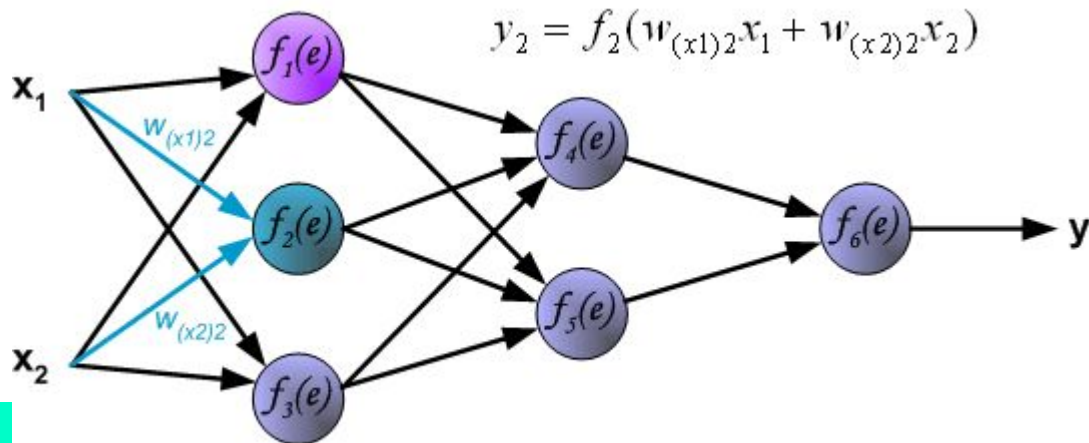
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992



## Redes de retropropagação

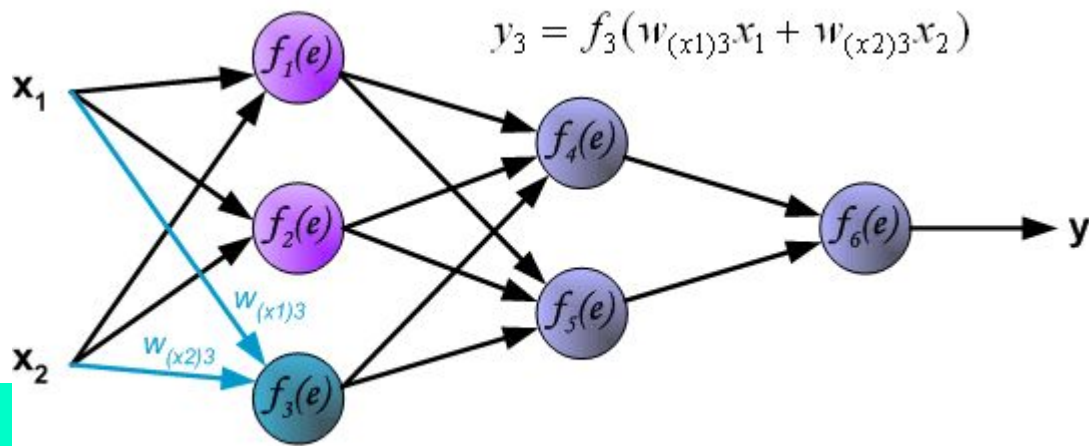


Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992





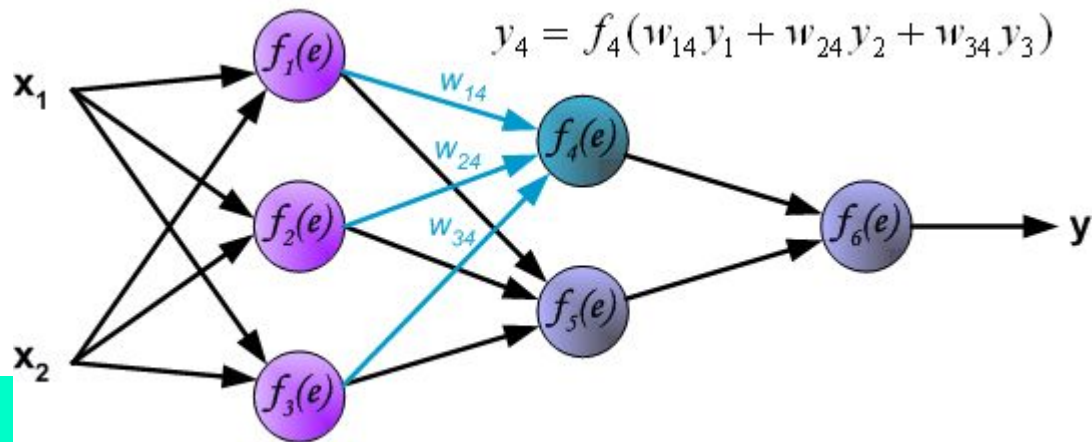
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992



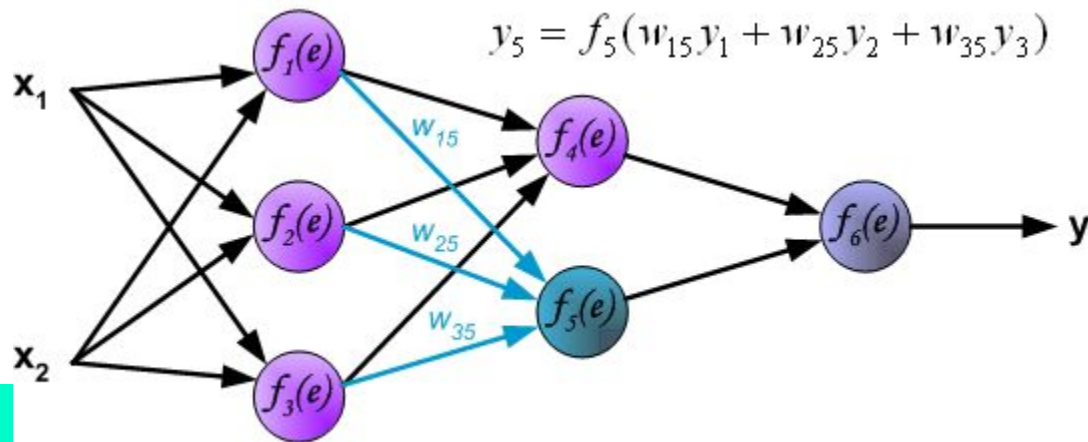
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992



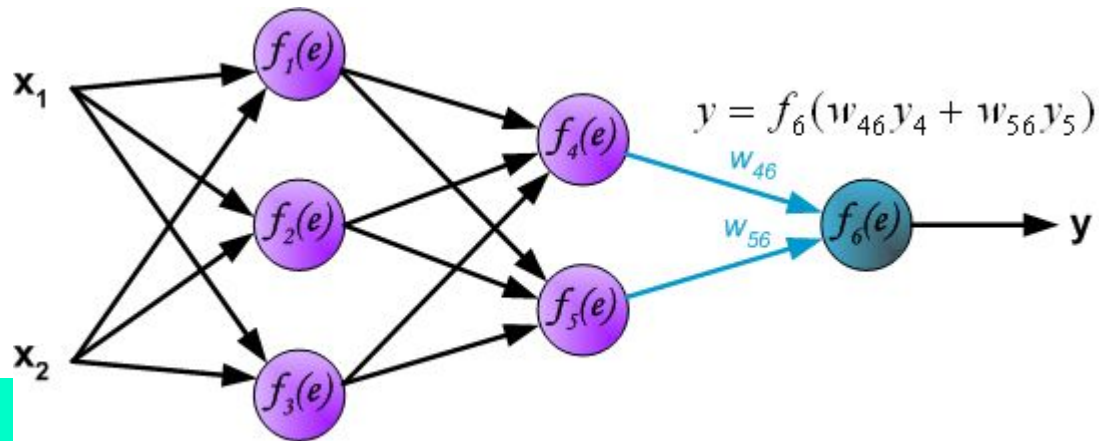
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992



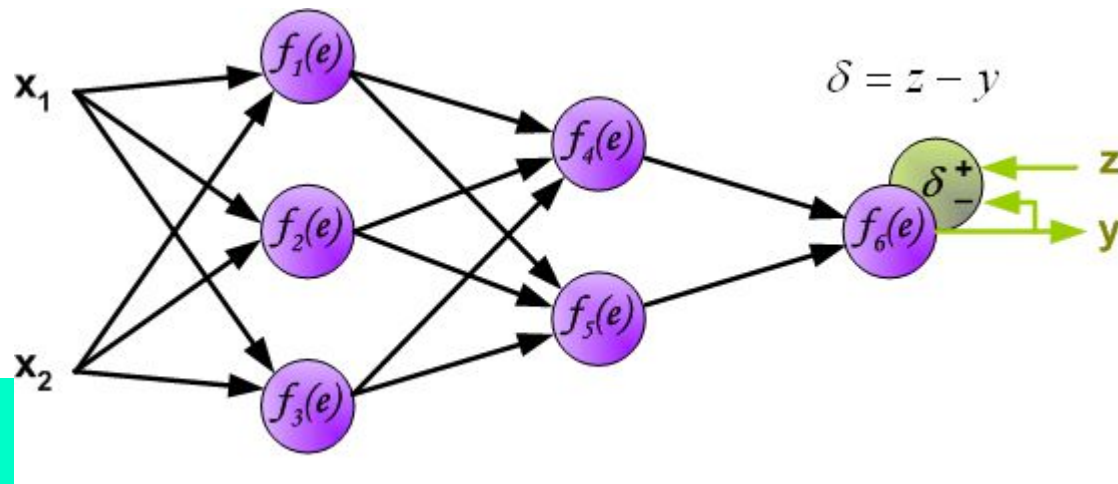
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992

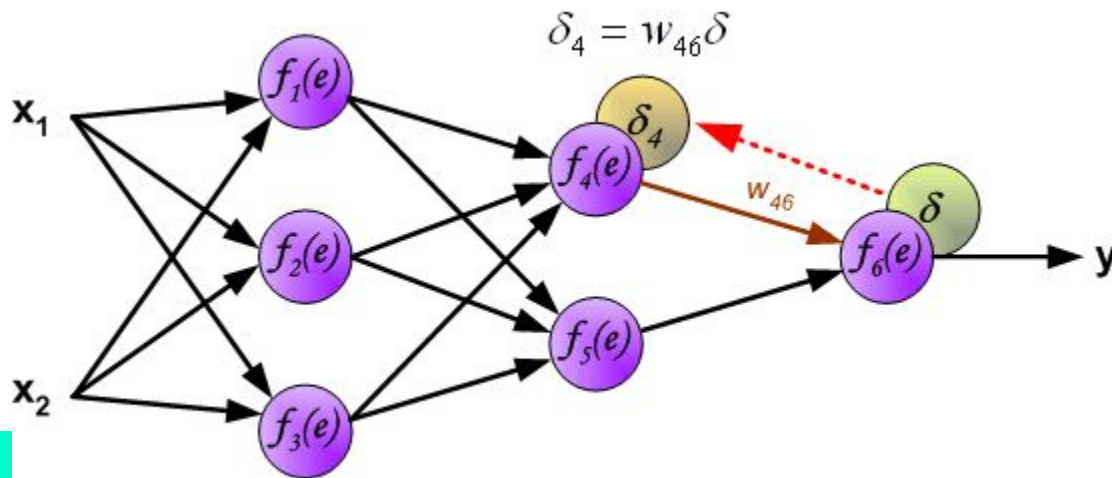


## Redes de retropropagação





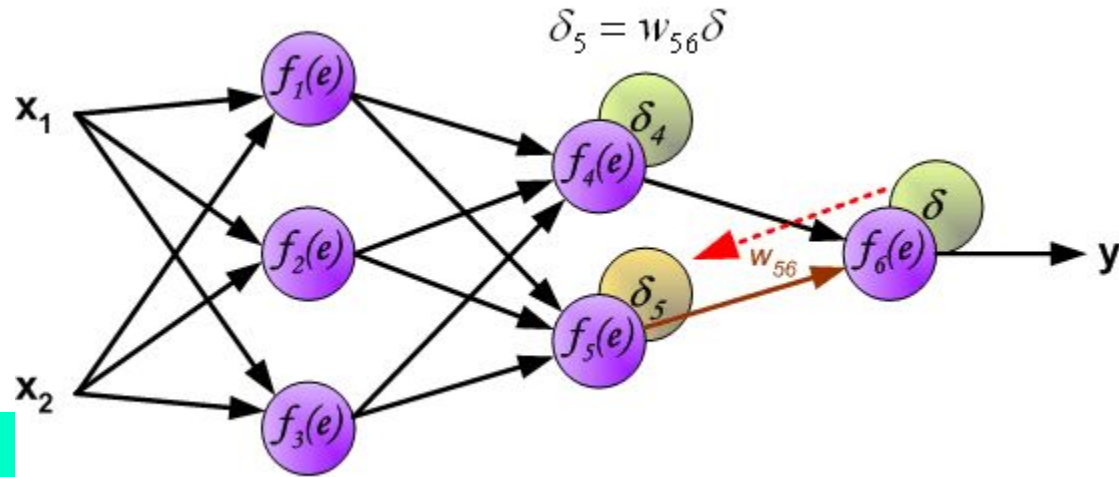
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992



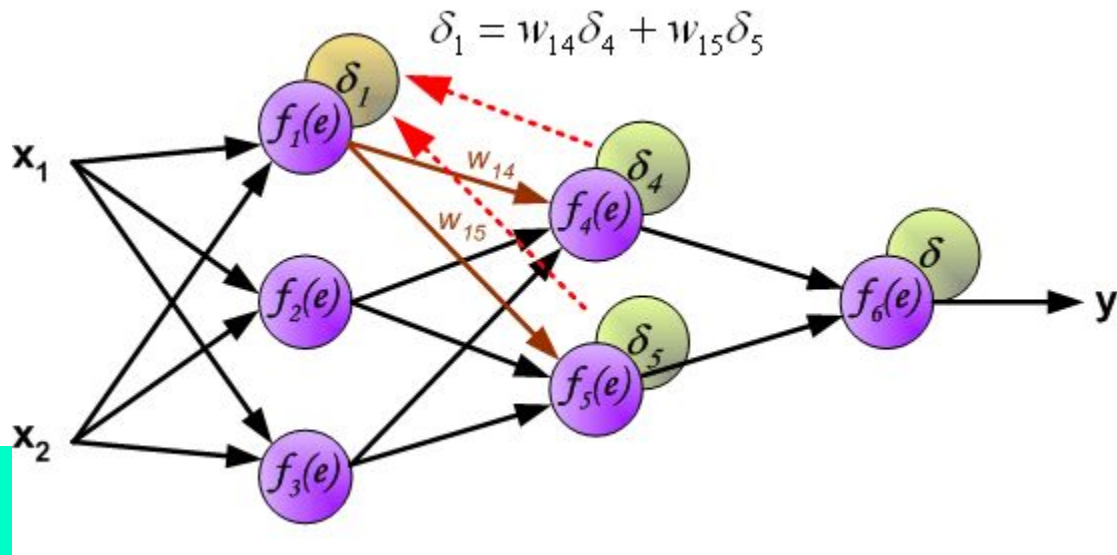
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992



## Redes de retropropagação

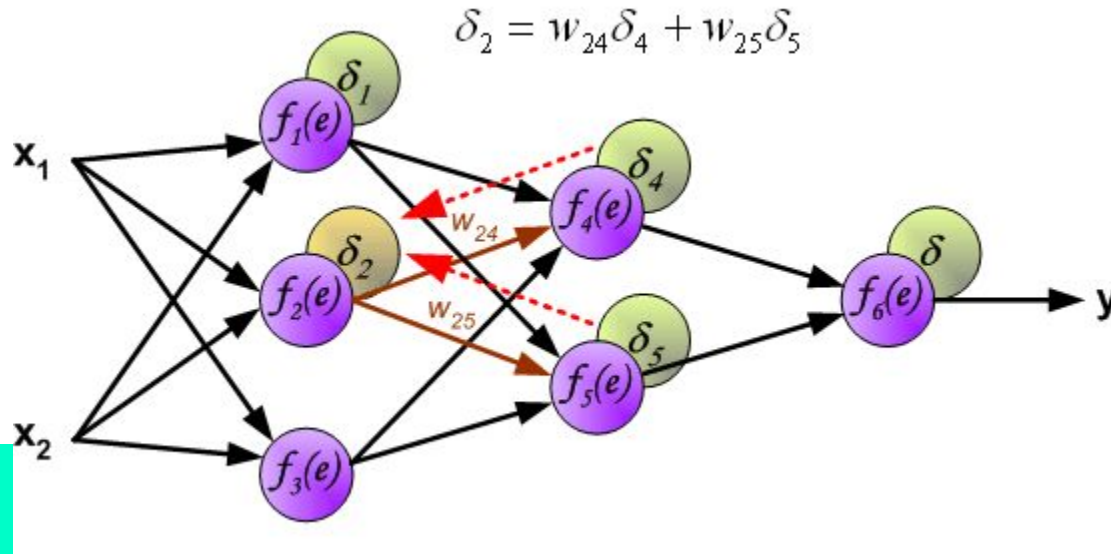


Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992





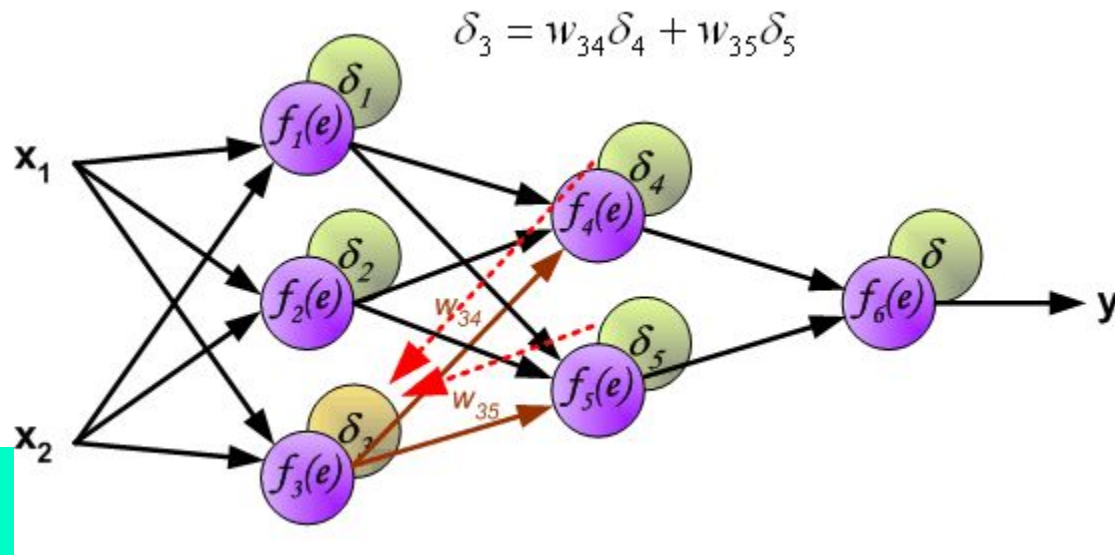
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992



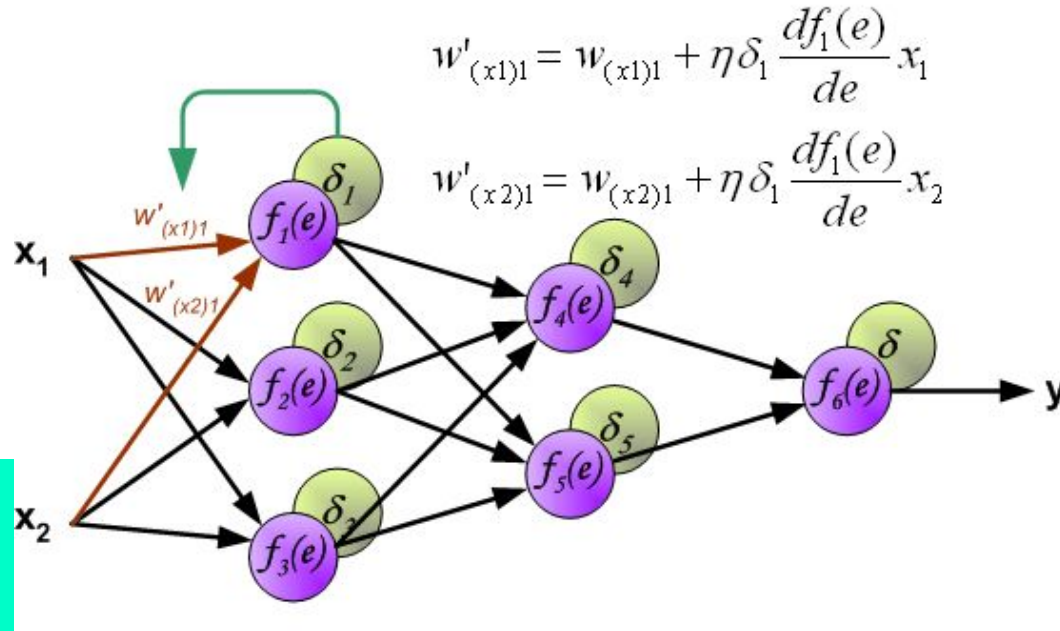
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992



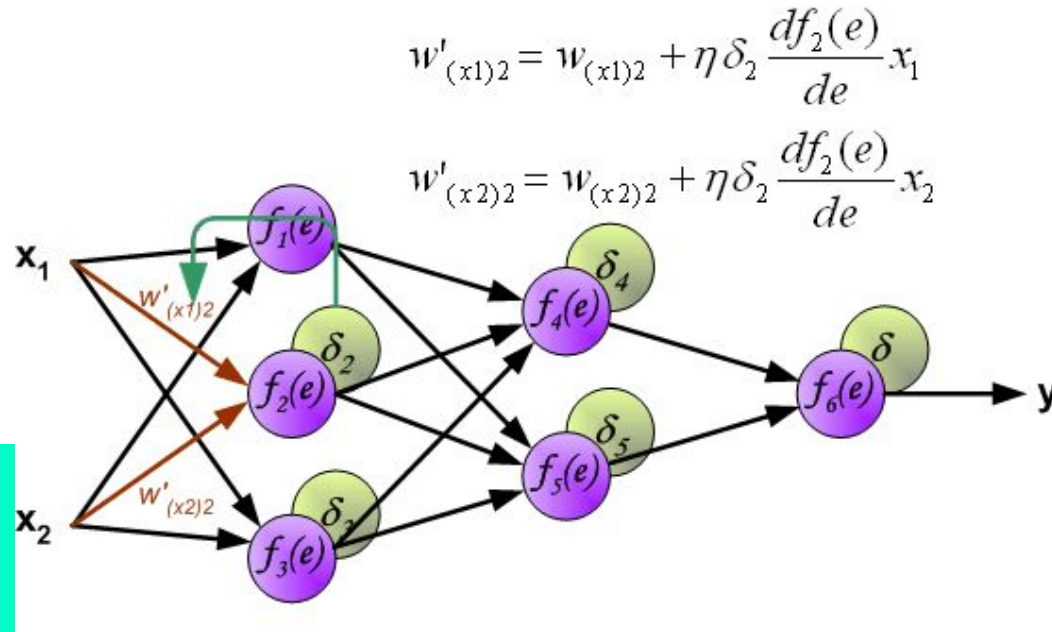
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992



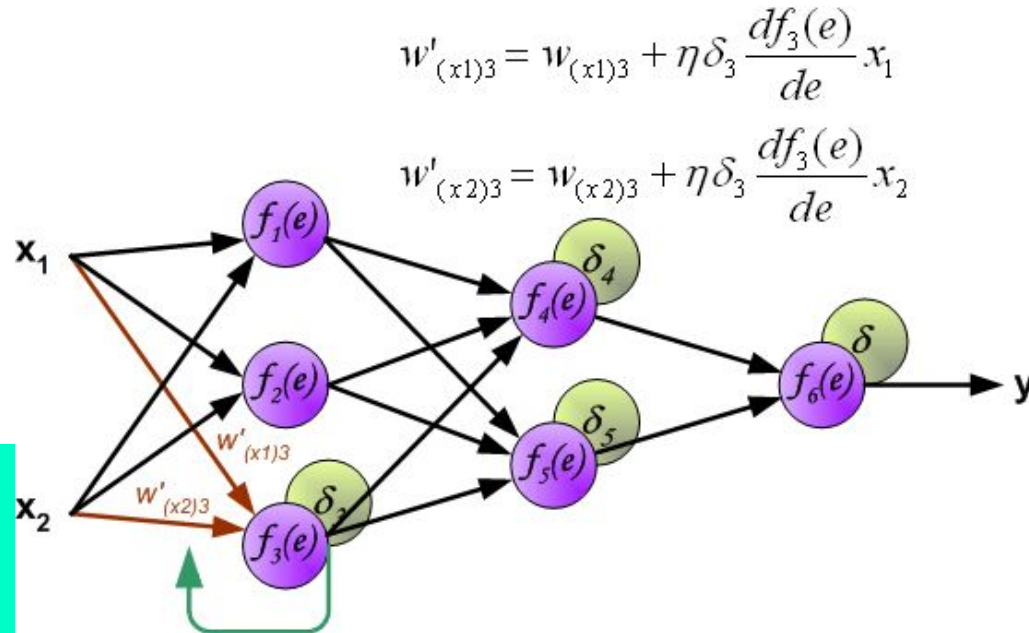
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992



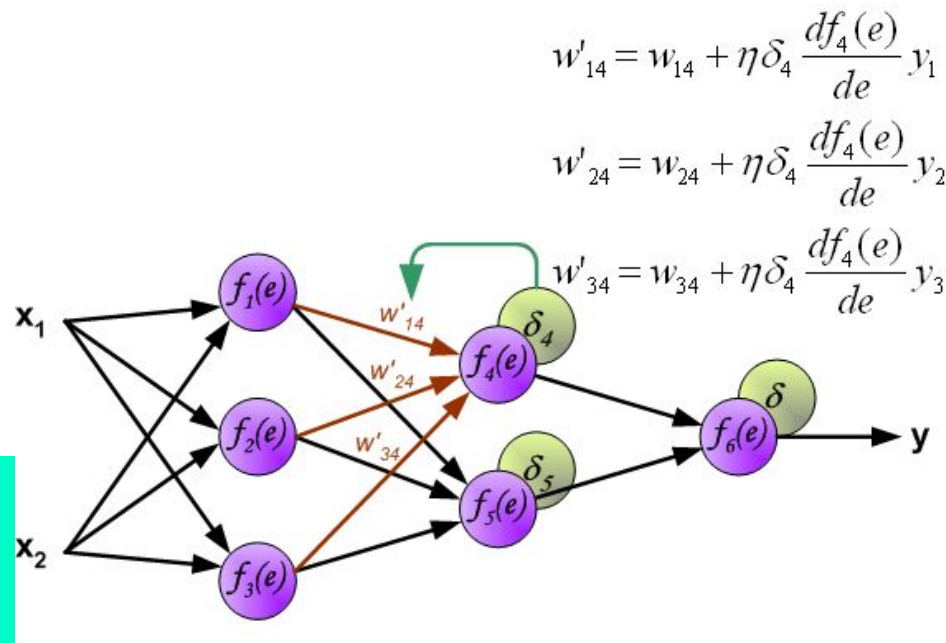
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992



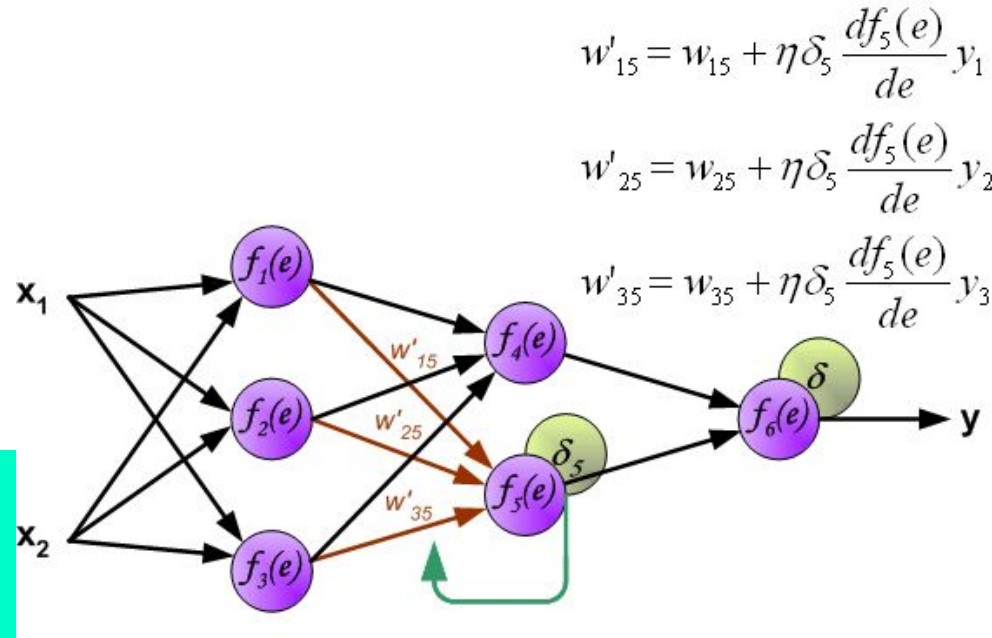
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992



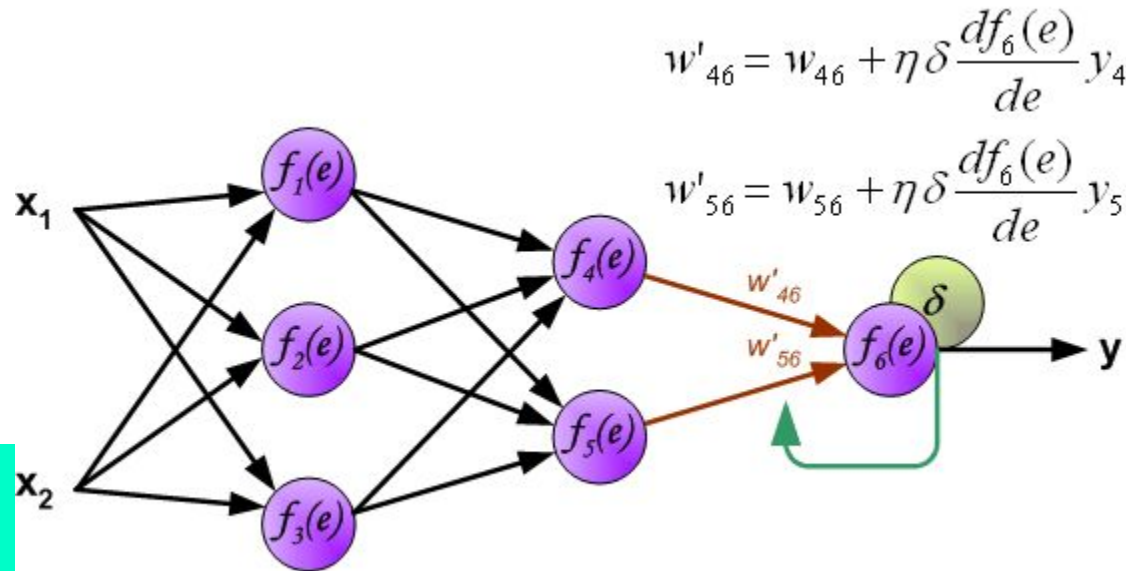
## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992



## Redes de retropropagação



Fonte: Ryszard Tadeusiewicz "Sieci neuronowe", Kraków 1992





## Redes de retropropagação

*Teorema: Qualquer função limitada pode ser aproximada, com uma precisão arbitrária, por uma rede neural com um número finito de neurônios oculto.*



## Redes de retropropagação

### Número de camadas:

- Redes que possuem apenas a camada de entrada e a de saída são úteis apenas para saídas que podem ser separadas de forma linear. Foi o primeiro perceptron.
- Redes com apenas uma camada oculta podem calcular uma função arbitrária qualquer (classificação simples).
- Usa-se pelo menos duas camadas ocultas quando os valores de saída variam de forma contínua.

| Estrutura | Regiões de Decisão                             | Problema Ou Exclusivo | Definição de Classes | Formatos mais Gerais |
|-----------|--|-----------------------|----------------------|----------------------|
|           | Meio Plano Limitado por Hiperplano             |                       |                      |                      |
|           | Regiões Convexas Abertas ou Fechadas           |                       |                      |                      |
|           | Arbitrárias: Complexidade Depende do Nº de nós |                       |                      |                      |



## Redes de retropropagação

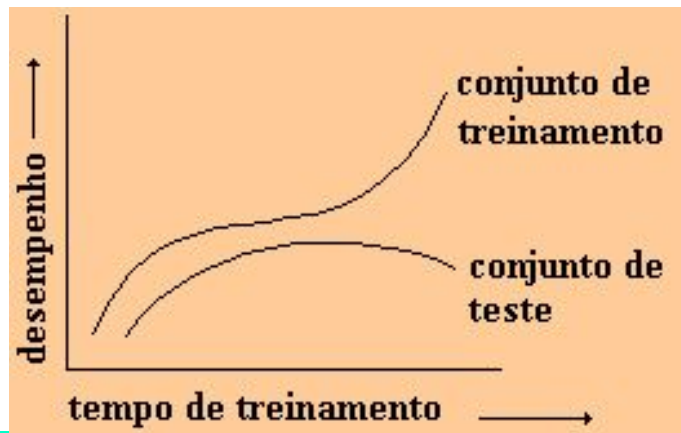
Número de neurônios nas camadas:

- **Hecht-Nielsen / Kolmogorov**: uma rede com 3 camadas pode modelar funções matemáticas contínuas desde que a camada oculta contenha  $2xI+1$  neurônios.
- **Kudricky**: numa rede com 2 camadas ocultas, obtém-se um desempenho ótimo quando há uma taxa de 3:1 entre o número de neurônios da 1ª e 2ª camadas ocultas.
- **Lippmann**: em redes com 2 camadas ocultas, a segunda camada deve ter  $2xO$  neurônios. Se a rede possuir apenas uma camada oculta, ela deverá ter  $Ox(I+1)$  neurônios.
- $H_{\max} = c/[10x(I+O)]$   $H = (OxI)^{1/2}$



## Redes de retropropagação

### Treinamento





## Redes de retropropagação

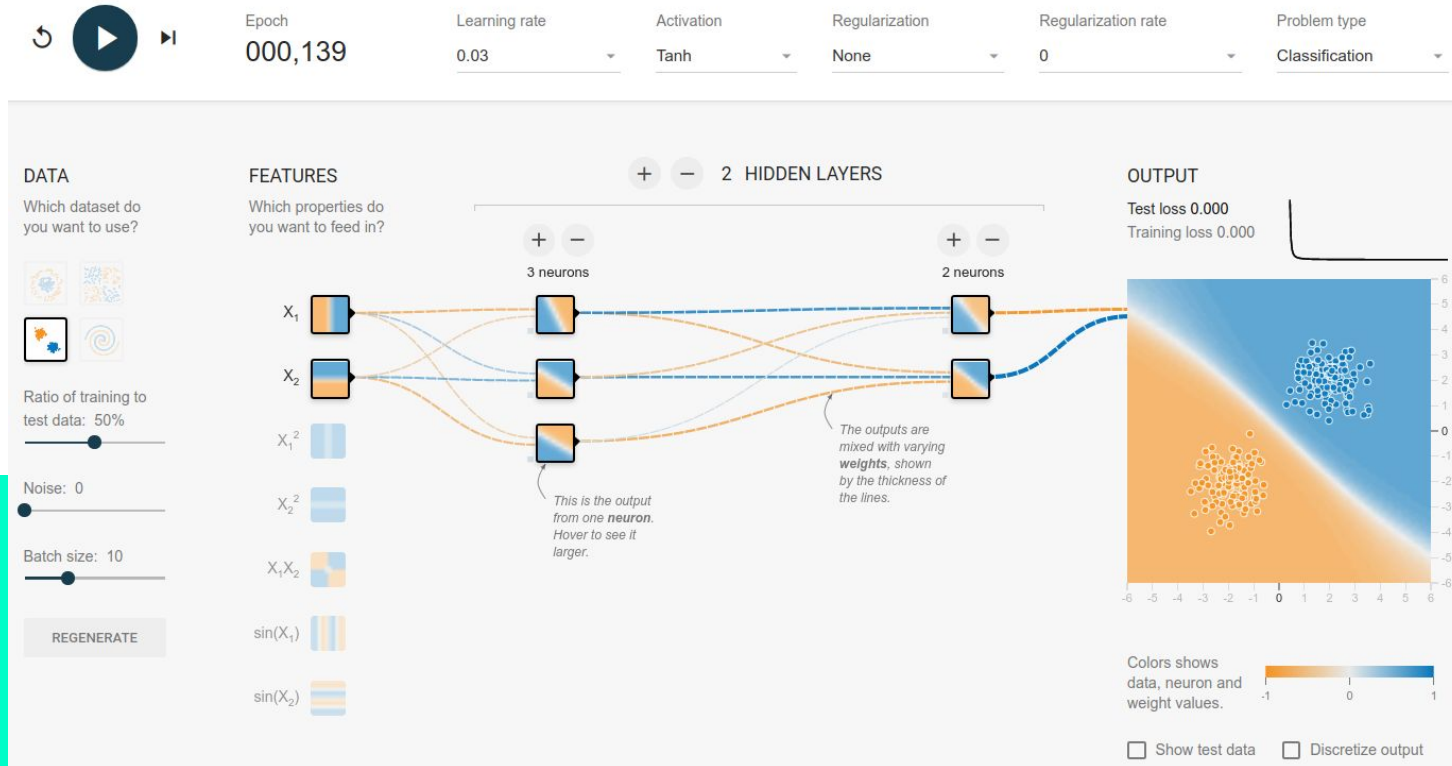
### Treinamento

- A explicação para o grafo é que a partir de certo ponto a rede “decorou” os exemplos e não generalizou mais. Desta forma ela passa a ter um desempenho pior nos casos de teste.
- Uma solução para isso é interromper o treinamento quando o platô for atingido.
- Outra solução é diminuir o número de elementos da camada intermediária, impedindo que ela utilize o grande número de pesos à sua disposição para memorizar os exemplos. (**Dropout**)
- Pode-se usar normalização (L1, L2)



## Redes de retropropagação

<https://playground.tensorflow.org/>

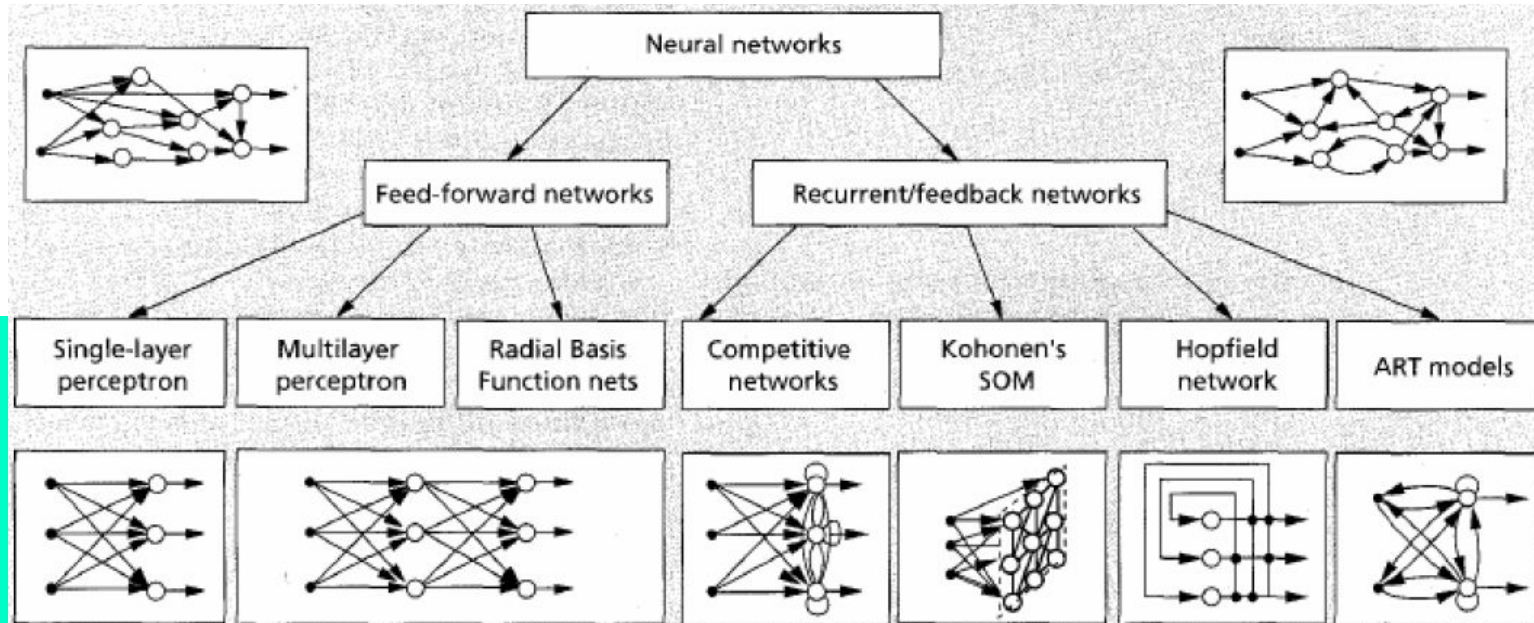




## Redes de retropropagação

### Topologia

### Classificação: Kain e Mao (1996)



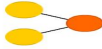


# A mostly complete chart of Neural Networks

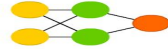
©2016 Fjodor van Veen - asimovinstitute.org

- Backfed Input Cell
- Input Cell
- Noisy Input Cell
- Hidden Cell
- Probabilistic Hidden Cell
- Spiking Hidden Cell
- Output Cell
- Match Input Output Cell
- Recurrent Cell
- Memory Cell
- Different Memory Cell
- Kernel
- Convolution or Pool

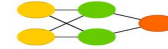
Perceptron (P)



Feed Forward (FF)



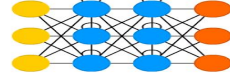
Radial Basis Network (RBF)



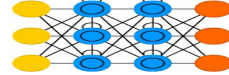
Deep Feed Forward (DFF)



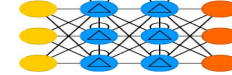
Recurrent Neural Network (RNN)



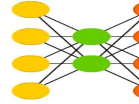
Long / Short Term Memory (LSTM)



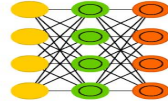
Gated Recurrent Unit (GRU)



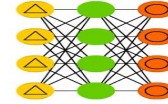
Auto Encoder (AE)



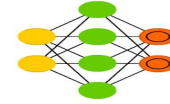
Variational AE (VAE)



Denoising AE (DAE)



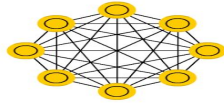
Sparse AE (SAE)



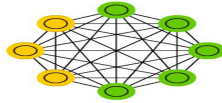
Markov Chain (MC)



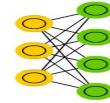
Hopfield Network (HN)



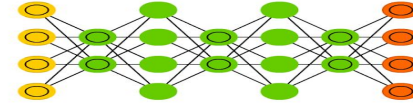
Boltzmann Machine (BM)



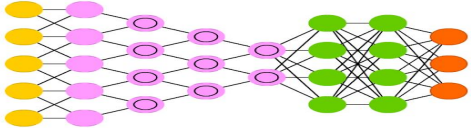
Restricted BM (RBM)



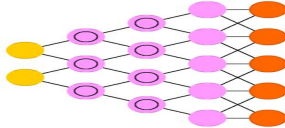
Deep Belief Network (DBN)



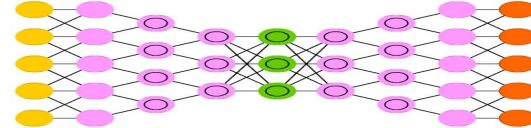
Deep Convolutional Network (DCN)



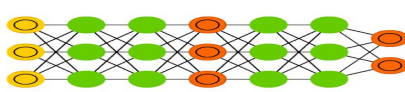
Deconvolutional Network (DN)



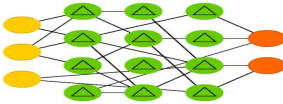
Deep Convolutional Inverse Graphics Network (DCIGN)



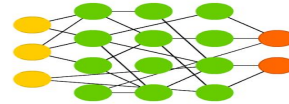
Generative Adversarial Network (GAN)



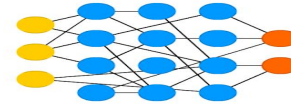
Liquid State Machine (LSM)



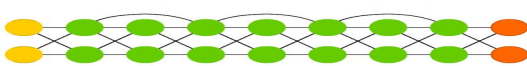
Extreme Learning Machine (ELM)



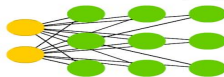
Echo State Network (ESN)



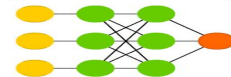
Deep Residual Network (DRN)



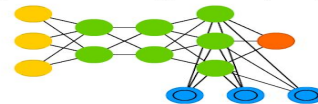
Kohonen Network (KN)



Support Vector Machine (SVM)



Neural Turing Machine (NTM)







## Links interessantes para redes neurais

- <https://sefiks.com/2017/01/21/the-math-behind-backpropagation/>
  - <https://stats.stackexchange.com/questions/154879/a-list-of-cost-functions-used-in-neural-networks-alongside-applications>
  - <http://www.deeplearningbook.org/>
  - <http://deeplearningbook.com.br/>
- 
- [http://parrt.cs.usfca.edu/doc/matrix-calculus/index.html?utm\\_campaign=Artificial%2BIntelligence%2BWeekly&utm\\_medium=web&utm\\_source=Artificial\\_Intelligence\\_Weekly\\_75](http://parrt.cs.usfca.edu/doc/matrix-calculus/index.html?utm_campaign=Artificial%2BIntelligence%2BWeekly&utm_medium=web&utm_source=Artificial_Intelligence_Weekly_75)
  - <https://alonalj.github.io/2016/12/10/What-is-Backpropagation/>



## Livros

Russel S. & Norvig P. *Inteligência Artificial, Campus*; ISBN: 8535211772, 2010. Terceira Ed.

Haykin, Simon. *Neural networks and learning machines*. 3 ed. Pearson Education Ed. 2009.