REDES NEURAIS

Revisão

TÓPICOS

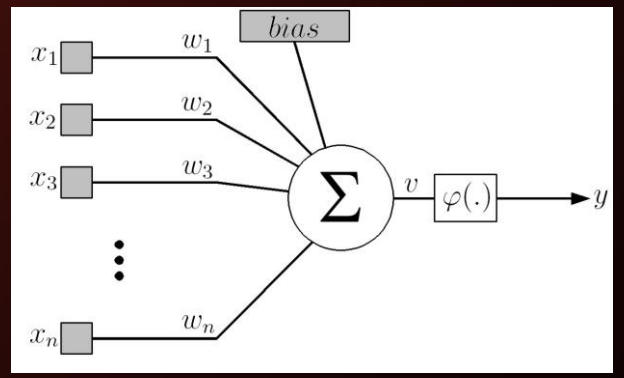
1. O que são as redes neurais artificiais
2. Principais fatos históricos
3. Neurônios matemáticos
4. Aprendizagem e principais arquiteturas
5. Tratamento dos dados para treinamento
6. Perceptron e Adaline

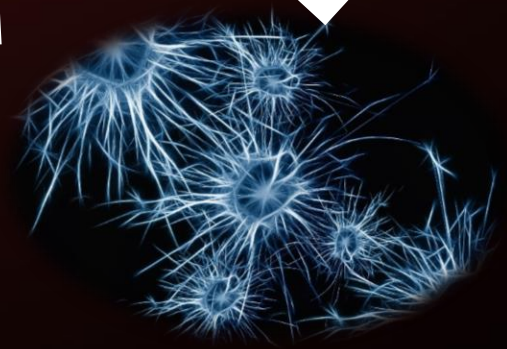
TÓPICOS

7. Rede MLP
8. Rede RBF
9. Rede SOM
10. Energia: Hopfield, Boltzmann e RBM
11. Redes Recorrentes: RNN, GRU, LSTM
12. Questões e Dúvidas

O que são as redes neurais artificiais?

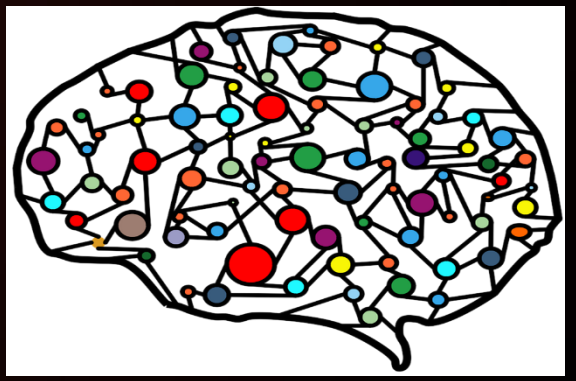
O QUE SÃO AS REDES NEURAIS





FONTE: https://www.deeplearningbook.com.br/uma-breve-historia-das-redes-neurais-artificiais/

O QUE SÃO AS REDES NEURAIS

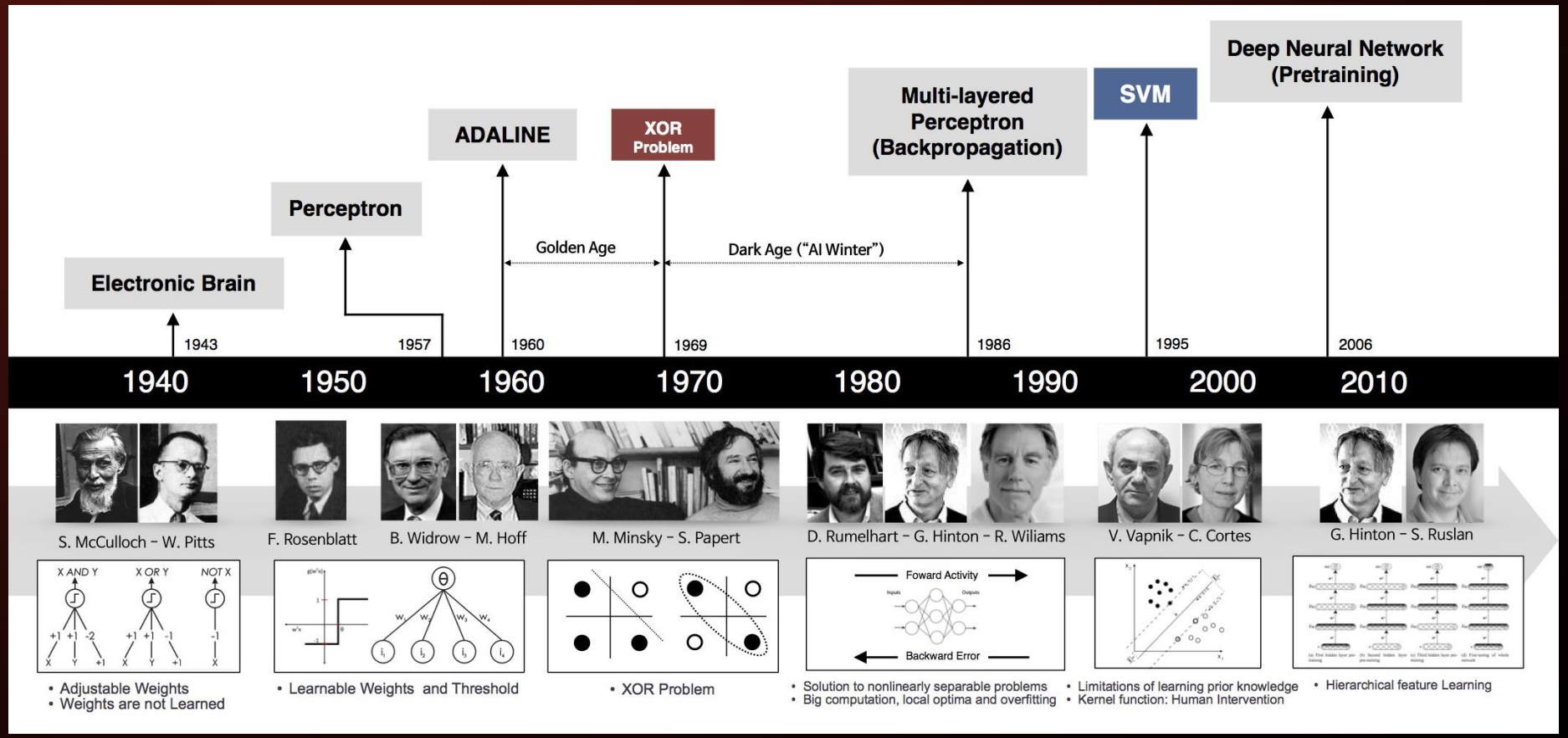


SÃO MODELOS (REDES) COMPUTACIONAIS INSPIRADOS NA ESTRUTURA E NO FUNCIONAMENTO DO SISTEMA NERVOSO

- O conhecimento é adquirido a partir do processo de aprendizagem
- O conhecimento é armazenado nos pesos da rede
- O comportamento inteligente emerge da rede (*bottom-up*)
- Existem diversos tipos de RNs

Como a área evoluiu?

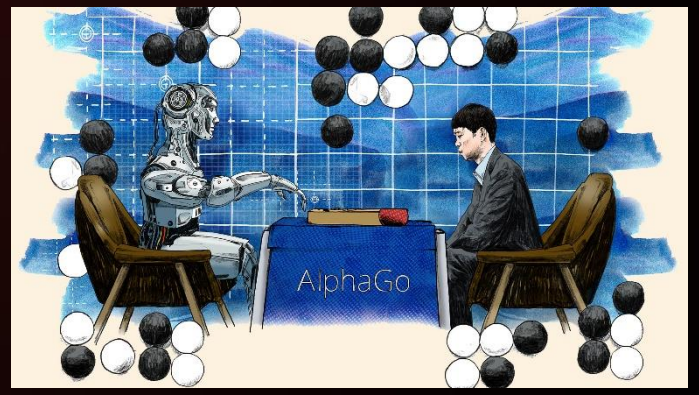
HISTÓRICO



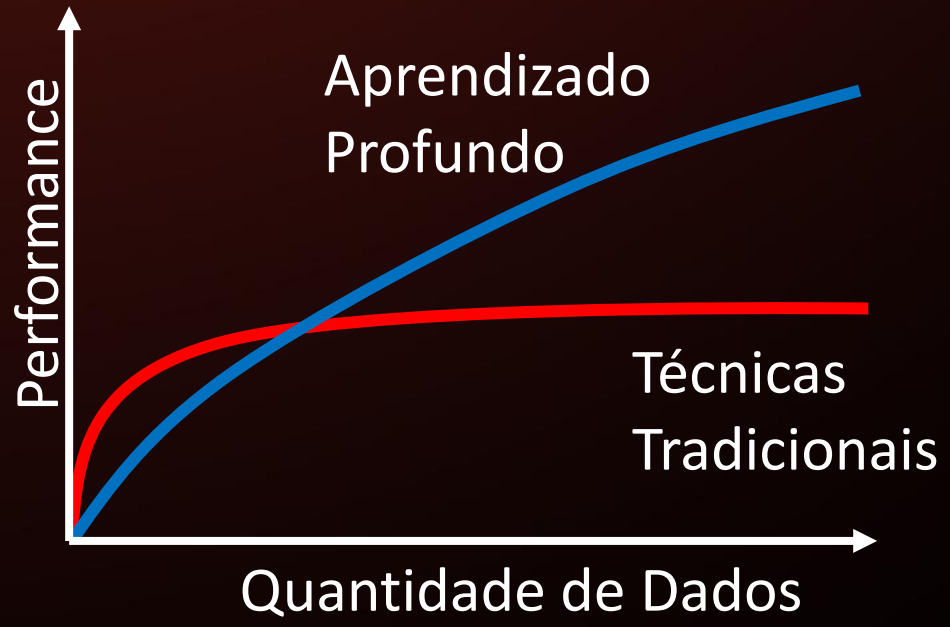
FONTE: https://www.deeplearningbook.com.br/uma-breve-historia-das-redes-neurais-artificiais/

HISTÓRICO: DEEP LEARNING

- Grandes avanços: dados + GPUs
  - AlexNet em 2012
  - Modelos Generativos: VAE, GAN
  - AlphaGo em 2016
  - Aplicações nas mais diversas áreas



Fonte: https://ai.plainenglish.io/



O Neurônio Matemático: Peça fundamental das redes neurais artificiais

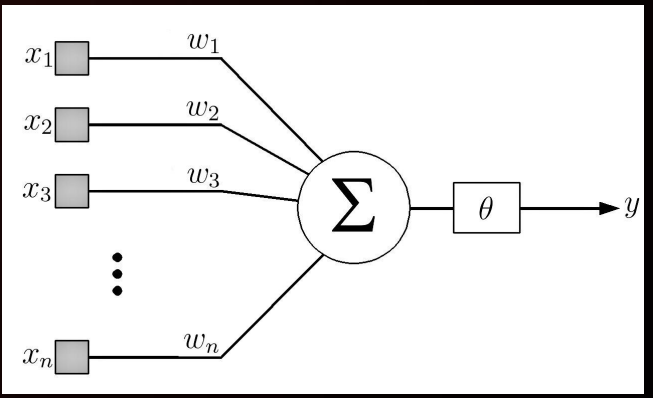
O NEURÔNIO MCP

Década de 40, MCP, Hodgkin-Huxley

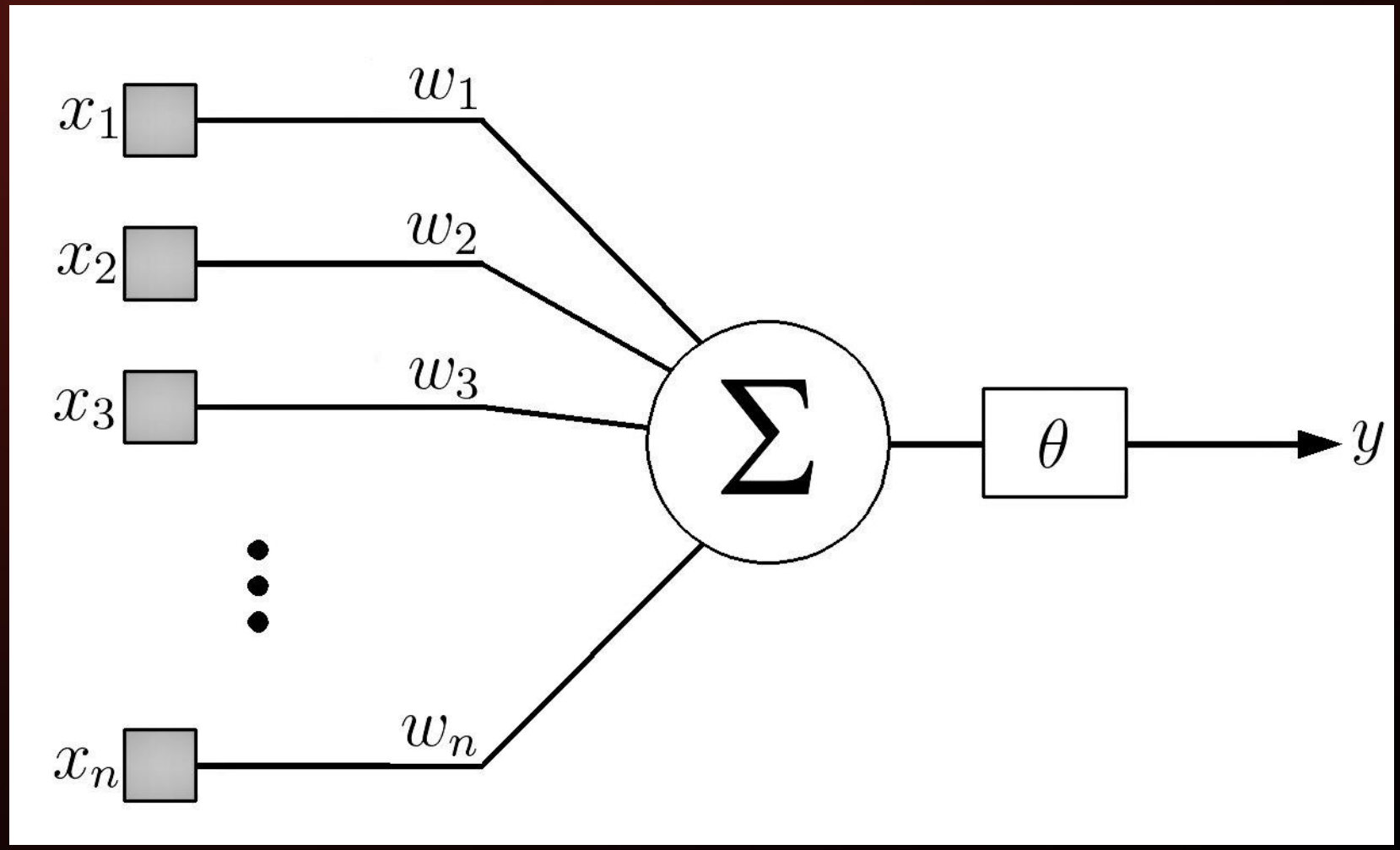
O Neurônio MCP é composto por:

- As Sinapses (dendritos)
- O corpo celular (somatório ponderado)
- E uma saída (axônio) que representa quando o neurônio está ativo ou não (função de ativação)

  - Saída binária
  - Ou está disparando potenciais de ação ou está em repouso



O NEURÔNIO MCP



NEURÔNIO ESTOCÁSTICO

Resposta probabilística

$$x = \begin{cases} +1 & \text{com probabilidade } P(v) \\ -1 & \text{com probabilidade } 1 - P(v) \end{cases}$$

$$P(v) = \frac{1}{1 + exp(-v/T)}$$

Como as redes aprendem?

O QUE É APRENDIZAGEM?

$$\Delta w = f(?)$$

O processo de aprendizado implica a seguinte sequência de eventos:

1. **A rede neural é estimulada pelo ambiente.**
2. **A rede neural sofre modificações nos seus parâmetros livres como resultado deste estímulo.**
3. **A rede neural responde de uma maneira nova ao ambiente, devido às modificações ocorridas na sua estrutura interna.**

REGRAS DE APRENDIZAGEM

- Aprendizado por correção do erro
- Aprendizado baseado em memória
- Aprendizado Hebbiano
- Aprendizado competitivo
- Aprendizado de Boltzmann

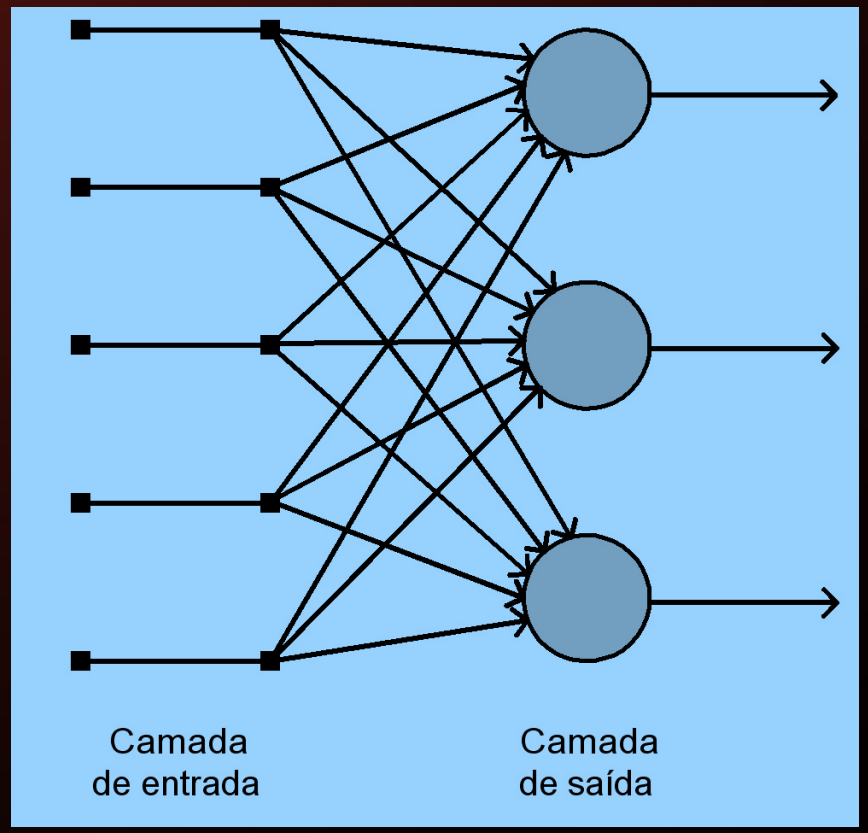
PARADIGMAS DE APRENDIZAGEM

- Aprendizado Supervisionado
- Aprendizado Não-Supervisionado
- Aprendizado por Reforço
- Outras formas:
  - Aprendizado Autossupervisionado
  - Aprendizado Semi-supervisionado
  - Aprendizado Ativo

Principais Arquiteturas de Redes Neurais

REDES COM CAMADA ÚNICA

- **Redes alimentadas adiante com camada única**
- **Fluxo único:**
  - **entrada → saída**

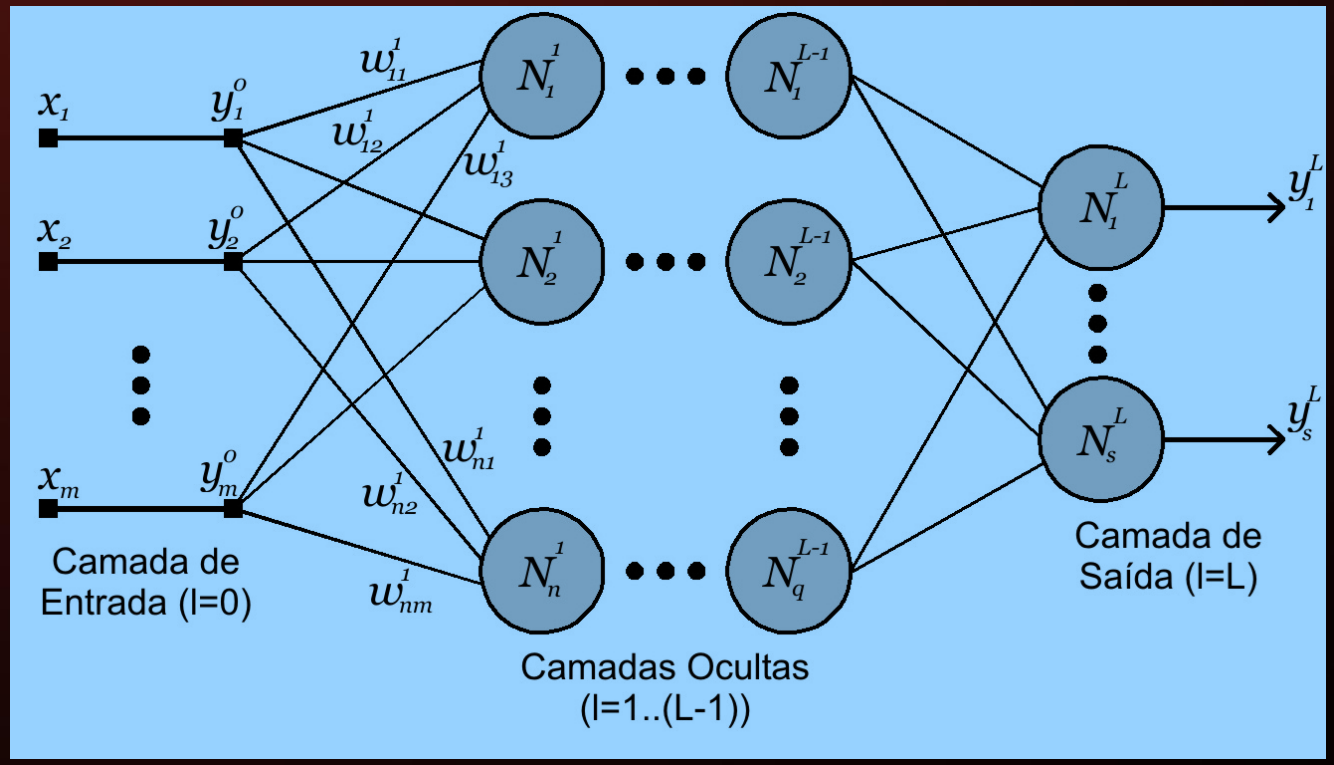


Fonte: Livro Simon Haykin (2001)

REDES DE CAMADA ÚNICA

- **Múltiplas camadas - Fluxo: entrada → saída**

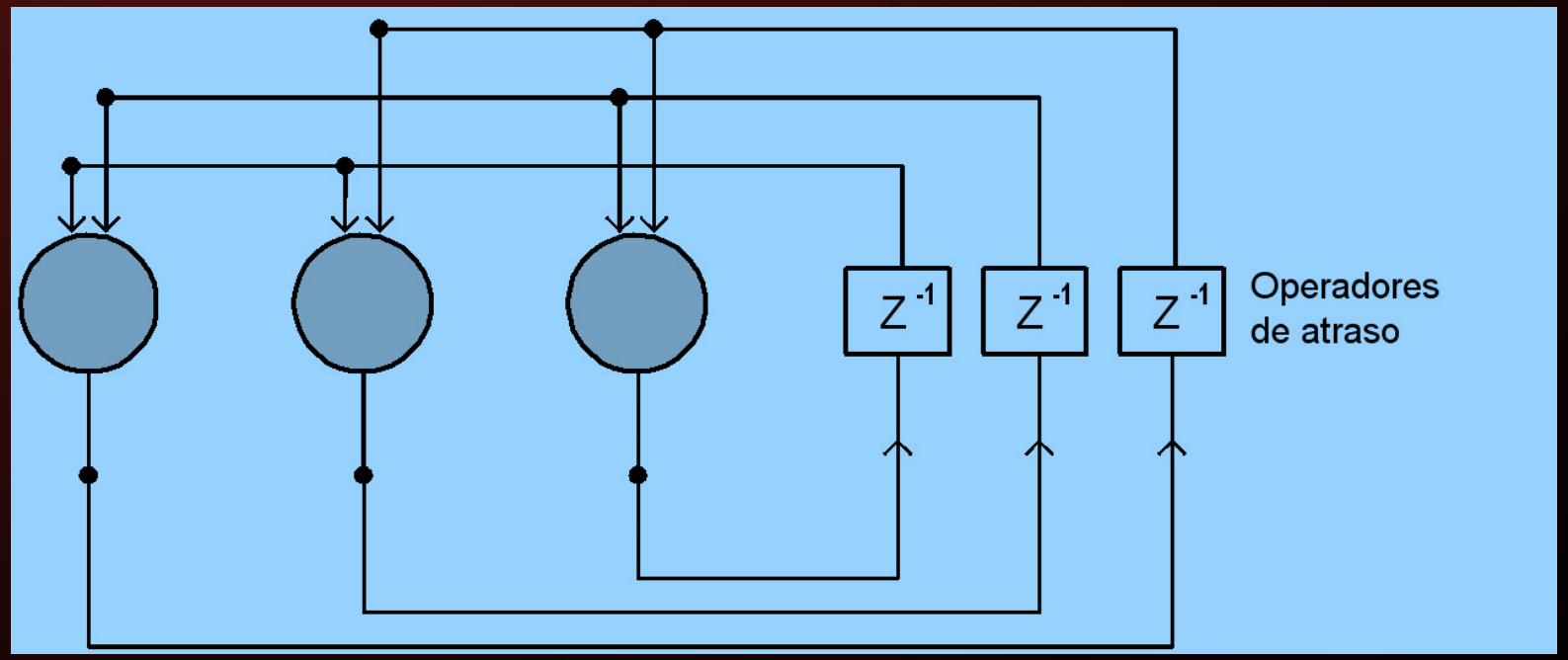
Fonte: Livro Simon Haykin (2001)



REDES RECORRENTES

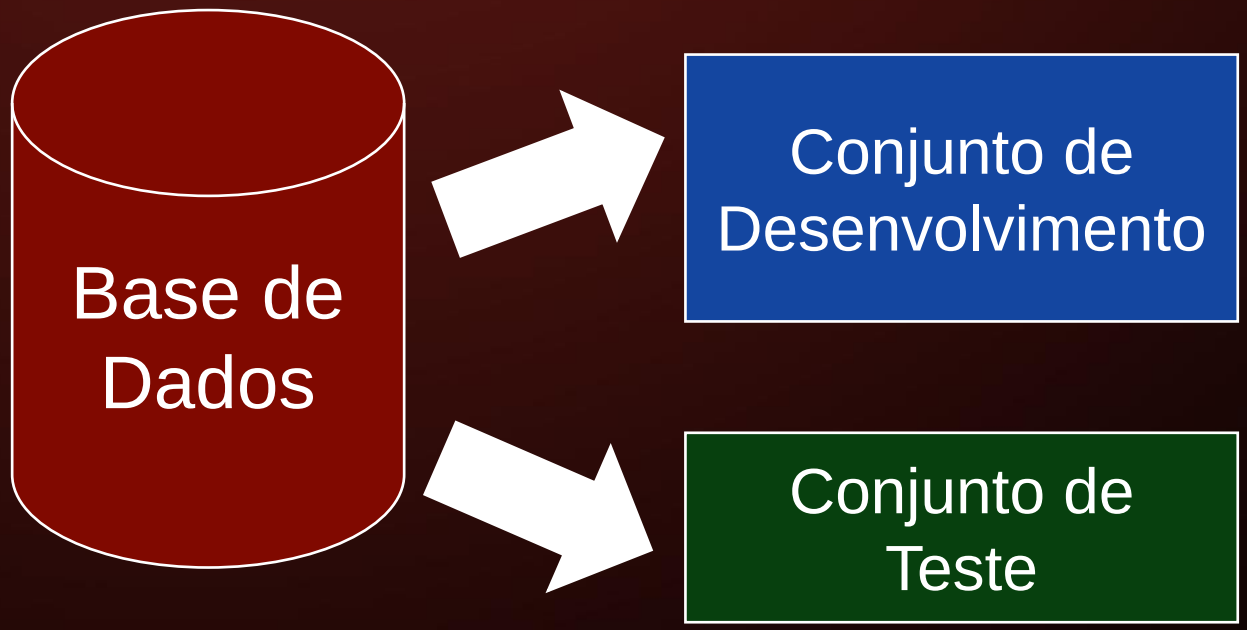
- **Possui laços de realimentação**

Fonte: Livro Simon Haykin (2001)



Tratamento e Divisão do conjunto de dados

DIVISÃO DO CONJUNTO DE DADOS



CONJUNTO DE DESENVOLVIMENTO

- **Usado para configurar os parâmetros e hiperparâmetros do modelo**

CONJUNTO DE TESTE

- **Utilizado para avaliar a resposta do modelo final já treinado**
- **Não pode ser usado durante o desenvolvimento**

DIVISÃO DO CONJUNTO DE DADOS

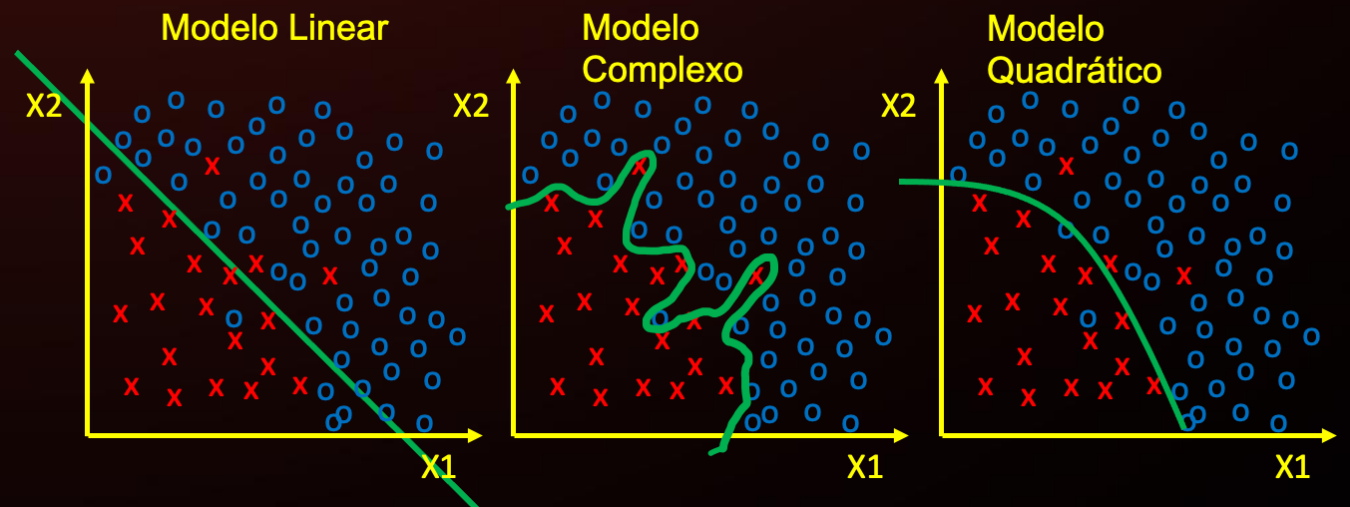
Conjunto de Desenvolvimento

Conjunto de Teste

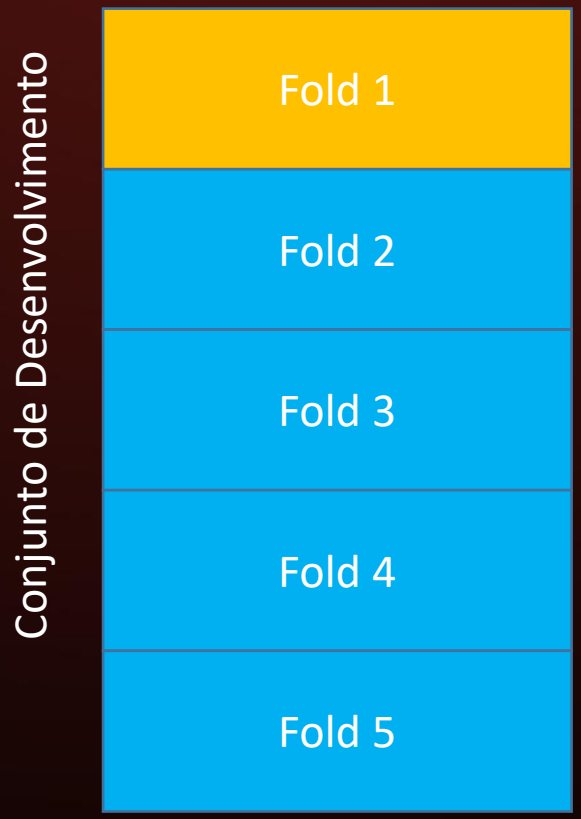
Conjunto de Treino

Conjunto de Validação

- **Treino → ajuste dos parâmetros**
- **Validação → ajuste dos hiperparâmetros**



VALIDAÇÃO CRUZADA



1. O modelo é treinado com 4 *folds* (azul) e validado com o *fold* extra (amarelo)
2. O processo é executado para todas as combinações
3. O erro de validação é o erro médio das execuções
4. k-fold cross-validation

PRÉ-PROCESSAMENTO DOS DADOS

- Dados reais podem conter problemas:

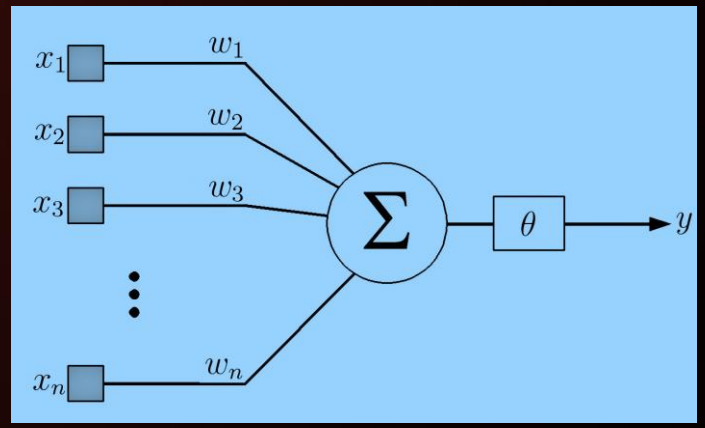
- O Pré-Processamento pode:

  - Melhorar a qualidade dos dados
    - Limpeza dos dados, imputação, seleção de atributos, etc.

  - Facilitar a aplicação de uma data técnica de aprendizado de máquina
    - Balanceamento, transformações, normalização, etc.

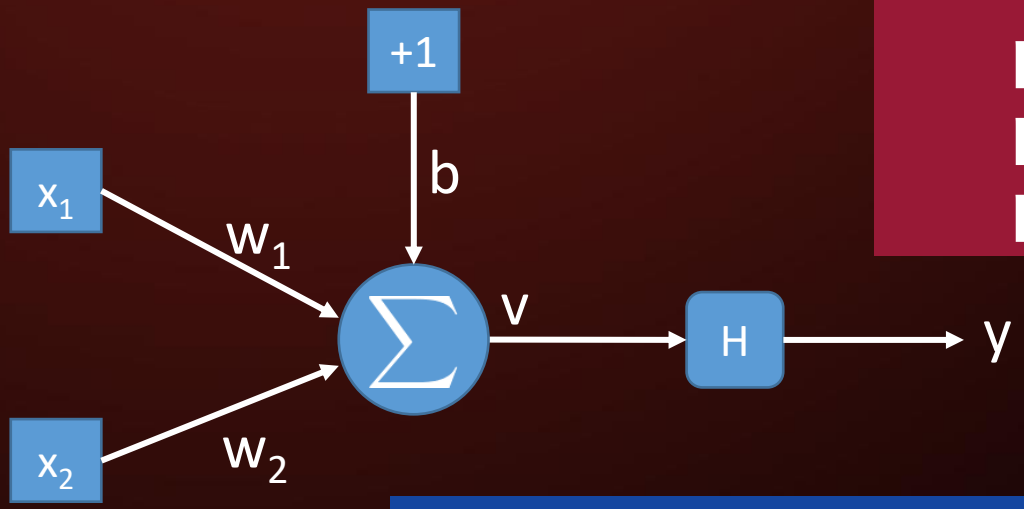
O Perceptron e o Adaline

O QUE FAZ O MCP?



1. Representa uma abstração do Neurônio Biológico
2. Pode ser configurado para implementar portas lógicas, i.e. AND, OR
3. Como configurá-lo (treiná-lo)?

MCP: PORTA AND



- **O que precisamos fazer para configurar a porta AND?**

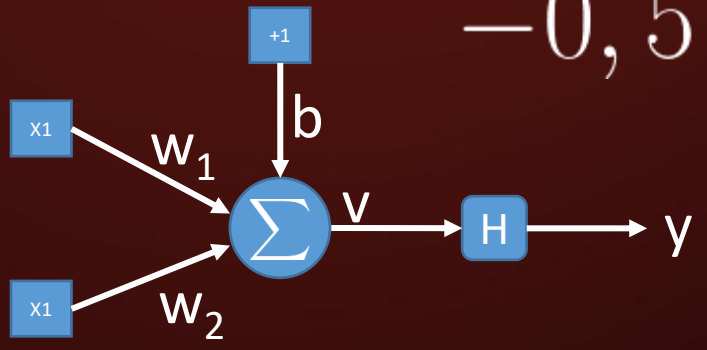
AND		
x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

- **Ajustar os valores de w_1, w_2 e b**

$$bx_0 + w_1x_1 + w_2x_2 \leq 0$$

- **b = -0,5**
- **w_1 = 0,3**
- **w_2 = 0,3**

MCP: PORTA AND

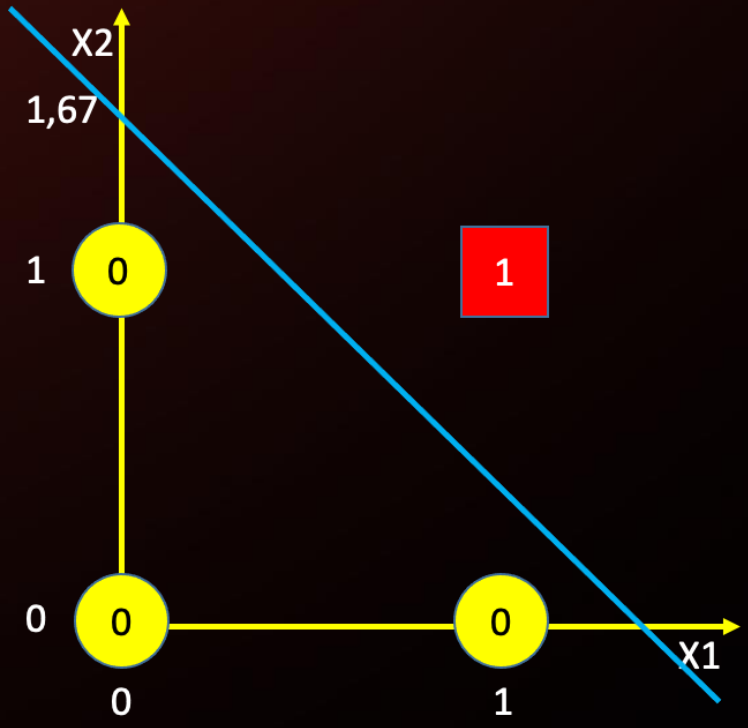


$$-0,5 + 0,3x_1 + 0,3x_2 = 0$$

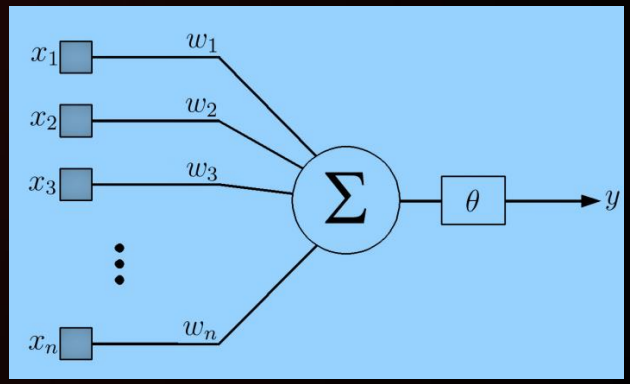
$$x_2 = -1x_1 + 1,67$$

AND		
x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

- O que os MCP representam?
- O que o bias representa?
- Qual é o problema do MCP?

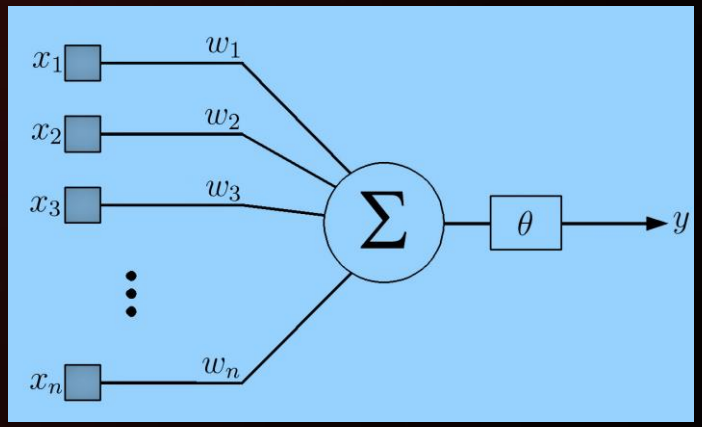


PERCEPTRON



- Proposto por Rosenblatt, em 1958
- Associa um algoritmo de aprendizagem ao neurônio MCP: ajuste automático dos pesos via correção de erros
- A rede possui apenas uma camada de neurônios binários ajustáveis
- Usado para classificação de padrões
- Converge com erro zero se as classes forem linearmente separáveis

PERCEPTRON



- **Regra de atualização dos Pesos:**
  - **Se o padrão é corretamente classificado, o peso não é alterado**
  - **Se o padrão for erroneamente classificado, o peso é atualizado por:**

$$w(n+1) = w(n) + \eta\,[d(n) - y(n)]\,x(n)$$

$$\Delta w = \eta e(n) x(n)$$

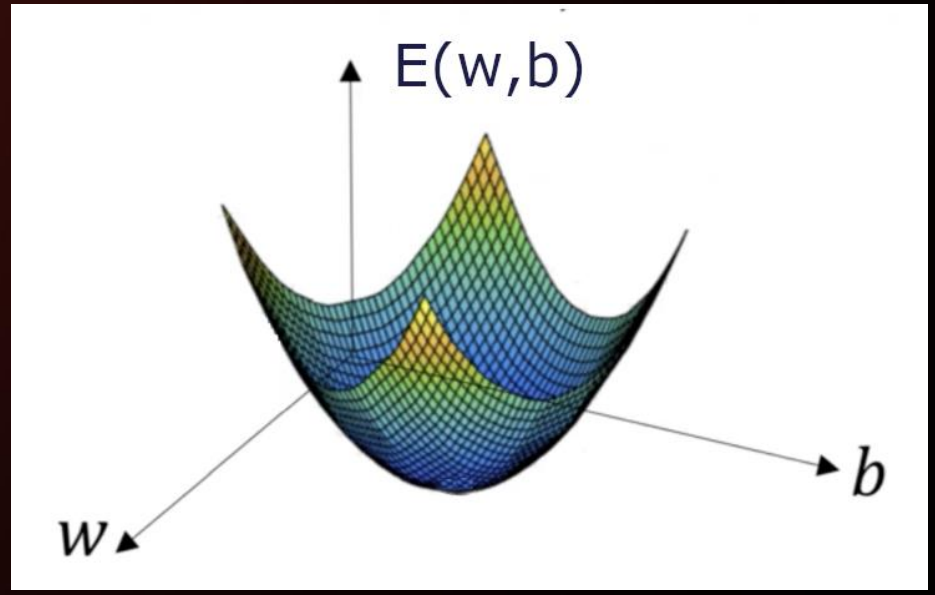
ADALINE

- O algoritmo de aprendizagem tem como objetivo minimizar o erro das saídas em relação aos valores desejados (conjunto de treinamento)

- A função de custo a ser minimizada é a soma dos erros quadráticos:

$$E(n) = \frac{1}{2} \sum (d_k(n) - y_k(n))^2$$

ADALINE



- **Processo de minimização do erro quadrático pelo método do Gradiente Descendente**

$$\Delta w_{ki} = -\eta \frac{\partial E}{\partial w_{ki}}$$

- **Cada peso sináptico i do neurônio k é atualizado proporcionalmente ao negativo da derivada parcial do erro em relação ao peso**

$$\Delta w_{ki} = \eta (d_k - y_k) x_i$$

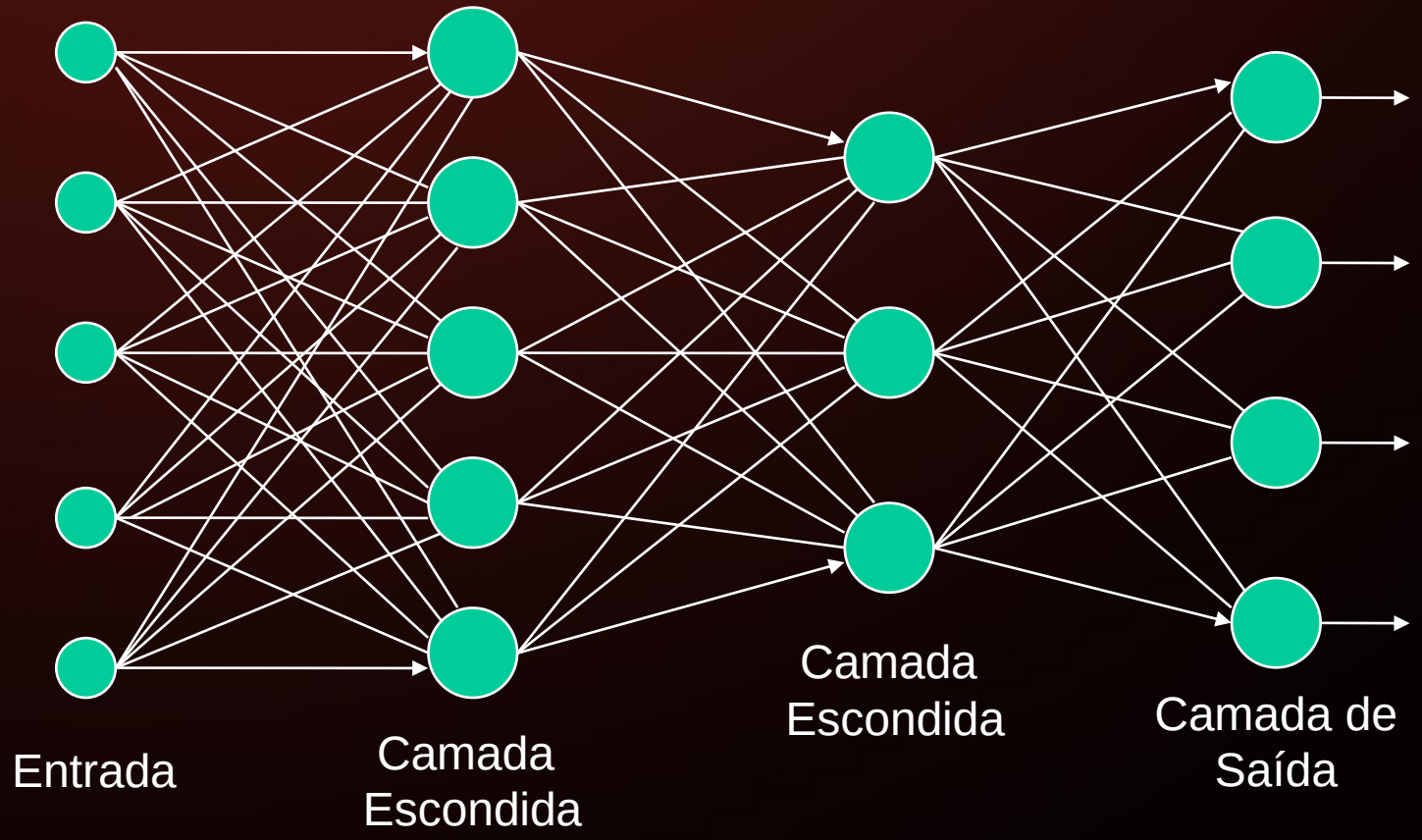
Adaptado de: https://builtin.com/data-science/gradient-descent

A rede Multilayer Perceptron (MLP)

REDE MLP

REDE COM 3 CAMADAS

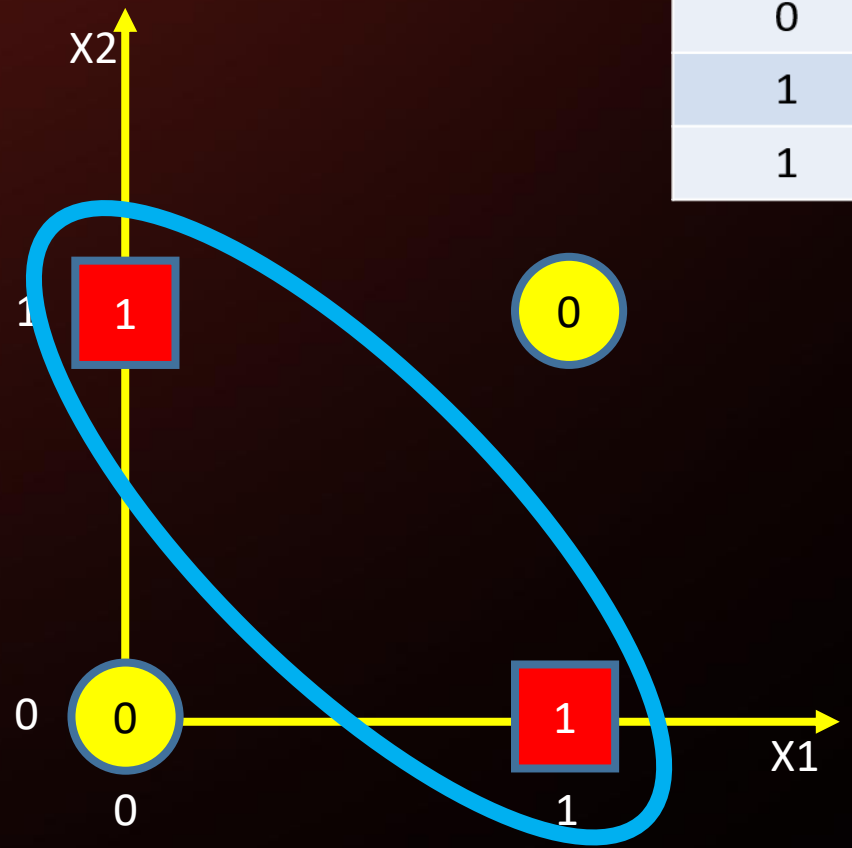
- Duas ocultas (escondidas)
- Uma camada de saída



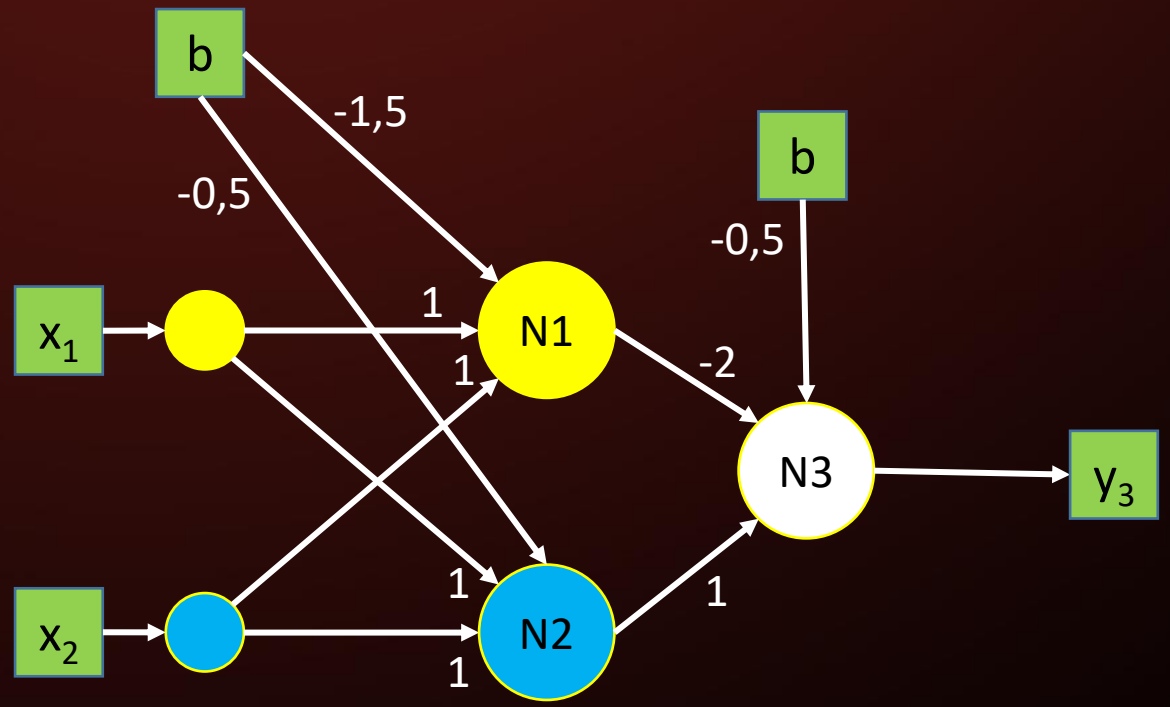
RETOMANDO O PROBLEMA XOR

- Como resolvê-lo com redes de múltiplas camadas?
- Qual é a configuração necessária?

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	0

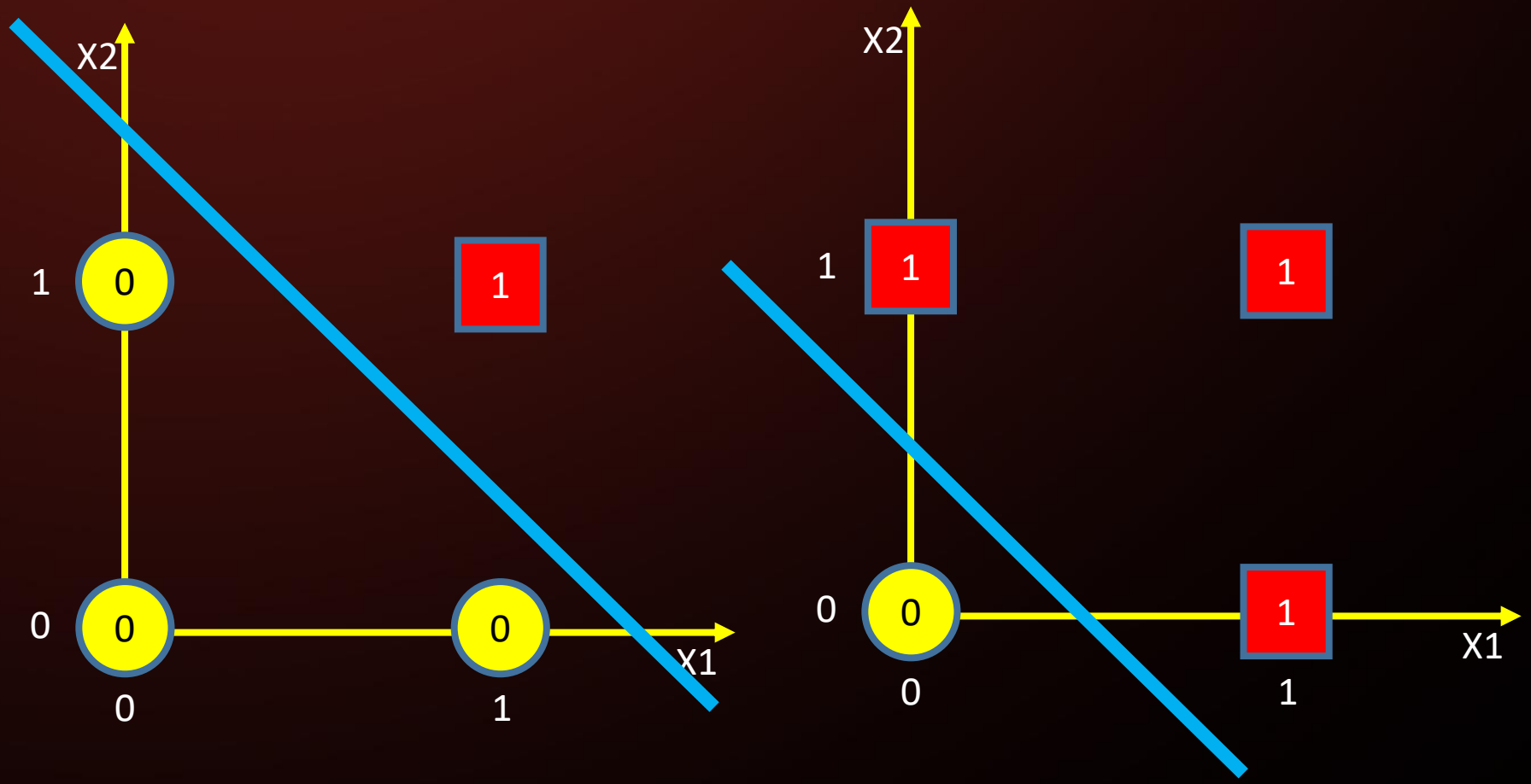


PROBLEMA XOR

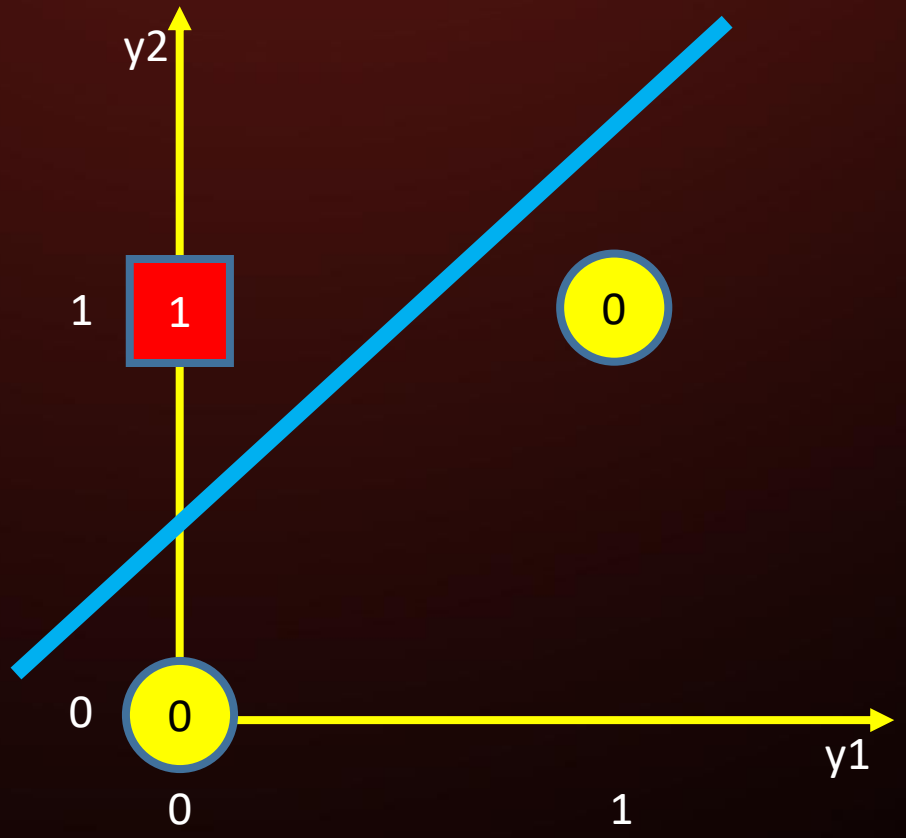


x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	0

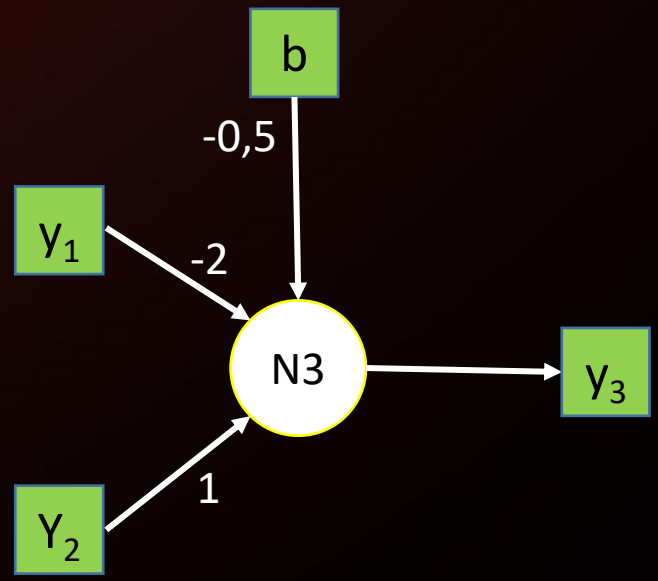
PROBLEMA XOR



PROBLEMA XOR

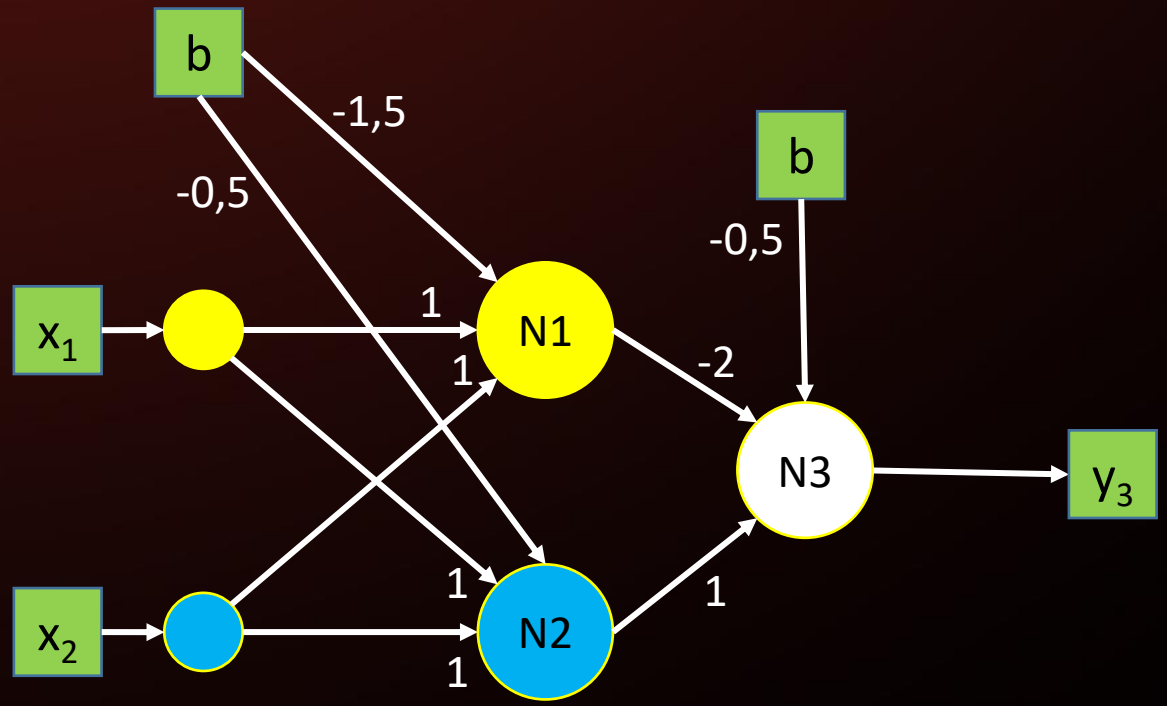


y_1	y_2	y_3
0	0	0
0	1	1
0	1	1
1	1	0



O QUE A REDE MLP FAZ?

Resolve problemas não-linearmente separáveis a partir da transformação do problema original em um problema linearmente separável (camada a camada).



RETROPROPAGAÇÃO

Processo de minimização do erro pelo método do gradiente descendente:

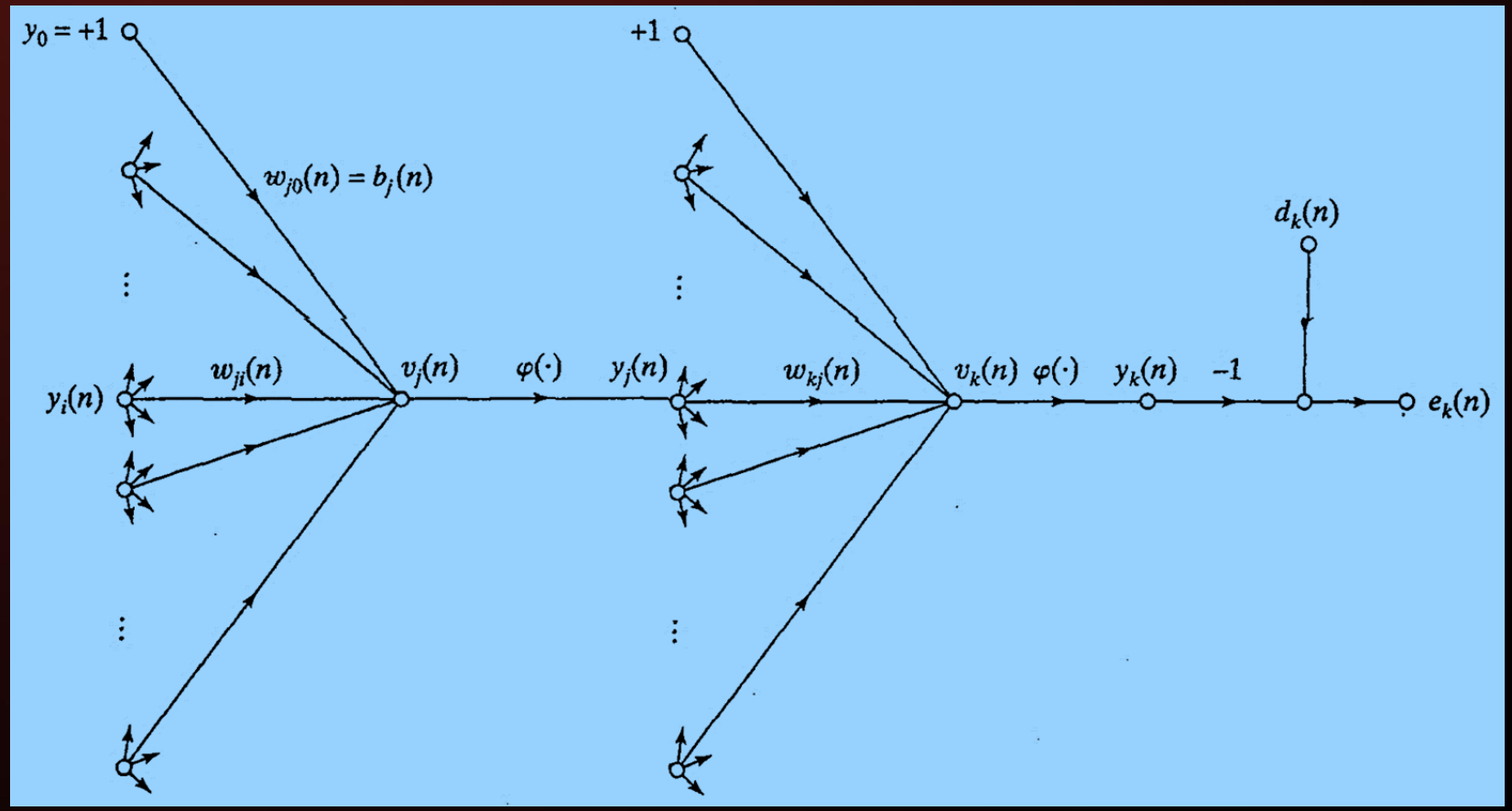
$$E(n) = \frac{1}{2}\sum (d_k(n) - y_k(n))^2$$

$$\Delta w_{ki} = -\eta \frac{\partial E}{\partial w_{ki}}$$

RETROPROPAGAÇÃO

$$\Delta w_{ki} = -\eta \frac{\partial E}{\partial w_{ki}}$$

GRAFO DE FLUXO:



RETROPROPAGAÇÃO

$$\Delta w_{kj} = -\eta \frac{\partial E}{\partial w_{kj}} = -\eta \frac{\partial E}{\partial v_k} \frac{\partial v_k}{\partial w_{kj}}$$

$$\delta_k = -\frac{\partial E}{\partial v_k}$$

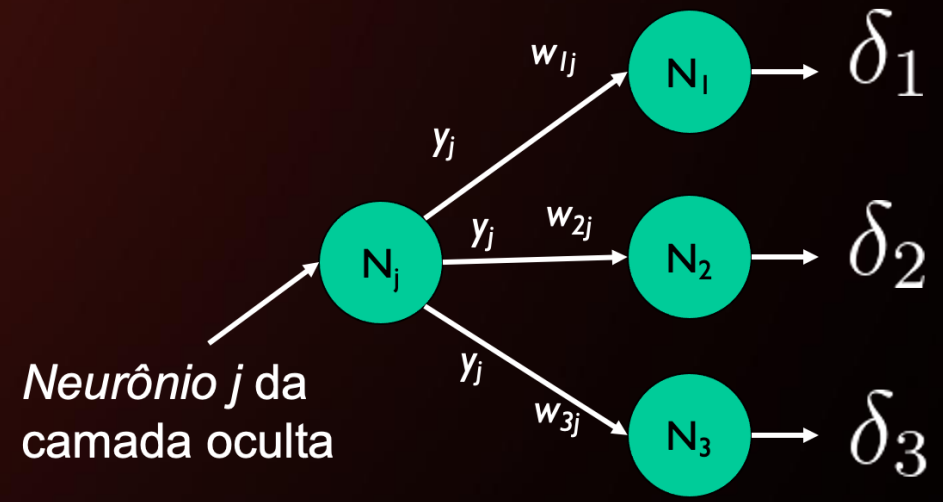
$$v_k = \sum w_{kj} x_j$$

$$\frac{\partial v_k}{\partial w_{kj}} = x_j$$

$$\Delta w_{kj} = \eta \delta_k x_j$$

ATUALIZAÇÃO

Temos: $\Delta w_{kj} = \eta \delta_k x_j$



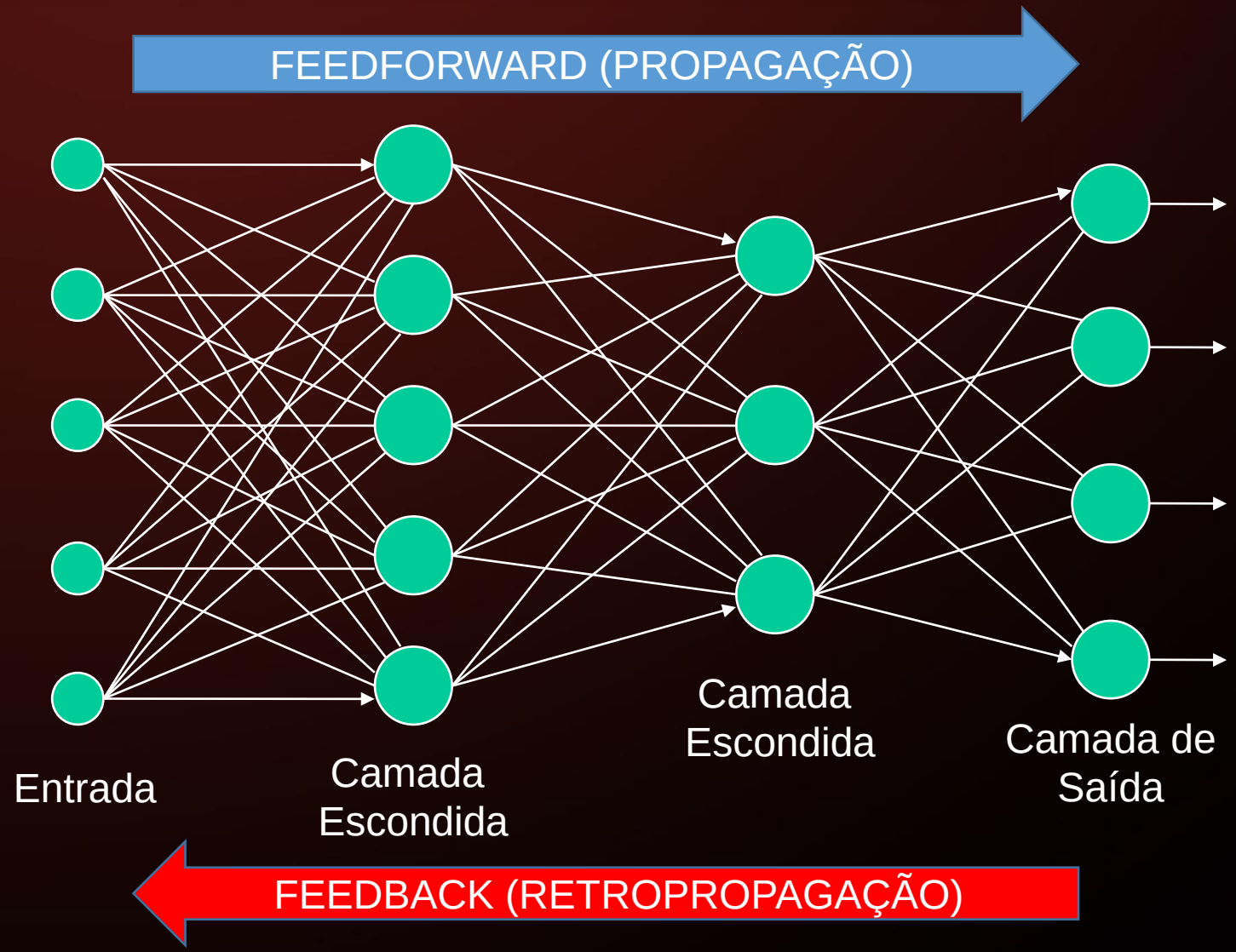
Para k sendo um neurônio de saída:

$$\Delta w_{kj} = \eta f'(v_k) e_k x_j$$

Para os neurônios ocultos:

$$\Delta w_{ji} = \eta f'(v_j) \left(\sum_k w_{kj} \delta_k \right) x_i$$

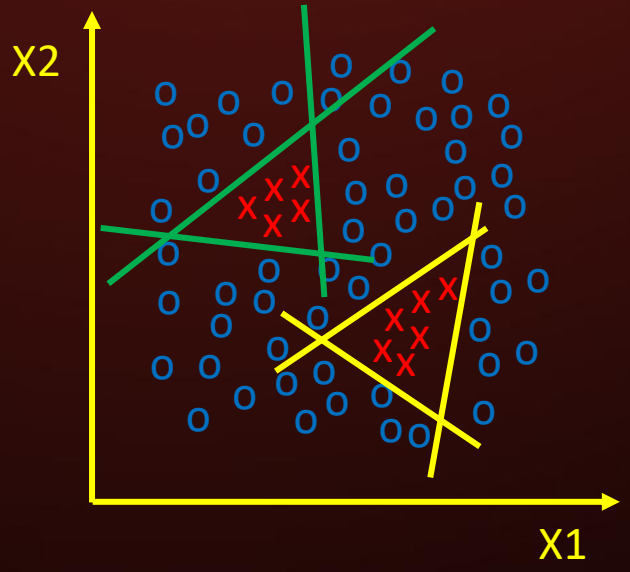
A RETROPROPAGAÇÃO

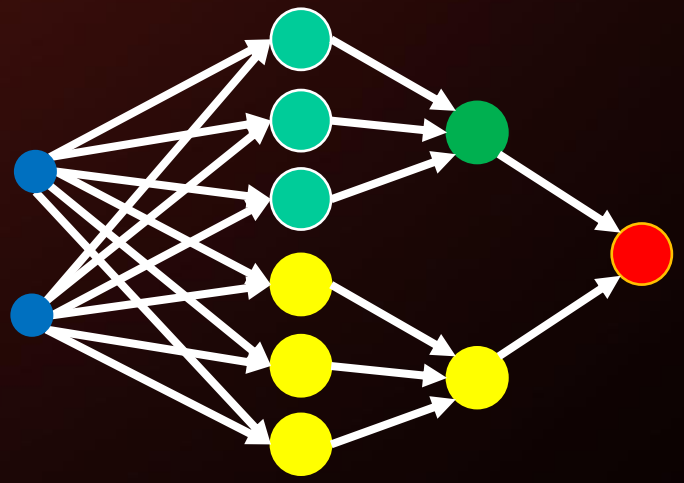


Hiperparâmetros, Regularização e Algoritmos Eficientes

HIPERPARÂMETROS DA MLP

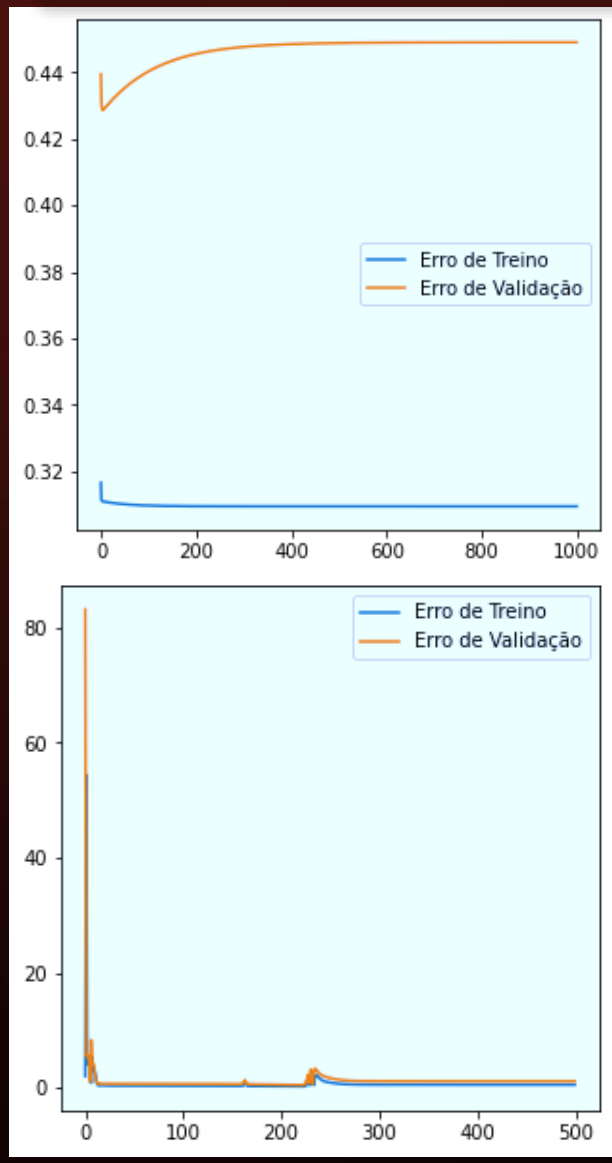
Qual topologia de rede MLP resolve este problema?

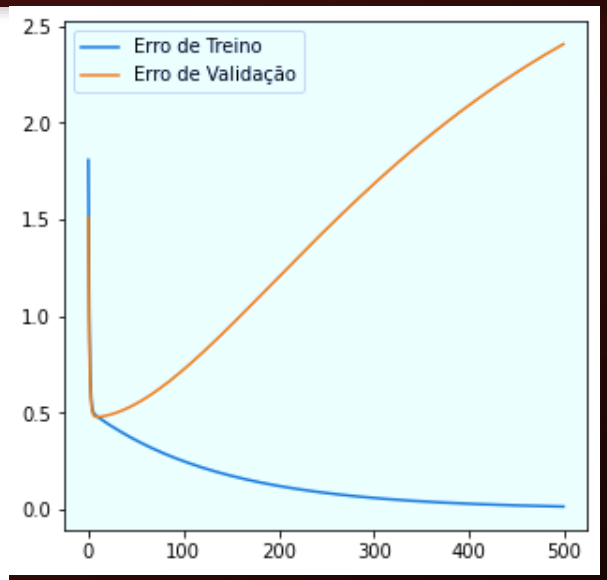


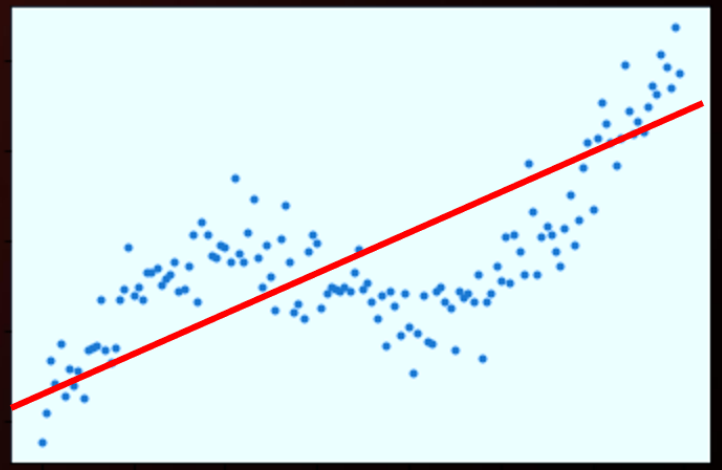


Qual é o problema desta abordagem?

HIPERPARÂMETROS DA MLP







Underfitting

Overfitting

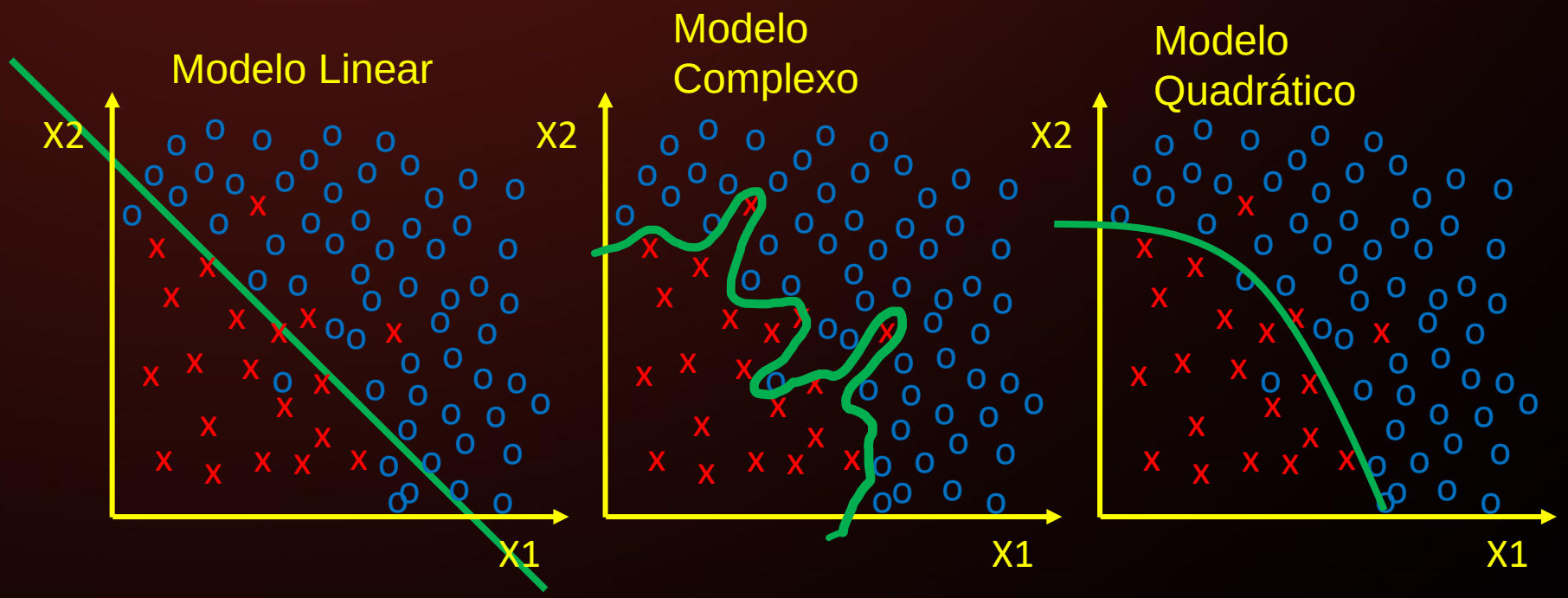
Adequado

BIAS VS VARIÂNCIA

Conjunto de Treino

Conjunto de Validação

Recordando



Alto viés - Underfitting

Alta variância Overfitting

Modelo adequado

BIAS VS VARIÂNCIA

Conjunto de Treino

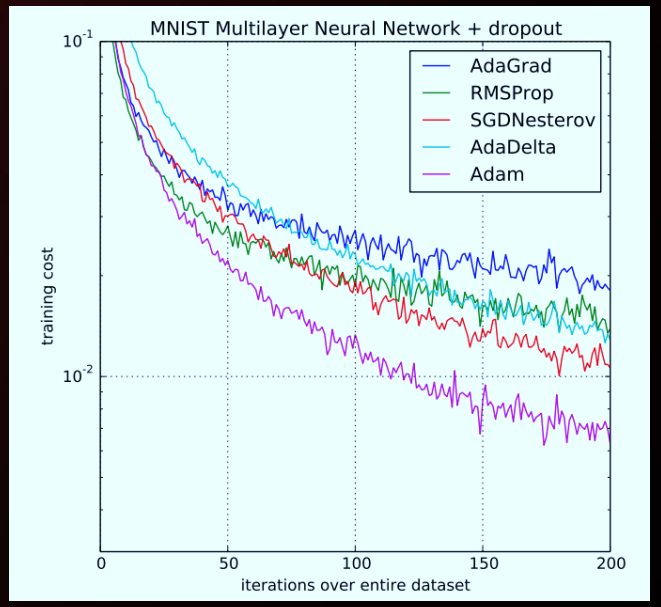
Conjunto de Validação

- Ajuste do Bias – Underfitting (treino)
  - Mais parâmetros, treinar por mais tempo
- Ajuste da Variância – Overfitting (validação)
  - Obter mais dados
  - Regularização / ajuste na topologia

$$\text{L2: } E = E_0 + \frac{\lambda}{2n} \sum_w w^2$$

- Buscamos pela menor rede capaz de resolver o problema de forma adequada

ACELERANDO O TREINAMENTO



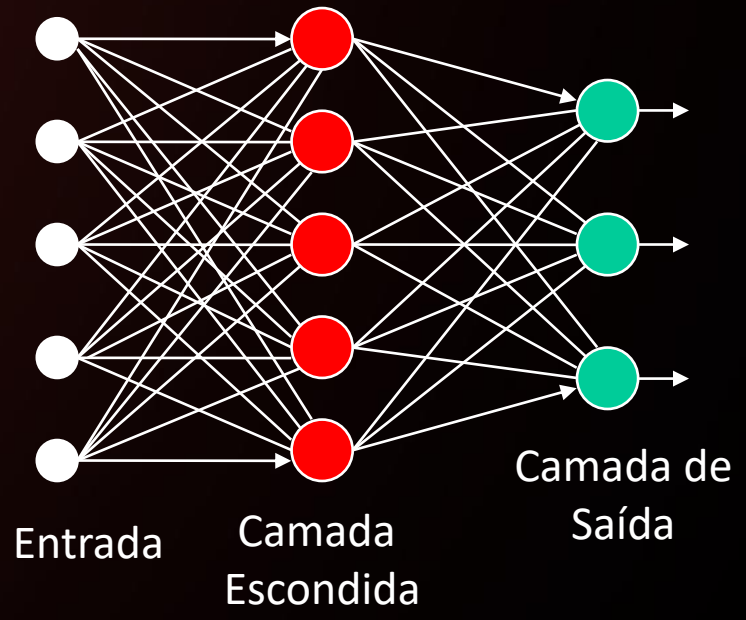
- Termo de Momentum
- Normalização dos dados
- Taxa treinamento decrescente
- Algoritmos mais eficientes: e.g., Adam

Adam: $\delta w = -\frac{\eta}{\sqrt{s_w}} M_w$

Redes de Função de Base Radial (RBF)

REDE RBF

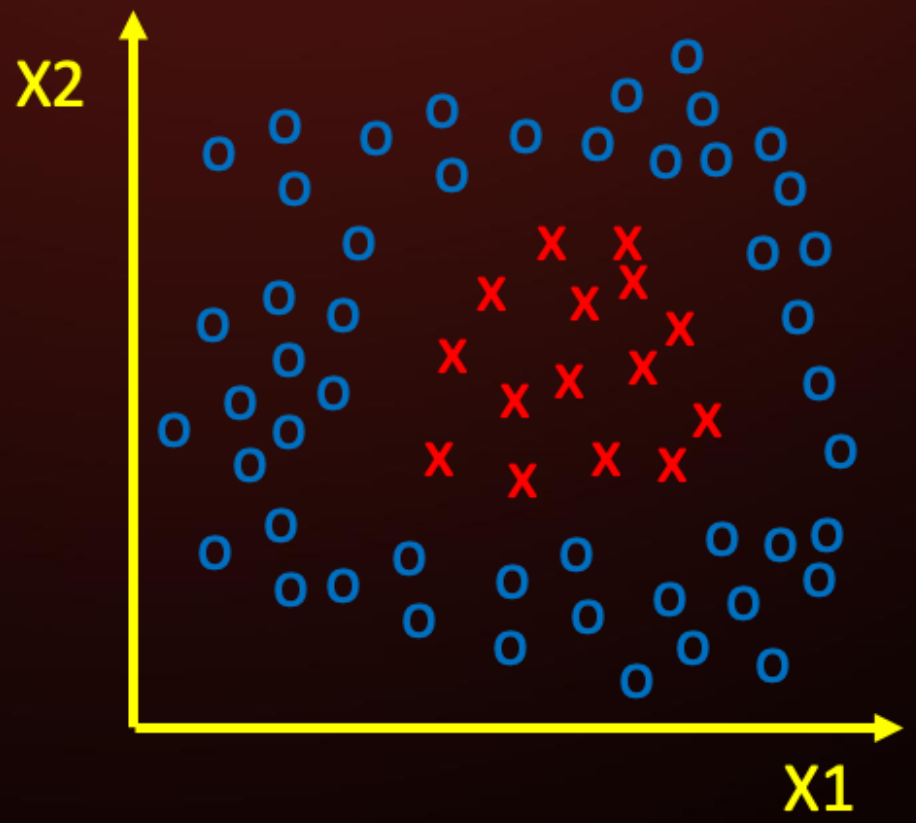
$$F(x) = \sum_{i=1}^{N} w_i \varphi\left(\|\mathbf{x} - \mathbf{x_i}\|\right)$$



1. **Comumente, com duas camadas ajustáveis:**
    1. **Oculta: Neurônios de bases radiais**
    2. **Saída: Neurônios lineares**
2. **Três formas de treinamento**
    1. **Centros fixos selecionados ao acaso**
    2. **Seleção auto-organizada dos centros (híbrida)**
    3. **Ajuste supervisionado dos centros**

REDE RBF

Qual rede pode resolver o problema abaixo?

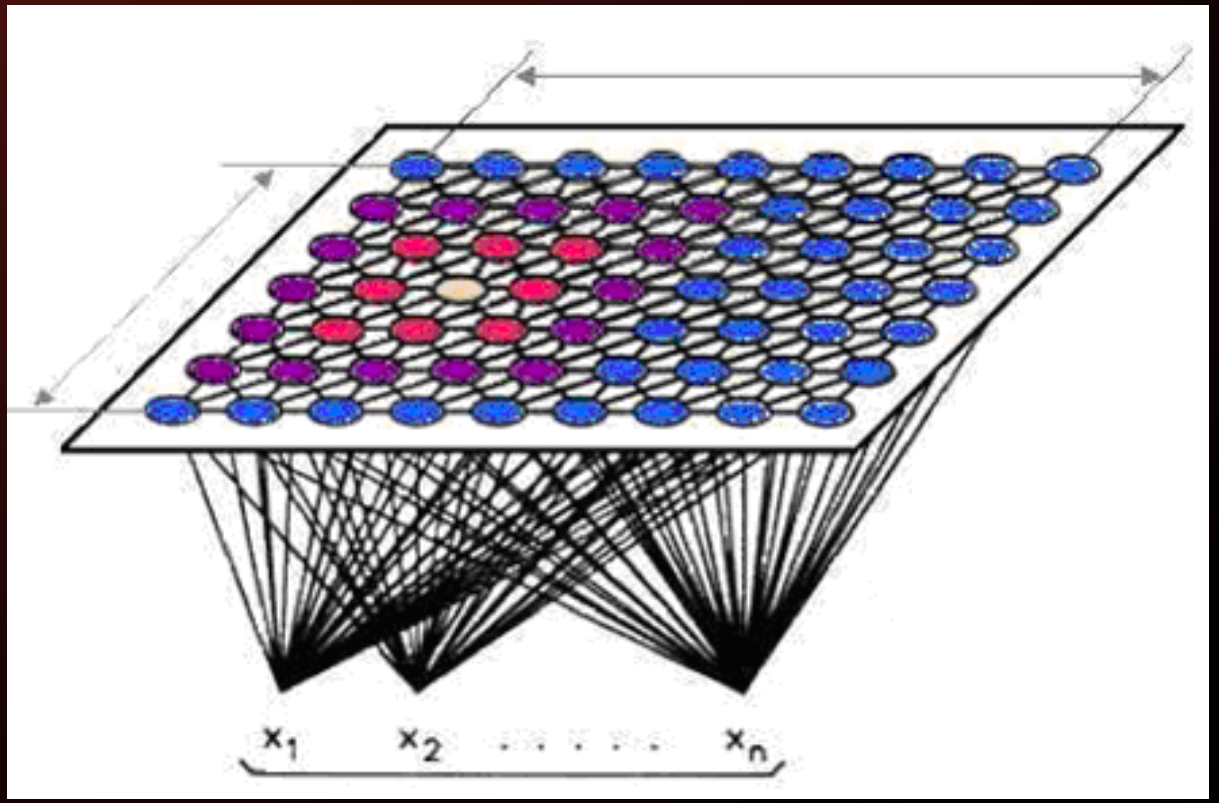


A rede Self-Organizing Maps (SOM)

SELF-ORGANIZING MAPS (SOM)

- **Normalmente é formada por um grid 2D**

- Forte inspiração neurofisiológica
- Ordenação topológica dos exemplos

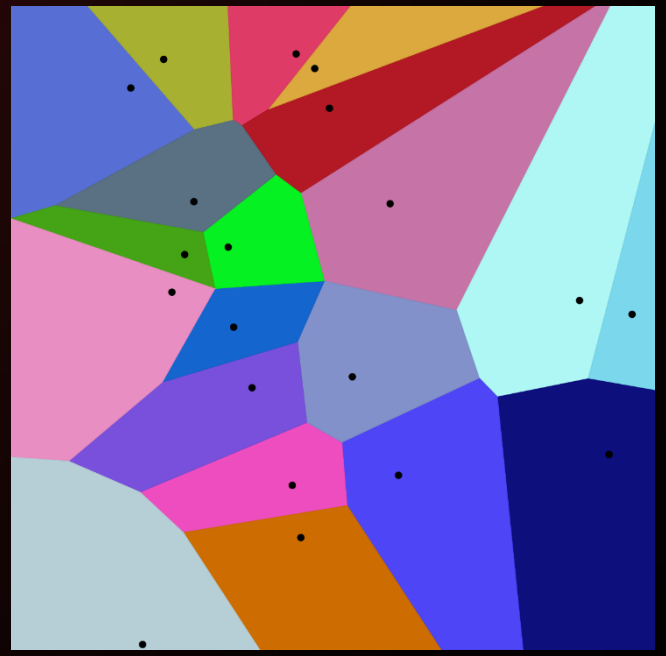


Fonte: https://www.researchgate.net/figure/Kohonen-topological-map_fig8_304161914

FORMAÇÃO DO MAPA

- Dada uma amostra x do espaço de entrada representando um padrão de ativação aplicado à rede, três processos estarão envolvidos na formação do mapa auto-organizável:
  - Competição
  - Cooperação
  - Adaptação Sináptica

REPRESENTAÇÃO DA SOM

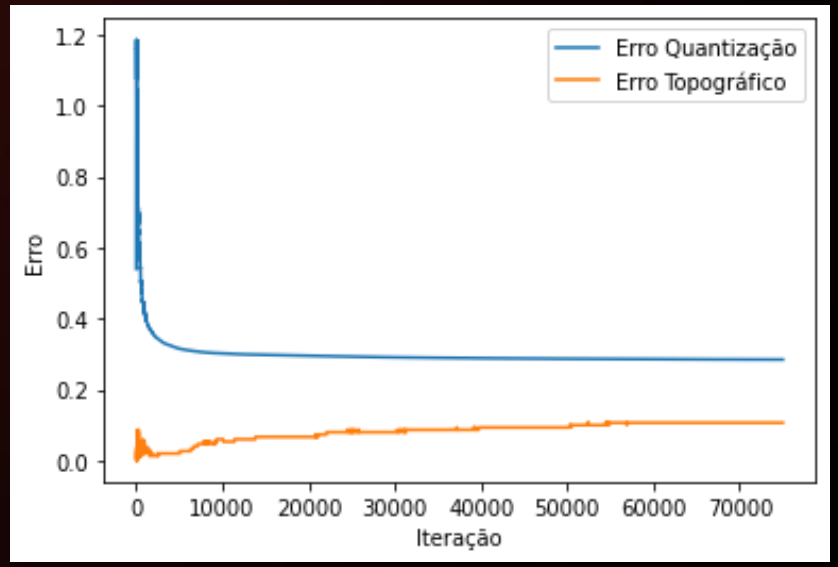


- Cada neurônio representa uma célula de Voronoi
- O vetor de pesos (neurônio) representa um protótipo de sua região
- Representa uma visualização 2D de um espaço R^m
- Neurônios próximos no grid tendem a representar padrões similares

Fonte: https://en.wikipedia.org/wiki/Voronoi_diagram

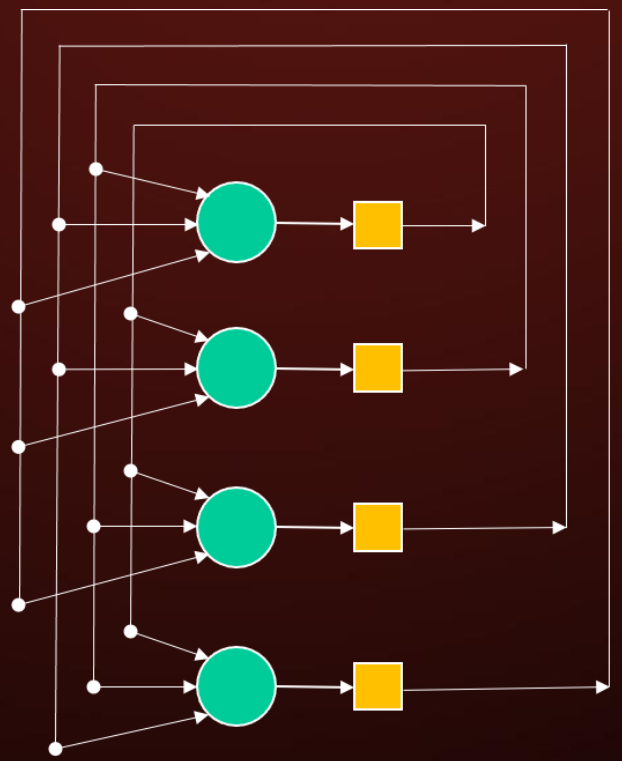
SOBRE O MAPA

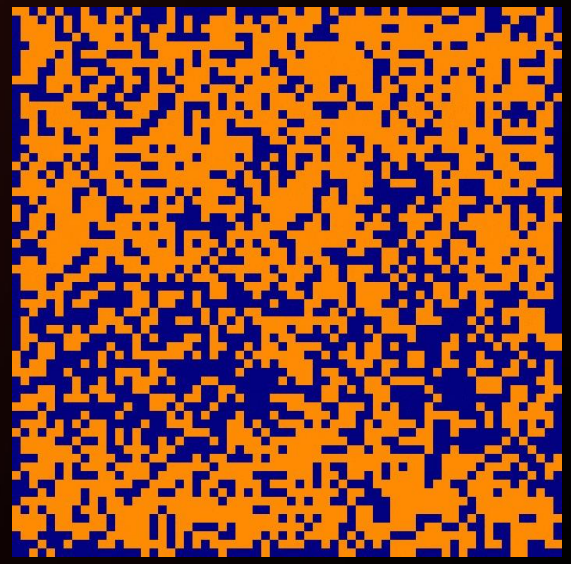
- Como configurar o tamanho do mapa?
- Como avaliar o mapa?
  - Com ou sem supervisão?
- No caso não-supervisionado:
  - Erro de Quantização:
  - Erro Topográfico
  - Inspeção visual:
    - Heat maps
    - Hit maps
    - U-Matrix



Redes Baseadas em Energia
Hopfield, Boltzmann e RBM

REDE DE HOPFIELD





Redes realimentadas de camada única com realimentação global

- **Neurônio Binário**
- **Treinamento: Encontrar pesos para armazenar os estados fundamentais**

$$E = -\sum_{i}\sum_{j \neq i} x_i x_j w_{ij}$$

$$\Delta E_i = \sum_{j} x_j w_{ij}$$

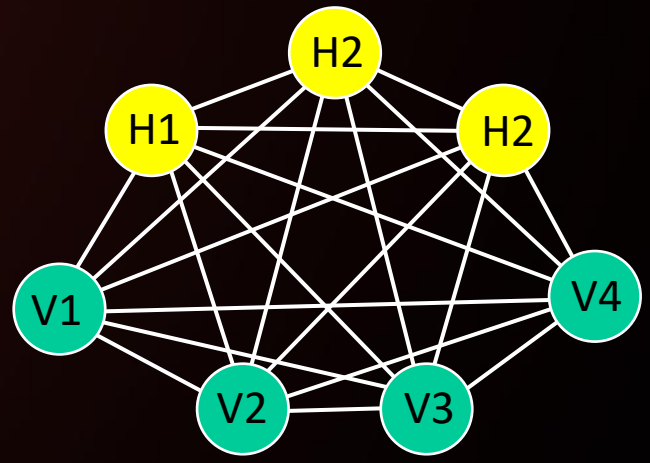
EVOLUÇÃO DOS ESTADOS

ILUSTRAÇÃO:

$$x_i = \begin{cases} 1 \text{ se } \sum_j x_j w_{ij} > 0 \\ -1 \text{ se } \sum_j x_j w_{ij} < 0 \end{cases}$$

- **LIMITAÇÕES:**
  - **Capacidade de Memória**
  - **Estados Espúrios**

A MÁQUINA DE BOLTZMANN



- É máquina similar a rede de Hopfield, porém, com unidades estocásticas

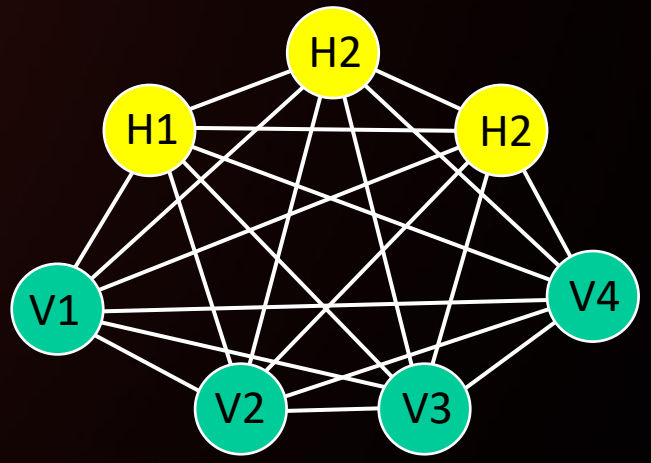
$$P(v) = \frac{1}{1 + exp(-v/T)}$$

- Além dos neurônios visíveis, a máquina também possui um grupo de neurônios ocultos
- O estado do sistema é definido com base na energia da rede

$$p(v, h) \propto \exp(-E(v, h))$$

- O estado de cada neurônio depende do gap de energia, ao longo da evolução, a rede irá convergir para um estado de menor energia

A MÁQUINA DE BOLTZMANN



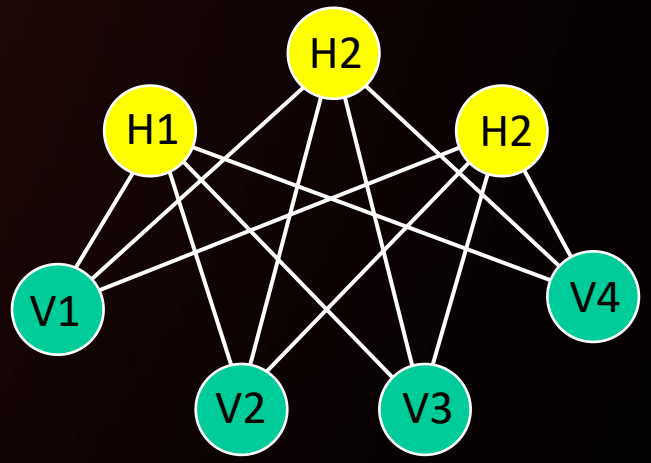
➢ OS NEURÔNIOS

- Neurônios visíveis representam os padrões
- Neurônios ocultos são extratores de padrões

➢ A MÁQUINA OPERA EM DUAS FASES DISTINTAS:

- Fase positiva (presa): neurônios visíveis permanecem fixos (representando um padrão do conjunto de treinamento)
- Fase negativa (livre): todos os neurônios operam livremente

A MÁQUINA DE BOLTZMANN



- É representada por um grafo completo, todos os neurônios estão conectados entre si
- Logo, o estado de cada neurônio depende dos estados de todos os demais neurônios da rede
- O processo para atingir o equilíbrio térmico em ambas as fases é muito custoso
- Podemos restringir a conectividade da rede para melhorar o processo de inferência e o aprendizado da rede
  - Máquina Restrita de Boltzmann (RBM)
  - Exclusão de conexões intracamada

CONSIDERAÇÕES SOBRE A RBM

1. A evolução é mais simples v→h, h→v
2. O custo da fase positiva é baixo (1 passo)
3. Porém, o custo da fase negativa é muito alto, limitando a aplicação do modelo em cenários reais

$$\Delta w_{ij} = \eta \left[(y_i y_j)^0 - (y_i y_j)^\infty \right]$$

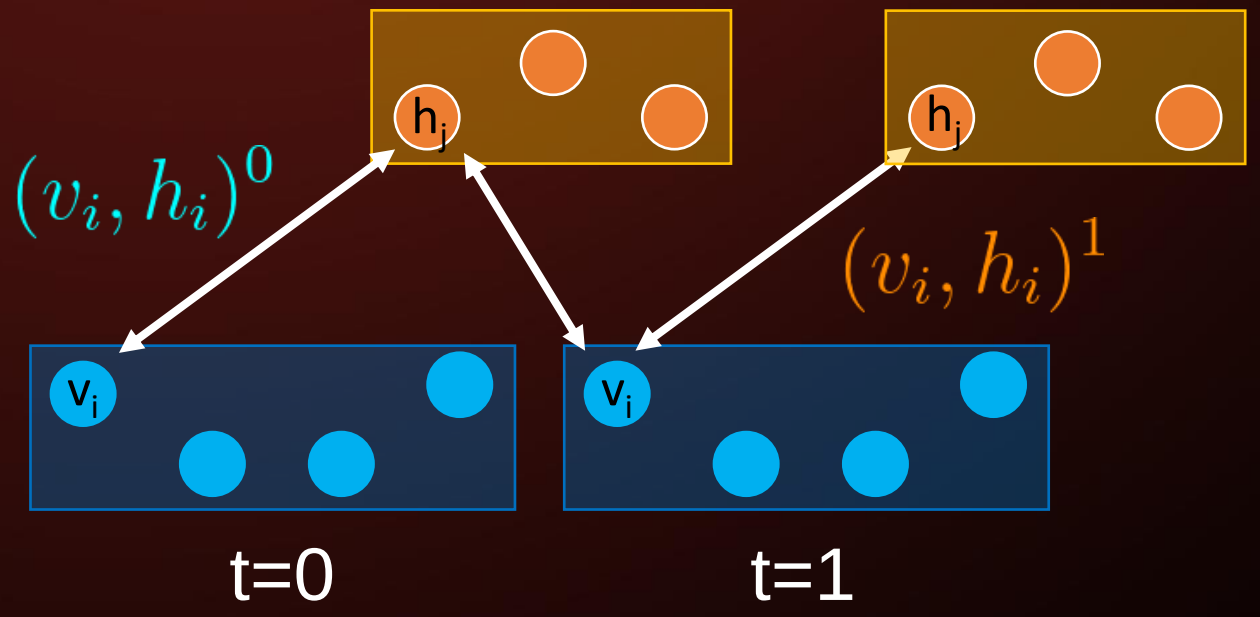
DIVERGÊNCIA CONTRASTIVA

QUATRO PASSOS:

1. **A camada visível é fixada com um padrão da base de treino**
2. **Atualiza os estados das unidades ocultas e calcula as correlações do passo 0 (positivo)**
3. **Atualiza os estados das unidades visíveis (reconstrução)**
4. **Atualiza os estados das unidades ocultas novamente e calcula as correlações do passo 1 (estimativa da fase negativa)**

$$\Delta w_{ij} = \eta \left[(y_i y_j)^0 - (y_i y_j)^1 \right]$$

DIVERGÊNCIA CONTRASTIVA



$$\Delta w_{ij} = \eta \left[(y_i y_j)^0 - (y_i y_j)^1 \right]$$

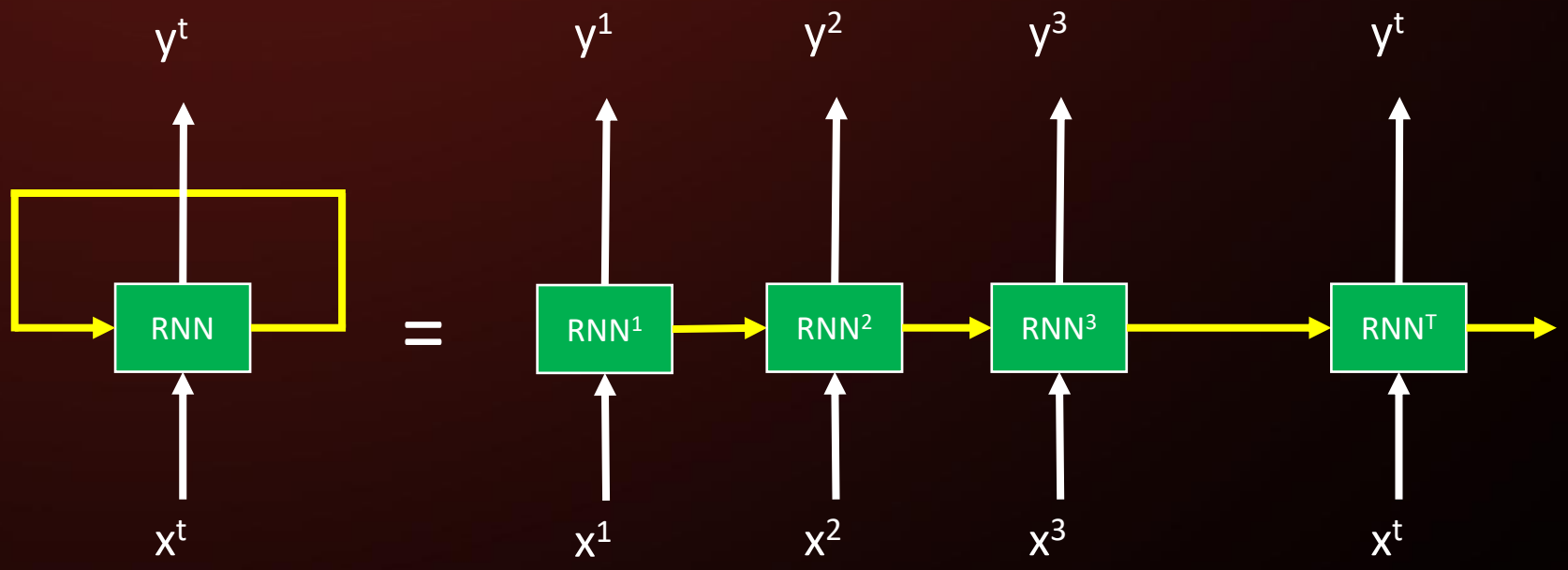
Redes Recorrentes
RNN, GRU, LSTM

REDES RECORRENTES

- O novo estado da rede depende tanto da entrada quanto do estado atual: memória de curto prazo associada ao estado interno
- Alguns modelos incorporam memórias de longo prazo: i.e. GRU, LSTM
- As RNNs podem se recordar de características importantes observadas nos sinais anteriores
- Bastante utilizada para tratamento de sinais sequenciais: séries temporais, textos, dados financeiros, sinais de áudio etc.
- RNN são máquinas universais

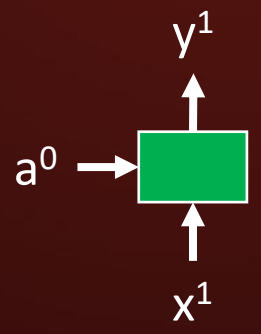
REDES RECORRENTES

Desenrolar a rede no tempo: *Unroll*

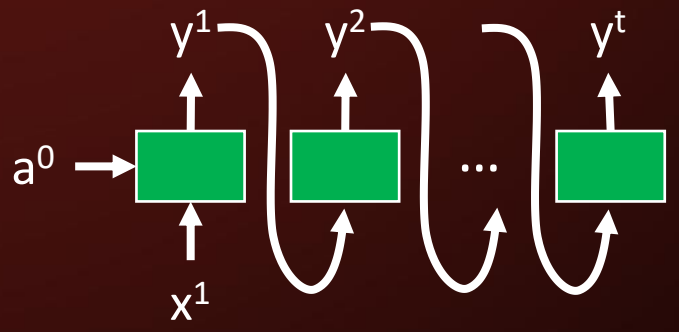


Problema: ao replicar o mesmo neurônio no tempo, podemos enfrentar o problema do desaparecimento ou explosão do gradiente

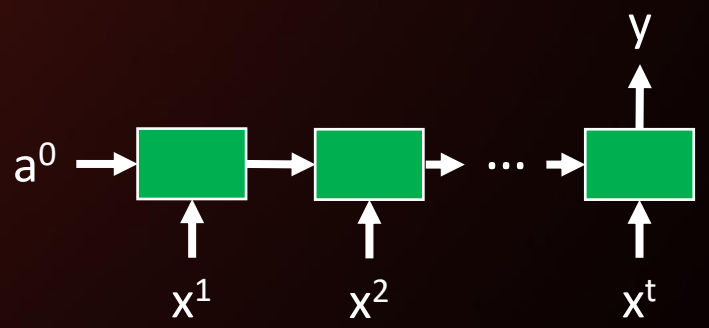
PRINCIPAIS ARQUITETURAS



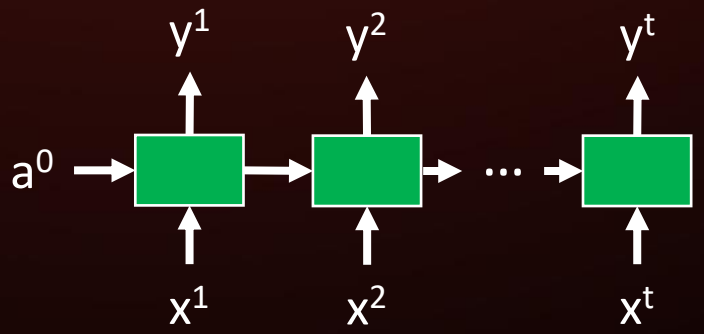
Um-para-um



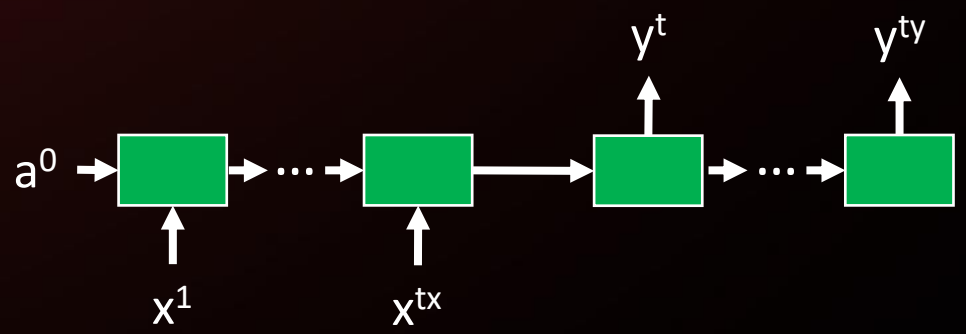
Um-para-muitos



Muitos-para-um



Muitos-para-muitos



Muitos-para-muitos

LIMITAÇÕES DAS RNNS

- Desaparecimento ou explosão do gradiente
- Sequências longas podem possuir dependências longas, i.e.:
  - “Os alunos, que estudam na Universidade Virtual do Estado de São Paulo, possuem grande competência.”
- As RNNs tradicionais possuem apenas memória de curto prazo, impossibilitando aprendizado de longas dependências
- Na prática: capacidade semelhante às redes MLP janeladas

LONG-SHORT TERM MEMORY

- Proposta em 1997, pelos professores Sepp Hochreiter e Jürgen Schmidhuber
- Ampliada por Felix Gers, em 2000
  - Inclusão da porta de esquecimento

Fontes: https://de.wikipedia.org/wiki/Sepp_Hochreiter
https://www.brainpreservation.org/team/juergen-schmidhuber/

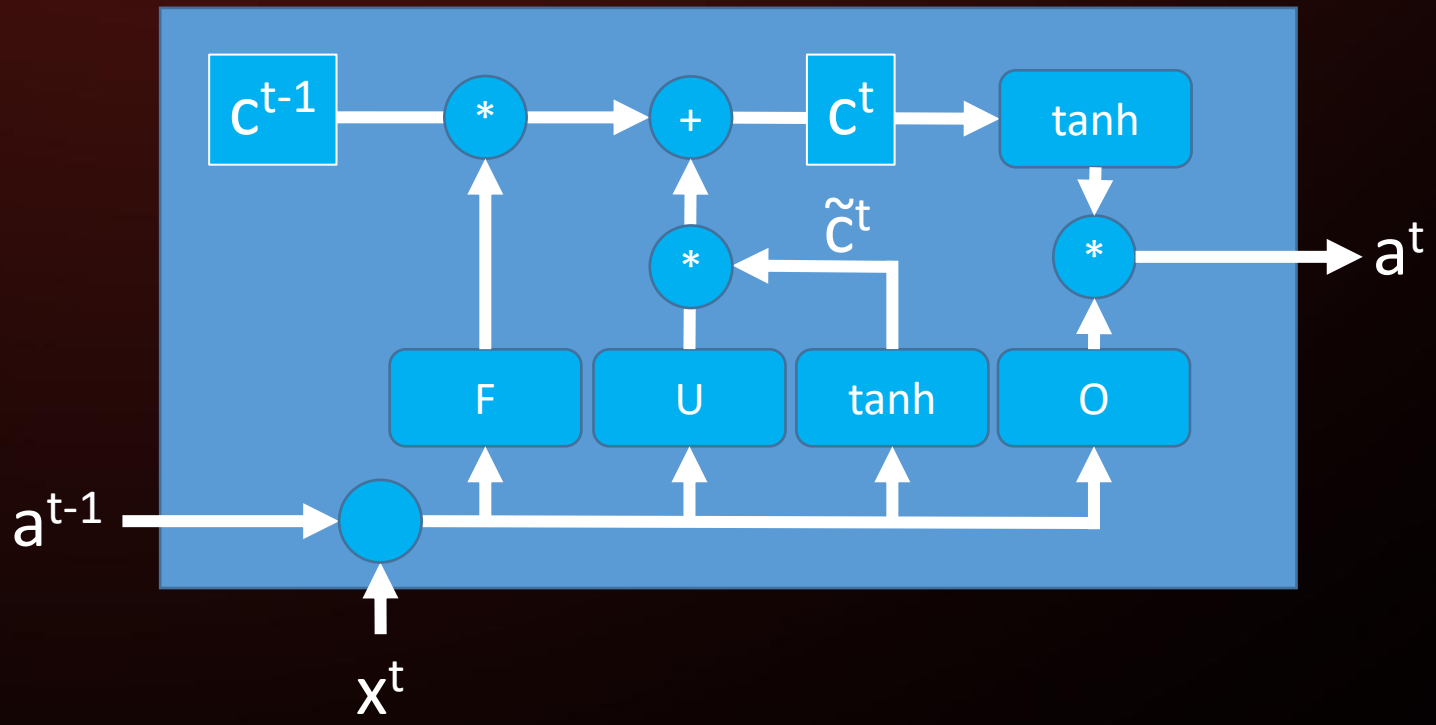
LONG-SHORT TERM MEMORY

- Uma célula LSTM é composta por três *gates*: entrada, saída e esquecimento (*forget*)
  - Gate de entrada (ou de atualização): Gate U
  - Gate de saída: Gate O
  - Gate de esquecimento: Gate F
- A célula é capaz de recordar sinais arbitrários do passado a partir da configuração dos *gates* – controle do fluxo de informação
- Teoricamente, sinais podem ser mantidos por longos períodos

LONG-SHORT TERM MEMORY

A célula LSTM possui dois sinais de memória

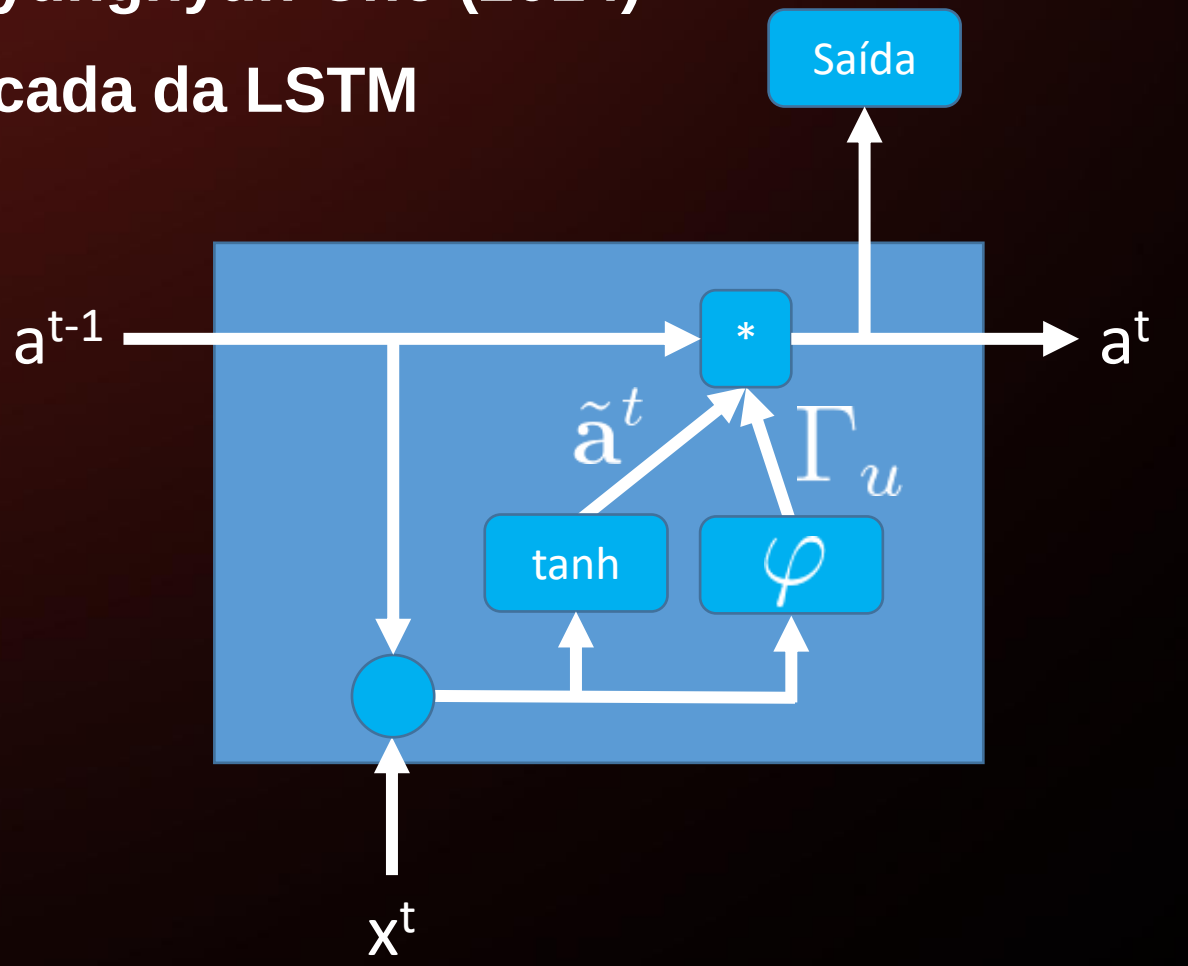
- **c^t – Sinal interno (estado da célula)**
- **a^t – Sinal externo (sinal de saída)**



GATED RECURRENT UNIT - GRU

- Proposta por Kyunghyun Cho (2014)
- Versão Simplificada da LSTM
- Duas versões

Fonte:
https://cims.nyu.edu/people/profiles/CHO_Kyunghyun.html



GRU x LSTM

GRU

- Um (ou dois) *gates*
- Duas (ou três) matrizes de pesos
- Apenas um sinal interno

LSTM

- Três *gates*
- Quatro matrizes de pesos
- Dois sinais internos (a e c)

Por possuir mais parâmetros, a LSTM tem mais flexibilidade. Na prática, os resultados são similares.

Parte II
Dúvidas?

O que o gradiente descendente faz?

Como selecionar o paradigma de aprendizagem adequado ao problema?

Qual é a importância dos conjuntos de treino e validação? Há problemas com essa abordagem? Como resolver?

Para que serve a normalização?

Quais são as limitações do Perceptron? Como as rede MLP resolvem essa limitação?

O que acontece se utilizarmos funções de ativação linear em redes MLP?

O que cada neurônio de uma rede MLP representa?

Qual é o papel de um neurônio de saída nas redes MLP?

Como definir a topologia (hiperparâmetros de uma rede MLP)?

O que é e como podemos resolver o underfitting/overfitting?

Qual é a função da regularização?

O que os neurônios de uma rede SOM representam?

Como definimos e treinamos uma rede SOM?

Como podemos avaliar o resultado de uma rede SOM?

Como podemos treinar uma rede RBF?

Qual é a diferença entre um neurônio de base radial e um MCP?

Como é definido o estado nos neurônios na rede de Hopfield?

O que é armazenado na rede de Hopfield?

Como é definido o estado nos neurônios na máquina de Boltzmann?

O que muda da máquina de Boltzmann para a sua versão RBF?

O que é uma rede recorrente?

Por que o problema do desaparecimento do gradiente é mais evidente nas redes recorrentes?

Por que as células GRU e LSTM podem resolver esse problema?